Discriminative Preprocessing of Speech: Towards Improving Biometric Authentication

Dissertation
zur Erlangung des akademischen Grades eines
Doktors der Philosophie
der Philosophischen Fakultäten
der Universität des Saarlandes
vorgelegt von

Dalei Wu

aus Han Dan, China

Saarbrücken, 2006

Die Dekanin: Univ.Prof. Dr. Ulrike Demske

Berichterstatter: Prof. Dr. William Barry
Prof. Dr. Dietrich Klakow
Prof. Dr. Jacques Koreman

Tag der letzten Prüfungsleistung: 21.07.2006

The dissertation of Dalei Wu

is approved and is acceptable in quality

and form for publication on microfilm:

Committee Chair

Saarland University, Saarbrücken

2006

Dedication

To

my wife and my daughter

for their endless love, patience and encouragement.

Also

to my parents and brother,

without their support, I could never have succeeded.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Acknowledgements

It is an exploratory journey of the mind to write a thesis. Many interesting discoveries on the way to the final draft are made by deviating from the straight path and topic of the thesis. At the same time, you know that there is a path, journey, and goal lying ahead of you - a thought which ensures that you do not go to far astray. In the course of this academic journey, many people have contributed ideas and comments to the present thesis. I would like to thank all of them for making it a wonderful and enriching enterprise.

First and foremost I wish to thank the chairman of the committee and one of my supervisors Prof. Barry. He always shares his strongest sense of humour with everyone, also including me, which skilfully changes the hard work into the much easier and happier one. Each time I meet a problem and go to discuss it with him, he always shed great lights on the problem by his profound thoughts and knowledge across multiple disciplines. It will be beneficial to me throughout all my life to learn from him.

Many thanks go also to my second supervisor, Prof. Jacques Koreman for his unwavering support, advice and patience. He gave me so much help that I could never forget. When I first arrived at Saarbrücken, It was him to pick up my family and me from the railway station in a very late evening. Later, he arranged everything I needed so well that I was able to be fully involved in my research work immediately. More important, he taught me the techniques and skills in almost every aspect which I need now for my research and later for my academic career. No matter how much gratitude I give him, it would never too much for him.

Further thanks go to my third supervisor Dr. Andrew Morris. He always gives me a lot of help on research direction and working attitudes. He has a solid background on mathematics which I can learn so much from him. He is a professional researcher in the domain of speech processing, which is definitely my future career I would build on. By learning from him, I can grow up more and more quickly, as he has set up such a good example for me. The only left thing for me is just to follow him up.

I also wish to thank Prof. Dietrich Klakow's group for their many kind and nice discussions and help. This work has certainly profited from their discussions. The cooperation with his group was very pleasant. Thanks so much for it.

Finally, I would like to thank my previous roommate Jürgen Trouvain, my colleagues, Manfred Pützer, Debbie Koreman, Bistra Andreeva, Lansun Chen and a lot of other people for their helping me so much and bringing me such a joyful working environment. I own too much to all of them. I am particularly indebted to Dr. Patricia Mueller-Liu for her very kind help to check the manuscript.

This work was supported by the SecurePhone project. I sincerely appreciate its support to my research and work.

Zusamenfassung

Im Rahmen des „SecurePhone-Projektes“ wurde ein multimodales System zur Benutzerauthentifizierung entwickelt, das auf ein PDA implementiert wurde. Bei der vollzogenen Erweiterung dieses Systems wurde der Möglichkeit nachgegangen, die Benutzerauthentifizierung durch eine auf biometrischen Parametern (E.: „feature enhancement“) basierende Unterscheidung zwischen Sprechern sowie durch eine Kombination mehrerer Parameter zu verbessern.

In der vorliegenden Dissertation wird ein allgemeines Bezugssystem zur Verbesserung der Parameter präsentiert, das ein mehrschichtiges neuronales Netz (E.: „MLP: multilayer perceptron“) benutzt, um zu einer optimalen Sprecherdiskrimination zu gelangen.

In einem *ersten* Schritt wird beim Trainieren des MLPs eine Teilmenge der Sprecher (Sprecherbasis) berücksichtigt, um die zugrundeliegenden Charakteristika des vorhandenen akustischen Parameterraums darzustellen.

Am Ende eines *zweiten* Schrittes steht die Erkenntnis, dass die Größe der verwendeten Sprecherbasis die Leistungsfähigkeit eines Sprechererkennungssystems entscheidend beeinflussen kann.

Ein *dritter* Schritt führt zur Feststellung, dass sich die Selektion der Sprecherbasis ebenfalls auf die Leistungsfähigkeit des Systems auswirken kann. Aufgrund dieser Beobachtung wird eine automatische Selektionsmethode für die Sprecher auf der Basis des maximalen Durchschnittswertes der Zwischenklassenvariation (between-class variance) vorgeschlagen. Unter Rückgriff auf verschiedene sprachliche Produktionssituationen (Sprachproduktion mit und ohne Hintergrundgeräusche; Sprachproduktion beim Telefonieren) wird gezeigt, dass diese Methode die Leistungsfähigkeit des Erkennungssystems verbessern kann.

Auf der Grundlage dieser Ergebnisse wird erwartet, dass sich die hier für die Sprechererkennung verwendete Methode auch für andere biometrische Modalitäten als sinnvoll erweist.

Zusätzlich wird in der vorliegenden Dissertation eine alternative Parameterrepräsentation vorgeschlagen, die aus der sog. „Sprecher-Stimme-Signatur“ (E.: „SVS: speaker voice signature“) abgeleitet wird. Die SVS besteht aus Trajektorien in einem Kohonennetz (E.: „SOM: self-organising map“), das den akustischen Raum repräsentiert. Als weiteres Ergebnis der Arbeit erweist sich diese Parameterrepräsentation als Ergänzung zu dem zugrundeliegenden Parameterset. Deshalb liegt eine Kombination beider Parametersets im Sinne einer Verbesserung der Leistungsfähigkeit des Erkennungssystems nahe.

Am Ende der Arbeit sind schließlich einige potentielle Erweiterungsmöglichkeiten zu den vorgestellten Methoden zu finden.

Schlüsselwörter: Feature Enhancement, MLP, SOM, Sprecher-Basis-Selektion, Sprechererkennung

Abstract

In the context of the SecurePhone project, a multimodal user authentication system was developed for implementation on a PDA. Extending this system, we investigate biometric feature enhancement and multi-feature fusion with the aim of improving user authentication accuracy.

In this dissertation, a general framework for feature enhancement is proposed which uses a multilayer perceptron (MLP) to achieve optimal speaker discrimination.

First, to train this MLP a subset of speakers (speaker basis) is used to represent the underlying characteristics of the given acoustic feature space.

Second, the size of the speaker basis is found to be among the crucial factors affecting the performance of a speaker recognition system.

Third, it is found that the selection of the speaker basis can also influence system performance. Based on this observation, an automatic speaker selection approach is proposed on the basis of the maximal average between-class variance. Tests in a variety of conditions, including clean and noisy as well as telephone speech, show that this approach can improve the performance of speaker recognition systems. This approach, which is applied here to feature enhancement for speaker recognition, can be expected to also be effective with other biometric modalities besides speech.

Further, an alternative feature representation is proposed in this dissertation, which is derived from what we call speaker voice signatures (SVS). These are trajectories in a Kohonen self organising map (SOM) which has been trained to represent the acoustic space. This feature representation is found to be somewhat complementary to the baseline feature set, suggesting that they can be fused to achieve improved performance in speaker recognition.

Finally, this dissertation finishes with a number of potential extensions of the proposed approaches.

Keywords: feature enhancement, MLP, SOM, speaker basis selection, speaker recognition, biometric, authentication, verification

Part I. General background: Speech and other biometrics

1. Introduction

Biometrics are measurements of an individual’s physical and/or behavioural characteristics which can be used to determine his or her identity. Typical physical biometrics are DNA, iris and fingerprint, while typical behavioural biometrics are gait, online signature and voice quality (see Figure 1-1). Among these biometrics some, such as DNA and iris pattern, are unchangeable and unique throughout a human life. Others, such as body shape, face and voice are more or less changeable as conditions change in dependency on age and health, but still contain a lot of information important for the differentiation of people’s identities. Biometric identity recognition uses pattern recognition techniques to determine a person’s identity on the basis of biometric measurements.

Biometric identity recognition is divided into two basic types of techniques: identification and verification (also known as authentication). Identification refers to the task of determining who a person is within a given group of candidates. Verification is the process of ascertaining a claimed identity. The verification decision is a binary one: the correctness of the claimed identity is either accepted or rejected. Note that throughout this dissertation we adhere to the widely adopted convention that the terms “verification” and “authentication” can be used interchangeably and that identification and verification may be referred to collectively with the term “recognition”.

Identification and verification have different applications. Whether identification or verification is adopted depends on the goal. To use a credit card system, a client biometric profile must be verified, i.e. it must be ascertained that the user is the claimed client. When several trials to gain access to a credit card system have failed, it may be useful to determine the user’s identity in case prosecution for fraud becomes necessary. An identification application could possibly be used in this case to select the impostor from a large database containing the biometric profiles of suspects.

Biometric recognition approaches are divided into two classes: unimodal and multimodal recognition (Figure 1-1). While unimodal recognition uses only one type of biometric information to ascertain an identity, multimodal recognition combines multiple modalities.

In applications using a highly reliable biometric like DNA a unimodal system is satisfactory. However, the evidence obtained from individual biometric modalities is often unreliable. In such a case, multimodal recognition is an alternative approach which combines several biometrics to provide stronger supporting evidence for recognition. Multimodal approaches represent an emerging trend whose importance may be expected to increase in future research on biometric recognition.

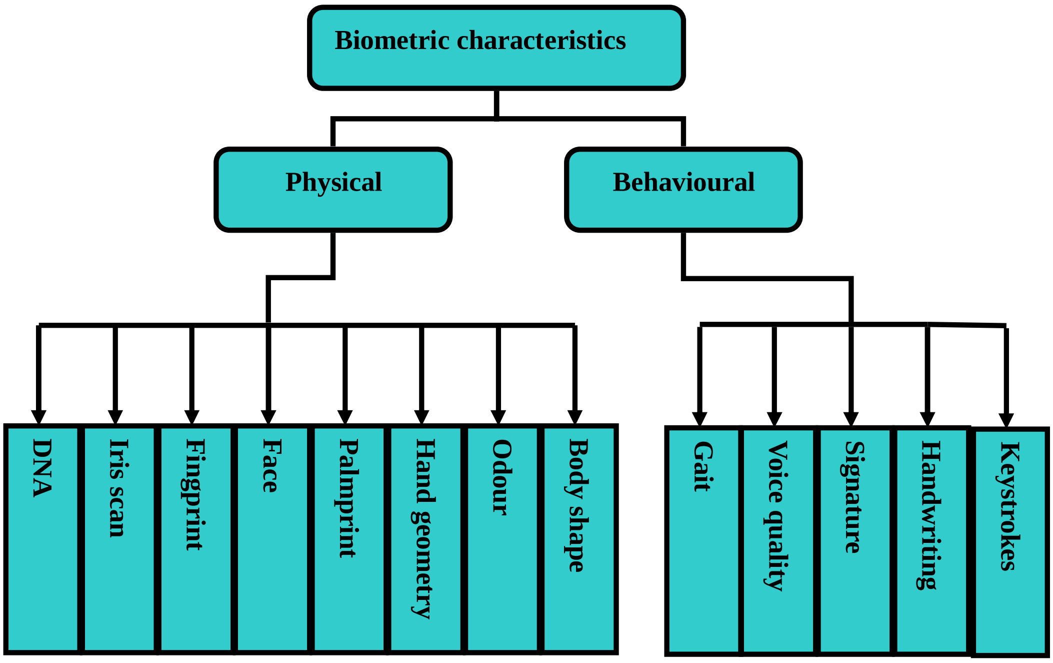


Figure 1-1: Biometric characteristics used for identity recognition

The remainder of Part I is organised as follows: In Chapter 2, the biometrics most commonly used for recognition are introduced. Following this, multimodal biometric authentication is overviewed in Chapter 3. In Chapter 4 the goals and structure of this dissertation are described.

2. Use of biometrics for identity recognition

2.1 Introduction

Biometric features may be physical or behavioural. The first are a direct reflection of a person's anatomical or physiological characteristics, while the latter are learnt or acquired over time and are therefore under the control of the individual. The most frequently used physical characteristics include DNA, iris, fingerprint, face, palm print, hand geometry, odour, body shape and dental radiography. Among the most commonly used behavioural characteristics are gait, signature, handwriting and voice quality.

All biometric-based recognition approaches consist of three basic processing steps: data acquisition, feature extraction and verification (including feature modelling and decision making). In the first step, biometric data are captured with special devices. In the second step, biometric features are extracted from the obtained data. In the third step, a biometric template or model is constructed and used to recognise an individual's identity. In the following subsections, a general description and an outline of these three processing steps are given for each of the commonly used biometrics.

2.2 Physical biometrics

Biometrics can be used for both identification and verification. However, as verification is more widely used, the following sections will focus on verification.

2.2.1 DNA

DNA (deoxyribonucleic acid) is present in every living cell and has a double helix structure (Figure 2-1). All the information required for the growth of an organism from a single cell is contained in its DNA. Humans have 23 homologous ("pairs of") chromosomes (a threadlike body in the cell nucleus that carries the genes in a linear order), resulting in a total of 46 chromosomes. It has been discovered that 99.7% of human DNA is shared by all human beings and that only 0.3% is variable across individuals. These variable regions, called "Short Tandem Repeats" (or STRs), may be examined to distinguish one person from another (Soltysiak et al. 2005; Commission 2005).

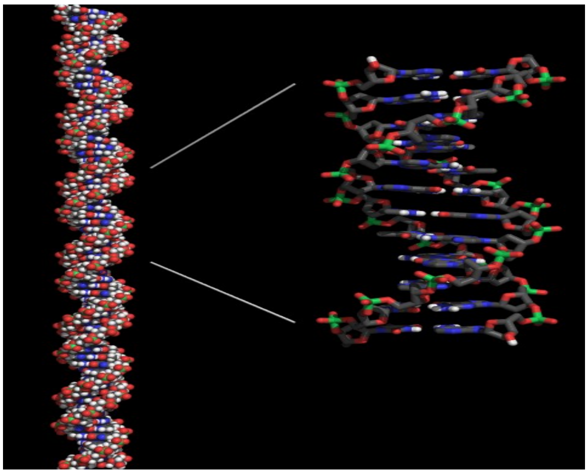


Figure 2-1: Expanded segment of a DNA helix, the right hand side an STR

Methods for DNA analysis and DNA profile generation are well established (Hashiyada et al. 2003; Yen 2004; Walsh et al. 1991, Hashiyada, 2004; Demidov et al. 2004).

Acquisition: In this first step of DNA extraction, DNA is obtained and isolated from the DNA sample. The sample is then amplified, i.e. multiple copies of the "target sequences" are created, as will be explained below.

Feature Extraction: An STR, also referred to as a "locus", is composed of a repeated DNA sequence that varies in length between individuals. A repeated unit is called an "allele". Each STR possesses 6 to 19 alleles (Hashiyada et al. 2003). The number of repeats at each location can be measured during DNA sequencing, one of the steps in DNA profiling. To generate the DNA profile, the number of repeats of these STRs are obtained and concatenated into a sequence of decimal numbers, this sequence then providing an accurate DNA profile. Soltysiak et al. (2005) computed the odds of two people sharing the same profile as about one in a trillion when 13 loci are used. Different numbers of STRs are used in different countries, depending on the law applied in each case. In Japan, for instance, 17 STR loci are used for paternity tests. The Japanese police force formally introduced 10 STR loci for crime investigation in August 2003 (Hashiyada, 2004). At present, the entire procedure takes between 2 and 3 hours (Demidov et al. 2004).

Verification: This step of DNA authentication is relatively simple, since only the decimal numbers are obtained after feature extraction. Therefore template matching can be applied efficiently.

2.2.2 Iris

The iris is the part of the human eyeball located between the pupil – a black-looking aperture which allows light to enter the eye – and the sclera – the white of the eye which forms part of the eyeball's supporting wall (Figure 2-2). It is a coloured circular muscle that controls the size of the pupil to allow different amounts of light to enter the eye depending on lighting conditions. It is pigmented, giving each eye its specific colour. The iris has characteristic patterns, described as ridges, flecks, crypts, pigmented dots and contraction furrows. Research has shown that these patterns can uniquely determine human identity. They are also associated with human personality (Coplan et al. 1998).

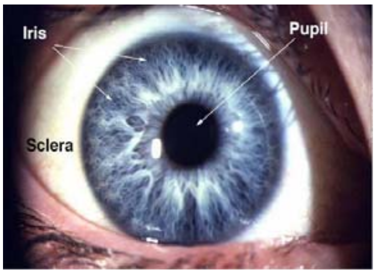


Figure 2-2: Human eye (the iris is shown between the pupil and the sclera) (Bio-tech-inc 2002)

The use of iris scans for human identity recognition was proposed in 1936 by ophthalmologist Frank Burch, but the method was not applied until the early 1990s when algorithms for iris recognition were developed (and patented) (Daugman 1993).

While many approaches have been proposed for iris recognition (Daugman 1993, 2003, 2004; Wildes 1997; Boles et al. 1998; Ma et al. 2003), the most widely used method today is that proposed by Daugman. As all other approaches are variants of this method, the following description will relate to Daugman's approach.

Acquisition: Using standard equipment, a person must be within 1 metre of a camera in order for it to obtain a clear image of his iris. If the eye is not positioned at a fixed distance, the digital camera used to capture iris patterns must have an auto-focus. An auto-focus adjustment algorithm was implemented by Daugman as part of his algorithms for iris recognition (Daugman 1993, 2003, 2004).

Feature extraction: Feature extraction comprises four main steps: iris separation, polar coordinate conversion, phase information extraction and iris code generation. In the first step, the inner and outer iris boundaries are detected in order to separate the iris ring from the pupil and sclera. For this, an exhaustive search is carried out for the circles which give maximal change of grey level (Daugman 1993, 2003, 2004). The whole iris scan is then partitioned into patches (e.g. 32x32 pixels) for further processing. In the second step, for each patch the pixel grid (*x, y*) of Cartesian coordinates is converted into a grid (r, θ) of polar coordinates. The isolated iris pattern is then demodulated to extract phase information using quadrature 2-*d* Gabor wavelets, which in fact consist of a pair of bits of either 0 or 1. Finally, all the bit pairs from all the patches are concatenated to generate a 2048-bit iris code (256 bytes).

Verification: Since each individual has a unique iris pattern, template matching works efficiently in iris recognition. The matching principle used in identity verification is based on the failure of a test of statistical independence of the phase structures of the iris patterns. The Hamming distance (HD) is calculated for a pair of iris codes. A HD less than a specific threshold indicates that the iris codes originate from the same source (Daugman 1993, 2003, 2004).

While this method has achieved a 0 error rate for successfully enrolled subjects, the rate of successful enrolment is only around 85% (Daugman 1993).

2.2.3 Fingerprint

The fingerprint is one of the earliest biometric characteristics used in forensic analysis. While fingerprints are highly individual, common features exist and several court cases are known in which people were mistakenly convicted on the basis of fingerprint evidence.

A fingerprint consists of a multitude of ridges and valleys. The ridge endings in fingerprints, which can be single or bifurcated, are generally referred to as minutiae. There are six basic patterns of minutiae used by a fingerprint recogniser: the arch, the tented arch, the right loop, the left loop, the whorl and the twin loop (Figure 2-3).

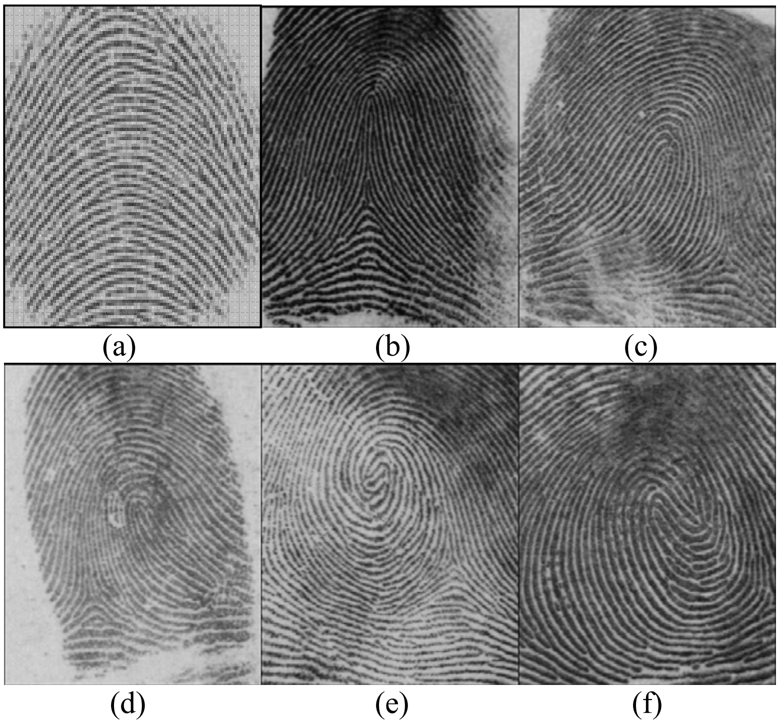


Figure 2-3: Basic patterns of fingerprint minutiae: (a) arch, (b) tented arch, (c) right loop, (d) left loop, (e) whorl and (d) twin loop (adapted from Jain et al. 2000)

Acquisition: Two primary methods exist for capturing a fingerprint image: the inked scan (off-line) and the live scan (inkless). An inked fingerprint is obtained from a person by putting some ink on his or her finger and pressing it on a sheet of paper. This inked fingerprint can be input into the computer with a document scanner. Live scan fingerprints are directly input into computers in digital format by means of a special peripheral device which can be designed either optically or electrically. The optical type is based on the concept of optically frustrated total internal reflection (FTIR). It uses the touching action on the plate to determine the positions of the ridges and valleys. Further details can be found in Jain et al. (2003) and Kawagoe et al. (1984). The electrical device uses a more recent technique, called capacitance-based solid state live-scan fingerprint sensing. This uses an array of electrodes to extract minutiae (Young et al. 1997). A variety of fingerprint sensing devices are sold on the market. Among the less expensive devices is the ThumbPod (Schaumont et al. 2005).

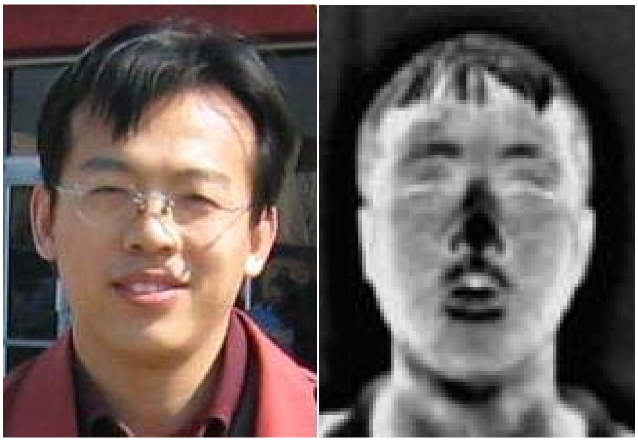
Feature extraction: Two different methods exist for carrying out fingerprint authentication. One is pattern-based (or image-based) and the other is minutiae-based. Pattern-based algorithms compare the basic fingerprint patterns of a previously stored template with those of a candidate fingerprint. As pixel-to-pixel comparison is used, no particular vector-based feature extraction is needed, this resulting in higher computational efficiency. However, one obvious drawback of this type of algorithm is that it requires images be accurately aligned and have the same orientation, making it vulnerable to noise and nonlinear distortion (Tico et al. 2001; Andrew et al. 2003). Minutiae-based approaches are more expensive to compute, since vector-based features such as ridge directional fields have to be estimated and extracted. On the other hand, this type of algorithm is highly robust to distortions such as rotation, translation and scaling and also robust to noise. The minutiae approach represents the most important direction in research activity today. Therefore, the following discussion will focus on this approach.

Ridge endings and ridge bifurcations (minutiae) from input fingerprint images are used as the discriminating features required for fingerprint authentication by (Jain et al. 2003). A reliable feature extraction algorithm consists of four components: orientation field estimation, ridge extraction, minutiae extraction and post-processing. In order to conveniently process it, a fingerprint image is partitioned into a number of non-overlapping blocks (e.g. 32x32 pixels). The underlying idea in orientation field estimation is to use an analysis of greyscale gradients in the processed block. Typical variants of the algorithm using pixel gradients are: averaging (Kavagoe et al. 1984), voting (Mehtre et al. 1989) and optimisation (Ratha et al. 1996). In ridge detection, the most common approach uses either simple or adaptive thresholding. Pixels can be identified as ridge pixels as long as the grey level values on the ridges attain their local maxima along a direction normal to the local ridge orientation. Furthermore, the ridges obtained are required to be thinned, using a standard thinning algorithm (Jain et al. 2000). In the stage of minutiae detection, a simple rule can be applied to detect minutiae: ridge pixels with three ridge pixel neighbours are identified as a ridge bifurcation, and those with one ridge pixel neighbour identified as ridge endings. Finally, the procedure of post-processing is applied to discard noise, spurious minutiae and extraneous minutiae using a number of heuristics (Jain et al. 2000). For example, too many minutiae in a small neighbourhood may indicate the presence of noise and are therefore discarded.

Verification: Recognition uses template matching. In this step identical procedures are followed for both image-based authentication and minutiae-based authentication.

2.2.4 Face

Face characteristics can be captured by both visual images (Figure 2-4a) and infrared images (Figure 2-4b). Infrared images record the levels of thermal radiation in the infrared spectrum range at 0.7-14.0μm, while visual images measure the electromagnetic energy in the visible spectrum range (0.4-0.7μm). As these two modalities are to some extent complementary, they can be usefully fused together for face recognition.



(a) (b)

Figure 2-4: Face image samples: (a) a visual image, (b) an infrared image (adapted from Kong et al. 2005)

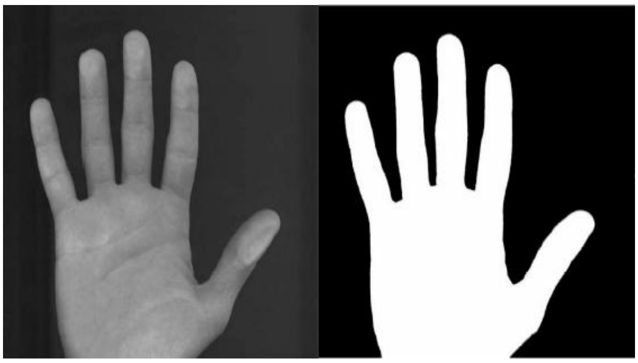
Acquisition: Automatic face recognition begins with the detection of the facial region and then proceeds to normalise it using information about the location and appearance of facial landmarks, such as eyes, nose and mouth. Motion and skin colour also provide useful clues for face detection. The required algorithms have been described by Wang et al. (1997), Satoh et al. (1999), Chai et al. (1999), Carcia et al. (1999), Hsu et al. (2002) and Wu et al. (1999). Eye detection plays an important role in face size normalisation and also facilitates the localisation of other facial landmarks. Most eye localisation methods are template-based approaches, but other approaches such as knowledge-based methods, feature invariant approaches and appearance-based methods have also been proposed. Details are found in Lam et al. (1996); Huang et al. (1998); Smeraldi et al. (2000), Kong et al. (2005).

Feature extraction: Feature extraction seeks information relevant to user-discriminating capacity from the detected face region. The performance of this process depends on the accuracy of face region detection and lighting normalisation. Two kinds of methods are available at this stage – feature-based and appearance-based methods. Feature-based approaches capture a set of geometrical facial features such as the shape of eyes, nose and mouth and the distances between them. Appearance-based approaches consider the global properties of the human face pattern without the detection of fiducial points. Dimension reduction approaches such as PCA, LDA or wavelet transform often follow feature extraction, since appearance features can be extremely high-dimensional and redundant.

Verification: A number of statistical and non-statistical classifiers are employed for face recognition. Among the most popular algorithms are eigenfaces by PCA or LDA (Kirby et al. 1990; Zhao et al. 1998), local feature analysis (Penev et al. 1996), independent component analysis (Bartlett et al. 1997; 2003), line edge map (Gao et al. 2002), elastic graph matching (Wiskott et al. 1997), neural networks (Laurence et al. 1997) and hidden Markov models (Samaria et al.1994; Nefian et al. 1999). The eigenface is an approach which uses a PCA transformation to find principal face eigenvectors, whereas local feature analysis focuses on local parameters on the face, such as the position of eyes and the shape of nose. ICA is a transformation-based approach, focussing on seeking a set of basis vectors which possess the maximum statistical independence of a face image (Kong et al. 2005).

2.2.5 Palmprint and hand geometry

The palmprint is an image of the texture of the human hand (Figure 2-5a). Hand geometry refers to the size and shape of the hand and fingers (Figure 2-5b). Recently a system in which the features of the palmprint are fused with those of hand geometry has been reported to achieve promising results (Ribaric et al. 2005).



(a) **(b)**

Figure 2-5: Palmprint and hand geometry samples

Acquisition: Palmprint and hand geometry images are always simple with a black background and are therefore easy to discern. A simple thresholding approach can be applied successfully.

Feature extraction: Regarding handprints, prominent palm-line features, the end points of these lines, texture, global texture energy as well as a combination of these characteristics can be used (Shu et al. 1998; Zhang et al. 2003; Duta et al. 2001; You et al. 2002; Han et al. 2003). Line feature matching algorithms can be used to extract the characteristics of prominent palm-line features (Shu et al. 1998; Han et al. 2003). A robust but simple method has been reported using four line detectors or directional masks to extract palm-line features (Kumar et al. 2003). In hand geometry, the length and width of the hand and fingers are usually used as features (Jain et al. 1999; Sanchez-Reillo et al. 2000; Golfarelli et al. 1997). Commonly used feature vectors include 4 finger lengths, 8 finger widths (2 widths per finger), palm width, palm length, hand area and hand length

(Kumar et al. 2003). Furthermore, dimension reduction approaches such as PCA or LDA can be applied to orthogonalise high-dimensional features (Slobodan et al. 2005).

Verification: At the verification stage, template matching can be used, based on distance measures such as Euclidian distance or Hamming distance. Alternatively, it is also possible to use statistical modelling such as HMM or GMM. These statistical approaches can more powerfully handle a large amount of variation, but the computing load is heavier than in the template matching approach.

2.2.6 Other physical biometrics

A number of other physical biometrics may be used for authentication, e.g. odour (Korotkaya 2004), body shape (Godil et al. 2003) and dental records (Chen et al. 2005; Jain et al. 2004). However, these biometrics are not very reliable, as they are easily affected by a variety of factors. For example, odour can be altered by contamination with different smells, while body shape can be changed by age, illness and stress. Dental authentication is also not entirely reliable and is vulnerable to forgery. As reported in Chen et al. (2005), top-4 dental matches could only achieve 91% accuracy for 24 subjects. In summary, although these biometric data contain useful characteristics, they do not form part of the main features used in biometric authentication. They will therefore receive no further mention here.

2.3 Behavioural biometrics

Although gait, signature and voice quality partly reflect physical characteristics, they are mostly characterised by dynamic, behavioural traits that are learnt and acquired over time (Brand et al. 2001). Therefore, these traits are regarded as behavioural biometrics.

2.3.1 Gait

Gait is defined as the coordinated, cyclic combination of movements that result in human locomotion (Figure 2-6). In gait authentication a salient property, e.g. style of walk or pathology, is recognised on the basis of the coordinated cyclic motions that result in human locomotion. As was pointed out above, gait is not independent of a person's physical properties like length and weight, but the focus is on dynamic characteristics.

Acquisition: A sequence of images (or a section of video) is captured.

Feature extraction: Moving objects can be identified against their background by static region subtraction. The outlining part of a set of obtained pixels representing the region of the moving object is important for recognition. This outline, referred to as a silhouette, is particularly useful for gait recognition (Bauberg et al. 1995; Wang et al. 2003). A motion field, referred to as an optical flow and representing the movement or flow of pixel brightness in an image sequence, can also be extracted (Barron et al. 1994). Motion energy and a motion history image can be also used as features for gait recognition (Davis et al. 1997).

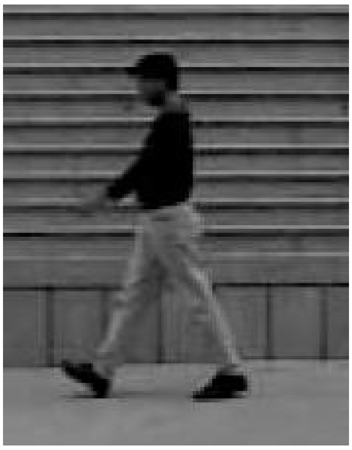


Figure 2-6: Image of gait

Verification: By their source of oscillations of the gaits, the algorithms used in verification can be categorized according to shape, joint trajectory, self similarity, and pixel oscillations. Each of these approaches uses its own designed measures to capture the oscillating property of different types of human gaits. For instance, in the shape based approach, a system uses optical flow to identify a moving figure in a sequence of images. It then describes the shape of the moving figure with a set of scalars derived from Cartesian moments (descriptors include the *x* and *y* coordinates of the object centroid, the *x* and *y* coordinates of the object centroid weighted by the magnitude of the optical flow, and the aspect ratio of the distribution of pixels). When the duration of the sequence is taken into consideration, each scalar forms a time series. The system extracts the oscillations from each series and then finds the frequency and phase of the oscillations, which are used for gait recognition. Through their evaluation an accuracy of around 90% is achieved for a sample size of six. Other methods follow roughly the same approach but with different foci. For instance, the joint trajectory approach captures the underlying characteristics of a joint trajectory in motion and uses these features for recognition. The self similarity approach, by contrast, exploits the property of self similarity characterising gaits for recognition. More details can be found in Boyd et al. (2005).

2.3.2 Signature

Offline signatures are frequently used to authenticate a writer's identity (Figure 2-7). However, online signatures, which also take account of the dynamics, pen pressure and pen angles, are considerably more reliable than offline signatures. Of course, focussing on dynamic aspects of the signature does not mean that these are not affected by physical characteristics of the writer, e.g. the size of the hand and fingers.

Acquisition: Offline signatures (written on paper) can be acquired by digital scanners. Online signatures, in which not only position coordinates, but also time, pen pressure and angles may be obtained, can be recorded either with specialised writing tablets or by recording input from a touch screen (in which case pen pressure and/or angle is not recorded).

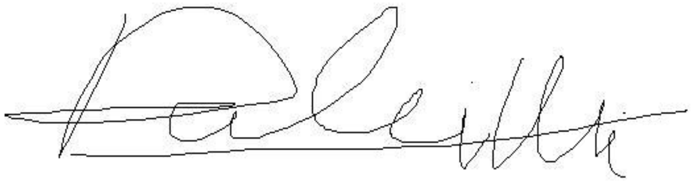


Figure 2-7: Signature sample

Feature extraction: From the low level features acquired directly from the input device, further features may be derived. These include local features, such as pen velocity, acceleration and line curvature, or global features, such as shape and spectral features (Hamilton et al. 1995; Wu et al. 1998; Matsuura et al. 1996; Kashi et al. 1996; Matsuura et al. 1998; Yang et al. 1995; Wirtz, 1995; Nalwa 1997).

Verification techniques: These techniques include dynamic time warping (DTW) (Wirtz 1995), probabilistic classifiers (Bauer et al. 1995; Kim et al. 1995), hidden Markov models (HMM) (Yang et al. 1995) and neural networks (Hamilton et al. 1995; Wu et al. 1997). If the features obtained for the signature are highly unique, a template-based matching approach such as string matching and dynamic programming algorithms may be used. Probabilistic classifiers may also be used, modelling the distributional characteristics of the feature space. These either include temporal sequence information, e.g. by using HMM, or exclude time information (except indirectly, in the form of velocity features etc.), e.g. by using GMM. The artificial neural network (ANN) is another approach, in which the neural net may be directly used as a classifier. The performance of the ANN approach is often only positive in a fairly small sample space.

2.3.3 Handwriting

Handwriting recognition is more complex than signature recognition because it is text independent. Text-independent, writer-specific writing characteristics are exploited in handwriting recognition. As with the signature, physical characteristics of the hand and fingers will affect the handwriting dynamics.

Acquisition: Handwriting can be obtained by using special input devices or digital cameras, which are similar to ones used in signature acquisition. Nevertheless, the devices for handwriting acquisition are generally larger in volume, equipped with higher resolution and therefore more powerful than ones for signature acquisition, due to more than one word being inputted as required in most cases of handwriting recognition.

Feature extraction: As is the case with signatures, handwriting acquisition can be both online and offline. Offline features include position coordinates, angle, shape and curvature. Online features permit the use of not only dynamic features but also pen pressure and writing sounds (Li, 2004; Mitra et al. 2005; Yu et al. 2004).

Verification: The verification techniques used in signature authentication are also applied to handwriting authentication, the only difference lying in the model units. In signature authentication the whole signature is used as a model unit, whereas in handwriting letter models are concatenated to form words and sentences using a template or a statistical model such as HMM.

2.3.4 Voice quality

Voice quality takes up a central place in this dissertation. We shall therefore discuss it in a more detail than the previous biometrics. As with the previous behavioural biometrics, voice quality has important physical aspects. As described in Laver (1968), the quasi-permanent quality of a speaker's voice derives from two main sources – the speaker's anatomical and physiological structures and the long-term muscular adjustments, which are also referred to as 'settings'. Since these settings are acquired idiosyncratically or by social imitation, voice quality is mostly regarded as a behavioural biometric.

Speech contains different types of information (cf. Table 2-1), not all of which are relevant for biometric recognition. Short-term voice quality is important for the realisation of phones, while medium-term voice quality is used by all speakers to reflect for instance his attitude or emotion. These cause variation in the speech signal which cannot be used directly to distinguish between speakers. Quasi-permanent and permanent settings, however, are very relevant for recognising a speaker's identity.

Permanent voice quality characteristics depend on the anatomy and physiology of the speaking organs (i.e. the larynx and the vocal tract) of a speaker and determine their potential operating range, while the quasi-permanent, long-term muscular adjustments of the speaking organs by speakers determine their habitual range in which these organs often work. For example, a singer's voice may be physically capable of spanning a wide pitch range, especially when he sings; in normal speech, however, he mostly speaks in a more restricted range within the total possibilities. The speaker's anatomy and physiology thus determine the possible extremes, and voluntary muscular settings, even if the speaker is not aware of them, determine the habitual range between those extremes.

The muscular settings are divided into two groups by Laver (1968): the laryngeal settings and the supralaryngeal settings of vocal tract.

Laryngeal settings are in turn divided into three sub-categories: phonation types, pitch types and loudness ranges. Besides 'normal voice', phonation types include 'breathy voice', 'whispery voice', 'creaky voice', 'falsetto voice', 'ventricular voice' and 'harsh voice', and combinations of these. Pitch ranges within the total possible range in any phonation type are evaluated on a five-point scale: 'very deep', 'deep', 'medium', 'high' and 'very high'. Similarly, loudness ranges are described with a five-point scale: 'very soft', 'soft', 'medium', 'loud' and 'very loud'.

Supralaryngeal settings of the vocal tract are divided into four groups according to the modification of the shape and acoustic characteristics of the vocal tract: longitudinal modifications through vertical displacement of the larynx or lip protrusion, latitudinal modifications including labialisation and laryngealisation, tension modifications influencing the acoustic damping characteristics of the vocal tract by changing the stiffness of the vocal tract walls, and nasalization as well denasalisation’.

Table 2-1: The relation between vocal variables and their marking functions (adapted from Laver et al. 1979)

Signal functions	informative		informative and communitive	
Relation to language	extralinguistic voice characteristics		paralinguistic ‘tone of voice’	phonetic realisations of linguistic units
Temporal perspectives	permanent	quasi-permanent	medium-term	short-term
Vocal variables	voice features deriving from anatomical differences between individuals influencing both quality and dynamic aspects	voice settings, i.e. habitual muscular adjustments of the vocal apparatus including voice quality settings and voice dynamic settings	‘tone of voice’ achieved by temporary use of voice settings, including paralinguistic quality settings and paralinguistic dynamic settings	momentary articulatory realisations of phonological units, including short-term manipulations of phonetic quality features and short-term phonetic dynamic features
Marking function	physical markers	social and psychological markers		
Potential control-ability	uncontrollable, therefore unlearnable	under potential muscular control, therefore learnable and imitatable		

As shown in Table 2-1, three types of the sources of indexical information are related to the voice qualities described above: biological, psychological and social information, where biological information, such as the size, sex, age, medical state and physique of a speaker and the size of his larynx and vocal tract, is regarded as the physical biometric basis of speaker recognition. But quasi-permanent characteristics which do not directly reflect a speaker’s anatomy and physiology are also useful. Psychological information related to a speaker such as his personality can also be derived from features of voice quality. It is believed, for example, that ‘a harsh voice is correlated with more aggressive, dominant, authoritative characteristics and a breathy voice with more self-effacing, submissive, meek personalities’ (Laver 1968). Finally, social information about a speaker, such as regional origin, social status, social values and attitudes, and profession or occupation, can also be partly judged from voice quality, for instance, from his accent. These properties are behavioural biometrics which can be used to distinguish between speakers.

Widely used features in speaker recognition, such as mel-frequency cepstrum coefficients (MFCC) and linear prediction coefficients (LPC), reflect all the voice quality characteristics in the Table 2-1 (including short- and medium-term properties). This emphasises the need to enhance those differences in speakers' voice quality which can help to distinguish them. Their processing details for derivation of the features such as acquisition, feature extraction and modelling will be introduced in Part II of this dissertation, while Part III focuses on feature enhancement to distinguish speakers on the basis of their voice quality.

2.4 Advantages and disadvantages of each biometric

In Section 2.2 and 2.3 the different characteristics of a number of widely used biometrics were introduced. Based on their characteristics, we summarise their advantages and disadvantages in a table (Table 2-2) to make their characteristics more comparable. To this end, the following criteria are suggested to be used, each of which is evaluated with only two levels for the sake of clarity: positive (√) and negative (×):

- **Non-changeability** (NON-CHANGE), indicating whether or not a biometric is unchangeable throughout human life.
- **Uniqueness** (UNIQ), indicating whether a biometric is unique to individuals.
- **Accuracy** (ACCURACY), indicating performance accuracy.
- **Difficulty of imposterisation,** (DIF-IMPOSTER) indicating the capacity to prevent imposterisation.
- **Non-Intrusiveness** (NON-INTRUSIVE), indicating the degree of non-intrusiveness of a biometric used for recognition.
- **Robustness** (ROBUST), indicating whether a biometric-based application is robust to variation under noisy conditions.
- **Speed** (SPEED), indicating the speed of recognition of an application system using a biometric.
- **Unawareness** (UNAWARE), indicating whether the subjects from whom biometric data are taken are necessarily aware of the action of acquisition.

As shown in Table 2-2 physical biometrics are generally less variable than behavioral biometrics, and therefore can distinguish between persons quite reliably. In particular, DNA, iris and fingerprint are unchangeable and unique from person to person throughout life. DNA analysis is considered very reliable to discriminate between people, but takes much time and is computationally intensive. It is normally only used for forensic purposes. The fingerprint too has strong forensic connotations, and like iris recognition, it is felt as intrusive by users, which makes it unsuitable for many applications. The fingerprint and iris can be impostorised fairly easily and therefore require technical measures (measurement of fingertip temperature and variation of lighting to induce pupil size variation) for aliveness testing.

Table 2-2: Comparison of advantages and disadvantages of different biometrics

	NON-CHANGE	UNIQ	ACCURACY	DIF-IMPOSTER	NON-INTRUSIVE	ROBUST	SPEED	UNAWARE
DNA	√	√	√	√	×	√	×	×
iris	√	√	√	×	×	√	√	×
fingerprint	√	√	√	×	×	√	×	×
face	×	×	×	×	√	×	×	√
palm print	×	×	×	×	×	×	×	×
body shape	×	×	×	×	√	×	×	×
odour	×	×	×	×	√	×	√	√
keystroke	×	×	×	√	×	×	×	×
gait	×	×	×	×	√	×	×	√
signature	×	×	×	×	√	×	√	×
handwriting	×	×	×	√	√	×	×	×
voice quality	×	√	√	√	√	×	×	√

Other biometrics, particularly behavioural ones, are intrinsically variable and therefore require more complex modelling than physical biometrics, which can often be verified using simple template matching techniques. The signature and handwriting are non-intrusive, but their accuracy is not high enough for discrimination. While the face is unique to each person, it is vulnerable to impostorisation (as for instance when an impostor wears a facial mask) and its accuracy is affected by changing illumination conditions. On the other hand it has the advantage that it can, if necessary, be obtained even from non-cooperative persons. This is also true for gait, body shape and odour, but these can also be affected by a variety of factors mentioned in Sections 2.2.6 and 2.3.1, so that their performance is not very high either. Last but not least, voice is generally regarded as the most natural, friendly and non-intrusive among all the biometrics. Unfortunately, by current approaches its performance degrades sharply under noisy conditions, a point which represents a topic of future research in this domain.

2.5 Applications

Biometric-based technologies have a wide range of applications. There are two main reasons for this. Firstly, conventional approaches such as PINs and smartcards are not sufficiently secure. Secondly, current security systems are not convenient to use as they have a lot of drawbacks. For instance, PINs are easily forgotten, a person having obtained a PIN can get free access to private systems, and smartcards can be lost or misplaced. These problems can be highly inconvenient for users and cause high financial looses. Biometric-based authentication techniques can be used alone or integrated with the conventional approaches to provide users with a form of automatic access control which is more user-friendly and safer.

The suitability of biometric applications depends on whether the subject is cooperative or uncooperative and aware or unaware of being observed, as well as on the requirements of the transactions. For instance, if DNA is used for recognition DNA samples must first be acquired. People may not agree to this, as DNA carries a lot of sensitive information. By contrast, other biometric types such as voice, signature and handwriting are non-intrusive and are part of one’s public identity, this making them suitable for a much wider range of applications.

We next give a few examples for each of the two distinct types of biometric application, identification and verification.

2.5.1 Identification systems

Identification can be applied to scenarios where a subject is required to be identified from a given set of known subjects. In such scenarios, although human-based selection can in some cases be more accurate than automatic biometric-based identification, the latter is greatly advantageous on the aspect of low cost and guarantee of uninterrupted operation. Biometric-based identification systems are especially well-suited for applications requiring continuous identification, such as surveillance systems.

2.5.2 Verification systems

Verification is used more widely than identification. Roughly speaking, identification is more relevant to government-related and public issues, whereas verification is more commonly associated with private human concerns. Thus, anything relevant to private access, such as access control to personal private data, vehicles, bank account information, house, office, computer or PDA can be enhanced by employing or integrating biometric-based authentication technologies, as binary decisions are required by these systems. Typical examples are personal identity (ID) card authentication systems (e-card), electrical-government (e-government), electrical-health (e-health) and electrical-banking (e-banking). Depending on the biometric which are practically available, the required accuracy of the authentication and the cooperativeness of the clients, one or several biometrics can be adopted as the carrier of the user’s discriminating information in these systems.

The range of biometric applications is already very wide and the number of potential applications may be expected to increase as costs are reduced and reliability and ease of use improve. Their applications are likely to make a deep impact on a variety of commercial and industrial activities and may even completely change their conventional modes of working. In future society, with the maturity of biometric-based authentication methodologies, their deployment may be expected to become more frequent and wider in range.

2.6 Summary

Biometrics can be divided into two different categories: physical and behavioural biometrics. The physical biometrics are a direct reflection of a person’s anatomical or physiological characteristics, while the behavioural biometrics are learnt or acquired over time and are therefore under the control of the individual. A set of physical biometrics including DNA, iris, fingerprint, face, palmprint, hand geometry were presented in Section 2.2. Behavioural biometrics such as gait, signature, handwriting and voice were introduced in Section 2.3. Following the detailed description of each biometric data, the advantages and disadvantages of each were clarified under the same criteria in a table in order for comparison. Voice data is generally rated as the most non-intrusive and user-friendly biometric feature, albeit of its relative difficulties in processing. Finally, the possibilities of the application of biometrics to practical systems were discussed in terms of two types of tasks, i.e. identification and verification.

3. Multimodal biometric authentication

3.1 Overview

Multimodal identity recognition aims to increase the performance and reliability of biometric authentication systems by combining multiple modalities instead of using a single one (Indovina et al. 2003; Snelick et al. 2005; Ly-Van et al. 2003; Ross et al. 2003; Chang et al. 2004). For instance, state-of-the-art commercial off-the-shelf (COTS) fingerprint and face biometrics have been combined to show improved performance (Indovina et al. 2003, Snelick et al. 2005). Signature and voice biometrics have been fused to achieve the improved performance over the baseline system (Ly-Van et al. 2003). Moreover, face, fingerprint and hand geometry have been combined to fully utilise the complementary properties of these multiple modalities and also obtained improved performance (Ross et al. 2003).

Multi-modal fusion techniques have also been applied to combine different feature sources from the same biometric to improve the performance of a uni-modal system. Thus, Chang et al. (2004) recognised faces by using 2D, 3D and infrared images for the acquisition of fused-face features and achieved better results. This idea has also been applied to other systems (e.g. speaker recognition) where different, complementary types of features can be combined. SOM features combined with standard MFCCs for speaker recognition is another example (Chapter 12). Koreman et al. (2006a) also combined different data models using the same feature data (GMMs with different number of Gaussians).

As a result, attention has been drawn more and more to the fusion of biometric measures for biometric authentication systems. Research in this field represents an emerging trend.

3.2 Advantages and disadvantages of multimodal systems

Since unimodal systems rely on the evidence from a single source of information for authentication, they often have the problem that a single biometric may not have sufficient discriminating capacity to reliably distinguish a large number of subjects, especially under noisy conditions. However, providing the different modalities contain complementary information, fusing different sources of information generally improves recognition performance. Concretely speaking, a multimodal system is superior to a uni-modal system in the following aspects (Ross et al. 2003):

Robustness to noisy data: The probability of simultaneously obtaining noisy data in different biometric modalities is much lower than when using only a single biometric modality. Therefore, the information lost in one type of noisy biometric data can be complemented probably by other types of biometric data. For instance, voice data captured under noisy conditions can be complemented by the confirmative information obtained from the subject's face and online signature. The results of fusion experiments using speech, face and online signature to enhance system performance strongly support this argument (Koreman et al. 2006).

Robustness to intra-class variation: Intra-class variance is the variance within a model which is caused by non-relevant factors, for instance, linguistic content for a voice model. Though introducing new evidence cannot reduce intra-class variance, additional information sources can increase interclass variance, so reducing the relative importance of intra-class variance, making a system more robust.

Less inter-class similarity: In a biometric system comprising a large number of users, there may be inter-class similarities (overlap) in the feature space of multiple users. Golfarelli et al. (2000) have stated that the number of distinguishable patterns in two of the most commonly used representations of hand geometry and face are only of a small order number (103). However, by adding more different feature sources, inter-class similarity can be greatly reduced, since more discriminating evidence is taken into consideration.

High accuracy: Accuracy is often increased due to the complementary information supplied by each different biometric data source. The more information used by a system, the better its performance.

Robustness to impostorisation: Behavioural traits such as signature or long term speech characteristics are more vulnerable to impostorisation, since they can be imitated. Combination of behavioural traits with non-behavioural biometrics can make impostorisation activities much harder.

The multimodality (or multi-expert combination) approach results in recognition systems that are more accurate and more flexible, and operate in a wider range of conditions. However, implementation complexity is unavoidably increased at the same time. This can be a prohibitive factor, especially for handheld devices such as mobile phones and PDAs, which have limited computational power and memory.

3.3 Fusion strategies

Three types of fusion strategy have been proposed (Ross et al. 2003, 2004; Jain et al. 2005):

Feature-level fusion: This combines modalities at the earliest stage. While this often gives good results (e.g. concatenation of time difference features in speaker recognition), this is not always the case. Increasing the size of the feature vector increases the number of model parameters and hence the amount of training data required to achieve accurate models. Furthermore, most commercial biometric systems do not provide access to the feature sets (nor to the raw data). The performance of feature-level fusion is therefore often not as good as the other fusion strategies.

Decision-level fusion: In this case, fusion occurs at the final decision stage, i.e accept/reject decisions from several experts are combined. This is considered to be rigid due to the limited availability of information, since a lot of discriminative information contained in different biometric sources is lost in the preceding process stages.

Matching-score-level fusion: This is a compromise between feature-level fusion and decision-level fusion strategies. It is generally preferred as it is relatively easy to access and combine the scores presented by the different modalities (Jain et al. 2005). Since fusion at the matching score level gives the best performance, several fusion rules have been tested, such as the simple sum rule, decision tree and discriminant analysis function (Ross et al. 2003). The simple sum rule was shown to be superior to the other two rules. Other fusion approaches, such as the use of GMM, have also been proposed and excellent results were obtained (Allano et al. 2006).

3.4 A typical multimodal system: SecurePhone

A typical multimodal application is SecurePhone system, which I present here as an example to illustrate the fast development of multimodal biometric authentication systems. It was built from 2004 till 2006, in a European project, in which Saarland University cooperated with ATOS Origin, Informa, Telefónica Móviles España, Nergal, the University of Buckingham and the Groupe des Ecoles des Télécommunications (ENST Ecole nationale supérieure des telecommunications and the INT Institut National des Télécommunications). The work for this dissertation was carried out within the framework of the SecurePhone project[1].

SecurePhone is a typical multimodal verification system combining three modalities, i.e. voice, face and online signature, to provide enhanced security in user authentication. As discussed in the whole Section 2 and Section 3.2, each of biometric data can represent in its own format one type of sources of discriminating information for an individual. As these different types of information may be complementary to some extent in identifying a person’s uniqueness, they can be used in a combinational way so as to improve the reliability of the authentication systems. A fusion strategy combining the matching scores was used for the scores derived from each of the three modalities. In verification, the three scores were concatenated into a vector, which was then passed through a GMM fusion unit modelling the client and a set of impostors (as described later). The obtained log-probability output from the GMM is the fusion score, which is used as a basis for the

[1] http://www.secure-phone.info/.

accept/reject decision. Multimodal verification significantly improved user verification in comparison to any of the biometrics on its own (Allano et al. 2006).

3.5 Summary

Multimodal systems improve recognition accuracy by combining complementary sources of biometric information. With more sources of information being taken into consideration by the recognition process, the inseparable cases due to, for instance, noise data, less inter-class separability and higher intra-class variations in the uni-modal recognition can be possibly converted into separable cases in the multimodal recognition, so that the system performance is able to be enhanced. In order to fuse multiple modalities, three strategies can be considered: feature-level fusion, decision-level fusion and matching-score-level fusion, among which, matching-score-level fusion have nice properties of higher performance and less training data required. As a typical example for illustrating multimodal biometric applications, SecurePhone project was introduced, which combined three user-friendly modalities, i.e. voice, face and online signature, with improved performance.

4. Goals and structure of dissertation

4.1 Goals

Although this research was conducted in the framework of the SecurePhone project, its achievements go beyond the techniques directly applied in this project. For instance, MLP-based feature enhancement (Chapter 11) is a generalised approach which can be applied in a similar way for any biometric for use in identity recognition.

The same features have been used for speaker recognition as for speech recognition throughout many years, although their aims are completely different. Finding the means of overcoming this drawback will lead to a significant improvement for both speaker and speech recognition in various conditions, *viz.* in noisy conditions or in low-bandwidth telephone speech (cf. Part III). Thus, seeking the speaker-specific features for speaker recognition was one of the objectives of this research.

One of possible ways to derive speaker-specific features for speaker processing is feature space transformation, as is commonly used in speech recognition. For speech recognition, PCA, LDA and NLDA, have been proposed for improving the system's robustness to various real-life conditions. The transformed features were found to be superior to the original mel-scaled features even if they were used alone (cf. Chapter 11).

Some of these approaches have also been applied in speaker recognition, but with limited success. For instance, Jin et al. (2000) proposed to apply LDA for speaker recognition. Heck et al. (2000) and Konig et al. (1998) further suggested the use of NLDA implemented by an MLP for speaker recognition. However, LDA can only achieve good results with a very small number of testing speakers. Heck et al. (2000) achieved consistently positive results by linearly combining the discriminative features with the mel-scaled features.

Two reasons may be given as an explanation for the difficulty of using the NLDA transformation for speaker recognition. Firstly, speaker features are far more overlapping than phonemes in an acoustic space (cf. Chapter 10). Secondly, the number of speakers enrolled in an application can be much larger than the number of speech units, i.e. the phonemes (which amount to around 60 in English). For these two reasons separating speakers is much harder than separating phonemes. Furthermore, the higher number of speakers generally causes learning problems for discriminating classifiers (e.g. LDA or MLP) in the training stage due to the low amount of training data per speaker.

In this research it was found that the problem can be solved by automatically selecting some “important” speakers (the “speaker basis”) for a discriminating classifier such as an MLP to be trained on. Although the MLP-based NLDA approach was already proposed by Konig et al. (1998), they trained the MLP with a different purpose. The purpose of their method was to deal with microphone mismatching by using approximately 30 speakers manually selected to balance the effect of using different handsets. However, their approach was consistently effective only when combining their derived features with the original MFCC features. In this research we further extend the work of Heck et al. by systematically investigating this method. It was found that discriminative features can consistently improve speaker recognition performance if they are well learned, even in cases where they are applied alone.

Besides discriminative feature transformation, an alternative feature representation obtained from a self-organising map (SOM) was also investigated in this dissertation. These features were found to contain information complementary to the MFCC features, even though they were derived from these features. This is because the phonotopic infrastructure of an acoustic space can be captured using SOM processing (cf. Chapter 12).

The contributions of this dissertation are:

1. *A speaker-phone distribution (SPD) describing the structure of an acoustic space based on feature space analysis was proposed to provide support to the necessity of the application of feature enhancement approaches (Chapter10).*
2. *LDA was compared with a linear MLP and NLDA. The discriminative features learned by LDA were not very effective for speaker recognition with a large population (Section 11.4.6).*
3. *It was tested and found that NLDA implemented by an MLP is more powerful than LDA for speaker feature transformation (Section 11.4.6).*
4. *Different types of MLP were tested for speaker feature transformation and a 3-hidden-layer MLP was found most efficient among them (Section 11.4.4).*
5. *It was established that the number of speakers used for the derivation of discriminative features is one of the most crucial factors affecting the performance of MLP-based feature enhancement (Section 11.4.4).*
6. *Besides the number of speakers, the method of selection of the speaker basis was also found to have an effect on system performance. Several methods of speaker basis selection in addition to random selection were proposed. One was knowledge-based, two others were data-driven. An automatic, data-driven selection method favouring boundary speakers was found to be most efficient (Section 11.5).*
7. *The essence of our feature enhancement approach was geometrically interpreted as stretching the speaker acoustic feature space by maximising the average between-class variance (Section 13.2).*

8. *The "speaker voice signature" was suggested as a complementary feature type which can be combined with the MFCC features to improve system performance (Chapter 12).*

In summary, this research was especially dedicated to the investigation of new and complementary types of speaker-discriminating feature representations for speaker recognition. Experiments showed that these new discriminative feature types can be used to improve the performance of a state-of-the-art speaker recognition system in various conditions.

4.2 Structure

After the previous introduction into widely used biometrics and their multi-modal combination, with the SecurePhone project as an example which forms the background to this thesis, the rest of this dissertation consists of three parts.

Part II presents the theoretical basis of speaker recognition. State-of-the-art speaker recognition technologies are summarised, including the three main stages of a speaker recognition system after data acquisition. The rest of this part is organised according to these three stages. In the first stage, feature extract approaches are addressed and the three feature types most commonly used for speaker recognition are described. After this, principal components analysis, linear discriminant analysis and nonlinear discriminant analysis, three techniques often used for speech and speaker feature transformations, are discussed. In the second stage, conventional data modelling approaches such as template-matching, GMM-based and HMM-based approaches are presented. Finally, in the decision stage, recognition decision theory and some related score normalisation techniques for speaker recognition are summarised.

Part III mainly addresses experimental techniques for feature enhancement. Firstly, feature space analysis is conducted on an MFCC-based acoustic space. Based on this analysis using different approaches, the speaker-phoneme distribution (SPD) describing the clustering structure of an acoustic feature space is used to provide strong evidence to support the motivation to develop feature enhancement approaches for speaker recognition. Secondly, the approach of NLDA-based data enhancement by MLP is proposed and tested in a variety of noisy conditions. Three methods of speaker basis selection are compared in experiments. Finally, complementary features by SOM processing are presented.

Part IV is a discussion and conclusion of this dissertation. In the discussion chapter, a physical interpretation for MLP-based feature enhancement approaches is given. A discussion of SOM processing and fusion-related issues follows. Finally, achievements and open issues are addressed in the conclusion chapter.

Part II. Speaker recognition

5. Introduction

In order to clearly explain the theoretical part of speaker recognition, an overview of a conventional speaker recognition system is first given, based on four functional stages of processing in recognition systems, i.e. data acquisition, feature extraction, data modelling and recognition decision procedures. Following this, we will proceed to a more detailed explanation of each stage in the subsequent chapters in this part.

But before outlining these stages in Section 5.2, we first introduce some important concepts in speaker recognition research in Section 5.1. An overview of Part II is given in Section 5.3.

5.1 Some important concepts

Some important (pairs of) concepts are often referred to in the literature, e.g. speaker identification vs. verification, text-dependent vs. text-independent techniques, closed vs. open set identification as well as fixed vs. incremental set techniques. The concepts of speaker identification and verification do not only apply to speaker recognition, but also to other types of biometrics. As they have already explained in Section 2.5, we focus here on explaining the other three pairs of concepts.

5.1.1 Text-dependent vs. text-independent techniques

Depending on the level of user cooperation and control of the spoken material in an application, the speech used for recognition can be either text-dependent or text-independent. In a text-dependent application, the recognition system has prior knowledge of the text to be spoken and expects the speaker to cooperatively speak that assigned text. Examples are a user specific password or a fixed phrase. Prior knowledge and constraints of the text can greatly boost the performance of a recognition system. In a text-independent application, the system has no knowledge of the text to be spoken, as in the case of extemporaneous speech. Text-independent recognition is more difficult but more flexible and secure, allowing for example the verification of a speaker while he/she is conducting other speech interactions. As speaker and speech recognition systems merge and speech recognition improves, the distinction between text-dependent and text-independent applications is slowly disappearing. Of these two basic tasks, text-dependent speaker verification is currently the most commercially viable and useful technology, although there has been much research conducted on both tasks (Reynolds et al. 2002).

The difficulty in text-independent speaker recognition systems is mainly due to the variance caused by the content of the speech material. Besides modelling speaker-specific information, text-independent speaker recognition is also required to model the other types of information, such as linguistic information (Hermansky et al. 1998). As a result, text-independent systems need to be able to deal with more variance than text-dependent systems, causing more difficulties for the process of recognition.

It is not easy to model linguistic information or speaker-specific information separately. Hermansky et al. (1998) have proposed that linguistic information can be modelled by using PCA or low-order PLP coefficients. However, they did not achieve satisfactory results. Alternatively, an approach using discriminative features (*viz.* LDA and NLDA-based) may be more suitable for this purpose (cf. Chapter 11).

5.1.2 Closed set vs. open set identification

In "closed set" identification it is assumed that the subject belongs to a given set of registered people and that he will be identified as one of this set. In "open set" identification, the task is to identify an individual who may or may not be a member of a given set of people.

5.1.3 Fixed set vs. incremental set

In "fixed set" recognition, when a new subject is enrolled, all other models which have been previously trained have to be retrained. Fixed-set systems are optimal for applications in which the number of subjects is fixed. In "incremental set" recognition, by contrast, the system only needs to train a new model for the new subject. Algorithms such as the MLP-based post-processing approach (Wang et al. 2002), which do not meet these requirements, are therefore not practical to use here. However, the set of discriminative approaches which we propose in this dissertation conforms to the incremental-set attribution. As the data enhancement transformation learnt by the MLP is speaker-independent, it can be applied to a system without any retraining when a new user is enrolled.

5.2 Stages of processing in speaker recognition

A speaker recognition system consists of at least three functional modules (stages): data acquisition, feature extraction, and recognition decision, as in any biometric-based recognition system (cf. Part 1). In systems where the extracted features for the biometrics are not easily separated, the recognition stage can be further divided into data modelling and score matching (decision making). Therefore, these four stages are used to describe the current speaker recognition systems (Figure 5-1).

At the first stage of data acquisition, speech signals are acquired with an electric device such as a simple PC microphone or an advanced microphone array, which performs the voice signal conversion from analog to digital (A/D) and additional information acquisition (e.g. providing the accurate location of a speaker).

At the second stage, the acquired discrete samples are converted into a sequence of frames using a shift window (e.g. 20ms window width with 10ms shift). A Hamming function is used to alleviate the effect of discontinuity at the frame boundaries. A variety of features can be extracted from these windows, e.g. linear prediction coefficients (LPC), mel-frequency cepstrum coefficients (MFCC) and perceptual linear prediction (PLP) features. These different features contain the underlying speaker-specific characteristics of the speech signals, which are then used in modelling and decision making. Therefore, feature extraction is crucial for speaker recognition. The quality of the extracted features has a significant impact on the overall performance of speaker recognition systems. Currently, the most widely used features for speaker recognition are MFCC features. However, since they were designed for speech recognition, they may not be optimal for speaker recognition. We will return to this point in a later chapter of this dissertation (cf. Chapter 10).

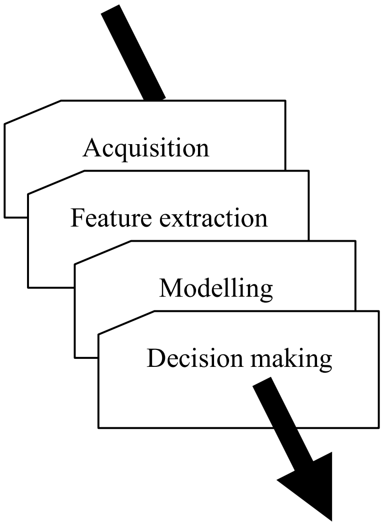


Figure 5-1: The four stages of speaker recognition

At the third stage, the speaker-specific feature distribution in the acoustic space is modelled. By comparing an incoming signal with candidate models, probability scores are obtained. At this stage, the vector quantisation (VQ)-based and the HMM-based approach are the two most common methods used in current systems. While the VQ-based approach models the clustering property of data samples in the acoustic space by means of discrete distributions, HMMs model the correlation of the sequential frames in the temporal domain by continuous distributions. The GMM-based approach may be derived from the HMM-based approach by assigning a single state to HMM, making the modelling ignore the temporal dependency in sequential speech frames. GMM is able to capture clustering characteristics of the feature space only by means of continuous distributions, i.e. Gaussian distributions, while VQ models them with discrete vectors.

Finally, at the matching stage, a decision is made based on the probability scores calculated at the previous modelling stage. Different decision criteria are used for different tasks: In closed-set identification tasks, the speaker is chosen as the one from a set of speakers who possesses the highest conditional probability given a voice signal *X*. In verification tasks, a claimed speaker is accepted provided that the ratio of the conditional probability for the claimed speaker model to that for its impostor model is larger than a particular threshold; otherwise it is rejected (Figure 5-2).

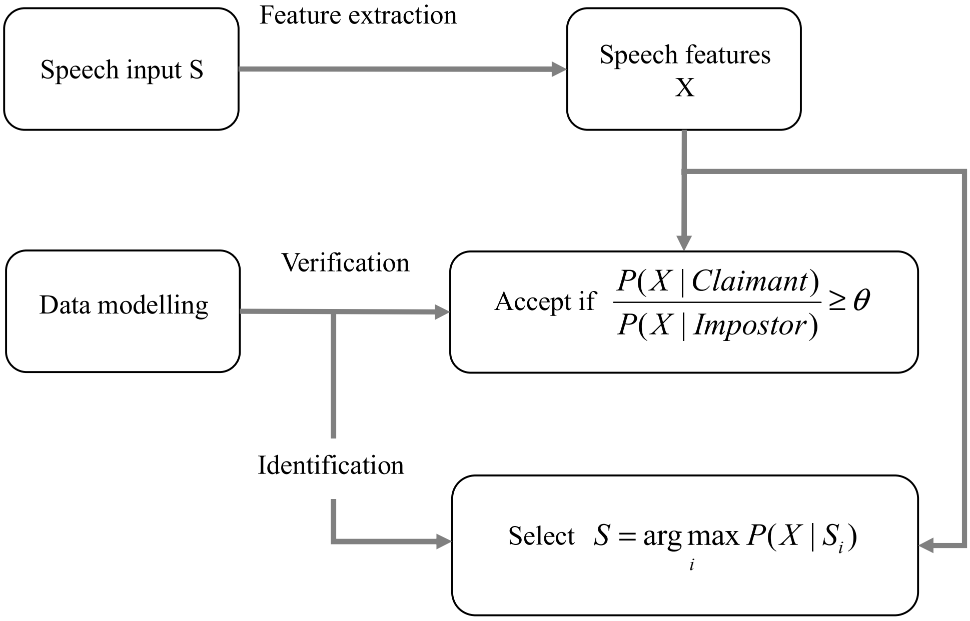

Feature extraction
Speech input S
Speech features X
Verification
Data modelling
Accept if $\frac{P(X \mid Claimant)}{P(X \mid Impostor)} \geq \theta$
Identification
Select $S = \arg\max_i P(X \mid S_i)$

Figure 5-2: Block graph of speaker recognition (identification and verification)

5.3 Overview of Part II

The rest of Part II is organised as follows: In Chapter 6, three types of feature extraction and selection techniques are presented briefly. In Chapter 7, a number of data modelling approaches are discussed. This is followed, in Chapter 8, by a brief outline of decision theory.

6. Feature extraction

Feature extraction is the first step in speaker recognition. The more speaker-discriminating the features are, the better – in terms of robustness and performance – the resulting system is. Many different feature types are used in the various different applications of speech processing, such as speech and speaker recognition. Among them, linear prediction coefficients (LPC), mel-frequency cepstrum coefficients (MFCC) and, more recently, wavelet features have been found to be the most efficient features for speaker recognition. In this chapter, a brief outline is given of these three feature-types.

A lot of information is contained in speech signals, e.g. speaker-specific and linguistic information. While speaker-specific information includes gender, age, dialectal affiliation and speaking style (e.g. speaking rate in terms of either slow or fast speed), linguistic information relates to phonemes, syllables, syntax and semantics. When a sound is articulated, its waveform contains both speaker and linguistic information. As a result, the features derived from this waveform also carry both types of information. Since speaker recognition aims at extracting speaker-specific information, some transformations such as principal components analysis (PCA), linear discriminant analysis (LDA) and nonlinear discriminant analysis (NLDA) may be used to capture or enhance this information. These are discussed in Section 6.2.

The rest of this chapter is organised as follows: In Section 6.1, a number of feature types are described briefly. In Section 6.2, feature enhancement is discussed, following which a summary is given in Section 6.3.

6.1 Features for automatic speech recognition

Many types of features are used for automatic speech recognition. The following discussion will focus on those features which are widely applied to speaker recognition, i.e. LPC, MFCC and the more recent wavelet features. Those feature types, such as PLP and RASTA-PLP (Hermansky et al. 1992), which are not suitable for speaker recognition, will not be discussed further.

6.1.1 Linear prediction coefficients (LPCs)

Linear prediction coefficients or LPC features (Campbell et al. 1997), derived with the help of the autoregression model (AR model), describe certain characteristics of the human vocal tract. This model is essentially a linear prediction equation which obtains its solution subject to the constraint of least mean square (LMS) errors. Ideally, a signal *s(n)* at the time *n* can be described as:

$$s(n)=\sum_{k=1}^{p}\alpha_k s(n-k)+\sqrt{g_s}u(n), \tag{6.1}$$

where *u(n)* is called the residue, g_s is a scaling parameter and *p* is the prediction degree. The *p* coefficients α_k are often used as *p*-dimensional vectors to represent a speech frame, in which case they are called linear prediction coefficients (LPC). It has found that the residue also contains a lot of information of use for speaker recognition (Venaz et al. 1995).

The prediction coefficients α_k can be determined by solving an optimisation equation which minimises the LMS errors between the real signal *s(n)* and its linear approximation $\hat{s}(n)$, i.e.

$$\{\alpha_i\}=\arg\min_{\alpha_i}\sum_{n=1}^{N}\left(s(n)-\hat{s}(n)\right)^2, \tag{6.2}$$

where *N* is the number of observed samples in a frame and

$$\hat{s}(n)=\sum_{k=1}^{p}\alpha_k s(n-p). \tag{6.3}$$

This optimisation equation can be solved by using the Yuler-Walker equation. For more details, see appendix B.

LPC is a classical speech feature type which has been used for speech and speaker recognition for many years. However, given its limited representation capacity due to linear prediction, it is generally used less frequently than MFCC features.

6.1.2 Mel-frequency cepstrum coefficients (MFCCs)

Mel-frequency cepstrum coefficients (MFCCs) are at present the most popular and most successful features for both speech and speaker recognition. They are also intensively applied in the baseline experiments of Chapter 10-12. Therefore, the description to MFCCs will be in more details in this section. This state-of-the-art feature type is generated with the following steps.

A/D conversion: In this method, it is assumed that a given continuous speech signal has been digitised at a sampling rate *R* into a sequence of discrete speech samples $S=\{s_t\},\ n=1,N$.

Pre-emphasis: A window with a fixed width (often 20-25 ms) is used to segment the speech signal into frames. The window shift is normally set at half the size of the window. The signal is then pre-emphasised by applying a first order difference equation

$$s_n{}'=s_n-ks_{n-1} \tag{6.4}$$

to the samples S in each window, where k is a pre-emphasis factor valued in the range of $0 \le k < 1$, usually set to 0.97. To counteract the effect of discontinuities at each window boundary, a Hamming window is usually applied

$$s''_n = \left\{ 0.54 - 0.46\cos\left(\frac{2\pi(n-1)}{N-1} \right) \right\} s'_n . \tag{6.5}$$

Frequency analysis: After the pre-emphasis step, the original signal s becomes s' but still represents a signal in the temporal domain. The temporal domain signal is transformed into a frequency domain signal by discrete Fourier transform (DFT), i.e.

$$f_k = \left| \sum_{n=0}^{N-1} s''_n \cdot e^{-2\pi ink/N} \right| , \tag{6.6}$$

whereby k=[f_L,f_H],f_L is the lower cut-off frequency, f_H the upper cut-off frequency.

Spectral Warping: The human ear resolves frequencies non-linearly across the audio spectrum. Empirical evidence suggests that designing a front-end which operates in a similar non-linear manner improves recognition performance. Therefore, a similar frequency warping is also applied to f_k according to

$$Mel(f) = 2595\log_{10}(1 + \frac{f}{700}) . \tag{6.7}$$

Following this, a conversion to the log-domain is always performed, resulting in

$$m_k = \log\left(Mel(f_k)\right) . \tag{6.8}$$

In practice, a Mel-scaled filterbank is always used to give approximately equal resolution on the mel-scale instead of directly applying (6.8). To implement this filterbank, the magnitude Fourier coefficients are binned by convolving them with each triangular filter. These filters are equally spaced along the mel-scale which is defined by (6.7) with the appropriate parameters to achieve an approximation of the effect of (6.8) where k is equal to the number of the triangular filters.

Orthogonalisation: After spectral warping, the discrete cosine transform (DCT) is applied to orthogonalise the Mel-scaled log filterbank amplitudes in order to eliminate correlations among them, i.e.

$$c_k = \sqrt{\frac{2}{N}} \sum_{j=0}^{N-1} m_j \cos\left(\frac{\pi i}{N}(j - 0.5) \right) , \tag{6.9}$$

where k is the number of the triangular filters.

Liftering: One of minor problems with MFCCs is that the higher order cepstra are numerically quite small and this results in a very wide range of variances when going from the low to high cepstral coefficients. To solve this problem, it is often convenient to re-scale the coefficients to have similar magnitudes according to the equation

$$c'_k = (1 + \frac{L}{2}\sin\frac{\pi k}{L})c_k . \tag{6.10}$$

This step is referred to as “liftering”, where L is the liftering value.

The resulting parameters, referred to as MFCC features, contain both linguistic and speaker-specific information in a warped spectrum domain. They may therefore be applied to both speech and speaker recognition with satisfactory results. However, in a later analysis (Chapter 10), it will be shown that the information contained in MFCCs is not balanced to represent linguistic and speaker-specific information equally well, but is more biased towards phonemes by clustering the acoustic space around phone or phoneme classes.

In noisy environments, especially in telephone speech, cepstrum mean subtraction CMS (CMS) is often helpful to remove channel noise, i.e. the mean estimated across a certain length of signals (online or offline) is subtracted from each speech frame, with the hopes to remove the noise in the signals and keep the useful parts remained. This process is usually referred to as “feature preprocessing”. For more discussions about it, see Chapter 11.

6.1.3 Wavelet-transformed features

Wavelet transformed features (WAVCs) differ from MFCCs in that they are derived by substituting the wavelet transformation for the Fourier transformation addressed in Section 6.1.2. In particular, (6.6) is replaced by

$$\psi_{a,b}(t) = \psi\left(\frac{t-b}{a}\right) \tag{6.11}$$

$$W_{\psi}(a,b) = \frac{1}{\sqrt{a}}\int_{-\infty}^{+\infty} s(t)\psi^{*}\left(\frac{t-b}{a}\right)dt\,, \tag{6.12}$$

whereby $\psi_{a,b}(t)$ is the wavelet function scaled (by a) and translated (by b) from the prototype wavelet $\psi(t)$. The wavelet function used in each case depends on the application. While Haar wavelets are often employed in face recognition (Koreman et al. 2006a), in speech recognition Daubechies wavelets are used more frequently (Sarikaya et al. 1998).

A wavelet packet tree is applied to divide the overall frequency range into a number of sub-bands. In each sub-band, a number of coefficients are obtained using variably scaled and translated wavelets. The energy of each sub-signal in each subband is then computed by summing up the squared magnitudes of all the coefficients in that subband, after which it is normalized by the number of coefficients. Following this, the DCT or another wavelet transformation is applied to eliminate correlations between the energy coefficients associated with each subband (Sarikaya et al. 1998).

6.2 Feature enhancement

Feature enhancement uses a transformation to strengthen useful information and reduce the harmful disturbances often contained in features. Two classes of transformations are used for feature enhancement. One consists of linear transformations such as principal components analysis (PCA) and linear discriminant analysis (LDA). The other contains nonlinear transformations (also referred to as NLDA) such as transformations implemented by self-organised mapping (SOM) and a multi-layer perceptron (MLP). Generally speaking, the linear transformations are not as powerful as the nonlinear transformations, as the linear transformations can be regarded as a special case of the nonlinear transformations, for instance, LDA vs. NLDA. Furthermore, besides the linear vs. nonlinear property of the transformations, different learning criteria, such as objective functions for optimisation, which are used in the training of the similar transformations, also have a significant impact on system performance. Thus, discriminative training, as used in LDA, is generally superior to non-discriminative training of the kind used in PCA.

Although these transformations differ in their capabilities of projection, each has its own characteristics and advantages. For instance, PCA is capable of finding a number of the maximal variance directions possessed by the overall data set, making it the most suitable tool to represent a feature space compactly and precisely. LDA is a transformation which is derived by solving an optimisation equation with the help of linear algebra, so that a unique solution is found. NLDA, by contrast, implemented by MLP does not possess the property of having a unique solution, because of the existence of multiple local minima in its training procedure. Therefore, selecting an appropriate initialisation scheme is crucial for NLDA training, while LDA training is relatively simple.

In particular, when these transformations are used for speaker recognition, NLDA implemented by MLP has been found to achieve better results than LDA (Wu et al. 2005a; 2005b; cf. Section 11.4.6). Similar results were reported for speech recognition (Somervuo 2003).

Therefore, in the context of speaker recognition, a brief overview is given in this chapter of PCA, LDA and MLP-based NLDA transformations.

6.2.1 Principal components analysis

The aim of principal components analysis (PCA) is to find the unit direction vectors (**u**) which (locally) maximise the variance of the distances of the projected data set in the direction of (**u**). These vectors, or principal components, can be shown to be the unit eigenvectors of the correlation matrix **C** (Morris 1992), whereby X is the data matrix with a sample as a column vector in it. The distance of each **x** in **X (**the data matrix with a data sample **x** as its column vector**)** along the direction of vector u is: $d = \mathbf{x} \cdot \mathbf{u} / |\mathbf{u}|$, so the variance to be maximised with respect to (**u**) is given by:

$$r = Variance\left(\frac{\mathbf{u}'\mathbf{X}}{|\mathbf{u}|}\right) = \left(\frac{\mathbf{u}'\mathbf{X}}{|\mathbf{u}|}\right)\left(\frac{\mathbf{u}'\mathbf{X}}{|\mathbf{u}|}\right)' = \frac{\mathbf{u}'\mathbf{X}\mathbf{X}'\mathbf{u}}{\mathbf{u}'\mathbf{u}} = \frac{\mathbf{u}'(n\mathbf{C}_{\mathbf{xx}})\mathbf{u}}{\mathbf{u}'\mathbf{u}}, \tag{6.13}$$

$$\begin{aligned} &\Delta_{\mathbf{u}} r = 0 \Rightarrow \Delta_{\mathbf{u}} \mathbf{u}'\mathbf{C}\mathbf{u} / \mathbf{u}'\mathbf{u} = 0 \Rightarrow (\mathbf{u}'\mathbf{u})2\mathbf{C}\mathbf{u} - (\mathbf{u}'\mathbf{C}\mathbf{u})2\mathbf{u} = 0 \\ &\Rightarrow \mathbf{C}\mathbf{u} = [(\mathbf{u}'\mathbf{C}\mathbf{u}) / \mathbf{u}'\mathbf{u}]\mathbf{u} = \lambda\mathbf{u}. \end{aligned} \tag{6.14}$$

The **u'**s are the unit eigenvectors of **Cxx** and their corresponding eigenvalues give the variance of **X** in these directions.

As these projection directions are uncorrelated, the PCA-transformed space is more compact and orthogonal. Therefore, PCA is often used for speech and speaker recognition as a basic approach to dimension reduction, rendering a recognition system more compact and occasionally more efficient.

6.2.2 Linear discriminant analysis

The simplest approach for deriving discriminative features for speaker (class) recognition is by linear discriminant analysis (LDA). LDA is an optimal linear transformation which maximises the ratio of the between-class covariance to the within-class covariance. Through this projection, some of the variation due to non-speaker (class) information may be reduced, while speaker (class) specific properties remain. This enhances speaker (class) discrimination.

Given an original d-dimensional feature space **X**, the goal of LDA is to seek an optimal linear transformation $\mathbf{W}^t$ ($m \times d$) to project the original features into a discriminative space **Y** so that N classes can be more easily separated in the m-dimensional space **Y**, i.e.

$$\mathbf{Y} = \mathbf{W}^t\mathbf{X}. \tag{6.15}$$

Assuming $\tilde{\mathbf{S}}_w$ to be the within-class scatter matrix ($m \times m$) and $\tilde{\mathbf{S}}_b$ the between-class scatter matrix in the mapped space ***Y***, $\mathbf{S}_w$ the within-class scatter matrix ($d \times d$) and $\mathbf{S}_b$ the between-class matrix in the original space ***X*** respectively, then we obtain:

$$\tilde{\mathbf{S}}_w = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M_i}(\mathbf{y}_{ij} - \tilde{\mathbf{m}}_i)(\mathbf{y}_{ij} - \tilde{\mathbf{m}}_i)^t, \tag{6.16}$$

where $\mathbf{y}_{ij}$ is the j-th sample vector in the sequence of class i, M_i the number of samples of class i, and $\tilde{\mathbf{m}}_i$ the mean vector of class i.

If we substitute (6.15) for $\mathbf{y}_{ij}$, then we obtain (6.17):

$$\begin{aligned}\tilde{\mathbf{S}}_w &= \frac{1}{N}\cdot\sum_{i=1}^{N}\sum_{j=1}^{M_i}(\mathbf{y}_{ij} - \tilde{\mathbf{m}}_i)(\mathbf{y}_{ij} - \tilde{\mathbf{m}}_i)^t \\ &= \mathbf{W}^t\cdot\frac{1}{N}\cdot\sum_{i=1}^{N}\sum_{j=1}^{M_i}(\mathbf{x}_{ij} - \mathbf{m}_i)(\mathbf{x}_{ij} - \mathbf{m}_i)^t\mathbf{W}. \\ &= \mathbf{W}^t\mathbf{S}_w\mathbf{W}\end{aligned} \tag{6.17}$$

Similarly we obtain $\tilde{\mathbf{S}}_b$:

$$\begin{aligned}\tilde{\mathbf{S}}_b &= \frac{1}{N}\cdot\sum_{i=1}^{N}M_i(\tilde{\mathbf{m}}_i - \tilde{\mathbf{m}})(\tilde{\mathbf{m}}_i - \tilde{\mathbf{m}})^t \\ &= \mathbf{W}^t\cdot\frac{1}{N}\cdot\sum_{i=1}^{N}M_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^t\mathbf{W}. \\ &= \mathbf{W}^t\mathbf{S}_b\mathbf{W}\end{aligned} \tag{6.18}$$

The purpose of LDA is to seek a set of axes $\mathbf{w}_i$ along which the features in the space **X** are projected subject to the constraint of maximising the between-class covariance and minimising the within-class covariance, which in turn can be described as the ratio of the determinant of $\tilde{\mathbf{S}}_b$ to that of $\tilde{\mathbf{S}}_w$ (Duda et al. 2001). Although it is difficult to carry out the optimisation of such a form, this problem can be solved by seeking a solution along each axis $\mathbf{w}_i$ as follows:

$\mathbf{w}_i^t\mathbf{S}_b\mathbf{w}_i$ and $\mathbf{w}_i^t\mathbf{S}_w\mathbf{w}_i$ correspond to scalar projections of between-class covariance and within-class covariance along the axis $\mathbf{w}_i$. Hence, maximising the ratio of these two scalar projections as in (6.19)

$$\max\frac{\mathbf{w}_i^t\mathbf{S}_b\mathbf{w}_i}{\mathbf{w}_i^t\mathbf{S}_w\mathbf{w}_i}, \tag{6.19}$$

is equivalent to maximising the nominator and minimising the denominator, therefore maximising the equation (6.20) by introducing the Lagrange multiplier

$$J(\mathbf{w}_i) = \mathbf{w}_i^t\mathbf{S}_b\mathbf{w}_i - \lambda(\mathbf{w}_i^t\mathbf{S}_w\mathbf{w}_i)\,. \tag{6.20}$$

Differentiating this equation with respect to $\mathbf{w}_i$, we obtain

$$\mathbf{S}_b\mathbf{w}_i = \lambda_i\mathbf{S}_w\mathbf{w}_i\,, \tag{6.21}$$

whereby λ_i is the eigenvalue associated with the eigenvector $\mathbf{w}_i$.

The most significant *p* eigenvectors are chosen as projection bases with the *p* largest eigenvalues. Since the ranks of $\boldsymbol{S_w}$ and $\boldsymbol{S_b}$ are at most *d*, the dimensionality of the space Y can be no larger than *d*, so making LDA another approach of dimension reduction.

Therefore, LDA is often used for speech and speaker recognition to project a feature space into a discriminative and also compact space. For a discussion of related research work, see Chapter 11.

6.2.3 Nonlinear discriminant analysis

Although linear discriminant analysis can enhance feature discrimination, there are clear limitations to its mapping capacity. When classes are nonlinearly separable, LDA is incapable of enhancing class separation and nonlinear transformations are therefore required. While there are alternative ways to implement nonlinear transformations (e.g. SOM-based transformation), the multi-layer preceptron (MLP)-based approach is one of the most powerful and efficient ones.

A multi-layer perceptron (MLP) is a neural net classifier often used in pattern recognition (Duda et al. 2001; Fontaine et al. 1997). Initially proposed by Frank Rosenblatt in 1958 (Rosenblatt 1962), MLP is a feed-forward network without any loop or feedback from a successive layer to any preceding layer. Therefore, it is also called a non-recurrent neural net. An MLP consists of any number of layers with any number of units in each layer. The first layer of MLP is generally called the input layer, while the last layer is referred to as the output layer. Any layer between the input and the output layer is referred to as a hidden layer. An MLP with L hidden layers is often called an L-hidden-layer MLP, or simply L-layer MLP, or sometimes an “L+2-layer MLP”. Each unit in layer n normally has forward connections to each unit in layer n+1.

Given the varying numbers of hidden layers and unit types in an MLP, both linear and nonlinear discriminative transformations can be trained. An MLP without hidden layers (Figure 6-1a) is always a linear MLP (LMLP), which is theoretically equivalent to LDA (Duda et al 2002). A one-hidden-layer MLP with a sufficient number of hidden units with sigmoid activation functions can theoretically approximate any kind of smooth nonlinear function. However, with more hidden layers in an MLP, a mapping function can be approximated at the same level of accuracy, with smaller weights for each link and less hidden units in each layer, while at the same time the MLP can be more efficiently trained (Bishop 1995).

The basic approach for MLP training is by error-gradient descent, using the algorithm of "error back-propagation", or simply "back-propagation". It is a training approach which uses gradient descent, which is calculated using a recursive propagation of the error gradient calculation backwards from the output layer through each layer in the network. Its formulation is given briefly as follows (see also Morris 1992):

Notation:

Each training example is a pair of input/output pattern vectors, (x_p, t_p).

p is the index of a training example pair.

s_{pi} is the output from unit i for pattern p.

w_{ij} is the weight at unit j for the link from unit i.

E is the sum of square errors between the set of output vectors Y actually produced by the network and the target set $T = \{t_p\}$, i.e.

$$E_p = \sum_{i=1}^{n_y} \left(t_{pi} - s_{pi}\right)^2 . \tag{6.22}$$

Define $net_{pj} = \sum_i x_{pi} w_{ji} + w_{j0}$,

$$E = \sum_p E_p \Rightarrow \nabla_W E = \nabla_W \sum_p E_p = \sum_p \nabla_W E_p$$

$$\frac{\partial E_p}{\partial w_{ij}} = \frac{\partial E_p}{\partial net_{pj}} \cdot \frac{\partial net_{pj}}{\partial w_{ij}} ; \frac{\partial net_{pj}}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} \left(\sum_i w_{ij} s_{pi} \right)$$

$= s_{pi}$ (where s_{p0} is the bias "output" which is always 1)

Define $\delta_{pj} = -\frac{\partial E_p}{\partial net_{pj}}$, so that $\frac{\partial E}{\partial w_{ij}} = \sum_p \frac{\partial E_p}{\partial w_{ij}} = -\sum_p \delta_{pj} s_{pi}$

$$\delta_{pj} = -\frac{\partial E_p}{\partial net_{pj}} = -\frac{\partial E_p}{\partial s_{pj}} \cdot \frac{\partial s_{pj}}{\partial net_{pj}}$$

$$\frac{\partial s_{pj}}{\partial net_{pj}} = \frac{\partial f(net_{pj})}{\partial net_{pj}} = f'(net_{pj})$$

$$\frac{\partial E_p}{\partial s_{pj}} = \frac{\partial E_p}{\partial net_p}\cdot\frac{\partial net_p}{\partial s_{pj}} = \sum_k \frac{\partial E_p}{\partial net_{pk}}\cdot\frac{\partial net_{pk}}{\partial s_{pj}} = \sum_k \frac{\partial E_p}{\partial net_{pk}}\cdot\frac{\partial}{\partial s_{pj}}\left(\sum_i w_{ik}s_{pi}\right)$$

$$= \sum_k \frac{\partial E_p}{\partial net_{pk}} w_{jk} = \sum_k \delta_{pk} w_{jk}$$

$$\Rightarrow \delta_j = \sum_p \delta_{pj} = \sum_p f'(net_{pj}) \sum_k \delta_{pk} w_{jk}$$

This result does not only apply to output units j, when w_{jk} is undefined, but to all other units j, δ_{pj} is now obtainable from the δs (deltas) for those units which unit j sends outputs to. For output units the deltas can be obtained directly as follows:

$$\frac{\partial E_p}{\partial s_{pj}} = \frac{\partial}{\partial s_{pj}} \sum_i (t_{pi} - s_{pi})^2 = \sum_i 2\left(t_{pi} - s_{pi}\right)\left(-\frac{\partial s_{pi}}{\partial s_{pj}}\right) = 2\left(t_{pj} - s_{pj}\right)$$

$$\Rightarrow \delta_j = \sum_p \delta_{pj} = 2\sum_p f'(net_{pj})(t_{pj} - s_{pj})$$

For the commonly used activation function *f(x) = 1/(1 + exp(-x))*:

$$f'(x) = f(x)(1 - f(x)) \Rightarrow f'(net_{pi}) = f(net_{pi})(1 - f(net_{pi})) = s_{pi}(1 - s_{pi})$$

If this unit function is used throughout the network, then all δs are obtained by:

$$\delta_{pj} = 2s_{pi}(1 - s_{pi})(t_{pj} - s_{pj})$$

$$\delta_{pj} = s_{pi}(1 - s_{pi}) \sum_k \delta_{pk} w_{jk}$$

The error gradient with respect to each weight in the network can now be calculated by first calculating the deltas for the output layer, then for the layer before this, followed by the layer before this, and so on.

The above derivation adopts the least squares cost function (6.22) as the training criterion, but this is not the only cost function which can be used. Besides this, other cost functions exist, e.g. the cross-entropy cost function as defined by

$$E = \sum_p E_p = -\sum_p \sum_i \left(t_{pi} \ln s_{pi} + (1 - t_{pi})\right) \ln(1 - s_{pi}) .$$

The training procedure using the cross-entropy cost function is very similar to the procedure described above, except that $\frac{\partial E_p}{\partial s_{pj}}$ is replaced by the following formula

$$\frac{\partial E_p}{\partial s_{pj}} = \frac{\partial \sum_i \left(t_{pi} \ln s_{pi} + (1 - t_{pi})\right) \ln(1 - s_{pi})}{\partial s_{pj}}$$

$$= \frac{t_{pj}}{s_{pj}} \ln(1 - s_{pj}) - \frac{t_{pj}}{1 - s_{pj}} \ln s_{pj} - \frac{1 - t_{pj}}{1 - s_{pj}} .$$

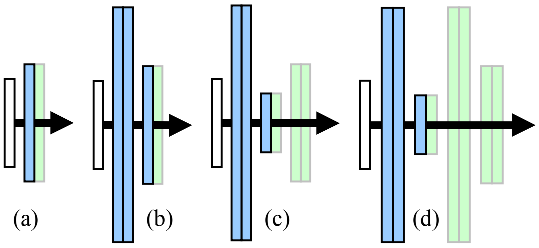


Figure 6-1: Four MLPs (a, b, c, d) types. Each active layer is shown as a [net-input function | non-linear activation function] sandwich. Dark sections of each MLP are used in data projection, light parts only in MLP training

When an MLP is used for feature transformation, the probabilities learned from the input to or output from one of its hidden layers or the input to its output layer, i.e. the "pre-squashed MLP outputs" (see Figure 6-12), can be obtained as discriminative internal representations (features) for learning targets. MLP (b) is the standard MLP used for feature projection in speech recognition (Hermansky et al. 2000), whereas with the more-hidden-layer MLPs used for feature projection, the more possibilities are there to obtain the optimal internal representations. In order to derive discriminative features, two steps are required. In the first step all the layers of MLP are used for training. In the second step only parts of it (the dark section shown in Figure 6-1) are used for projection. The discriminative features are obtained from these outputs.

6.3 Summary

In this chapter, the three most important approaches used in feature transformation (linear and nonlinear transformations) were summarised. Among these, PCA aims to reduce the correlated redundancy of a feature space, so that it maps the original feature space to a lower-dimensional orthogonal space. LDA tries to separate classes by maximising the between-class variance and at the same time minimising the within-class variance. Finally, the more powerful nonlinear transformations implemented by MLPs were presented. Discriminative features can be obtained from any layer in an n-layer MLP. The detailed information will be further discussed in Chapter 11.

7. Data modelling

7.1 Introduction

Data modelling is a crucial stage in speaker recognition. A model for each person enrolled in the speaker recognition system, referred to as a client model, is required to describe the distinct distributional characteristics of data from this client. Particularly in verification tasks, an alternative model is also needed to represent the distribution of all the non-client data.

There are two types of data models, template models and stochastic models, both of which may be used for speaker recognition. In template modelling, a template is chosen in the recognition process based on the minimal distance between a given sequence of input samples and the template's frames. As it is based on this distance measure instead of probability, the template-based matching approach is deterministic. By contrast, the stochastic model-based matching approach is probabilistic, as it makes decisions based on a measure of the class likelihood, or conditional probability, of the observation given by the model. Compared with template modelling, stochastic modelling enables us to capture more (or more delicate) information on the distribution of the data samples. Therefore, stochastic models are more powerful than template models, especially when used for text-independent speaker recognition. But also in text-dependent speaker recognition in clean speech, template models achieve only moderate performance. In this chapter, we will briefly introduce two template-based matching approaches and two stochastic model-based matching approaches (in Sections 7.2 and 7.3).

Verification is a binary classification task in which the claimed speaker or the non-claimed speaker is selected. To model the claimed speaker, modelling approaches such as those mentioned above can be used, as in identification. However, to model the non-claimed speaker, special techniques are required, e.g. universal background modelling (UBM) or cohort modelling. In Section 7.4, we will briefly address these techniques.

The rest of this chapter is organised as follows: In Section 7.2, template-based matching approaches are addressed. In Section 7.3, stochastic models are overviewed, including GMM and HMM. In Section 7.4, verification background modelling is presented, followed by a summary in Section 7.5.

The following discussions are based in the main on Campbell et al. (1997).

7.2 Template-based matching approaches

Template-based matching uses template models to describe the characteristics of the target models. For a given prompt with a sequence of frames $\{\mathbf{x}_i\}$, $i = 1 \ldots N$ where N is the number of frames, the template model for a speaker consists of a template sequence $\overline{\mathbf{x}} = \{\overline{\mathbf{x}}_i\}$ with the same length. The template model $\overline{\mathbf{x}}$ can be trained using a set of K training vectors

$$\overline{\mathbf{x}}_i = \frac{1}{K}\sum_{i=1}^{K}\mathbf{x}_i \,. \tag{7.1}$$

In recognition, many different distance measures between the vectors $\mathbf{x}_i$ and $\overline{\mathbf{x}}_i$ can be represented as

$$d\left(\mathbf{x}_i,\overline{\mathbf{x}}_i\right) = \left(\mathbf{x}_i - \overline{\mathbf{x}}_i\right)^T \mathbf{W}\left(\mathbf{x}_i - \overline{\mathbf{x}}_i\right), \tag{7.2}$$

whereby ***W*** is a weighting matrix. If ***W*** is an identity matrix, the distance is Euclidean. If ***W*** is an inverse covariance matrix corresponding to mean $\overline{\mathbf{x}}_i$, then it is a Mahalanobis distance. The Mahalanobis distance gives less weight to the components with more variance and is equivalent to a Euclidean distance on principal components, which are the eigenvectors of the original space, as determined from the covariance matrix (Campbell et al. 1997; Duda et al. 2001).

When the speaking-rate variability is taken into account, the frame number in the input sequence $\{\mathbf{x}_i\}$, $i = 1 \ldots N$ may not be the same as the number of reference vectors in the speaker template model $\{\overline{\mathbf{x}}_i\}$, $i = 1 \ldots M$, where $M \neq N$. In such a case, the above simplest matching process can therefore not be conducted. In order to compensate the effect of speaking-rate variability on template-based matching, dynamic time warping is required.

7.2.1 Dynamic time warping

Dynamic time warping can solve the above problem, which is caused by time variation in human speech. The asymmetric match score *z* is given by

$$z = \sum_{i=1}^{N} d\left(\mathbf{x}_i, \overline{\mathbf{x}}_{j(i)}\right), \tag{7.3}$$

whereby the template indices *j(i)* are generally given by a DTW algorithm. On the basis of the template vectors and input signals, the DTW algorithm carries out a constrained, piece-wise linear search, beginning from a certain point and ending at another (Figure 7-1). This is a technique of dynamic programming which searches the optimal path in a grid labelled with the indices of a given template model and a sequence of data vectors (labelled by their time indices) (Figure 7-1).

At the end of time warping, the accumulated distance gives an estimation of the match score according to a particular mapping function. This method accounts for the variation over time (trajectories) of the speech features, which are corresponding to the dynamic configuration of the articulators and vocal tract. In practice, the Sakoe slope constraints of the warp can be used to act as boundary conditions to prevent excessive warping over a given segment (Campbell 1997).

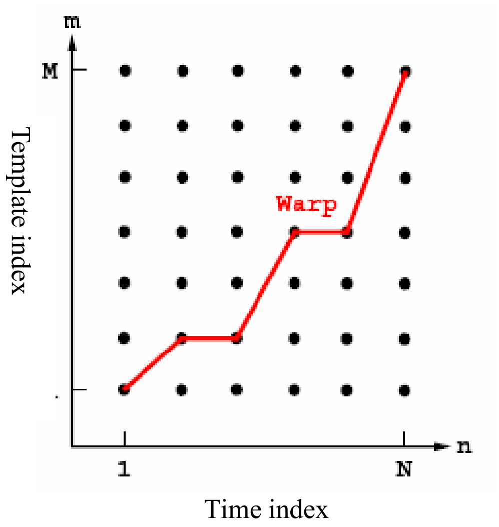


Figure 7-1: Schematic illustration of dynamic time warping

7.2.2 VQ source modelling

Another type of template model, referred to as vector quantisation (VQ) source modelling, uses multiple templates to represent frames of speech without considering their temporal correlation (Campbell 1997). A VQ codebook is created by standard clustering procedures for each enrolled speaker using his training data. As the template match score is taken as the distance between an input vector and the minimum distance codebook vector in the VQ codebook C, this score for L frames of speech is then defined as

$$z=\sum_{i=1}^{L}\min_{\bar{\mathbf{x}}\in C}\left\{d\left(\mathbf{x}_i,\bar{\mathbf{x}}\right)\right\}. \tag{7.4}$$

The clustering procedure used to train the codebook may be any one of the standard clustering algorithms, such as K-means clustering, which don’t take in consideration temporal information contained in a frame sequence. Hence, there is no need to perform a time alignment. However, the neglect of the speaker-dependent temporal information that is present in the utterance may cause a reduction of system performance.

7.3 Stochastic model based approaches

7.3.1 GMM-based approach

GMM is the state-of-the-art data modelling approach for speaker recognition (Reynolds et al. 1994, 1995a, 1995b, 1995c, 2000). It statistically models the data distribution by means of clustering techniques, followed by expectation maximisation (EM) training algorithm. Each cluster in the feature space is described by a Gaussian density function (a normal distribution). The overall characteristics of the feature space are then captured by a linear combination of all the Gaussian components. Formally, the GMM-based framework for speaker recognition can be expressed as follows.

In GMM training a GMM data pdf *p(***x***|S)* (7.5) is trained for each speaker for a fixed number of Gaussians *M*. The GMM models the pdf for a data frame, $\mathbf{x_t}$ (where *t* is the time index), taking no account of the time order of the data frames in the full speech sample **X**.

$$p(\mathbf{x}_t \mid S) = \sum_{i=1}^{M} w_i p(\mathbf{x}_t \mid G_i, S), \tag{7.5}$$

whereby $G_i = N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, where N(.) is the Gaussian or Normal function with the mean $\boldsymbol{\mu}_i$ and a diagonal covariance matrix $\boldsymbol{\Sigma}_i$, which has δ_i^2 as its diagonal components.

Hence,

$$p(\mathbf{X} \mid S) = \sum_t \sum_{i=1}^{M} w_i p(\mathbf{x}_t \mid G_i, S). \tag{7.6}$$

The training of a GMM consists of two steps. In the first step, the model parameters (the class means) are initialised using the K-means clustering algorithm which captures the coarse characteristics of the data clustering from a speaker. Each δ_i of $\boldsymbol{\Sigma}_i$ is initialised with a small random value. The mean vector $\boldsymbol{\mu}_i$ is updated (the j-th updating) by

$$\boldsymbol{\mu}_i^{(j)} = \frac{1}{N_i^{(j)}} \sum_{\mathbf{x} \in C_i} \mathbf{x}, \quad 1 \le i \le K, \tag{7.7}$$

where $N_i^{(j)}$ is the number of samples in class C_i. The class index a sample **x** belongs to is decided according to

$$i = \arg\max_i d(\mathbf{x}, \boldsymbol{\mu}_i^{(j-1)}). \tag{7.8}$$

In the second step, the expectation maximization (EM) algorithm is applied to optimise the model parameters (mixture weights, means and variances) so as to maximise the data likelihood *P(X|S)*.

Mixture weights:

$$w_i = \frac{1}{T} \sum_{t=1}^{T} p(i \mid \mathbf{x}_t, S) \tag{7.9}$$

Means:

$$\boldsymbol{\mu}_i = \frac{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, S) \mathbf{x}_t}{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, S)} \tag{7.10}$$

Variances:

$$\delta_i^2 = \frac{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, S) \mathbf{x}_t^2}{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, S)} - \boldsymbol{\mu}_i^2 \tag{7.11}$$

The a *posteriori* probability for acoustic class *i* is given by

$$p(i \mid \mathbf{x}_t, S) = \frac{w_i p(\mathbf{x}_t \mid G_i, S)}{\sum_{k=1}^{M} w_k p(\mathbf{x}_t \mid G_k, S)}. \tag{7.12}$$

In fact, as the GMM may be regarded as a single state HMM, the temporal correlation of the frame sequence cannot be modelled by this technique. Experiments examining the performance of HMM and GMM suggest that the temporal information of the frame sequence is not important for the distinction between speakers.

7.3.2 HMM-based approach

Hidden Markov modelling (HMM) is an alternative modelling approach also used in speaker recognition. A continuous left-to-right HMM (Figure 7-2) is normally used to capture the temporal characteristics of a sequence of frames from a given prompt, since a speech prompt can be modelled through a double stochastic process. This process is characterised by a given number of states with an associated set of transition probabilities among them. In each state, a continuous density, a multivariate Gaussian mixture, is used to model the emission probability density. The HMM is a finite-state machine in which a pdf $p(\mathbf{x}_t \mid s_i)$ is associated with each state s_i. These states are connected by a transition network, where the state transition probabilities are $a_{ij} = p(s_i \mid s_j)$, where i and j are state indices. Other topologies for HMM than the continuous left-to-right can also be used (e.g, with skip transitions from one state to a later state other than the next one), but they were showed not to be superior (Rabiner et al. 1993).

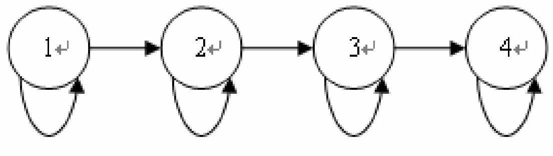


Figure 7-2: HMM topology

The probability that a sequence of speech frames is generated by this model can be determined by means of Baum-Welch decoding (Rabiner et al. 1993). This likelihood corresponds to the score for L frames of input speech, given the model

$$p(\mathbf{x}(1;L) \mid S) = \max_{\substack{\text{all state}\\ \text{sequence}}} \prod_{t=1}^{L} p(\mathbf{x}_t \mid s_j) a_{j,j-1}. \tag{7.13}$$

7.4 Verification background modelling

As mentioned earlier, verification is a binary classification problem in which the claimed speaker is differed from the non-claimed speaker. As with the techniques used for modelling a speaker in identification, the same approaches can be used for modelling the claimant data. However, there are different techniques to model the non-claimed model, because this model should capture the characteristics to enable to represent all speakers other than the claimed one. Two solutions to this problem are universal background modelling (UBM) and cohort modelling.

The UBM uses a single, speaker-independent background model to represent impostors in terms of all the claimed speakers. However, the cohort modelling creates a background model for each claimant, which represents the population of expected impostors for each claimant. Ideally, the number of background speakers should be as large as possible to better model the impostor population, but practical considerations of computation and storage restrict the number of background speakers. In Reynolds et al. (1995a), the number of background speakers was set to ten. Given the size of the final background speaker set B, each speaker's *N* closest or farthest neighbours were selected as his/her "close/far cohort", according to the pair-wise distance measure

$$d(\lambda_i,\lambda_j)=\log\frac{p(X_i\mid\lambda_i)}{p(X_i\mid\lambda_j)}+\log\frac{p(X_j\mid\lambda_i)}{p(X_j\mid\lambda_j)} \tag{7.14}$$

for speaker *i* and *j* with models (λ_i,λ_j) and training utterances (X_i,X_j).

An alternative distance was suggested by Zigel et al. (2003):

$$d(\lambda_i,\lambda_j)=\frac{1}{2}\log p(X_i\mid\lambda_i)+\frac{1}{2}\log p(X_j\mid\lambda_i) \tag{7.15}$$

UBM can also be used for score normalisation as

$$score(X)=\log\left(\frac{p(X\mid\lambda_{claimed})}{p(X\mid\lambda_{UBM})}\right)=\log\left(p(X\mid\lambda_{claimed})\right)-\log\left(p(X\mid\lambda_{UBM})\right). \tag{7.16}$$

When the background model is trained by speakers other than the target speakers uttering general text-independent utterances (text-independent tasks) or the user's phrase (text-dependent tasks), $p(X\mid\lambda_{UBM})$ represents a dynamic threshold which is sensitive to variations in *X* from trial to trial (Zigel et al. 2003).

7.5 Summary

Data modelling is one of the crucial steps in speaker recognition. Firstly, in this chapter, different techniques for data modelling in speaker recognition were presented. Template-based modelling approaches utilise a template pattern to store the necessary discriminating information for a speaker. Due to the limited capacity of modelling data variations, template-based approaches are more successful in text-dependent applications. By contrast, statistical modelling approaches, such as the GMM-based and HMM-based approaches, are advantageous in modelling more variations in the data distribution of a speaker caused by difficult conditions, e.g. in text-independent applications. Therefore, statistical modelling techniques represent the state-of-the-art modelling techniques for speaker recognition. The relevant techniques were overviewed in this chapter. Secondly, some special modelling techniques for speaker verification were summarised. In speaker identification a client model is indispensable for representing the characteristics of the speaking style of a given speaker (a client). In verification, besides modelling the claimed speaker, the UBM or cohort model is also required for modelling possible impostors. These techniques were explained in this chapter as well.

8. Decision theory

Based on the techniques described in previous chapters, a match score between the input speech-feature vector and a given speaker model is derived. Following this, the next question is how to make the final decision for recognition.

Bayesian decision theory provides the key to decision making in speaker recognition. In this chapter, we will therefore first introduce the relevant aspects of this theory. Following this, Bayesian decision theory is applied to derive decision formulae for both identification and verification.

With respect to verification, the likelihood ratio test and two types of the resulting curves, often met in the literature, are explained.

Score normalisation is also very useful in speaker verification to alleviate the impact of different variance scaling in the dataset of each speaker. T-norm and Z-norm are the two main methods used for this purpose. In this chapter, we briefly describe the principles of these two normalisations.

The rest of this chapter is organised as follows: In Section 8.1 Bayesian decision theory is presented. In Sections 8.2 and 8.3, the formulae describing the decision making of identification and verification are derived from Bayesian decision rule. In Section 8.4, score normalisation techniques are presented, followed by a summary in Section 8.5.

8.1 Bayesian decision theory

Bayesian decision theory is used in both speaker identification and verification. In later sections we will show that the decision procedures used are derived by the concrete application of the Bayesian decision rule. Therefore, theoretical knowledge of Bayesian decision is helpful for an understanding of the essential ideas of the decision procedure in speaker recognition.

Given a multi-class $\{\omega_i\}$, $1 \le i \le c$, classification problem and the posterior probability $p(\omega_i \mid X)$ for each class ω_i and a sample *X*, the following decision strategy is referred to as the Bayesian decision rule (Theodoridis et al. 2003):

$$\text{decide } \omega_i \text{ if } P(\omega_i \mid X) > P(\omega_j \mid X) \text{ for all } j \ne i. \tag{8.1}$$

The Bayesian decision rule is the decision with the least average risk (Bayesian risk or expected loss), if the risk function is defined by

$$\lambda(\alpha_i \mid \omega_j) = \begin{cases} 0 & i = j \\ 1 & i \ne j \end{cases} \qquad i, j = 1, \ldots, c, \tag{8.2}$$

whereby c is the overall number of classes and α_i relates to an action which is always interpreted as a decision. If decision α_i is taken and the true state of nature is ω_i, then the decision is correct if *i=j*, otherwise it is wrong, i.e. *i≠j*.

[Proof]
Given a data *X*, the average decision risk can be defined as

$$\begin{aligned} R(\alpha_i \mid X) &= \sum_{j=1}^{c} \lambda(\alpha_i \mid \omega_j) P(\omega_j \mid X) \\ &= \sum_{j \ne i} P(\omega_j \mid X) \qquad . \\ &= 1 - P(\omega_i \mid X) \end{aligned} \tag{8.3}$$

Thus, in order to minimise the average risk for simple classification, it is necessary to select the class by the maximum of $P(\omega_i \mid X)$, which complies with (8.1).

Several theoretical results are also related to Bayesian decision, although they are of a more theoretical interest. As is generally known, it is very difficult to analytically check the error bound of Bayesian decision. However, in the case of a two-class classification with the conditional distribution being a multivariate Gaussian distribution, the error bound can be estimated theoretically by the Chernoff and computationally simpler Bhattacharyya bounds.

The Chernoff bound is derived as

$$P(error) \le P^{\beta}(\omega_1) P^{1-\beta}(\omega_2) \int P^{\beta}(X \mid \omega_1) P^{1-\beta}(\omega_2) dX \quad \text{for } 0 \le \beta \le 1, \tag{8.4}$$

which states that the theoretical error bound is inferior to a certain value in turn related to a variable β. Under normal conditional probabilities, the integral in (8.4) can be evaluated analytically, yielding

$$\int p^{\beta}(X \mid \omega_1) p^{1-\beta}(X \mid \omega_2) dX = e^{-k(\beta)}, \tag{8.5}$$

where

$$\begin{aligned} k(\beta) &= \frac{\beta(1-\beta)}{2}(\mu_2-\mu_1)^t\left[\beta\Sigma_1+(1-\beta)\Sigma_2\right]^{-1}(\mu_1-\mu_2) \\ &+\frac{1}{2}\ln\frac{\beta\Sigma_1+(1-\beta)\Sigma_2}{|\Sigma_1|^{\beta}|\Sigma_2|^{1-\beta}} \end{aligned} . \qquad (8.6)$$

By setting $\beta = 1/2$, a computationally simpler but slightly less tight bound can be derived, giving the so-called Bhattacharyya bound on the error

$$\begin{aligned} P(error) &\le \sqrt{P(\omega_1)P(\omega_2)}\int P(X\mid\omega_1)P(X\mid\omega_2)dX \\ &= P(\omega_1)P(\omega_2)e^{-k(1/2)}, \end{aligned} \qquad (8.7)$$

where

$$k(1/2) = 1/8(\mu_1-\mu_2)^t\left[\frac{\Sigma_1+\Sigma_2}{2}\right]^{-1}(\mu_1-\mu_2)+\frac{1}{2}\ln\frac{\left|\frac{\Sigma_1+\Sigma_2}{2}\right|}{\sqrt{|\Sigma_1||\Sigma_2|}}. \qquad (8.8)$$

The Chernoff and Bhatacharyya bounds may still be used even if the underlying distributions are not quite Gaussian. However, for distributions that deviate markedly from Gaussian ones, the bounds are generally not informative (Duda et al. 2001).

Although the error bounds are rarely useful for practical applications, certain insights may nevertheless be obtained by applying Bayesian decision rule to speaker recognition.

Bearing in mind Bayesian decision theory, we now proceed to see how it is applied to speaker identification and verification.

8.2 Identification decision procedure

The decision procedure in speaker identification is the outcome of directly applying Bayesian decision rule (8.1) by replacing ω_i by S_i. This means that the identified speaker is chosen as the one holding the maximum probability among the candidate speaker models S_i, given a speech input **X**.

$$S_i = \arg\max_k p(S_k\mid\mathbf{X}) = \arg\max_k \frac{p(\mathbf{X}\mid S_k)p(S_k)}{P(X)} = \arg\max_k p(\mathbf{X}\mid S_k), \qquad (8.9)$$

as $p(S_k)$ and $p(\mathbf{X})$ same for all *k*.

When the maximum probability density is below a fixed threshold (i.e. $\max p(\mathbf{X}\mid S_i) < \theta$), no speaker is chosen; Otherwise a decision is made according to (8.9). Thus, the decision rule for identification is relatively simple.

8.3 Verification decision procedure

In verification the Bayesian decision rule (8.1) is applied to the two-class case. More explicitly, given a speech input *X*, a claimed speaker model S_c and an impostor model $S_{impostor}$, the decision is to check such a ratio as

$$ratio = \frac{p(S_c \mid X)}{p(S_{impostor} \mid X)} \quad . \tag{8.10}$$

If $ratio \geq \theta$ (a threshold), the speaker is accepted as the claimant; Otherwise he is rejected.

If we apply the Bayesian rule to the above decision criterion(8.10), we obtain

$$ratio = \frac{p(S_c \mid X)}{p(S_{impostor} \mid X)} = \frac{p(X \mid S_c)p(S_c)}{p(X \mid S_{impostor})p(S_{impostor})} = \frac{p(X \mid S_c)}{p(X \mid S_{impostor})}, \tag{8.11}$$

if $p(S_c) = p(S_{impostor})$.

$p(X \mid S_c)$ and $p(X \mid S_{impostor})$ are likelihoods, therefore (8.11) is also referred to as a likelihood ratio test.

In practice, verification assigns more flexible costs *CFR* (cost of false rejection) and *CFA* (cost of false acceptance) to the posterior probabilities in eq. (8.11) in order for the service provider to adjust and control the required preference to either low false rejection rate or low false acceptance rate or a balance between them, i.e.

$$ratio = \frac{p(X \mid S_c) \cdot CFR}{p(X \mid S_{impostor}) \cdot CFA}. \tag{8.12}$$

8.3.1 Likelihood ratio test

Generally speaking, a likelihood ratio test (or hypothesis testing) is a test which relies on a test statistic computed by taking the ratio of the maximum value of the likelihood function under the constraint of the null hypodissertation (Θ_0) to the maximum with that constraint relaxed (Θ_1). A likelihood function is in fact a statistical model which is often a parameterized family of probability density functions or probability mass functions $f_\theta(x)$. A null hypothesis is often stated by saying the parameter θ is in a specified subset Θ_0 of the parameter space Θ. The likelihood function $L(\theta) = L(\theta \mid x) \cong p(x \mid \theta) = f_\theta(x)$ is a function of the parameter θ given the data x. The likelihood ratio is

$$\Lambda(x) = \frac{\sup\{L(\theta \mid x) : \theta \in \Theta_0\}}{\sup\{L(\theta \mid x) : \theta \notin \Theta_0\}}. \tag{8.13}$$

This is a function of the data x and is therefore a statistic. The likelihood-ratio test rejects the null hypothesis if the value of this statistic is below a given threshold (Wikipedia 2000).

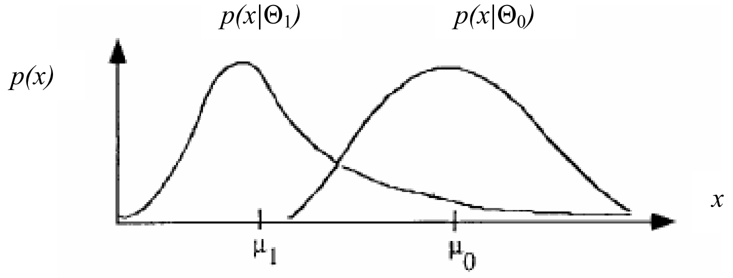


Figure 8-1: Sample of score densities

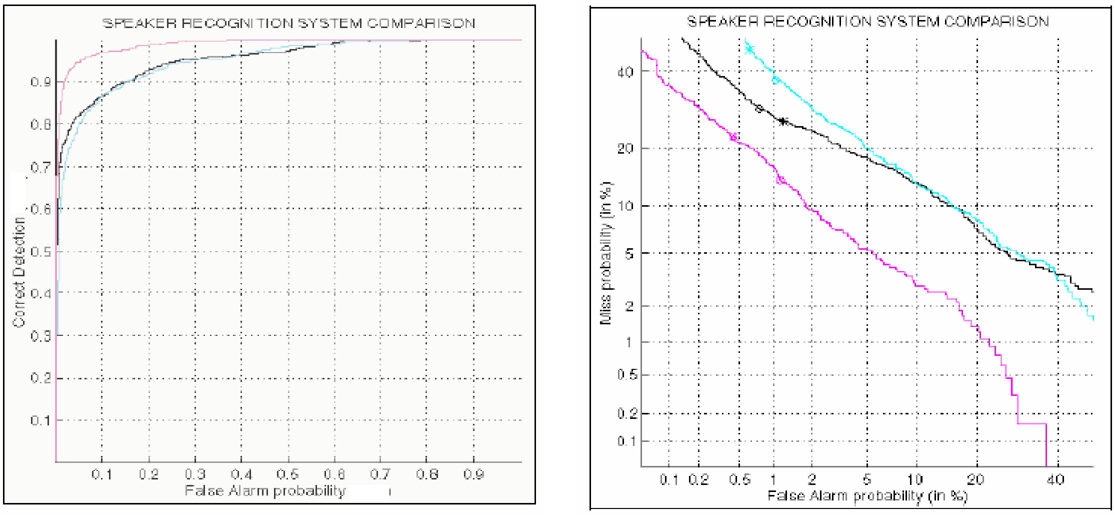
Figure 8-1 shows an example of two score pdf's. The probability of error, minimised by Bayesian decision rule, is determined by the amount of overlap in the two pdf's. The smaller the overlap between two pdf's, the smaller the probability of error. The overlap in two Gaussian pdf's with the mean μ_0, μ_1 and the equal variance σ can be measured by the F-ratio

$$F = \frac{(\mu_0 - \mu_1)^2}{\delta^2}. \tag{8.14}$$

8.3.2 DET vs. ROC curves

Detection error trade-off (DET) and receiver operating characteristic (ROC) curves are often encountered in the literature. We therefore give a short description of these concepts. There are two types of errors in speaker verification, false acceptance (FA) and false rejection (FR). Either of the two types of errors can be reduced at the expense of an increase in the other. A single performance number is therefore inadequate to represent the capabilities of a system. Such a system has many operating points (a different point for each given value of the false acceptance threshold) and is best represented by a performance curve.

The ROC curve plot is one of the approaches used for this purpose. It obtains the alarm rate (false acceptance), which is plotted on the horizontal axis, and the correct detection rate, plotted on the vertical axis (Figure 8-2a).



(a) (b)

Figure 8-2: (a) Plot of ROC curves for a speaker recognition evaluation. (b) Plot of DET curves for the same evaluation data (from Martin et al. 1997)

The DET curve plot is a more recently applied visualisation of speaker verification results. It plots the error rates (false acceptance or false alarm, and false rejection or miss) on both axes, giving uniform treatment to both types of error and using a logarithmic scale for both axes. The error rates are spread out across the plot, distinguishing more clearly between the different well-performing systems and producing plots that are close to linear (Figure 8-2b).

8.4 Score normalisation

Decision scores of different models, derived either from likelihood (8.11) or from posterior probability(8.10), always vary in scaling ranges, these differences caused by the mismatch between training and test conditions. For instance, in text-independent recognition, the linguistic difference between the training and test set often leads to a certain mismatch. Background noise changes, such as different types and levels of noise as well as crosstalk, also frequently result in a mismatch between training and test conditions.

The different scaling ranges between the models under comparison represent an impediment to improving model discrimination, because they cause a strong overlap between two different pdfs (Figure 8-1). Moreover, the diversity of the scaling ranges of the match scores renders the global thresholding scheme less effective. A frequently applied solution to this problem uses the techniques of score normalisation. Two widely applied score normalisations are zero normalisation (Z-norm) and test normalisation (T-norm) (Auckenthaler et al. 2000).

8.4.1 Z-norm

Zero normalisation (Z-norm) is a normalisation technique which uses a mean and variance estimation for distribution scaling. This normalization has the form

$$Score_{Norm} = \frac{p(S\,|\,\mathbf{X}) - \mu_I}{\sigma_I}, \tag{8.15}$$

whereby μ_I and σ_I are the estimated impostor parameters derived from the UBM or cohort model (cf. Chapter 7) for speaker model S and $Score_{Norm}$ is the normalised score distribution.

By applying Z-norm to the match score $p(S\,|\,\mathbf{X})$, the distribution of the match score is transformed to the same scaling as that of the impostor distribution. Therefore, the comparison of the match scores between the claimed distribution and the impostor distribution is more effective.

The advantage of Z-norm is that estimations of the normalisation parameters can be performed off-line during training. A claimed speaker model is tested against example impostor (cohort speaker) utterances and log-likelihood scores are used to estimate a speaker-specific mean and variance for the impostor distribution.

8.4.2 T-norm

Another normalisation method which is also based on mean and variance estimation for distribution scaling is test normalization (T-norm). This type of normalisation is an online method which is different from Z-norm in that the impostor model parameters are estimated directly from the test set (Z-norm uses the training set). During testing, a set of example impostor models is used to calculate impostor log-likelihood scores for a test utterance, in a manner similar to a cohort approach. However, unlike the cohort approach, both a mean and a variance parameter are estimated from these scores. These parameters are then used to perform the distribution normalisation according to (8.15).

The advantage of T-norm over Z-norm is the use of the variance parameter which approximates the distribution of the impostor population more accurately. The estimation of these distribution parameters is carried out on the same utterance as in the target speaker test. Thus, an *acoustic mismatch* between the test utterance and normalisation utterances, possible in Z-norm, is avoided.

8.5 Summary

This chapter focused mainly on decision theory, the last stage of speaker recognition. Taking the introduction of Bayesian decision theory as a starting point, Bayesian decision rule was applied to the derivation of the decision formulae for speaker identification and verification. This fundamentally reflects the essence of the application of statistical modelling to speaker recognition not only in the modelling stage, but also in the decision stage. In addition, a number of verification related issues, such as likelihood ratio test and two useful curves (ROC and DET), were described. Likelihood ratio test is the theoretical foundation for the decision process of verification, whereas ROC and DET curves are the popular measurements to evaluate the performance of standard verification systems. Especially, DET curve is advantageous over ROC in its uniform treatment of two types of error rates (false acceptance and false rejection), so that it provides more reasonable visualisation to the performance comparison of different verification systems. In the final section, score normalisation techniques (Z-norm and T-norm) were addressed. With these normalisations, the derived scores can be normalised in a similar range for facilitating a more accurate decision. To some extent, T-norm can lead to better system performance, since the normalised scores can maximally reduce the mismatch between the training set and test set, simply by estimating the normalisation parameters directly on the test set. However, its negative effect is that its normalisation parameters have to be estimated off-line.

Part III. Feature enhancement

9. Introduction

With an overview on biometric-based authentication and a theoretical introduction to speaker recognition, we have seen a general picture of speaker recognition. This can be summarised as follows: With the state-of-the-art techniques described in Part II (GMM and MFFCs), speaker recognition works successfully in clean speech (Reynolds et al. 1995, 2000). Its performance in wide-bandwidth clean speech has been shown to be close to 100%. This is very beneficial for the practical deployment of speaker recognition technologies.

However, the practical scenarios for applications of speaker recognition do not always allow clean speech conditions. A variety of noisy conditions, low-bandwidth speech and telephone speech all occur frequently in practical applications. Even the most successful speaker recognition technologies efficient for clean speech, therefore, meet with difficulties in low-bandwidth speech, telephone speech and speech under noisy conditions. Consequently, system performance substantially degrades under these conditions, this drawback significantly limiting the deployment of speaker recognition technologies for practical applications. The purpose of this dissertation is to find ways of improving speaker recognition performance in these difficult conditions.

This part of the dissertation is divided into three stages. Firstly, we examine the reasons why the performance of state-of-the-art speaker recognition systems degrades under the conditions named above by analysing the acoustic feature space in which the speaker-phoneme distribution (SPD) describing the acoustic space structure is discussed. The special structure of an acoustic space logically explains the difficulties in decision of the state-of-the-art speaker recognition systems, especially under noisy conditions. Secondly, based on this analysis, a general framework of NLDA-based feature enhancement by multi-layer perceptron (MLP) is suggested to reduce the problem. Experiments under a variety of conditions have shown the effectiveness of this general feature enhancement approach. Thirdly, an alternative feature type derived by SOM processing is found to provide complementary information for speaker recognition.

The reminder of Part III is organised as follows: In Chapter 10, an analysis of feature space is presented. MLP-based feature enhancement approaches are described in Chapter 11 and a complementary feature type generated by SOM processing is presented in Chapter 12.

10. Feature space analysis

10.1 Overview

The objective of feature space analysis presented here is to obtain a clearer understanding of the reasons for some of the drawbacks confronted by state-of-the-art speech processing systems such as low efficiency under noisy conditions (non-robustness) and the fact of the same types of features are used for both speech and speaker recognition. While the analysis developed may not directly solve these problems, it can provide us with a sound basis for the necessity of using the discriminative features for speaker recognition which will be discussed in Chapter 11.

MFCC features especially designed for speech recognition represent the most widely used feature type in speaker recognition. This implies that there must be a variety of useful information in MFCCs, related to linguistic content, speaker identity, speaker's gender and information pertaining to dialectal affiliation. Although speaker's gender and dialectal affiliation can be considered as a part of speaker identity, they are separate factors which can be identified in contributing to variation in acoustic patterns which give the speaker identity. The questions pertaining to these different sources of information are:

A. *Which type of information (linguistic, speaker identity, gender and dialectal affiliation) is most suitably represented by MFCC features, making the features correspondingly suitable for the recognition application?*

B. *Based on the question A, how are these types of information represented in feature space and what is the representative distributional structure for a feature space?*

C. *How can we utilise this structure for a particular recognition application, esp. speaker recognition?*

D. *Can this structure explain the lack of robustness of these features when they are used under noisy conditions?*

In this chapter we set out to find the answers to these four questions. The rest of this chapter is organised as follows: In Section 10.2, the speaker phoneme distribution (SPD) of a feature space is described and in Section 10.3, four analysis methods are presented. In Section 10.4, the data used for analysis are described. The results of the feature space analysis are given in Section 10.5, followed by conclusions, presented in Section 10.6.

10.2 Speaker-phoneme distribution (SPD) of a feature space

Throughout this section, an acoustic space corresponds to a voice space before feature extraction, while the term a feature space is used for a space after feature extraction.

A sentence with which a meaning is communicated is created by concatenating a sequence of phonemes. Phonemes are viewed as the basic sound units in an acoustic space. Therefore, a corresponding feature space derived from the acoustic space by means of signal processing (e.g. the MFCC feature space) also has phonemes as its basic units. Implicitly, the term "basic units" means their units may be separated (Property I).

Speakers can produce all the phonemes of a language. It can therefore be induced that speaker data probably contain all the phonemes in an acoustic space (Property II). Any feature space derived from an acoustic space may be assumed to also have this property.

Therefore, intuitively, a feature space should possess the above two properties (Property I and II), which are summarised and illustrated in Figure 10-1. In this figure, we can see the same or similar-sounding phonemes (broad phoneme classes) clustered together in a feature space. Within each phoneme (or broad phoneme class) speakers' locations overlap.

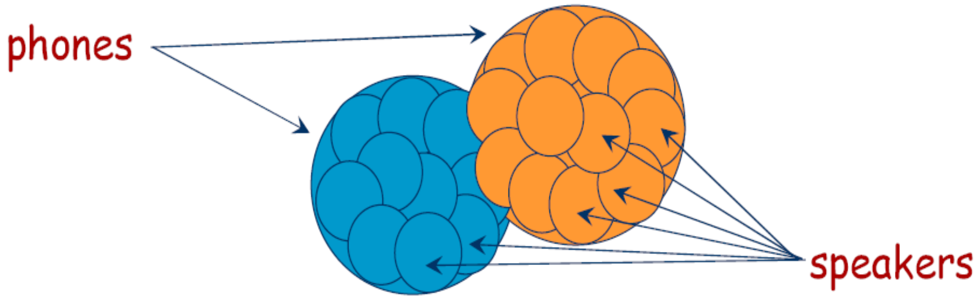


Figure 10-1: Speaker-phoneme distribution of a feature space, in which a big circle represents a phone cluster, a small circle in the big circles describes a speaker producing that phoneme sound

Although this SPD is simple, it has important implications for the triggering of new ideas for speaker recognition if it can be proved valid. Firstly, it is suggested on the basis of this model that MFCC features are more suitable for speech recognition than for speaker recognition, as phonemes are more easily separable than speakers (answering question A and the SPD answers question B). Secondly, more speaker discriminative features are required for speaker recognition in order to improve its performance (answering question C). Thirdly, the application of GMM to speaker modelling can be shown to be a reasonable, natural and valid method, because a Gaussian component is assigned within each phoneme (broad phoneme) cluster to model local characteristics of each speaker. Fourthly, the non-robustness of GMM can be more clearly understood on the basis of this model. Any other factor causing disturbance in the fine clustering of a feature space will have a negative impact on the performance of speaker recognition systems (within any phoneme, speaker clusters are already very close, so that any disturbance by noise could result in misclassification). Therefore, it is not difficult to understand that the performance of the GMM-based recognition systems significantly degrades in a variety of noisy conditions (answering question D).

In the following sections, the purpose is to apply different approaches to prove the validity of this SPD proposed in this section.

10.3 Methods used in speaker-phoneme distribution analysis

Since a common feature space is high-dimensional, it is difficult to prove the validity of the proposed SPD by means of a single analysis: A single analysis can only capture a particular characteristic of a feature space from a particular perspective. However, when a number of analysis approaches are used simultaneously, combination of the derived different perspectives can establish a panoramic view of a feature space.

The following four different approaches will be used here for analysing a feature space from different perspectives: visual space analysis, GMM-based analysis, LDA-based analysis and Separability-based analysis. Visual space analysis shows a feature space by a small dataset (two speakers) giving a realistic example for the feature space. GMM-based analysis is used to graphically show the clustering of different classes (e.g. phonemes, speakers, gender and dialect regions). LDA-based analysis is used to observe a feature space in a linear way. Finally, separability-based analysis is a more general approach to analyse a feature space in an analytical and at times nonlinear way.

10.3.1 Visual space analysis

Visual space analysis illustrates the structure of a feature space by observing a small realistic dataset. The acoustic feature space has a high dimension, so dimension reduction to 2 or 3 dimensions must be applied first. Principal components analysis (PCA) is one of the most popular means of dimension reduction. Therefore, for visualising the actual distribution of a dataset, PCA is first applied to transform this dataset. The two most important principal components corresponding to the two largest eigenvalues are adopted as the coordinates in a 2-*d* plane. The generated 2-*d* plots can show the real structure of a feature space visually, a fact which helps to better understand the space structure and the feature distribution.

In order to conduct visual space analysis, a small dataset from only two speakers is used. A distributional pattern similar to that proposed in the SPD will be illustrated in the result section (Section 10.5.1).

10.3.2 GMM-based analysis

GMM-based analysis is an approach to show graphically the distribution of the labelled classes (phonemes, speakers, genders, dialect regions) by a global GMM classifier. In other words, a GMM classifier is first trained on the full dataset. This is then used to label the clustered data by selecting the Gaussian component which gives the maximum likelihood for the given data frame. The proportion of data from each class clustered within each Gaussian component is used as an indicator to evaluate the separability of each type of information concerned with phonemes, speakers, genders and dialect regions.

The global GMM classifier is trained by using the K-means clustering algorithm followed by expectation maximisation (EM). This procedure is implemented using the Torch machine learning API (Collobert et al. 2002) with a variance threshold factor of 0.01 and a minimum Gaussian weight of 0.05, which are optimal for the used features (cf. Section 11.4).

For the classification stage, a frame is classified by selecting the maximum likelihood conditioned on a specific Gaussian mixture for a given data frame $\mathbf{x}_t$, i.e.

$$i = \arg\max_{j} w_j \cdot P(\mathbf{x}_t \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \qquad (10.1)$$

where w_j, $\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_j$ are the weight, mean vector and covariance matrix of the j-th Gaussian of the global GMM.

10.3.3 LDA-based analysis

LDA-based analysis trains an LDA classifier on a training set and uses *c* linear discriminant functions to classify *c* classes on a test set. The derived correct recognition percentage for class division is used as an evaluator for the linear class separability.

Notation:
d: the dimensionality of a feature space
n: the number of training samples
X: an original dataset ($d \times n$)
Y: an LDA transformed dataset ($d \times n$)
W: a linear transformation ($d \times d$)

W is optimized on a training set (see Section 6.2.2), such that

$$\mathbf{Y} = \mathbf{W}\mathbf{X}. \qquad (10.2)$$

For the classification stage, *c* discriminant functions are used to classify *c* classes on a test set, i.e.

$$\mathbf{a}_i^T \mathbf{Y} = \mathbf{b}_i^T,\ 0 \le i < c. \qquad (10.3)$$

Defining

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_0^T \\ \mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_{c-1}^T \end{bmatrix} \text{ and } \mathbf{B} = \begin{bmatrix} \mathbf{b}_0^T \\ \mathbf{b}_1^T \\ \vdots \\ \mathbf{b}_{c-1}^T \end{bmatrix},$$

then we obtain

$$\mathbf{AY} = \mathbf{B} \Rightarrow \mathbf{AWX} = \mathbf{B} \Rightarrow \mathbf{AWX}(\mathbf{WX})^T = \mathbf{B}(\mathbf{WX})^T \Rightarrow \mathbf{A} = \mathbf{B}(\mathbf{WX})^T(\mathbf{WXX}^T\mathbf{W}^T)^{-1}.$$

For a given number of classes, LDA is obtained for a training dataset. This transformation is then used to predict the test data. The prediction correct percentage is used as an indicator to describe the degree of separability of the class set, which in turn indicates how difficult it is to separate these classes.

10.3.4 Separability-based analysis

Separability-based analysis is a more general approach than LDA-based analysis, as it is not necessarily linear. Furthermore, separabilitiy-based analysis can be conducted on the whole dataset without the requirement of partitioning the training and test set as in the case of LDA-based analysis. Therefore, it is more powerful than LDA-based analysis.

In this analysis, the separability for a given class division D of data set X is denoted by $Sep(D,X)$. *Sep* is an indicator of the class separability. It can be combined with the normalised class entropy $NH(D,X)$ (10.5) and the normalised mutual information $RI(D_1,D_2,X)$ to infer the structure of a feature space. The NH measure is a measure of the uncertainty for the class to which a given observation belongs. The RI (relative information) measure is a measure of the degree to which any two class divisions are related, which is independent of the type of mapping between them.

- $Sep(D,X)$, the "(class) separability index", for a given class division D of data set X , is the ratio of the sum of between-class variances to the sum of within-class variances. Sep (10.4) increases as the number of clusters for each class, the interlacing of data from different classes, and class boundary complexity decrease.

$$Sep = trace(S_b)/trace(S_w) \qquad (10.4)$$

- $NH(D,X)$, the "normalised (class) entropy", is a measure in [0,1] of the uncertainty as to which category in class division D the data in a given set X belongs. NH (10.5) gives the proportion of classification perplexity.

$$NH(D,X) = H(D,X)/\log_2\|D\| \qquad (10.5)$$

- $RI(D_1,D_2,X)$, the "normalised (mutual) information", or Relative Information (Morris, 2000), (10.6) is a value in the range of [0,1] which tells you how much, for a given data subset X , the class in D_1 is statistically dependent on the class in D_2, and vice versa. RI (10.6) is obtained by suitable normalisation of the Chi-squared statistic, *L* (see Appendix A). It makes no assumptions about the mapping between D_1 and D_2 .

$$RI(D_1,D_2,X) = L(D_1,D_2,X)/(2N\log_e(\min(\|D_1\|,\|D_2\|))) \qquad (10.6)$$

The above data set X can be a data subset which satisfies a certain selection criterion. For instance, if we are interested in the separability of data within each Gaussian cluster (a Gaussian classifier is used to partition data into each Gaussian cluster before this step), then X can represent a data subset which falls in a Gaussian cluster. It is helpful to analyse the structure of a feature space by calculating the separability in each Gaussian cluster.

10.4 Data

The TIMIT database is used for all the analyses. Since TIMIT is an excellent, phonetic-abundant database, hand-labelled in a precise manner with other speaker-related information such as speaker identity, gender and dialect region included, it is highly suitable for the present analysis.

10.4.1 Data features

All four types of analysis are carried out in the MFCC-based feature space. But the analysis results must be valid for any other acoustic feature space. To validate this point, a wavelet-based feature space is analysed by using the most powerful approach – separability-based analysis.

Both of these feature types were used in (Sarikaya et al. 1998), where it was shown that the more recently introduced wavelet based features perform marginally better than the standard MFCC features, also used in (Reynolds et al. 1995a). As in (Sarikaya et al. 1998), all of the Timit signal data was first downsampled to 8 kHz, to simulate telephone line transmission, but no further low- or high-pass filters were applied.

MFCC features used 20ms windows and 10ms shift, with a pre-emphasis factor of 0.97, a Hamming window, 20 Mel scaled feature bands, and all MFCC coefficients except c0.

Wavelet-based features were extracted in two steps, as in (Sarikaya et al. 1998). In the first step, 32nd order Daubechies orthogonal wavelet filters were used for subband decomposition of a 24ms Hamming window with a preemphasis of 0.97 and a window shift of 10ms as for the MFCCs. The subband wavelet packet tree was the same as in (Sarikaya et al. 1998) and represented a roughly Mel-scaled distribution of the subbands across frequency. The log energy in the 24 subbands was decorrelated by DCT, the same as for MFCCs, resulting in subband based cepstral parameters. We will denote these wavelet coefficients WAVC.

10.4.2 Data labelling

Timit contains 630 speakers (438 male, 192 female) from 8 dialect regions in the USA, each speaking 10 sentences. Each utterance in Timit is phonetically hand-labelled and provided with codes for speaker, gender and dialect region. The features were processed as described in Section 10.4.1. A global GMM was trained on the whole dataset as described in Section 10.3.2. GMMs with 2, 4, 8, 16 and 32 Gaussians (which we will refer to as GMM2 to GMM32, respectively) were trained separately. After feature processing and GMM training, each feature frame (x) was labelled with the index of the Gaussian (G_i) according to (10.1). Broad phone and speaker classes are also labelled because broad class detection can be relatively robust, and in speech or speaker recognition an initial broad class identification is often used to either select or condition the model used for fine class recognition. A record was therefore compiled for every feature frame, with fields as shown in the table below (Table 10-1).

Table 10-1: Class divisions with the number of categories

Partition	Num. categories
phone, P61	61
speaker, SPK	18, 630
gender, GEN	2
dialect region, DRE	8
broad phone 1, P20	20
broad phone 2, P07	7
broad phone 3, P04	4
Gaussian index, GID	2, 4, 8, 16, 32

Categories P61 (61 phonemes) to DRE (8 dialect regions) above were directly obtained from the Timit labelling. In order to visualise the speaker division, a small set of speakers (SPK18) was also selected. The full set of speakers (SPK630) contains all the 630 speakers. P61 was grouped into 3 broad class partitions, P*nn*, where *nn* is the number of categories. The categories in P04 are obstruent, sonorant consonant, vowel and silence. The categories in P07 are obstruent, sonorant consonant, front vowel, back vowel, central vowel, diphthong and silence. P20 consists of 4 stop segment categories: closure_voiced, plosive_voiced, closure_voiceless and plosive_voiceless (for closure and release portions of voiced and voiceless plosives, respectively; closure can also denote the closure phase of affricates); 2 fricative categories: fricative_voiced, fricative_voiceless (including the fricative parts of affricates); 1 nasal category; 1 category of liquids; 1 category of glides; 10 vowel categories: cfv[2], mfv, ofv, obv, di, ocv, mbv, cbv, mcv, ccv; 1 silence). Each data frame was also labelled with the index of the Gaussian which gave the highest probability density.

10.5 Results

The purpose of this section is to validate the SPD proposed in Section 10.2. The different types of analysis provide support to this SPD from different perspectives.

10.5.1 Visual space analysis

Visual space analysis is conducted on a small dataset containing data from only two speakers for simplicity, but it is also applicable to a case of multiple speakers.

[2] Pos 1: c: central, m: mid, o: open
Pos 2: f: front, c: central, b: back
Pos 3: v: vowel
di: diphthong

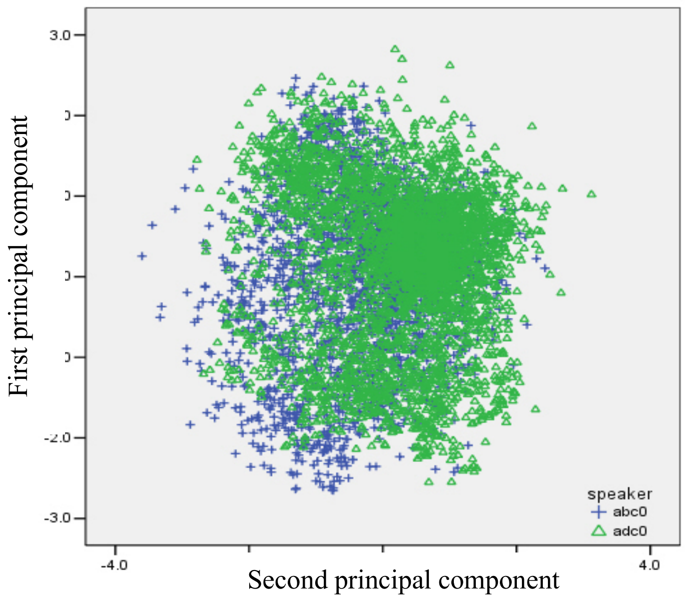
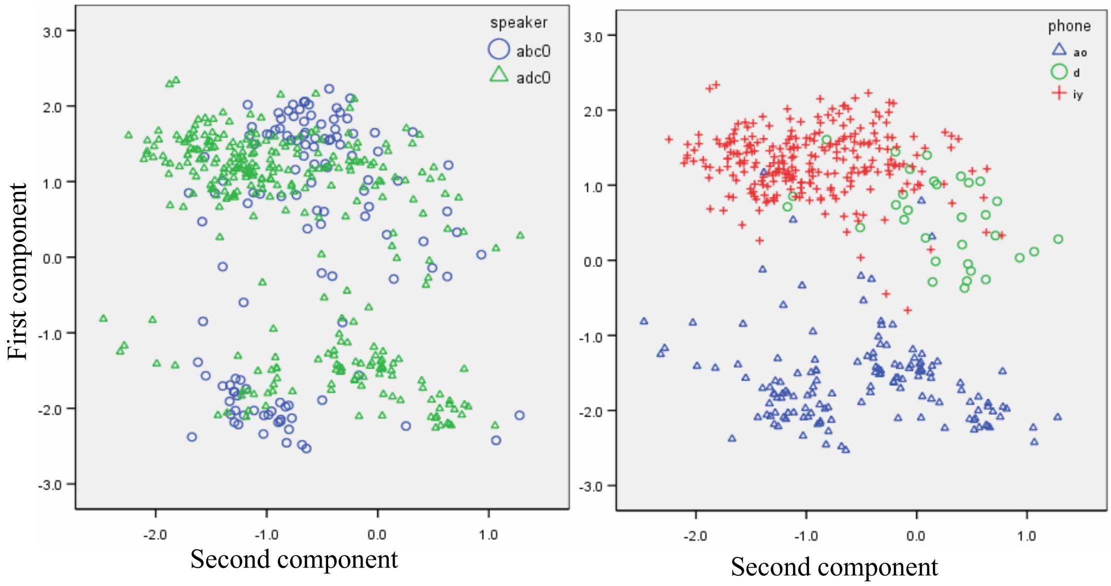


Figure 10-2: Subset (2 speakers, all phonemes) feature space represented by two principal components



(a) (b)

Figure 10-3: (a) Smaller subset feature space represented by two principal components marked by speaker. (b) Smaller subset feature space represented by two principal components marked by phoneme. (MFCCs, three phones /ao/, /iy/, /d/ from the same two speakers)

We arbitrarily chose two speakers from the TIMIT database, whom are called them speaker 1 (speaker label abc0) and speaker 2 (speaker label adc0). Then we used all the utterances (19-*d* MFCCs) from these two speakers and applied PCA transformation to them so that PCA projected them from a 19-*d* space into a 2-*d* dimensional space by only choosing the two largest principal components. In other words, the two principal components representing the two largest variance directions of these data were selected. The resulting graph is shown in Figure 10-2.

Figure 10-2 was plotted by using the frames of all the phonemes from two speakers. In this figure, the two speakers are shown overlapping, although in the left bottom corner, the data from one speaker occurs more often than that from the other.

In order to observe these data in a clearer way, we only selected three phonemes (/ao/, /iy/, /d/) for visualisation, which were arbitrarily chosen. Then, the same figure was replotted by using a smaller subset of data with only three phonemes selected from two speakers.

It is clearly shown in Figure 10-3b, the samples of the phoneme /ao/ are in the separate cluster from those of /d/ and /iy/ and vice versa. However, within each phoneme, data from both of speakers overlap (Figure 10-3a). The pattern shown in these two graphs strictly complies with the SPD.

10.5.2 GMM-based analysis

Visual space analysis was carried out on a two-speaker dataset. In order to analyse the characteristics of the overall feature space using as many data samples as possible, GMM-based analysis was conducted.

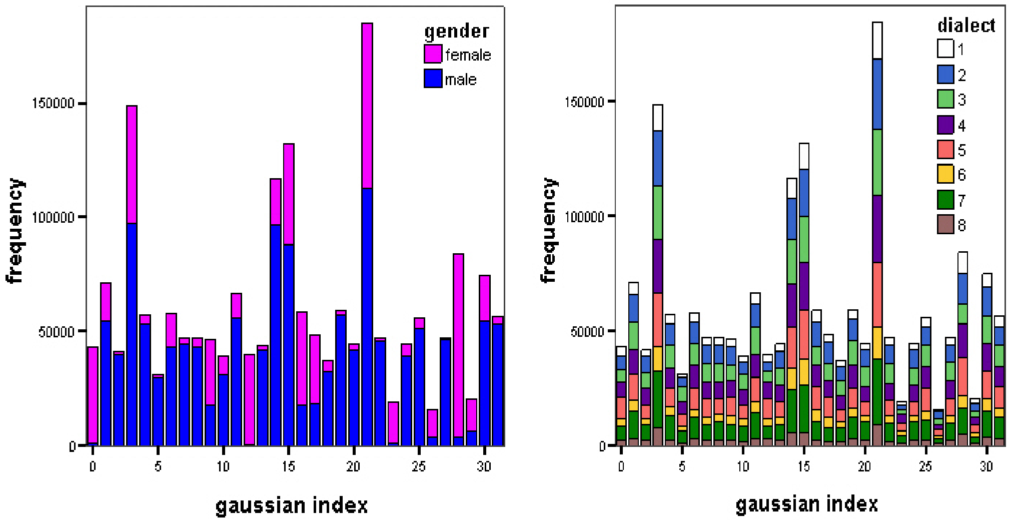


Figure 10-4 (a,b): Histograms for data frames falling into each Gaussian in GMM32 for (a) gender, (b) dialect region

The full set of data frames was used by GMM-based analysis for gender, dialect regions and phonemes. However, it could not be employed to observe the clustering of data for individual speakers (too many speakers cannot be shown in a figure). Therefore, the data selected from a subset of just 18 distinctive speakers, consisting of 3 males and 3 females from each of 3 dialect regions, with 2 sentences per speaker (the two SA sentences) were used for speaker analysis.

Figure 10-4 shows histograms for gender (a) and dialect region (b), in each Gaussian set in MFCC GMM32. Each Gaussian selects a subset of data from a different region of acoustic space, which accounts for a different proportion of data from each class division. For example, some Gaussians, like 0, 12, 23, 28 of Figure 10-4a clearly separate male/female (sonorants in Figure 10-5b), while others, like 3, 15, 21 do not (voiceless obstruents and silence). Figure 10-4b shows that each dialect region has almost an equal probability of being in each Gaussian, which reflects the results in Table 10-2.

Figure 10-5a shows that most Gaussians have data from most speakers(for a subset of 18 speakers in the Timit database), although each selects a different proportion of data from each speaker, sometimes excluding a number of speakers completely, e.g. Gaussian 19, which is dominated by a single speaker. If viewed in colour, Figure 10-5b shows that vowels and sonorant consonants (which carry more speaker distinguishing information) are clearly distinguished from obstruents and silence. Closer inspection of the data showed further systematic patterns. For instance, when obstruents and silence frames are represented by the same Gaussian, the obstruents are mainly voiceless.

From these figures, we can see phonemes are most discriminately represented by Gaussian clusters. Within each phoneme cluster, most the speakers have some data, although there is often an unequal amount of data for each speaker. This also complies with the SPD. Moreover, almost an equal amount of data from each gender and dialect region falls into each Gaussian cluster. This shows that phonemes, speakers, gender, dialect regions are separable approximately in this order. This answers the question A, i.e. to what extent is each type of information among them (linguistic, speaker identity, gender and dialectal affiliation) represented by MFCC features.

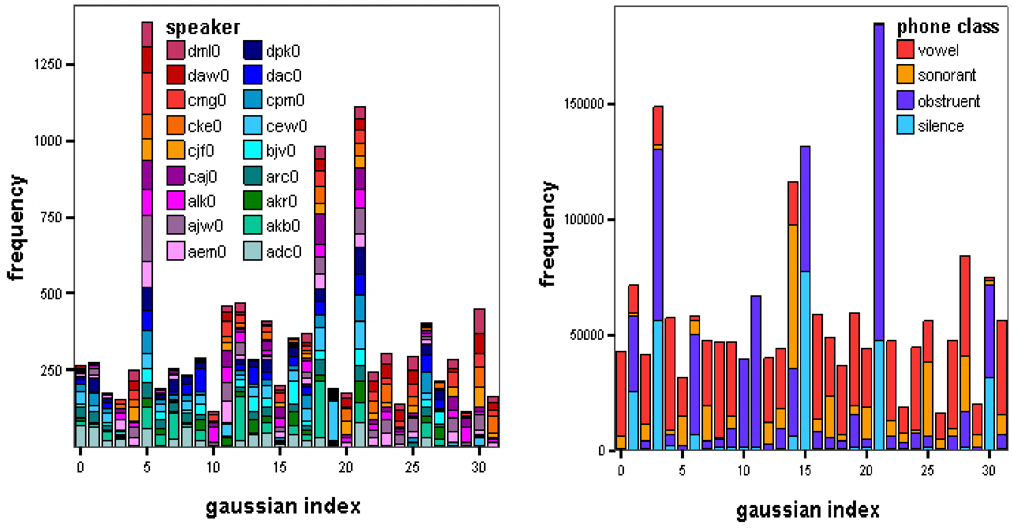


Figure 10-5 (a,b): Histograms for data frames falling into each Gaussian in GMM32 for (a) speaker, (b) phone class

10.5.3 LDA-based analysis conditioned on broad phonetic class

The LDA-based analysis conducted here allows us to inspect the separability of each class division (speaker, gender and dialect region) when conditioned on knowledge of broad phonetic class.

Table 10-2: Percent correct speaker, gender and dialect region classification by LDA on data within the set belonging to each of the 20 broad phonetic classes in set P20. (nas=nasal, opn=open, cen=central, vow=vowel, bre=breathing, vcl=voiceless, plo=plosive, rel=release, clo=closure, fri=fricative, bak=back, fro=front, beg=begin, sil=silence). Broad-class rows in each column are arranged in order of decreasing identification accuracy.

Speaker		**Gender**		**Dialect region**	
nas	27.2	opn fro vow	94.7	opn cen vow	22.9
opn cen vow	25.3	glide	94.3	pau	21.8
mid bak vow	23.1	clo bak vow	93.8	nas	21.7
clo bak vow	22.8	mid bak vow	93.7	opn fro vow	21.46
opn fro vow	20.7	clo cen vow	93.6	clo bak vow	21.6
diphthong	18.2	opn cen vow	93.0	mid bak vow	21.4
mid fro vow	15.9	clo fro vow	92.9	opn bak vow	21.4
epenthetic sil	15.5	mid bak vow	92.6	diphthong	21.4
opn bak vow	15.0	nas	92.1	mid cen vow	21.0
clo cen vow	14.0	mid cen vow	92.0	mid fro vow	20.9
glide	13.5	diphthong	92.0	glide	20.6
clo fro vow	13.1	opn bak vow	91.7	clo fro vow	20.4
pau	12.0	liquid	91.2	liquid	20.3
mid cen vow	11.0	voiced clo	82.6	epenthetic sil	20.1
Liquid	8.6	voiced fri	81.3	clo cen vow	19.8
voiced clo	6.3	voiced plo rel	80.7	voiced clo	19.5
bre	5.4	pau	75.4	voiced plo rel	19.2
voiced plo rel	4.4	vcl fri	75.0	voiced fri	18.8
voiced fri	3.6	epenthetic sil	74.7	bre	18.6
vcl fri	2.5	vcl plo rel	73.6	vcl plo rel	18.2
vcl plo rel	2.4	bre	71.2	vcl clo	18.2
vcl clo	2.2	vcl clo	70.9	vcl fri	18.0
average	10.7	*average*	80.9	*average*	18.0
random choice	0.2	*random choice*	50.0	*random choice*	12.5

When data is restricted to one phonetic class, the proportion of inter-speaker acoustic variation to other sources of variation is increased, while each speaker is still represented by approximately the same amount of data. Broad phonetic class is often more reliably estimated than fine class. In some situations, such as text prompted speaker recognition, the phoneme sequence is specified a-priori and can therefore be used to condition the speaker separation. In this section we perform a systematic analysis of the relative effect on speaker, gender and dialect region separability of conditioning on each broad phonetic class.

As mentioned in Section 10.3.3, each point was projected by LDA trained on the training set, either from MFCC features or from its associated log Gaussian probabilities, onto a transformed space and then tested on the test set. Because the classification performance (correct classification percentage) of these two projections was similar, results in Table 10-2 below are reported only for the latter. The broad phonetic classes for predicting speaker, gender and dialect region are shown in Table 10-2.

We see in Table 10-2 that dialect region was very little separated at all, even when conditioned by the phone class (P20). This suggests that, within this database at least, dialect is not well characterised by the short term MFCC features obtained from a 20ms window. Gender is better separated by sonorants (top 3 lines) since sonorants contains more information about the speaker's fundamental frequency. Speakers are best separated by nasals, which convey the characteristic shape of the nasal cavity by their timbre, as well as pitch. All classes are least separated by voiceless sounds, which carry least information about vocal tract shape and none about the characteristics of the glottal source.

LDA-based analysis shows that knowledge of broad phoneme class can have a strong effect on the separability of speaker, gender and dialect region. Phones such as nasals and vowels, which reflect characteristics of the speaker's vocal tract, carry more speaker-discriminating information than oral consonants. From this angle, it also complies with the logical corollary of the SPD, i.e. speakers can be discriminated differently within each different phone cluster.

10.5.4 Separability-based analysis conditioned on Gaussian index

As mentioned above, LDA-based analysis uses a linear classifier trained on a training set to obtain the correct classification percentage on a test set. An alternative approach is to apply the separability measure used in cluster analysis directly to the full set of data in the feature space.

However, here, instead of using phonetic information, the analysis is conditioned on the Gaussian index, which is available to a GMM classifier. As the SPD suggests, similar phonemes tend to be classed into the same Gaussian cluster (cf. Section 10.5.2). Therefore, if what the SPD suggests is true, then the separability of the speakers conditioned on Gaussian index should be increased. Furthermore, the more clusters are used, the higher the separability of the speakers. In the rest of this section, this will be shown with separability-based analysis.

As described in Sections 10.3.2 and 10.3.4, a global GMM was used for clustering data. Then Sep and NH values were obtained for every Gaussian within each of GMM2 to GMM32. We report here the average Sep and NH values over all of the Gaussians within each GMM. We also report the RI value for each GMM, because it provides a direct measure of statistical dependence. In order to test the sensitivity of the results reported to the choice of data features, all tests were repeated for WAVC as well as MFCC features.

In Figure 10-6(a,b) we see that as the number of Gaussians increases, phone class separability for MFCC data decreases by about 50% (overall), while speaker separability increases by a factor of about 3. For GMM32, speaker separability is greater than phone separability. Approximately the same is true for WAVC data, although the wavelet features provide greater speech separability for GMM1 (the raw data without GMM modelling). This suggests that within each Gaussian, it becomes much easier to separate speakers. By contrast, it is harder to distinguish phones.

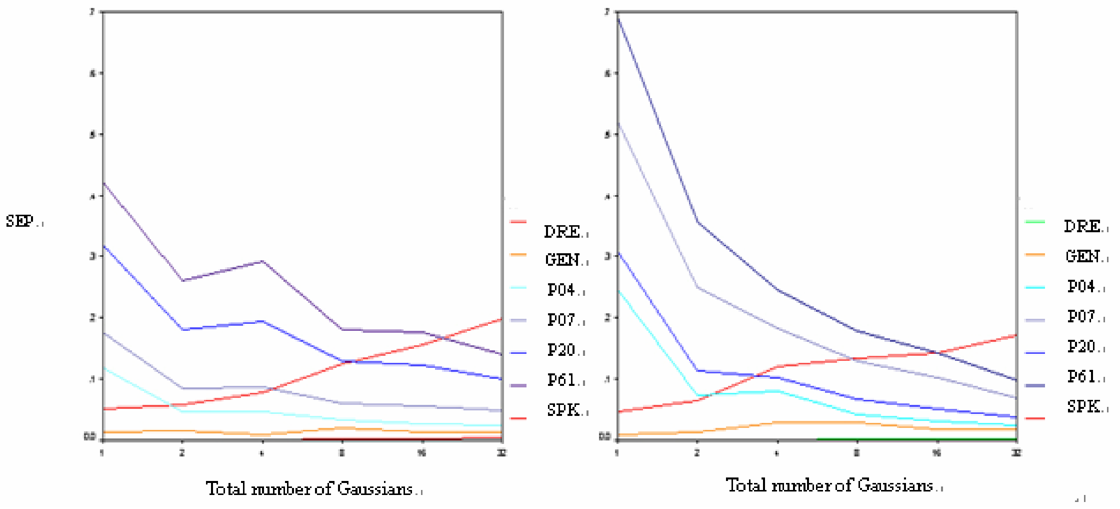


Figure 10-6: (a) (left) shows speaker separability Sep values for MFCC data against the total number of Gaussians. (b) (right) shows same for WAVC features. Classes separated are dialect region (DRE), gender (GEN), speaker (SPK), different numbers of broach phoneme classes (P04, P07, P20, P61)

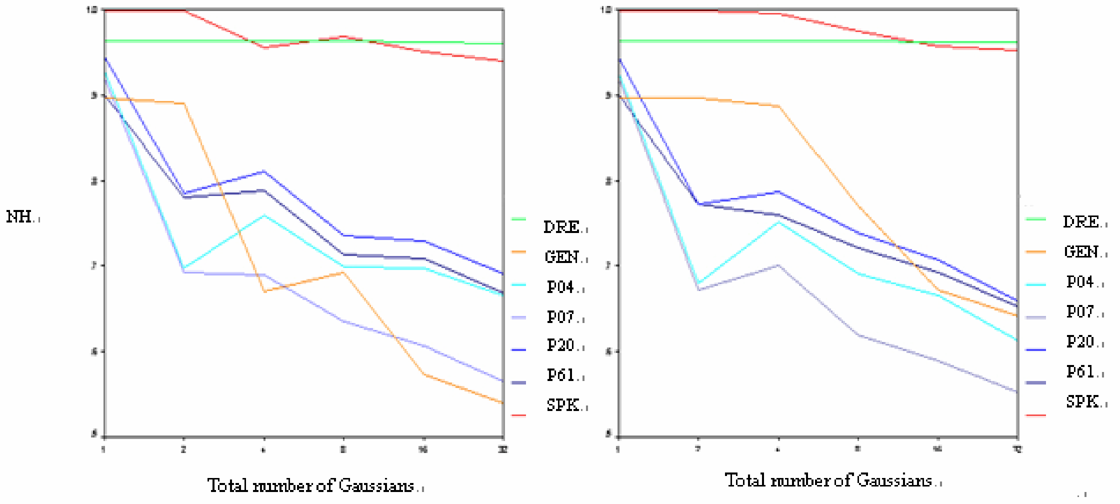


Figure 10-7: (a) (left) shows speaker normalised entropy NH values for MFCC data against the total number of Gaussians. (b) (right) shows same for WAVC features

In both cases phone entropy is consistently lower than speaker entropy, and decreases as the number of Gaussian increases. This suggests that the phonetic information is more and more certain as the Gaussian number increases. In Figure 10-7 it can be observed that phone entropy, NH, is lower than speaker entropy for all GMMs, and decreases as the number of Gaussians increases. By contrast, both speaker and dialect region entropies are close to their maximum possible value (1.0) for every GMM, with little decrease as the number of Gaussians increases. This confirms the SPD that, unlike phoneme classes, each of which is well clustered and little fragmented, the distribution of speaker data is much more 'holistic', being almost invariant with respect to the region of feature space sampled.

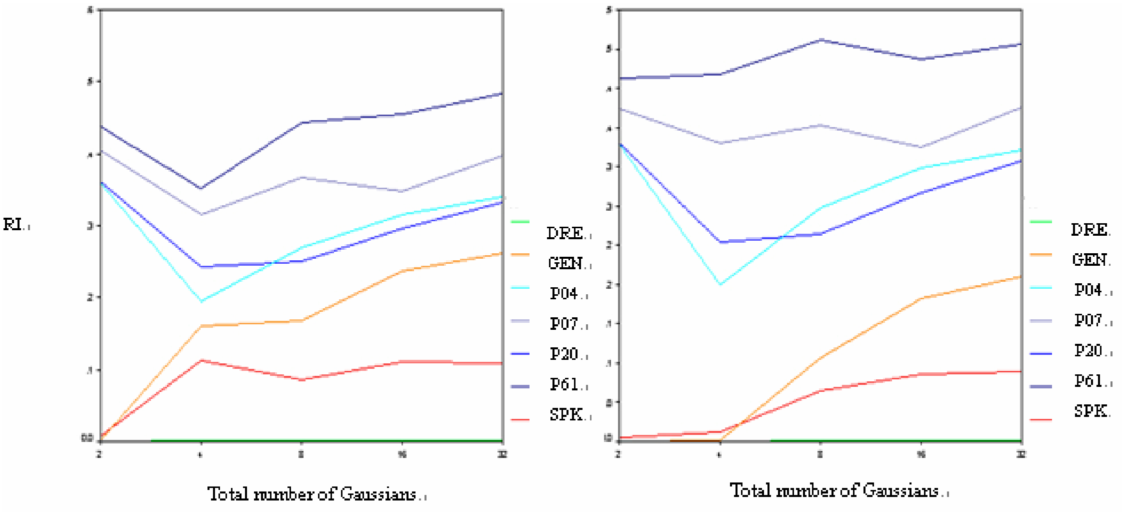


Figure 10-8: (a) (left) shows speaker RI between each of the speech and speaker partitions and Gaussian index, for MFCC data, against the total number of Gaussians. (b) (right) shows same for WAVC features

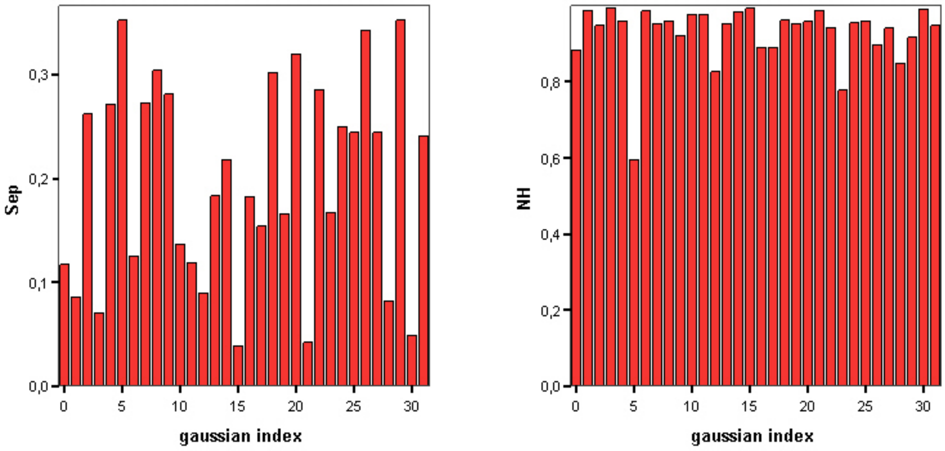


Figure 10-9: (a)(left) shows speaker separability Sep values for MFCC data in each Gaussian of GMM32, (b) (right) shows NH values for MFCC data in each Gaussian of GMM32

In Figure 10-8 we see that Gaussian index is strongly dependent on phonetic class for all GMMs, and speaker RI increases at first, but levels out at a low level (confirming the SPD). RI for dialect region is near to zero throughout, showing that neither MFCC nor WAVC features capture any information required for dialect region separation with maximum 32 Gaussians. Gender dependence keeps increasing with the number of Gaussians in the GMM. This indicates an increasing separation of phonetic clusters into separate sets for males and females.

In Figure 10-9 we see that speaker separability varies strongly between Gaussian subsets for GMM32, but Sep does not correlate with NH. This suggests that the observed differences in speaker separability within Gaussian clusters are mainly due to differences in speaker class overlap and/or fragmentation, rather than to differences in speaker distribution perplexity (number of different speakers).

10.6 Conclusions

In the above analyses from different perspectives, we identified cluster entanglement, rather than perplexity or class overlap, as the major factor limiting speaker separability in the Timit speech database. This confirms the proposed SPD.

As mentioned in Section 10.2, with the confirmation of the proposed SPD all the questions (A–D) raised in the introduction to this chapter can be answered. The most significant conclusion is that MFCC is most suitable for speech recognition, but is not an optimal feature type for speaker recognition, due to the entanglement of speaker classes within each Gaussian subset. This motivates us to propose the use of a new feature type – discriminative features for improved speaker recognition (Chapter 11).

In Section 10.5.3 we used LDA based classification to show that speaker entanglement can be reduced by conditioning on broad phonetic class. In Section 10.5.4 we used the three measures Sep, NH and RI to show that, as the number of Gaussians in the GMM increases from 2 to 32, the average speech entropy in each Gaussian decreases, while the average speaker entropy remains near constant, with the effect that the ratio of speaker to speech separability increases. This suggests that the entanglement of speaker classes within each Gaussian subset is significantly reduced, thereby increasing speaker separability.

11. NLDA-based feature enhancement by MLP

11.1 Introduction

In Chapter 10, we analysed the MFCC feature space and found that MFCCs tend to be clustered around phonemes rather than speakers. This is a potentially negative factor for speaker recognition, because a speaker recognition system would prefer the acoustic features to be clustered around speakers, as this would make it easier to discriminate them.

In spite of that, almost all the state-of-the-art speaker recognition systems (both identification and verification) use MFCC features to represent speaker-discriminating information, although these were specially designed for speech recognition systems (Reynolds et al., 1994, 1995a, 1995b). These MFCC features may not be the optimal features for speaker recognition, as the purpose of speaker recognition is different from that of speech recognition.

One possibility to improve the performance of speaker recognition systems in a simple way is to eliminate as much linguistic information in signals as possible, leaving only speaker-specific features, such as voice quality parameters characterising, for example, a speaker's nasal cavity or the fundamental frequency of the voice. This would parallel the human process by which a speaker can be recognised from his voice without the linguistic content being understandable. Linguistic characteristics such as certain idiosyncratic phrases or word selection also contain useful information to distinguish speakers, but, since they can be relatively easily learnt by impostors or changed by the social environment, they are not regarded as essential features to differentiate speakers. In speech recognition, the purpose is obviously to extract the linguistic content and nothing else. Hence, there is no absolute necessity for speaker recognition to use the same feature type as speech recognition does.

Most importantly, the performance of state-of-the-art speaker recognition systems significantly degrades in a variety of noisy conditions, although their performance is almost 100% in clean speech (Reynolds et al., 1994, 1995a, 1995b). To solve this problem, some channel compensation approaches such as cepstral mean subtraction (CMS) (Atal 1976; Furui 1981), RASTA processing (Hermansky et al. 1992) and Quadratic trend removal (Mistretta et al. 1990) have been applied to deal with noise in the signal. CMS was found more efficient than the other two approaches for the King speech database (Reynolds et al. 1994; Campbell et al. 1999). Similarly, in Wildermoth et al. (2003) it was also found that CMS only works well for the YOHO database, but not for the 16k and 8k sampled TIMIT/NTIMIT/CTIMIT (Campbell et al. 1999). Moreover, in Wildermoth et al. (2003), the time derivatives showed a negative effect for speaker recognition on TIMIT/NTIMIT/YOHO databases.

When the cepstral means are subtracted from the cepstra to remove diverse channel and background noises in the CMS, part of the speaker-specific information is also eliminated. This may serve as an explaination why CMS does not work for TIMIT testing series. Apart from these channel compensation methods, alternative approaches based on discriminative training have also been proposed to remove channel noises. For instance, linear discriminant analysis (LDA) was applied to NIST (Campbell et al. 1999) and shown to be able to improve speaker identification accuracy (Jin et al. 2000); a single 3-hidden-layer multi-layer perceptron (MLP) was also applied as a nonlinear transformation preprocessor for speaker verification on NIST97 and NIST98 databases to alleviate the effect of microphone mismatch and channel noise. A consistent improvement was found when discriminative features were linearly combined with the original mel-scaled cepstral features (Konig et al. 1998; Heck et al. 2000). In their work, no particular channel compensation approach such as CMS was applied. Therefore, the discriminative training based approach can be regarded implicitly as having the same function as CMS in dealing with channel noise.

Apart from eliminating the effect of microphone mismatch and channel noise, discriminative features also possess other advantages over the original cepstra. In fact, they are not particularly designed for handling noise, but for enhancing speaker discrimination. Noise cancellation is a by-product of the enhancing procedure.

In this chapter, we address a series of questions on how to obtain discriminative features to improve speaker recognition. A general framework for speaker discrimination enhancement is proposed, and a number of representative speakers (speaker basis), found to play a crucial role in the generation of discriminative features are presented. Furthermore, an automatic method for speaker basis selection is proposed. These approaches are tested on low-bandwidth speech (TIMIT-8k), telephone speech (NTIMIT) and low-bandwidth speech with additive noise (TIMIT-8k+Noisex). Substantial improvements are found in all these experiments.

The rest of this chapter is organised as follows: In Section 11.2, we first review previous related studies. In Section 11.3, a general framework for MLP-based feature enhancement is introduced. In Section 11.4, MLP-based feature enhancement on clean speech is presented and discussed. In Section 11.5, speaker basis selection approaches are proposed, followed by enhancement tests in a variety of noisy conditions, in Section 11.6. In Section 11.7, tests are carried out on telephone speech. Finally in Section 11.8, a summary is given.

11.2 Related research

A number of transformations such as principal components analysis (PCA), independent component analysis (ICA) (Potamitis et al. 2000; Kwon et al. 2004), LDA and MLP were first applied to phoneme recognition (Hermansky et al. 2000; Kajarekar et al. 2001; Somervuo et al. 2003; Shire et al. 2000; Fontaine et al. 1997). These transformations were compared on TIMIT database in Somervuo et al. (2003) and shown to outperform the baseline system, which consisted of the standard feature representation based on MFCC with the first-order deltas, using a mixture-of-Gaussians HMM recogniser. Moreover, it was also found that nonlinear transformation MLP generally achieved better performance than linear transformations, e.g. PCA, ICA and LDA. PCA and ICA had almost the comparable phone errors, but both are superior to LDA. In addition, further improvement was gained by forming the feature vector as a concatenation of the outputs of all four feature transformations (Somervuo et al. 2003).

The transformation approaches can be applied to speaker recognition as well. In Jin et al. (2000), LDA learned on 230 speakers was used to reduce the dimensionality of MFCC features on NIST 1999 and a positive improvement over the original MFCCs was found. PCA and ICA were applied to the normalised audio spectrum envelope features instead of MFCCs achieving the improved results. However, these features were mainly based on MPEG-7 descriptors and the system using them didn't outperform the baseline system using MFCC features (Kim et al. 2003; 2004a; 2004b). In Jang et al. (2001), ICA was applied to speaker recognition on TIMIT database by substituting the ICA transformation for the conventional short-time Fourier transformation and a generalised mixture model for GMM. Although the positive results were claimed, the correct percent was lower than that of a GMM-based system using MFCC features. A nonlinear transformation based on MLP which achieved the best performance in phoneme recognition was also tested for speaker recognition (Konig et al. 1998; Heck et al. 2000). However, it was found that system performance was only consistently improved by a linear combination of transformed features with the original mel-scaled features. This was to some extent in contrast to the finding for phone recognition that the transformed features by either of four transformations (PCA/ICA/LDA/NLDA) were able to improve performance even though they were applied alone (Somervuo et al. 2003).

The reason for the failure of NLDA to consistently improve speaker recognition is mainly that the number of classes for discriminative training in speaker recognition is much larger than that in speech recognition. For instance, there are normally around sixty phonemes (the classes used for discriminative speech training) in English, especially 61 in TIMIT database. However, in speaker recognition, the number of speakers is much larger than 61. For instance, there are 630 speakers in TIMIT, 138 speakers in YOHO and 500 speakers in NIST 1998. This large number of training classes may cause a learning classifier such as MLP not to be well trained, since each class is assigned to correspond to an output in the MLP. The bigger size of the neural net, the more difficult it is to train it well. As a result of under-training, discriminative features derived may not be useful for discrimination.

To solve this problem, Heck et al. employed a fixed number of speakers (31) with a balanced mix of carbon and electret handsets, and balanced across gender to prevent from training on too many speakers (Konig et al. 1998; Heck et al. 2000). The purpose of their work of using NLDA is to remove the effect of microphone mismatch, as mentioned before. However, in our approach, what we are focussing on is the extraction of the representative speaker-specific discriminating features, but certainly at the same time, all other characteristics that may cause non-speaker-specific variances are also removed, such as microphone mismatch, channel noise or linguistic variation.

It is worth noting the difference between our work and Heck et al.'s work. Firstly, the purposes of using MLP are different. Heck's aim is to alleviate the microphone mismatch. Our aim is to extract general speaker-specific features. Secondly, as will be described later, we found that the difficulty for an MLP to be well trained for speaker recognition could be overcome by using a subset of speakers for the MLP training. The learned MLP can be also useful for any other class separation. Thirdly, the size of the subset plays a crucial role in determining the performance of a system. Only if this size is larger than a desired number (which depends on the conditions under which the tested system is built, such as the number of enrolled speakers, the level or type of noise, etc.) will the preprocessed discriminative features improve the system performance even though they are used alone. Fourthly, besides the importance of the number of speakers used for MLP training, it is also crucial to decide which speakers are better to be selected than others. These more important speakers are referred to as "speaker basis", which acts like the basis of the speaker feature space. An automatic approach to speaker basis selection based on the average between-class variance is proposed and shown to be better than the other two approaches. Fifthly, based on these original discoveries, a general framework of feature enhancement for speaker recognition is presented. This set of approaches is expected to be relevant to other pattern recognition applications. Moreover, these approaches are tested in a variety of conditions.

11.3 A general framework for MLP-based feature enhancement

The general framework for MLP-based feature enhancement proposed is illustrated based on speaker recognition in Figure 11-1. Although the framework illustrated here is based on speaker recognition, it may be assumed to be applicable to other pattern recognition systems as well.

The framework consists of two basic steps. The first step is speaker (or class) selection. The objective of this step is to select the most "representative" speakers (classes) in the speaker (class) space, i.e. those who can represent the discriminating characteristics of the overall speaker space. As to the question which class is most representative and should therefore be selected, see Section 11.5 and 13.2. These selected basis speakers (classes) are used for feature enhancement, which is the second basic step. In this step, an MLP-based NLDA is trained using the basis speakers (classes) and then used for feature enhancement (see Section 11.5). The enhanced features are then used for recognition. The overall effect of feature enhancement is illustrated in the left half part of Figure 11-1. With the transformation of the trained NLDA, the speaker (class) space is somehow disentangled and stretched further apart to improve the speaker (class) discrimination. For more details on this discussion, see Section 13.2.

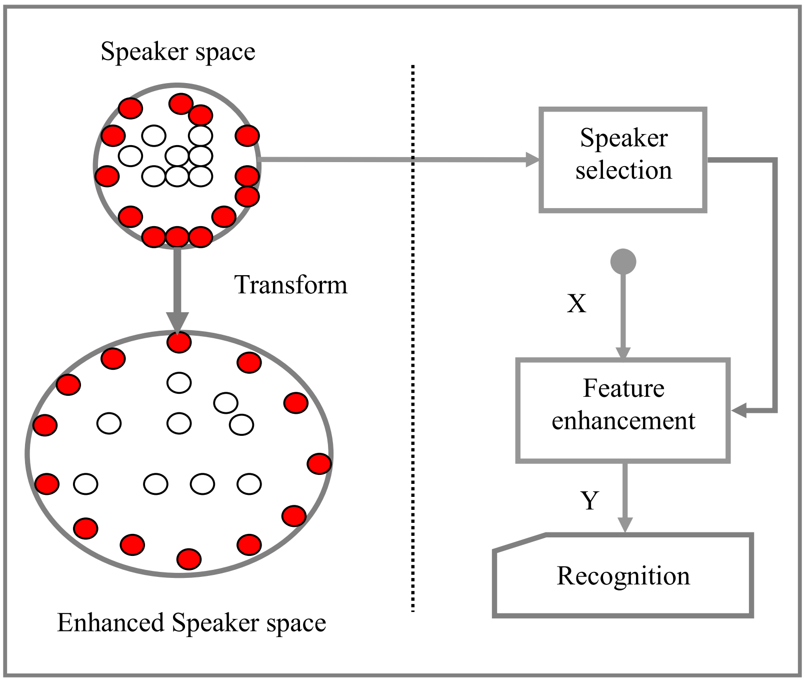


Figure 11-1: General framework for feature enhancement, illustrated based on speaker recognition

It is not straightforward to realise the fact that speaker selection is crucial for the effectiveness of feature enhancement when the enhancement approach is applied to speaker recognition. The initial idea was to use only a small set of speakers for MLP training. As more experiments were run, our understanding of the essence of this problem was deepened. As a result, we realised the importance of the selection of speaker basis.

Therefore, in the following sections we will trace the course of the investigation initially followed in this study. In Section 11.4, a small set of speakers is first randomly used to train an NLDA transformation to enhance speaker discrimination. The enhanced features are then tested on low-bandwidth clean speech. After this, the question as to whether random speaker selection was optimal is raised. In Section 11.5, three approaches to speaker basis selection are presented and compared. An automatic data-driven method is found to have a consistently better performance. Finally, in Section 11.6 and 13.7 this set of approaches is tested under different conditions with a variety of additive and channel noise.

11.4 MLP-based feature enhancement on clean speech

Our initial idea of using NLDA for speaker recognition was inspired by the success of NLDA being used for speech recognition (see Section 11.2). This was achieved by training an MLP with one output per phoneme to estimate phoneme posterior probabilities, and then using this MLP to project each data frame onto an internal representation of the data which the MLP had learnt (see Figure 11-2). This representation may be the net-input values to, or output values from, one of its hidden layers or the input to its output layer, i.e. the "pre-squashed MLP outputs" (see Figure 11-3).

As mentioned in Section 11.2, previous attempts had only limited success, except when used in combination with other techniques (Heck et al. 2000; Konig et al. 1998).

There are both practical and theoretical reasons for the lack of success of NLDA based enhancement for speaker recognition. From a practical point of view, if the MLP has one output for each speaker in the closed speaker set then it requires retraining every time a new speaker is added, while from the theoretical point of view, when the number of speakers is large, the number of free parameters in the MLP becomes so great that it cannot learn to generalise well from the limited training data available. Furthermore, while phoneme data is well clustered and relatively easy to classify (Chapter 1), data for each speaker is clustered around every phoneme centre and is therefore harder to separate. Thus, the MLP classification error remains high, in which case the features it generates may reduce, rather than enhance, speaker recognition performance (Konig et al. 1998). Reasoning that

> *the internal representation which the MLP learns to enhance separation between a small number of speakers (covering the required range of speaker types) should also be of some use in separating other speakers*,

in this section we train an MLP to recognise (i.e. estimate posterior probabilities for) a limited number of speakers selected at random from the population. By limiting the number of speakers on which the MLP is trained, both the practical and theoretical problems mentioned above are avoided.

Before training the speaker model for each new speaker to be enrolled into the GMM or HMM based speaker recognition system, and also before processing the data for a speaker to be recognised, each frame of speech data is now projected through the first few layers of this MLP onto its discriminative internal representation (see Figure 11-2).

In the present experiment, the speakers with which the MLP is trained (which we shall refer to as the *speaker basis set*) are selected from the population by balancing their dialect region, since this information can often be easily obtained in a real system. The size of the speaker basis set is varied. Within each dialect region, selection of the speakers to train the MLP is random. Results for several such random, non-overlapping selections are presented. In (Morris et al. 2005) it is shown that an automatic selection of the speakers can further enhance speaker identification. It will be further addressed in section 11.5.

Further, this section compares the MLP which was successfully applied in (Heck et al. 2000; Konig et al. 1998) with several other, simpler architectures, to evaluate the gain in speaker identification accuracy obtained by adding extra layers. A linear MLP which is theoretically equivalent to LDA is compared with LDA.

Before training the speaker model for each new speaker to be enrolled into the GMM based speaker recognition system, and also before processing the data for a speaker to be recognised, each frame of speech data is now projected by the MLP onto its discriminative internal representation (see Figure 11-2).

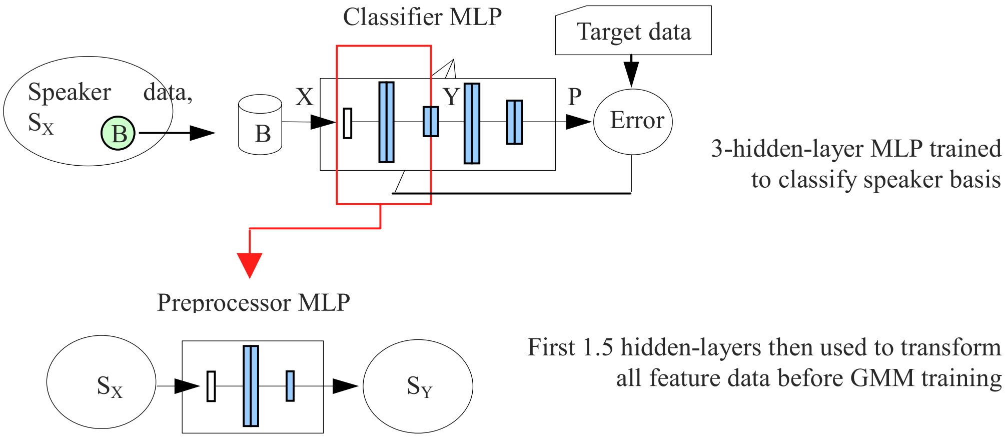

Classifier MLP
Target data
Speaker data, S_X
B
B
X
Y
P
Error
3-hidden-layer MLP trained to classify speaker basis
Preprocessor MLP
S_X
S_Y
First 1.5 hidden-layers then used to transform all feature data before GMM training

Figure 11-2: Data enhancement procedure. A small random set of basis speakers, B, is selected. This is used to train an MLP with several hidden layers to estimate a-posteriori probabilities (P) only for speakers in B. All data S_X from speakers in the full closed set of speakers to be recognised is then passed through the first 2 layers of the trained MLP to produce new data features S_Y, with enhanced speaker discrimination

The proposed approach to harness the discriminative power of MLPs for speaker recognition is a conceptually simpler and more direct application of MLPs for data enhancement than in the application of an MLP in speech recognition (Genoud et al. 1999).

In Section 11.4.1 we present the baseline GMM based speaker identification model whose performance we are aiming to improve (Reynolds et al. 1995a). In Section 11.4.2 we give the procedure used for the design and training of the MLP which we use for data enhancement. Section 11.4.3 describes the data features and procedures used for system testing, and in Section 11.4.4 we present experimental results. These results show that the data enhancement procedure described can give significantly improved speaker recognition performance. This is followed by a discussion and conclusion.

11.4.1 Speaker identification baseline

A GMM is used to model the characteristics of each speaker (see Section 7.3.1 & 8.2). The GMM design, feature data and database used here (32 Gaussians, MFCC features, Timit) are taken from (Reynolds et al. 1995a). This simple model gives state-of-the-art speaker recognition performance. With Timit (though not with other databases, such as the CSLU speaker recognition database) no gain is found in training speaker models by adaptation from a global model.

As in (Reynolds et al. 1995a), GMMs were trained by k-means clustering, followed by EM iteration. This was performed by the Torch machine learning API (Collobert et al. 2002). We used a variance threshold factor of 0.01 and minimum Gaussian weight of 0.05 (performance falling sharply if either was halved or doubled).

11.4.2 MLP design and training

The four MLP types tested are shown in Figure 11-3. Types *a,b,c* have previously been used successfully for data enhancement in ASR (Fontaine et al. 1997; Sharma et al. 2000). These are all feedforward MLPs in which each layer is fully connected to the next. The “neurons” in each layer comprise the usual linear net-input function followed by a non-linear squashing function, which is the sigmoid function for all layers except the output layer, which uses the softmax function to ensure that all outputs are positive and sum to 1 (Bishop 1995).

Also using Torch (Collobert et al. 2002), each MLP is trained, by gradient descent, to maximise the cross entropy objective (i.e. the mutual information between the actual and target outputs). We trained in batch mode, with a fixed learning rate of 0.01. The data in each utterance was first normalised to have zero mean and unit variance. The estimated probabilities are often close to 0 or 1 and data with such a peaked distribution is not well suited as feature data. The enhanced features taken from the trained MLP of types *a* and *b* are therefore usually taken as the net input values in the output layer, prior to squashing. For type *c* they are normally taken as the squashed output from the last hidden layer (these values having less peaked distributions than the outputs from the output layer), but here we have taken the enhanced features from MLPs *c* and *d* both as the net input to the second hidden layer.

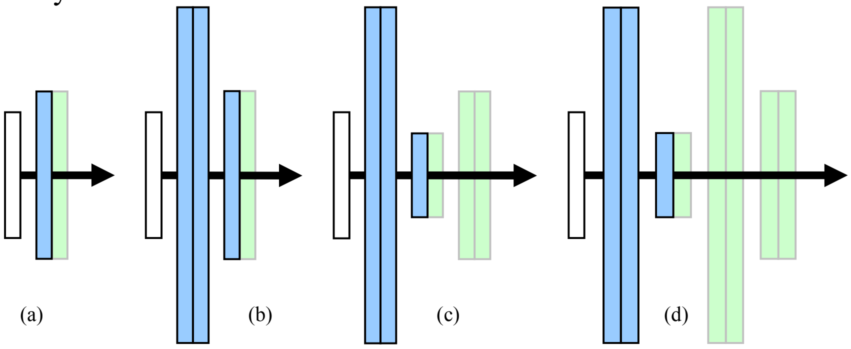


Figure 11-3: Four MLP types *(a-d)* tested for data enhancement. Each active layer is shown as a (net-input function / non-linear activation function) sandwich. Only the dark sections of each MLP were used in data projection. The light parts were used only in training

In ASR the MLP is trained to output a probability for each phoneme. In the model used here we select a random subset of the Timit speakers available for training (the *speaker basis set*) and train the MLP to output a probability for each of these speakers.

Although none of the MLPs *a-d* gave a high basis speaker classification score, the test results in Section 11.4.4 show that the speaker discriminative internal data representation which some of them learn can be very beneficial for GMM based speaker modelling.

11.4.3 Test procedure

Our baseline system is taken from the state of the art GMM based speaker identification system in (Reynolds et al. 1995b), using the Timit speech database (Fisher et al. 1986), GMMs with just 32 Gaussians, and 19 MFCC features.

1 Baseline feature processing

As in (Reynolds et al. 1995b), all of the Timit signal data was first downsampled to 8 kHz, to simulate telephone line transmission (without down-sampling, GMMs already achieve above 99.7% correct speaker identification). No further low- or high-pass filters were applied. Also as in (Reynolds et al. 1995b), MFCC features, obtained using HTK (Young et al. 2002), were used, with 20ms windows and 10ms shift, a pre-emphasis factor of 0.97, a Hamming window and 20 Mel scaled feature bands. All 20 MFCC coefficients were used except c0. On this database neither silence removal, cepstral mean subtraction, nor time difference features increased performance, so these were not used.

2 Test protocol

Timit does not have a standard division into training, development and test sets which is suitable for work on speaker recognition. For this we first divided the 630 speakers in Timit into disjoint training, development and test speaker sets of 300, 162 and 168 speakers respectively. The speaker sets are all proportionally balanced for dialect region.

Data enhancement MLPs a-d (Figure 11-3) were trained using a speaker basis set of between 30 and 100 speakers, again proportionally balanced for dialect region. Within dialect region, the speakers are selected at random from the training set. Only one frame consisting of 19 MFCC features was used as input, in parallel to the GMM baseline system which also used no information of variation of the features over time. In each case the number of units in hidden layer 1, and also in hidden layer 3 in MLP *d*, was fixed at 100. The number of units in hidden layer 2 in MLPs *c* and *d* was fixed at 19 (the same as the number of MFCC features in the baseline system). Performance could have been improved by stopping MLP training when identification error on the development test set (using GMMs trained on data preprocessed by the MLP in its current state) stopped increasing. However, in the tests reported here, each MLP was simply trained for a fixed number (35) of batch iterations, after which mean squared error on the training basis stopped significantly decreasing.

Each MLP type was tested just once with each number of basis speakers. For the best performing MLP (MLP *d*), test-set tests were made with multiple different speaker basis subsets obtained by dividing the training data into as many equal parts as each speaker basis size would permit.

Timit data is divided into 3 sentence types, $SX_{1\text{-}5}$, $SI_{1\text{-}3}$ and $SA_{1\text{-}2}$. The text independent GMM for each speaker to be tested was trained on MLP projected sentences of type ($SX_{1\text{-}2}$, $SA_{1\text{-}2}$, $SI_{1\text{-}2}$) and tested on MLP projected sentences of type (SX_4, SX_5). Baseline GMMs were trained on MFCC features. The speaker identification procedure was as described in Section 11.4.1. Both training and testing used Torch (Collobert et al. 2002).

11.4.4 Results

Test set speaker identification scores, for MLP type *a-d* against speaker basis size, are shown in Table 11-1 and Figure 11-4. The baseline test set identification error was 3.87%.

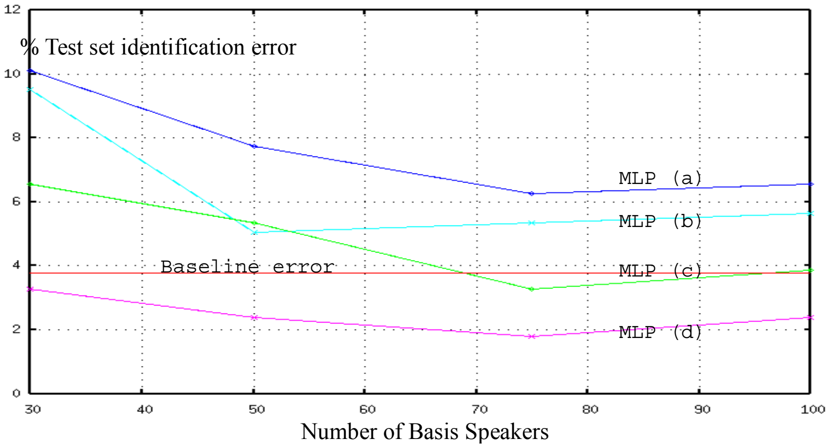


Figure 11-4: Speaker identification error rate for the 168 speakers in the test set, for data enhancement using MLPs *a,b,c,d*, with varying numbers of basis speakers

Table 11-1: Test set speaker identification error for MLPs *a-d* in Figure 11-3 against speaker basis size

Speaker basis size	30	50	75	100	best % rel. error reduction
MLP *a*	10.10	7.74	6.25	6.55	-61.5
MLP *b*	9.52	5.06	5.36	5.65	-30.7
MLP *c*	6.55	5.36	3.27	3.87	15.5
MLP *d*	3.27	2.38	1.79	2.38	53.8

The best scoring MLP (MLP *d*) was then tested many times, for each number of basis speakers, also on the test set (Table 11-1). While results for different repetitions for each speaker basis size varied considerably, in 28 out of 30 tests the speaker identification error was lower than the baseline error. The optimal size of the speaker basis set used for training was 100, giving a relative error reduction of up to 77.0 %.

Table 11-2: MLP *d* speaker identification test-set % error against speaker basis size. For each number of basis speakers, test-set tests were repeated, using disjoint speaker basis sets, as many times as were permitted by the number of available speakers (Baseline error 3.87%)

Repetition \ Basis size	30	50	60	75	100	150
1	3.57	2.68	2.68	2.37	1.49	2.08
2	2.98	2.68	1.79	2.08	0.89	1.79
3	3.87	2.08	2.68	3.57	1.49	
4	2.08	2.08	1.79	2.08		
5	3.27	1.79	1.79			
6	4.76	1.49				
7	2.68					
8	3.27					
9	1.49					
10	3.57					
Mean % error	3.15	2.13	2.15	2.53	1.29	1.93
Max % rel. err. reduction	61.5	61.5	53.7	46.3	77.0	53.7

11.4.5 Discussion

Results reported show up to 2.98% absolute (77.0% relative) performance improvement over the state of the art baseline on the Timit database. This was achieved with minimal fine-tuning and confirms our working hypothesis that the transformation learnt by the MLP to separate a random subset of speakers also substantially enhances separability between any speakers from the same population. An increase in identification accuracy has been found before with LDA when one output was trained for each speaker to be recognised (Jin et al. 2000). By contrast, our MLP (a), which performs a linear separation equivalent to LDA (Duda et al. 2001), performs on average very badly. However, this could be because in our case none of the test speakers are used in training, so that the MLP is required to generalise to new speakers.

It appears that the ability of the features provided by the MLP to enhance speaker discrimination increases with the number of hidden layers. However, from the application viewpoint it would be advantageous to keep the MLP size and data transformation complexity to a minimum. It would be interesting to know whether the quality of data enhancement can be increased by dividing a given number of neurons into a greater number of layers, allowing for a more highly non-linear transformation.

Because of the large search space of possible MLP configurations, our search is still far from being optimised. Our decision to alternate large with small hidden layers is based on the intuition that the benefits of non-linear vector space expansion and data compression should possibly be balanced. Our choice of MLP types *a-c* for testing was also guided by what has been used successfully before in ASR (Fontaine et al. 1997;

Sharma et al. 2000), while MLP *d* was used in (Heck et al. 2000; Konig et al. 1998) for speaker recognition feature enhancement. The features it produced did not, however, consistently improve speaker verification for shorter test utterances, even though the quantity of training data used was at least 4 times larger than in our experiments. In future we could try varying layer sizes, and also test the discriminatory power of features from every compressive hidden layer, not just the second. So far we have seen performance always increasing with the number of hidden layers used in MLP training (while always using just three layers for data enhancement). We have yet to find the point where this benefit stops increasing.

To reduce the amount of experimentation required, the number of MLP batch training iterations was fixed at 35, although it is well known that MLPs tend to overfit to training data after the learning curve begins to flatten out. In future we should use cross validation testing to permit training to stop when MLP preprocessing maximises speaker identification performance on the development set.

Results are only reported here for multiple *random* but balanced selections of each given number of basis speakers. While the number of speakers selected was always large enough to guarantee a fairly representative sample from the full speaker population, the somewhat erratic variation in identification performance resulting from different random speaker bases of the same size suggests that it would be instructive to see whether more principled methods could be used for basis speaker set selection. First results in this direction are reported in (Morris et al. 2005).

Although the improvement in this section was reported based on identification experiments, the similar approach can also be applied to any verification task due to the enhanced discriminating property of the transformed features. Some further possible improving scheme will be discussed in Chapter 13.

11.4.6 Comparison between a linear MLP and LDA

Theoretically, a linear MLP is equivalent to LDA except that a linear MLP is learned by gradient descent algorithms instead of matrix calculus. These different learning methods result in that fact that LDA has a unique solution whereas a linear MLP often has a solution region (Bishop 1995). Moreover, due to the existence of many local extrema, a linear MLP may stop its training at any of these points, in which case the performance of a linear MLP may be not equal to or even inferior than that of LDA. This point was shown on the downsampled TIMIT (TIMIT-8k) database by comparing with the performance of transformed systems using a linear MLP and LDA. In addition, in order to show the different efficiency of using linear transformations and nonlinear transformations (NLDA), the 3-layer MLP described in Section 11.4.2 (100-19-100) was compared with a linear MLP and LDA for data transformation.

A series of different numbers of speaker basis (selected by an approach described in Section 11.5) were used for training a linear MLP, LDA and a 3-layer MLP. The baseline system was described in Section 11.4.1.

It can be seen in Figure 11-5 that the LDA-based system was marginally better than the linear MLP-based one, whereas both of them were worse than the baseline system. The 3-layer MLP-based system achieved the highest accuracy with approximate 45.6% relative error reduction compared to the baseline system.

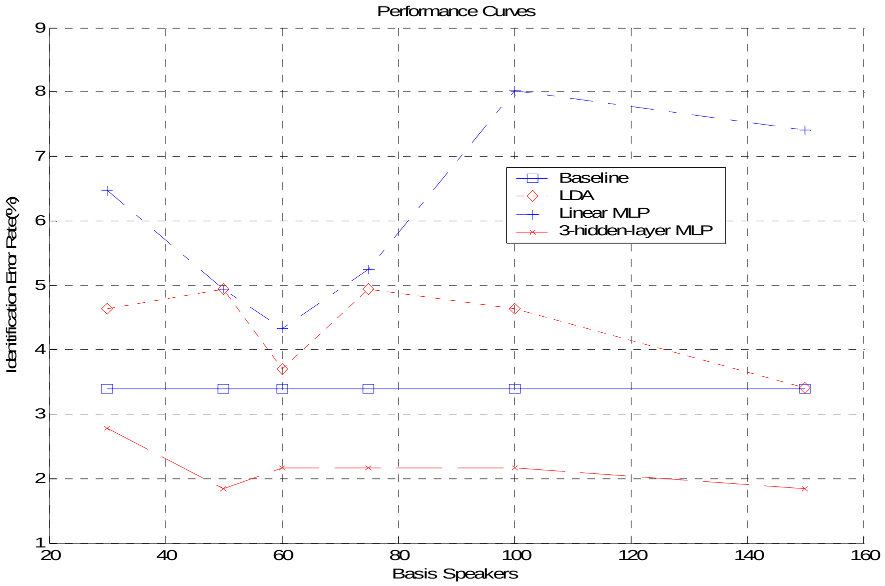


Figure 11-5: Performance Comparison between a GMM+MFCC speaker identification system (the baseline system) and discriminative feature systems transformed (by LDA, a linear MLP and a 3-layer MLP)

From the three comparative results shown in Figure 11-5 it may be inferred that a linear MLP is not always equivalent to LDA, although theoretically it should be. As mentioned earlier, the reason which causes the different performance between the linear MLP and LDA-based systems is the existence of many local optima, at any of which points the linear MLP tended to stop its training. It may be seen, for instance, that with 30, 100 and 150 basis speakers, the linear MLP has clearly stopped the training at local extrema, since the performance of the linear MLP transformed system is lower than that of the LDA transformed system. However, when using 50 speakers, the performance of the linear MLP and LDA-based systems is equal. Unfortunately, in this figure we cannot see that the linear MLP works more efficiently than LDA, but nevertheless, this may occur, depending on the actual shape of the error surface of the linear MLP training (Duda et al. 2001). Finally, it was also shown that when used for feature enhancement the non-linear MLP was by far superior to both LDA and the linear MLP, and that the NLDA could substantially improve recognition performance.

11.4.7 Conclusions

The test results reported here show that the use of MLP based data enhancement for speaker identification using different handsets (Heck et al. 2000; Konig et al. 1998) is also useful for speaker identification using very limited clean speech data. The number of target speakers which the MLP is trained to recognise must be small enough to avoid the classification problem becoming too difficult to train, but large enough to provide a feature basis sufficient to separate all speakers within a large population. The internal representation learnt by this MLP in separating the small set of basis speakers provides an enhanced feature vector which can improve GMM based speaker recognition performance. This form of data enhancement can be applied to speaker verification, as in (Heck et al. 2000; Konig et al. 1998), as well as to speaker identification. It can also be used with growing speaker sets, of unlimited size, with no need for further training as new speakers are added.

11.5 Speaker basis selection

In Section 11.4, we investigated the effect of training the MLP to classify different speaker basis sizes (i.e. on different numbers of speakers). If too many speakers were used the MLP could not learn to separate them. However, we found that if an MLP was trained on a moderately sized and suitably selected subset of speakers, a significant improvement could be achieved in speaker identification using the MLP-enhanced features on their own as input to a standard GMM based speaker recognition system (Wu et al. 2005a; Reynolds et al. 1995a). We initially tried random selection. In this case speaker identification results on the development test set varied greatly with each different random selection. Furthermore, the development test performance was not a useful predictor of evaluation test performance. To solve this problem we developed several speaker basis selection methods, which we describe in this section. It is shown experimentally that one of these approaches achieves consistently better results than the others.

In Section 11.5.1 the MLP-GMM system is briefly reviewed. In Section 11.5.2.1 and 11.5.2.2 we look at random and knowledge-based speaker basis selection. Section 11.5.2.3 then presents a number of deterministic basis selection methods and describes tests and results. This is followed by conclusions.

11.5.1 MLP-GMM speaker identification

Speaker identification experiments were carried out on the TIMIT speech database (Garofolo et al. 1993). We created our own division into speaker-disjoint training, development and evaluation data, with 630, 168 and 162 speakers, respectively. To make the speaker identification system text-independent, we used all sentences of type SA_{1-2}, SI_{1-2} and SX_{1-2} for training, sentences of type SX_3 and SI_3 for development and sentences of type SX_4 and SX_5 for evaluation.

Figure 11-6 shows the architecture of our MLP-GMM system (Wu et al. 2005a). As in ASR, the speaker identification system consists of two stages which work in tandem. The input features are first preprocessed by an MLP which has been pretrained to classify a given set of basis speakers. As the MLP is discriminatively trained, the transformation provided by the MLP gives features which better discriminate between the speakers than the original conventional features such as MFCCs. The MLP has three hidden layers. Layers 2 and 4 have of 100 nodes, while layer 3 is a compression layer with just 19 nodes.

It is the net-input values to each node in this compression layer which comprises the enhanced data features. The size of the compression layer was chosen to match the number of input features so that the input vectors to the GMM have the same size for MFCC and MLP coefficients. This internal representation of the MLP is assumed to capture the main signal characteristics which discriminate between speakers (Konig et al. 1998; Wu et al. 2005a).

The MLP is trained with Torch (Collobert et al. 2002) using a speaker basis set of between 30 and 100 speakers selected from the training set as described in Section11.4.2. To prevent overfitting, MLP training was stopped when the value of the error-objective for the development set started to increase. This was the case after 35 iterations (i.e. 35 full training epochs, using on-line rather than batch training).

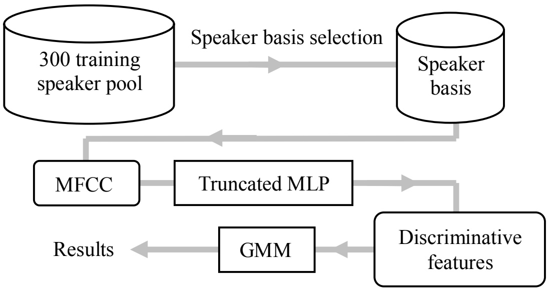


Figure 11-6: The fundamental MLP-GMM architecture

As in (Wu et al. 2005a; Reynolds et al. 1995a) the GMMs, using 32 Gaussians, were trained by k-means clustering, followed by EM iteration. This was performed by the Torch machine learning API (Collobert et al. 2002). We used a variance threshold factor of 0.01, a minimum Gaussian weight of 0.05 and the maximum number of k-means and EM iterations set to 100. With Timit (though not with other databases, such as the CSLU speaker recognition database) no gain was found in training speaker models by adaptation from a world model.

This simple baseline GMM model, without MLP feature enhancement, gives state-of-the-art speaker recognition performance with 19 coefficient MFCCs (Reynolds et al. 1995a).

11.5.2 Speaker basis selection

The assumption behind the idea that the preprocessing MLP can be effectively trained, for the purpose of open set feature enhancement, by training it to classify a subset of speakers is that we can capture the characteristics of the whole speaker space using only a small but representative set of speakers. The use of only a subset of speakers for MLP training has several advantages. Firstly, by limiting the number of target classes we also limit the amount of data which is required in order for the MLP to learn to generalise correctly. The training is fast and it can converge to a useful solution even when (as is often the case in practice) the amount of training data per speaker is limited. Also, as we are training for open set data enhancement, the MLP need not be retrained when new speakers enroll in the speaker recognition system. But an important requirement is that the subset of speakers used to train the MLP represent *all* enrolled speakers. In this section, we shall present several speaker basis selection methods, going from random to knowledge constrained random and deterministic basis selection.

11.5.2.1 Random speaker basis selection

We shall first present results for GMM speaker identification experiments which show that performance can vary strongly for different random speaker basis selections. We trained the MLP on different random selections of different numbers of speakers, using the same test protocol and configuration as tested in (Wu et al. 2005a), except that completely random selections of speaker bases are used. The random selection of speaker bases are obtained for basis sizes of 30, 50, 60, 71, 100 and 150 speakers. Each speaker basis set of a given size is randomly selected from the same group of 300 training speakers with replacement, i.e. every time a speaker basis is extracted from the training set, they are put back for the second independent random selection. The results are shown in Table 11-3.

In Table 11-3, identification error rates of 3 random speaker basis selections for given number are reported. Their average and standard deviations are calculated. The variance for the different speaker bases at each given size is quite large, so that the particular random selection of the speaker basis substantially influences system performance. This indicates how important it is to find a reliable method for speaker basis selection.

Table 11-3: Speaker identification error rates for MLP-GMM for three different random speaker basis selections of different sizes

basis	30	50	60	75	100	150
1	3.40	3.09	2.47	1.85	2.78	0.93
2	2.47	3.40	4.01	1.85	2.47	1.85
3	2.78	3.09	1.85	3.09	4.01	1.54
mean	2.88	3.19	2.78	2.26	3.09	1.44
sd	0.47	0.18	1.11	0.72	0.81	0.47

11.5.2.2 Knowledge constrained random speaker basis selection

For the TIMIT database on which the experiments reported here are carried out, several speaker properties are known beforehand. Age, height, race, education level, gender and dialect region of the speakers are known. Since the latter two are recognisable from the filenames and are likely to cause a large part of the speaker variation, this prior knowledge can be exploited for speaker basis selection. The division of gender and dialect region is not entirely balanced. TIMIT contains speech from 438 male and 192 female speakers. Of the eight dialect regions, speakers from dialect regions 2 (Northern), 3 (North Midland), 4 (South Midland), 5 (Southern) and 7 (Western) are overrepresented compared to the other regions (1=New England, 6=New York City, 8=Army Brat). We therefore selected several speaker basis sets by proportionally balancing gender and dialect region. Speakers within each gender/ dialect region group were selected randomly. The aim of this method of speaker basis selection is to use prior knowledge as much as possible. Table 11-4 shows the results for different non-overlapping speaker bases. This shows that on average the constrained random selection gives very similar results to pure random selection.

For small speaker basis sets (30 speakers), the identification error rates are mostly higher than for the baseline GMM system (3.40%). When we compare this with the random selection results in Table 11-3, this is somewhat surprising, since there the error rates are lower or the same as the baseline. For larger speaker basis sets, the error rates are generally lower than the baseline.

Table 11-4: Speaker identification error rates for MLP-GMM for proportionally balanced knowledge-based speaker basis selections of different sizes

basis	30	50	60	75	100	150
1	3.09	2.78	1.85	2.47	2.16	1.23
2	3.70	2.16	2.16	4.01	2.47	2.47
3	4.32	2.47	3.09	1.23	2.47	
4	4.32	2.47	4.32	3.40		
5	3.40	3.09	4.32			
6	3.70	3.70				
7	3.40					
8	5.86					
9	4.32					
10	3.40					
mean	3.95	2.78	3.15	2.78	2.37	1.85
sd	0.80	0.55	1.16	1.21	1.18	0.88

Of course, constrained random speaker basis selection is only possible if the database is labelled with the relevant properties related to the main sources of variation in the speech signals. Despite the controlled representation of speakers in each of the speaker basis sets, there is still considerable variation in the test results. We are forced to conclude, therefore, that gender and dialect region still leave a lot of the variation in the speech signal unaccounted for. Other variables play an important role for the discrimination between speakers.

11.5.2.3 Deterministic speaker basis selection

As the results in Table 11-4 show, different speaker bases using knowledge-based speaker basis selection can still lead to quite variable speaker identification results, so that it would still be necessary to obtain results for several speaker basis selections to find the optimal speaker basis. Besides that, most databases are not labelled with variables which may be expected to explain a large part of the variation between speakers, and it would normally be impracticable to add these labels.

All the above reasons call for an automatic approach for speaker basis selection. In this section, we present several such methods. They are all based either on GMM separability or on the log likelihoods for each speaker for each test example from the baseline GMM speaker identification task on the 2 *development* sentences for the *training* speakers (300 speakers). These likelihoods indicate the confusions between speakers on the basis of the original MFCC features. Before the automatic speaker basis selection is carried out, this log likelihoods matrix is first converted to a matrix of probabilities by first converting log likelihoods to likelihoods and then dividing each row by its row sum.

1. Selection by decreasing speaker pair confusion (M1)

In the probability confusion matrix, the probability of some confusion is higher than for others. By selecting maximally confused speaker pairs for classification in the MLP, we aim to reduce confusion between these type of speakers. After sorting the speaker pairs according to their confusion probability, the confused speaker pairs with the *n* highest frequency are selected as speaker bases (*n* is the size of the speaker basis).

2. By decreasing confusability (M2)

As M1, but GMM confusability estimated by monte-carlo sampling instead of by closed form calculation.

3. By increasing a-priori separability (M3)

First generate square speaker separability matrix, using the trained GMM models for each speaker and the separability measure trace ($\mathbf{S}_w + \mathbf{S}_b$) / trace ($\mathbf{S}_w$), where $\mathbf{S}_b$ is the expected between-class covariance matrix

$$\mathbf{S}_b = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)' \qquad (11.1)$$

and $\mathbf{S}_w$ is the expected within-class covariance matrix (Duda et al. 2001, p.p. 119)

$$\mathbf{S}_w = \mathbf{S}_1 + \mathbf{S}_2 . \qquad (11.2)$$

$\mathbf{m}_i$ and $\mathbf{S}_i$ can be easily derived from the GMMs.

This method uses “expected values” calculated from the GMM parameters, so it does not make use of the actual test or development data. Pairs of speakers are then chosen in the order of increasing separability, i.e. least separable speakers first.

4. **Maximum average distance** *(M4)*

This method uses the distance between two speaker models. Given a speaker modelled by a GMM, i.e., for any test utterance *X={x_t}*, *t=1...n*, we can easily obtain its likelihood, given speaker S_j:

$$p(X \mid S_j) = \sum_{t=1}^{n} \sum_{i=1}^{M} w_i \cdot N(x_t, \mu_i, \Sigma_i^{-1}) \qquad (11.3)$$

where *M* is the number of Gaussian mixtures.

From the likelihood, we can easily derive the *posterior probability* each speaker model has by given any test utterance *X*, i.e.

$$p(S_j \mid X) = \frac{p(X \mid S_j) p(S_j)}{p(X)} = \frac{p(X \mid S_j) p(S_j)}{\sum_{k=1}^{N} p(X \mid S_k) p(S_k)} \qquad (11.4)$$

where N is the overall number of speakers.

If we assume the prior probabilities $p(S_j)$ are the same for any speaker model, then we obtain the posterior probability given any test utterance:

$$p(S_j \mid X) = \frac{p(X \mid S_j)}{\sum_{k=1}^{N} p(X \mid S_k)} \tag{11.5}$$

In fact, posterior probabilities are just the normalised likelihoods across all the speaker models; nevertheless it allows us to much more easily derive the distance between any two models.

Define Kullback-Leibler distance

$$D_{KL}(P \| Q) = \int_X p_P(X) \frac{p_P(X)}{p_Q(X)} dX . \tag{11.6}$$

Define the distance between any two speaker models S_j and S_k by symmetric Kullback-Leibler distance as

$$KL(S_j,S_k) = D_{KL}(S_j \| S_k) + D_{KL}(S_k \| S_j) = \int \big(p(X \mid S_j) - p(X \mid S_k)\big) \log \frac{p(X \mid S_j)}{p(X \mid S_k)} dX \tag{11.7}$$

This cannot be evaluated in closed form when $p(X|S_j)$ is modelled by a GMM. However, provided $P(S_j)=P(S_k)$,

$$\begin{aligned} KL(S_j,S_k) &\propto \int p(X)\big(p(S_j \mid X) - p(S_k \mid X)\big) \log \frac{p(S_j \mid X)}{p(S_k \mid X)} dX \\ &= \int p(X) K(S_j,S_k,X) dX = E\big[K(S_j,S_k,X)\big] \end{aligned}$$

$$KL(S_j,S_k) = \int_X \big(p(S_j \mid X) - p(S_k \mid X)\big) \times \ln\left(\frac{p(S_j \mid X)}{p(S_k \mid X)}\right)$$

KL(S_j, S_k) can therefore be estimated by averaging K(S_j, S_k, X) over the development test data, with $p(S_j \mid X)$ evaluating using (11.4). Then we have the distance between two speaker distributions as

$$KL(S_j,S_k) \cong \sum_{t=1}^{S} K(S_j,S_k,X_t) \tag{11.8}$$

where S is the overall number of utterances in the development test data and X_t is a utterance in the development test set. The distance matrix KL is a symmetric matrix and $KL(S_j,S_j)=0$.

So far, we have obtained the distances $KL(S_j,S_k)$ between any two speaker models given a development test set. We can now define the sum distance from one speaker to all other speakers as:

$$SK(S_j) = \sum_{k=1}^{N} KL(S_j,S_k) \tag{11.9}$$

where N is the number of all the speakers (N includes the speaker model of itself, since $KL(S_j,S_j)=0$).

We now select the top-n boundary speakers with the largest average distances to all other speakers:

$$\{S_1\cdots S_n\}=top\_n\_\max\left(SK(S_j)\right)\cdot \tag{11.10}$$

We refer to this as the maximum average distance (MaxAD) method for speaker basis selection.

Table 11-5: Identification % Error rates of MLP-GMM by automatic speaker basis selections (baseline 3.4%)

Basis selection methods	30	50	60	75	100	150
1	3.40	2.47	2.78	3.39	4.01	2.47
2	3.40	3.70	3.40	2.47	2.78	2.78
3	3.40	3.40	3.40	1.85	2.47	3.09
4	2.78	1.85	2.16	2.16	2.16	1.85
Random average[3]	2.88	3.19	2.78	2.26	3.09	1.44

We summarise the speaker identification error rates of MLP-GMM experiments by using the four automatic speaker basis selection approaches in Table 11-5.

It is shown in Table 11-5 that all four speaker basis selections result in improved or equal performance relative to an improvement over the baseline system (equal error rate: 3.40%), except in one case. This proves the efficiency of using automatic speaker basis selection. In particular, method 4 works consistently better than all other approaches, with a 45.59% relative error reduction over the baseline system using only 50 speakers to train the MLP.

Moreover, In Morris et al. (2005), it was shown that when each test repeated 10 times, MaxAD showed clear advantage over random selection.

11.5.3 Conclusions

Discriminative feature enhancement by MLP preprocessing of standard MFCCs for a subset of the training speakers (speaker basis) can enhance speaker identification substantially. This is true for randomly selected speaker basis sets as well as proportionally balanced speaker basis sets. We also proposed several automatic speaker basis selection methods. Although these did not always improve the speaker identification results, a consistent improvement was found for automatic speaker basis selection when method 4 (maximum average distance) was used (lowest equal error rates in Table 11-5). This method showed a relative error reduction of 45.59% over the baseline system.

[3] The reported figures are the average error rates obtained by three repeats of random speaker selection, cf. Table 11-3.

11.6 Feature enhancement on the speech with additive noise

This section, adapted from (Wu et al. 2005c), addresses the issue of robust speaker recognition in noise, and in particular investigates the possibility of using a multi-layer perceptron (MLP) to enhance discrimination between speakers. Three different, realistic types of additive noise were selected and added to the 8kHz downsampled TIMIT (TIMIT-8k) clean-speech database (Garofolo et al. 1993) at different signal-to-noise ratios (SNRs).

State-of-the-art Gaussian mixture models (GMMs) for speaker recognition, like hidden Markov models (HMMs) in automatic speech recognition (ASR), can achieve very good performance in clean speech, but performance degrades strongly in the presence of noise (Reynolds et al. 1994). ASR performance in noise can be increased significantly by using a feature projection provided by the pre-squashed outputs from a one hidden layer MLP, pre-trained to output a posterior probability for each phoneme (Sharma et al. 2000). It is not possible to apply an MLP in the same way to speaker recognition. The reason is that in speaker recognition there are no fixed target classes like phonemes in ASR. For the purpose of speaker recognition, the MLP is trained with a representative subset of speakers *(speaker basis)* as its target classes, comparable to phonemes as target classes for ASR. The transformation which the MLP learns for the speaker basis has been shown beneficial for any other speaker from the same population for clean speech (Wu et al. 2005a).

Unlike for ASR, a one-layer MLP applied to speaker recognition in clean speech does not lead to an increase in the percentage correct speaker identification. This may be because speaker data, being clustered around every phoneme, is less easy to partition than speech data. But the separating power of an MLP can be increased by using more hidden layers. In (Wu et al. 2005a; Heck et al. 2000) an MLP with three hidden layers was trained to recognise 31 speakers, and discriminative features were taken as the outputs from the central, linear bottleneck hidden layer. The 31 speakers were selected because they had been recorded over multiple handsets. It was found that the features obtained from the MLP provide a performance enhancement, although not consistently across all training and test conditions (Heck et al. 2000) and sometimes only when the feature vector was concatenated with the original MFCC features which were used as input to the MLP (Konig et al. 1998). The good results may be due to a better compensation for the different handsets that were used, or to a better separation of the speakers in the acoustic space, even if the speakers were not selected with this aim -- or by a combination of the two.

In tests with TIMIT-8k in (Wu et al. 2005a; Morris et al. 2005) the automatic selection of the speaker basis to train the MLP was investigated with the specific aim of making an optimal selection of the target speakers for training the MLP. It was shown that the performance of the features provided by the MLP leads to good speaker identification even for a small number of target speakers used train the MLP. Clearly, the good results must be ascribed to the ability of the MLP to enhance discrimination between speakers, since no variable noise or channel conditions are present. When only a small set of speakers (speaker basis) is used to train the MLP, it is especially important that they are selected so as to represent the whole population. It was shown in (Wu et al.

2005a; Morris et al. 2005) that even a speaker basis of 50 speakers automatically selected on the basis of the GMM confusion matrix for 300 training speakers can lead to improved identification of 162 “unseen” test speakers for clean speech.

Here the effect of different types of additive noise, and the ability of the previously applied MLP to enhance speaker discrimination, in matched training and test conditions is investigated and compared with the system performance in clean speech (training and test) conditions. But in many practical applications, there is a mismatch between training and test conditions. Enrollment may take place in fairly clean speech conditions, e.g. when a new user has to go to a registration/certification authority so that his identity can be confirmed on the basis of official documents he possesses (passport, identity card, social security card, etc.). In this case, there will be a mismatch between the training data obtained during enrollment and the test data when the user requests verification in a noisy environment. We not only investigate the effect of a mismatch, where the training data is clean and the test data contain noise, we also evaluate the effect of simply adding several additive noise types to the training data, attempting to deal with the presence of noise in the test data.

In this section the effectiveness of data enhancement is tested, using the MLP from (Wu et al. 2005a; Morris et al. 2005) for speaker identification in different kinds of additive noise at different SNRs. Since MLPs have been shown to be able to deal well with noisy speech in ASR (Sharma et al. 2000), we expect they may also enhance speaker recognition when noise is present. In Section 11.6.1 we describe the data used in the experiments, the MLP used for feature enhancement and the baseline GMM speaker recognition system. Section 11.6.2 then describes the results of the experiments for speaker recognition in clean speech and noise. This is followed by a discussion of the results in Section 11.6.3 and conclusions in Section 11.6.4.

11.6.1 Method

1. Data

The TIMIT-8k (clean) speech database is used in all experiments (Garofolo et al. 1993). The reason for choosing this database is that we want to focus our investigations on the separation of speakers in the acoustic space first, and then add noise to evaluate its effect on MLP feature enhancement. Since the standard TIMIT-8k division does not include a development set, we created our own division into speaker-disjoint training, development and evaluation data, with 300, 168 and 162 speakers, respectively. The three sets are selected such that gender and dialect region have an equal proportional representation in the three sets. To make the speaker identification system text-independent, we used all sentences of type SA_{1-2}, SI_{1-2} and SX_{1-2} for training, sentences of type SX_3 and SI_3 for development and sentences of type SX_4 and SX_5 for evaluation. Whereas SA_1 and SA_2 sentences are always the same for different speakers, SI_n and SX_n sentences can be different ones and the index *n* only indicates the order as indicated by the numbers in the TIMIT database. The strict division optimises text-independence of the speaker recognition system.

2. Added noise

To evaluate the robustness of discriminative features for speaker identification in various kinds of noise, (stationary) car as well as (non-stationary) factory-1 noise and babble from the NOISEX-92 database (Varga et al. 1993) were added to the TIMIT-8k database at SNRs of 20, 10 and 0 dB, using the ITU software (ITU recommendation P.56, 1993) to determine SNRs (these three types of noise are more related to the application area focused in our work, e.g. car environments). These noises are also used in the Aurora evaluations for speech recognition in noise.

3. Feature extraction

The TIMIT speech data was first downsampled from 16 kHz to 8 kHz. At 16 kHz our baseline system (as in (Reynolds et al. 1995a)) obtains 100% correct speaker identification. However, the interest here is to work with speech data which is close to telephone quality. Using 20ms frames and a 10ms step size, 20 Mel-scaled filterbank log power features were extracted, using a Hamming window and a pre-emphasis factor of 0.97. A DCT was then applied to these to obtain MFCC features, from which the c0 energy coefficient was dropped. Time difference features were not appended, because these did not improve performance with TIMIT-8k. Neither silence removal, dynamic features or cepstral mean subtraction were used, since none of these led to any performance improvement with TIMIT-8k.

4. MLP feature enhancement

When an MLP is trained to map speech feature frames onto their phone class probabilities in ASR, not only are the MLP output values useful for deciding which class the speech frame belongs to, but the outputs from its hidden layers within the MLP are also discriminative for the phone classes which were the targets during training. Each unit in a standard MLP has a two stage function. The first stage, the net-input function, is a many-to-one linear combination of the neuron’s inputs. The second stage is a one-to-one non-linear sigmoid function which squashes the net-input to a value between zero and one. From the point of view of using the MLP internal feature representation to provide discriminative features, the squashed outputs are not very suitable because they tend to be close to zero or one, thereby not complying with the GMM assumption that all features have an approximately Gaussian distribution.

Using Torch (Collobert et al. 2002), discriminative preprocessing is carried out for the different noisy conditions with the aim of feature enhancement. Each single frame of the standard MFCC features are preprocessed by a 5-layer MLP, as in (Konig et al. 1998; Heck et al. 2000). This MLP was found to outperform MLPs consisting of fewer layers (Wu et al. 2005). Training the MLP with single frames instead of the usual input vector of 9 concatenated frames gives the best results for this particular database. The MLP is trained, by gradient descent, to maximise the cross entropy objective (i.e. the mutual information between the actual and target outputs). We trained in batch mode, with a fixed learning rate of 0.01. The data in each utterance was first normalised to have zero mean and unit variance. Of the 3 hidden layers, the first and last hidden layer, which are both non-linear, have 100 units and the middle, linear hidden layer has 19 (compression or bottleneck layer). The features obtained from the compression layer were used as input to a GMM system, as in (Wu et al. 2005a). The assumption behind this is that this simple representation, which consists of vectors of the same size as the original MFCC vectors, is an internal representation of the acoustic signal which enhances discrimination between the target speakers and is generalisable to the speakers in the entire population. We used the net input to the second hidden layer as input to GMM modelling. The MLP and its application in GMM modelling are represented in Figure 11-6. The MLP was trained with a fixed number of iterations (35), after which the error reduction on the training and development data in the MLP frame-based recognition was very small.

Instead of using the 6 training sentences of *all* speakers to train the MLP, it was shown in (Wu et al. 2005a; Morris et al. 2005) that a smaller selection of speakers, the speaker basis, is sufficient to obtain a good speaker discrimination. Here, a speaker basis consisting of 150 speakers was automatically selected from the training speakers. The automatic selection is made on the basis of the confusion matrix obtained from a GMM experiment. The confusion matrix is produced by classifying the MFCC features from the 2 evaluation sentences of the 300 training speakers with the speaker models for these speakers trained on the 6 training utterances (cf. next section). The log likelihoods in the confusion matrix are converted into likelihoods and then into posterior probabilities, by dividing each value by the row sum. The resulting table of posterior probabilities is then used to select the speaker basis.

The speaker basis which we use to train the MLP is selected using the speaker posterior probabilities $P_{ji} = P(S_j|X_i)$ for a set of development test data. These probabilities are obtained from the test data log likelihoods by dividing the log likelihood for each speaker by their sum over all speakers.

As a distance measure between speaker pdfs we use the symmetric Kullback-Leibler distance $KL(S_j, S_k)$ (Theodoridis et al. 2003). This cannot be evaluated in closed form when pdfs $p(X|S_j)$ are modelled by GMMs, but in (Morris et al. 2005) it is shown that $KL(S_j, S_k)$ is the expected value of $K(S_j, S_k, X)$, where

$$K(S_j, S_k, X) = \left(P(S_j \mid X) - P(S_k \mid X)\right)\left(\log P(S_j \mid X) - \log P(S_k \mid X)\right) \qquad (11.11)$$

$KL(S_j, S_k)$ can therefore be estimated by averaging $K(S_j, S_k, X)$ over the development test data.

$$KL(S_j, S_k) \cong \sum\nolimits_{X_i \in testSet} K(S_j, S_k, X_i) \quad = \sum\nolimits_i \left(P_{ji} - P_{ki}\right)\left(\log P_{ji} - \log P_{ki}\right) \qquad (11.12)$$

The resulting speaker-distance matrix can be used in various ways to select a subset of speakers for MLP training. The method which gave the best results for clean data in (Wu et al. 2005a; Morris et al. 2005) was to choose speakers in order of decreasing average distance from every other speaker (Maximum Average Distance).

5. GMM modelling

The MFCCs or, alternatively, the enhanced features obtained from the compression layer of the MLP (as explained in the previous section) are used as input to GMM modelling of the diagonal covariances using 32 Gaussians, as in (Reynolds et al. 1995a; 1995b). The baseline model trained with MFCCs of the 6 training utterances gives state-of-the-art speaker recognition performance. With TIMIT-8k (though not with other databases, such as the CSLU speaker recognition database) no gain is found in training speaker models by adaptation from a global model for all 300 training speakers, so that each speaker model was trained from scratch with data for that speaker only.

As in (Reynolds et al. 1995b), GMMs are trained by *k*-means clustering, followed by EM iteration. This is performed by the Torch machine learning API (Collobert et al. 2002), using a variance threshold factor of 0.01 and minimum Gaussian weight of 0.05 (performance falling sharply if either was halved or doubled), determined on the basis of the development sentences of the development speakers.

Test results are obtained for 162 test speakers (for two test sentences per speaker, cf. section 11.6.1-1). Speaker identification for utterance feature data X is performed by selecting the speaker S_j with the largest posterior probability, $P(S_j|X)$ (which corresponds here to the largest data likelihood $p(X| S_j)$, as all speaker priors $P(S_j)$ ar equal).

11.6.2 Results

In this section, the speaker identification results are compared for the different noises and at different SNRs. The MLP-enhanced features are compared with the baseline system, in which the MFCCs are not preprocessed by the MLP and used as input to GMM directly. Table 11-6 shows the results for clean data, and for car, factory and babble noise at SNRs of 20, 10 and 0 dB. The conditions presented in Table 1 are all *"matched"* conditions, in which the test data were used with a system trained on data of the same noise type and at the same SNR.

Table 11-6: **Speaker identification error for training on matched noise type and level**

	clean	car			factory			Babble			mean
		20 dB	10 dB	0 dB	20 dB	10 dB	0 dB	20 dB	10 dB	0 dB	
MFCC baseline	3.40	11.42	18.21	38.89	18.83	43.83	87.35	17.90	36.11	87.65	36.36
MLP-enhanced	1.85	6.48	12.96	26.85	15.74	40.43	82.72	10.80	25.00	81.17	30.40

The results show a strong increase in speaker recognition error with decreasing SNR. Although still well above chance level, speaker identification is particularly poor for the non-stationary noise types (babble and factory noise) at a SNR of 0 dB. Notice that no cepstral mean subtraction (CMS) was performed, even for the noisy data, which may result in better speaker identification performance. In the clean speech condition CMS leads to poorer performance, probably because subtraction of the spectral mean across an utterance filters out part of the speaker characteristics. As expected, the best results are found for clean speech.

Preprocessing of the MFCCs by an MLP to enhance speaker discrimination *always* reduces the speaker identification error. The absolute reduction is greatest for the stationary car noise, particularly at the lowest SNR. The positive effect, though present in all conditions, is greatly reduced for non-stationary noise types, with the exception of babble at an SNR of 10 dB.

In many applications, the user may want to be recognised in widely varying conditions, but the condition in which he must be recognised is not known beforehand. Two scenarios are possible. In the first scenario, the speaker enrolled in the system in a quiet environment, so that the speaker model (and the MLP) is trained on clean speech. But the actual conditions in which the system is subsequently used may vary from one occasion to the next. In order to evaluate the performance of our system under these *mismatched* conditions, the test data from all the noisy conditions in Table 11-6 were scored with the GMM speaker models (and MLP) trained with clean speech only. (The data for clean speech are the same as in Table 11-6 and does not represent a mismatch. It is only included in Table 11-7, because it is used to compute the mean percentage error for correct speaker identification across all possible test conditions in the right-hand column.)

Table 11-7: **Speaker identification error for training on clean speech and testing in various conditions**

	clean	car			factory			Babble			mean
		20 dB	10 dB	0 dB	20 dB	10 dB	0 dB	20 dB	10 dB	0 dB	
MFCC baseline	3.40	51.23	72.22	93.52	83.95	98.15	99.07	74.38	88.89	97.53	76.23
MLP-enhanced	1.85	79.63	88.27	96.60	95.37	98.46	99.38	91.05	94.44	97.84	84.29

As the results in Table 11-7 show, the mismatch between the noisy test data and the clean training data causes a severe deterioration of the performance of the speaker recognition system. For some of the conditions, recognition is only just above chance level (p=100/162=0.62%, i.e. error=99.38%). The effect, which is present for the MFCC features (comparison of first data rows in Table 11-6 and Table 11-7), is even greater after MLP enhancement, with only chance level speaker identification for factory noise at 0 dB SNR.

In the case of (known) additive noise, the noises can easily be used to create “virtual” data containing this noise before the speaker model is trained. By catering for a variety of testing conditions in the training phase, the system is expected to better cope with the variability in real test conditions. As the results in Table 11-8 show, the performance in all noisy conditions is substantially better than when the GMM speaker models (and the MLP) are trained on clean speech only.

Table 11-8: **Speaker identification error for training across all noisy conditions**

	clean	car			factory			Babble			mean
		20 dB	10 dB	0 dB	20 dB	10 dB	0 dB	20 dB	10 dB	0 dB	
MFCC baseline	14.20	12.96	19.75	56.48	19.75	36.11	85.19	17.90	25.62	74.69	36.27
MLP-enhanced	14.51	12.35	11.11	33.02	12.65	33.02	83.02	14.51	18.52	70.37	30.31

In some cases, the speaker identification error is even lower than in the matched noise condition in Table 11-6, e.g. for factory noise at 10 dB SNR and for babble at 10 and 0 dB SNR. In all conditions except for clean speech is the speaker identification error lower when MLP-enhanced features are used compared to the baseline system using MFCCs.

11.6.3 Discussion

In comparison to the stationary car noise, the speaker identification error is very high for non-stationary noises (factory noise and babble) with low SNRs.

As expected, a mismatch between (clean) training and (noisy) test data always leads to a strong deterioration in performance. This deterioration is even greater when an MLP-enhancement based on clean speech is performed on the MFCC features, since it is not appropriate for the noisy acoustic space.

Both for matched training and test conditions and for GMM speaker models trained across noisy conditions (i.e. using both clean speech and the same signals with added noise to train the speaker models), the MLP-enhanced features always outperform standard MFCC features.

The GMM speaker models trained across noisy conditions even outperform GMMs trained on the same noise type as used for testing (matched condition). This is particularly the case for test signals with added factory noise at 10 dB SNR as well as for added babble at 10 and 0 dB SNR. The differences are not just caused by the different speaker basis selections for matched (Table 11-6) versus multi-condition models (Table 11-8), because the effect is found both for MLP-enhanced and for MFCC features. The effect is most likely due to an under-representation of the intra-speaker variance in the limited available training data in the matched tests, which is compensated for by the (artificial) addition of noisy data in the multi-condition tests. It is well-known in speaker recognition that the amount and above all the variety of training data from each speaker is critical. But the results do not show a systematic pattern across noise type and SNR, so that further investigations are needed to fully understand the observed effect.

Of course, the MLP preprocessing described here must also be used for other, more realistic data, e.g. NIST. This was done by (Konig et al. 1998; Heck et al. 2000), but there the speaker basis was selected to contain speakers who used all different handset types, so that it is not clear whether the results are due to the MLP performing speaker discrimination or compensation for different handset types. The results shown here lead us to believe that the MLP can discriminate between speakers. The MLP feature enhancement may therefore also be helpful for databases where there is no subset of speakers who used all different handset types, but this remains to be verified.

11.6.4 Conclusions

In this section, text-independent speaker identification was performed in clean and noisy conditions. The performance of GMM speaker modelling using standard MFCC features was compared with MLP-enhanced features, where the MLP was trained on a subset of the speakers which was not used in testing (hence the feature enhancement is speaker-independent). MLP-enhanced features strongly improve speaker identification performance, except when the noise condition of the test data is not represented in the training data. It was shown that, as in ASR, speaker identification in matched and multi-condition training is considerably better than when there is a mismatch between training and test data, both with MFCC and MLP-enhanced features. In some cases, GMM speaker models trained across noisy conditions, either with MFCCs or with MLP-enhanced features, perform better than speaker models trained in matched training and test conditions. An explanation is offered in terms of the amount and variability in the training data, but this cannot fully explain the observed results.

11.7 Enhancement test on telephone speech

We have tested MLP-based feature enhancement for the low-bandwidth clean speech (Section 11.4) and the speech with additive noise (Section 11.6). In this section, we apply this approach to channel noise, which is always present in telephone speech, caused by the transmission of speech signals through the telephone lines. Since MLP-based feature enhancement was shown to be able to eliminate the variance caused by the non-speaker related characteristics (e.g. additive noise), it was also expected to be efficient in the alleviation of the effect of channel noise.

A telephone speech database NTIMIT was adopted for this evaluation. In order to keep the tests compatible with the ones conducted in Section 11.4, the same training, development and testing configurations were used. Regarding the baseline identification system, feature generation, MLP training and all other related setups, the same procedures and values for the parameters were used as in Section 11.4.

Table 11-9: Percentage correct identifications from an MLP-GMM system for different speaker basis sizes, with speakers selected using the Max-AD method (baseline 58.95%)

speaker basis size	30	50	60	75	100	150
percent	55.56	58.02	57.10	59.88	63.89	59.57

From the above table, it may be seen that, firstly, MLP-based feature enhancement approach improves the performance of speaker recognition on telephone speech with channel noise. Secondly, it is also worth noting that the transformed system does not outperform the baseline system until the number of speaker basis used for discriminative training is larger than 75, suggesting that speaker basis for less than this number cannot represent the overall distributional characteristics of the speaker feature space, due to the presence of the channel noise (more variance caused by the channel noise). Thirdly, 100 basis speakers are shown as the optimal selection, while a speaker basis of more than 100 reduces the recognition performance, although it is still superior to that of the baseline system.

Based on this section, as well as Section 11.4, 11.5 and Section 11.6, the conclusion may be drawn that MLP-based feature enhancement can substantially increase between-speaker discrimination by reducing or, indeed, eliminating most of the disturbing variance caused by additive and channel noise. Moreover, this set of approaches may be also used for other pattern recognition applications to enhance between-class discrimination.

11.8 Summary

In this chapter, the MLP-based feature enhancement approaches were proposed and tested for different types of databases: low-bandwidth clean speech, additive noise speech and telephone speech. All these tests confirmed the effectiveness of this set of approaches. The same approaches may be expected to be applicable to other pattern recognition applications.

12. Complementary features by SOM processing

12.1 Introduction

Multi-stream speech processing operates on multiple different representations of the speech signal either at the feature level (concatenating them and then treating them as a single stream), at the frame level (combining the likelihood of each stream for each frame), or at the score level (processing each stream separately and then combining the results) (Hagen et al. 2003; Morris et al. 2001).

While speaker recognition accuracy can be quite high using a single feature stream, it is often possible to further enhance the level of accuracy by multi-stream processing. In particular, it can alleviate the problem of performance degradation under noisy conditions.

The additional benefit provided by each new feature stream in a multi-stream system depends on the degree to which the information in the new stream is complementary to the information already available. In this paper we investigate the combination of MFCC features, commonly used in speaker recognition systems, with features derived through the projection of these MFCC features onto a trained Kohonen self-organizing map (Kohonen et al. 1997). In the case of a 2 dimensional SOM, the trace of the cells in the map visited throughout the utterance of a short spoken prompt provides a 2D trajectory which can be processed in an analogous way to that in which signature recognition is usually performed. This would have the additional advantage that, in a multimodal system which uses both voice and signature, such as that used in the SecurePhone project (Allano et al. 2006), the voice and signature data, after SOM projection, can share the same processing. SOM trajectory coefficients (STC), besides capturing local information in the SOM coordinates (and their derived features), also allow us to model the global shape of the voice "signature". As in signature verification, STC features would consist not simply of SOM trajectory coordinates, but also of features derived from these.

In Section 12.2 we describe how we can use a SOM for projecting high dimensional speech data down to a low dimensional SOM, demonstrating the tonotpic organisation on the basis of a labelled database. In Section 12.3 we show some visual examples of the use of a 2-dimensional SOM to represent the speaker trajectory for a given prompt. In Section 12.4 we describe how the SOM coordinates are augmented by the addition of a number of derived features in a manner analogous to the processing used in signature recognition. In Section 12.5 we present our baseline GMM based system for speaker recognition. In Section 12.6-8 we describe the tests we have made, the results of which are presented in Section 12.9, both for the concatenation of MFCC with STC (SOM trajectory coordinates) features and for MFCC with STC scores fusion. A discussion and conclusion follow in Sections 12.10 and 12.11.

12.2 SOM projection

The SOM training procedure (Kohonen et al. 1997, 1996) is a form of unsupervised clustering which is in some ways similar to K-means clustering. What distinguishes them is that in SOM clustering the cluster centres are arranged in a regular grid (normally in two dimensions), in which cluster centres which are close in the grid are also close to eachother in the codebook vector space. We will refer to a SOM which has this property as being “well organised”.

Although the training procedure for a SOM is well known, it includes a number of steps the details of which often differ between implementations. These differences can have a significant effect on the outcome of SOM training, so we give here some of the details of our implementation. The SOM training procedure was implemented, in C++ and with the aid of the Torch API (Collobert et al. 2002), using the algorithm from Kohonen et al. (1996). All training (and test) vectors are first normalised to have unit length. SOM codebook vectors are initialised to the value of randomly selected training data vectors (MFCC speech frames). We will refer to the individual speech feature frames x_t used in SOM training as training tokens.

Let *t* be the absolute token count and *u* the token count within the training set. Let *r(t)* be the update radius and *h(t,r)* be the learning rate. Let the closest codebook vector to the current training token be referred to as the *active* codebook vector. Let *dst* be the Euclidean distance between the active codebook vector and vector being updated.

For each training token x_t , all codebook vectors m_i within grid distance *r(t)* of the active codebook vector are updated according to (12.1) and then renormalized to have unit length.

$$m_i = m_i + h(t,r).(m_i - x_t) \tag{12.1}$$

The radius *r(t)* of the update neighbourhood and the learning rate *h(t,r)* are updated once every bsize codebook updates, according to (12.2)(12.3)(12.4).

$$len = \frac{s \cdot epochs}{bsize} \tag{12.2}$$

$$r(t) = 1 + \frac{(r(0)-1)\cdot(len-t)}{len} \tag{12.3}$$

$$h(t,r) = 0.05\frac{len}{len+100.t}\exp\left(\frac{-dst^2}{2r^2(t)}\right) \qquad (12.4)$$

When the trained SOM is used to map a given pattern vector, x, onto the SOM grid, we will refer to the SOM grid coordinates of the SOM codebook vector which is closest to this vector as the 2D "SOM projection" of x. In this way a trained 2D SOM can be used to project a data set with N dimensions onto a corresponding dataset with just 2 dimensions.

Projection of data onto a 2D SOM is often used as a tool for visualising high dimensional data. If the intrinsic dimension of the pattern data has more than two dimensions, then the SOM training algorithm will not usually be able to produce a well organised 2D map. However, in practice it is quite common for a 2D SOM to organise in this way. For example, it is well known that when a SOM is trained on the acoustic feature vectors for speech data which is restricted to vowels, it will produce a well organised "tonotopic map" (Kohonen et al. 1997) which resembles some symmetry of the "vowel triangle" shown in Figure 12-1.

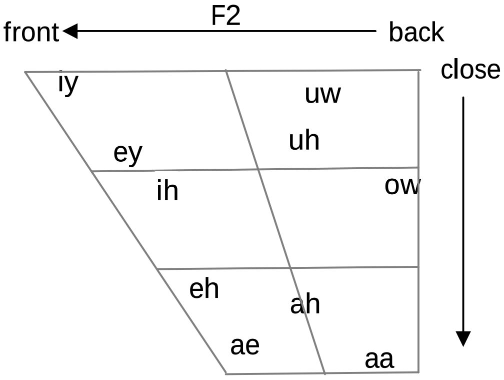
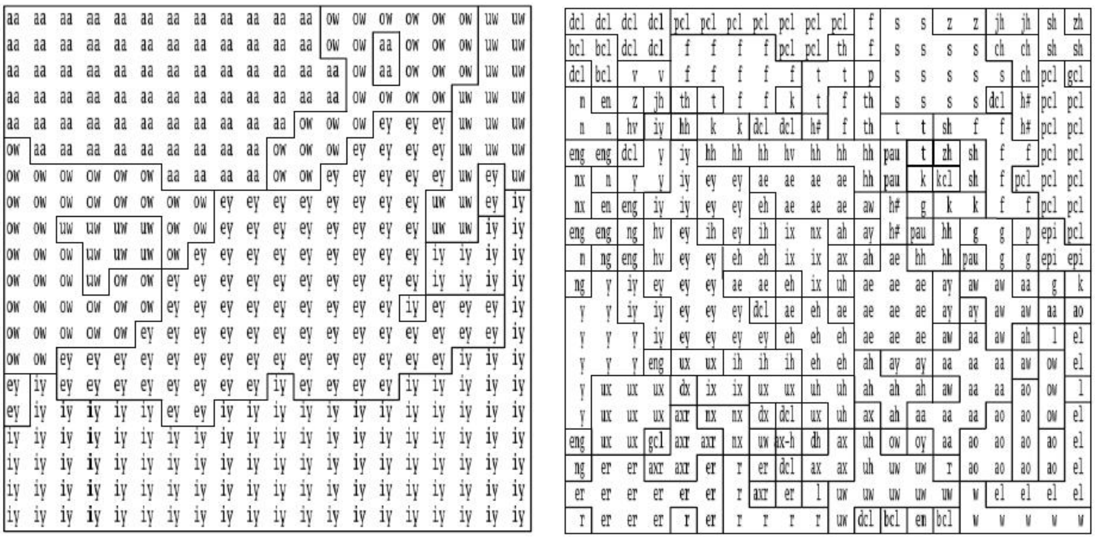


Figure 12-1: Vowel triangle for Am.E. (Handbook, 1999; IPA)

This results from the fact that, at a first approximation, all vowels are perceived according the centre frequency of their two first vocal tract resonances, or "formants" (F1, F2). This means that, no matter what type of acoustic features are used to represent the speech data, this data always has an intrinsic dimension of 2. In this case, if the sequence of vowel sounds pronounced varies continuously in time, then the 2D trajectory of the corresponding SOM projection of this speech data will also vary continuously in a smooth path moving over the SOM grid.

Figure 12-2a shows a SOM which was trained with all realisations of the vowels /iy, ey, aa, ow, uw/ in the TIMIT database (Garofolo et al. 1993). TIMIT was used for this labelled SOM instead of the CSLU Speaker Recognition corpus, which is used in the experiments in the rest of this paper (Cole et al. 1998), because, unlike TIMIT, the CSLU database is not phonetically labelled. Despite the variability in the realisation of each vowel, the acoustic similarity between their different realisational variants leads to their self-organisation in large contiguous areas representing each of the vowels. Acoustically more similar vowels, e.g. /iy/ and /ey/ or /uw/ and /ow/, are generally closer together in the SOM.



aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	ow	ow	ow	ow	ow	ow	uw	uw
aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	ow	ow	aa	ow	ow	ow	uw	uw
aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	ow	aa	ow	ow	ow	uw	uw
aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	ow	ow	ow	ow	uw	uw	uw
aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	aa	ow	ow	ow	ey	ey	ey	uw	uw	uw
ow	aa	aa	aa	aa	aa	aa	aa	aa	aa	ow	ow	ow	ey	ey	ey	ey	uw	uw	uw
ow	ow	ow	ow	ow	ow	aa	aa	aa	aa	ow	ow	ey	ey	ey	ey	ey	uw	ey	uw
ow	ow	ow	ow	ow	ow	ow	ow	ey	ey	ey	ey	ey	ey	ey	ey	uw	uw	ey	iy
ow	ow	uw	uw	uw	uw	ow	ow	ey	ey	ey	ey	ey	ey	ey	ey	uw	uw	iy	iy
ow	ow	ow	uw	uw	uw	ow	ey	ey	ey	ey	ey	ey	ey	ey	ey	iy	iy	iy	iy
ow	ow	ow	uw	ow	ow	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	iy	iy	iy	iy
ow	ow	ow	ow	ow	ow	ey	ey	ey	ey	ey	ey	ey	ey	ey	iy	ey	ey	ey	iy
ow	ow	ow	ow	ow	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	iy
ow	ow	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	ey	iy	iy	iy
ey	iy	ey	ey	ey	ey	ey	ey	ey	ey	iy	ey	ey	ey	ey	iy	iy	iy	iy	iy
ey	iy	iy	iy	iy	iy	ey	ey	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy
iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy
iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy
iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy
iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy	iy

dcl	dcl	dcl	dcl	pcl	pcl	pcl	pcl	pcl	pcl	pcl	f	s	s	z	z	jh	jh	sh	zh
bcl	bcl	dcl	dcl	f	f	f	f	pcl	pcl	th	f	s	s	s	s	ch	ch	sh	sh
dcl	bcl	v	v	f	f	f	f	f	t	t	p	s	s	s	s	s	ch	pcl	gcl
n	en	z	jh	th	t	f	f	k	t	f	th	s	s	s	s	dcl	h#	pcl	pcl
n	n	hv	iy	hh	k	k	dcl	dcl	h#	f	th	t	t	sh	f	f	h#	pcl	pcl
eng	eng	dcl	y	iy	hh	hh	hh	hv	hh	hh	hh	pau	t	zh	sh	f	f	pcl	pcl
nx	n	y	y	iy	ey	ey	ae	ae	ae	ae	hh	pau	k	kcl	sh	f	pcl	pcl	pcl
nx	en	eng	iy	iy	ey	ey	eh	ae	ae	ae	aw	h#	g	k	k	f	f	pcl	pcl
eng	eng	ng	hv	ey	ih	ey	ih	ix	nx	ah	ay	h#	pau	hh	g	g	p	epi	pcl
n	ng	eng	hv	ey	ey	eh	eh	ix	ix	ax	ah	ae	hh	hh	pau	g	g	epi	epi
ng	y	iy	ey	ey	ey	ae	ae	eh	ix	uh	ae	ae	ae	ay	aw	aw	aa	g	k
y	y	iy	iy	ey	ey	ey	dcl	ae	eh	ae	ae	ae	ae	ay	ay	aw	aw	aa	ao
y	y	y	iy	ey	ey	ey	ey	eh	eh	eh	ae	ae	ae	aw	aa	aw	ah	l	el
y	y	y	eng	ux	ux	ih	ih	ih	eh	eh	ah	ay	ay	aa	aa	aa	aw	ow	el
y	ux	ux	ux	dx	ix	ix	ux	ux	uh	uh	ah	ah	ah	aw	aa	aa	ao	ow	l
y	ux	ux	ux	axr	nx	nx	dx	dcl	ux	uh	ax	ah	aa	aa	aa	ao	ao	ow	el
eng	ux	ux	gcl	axr	axr	nx	uw	ax-h	dh	ax	uh	ow	oy	aa	ao	ao	ao	ao	el
ng	er	er	axr	axr	er	r	er	dcl	ax	ax	uh	uw	uw	r	ao	ao	ao	ao	el
er	er	er	er	er	er	r	axr	er	l	uw	uw	uw	uw	uw	w	el	el	el	el
r	er	er	er	r	er	r	r	r	r	uw	dcl	bcl	em	bcl	w	w	w	w	w

Figure 12-2: SOM for vowels only (a) (left) and for all phones (b) (right)

In the application of SOM projection which we are investigating in this article we should like to be able to project all phonemes onto a single SOM in such a way that all speech trajectories are quite smooth. When consonants are represented together with vowels in the same SOM, however, its structure becomes somewhat less clear. Figure 12-2b visualises a SOM based on all the phones in the TIMIT database. As in the vowel SOM in Figure 2a, contiguous areas representing the same phone can be recognised. Although the phone map has a higher intrinsic dimensionality than the vowel map, we can clearly recognise that acoustically similar phones are located close together in the SOM. For instance, plosive closures are often close together, like /pcl, bcl, dcl/ in the top left-hand corner, labiodental fricatives /f, v/ are close together, syllabic consonant /el, er, em, en, eng/ are close to their non-syllabic counterparts /l, r, m, n, ng/ and, as in Figure 12-2a, acoustically similar vowels are closer together.

12.3 Speaker voice signature

Although it is clear that SOMs can be used to represent speech, as in (Kohonen et al. 1988), it is not immediately obvious that the self-organising structure retains the finer distinctions between speakers in the way they produce the same phoneme; the discretisation caused by the size of the SOM dimensions may not be fine enough to retain speaker differences, which are more subtle than the distinctions between phones.

Each time a speaker produces an utterance, a corresponding graphic pattern can be visualized by showing the mapping of each MFCC frame for a prompt to the SOM coordinates. The resulting graphic pattern is referred to as a Speaker Voice Signature

(SVS). It is defined as the speaker-specific trajectory for an utterance visualized by a trained SOM classifier, showing some similarity to a written signature. Speaker differences for a given prompt are reflected in differences in the trajectories through the SOM space. Figure 12-3 demonstrates the trajectories for the speech signal corresponding to “two four” from the same prompt spoken by two speakers. Although the intra-speaker differences (comparison between left and right figures) are smaller than the inter-speaker differences (comparison between top and bottom figures), they are fairly subtle.

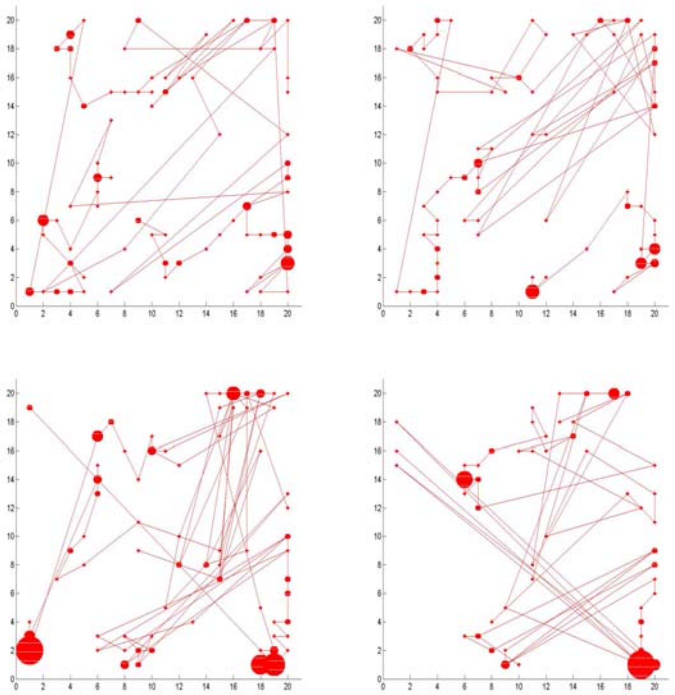
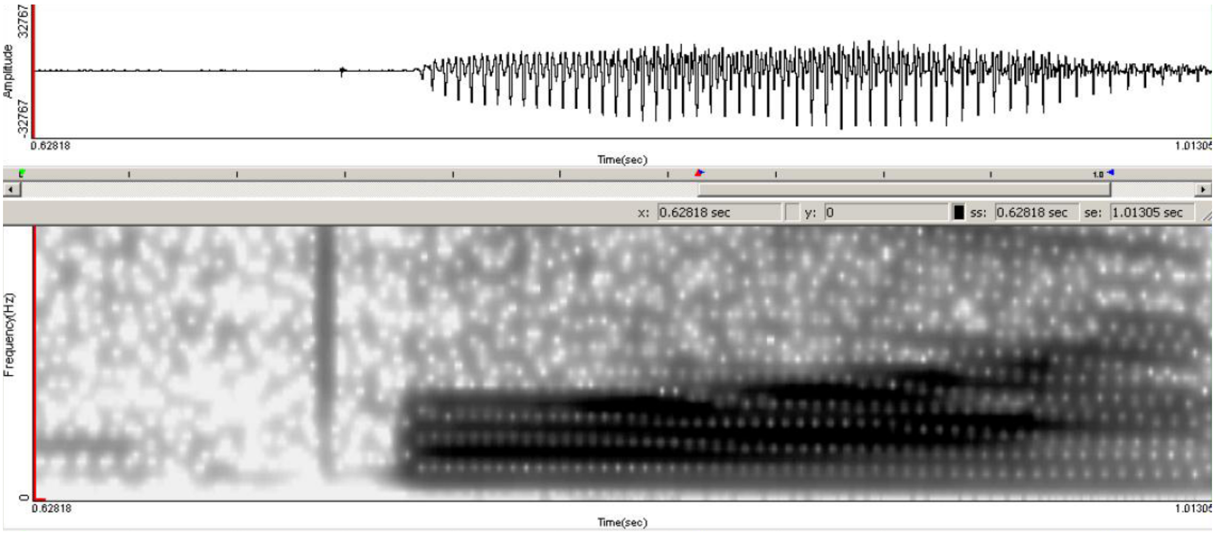


Figure 12-3: Top: Trajectory from 2 repetitions of “two four” by speaker 0038. Bottom: 2 repetitions of the same prompt by speaker 0040. These trajectories were projected from their 38-*d* MFCCs by a SOM classifier trained on the training set of the CSLU database (cf. the description in Section 12.6 and 12.7)

The speaker discriminating information is illustrated not only by these trajectories, but also by the time duration for which the trajectory stays in a position. A bigger blob shown in each position implies that the trajectory stays there for a longer time. In fact, the distribution of the time duration can be efficiently modelled by a GMM with the two x-y coordinates of the SOM features. The global shape of each SVS can be captured by the other six components of the SOM features such as x-y speeds, x-y accelerations and the curvature.

Because it is not immediately obvious from the representations in Figure 12-3 that the speech signal for “two four” actually leads to a smooth trajectory, the speech signal corresponding to the last part of the prompt represented in top left-hand figure is represented together with the x-y coordinates for each frame of the signal in Figure 12-4. Here it is clear that the trajectory is quite smooth. The trajectories in Figure 12-4 appear to show a large number of sudden jumps in the speech trajectory (indicated by the light vertical lines between the x-y coordinates). However, most jumps appear in low energy regions where the mapping onto SOM coordinates becomes arbitrary, while slowly changing acoustic signals correspond to smooth coordinate paths.



x: 0.62818 sec y: 0 ss: 0.62818 sec se: 1.01305 sec

x	20	9	9	8	19	2	8	15	13	17	19	20	18	20	18	20	20	20	20	17	20	20	20	20	20	20	19	19	18	17	17	17	16	15	14	12	12	11
y	12	20	20	18	18	1	4	12	16	20	20	1	1	3	2	3	3	3	3	1	4	4	4	5	5	5	5	5	5	5	7	7	6	5	4	3	3	2

Figure 12-4: Representation of a speech signal for "four" excised from CSLU file 0038aaq1.wav (oscillogram + spectrogram) together the with x-y coordinates of the units in the SOM activated by each frame

12.4 Computing SOM trajectory coefficients

In on-line handwritten signature verification, the feature vector associated with each pen position (x-y coordinates, plus pen pressure and angles) is generally augmented with a number of derived parameters before it is submitted for data modelling. These extra parameters can be derived from local features associated with the first and second time derivatives of the position, including velocity, acceleration, line direction, and curvature. They can also be derived from global features, such as the first and second moments, which carry information about the overall shape of the trajectory. Exploiting the analogy with on-line signature verification, the vector of SOM projection coordinates associated with each speech frame (which we refer to as "raw" SOM features) is similarly augmented.

In this paper we only look at augmenting STCs with time derivative based features. Time difference coordinates for a sequence of points *x(t)* on a smooth trajectory could be estimated simply as *x(t+1)-x(t-1)*. However, a smoothing is normally applied to this estimate by estimating the direction of the velocity vector as that of the least squares fit straight line between the points *x(t-W)* to *x(t+W)* inclusive, for some given smoothing window size, *W*. Indeed it is clear from Figure 12-3 that SOM trajectories are generally not entirely smooth. This is partly because of the discrete nature of the SOM grid positions, but it is also partly due to the imperfectly tonotopic mapping which any SOM trained on unconstrained speech data will tend to exhibit. If *x(t)* is any coordinate at time *t*, (12.5) shows the regression formula of order W (Soong et al. 1988; Van et al 2004).

$$reg(x(t),W) = \frac{\sum_{k=1}^{W} k(s(t+k) - x(t-k))}{2\sum_{k=1}^{W} k^2} \qquad (12.5)$$

As the SOM clustering algorithm can be applied not only to a 2D grid, but to a grid with any number of dimensions, we are interested in testing the suitability of SOM grids with nor only 2 dimensions, but also 3 or more dimensions.

The augmented *stc(t)* feature vector which we derive from the N-dimensional SOM coordinate data *x(t)*, are as follows.

- **Position**. Handwritten signature coordinates are relative to the signature centre of gravity, but the SOM grid position is highly significant, so we use absolute SOM coordinates. Let d denote dimension.

$$\textit{For } d = 1..N, \text{ use } x(d,t) \qquad (12.6)$$

- **Velocity**. *For d = 1..N, use*

$$v(d,t) = \frac{dx(d,t)}{dt} \cong reg(x(d,t),W) \qquad (12.7)$$

- **Line angle**. *For d = 2 to N, use*

$$\psi(d,t) = \arctan(v(d,t)/v(d-1,t)) \qquad (12.8)$$

- **Curvature**. *For d = 2 to N,* for arc length *s*, use

$$\kappa(d,t) = \frac{d\psi(d,t)}{ds} = \frac{d\psi(d,t)}{dt} / \frac{ds}{dt} \qquad (12.9)$$
$$\cong reg(\psi(i,t),W)/\|v(t)\|$$

Further time derivatives: $\frac{d\psi(i,t)}{dt}$ and $\frac{d\kappa(i,t)}{dt}$

For 2 SOM dimensions, stc(t) has *8* coordinates. For N SOM dimensions, stc(t) has *6N-4* coordinates.

12.5 Baseline speaker recognition system

The speech signals were recorded over a digital telephone line at a sampling rate of 8 kHz and with a 16-bit amplitude resolution. From these signals, 20 Mel-scaled filterbank log power features were extracted over 20ms frames and with a 10ms step size, using a Hamming window and a pre-emphasis factor of 0.97. A DCT was then applied to these to obtain MFCC features, from which the c0 energy coefficient was dropped. Cepstral mean subtraction (CMS) was applied to the MFCC vector and time difference features were appended. Thus, 38-dimensional MFCC vectors were used as the input data for the experiments reported in this chapter.

Both speaker identification and verification tests use state of the art systems based on Gaussian mixture models (GMMs) and MFCC speech features. As in (Reynolds et al. 1995) our identification system trained a UBM on data from a large set of speakers who are not used in testing. A client GMM distribution is then trained on data from each client, using only data for the prompt being tested (in this case the first numbers prompt, "58312"). The client model is initialised equal to the UBM, after which the Gaussian means alone are updated using MAP adaptation (Mariethoz et al. 2000). In testing, the speaker for each test utterance is selected as the person who has the highest speech data likelihood. In verification the UBM and client models are trained in the same way, and a test utterance is accepted as having the claimed identity if the UBM-normalised data likelihood is above a certain threshold, where the value of this threshold is set to give the maximum verification score on a development test set. GMM training and testing used the Torch machine learning API (Collobert et al. 2002).

12.6 Speaker recognition experimental protocol

The same 5-digit sequence ("5 3 8 2 4") spoken by 61 speakers was selected from the CSLU Speaker Recognition corpus (Cole et al. 1998). The prompts were recorded in four sessions, with four repetitions of the prompt in each session. Because we are particularly interested in speaker recognition for very small amounts of data, we only selected three sessions per speaker, one for training, one for development and one for testing. Only sessions in which all the prompts were produced correctly were selected. For identification, one session was used for GMM training for each speaker, one session was used for development and one session was used for evaluation. The other session was not used. For verification, the experimental protocol was the same, i.e. one session was used for GMM training, development and testing. A UBM for each speaker was trained on one training session across all the speakers except the claimed speaker in a similar way as it was used by Reynolds et al (1995a, 1995b, 2000).

In order to increase the chances of finding a SOM which is well organised for all speech sounds together, we experiment with SOMs with dimension from one to seven, with different numbers of grid cells per dimension, and using different numbers of updating iterations in SOM training.

12.7 Data

These MFCC features were either used as input to the GMM directly (baseline experiment), or were used as input to the SOM, which varied in the number of dimensions or in the size of the dimensions. After allowing the SOM to self-organize on the basis of the training data obtained from a single session, with data from another session being used for optimization, it was used to map the MFCC features onto the x-y coordinates of the SOM, from which the parameters described in Section 12.4 were then derived. The SOMs that were used for MFCC feature projection onto x-y coordinates was varied in the number of dimensions, which was varied between 2 and 5, keeping the total number of units in the SOM roughly the same. Best results were obtained for 2- and 3-dimensional SOMs, so only these are presented here. For the 2-dimensional SOM, the size of the dimensions was also varied (but always the same for each dimension). The speaker's voice SOM trajectory coefficients which derived from x-y coordinates of the trajectory through the SOM was used as input to GMM, described in the following section.

12.8 Fusion

The effect of fusion is investigated by comparing early fusion, in which the STC parameter vector is concatenated with the MFCC vector before it is used as input to the modelling stage, with late fusion, in which the scores of the separately modelled MFCCs and STCs were linearly combination. The weights given to each type of scores (speaker recognition scores for MFCCs and STCs, respectively) are varied from 0 to 1 in steps of 0.1, with their summation equal to 1.

12.9 Results

In this section the results for speaker identification as well as for speaker verification experiments are presented. In speaker identification, the system's task is to determine which of the 61 speakers is the most likely to have uttered a given test prompt. In speaker verification, the system must accept or reject the claimed identity of the speaker. Only results for the optimal smoothing factor (sm.) and MFCC/STC weight are presented in the tables. The results of the speaker identification experiments using STCs are presented in Table 12-1. The baseline result for speaker identification for MFCCs is **20.9%** identification error.

Clearly, the STCs obtained from the SOM always lead to a substantially higher speaker identification error than the baseline. Also, early fusion does not improve speaker identification, and is between 2.9 and 5.7 percent points higher for the SOMs presented. Late or score fusion, on the other hand, does lead to a modest improvement in speaker identification. The improvement for late fusion of the scores for MFCCs with those of STCs obtained from a 20x20 SOM is 1.2 percent points, or 6% relative to the baseline speaker identification error.

Table 12-1: Identification percentage error for STCs alone and in early and late fusion with MFCCs, for different SOM sizes and number of dimensions (sm = optimal smoothing factor, cf. (12.5))

Size	num cells	no fusion	early	Late	sm.	MFCC wt
10x10	100	55.33	23.77	20.08	5	0.7
20x20	400	47.13	25.41	**19.67**	4	0.7
30x30	900	50.82	24.59	19.67	6	0.7
7x7x7	343	63.52	26.64	20.90	3	1.0

Speaker verification results for the same SOMs as in Table 12-1 are given in Table 12-2. The baseline EER for GMM of MFCC parameters is **8.49%**. With an absolute difference of 6.6 – 11.1 per cent points, STCs perform about twice as poorly as MFCCs. In contrast to the speaker identification experiments, early fusion can improve speaker verification (with a 0.8 per cent point improvement for STCs derived from the 20x20 SOM coordinates). For late fusion, the EER is always lower than for MFCCs on their own. The 20x20 SOM leads to a 0.9 per cent absolute or a 10.2% relative error reduction.

Table 12-2: Verification percentage EER for STCs alone and in early and late fusion with MFCCs, for different SOM sizes and No. of dimensions

Size	num cells	no fusion	early	Late	sm.	MFCC Wt
10x10	100	19.63	8.99	8.15	6	0.9
20x20	400	15.13	7.72	**7.62**	4	0.5
30x30	900	16.50	8.23	8.07	6	0.9
7x7x7	343	18.00	9.20	8.10	3	0.9

12.10 Discussion

The results of our experiments have shown that STCs derived from the x-y coordinates of a SOM trajectory can enhance speaker recognition when combined with MFCC vectors. This is particularly true when late or score fusion is used to combine the speaker recognition results for the two parameter types. For speaker verification an improvement was also found for combination by early fusion.

A 2-dimensional SOM with 20x20 units gave best performance, both for speaker identification and for speaker verification. It should be noted that the SOMs were trained with an extremely small amount of data from each speaker, so that an increase in the amount of training data may lead to better results for a larger SOM which can represent the acoustic space in finer detail.

Higher-dimensional SOMs, of which results were only given for a 7x7x7 SOM, give consistently worse performance, despite a roughly similar number of units as a 20x20 SOM. Although 2-dimensional SOMS are usually used to visualize the acoustic space, there is no intrinsic reason, as discussed in Section 12.6, why this representation should be optimal to represent speech. In fact, it is highly unlikely that this is the case. The fact that a 3-*d* 7x7x7 SOM does not lead to better speaker recognition performance than a similar-sized 2-dimensional SOM does not reflect the intrinsic dimensionality of speech data, but may be related to the rough categorization of the data into three dimensions. Here, too, more data, allowing for an increase in the number of units in the SOM, may lead to better results for a 3-dimensional SOM. For comparison, it would be interesting to reduce the feature vector size by PCA to evaluate the optimum length of the vector.

GMM modelling, as used in the experiment presented in this paper, does not reflect the concept of a trajectory in the modelling, since a GMM consists of only one state and therefore does not reflect time information. Nevertheless, some time information is present in the input features to GMM, since the Gaussian mixtures model the voice signatures, which consist of x-y coordinates with additional velocity and acceleration parameters as well as angle and curvature information at each point. Such vectors gave better results than using the x-y coordinates alone, so that we can conclude that time information in the SVS trajectories is useful. Despite large jumps in the SOM space, as could be observed in Figure 12-3, the trajectories are smooth where there are smooth changes in the acoustic space. This was demonstrated by Figure 12-4. This supports the usefulness of considering the voice signature as a trajectory.

On the basis of these results, it may be possible to improve the results further by using hidden Markov modelling (HMM) instead of GMM. Standard left-to-right HMMs can explicitly model time information available in the trajectories by the transition probabilities between states. The use of HMM may therefore further improve speaker recognition, although it was shown in Morris et al. (2004) that GMM can have equal performance to HMM.

The current SOM was trained using all the speakers (61) in the CLSU database. This procedure was therefore regarded as speaker-dependent SOM training. However, a speaker-independent training strategy if used is also expected to enable to capture the general representation of the whole model space with training on a separate subset of training speakers. In other words, SOM is trained using only the speakers in the training set. It is then used to transform all the samples of the speakers in the test set. The transformation learnt by the SOM on the training speakers should be also useful for obtaining the complementary features for the test speakers. This can be verified in future work.

The method presented here was used to generate a complementary signal representation to standard MFCCs. By using complementary information, we attempt to counteract the effects of the small amount of data available for modelling. The GMM models used, however, still generalize across the data, as would HMMs. Whether generalisation is optimal across small amounts of data is questionable. It is possible that an exemplar-based approach, in which each trajectory of a speaker's voice signature is compared with a test signal, leads to better results. Such a comparison can be made using dynamic programming techniques.

Besides these results being positive, the SOM trajectory should provide a representation which is complementary to most other speech parameterisations and can therefore be expected to continue to provide a positive contribution, in both speech or speaker recognition, when used in combination with most other speech representations.

12.11 Summary

In this chapter, a novel approach to speaker recognition by using the speaker voice signature (SVS) was investigated. The SVS is a vector obtained by a parameterization of the x-y coordinates in a trained SOM, similar to the parameterization of an on-line signature's x-y coordinates. By combining speaker recognition scores from GMM of STCs with those for GMM using MFCCs, speaker identification and verification could be improved; in some cases, early fusion by concatenating STCs with MFCCs before GMM modelling also improved speaker verification. The SVS therfore provides complementary information to MFCCs which is useful for speaker recognition. Best speaker recognition results were found for late score fusion using STCs derived from the x-y coordinates of the trajectories in a 20 x 20 SOM. This SOM gave a relative error improvement over the state-of-the-art MFCC baseline of 6% for identification and 10% for verification.

Although the improvements obtained are modest, it should be noted that the SOM has not yet been fine-tuned. Possible approaches to improve the speaker recognition results using the SVS were also discussed.

13. Discussion and future work

13.1 Introduction

After the discussion of the approaches proposed in the previous chapters, this chapter discusses the approaches in a wider context and suggests a number of possible research issues which can be explored in future work.

In Chapter 13 the details of MLP-based feature enhancement as used for speaker recognition were described, including the algorithms of MLP training and speaker basis selection. These algorithms were tested in conditions with varying types and levels of noise and the telephone speech. These experiments have shown the robustness of this approach. In the present chapter, we will address a number of questions related to this approach which have been left unaddressed hitherto. This discussion will give the reader a clearer understanding of MLP-based feature enhancement.

In Section 13.2, we shall first focus on a general geometrical interpretation of MLP-based feature enhancement. To begin with, an overall effect of discrimination enhancement is presented. Then, based on this principle, the speaker basis selection approach described above is interpreted from the perspective of average between-class variance. This will show the basic reason why this algorithm of speaker basis selection works and why it is crucial for speaker feature enhancement. Following that, an alternative method based on convex hull selection is proposed for speaker basis selection.

We shall also suggest how the MLP-based feature enhancement methods developed for speaker identification can possibly be used for speaker verification in Section 13.3.

In Section 13.4, the possibility of using symmetric KL distance for other applications as well as for cohort speaker selection in speaker verification is considered. The symmetric KL distance is suggested to be used as a measure to calculate the distance between two statistical distributions, which is the basis of cohort speaker selection in speaker verification.

Section 13.5 discusses the use of complementary speech features acquired by SOM processing. The most essential aspects of this approach are addressed and clarified. Other possible alternative approaches to feature generation are also proposed.

Finally, the fusion of different types of biometrics is discussed. As this dissertation was developed in the framework of SecurePhone project, the human voice was taken as one type of biometric feature for user authentication. As mentioned in the introductory chapter, multi-modalities are always helpful to enhance system performance. A number of issues related to multi-modalities are therefore discussed in Section 13.6. This chapter finishes with a summary in Section 13.7.

13.2 A geometrical interpretation of MLP feature enhancement

The details of the procedures followed in MLP-based feature enhancement were addressed in Chapter 11. These procedures were also tested under different conditions, including low-bandwidth clean speech (TIMIT-8k), telephone speech (NTIMIT) and noisy speech with a variety of types and levels of additive noise (TIMIT-8k+Noisex). All these experiments showed the effectiveness of the proposed approach. However, a geometrical interpretation of this approach, which may help to understand its essence more deeply, is still missing. On the basis of this interpretation, another method for speaker basis selection will be suggested.

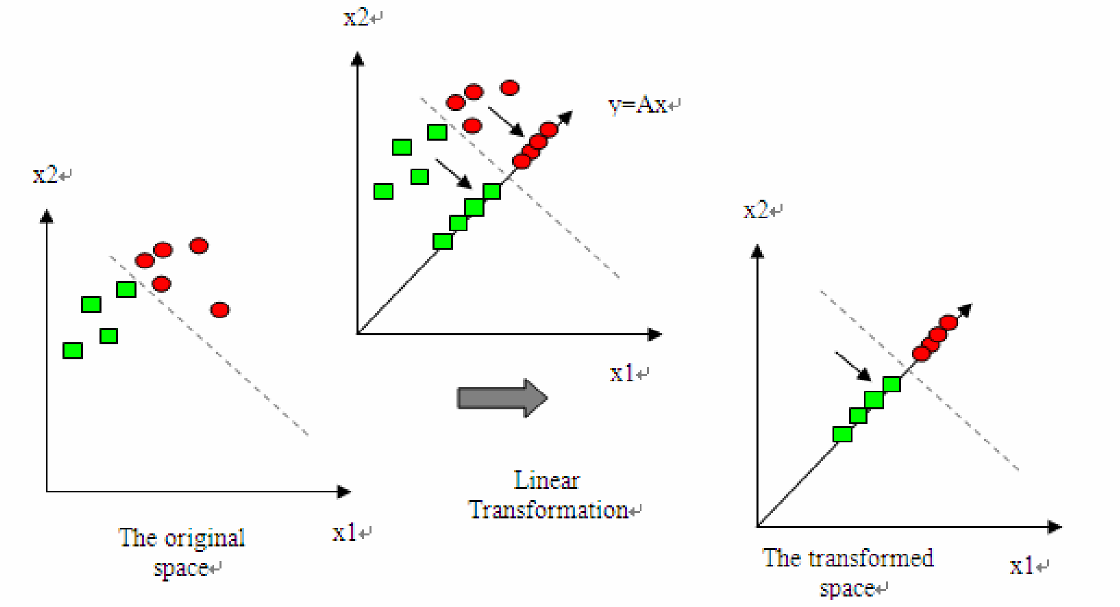


Figure 13-1: Demonstration of the projection of LDA and a linear MLP (adapted partly from Campbell et al. 1997)

First of all, it is well known that the function of LDA (a linear MLP) is to simultaneously maximise between-class variance and minimise within-class variance. The effect of these two optimisations on an original feature space is a linear discriminative transformation, based on which clusters in the projected feature space are more easily distinguished. Therefore, the features are enhanced (Figure 13-1). As shown in Figure 13-1, the original features are linearly projected onto a hyperplane (a line in a 2-d space). Hence, the projected features have a smaller intra-class variance (the data within a class have been squashed) and a larger between-class variance (the between-class distance has been stretched).

While in LDA where the projection is onto a hyperplane (or a line), the projection direction may be nonlinear in NLDA. As shown in Figure 13-2, the class samples are projected onto a curve illustrated in a 2-*d* space. During this projection, the squashing and stretching effects are carried out more successfully than in LDA. Hence, to borrow the language of LDA, it may be said that NLDA is a transformation which maximises the between-class variance and minimises the within-class variance in a nonlinear way.

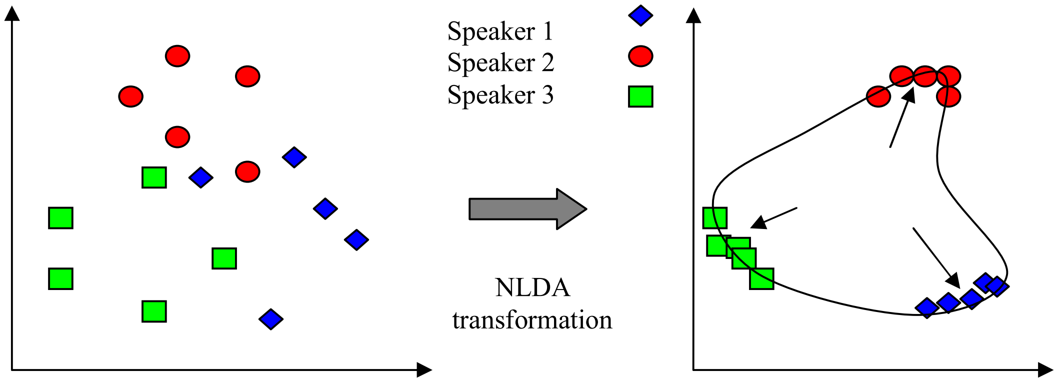


Figure 13-2: Demonstration of the projection of NLDA

Therefore, we can clearly draw the conclusion that a feature space transformed by NLDA implemented by an MLP is an enhanced and more discriminative space because it holds a larger between-class variance and a smaller within-class variance. *The overall effect of NLDA is as if an original feature space were stretched toward the outside, this making the classes in it more separable, simultaneously squashing data samples within each class.*

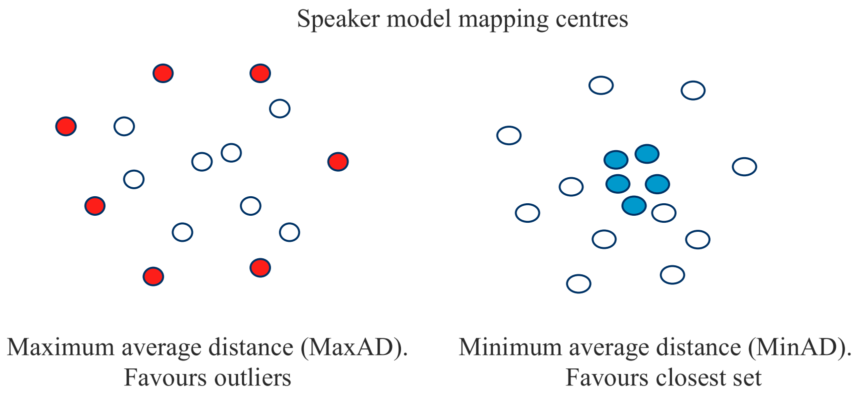

Speaker model mapping centres
Maximum average distance (MaxAD).
Favours outliers
Minimum average distance (MinAD).
Favours closest set

Figure 13-3: Demonstration of speaker basis selection using MaxAD vs. MinAD selection

Based on the above geometrical interpretation, it is easy to understand that given a subset of classes used for MLP training to enhance class discrimination, the best selection of these classes may be expected to be the ones which are close to or on the boundary of the feature space. In Chapter 13, an automatic approach favouring boundary points by selecting speakers with maximal between-class variance (14.8) was used to find these classes. It can be seen (Figure 13-3) that the projection of the closest speaker set is not helpful for the separation of the other speaker models, whereas the projection of the boundary speakers may be expected to also benefit the separation of the others. This is the essence of basis speaker selection.

Extending this idea further, it can be predicted that selecting the classes on the convex hull may lead to a better class separation. The convex hull of a set of points (speaker models) is the smallest convex set that includes the points. For a two dimensional finite set the convex hull is a convex polygon (red points in Figure 13-4). The convex hull of a set of points is the minimum set of points which are closest to the boundary of a feature space. Based on the previous two arguments, if a transformation is trained to stretch the speaker points on the convex hull towards the outside of the feature space, it may as well benefit the separation of other points in the feature space in the most efficient way. Following the same idea as in the case of MaxAD to seek a speaker basis, the convex hull approach may find the minimum set of speakers, which for training may achieve the same performance as using more speakers. Therefore, this approach would be the most economical and efficient for MLP training.

In order to find the speakers on the convex hull of a given set of points (i.e. the training speakers), such an algorithm can be used as follows:

[Algorithm of seeking the convex hull speakers for speaker basis selection]

(1) The symmetric KL distance is first obtained for any pair of points using (11.6). When the distance between any two points is determined, the geometric positions of these points in a 2-d plane are then determined according to the method described in (2).

(2) Position all points in a 2-d plane according to the distance between any pair as follows:

   a) Pick up the first point *x1* randomly and assign it as the origin (*0,0*).

   b) Take any one *x2* from the rest of (*n-1*) points, assigning it as (*d,0*), whereby *d* is the distance between *x1* and *x2*.

   c) Take another point *x3* from the rest of (*n-2*) points, positioning it by *d1* and *d2*, where *d1* is the distance between *x1* and *x3*, *d2* the distance between *x2* and *x3*.

   d) Repeat c) until no points are left.

(3) Finding the convex hull of a given point set, using a well-established algorithm like Thomus et al. (2001) or Preparata et al. (1977).

This issue will need to be investigated in future work.

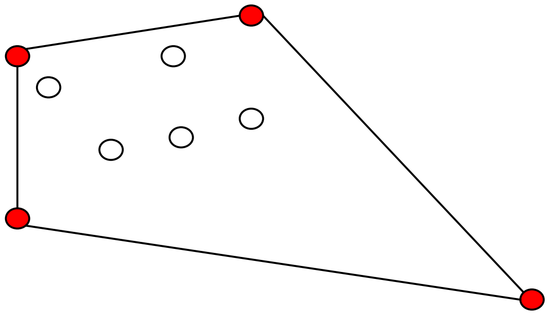


Figure 13-4: The convex hull of a set of points

13.3 Application of feature enhancement to speaker verification

Besides for identification, the approach presented in Chapter 11 can also be applied to verification task if a global MLP is trained on basis speakers to transform all the speakers' feature samples as used in identification tasks (Figure 13-5, top). However, as the purpose of a verification task is to distinguish only two candidate classes (the claimed speaker and his impostors), rather than the whole population, a speaker-dependent variant scheme for speaker verification may be expected to be more efficient. In this approach, a separate MLP with only two outputs trained on each speaker (or class) and its impostors is used for each speaker (or class) (Figure 13-5, bottom).

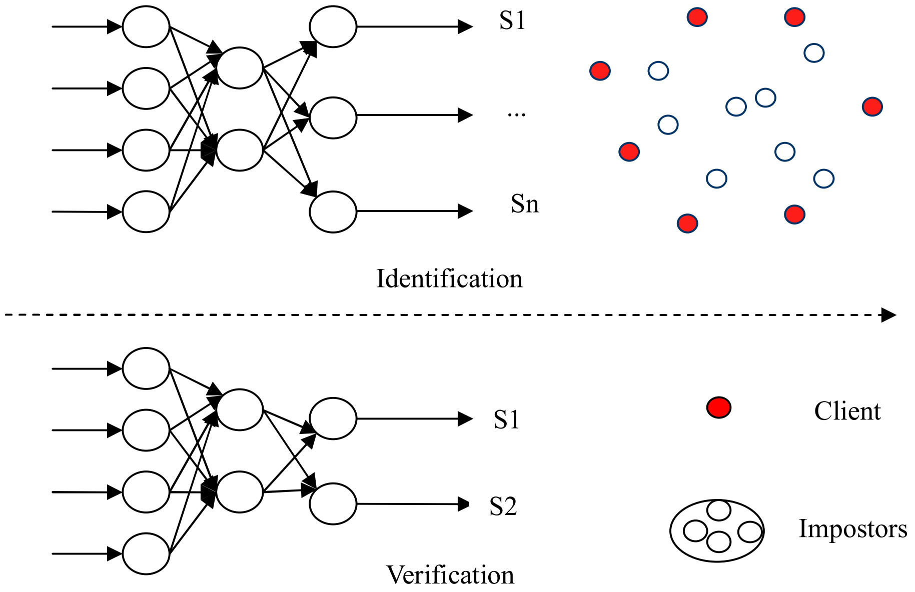


Figure 13-5: MLP-based feature enhancement in speaker identification and verification

Nevertheless, the separation of all the classes may still help to separate two classes as well. Thus, the former (the separation of all the classes) may be a sufficient condition for the latter (the separation of two classes). This further idea must be clarified by future work.

Of course, this approach is not limited to speaker recognition tasks. It may also be expected to be valid for any other pattern recognition task.

13.4 Other applications of the symmetric KL distance

In Section 11.5.2.3, the symmetric KL distance (11.6) was proposed as a method of measuring the distance between two pdfs. Although this distance measure was proposed to solve the problem of speaker basis selection in speaker identification, it is able to be generally used in other cases than speaker basis selection, where the distance measure between two pdfs is required, such as in other classification applications and in speaker verification.

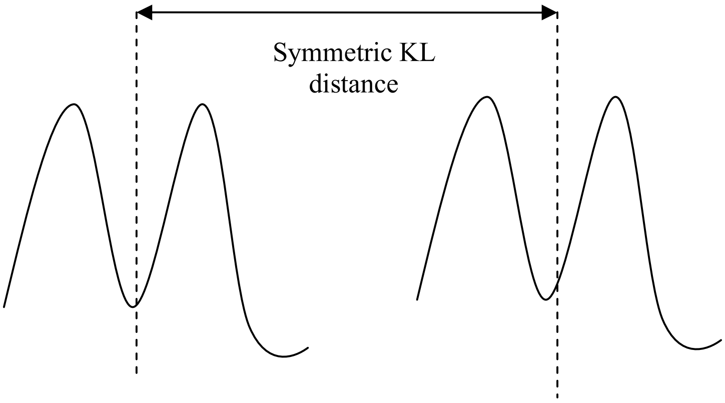


Figure 13-6: The symmetric KL distance between two pdfs

In the case of other classification applications (e.g. face recognition and object recognition), the symmetric KL distance can be used to evaluate the distance between two class (face or object) distributions (Figure 13-6).

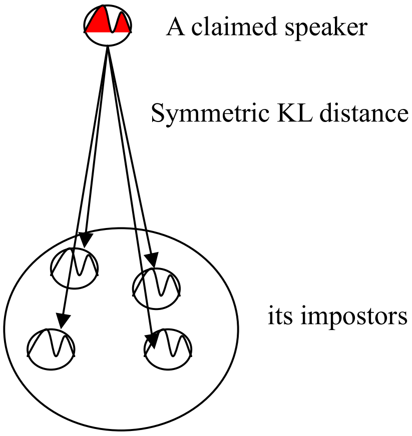


Figure 13-7: Applying the symmetric KL distance to cohort speaker selection

The symmetric KL distance can also be applied to cohort speaker selection in speaker verification. As mentioned in Section 7.4, cohort speaker selection is often used to choose impostors for background model training in speaker verification (Figure 13-7). The distances (7.14) and (7.15) which are currently used to measure how far two pdfs are separated from each other only take their likelihood into consideration, but not the entropy (Duda et al. 2001), whereas the symmetric KL distance is derived from the entropy. It can therefore more accurately capture the distance between two stochastic distributions. As a result, the symmetric KL distance is likely to be a more appropriate distance measure for selecting cohort speakers. This may be verified by future work.

13.5 The selection and complementarity of speech features

In Chapter 12, complementary features obtained by SOM processing were discussed. In this section a number of additional points relating to this method are discussed. The complementary features generated by SOM processing can also be considered as an outcome of a nonlinear transformation. However, in contrast to LDA and NLDA, this transformation does not have a discriminative objective. Instead, it preserves the global shape of a feature space in a 2-dimensional representation. Therefore, SOM-based features have much less discriminating power than MLP-based features – a fact which is supported by the results of experiments using SOM-based features alone, in which the performance was only 56% speaker identification accuracy, which was by far inferior to that of NLDA-based features.

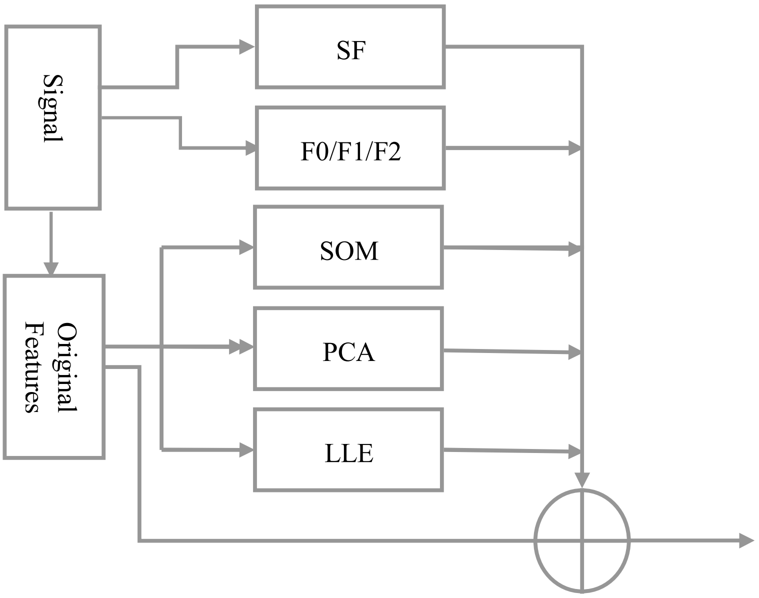


Figure 13-8: Original features fused with source features (SF) and SOM/PCA/LLE transformed features are used for speaker recognition

SOM processing is in fact an approach of dimension reduction except that it preserves the topological structure of a feature space. From this point of view, locally linear embedding (LLE) may also have the same function (Saul et al. 2000). Moreover, the features generated using principal components analysis (PCA) may be more powerful than SOM-based features, since they can hold more dimensional components without exploding the complexity of a system as in the case of SOM. All these points are worth investigating in future work (Figure 13-8).

Despite the low performance of SOM-based features on their own, they do contain some useful information pertaining to speaker characteristics which is complementary to the information contained in MFCCs. The combination of SOM-based features with MFCC features leads to an improvement of system accuracy. Extending this idea, it would be interesting to fuse other types of helpful information to further improve system performance. These are too numerous to be discussed in detail here. For instance, the first formant (F1) and the second formant (F2) can be fused with the MFCC features to improve speaker recognition systems. Moreover, source features such as glottal flow parameters can also be combined with MFCC features. These issues can be investigated in future research.

13.6 Fusion strategies

Fusion systems always work better than systems using a single feature-type. This has been proved by multi-modal authentication systems in which features are usually even more complementary. Within the context of the SecurePhone project, fusing voice features with face and signature features at the score level achieved promising results. A GMM-based fusion scheme at the score level was proposed to give the best performance in Koreman et al. (2006). This scheme trains a GMM to model the distribution of the joint score vector, each of whose components is derived from the match score obtained based on each modality (Figure 13-9, top). This GMM-based fusion is an unsupervised learning approach, since it does not make use of any prior knowelege such as the class label of each data sample. An MLP-based fusion strategy at the score level used for speaker verification might therefore be expected to achieve better fusion performance on the basis of supervised discriminative training, where the fused score is output from the single-unit hidden layer (Figure 13-9, bottom). It would be interesting to compare GMM-based fusion with MLP-based fusion in future work.

Although the MLP-based fusion at the score level is illustrated for speaker verification in Figure 13-9, this fusion can also be applied at the feature level and to identification tasks. The MLP-based fusion training scheme used in identification is similiar to that shown in (Figure 13-9, bottom), except there will be multiple outputs in the output layer.

It is worth mentioning that although the multi-modal approach is more effective, it cannot entirely replace the uni-modal approach: Firstly, it must be based on the techniques used in the uni-modal approach, and secondly, its complexity is much higher than the uni-modal approach.

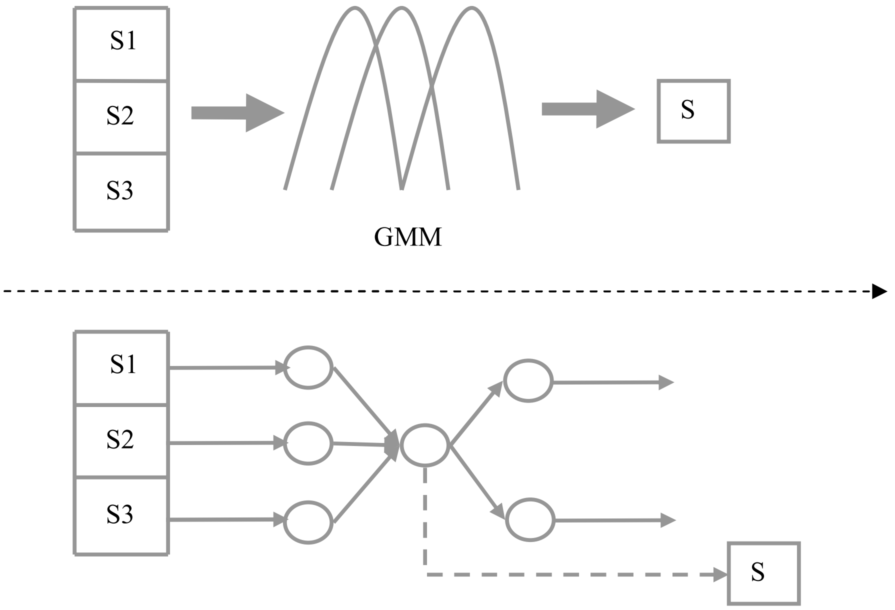


Figure 13-9: GMM-based and MLP-based fusion strategies at the matching score level for speaker verification (S1: face score, S2: voice score, S3: signature score)

13.7 Summary

In this chapter, we discussed several issues associated with MLP-based and SOM-based feature enhancement approaches. In particular, a geometrical interpretation to MLP-based feature enhancement was described to elucidate the essence of this approach. A possible extension to speaker basis selection was proposed based on convex hull section. Moreover, the possibility of applying feature enhancement to speaker verification was also discussed. Following that, other applications of the symmetric KL distance were addressed. Finally, in the context of the SecurePhone project, the selection and complementarity of speech features and a number of fusion strategies were discussed for future work to improve the performance and security of user authentication systems.

14. Conclusion

The main goal of this dissertation was to investigate speaker recognition from the perspective of using discriminative features to improve the performance and robustness of biometric recognition systems. A large amount of effort is dedicated to this question. In Part I, a general background of speech and other biometrics for human identity recognition was outlined. In Part II, state of the art speaker recognition techniques were summarised. Following that, experimental techniques were proposed and discussed in Part III. Several important achievements are achieved, among which the following:

First, we analysed a realistic feature space (mel-scaled cepstral space), and proposed using the speaker-phoneme distribution, i.e. a feature space is organised around phonemes rather than speakers, to support the motivative of applying feature enhancement for speaker recognition, since this distribution provides an impediment for speaker recognition.

Second, based on the analysis work, feature enhancement approaches were systematically investigated. In particular, linear discriminant analysis and several nonlinear discriminant approaches implemented by MLPs were compared and analysed. It was found that a 3-hidden-layer MLP, by which a nonlinear transformation is carried out, outperformed linear transformations implemented by LDA and a linear MLP.

Third, a generalised framework for acquiring discriminative features for speaker recognition was proposed. Although discriminative features used for speaker recognition were not first proposed by us, it was found that the number of speakers used for the MLP learning is a highly crucial factor. In fact, a sufficiently large number of speakers is a very important factor for an MLP to learn the discriminating information used in feature transformation for speaker recognition. If the number of speakers is not sufficient to cover all the acoustic characteristics of a feature space, the trained MLP is not efficient enough to optimally and discriminatively project an original feature space into another space.

Four, based on the third finding, a further concept, i.e. speaker basis, was proposed to optimise the selection of speakers for MLP-based feature enhancement, given a fixed number of speakers in a group. It was found that a number of speakers are not all of the same importance and only the most important ones (speaker basis) should be selected. This approach has two advantages. The first is that it can substantially improve system performance (cf. the experimental part of Chapter 11) and the second is that it can significantly reduce the complexity of the training of MLP.

Five, the proposed approach to feature enhancement was anticipated to have wide range of applications in non-speech related fields. In principle, it should be possible to apply it to any other pattern recognition application.

Finally, a complementary feature type was also proposed, derived from the mel-scaled cepstral features by using SOM processing. This type of features was captured from speaker voice signatures generated by SOM by means of a feature extraction approach analogous to signature recognition. Speaker voice signatures were found to contain a lot of distinguishing information complementary to that contained in mel-scaled cepstral features. A linear fusion strategy at the score level was found to improve the performance of speaker recognition (both identification and verification).

References

Allano, L., Garcia-Salicetti, S., et al. (2006). "Non intrusive multi-biometrics on a mobile device: a comparison of fusion techniques." *Proc. SPIE conference on Biometric Techniques for Human Identification III.*

Andrew, T. B. J. and David, N. C. L. (2003). "Integrated Wavelet and Fourier-Mellin Invariant Feature in Fingerprint Verification System." *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*: 82-88.

Atal, B. (1976). "Automatic recognition of speakers from their voices." *Proc. IEEE* **64**: 460-475.

Argente, J. (1991). "From speech to speaking styles." *Proc. Phonetics and Phonology of Speaking Styles: Reduction and Elaboration in Speech Communication*, Barcelona, Catalonia, Spain, 1991.

Auckenthaler, R., Carey, M., et al. (2000). "Score normalization for text-independent speaker verification systems." *Digital Signal Processing* **10**: 42-54.

Barron, J. L., Fleet, D. J., et al. (1994). "Performance of optical flow techniques." *International Journal of Computer Vision* **12**: 43-77.

Bartlett, M., Movellan, J., et al. (2002). "Face recognition by independent component analysis." *IEEE Trans. Neural Netw.* **13**(6): 1450-1464.

Bartlett, M. and Sejnowski, T. (1997). "Independent components of face images: a representation for face recognition." *Proc. 4th Annual J. Symp. Neural Computation.*

Bauer, E. and Wirtz, B. (1995). "Parameter reduction and Personalized Parameter Selection for Automatic Signature Verification." *Proc. Third Int'l Conf. Document Analysis and Recognition*: 183-186.

Baumberg, A. M. and Hogg, D. C. (1995). "Learning spatiotemporal models from training examples." *British Machine Vision Conference.*

Bengio, S., Bimbot, F., et al. (2002). Expermental protocol on the BANCA database.

Bio-tech-inc (2002). "Biometric Technical Assessment." *http://www.bio-tech-inc.com/ Bio_Tech_Assessment.html.*

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*, Oxford university press.

Blackwell, D. A. and Girshick, M. A. (1979). *Theory of Games and Statistical Decisions*, Dover Publications.

Boles, W. and Boashash, B. (1998). "A human identification technique using images of the iris and wavelet transform." *IEEE Trans. Signal Proc.* **4**: 1185-1188.

Bourlard, H. and Morgan, N. (1994). *Connectionist speech recognition – a hybrid approach*, Kluwer.

Boyd, J. E. and Little, J. J. (2005). "Biometric Gait Recognition." *Biometrics School 2003, LNCS 3161*: 19-42.

Brand, J.D., Mason, J.S. and Colomb, S. (2001). "Visual speech: a physiological or behavioural biometric? " *Lecture Notes in Computer Science.*

Bredin H., Miguel A., Witten I. and Chollet G. (2005). "Detection replay attacks in audiovisual identity verification." *Proc. ICASSP'05.*

Campbell, J. P. (1997). "Speaker recognition: a tutorial." *Proc. IEEE* **85**(9): pp. 1437-1462.

Campbell, J. P. and D.A.Reynolds (1999). "Corpora for the evalation of speaker recognition systems." *Proc. ICASSP'99.*

Cardinaux, F., Sanderson, C., et al. (2003). Comparison of MLP and GMM classifiers for face verification on XM2VTS.

Chai, D. and Ngan, K. N. (1999). "Face segmentation using skin-color map in videophone applications." *IEEE Trans. Circ. Syst. Video Technol.* **9**(4): 551–564.

Chang, K. I., Bowyer, K., et al. (2004). "Multi-biometrics Using Facial Appearance, Shape and Temperature." *IEEE Proc. AFER'04.*

Chen, H. and Jain, A. K. (2005). "Dental Biometrics: Alignment and Matching of Dental Radiographs." *IEEE Trans. PAMI* **27**(8): 319-1326.

Cohen, S. and Intrator, N. (2002). "A hybrid projection based and radial basis function architecture: Initial values and global optimisation." *Pattern Anal. Appl. (Special issue on Fusion of Multiple Classifiers)* **5**(2): 113-120.

Cole, R., Noel, M., et al. (1998). "The CSLU speaker recognition corpus." *Proc. ICSLP'98*.

Collobert, R., Bengio, S. & Mariéthoz, J. (2002). Torch: a modular machine learning software library.

Commission, E. (2005). Biometrics at the Frontiers: Assessing the impact on Society.

Coplan, R. J., Coleman, B., et al. (1998). "Shyness and little boy blue: iris pigmentation, gender, and social wariness in preschoolers." *Developmental Psychobiology* **32**: 37-44.

Cormen, T., Leiserson, C., et al. (2001). *Introduction to Algorithms*, Second Edition. MIT Press and McGraw-Hill, Section 33.3: Finding the convex hull, pp.947-957.

Cortes, C. and Vapnik, V. N. (1995). "Support vector networks." *Machine Learning* **20**: 273–297.

Daugman, J. (2003). "Demodulation by complex-valued wavelets for stochastic pattern recognition." *Int'l Journal of Wavelets, Multi-resolution and Information Processing* **1**(1): pp 1-17.

Daugman, J. G. (1993). "High confidence visual recognition of person by a test of statistical independence." *Trans. PAMI* **15**: 1148-1161.

Daugman, J. G. (2004). "How iris recognition works." *IEEE Trans. Circuits and Syst. for Video Tech.* **14**(1): 21-30.

Davis, J. W. and Bobick, A. F. (1997). "The representation and recognition of human movement using temporal templates." *IEEE Computer Vision and Pattern Recognition*: 928-934.

Delsecur (2000). "Fraud & Industry." *http://www.delsecur.com/html/body_fraud.html*.

Demidov, V. V. and Broude, N. E. (2004). *DNA Amplification: Current Technologies and Applications*, Boston University, USA.

Duda, R. O., Hart, P. E., et al. (2001). *Pattern classification*, Wiley.

Dupont, S. and Luettin, J. (2000). "Audio-Visual Speech Modeling for Continuous Speech Recognition." *IEEE Trans. on Multimedia* **2**(3).

Duta, N., Jain, A. K., et al. (2001). "Matching of Palmprints." *Pattern Recognition Letters* **23**(4): 477-485.

Ellis, D. and Reyes-Gomez, M. (2001). "Investigations into Tandem acoustic modeling for the Aurora task." *Proc. Eurospeech 2001*: 189-192.

Falthhauser, R. and Ruske, G. (2001). "Improving speaker recognition preformance using phonetically structured gaussian mixture models." *Proc. Eurospeech' 01*.

Faúndez-Zanuy, M. and Rodríguez-Porcheron, D. (1998). "Speaker recognition using residual signal of linear and nonlinear prediction models." *Proc. ICSLP'98*.

Fisher, W. M., Doddingtion, G. R., et al. (1986). "The DARPA speech recognition research database: Specifications and status." *Proc. DARPA Workshop on Speech Recognition, February 1986*: 93-99.

Fontaine, V., Ris, C., et al. (1997). "Nonlinear Discriminant Analysis for improved speech recognition." *Proc. Eurospeech'97*: 2071-2074.

Furui, S. (1981). "Cepstral analysis technique for automatic speaker verification." *IEEE Trans. on Speech and Audio Processing* **29**: 254-272.

Furui, S. (1997). "Recent advances in speaker recognition." *Patern Recognition Letters* **18**: 858-972.

Gao, Y. and Leung, M. K. H. (2002). "Face recognition using line edge map." *IEEE Trans. Patt. Anal. Mach. Intell.* **24**(6): 764-779.

Garcia, C. and Tziritas, G. (1999). "Face detection using quantized skin color regions merging and wavelet packet analysis." *IEEE Trans. Multimedia* **1**(3): 264-277.

Garicia-Salicetti, S., Mellakh, A., Allono, L. and Dorizzi, B. (2005). "Multimodal biometric score fusion: the mean rule vs. support vector classifiers." *Proc. EUSIPCO'05*, Antalya, Turkey.

Garofolo, J. S., Lamel, L. F., et al. (1993). "TIMIT Acoustic-Phonetic Continuous Speech Corpus." *http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S1*.

Genoud, D., Ellis, D., et al. (1999). "Combined speech and speaker recognition with speaker-adapted connectionist models." *Proc. ASRU*.

George, D. and Mallery, P. (2002). *SPSS for Windows Step by Step: A Simple Guide and Reference, 4th Ed.*, Allyn & Bacon.

GlobalSecurity (2001). "Biometrics." *http://www.globalsecurity.org/security /systems/biometrics.htm*.

Godil, A., Grother, P., et al. (2003). "Human Identification from Body Shape." *Fourth International Conference on 3-D Digital Imaging and Modeling (3DIM '03)*: 386-393.

Gold, B. and Morgan, N. (2000). *Speech and audio signal processing: processing and perception of speech and music*, Wiley.

Golfarelli, M., Maio, D., et al. (1997). "On the Error-Reject Trade-Off in Biometric Verification Systems." *IEEE Trans. Pattern Analysis and Machine Intelligence* **19**(7): 786-796.

Golfarelli, M., Maio, D., et al. (2000). "On the error-reject tradeoff in biometric verification systems." *IEEE Trans. on Patt. Anal. and Mach. Intell.* **19**: pp. 786-796.

H. Hermansky et al. (1992). "RASTA-PLP speech analysis technique." *Proc. ICASSP'92*: 121-124.

Hagen, A. and Morris, A. C. (2003). "Recent advances in the multi-stream HMM/ANN hybrid approach to noise robust ASR." *Computer, Speech and Language* **19**(1): 3-30.

Hamilton, D. J., Whelan, J., et al. (1995). "Low Cost Dynamic Signature Verification System." *IEEE Conf. Publications* **408**: 202-206.

Hamilton, J., Wliclan, J., et al. (1995). "Low Cost Dynamic Signature Verification Systcm." *IEEE Conf. Publications* **408**: 202-206.

Han, C. C., Cheng, H. L., et al. (2003). "Personal Authentication Using Palmprint Features." *Patern Recognition* **36**(2): 371-381.

Handbook, I. (1995). *A Guide to the Use of the International Phonetic Alphabet*, Cambridge University Press.

Hashiyada, M. (2004). "Development of Biometric DNA Ink for Authentication Security." *Tohoku J. Exp. Med.* **204**: 109-117.

Hashiyada, M., Itakura, Y., et al. (2003). "Polymorphis of 17 STRs by multiplex analysis in Japanese polulation." *Forensic Sci. Int.* **133**: 250-253.

Heck, L., Konig, Y., et al. (2000). "Robustness to telephone handset distortion in speaker recognition by discriminative feature design." *Speech Communication* **31**: 181-192.

Hermansky, H., Ellis, D., et al. (2000). "Tandem connectionist feature stream extraction for conventional HMM systems." *Proc. ICASSP'00.*

Hermansky, H. and Malayath, N. (1998). "Speaker verification using speaker-specific mappings." *Proc. of Speaker Recognition and its Commercial and Forensic Applicaitons.*

Hsu, R. L., Abdel-Mottaleb, M., et al. (2002). "Face detection in color images." *IEEE Trans. Patt. Anal. Mach. Intell.* **25**(4): 696-706.

http://en.wikipedia.org/ Likelihood ratio test.

Huang, K. and Yan, H. (1995). "On-Line Signature Verification Based on Dynamic Segmentation and Global and Local Matching." *Optical Eng.* **34**(12): 3,480-3,488.

Huang, W., Sun, Q., et al. (1998). "A robust approach to face and eyes detection from images with cluttered background." *Proc. Int. Conf. Patt. Recogn.* **1**: 110-114.

Hultzen, L. S., Allen, J. H. D., Jr., et al. (1964). *Tables of Transitional Frequencies of English Phonemes*, University of Illinois Press, Urbana.

Indovina, M., Uludag, U., et al. (2003). "Multimodal Biometric Authentication Methods: A COTS Approach." *Proc. MMUA'03.*

IPA *http://www2.arts.gla.ac.uk/IPA/vowels.html.*

ITU-recommendation-P.56 (March 1993). "Objective measurement of active speech level."

Jain, A., Hong, L., et al. (1997). "On-line identity-authentication system using fingerprints." *Proceedings of IEEE (Special Issue on Automated Biometrics)* **85**: 1365-1388.

Jain, A. K. and Chen, H. (2004). "Matching of Dental X-ray Images for Human Identification." *Pattern Recognition* **37**(7): 1519-1532.

Jain, A. K., Nandakumar, K., et al. (2005). "Score Normalization in Multimodal Biometric Systems." *Patern Recognition* **38**(12): 2270-2285.

Jain, A. K. and Pankanti, S. (2000). "Fingerprint Classification and Recognition." *The Image and Video Processing Handbook.*

Jain, A. K., Ross, A., et al. (1999). "A Prototype Hand Geometry-Based Verification System." *Proc. Second Int'l Conf. Audio- and Video-Based Biometric Person Authentication*: 166-171.

Jang, G.-J., Lee, T.-W., et al. (2001). "Learning statistically efficient features for speaker recognition." *Proc. ICASSP'01.*

Jin, Q. and Waibel, A. (2000). "Application of LDA to speaker recognition." *Proc. ICSLP'00.*

Kajarekar, S., Yegnanarayana, B., et al. (2001). "A study of two dimensional linear discriminats for ASR." *Proc. ICASSP'01.*

Kalman, R. E. (1960). "A New Approach to Linear Filtering and Prediction Problem."

Trans. Of the ASME: 35-45.

Kashi, K. S., Turin, W., et al. (1996). "On-Line Handwritten Signature Verification Using Stroke Direction Coding." *Optical Eng.* **35**(9): 2,526-2,533.

Kawagoe, M. and Tojo, A. (1984). "Fingerprint Pattern Classification." *Pattern Recognition* **17**(3): 295-303.

Kenney, J. F. and Keeping, E. S. (1951). *Mathematics of Statistics, Pt. 2, 2nd ed.* Princeton, NJ, Van Nostrand.

Kim, H.-G., Berdahl, E., et al. (2003). *Speaker Recognition Using MPEG-7 Descriptors*. Eurospeech'03, Geneva, Switzerland.

Kim, H.-G., Haller, M., et al. (2004b). *Comparison of MPEG-7 Basis Projection Features and MFCC applied to Robust Speaker Recognition*. ISCA – A Speaker Odyssey, Toledo, Spain.

Kim, H.-G. and Sikora, T. (2004a). "Comparison of MPEG-7 Audio Spectrum Projection Features and MFCC applied to Speaker Recognition, Sound Classification and Audio Segmentation." *Proc. ICASSP'04*.

Kim, S. H., Park, M. S., et al. (1995). "Applying Personalized Weighls to a Feature Set for On-Line Signature Verification." *Proc. Third Int'l Conf. Document Analysis and Recognition*: 882-885.

Kinnunen, T., Karpov, E., et al. (2004). "Efficient Online Cohort Selection Method for Speaker Verification." *ICSLP'04*.

Kirby, M. and Sirovich, L. (1990). "Application of the Karhunen–Loeve procedure for the characterization of human faces." *IEEE Trans. Patt. Anal. Mach. Intell.* **12**(1): 103–108.

Kohonen, T. (1988). "The 'neural' phonetic typewriter." *Computer* **12**(3): 11-12.

Kohonen, T. (1997). *Self-organizing Maps (second edition)*. Berlin, Springer Verlag.

Kohonen, T., Hynninen, J., et al. (1996). "SOM PAK: The Self-Organizing Map Program Package."

Kong, S. G., Heo, J., et al. (2005). "Recent advances in visual and infrared face recognition: a review." *the Journal of Computer Vision and Image Understanding* **97**(1): 103-135.

Konig, Y., Heck, L., et al. (1998). "Nonlinear discriminant feature extraction for robust text-independent speaker recognition." *Proc. RLA2C, ESCA workshop on Speaker*

Recognition and its Commercial and Forensic Applications: 72-75.

Koreman, J. and Morris, A. C., et al. (2006b). "Applicability and Complementarity of Source Authentication Techniques." *Technical Report for the SecurePhone Project*.

Koreman, J., Morris, A. C., et al. (2006a). "Non-intrusive multi-modal biometric authentication on the SecurePhone PDA." *MMUA'06*.

Korotkaya, Z. (2004). "Biometric Person Authentication: Odor." *http://www.it.lut.fi /kurssit/03-04/010970000/seminars/Korotkaya.pdf*.

Kuchera, H. and Francis, W. N. (1967). *Computational Analysis of Present-Day American English*, Brown University Press, Providence, Rhode Island.

Kumar, A., Wong, D. C. M., et al. (2003). "Personal Verification Using Palmprint and Hand Geometry Biometric." *Proc. of 4th Int'l Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA)*: 668-678.

Kuncheva, L. I., Whitaker, C. J., et al. (2000). "Is independence good for combining classifiers?" *Proc. of Int'l Conf. on Pattern Recognition (ICPR)* **2**: pp. 168-171.

Kwon, O.-W. and Lee, T.-W. (2004). "Phoneme recognition using ICA-based feature extraction and transformation." *Signal Processing* **84**: 1005-10019.

Lam, K. M. and Yan, H. (1996). "Locating and extracting the eye in human face images." *Patern Recognition* **29**(5): 771-779.

Lapedes, A. a. R. F. (1988). "How neural nets work." *Neural Information Processing Systems*: 442-456.

Laver, J. and Trudgill, P. (1979). "Phonetic and linguistic markers in speech." *Social markers in Speech*, Scherer, K. R. and Giles, H. (eds), Cambridge University Press.

Laver, J. (1968). "Voice quality and indexical information." *British Journal of Disorders of Communication*, vol. 2, 43-54.

Lawrence, S., Giles, C. L., et al. (1997). "Face recognition: a convolutional neural-network approach." *IEEE Trans. Neural Netw.* **8**(1): 98-113.

Lee, I., Berger, T., et al. (1996). "Reliable On-Line Signature Verification Systems." *IEEE Trans. PAMI* **28**(6): 643-647.

Li, F. F. (2004). "Handwriting Authentication by Envelopes of Sound Signatures." *17th International Conference on Pattern Recognition (ICPR'04)* **1**: 401-404.

Lucey, S., Chen, T., et al. (2005). "Integration strategies for audio visual speech processing: applied to text dependent speaker identification/ verification." *IEEE Trans. on Multimedia* **7**(3): 496-506.

Ly-Van, B., Blouet, R., et al. (2003). "Signature with Text-Dependent and Text-Independent Speech for Robust Identity Verification." *Proc.MMUA'03*.

Ly-Van, B., Garcia-Salicetti, S., et al. (2004). "Fusion of HMM's Likelihood and Viterbi Path for On-line Signature Verification." *Biometric Authentication Workshop (BioAW), Lecture Notes in Computer Science (LNCS) 3087*: 318-331.

Mai, L., Tan, T., et al. (2003). "Personel identification based on iris texture analysis." *IEEE Trans. on Pattern Analysis and Machine Intelligence* **25**(12): 1519-1533.

Manduchi, R. (1999). "Bayesian Fusion of Color and Texture Segmentations." *Proceedings of the 7th IEEE International Conference on Computer Vision, Kerkyra*: 956-962.

Mariéthoz, J. and Bengio, S. (2002). "comparative study of adaptation methods for speaker verification." *Proc. ICSLP 2002*.

Mariéthoz, J., Lindberg, J., et al. (2000). "A MAP approach, with synchronous decoding and unit-based normalisation for test-dependent speaker verification." *Proc. ICASSP'00*.

Martens, R. and Claesen, T. (1998). "Utilizing Baum-Welch for On-line Signature Verificaiton." *Proc. Sixth International Workshop Frontiers in Handwriting Recognition*: 389-397.

Martin, A., Doddington, G. , Kamm, T. , Ordowski, M., Przybocki, M. (1997). "The Det Curve In Assessment Of Detection Task Performance " *Proc. Eurospeech'97*.

Matsuura, T. and Yamamolo, S. (1998). "Signature Verification Using Distribution of Angular Direction of Pen-point Movement." *Proc. Sixth International Workshop Frontiers in Handwriting Recognition*: 537-545.

Matsuura, T. and Yu, T. S. (1998). "On-Line Signature Verification by IIR system." *Proc. Fifth International Workshop Frontiers in Handwriting Recognition*: 413-416.

McTait, K., Bredin, H., Colon, S., Fillon, T. and Chollet, G. (2005). "Adapting a high quality audiovisual database to PDA quality." *Proc. ISPA 2005*, Zagreb, Croatia, pp.262-267.

Mehtre, B. M. and Chatterjee, B. (1989). "Segmentation of fingerprint images – A composite method." *Pattern Recognition* **22**(4): 381-385.

Mistretta, B., Morgan, D., et al. (1990). "Experiments with open set speaker identification. Sanders. " *Technical Report*.

Mitra, A., Banerjee, P. K., et al. (2005). "Automatic Authentication of Handwritten Documents via Low Density Pixel Measurements." *International Journal of Intelligent Technology* **1**(1): 7-11.

Moon, Y. S., Leung, C. C., et al. (2003). "Fixed-point GMM-based speaker verification over mobile embedded system." *Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*: 53-57.

Moore, T., Jesse, C., et al. (2001). An Overview and Evaluation of Decision Tree Methodology, url: www.ydyn.com/pubs/2001/asa.pdf.

Morris, A. (1992), "An information-theoretical study of speech processing in the peripheral auditory system and cochlear nucleus: application to the recognition of French voicesstop consonants." *Ph.D. dissertation*, ICP, INPG, Grenoble, France.

Morris, A. C., Hagen, A., et al. (2001). "Multi-stream adaptive evidence combination for noise robust ASR." *Speech Communication* **34**: 25-40.

Morris, A. C. (2002). An information theoretic measure of sequence recognition performance. *IDIAP Communication com02-03.*

Morris, A. C., Koreman, J., et al. (2004b). "Comparison of HMM and GMM for speaker recognition." *CoLi Technical Report, Saarland university, Germany.*

Morris, A., Wu, D., et al. (2004a). "An analysis of GMM based clustering in the acoustic feature space of the Timit speech database." *CoLi TR.*

Morris, A. C., Wu, D., et al. (2005). "MLP trained to classify a small number of speakers generates discriminative features for improved speaker recognition." *ICCST 2005.*

Nalwa, V. S. (1997). "Automatic On-Line Signature Verification." *Proc. IEEE* **85**(2): 215-240.

Nefian, A. and Hayes, M. H. (1999). "An embedded HMM for face detection and recognition." *Proc. Int. Conf. Acoustics, Speech, and Signal Process* **6**.

Newman, M., Gillick, L., et al. (1996). "Speaker verification through large vocabulary continuous speech recognition." *Proc. ICSLP'96.*

Penev, P. S. and Atick, J. J. (1996). "Local feature analysis: a general statistical theory for object representation." *Netw. Comput. Neural Syst.* **7**(3): 477-500.

Platt, J. (1998). "How to implement SVMs." *IEEE INTELLIGENT SYSTEM*: 26-28.

Podio, F. L. (2001). "Biometric Technologies for highly secure personal authentication." *http://csrc.nist.gov/publications/nistbul/itl05-2001.txt*.

Potamitis, I., Fakotakis, N., et al. (2000). "Spectral and cepstral projection bases constructed by independent component analysis." *Proc. ICSLP'00*.

Preparata, F. P., Hong, S. J. (1977). "Convex Hulls of Finite Sets of Points in Two and Three Dimensions", Commun. *ACM*, vol. 20, no. 2, pp. 87-93.

Rabiner, L. and Juang, B. H. (1993). *Fundamentals of Speech Recognition*, Prentice Hall Signal Processing Series.

Rabiner, L. R. (1987). "A tutorial on hidden Markov models and selected applications in speech recognition." *Proceedings of the IEEE* **77**(2): 257-286.

Ratha, N., Karu, K., et al. (1996). "A Real-time Matching System for Large Fingerprint Database." *IEEE Trans. on Pattern Anal. Machine Intell.* **18**(8): 799-813.

Reynolds, D. (1994). "Experimental Evaluation of features for Robust Speaker Indentification." *IEEE Trans. on Speech and Audio Processing* **2**(4): 639-643.

Reynolds, D. A. (1995c). "Large Population Speaker Identification Using Clean and Telephone Speech." *IEEE Signal Processing Letter* **2**(3).

Reynolds, D. A. (1995b). "Speaker identification and verification using Gaussian mixture speaker models." *Speech Commun.* **17**: 91-108.

Reynolds, D. A., Zissman, M. A., et al. (1995a). "The effect of telephone transmission degradations on speaker recognition performance." *Proc. ICASSP'95*: 329-332.

Reynolds, D. A. (2002). "An Overview of Automatic Speaker Recognition Technology." *Proc. ICASSP'02*.

Ribaric, S. and Fratric, I. (2005). " A biometric identification system based on eigenpalm and eigenfinger features." *IEEE Trans. PAMI* **27**(11): 1698-1709.

Rosenblattm, F. (1962). *Principles of Neurodynamics: Perceptrons and Theory of Brain Mechanisims*. Washington DC, Spartan.

Ross, A. and Jain, A. K. (2003). "Information Fusion in Biometrics." *Pattern Recognition Letters* **24**(13): pp. 2115-2125.

Ross, A. and Jain, A. K. (2004). "Multimodal Biometrics: An Overview." *Proc. of 12th European Signal Processing Conference (EUSIPCO)*: pp. 1221-1224.

Roweis, S. and Saul., L. (2000). "Nonlinear dimensionality reduction by locally linear embedding." *Science* **290**: 2323-2326.

S. Young et al. (2002). *HTKbook (V2.2)*, Cambridge University Engineering Dept.

Samaria, F. and Young, S. (1994). "HMM-based architecture for face identification." *Image Vis. Comput.* **12**(8).

Sanchez-Reillo, R., Sanchez-Avila, C., et al. (2000). "Biometric Identification through Hand Geometry Measurements." *IEEE Trans. Pattern Analysis and Machine Intelligence* **22**(10): 1168-1171.

Sarikaya, R., Pellom, B. L., et al. (1998). "Wavelet packet transform features with applications to speaker identification." *Proc. IEEE Nordic Signal Proc. Symp., (NORSIG'98)*: 81-84.

Satoh, S., Nakamura, Y., et al. (1999). "Name-It: naming and detecting faces in news videos." *IEEE Multimedia* **6**(1): 22-35.

Saul, L. and Roweis, S. (2000). "An Introduction to Locally Linear Embedding." *http://www.cs.toronto.edu/~roweis/lle/papers/lleintroa4.pdf.*

Schaumont, P., Hwang, D., et al. (2005). "Platform-Based Design for an Embedded-Fingerprint-Authentication Device." *IEEE Trans. on Computer-aided design of Integrated Circuits and Systems* **24**(12): 1929-1936.

Schiel, F. and Draxler C. (2004). "The production of speech corpora." http://www.phonetik.uni-muenchen.de/Forschung/BITS/TP1/Cookbook/Tp1.html.

Schölkopf, B. (1998). "SVMs-a practical consequence of learning theory." *IEEE Intelligent System*: 18-21.

Sellahewa, H. and Jassim, S. (2005). "Wavelet-based face verification for constrained platforms." *Proc. SPIE on Biometric Technology for Human Identification II,* orlando, Florida, Vol. 5779, pp.173-183.

Senior, A. "Biometrics short course." h*ttp://www.research.ibm.com/people/a/aws/ documents/CVPR-BiometricsShortCourse-Part1.zip.*

Seung, H. S. and Lee, D. D. (2000). "The Manifold Ways of Perception." *Science* **290**(5500): 2268-2269.

Shapiro, J. (2002). "MLP Design: Tricks of the Trade." *Lecture 2.3, Department of Computer Science, University of Manchester.*

Sharma, S., Ellis, D., et al. (2000). "Feature extraction using non-linear transformation for robust speech recognition on the Aurora database." *ICASSP' 00.*

Shi, X. and Manduchi, R. (2003). "A Study on Bayes Feature Fusion for Image Classification." *IEEE Workshop on Statistical Analysis in Computer Vision.*

Shire, M. L. and Chen, B. Y. (2000). "On data-derived temporal processing in speech feature extraction." *Proc. ICSLP'00.*

Shu, W. and Zhang, D. (1998). "Automated Personal Identification by Palmprint." *Optical Eng.* **37**(8): 2359-2362.

Smeraldi, F., Carmona, O., et al. (2000). "Saccadic search with Gabor features applied to eye detection and real-time head tracking." *Image Vis. Comput.* **18**(4): 323-329.

Snelick, R., Uludag, U., et al. (2005). "Large Scale Evaluation of Multimodal Biometric Authentication Using State-of-the-Art Systems." *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**(3): pp. 450-455.

Soltysiak, S. and Valizadegan, H. (2005). DNA as a Biometric Identifier, cse981, CMU.

Somervuo, P. (2003). "Experiments with linear and nolinear feature transformations in HMM based phone recognition." *Proc. ICASSP'03.*

Soong, F. K. and Rosenberg, A. E. (1988). "On the Use of Instantaneous and Transitional Spectral Information in Speaker Recognition." *IEEE Trans. Acoust., Speech, and Signal Proc.* **ASSP-36**(6): 871-879.

Sun, D. X. and Deng, L. (1995). "Analysis of Acoustic-Phonetic Variations in Speech." *Proc. ICASSP'95.*

Tang, T. Y., Y.S.Moon, et al. (2004). "Efficient implementation of fingerprint verification for mobile embedded systems using fixed-point arithmetic." *Proceedings of the 2004 ACM symposium on Applied computing*: 821-825.

Theodoridis, S. and Koutroumbas, K. (2003). *Pattern Recognition, 2nd Ed.*, Academic Press.

Tico, M., Immonen, E., et al. (2001). "Fingerprint Recognition Using Wavelet Features." *Proc. of IEEE International Symposium on Circuits andSystems* **2**: 21-24.

Van, B. L., Garcia-Salicetti, S., et al. (2004). "Fusion of HMM's Likelihood and Viterbi Path for On-line Signature Verification." *Lecture Notes in Computer Science* **3087 / 2004**.

Vapnik, V. N. (1998). *Statistical Learning Theory*. New York, John Wiley and Sons.

Varga, A. and Steeneken, H. J. M. (1993). "Assesment for automatic speech recognition:II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems." *Speech Communication* **12**(3): 247-252.

Venaz, P. T. and Hugli, H. (1995). "Usefulness of the LPC-residue in text-independent speaker verification." *Speech Communication* **17**(1-2): 145-157.

Walsh, P. S., Metzger, D. A., et al. (1991). "Chelex 100 as a medium for simple extraction of DNA for PCR-based typing from forensic material." *Biotechniques* **10**: 506-518.

Wan, V. and Renals, S. (2005). "Speaker Verification using Sequence Discriminant Support Vector Machines." *IEEE Transactions on Speech and Audio Processing* **13**(2): 203-210.

Wang, H. and Chang, S. F. (1997). "A highly efficient system for automatic face region detection in MPEG video." *IEEE Trans. Circ. Syst. Video Technol.* **7**(4): 615-628.

Wang, L., Chen, K., et al. (2002). "Capture Interspeaker Information With a Neural Network for Speaker Identification." *IEEE Trans. on Neural networks* **13**(2): 436-445.

Wang, L., Tieniu Tan, et al. (2003). "Silhouette Analysis-Based Gait Recognition for Human Identification." *IEEE Trans. PAMI* **25**(12): 1505-1518.

Wikipedia (2000). "Likelihood ratio test." h*ttp://en.wikipedia.org/wiki/*
Likelihood_ratio_test.

Wildermoth, B. and Paliwal, K. K. (2003). "GMM based speaker recognition on readily available databases." *Proc. Microelectronic Engineering Research Conference.*

Wildes, R. P. (1997). "Automated iris recognition: An emerging biometric technology." *Proceedings of the IEEE* **85**: 1348-1363.

Wirtz, B. (1995). "Stroke-based Time Warping for Signature Verification." *Proc. Third Int'l Conf. Document Analysis and Recognition*: 179-182.

Wiskott, L., Fellous, J. M., et al. (1997). "Face recognition by elastic bunch graph matching." *IEEE Trans. Patt. Anal. Mach. Intell.* **19**(7): 775-779.

Wu, D., Morris, A. C., et al. (2005c). "Discriminative features by MLP preprocessing for robust speaker recognition in noise." *Proc. ESSV 2005.*

Wu, D., Morris, A. C., et al. (2005b). "MLP internal representation as discriminative features for improved speaker recognition." *Nonlinear Analyses and Algorithms for Speech Processing Part II (series: Lecture Notes in Computer Science).*

Wu, D., Morris, A. C., et al. (2005a). "MLP internal representation as discriminative features for improved speaker recognition." *Proc. NOLISP'05*: 25-32.

Wu, Dalei, (2006), “Capturing the locality of the acoustic space by locally nonlinear transformations with application to improved text-independent speaker Identification.” *Technical Report, Saarland University.*

Wu, H., Chen, Q., et al. (1999). "Face detection from color images using a fuzzy pattern matching method." *IEEE Trans. Patt. Anal. Mach. Intell.* **21**(6): 557-563.

Wu, Q. Z., Jou, I. C., et al. (1997). "On-Line Signature Verification Using LPC Cepstrum and Neural Networks." *IEEE Trans. Systems, Man, and Cybernectics* **27**(1): 148-153.

Wu, Q. Z., Lee, S. Y., et al. (1998). "On-line signature verification based on logarithmic spectrum." *Patern Recognition* **31**(12): 1,865-1,871.

Yan, Y. and Chengyi Zheng et. al (2003). "A Dynamic Cross-Reference Pruning Strategy For Multiple Feature Fusion at Decoder Run Time." *Proc. Eurospeech'03*.

Yang, I., Widjaja, L. K., Prasad, R. (1995). "Application of Hidden Markov Modols for Signature Verification." *Patern Recognition* **28**(2): 161-170.

Yen, R. C. (2004). "Forensic DNA Typing and Prospects for Biometrics." *Biometric Identification Seminar*.

You, J., Li, W., et al. (2002). "Hierarchical Palmprint Identification via Multiple Feature Extraction." *Pattern Recognition* **35**(4): 847-859.

Young, N. D., Harkin, G., et al. (1997). "Novel Fingerprint Scanning Arrays Using Polysilicon TFT's on Glass and Polymer Substrates." *IEEE Electron Device Letters* **18**(1): 19–20.

Yu, K., Wang, Y., et al. (2004). "Writer authentication based on the analysis of strokes." *Proc. SPIE, Biometric Technology for Human Identification* **5404**: 215-224.

Zhang, D., Kong, W. K., et al. (2003). "Online Palm Print Identification." *IEEE Trans. Pattern Analysis and Machine Intelligence* **25**(2): 1041-1050.

Zhao, W., Chellappa, R., et al. (1998). "Discriminant analysis of principal components for face recognition." *Proc. IEEE Conf. Comput. Vis. Patt. Recogn.*: 336–341.

Zigel, Y. and Cohen, A. (2003). "On cohort selection for speaker verification." *Proc. Eurospeech 2003*: 2977-2980.

Appendix A: notation and standard formulae

Notation

$\mathbf{x}$	a feature data point (vector)
X	data matrix, with x as rows
Y	target output matrix, with 0/1 target vectors as rows
$f_i(\mathbf{x})$	Gaussian pdf for GMM cluster (i)
G_i	$\{x : f_i(x) \geq f_j(x) \forall i \neq j\}$, the set of x in Gaussian cluster (i)
D	speech or speaker class division (e.g. phone, speaker, or gender)
d_i	categories within class D (e.g. male, female)
$\|D\|$	number of categories in class division D
X_i	$D \cap G_i$, the set of all D in Gaussian (i)
S_i	covariance matrix for data set X_i
S_b	between-class covariance matrix
S_w	within-class covariance matrix
$\|X\|$	number of data points in set X
$T(D_1, D_2, X)$	contingency table counts of x for class division D_1 against D_2
n_{ij}	element (i,j) of T
N	$\sum_{ij} n_{ij}$

Standard formulae

$\mu_i = \sum_{x \in C_i} x / \|C_i\|$**,** mean of data in category C_i

$S_i = E[(x-\mu_i)(x-\mu_i)'] = C_i C_i' / \|C_i\| - \mu_i \mu_i'$, within-class covariance matrix

$P_i = \|C_i\| / \|X\|$, relative frequency of category C_i

$\mu = \sum_i P_i \mu_i$, overall data mean

$S_b = \sum_i P_i (\mu_i - \mu)(\mu_i - \mu)'$, between-class covariance matrix

$S_w = \sum_i P_i S_i$, overall within-class covariance matrix

$H(D) = -\sum_{d \in D} P(d) \log_2 P(d)$, entropy of the probability distribution of D

$L(D_1, D_2) = \Sigma ij\,(n_{ij} - (n_i n_j \big/ N))^2 \Big/ (n_i n_j / N)$, Pearson’s large sample (or Chi-squared) statistic

Appendix B: The derivation of linear prediction coefficients

The signal *s(t)* at time *t* can be modelled by a linear equation as:

$$s(t) = \sum_{k=1}^{p} \alpha_k s(t-k) + \sqrt{g_s}\, u(t), \tag{B.1}$$

u(t) is called the residue, g_s is a scaling parameter and p is the prediction degree. The p coefficients α_k are often used as a p-dimensional vector to represent a speech frame, when they are called linear prediction coefficients (LPC).

Formula (B.1) can be written in a vector form. Thus we can obtain

$$X(t) = \Phi X(t-1) + U(t) \tag{B.2}$$

where

$$\Phi = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & & & \vdots \\ \alpha_1 & \alpha_2 & \dots & \alpha_p \end{bmatrix}, \tag{B.3}$$

$$X(t) = \left[s(t-p), s(t-p+1), \dots, s(t)\right]', \tag{B.4}$$

and

$$U(t) = \left[\sqrt{g_s}\, u(t-p), \sqrt{g_s}\, u(t-p+1), \dots, \sqrt{g_s}\, u(t)\right]'. \tag{B.5}$$

From these formulae, we can see that the AR model is a special case of a linear dynamic system with the noise term in (B.1) given by the residual.

Solution to the AR model: Let $\tilde{e}(n)$ be the square distance between the true value *s(n)* and its estimate $\hat{s}(n)$, i.e.

$$\tilde{e}(n) = \left(s(n) - \hat{s}(n)\right)^2, \tag{B.6}$$

where $\hat{s}(n)$ is obtained by linear prediction according to its past p values, i.e.

$$\hat{s}(n) = \sum_{k=1}^{p} \alpha_k s(n-p). \qquad \text{(B.7)}$$

The optimisation objective is to minimise the sum E of $\tilde{e}(n)$ over the time index *n*, i.e.

$$\{\alpha_i\} = \arg\min_{\alpha_i} \sum_{n=1}^{N} \left(s(n) - \hat{s}(n)\right)^2, \qquad \text{(B.8)}$$

where *N* is the number of observed samples in a frame.

Differentiating (B.8) with respects to α_k, we have

$$\frac{\partial E}{\partial \alpha_k} = \sum_{n=1}^{N} 2\left(s(n) - \hat{s}(n)\right)s(n-k) = 0, \ \vee\, k = 1 \ldots p. \qquad \text{(B.9)}$$

Substituting (B.7) into (B.9) and rearranging it, it yields

$$\sum_{n=1}^{N} s(n)s(n-k) = \sum_{n=1}^{N} \hat{s}(n)s(n-k) = \sum_{j=1}^{p} \alpha_j \sum_{n=1}^{N} s(n-j)s(n-k), \qquad \text{(B.10)}$$

and the left-hand side is referred to as the *k-th* order of autocorrelation. Thus (B.10) leads to

$$R_k = \sum_{j=1}^{p} \alpha_j R_{j-k}. \qquad \text{(B.11)}$$

Rewriting these *p* linear equations in matrix form and noting $R_{-k} = R_k$, we have

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_{p-1} \\ R_1 & R_0 & \cdots & R_{p-2} \\ \vdots & \vdots & \ddots & \vdots \\ R_{p-1} & R_{p-2} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_p \end{bmatrix}. \qquad \text{(B.12)}$$

This is called the Yuler-Walker equation.

As shown in (B.12), solving this equation requires the computation of all the *p* autocorrelations and matrix inversion. The matrix inversion problem can be greatly simplified because of the symmetric Toeplitz autocorrelation matrix on the left-hand side, and the form of autocorrelation vector on the right-hand side of (B.12). Durbin's recursive algorithm can be used to find a solution (B.13). Note that in the process of solving the predictor coefficients α_k of the order *p*, the α_k for all orders less than *p* are obtained with their corresponding mean square prediction error $MSE_i = E_i / R_0$ (B.13). In each recursion, the prediction order is increased and a corresponding error is determined. This can be monitored as a stopping criterion on the prediction order *p*.

$$\left.\begin{aligned} & E_0 = R_0 \\ & k_i = -\left[R_i + \sum_{j=1}^{i-1} \alpha_j^{(i-1)} R_{i-j} \right] / E_{i-1}, \quad \vee\, i = 1 \ldots p \\ & \alpha_i^{(i)} = k_i \\ & \alpha_j^{(i)} = \alpha_j^{(i-1)} + k_i \alpha_{i-j}^{i-1}, \quad \vee\, 1 \le j \le i\text{-}1 \\ & E_i = (1 - k_i^2) E_{i-1} \end{aligned}\right\}$$

$$\vee\, i = 1 \ldots p \quad . \tag{B.13}$$

$$\alpha_{\mathrm{j}} = \alpha_j^{(p)}, \qquad \vee\, j = 1 \ldots p$$

Appendix C: Abbreviations:

AMR:	arithmetic mean rule
ANN:	artificial neural network
API:	application program interface
ASR:	automatic speech recognition
CMS:	cesptrum mean subtraction
COTS:	commercial off-the-shelf
DCT:	discrete cosine transform
DET:	detection error tradeoff
DFT:	discrete Fourier transform
DNA:	deoxyribonucleic acid
EM:	expectation maximization
FA:	false acceptance
FR:	false rejection
FTIR:	frustrated total internal reflection
GMM:	Gaussian mixture model
HD:	Hamming distance
HMM:	hidden Markov model
ICA:	independent component analysis
LDA:	linear discriminant analysis
LLE:	locally linear embedding
LMLP:	linear multi-layer perceptron
LMS:	least mean square
LPC:	linear prediction coefficients
MaxAD:	maximum average distance
MFCC:	mel-freqency cepstrum coefficients
MLP:	multi-layer perceptron
MMLP:	multiple MLPs
NLDA:	nonlinear discriminant analysis
PCA:	principal components analysis
PKI:	public key infrastructure
PLP:	perception linear prediction
ROC:	receiver operating characteristic curves
SNR:	signal-to-noise ratios

SOM:	self organisation map
SPD:	speaker-phoneme distribution
STC:	SOM trajectory coefficients
STR:	short tandem repeats
SVS:	speaker voice signature
T-norm:	test normalisation
UBM:	universal background model
WAVC:	wavelet transform coefficients
Z-norm:	zero normalisation

Index: