SENSOR FUSION FOR AUDIO-VISUAL BIOMETRIC AUTHENTICATION

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRONIC AND INFORMATION ENGINEERING
THE HONG KONG POLYTECHNIC UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF PHILOSOPHY

Cheung Ming Cheung

October 2004

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best to my knowledge and belief, it reproduces no material previously published or written nor material which has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

____________________(Signed)

Cheung Ming Cheung (Name of student)

Abstract

Although financial transactions via automatic teller machines (ATMs) have become commonplace, the security of these transactions remains a concern. In particular, the verification approach used by today's ATMs can be easily compromised because ATM cards and passwords can be lost or stolen. To overcome this limitation, a new verification approach known as biometrics has emerged. Rather than using passwords as the means of verification, biometric systems verify the identity of a person based on his or her physiological and behavioral characteristics.

Numerous studies have shown that biometric systems can achieve high performance under controlled conditions. However, the performance of these systems can be severely degraded under real-world environments. For example, background noise and channel distortion in speech-based systems and variation in illumination intensity and lighting directions in face-based systems are known to be the major causes of performance degradation. To enhance the robustness of biometric systems, multimodal biometrics have been introduced. Multimodal techniques improve the robustness of biometric systems by using more than one biometric traits at the same time. Combining the information from different traits, however, is an important issue. This thesis proposes a multiple-source multiple-sample fusion algorithm to address this issue. The algorithm performs fusion at two levels: intramodal and intermodal.

In intramodal fusion, the scores of multiple samples (e.g., utterances and video shots) obtained from the same modality are linearly combined, where the fusion weights are made dependent on the score distribution of the independent samples and the prior knowledge about the score statistics. More specifically, enrollment data are used to compute the mean scores of clients and impostors, which are considered to be the prior scores. During verification, the differences between the individual scores and the prior scores are used to compute the fusion weights. Because the fusion weights depend on verification data, the position of scores in the score sequences is detrimental to the final fused scores. To enhance the discrimination between client and impostor scores, this thesis proposes sorting the score sequences before fusion takes placed. Because verification performance depends on the prior scores, a technique that adapts the prior scores during verification is also developed.

In intermodal fusion, the means of intramodal fused scores obtained from different modalities are fused by either linear weighted sums or support vector machines. The final fused score is then used for decision making.

The intramodal multisample fusion was evaluated on the HTIMIT corpus and the 2001 NIST speaker recognition evaluation set, and the two-level fusion approach was evaluated on the XM2VTSDB audio-visual corpus. It was found that intramodal multisample fusion achieves a significant reduction in equal error rate as compared to a conventional approach in which equal weights are assigned to all scores. Further improvement can be obtained by either sorting the score sequences or adapting the prior scores. It was also found that multisample fusion can be readily combined with support vector machines for audio-visual biometric authentication. Results show that combining the audio and visual information can reduce error rates by as much as 71%.

STATEMENT OF ORIGINALITY

The major contributions of this dissertation are summarised below.

- This dissertation proposes a data-dependent decision fusion algorithm in which fusion weights for individual scores are based on the score distribution of the utterances and on the prior score statistics determined from enrollment data.

- This dissertation proposes sorting the score sequences and adapting prior scores for further improvement. Experimental results show that the data-dependent decision fusion outperforms equal-weight fusion.

- This dissertation proposes integrating data-dependent decision fusion with blind feature-based transformation for channel robust speaker verification. Results show that the proposed transformation approach achieves significant improvement in both equal error rate and minimum detection cost.

- This dissertation proposes a two-level fusion approach for audio and visual biometric authentication. In intramdol fusion, the data-dependent fusion is used to combine the scores from the same modality. In intermodal fusion, either linear weighted sums or support vector machines are used to combine the means of intramodal fused scores obtained from different modalities.

Acknowledgments

First of all, I would like to express my sincere gratitude to my Supervisor, Dr. Man-Wai Mak, for his patient and helpful guidance throughout the study. Without his support, this research work would not have been completed. He also offered me many invaluable ideas and suggestions in writing my thesis.

I am also thankful to all the members of the DSP Research Laboratory, especially for K. K. Yiu, K. Y. Leung, C. H. Sit, K. K. Kwok, and C. L. Tsang. The countless discussions I had with them have been proved to be both fruitful and inspiring.

I would also like to take this opportunity to thank the Centre for Multimedia Signal Processing of the Department of Electronic and Information Engineering, and the Research and Postgraduate Studies Office of The Hong Kong Polytechnic University for theirs generous support over the past two years.

Last but not least, without the patience and forbearance of my family, the preparation of this research work would have been impossible. I appreciate their constant and continuous support and understanding.

Glossary of Symbols

w_i	Fusion weight for the i-th modality
s_i	Score obtained from the i-th modality
K	Number of modalities
$p(\mathbf{x}_t \vert \Lambda_{\omega_i})$	Likelihood function for model ω_i
$\Theta_{m\vert i}$	The parameters of the m-th mixture component
M	The total number of mixture components
$p(\mathbf{x}_t \vert \omega_i, \Theta_{m\vert i})$	Probability density function of the m-th component
$P(\Theta_{m\vert i} \vert \omega_i)$	The prior probability (also called mixture coefficients) of the m-th mixture component
$\mathcal{N}(\mu_{m\vert i}, \Sigma_{m\vert i})$	A Gaussian distribution with mean $\mu_{m\vert i}$ and covariance matrix $\Sigma_{m\vert i}$
π_i	The prior probability of occurrence of the i-th mixture component
$s(X; \Lambda)$	The normalized score given a sequence of vectors X
$\mathbf{y}_t$	The distorted vector at frame t
$\hat{\mathbf{x}}_t$	A transformed vector at frame t
$s_t^{(k)}$	A pattern-based score for modality k at time t
$\alpha_t^{(k)}$	A fusion weight for score $s_t^{(k)}$
$\tilde{\mu}_c$	The mean score of a client speaker
$\tilde{\mu}_b$	The mean score of background speakers
$\tilde{\mu}_p$	Prior score
$\tilde{\sigma}_p^2$	Prior variance
Y_ν	A sequence of transformed vectors
$\tilde{s}$	Utterance-based background speakers' score
Ω_s	Pseudo-imposters' scores model
ζ	The normalized likelihood that the claimant is an impostor
$\tilde{s}_k$	Utterance-based verification score from utterance k
$\widehat{\mu}_p^{(k)}$	The adapted prior score of utterance k
$\bar{s}^{(m)}$	The mean score of modality m
$\hat{s}^{(m)}$	The mean fused score of modality m using data-dependent fusion
$s_{norm}^{(m)}$	The zero-normalized score of modality m
β	The combination weight of audio score
α_j	Lagrange multipliers in support vector machines
Ω	A set containing the indexes to the support vectors
$K(\mathbf{x}, \mathbf{x}_j)$	A kernel function of support vector machines

Glossary of Abbreviations

ATM	Automatic teller machine
AV	Audio-visual
PCA	Principle component analysis
LDA	Linear discriminant analysis
SNR	Signal to noise ratio
GMMs	Gaussian mixture models
LPCCs	LP-derived cepstral coefficients
MFCCs	Mel-frequency cepstral coefficients
EM	Expectation-maximization (EM)
MAP	Maximum a posteriori
CMS	Cepstral mean subtraction
BSFT	Blind Stochastic Feature Transformation
FR	False rejection
FA	False acceptance
FRR	False rejection rate
FAR	False acceptance rate
HTER	Half total error rate
EER	Equal error rate
ROC	Receiver operating characteristic
DET	Decision error tradeoff
EW	Equal-weight
DF	Data-dependent fusion
DCF	Decision cost function
PS	Prior score
Z-norm	Zero normalization
SVM	Support vector machine

List of Publications

Chapters in Books

1. M. C. Cheung, M. W. Mak, and S. Y. Kung. "Probabilistic Fusion of Sorted Score Sequences for Robust Speaker Verification," to appear in Y. P. Tan, K. H. Yap and L. Wang, editors, *Intelligent Multimedia Processing with Soft Computing*, Springer.

Journal Papers

2. K. K. Yiu, M. W. Mak, M. C. Cheung and S. Y. Kung. Blind Stochastic Feature Transformation for Channel Robust Speaker Verification, accepted by *J. of VLSI Signal Processing*.

International Conference Papers

3. M. W. Mak, M. C. Cheung, and S. Y. Kung. "Robust Speaker Verification from GSM-transcoded Speech Based on Decision Fusion and Feature Transformation," in *ICASSP'03*, Hong Kong, March. 2003, vol.2, pp. 745-748.

4. M. C. Cheung, M. W. Mak, and S. Y. Kung. "Adaptive Decision Fusion for Multi-Sample Speaker Verification over GSM Networks," in *Eurospeech'03*, Geneva, Sept. 2003, pp. 2969-2972.

5. M. C. Cheung, M. W. Mak, and S. Y. Kung. “Multi-sample Data-dependent Fusion of Sorted Score Sequences for Biometric Verification,” in *ICASSP’04*, Montreal, May. 2004, vol. 1, pp. 85-88.

6. M. C. Cheung, K. K. Yiu, M. W. Mak, and S. Y. Kung. “Multi-Sample Fusion with Constrained Feature Transformation for Robust Speaker Verification,” in *Int. Conf. on Spoken Language Processing*, Korea, October 2004, pp. 1813-1816.

7. K. K. Yiu, M. W. Mak, M. C. Cheung, and S. Y. Kung. “A New Approach to Channel Robust Speaker Verification via Constrained Stochastic Feature Transformation,” in *Int. Conf. on Spoken Language Processing*, Korea, October 2004, pp. 1753-1756.

8. M. C. Cheung, M. W. Mak, and S. Y. Kung. “Intramodal and Intermodal Fusion for Audio-Visual Biometric Authentication,” in *International Symposium on Intelligent Multimedia, Video & Speech Processing*, October 2004, pp. 25-28.

9. K. K. Yiu, M. W. Mak, M. C. Cheung, and S. Y. Kung. “Blind Stochastic Feature Transformation for Speaker Verification over Cellular Networks,” in *International Symposium on Intelligent Multimedia, Video & Speech Processing*, October 2004, pp. 679-682.

10. M. C. Cheung, M. W. Mak, and S. Y. Kung. “A Two-level Fusion Approach to Multimodal Biometric Verification,” in *ICASSP’05*, Philadelphia, March. 2005, vol.1, pp. 181-184.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter, a brief overview of biometrics is presented. Then, common fusion strategies are introduced. Finally, the objectives of this research are stated.

1.1 An Overview of Biometrics

Person verification systems for financial transactions and access controls are commonplace in today's information societies. Most of these systems compare the personal identity numbers (PINs) or passwords entered by their users against the ones stored in a central database for user authentication. This authentication method, however, may introduce inconvenience to users. For example, users of automatic teller machines (ATMs) need to carry an ATMs card and remember a password to access their bank accounts. Even if the users can remember the passwords, this kind of system is not capable of verifying whether the password is entered by a registered user or an impostor. If an impostor possesses an ATM card together with its associated password, the system will accept him or her. On the other hand, if a registered user forgets the password, he or she will not be able to access the system.

To cope with the above problem, biometric verification has emerged. Biometric authentication systems measure the physiological and/or behavioral characteristics of a person, such as speech, face image, fingerprints, iris, etc., as a means of verifying the person's identity. Although this kind of system may have a small chance of accepting an impostor or rejecting a registered user, they can greatly reduce the risk of forgery, because biometric systems identify a user according to his or her biometric characteristics, instead of what he or she knows (e.g., a code) or possesses (e.g., a card).

The choice of biometric traits for verification is very important. There are two requirements that a biometric trait should fulfill. First, the biometric trait should be distinguishable, which means that the trait should have a large variation among users. Second, the trait should lead to user friendly systems. In particular, the sensors that capture the biometric trait should not annoy users. Therefore, it is better to capture the trait in an unconstrained and contactless manner. Speech and face are two biometric traits that fulfill these two requirements.

The advantage of using speech-based verification is that speech is commonly available in communication systems, e.g., mobile phone [1]. Speech-based systems can be divided into two categories: text-dependent [2] and text-independent [3]. In text-dependent systems, claimants should utter a sentence specified by the systems. Therefore, text-dependent systems allow interactive authentication. On the contrary, claimants can speak whatever they like in text-independent systems. This kind of verification system is more appropriate for forensic and surveillance applications where predefined keywords are not available. The major problem in speech-based authentication is that the performance can be severely degraded by the mismatch between the training and verification conditions. The mismatch can be caused by channel distortion, transducer mismatches, and background noise.

Besides speech, faces are another biometric trait that has been widely used for verification. Face verification is user friendly in that the claimants only need to look at a camera for a few seconds. However, similar to speech-based verification, environmental conditions are critical; for example, change in illumination directions can severely affect the recognition performance [4].

To cope with the limitations of individual biometrics, researchers have proposed using multiple biometric traits concurrently for verification. Such systems are commonly known as multimodal verification systems [5]. By using multiple biometric traits, systems become more robust to intruder attack. For example, it will be more difficult for an impostor to impersonate another person using both audio and visual information simultaneously. Therefore, audio-visual (AV) biometrics has attracted a great deal of attention in recent years. However, multiple biometrics should be used with caution because catastrophic fusion may occur if the biometrics are not combined properly. Catastrophic fusion occurs when the performance of an ensemble of combined classifiers is worse than any of the individual classifiers.

1.2 An Overview of Sensor Fusion

Sensor fusion is a technique developed for optimal information processing through combination of information produced by several sources [6, 7]. The human brain is a good example to illustrate the fusion concept; it receives five different signals (sight, hearing, taste, smell, and touch) from five different sensors (eyes, ears, tongue, nose, and skin). Typically, the human brain fuses signals from different sensors to make optimal decisions. The human brain also fuses signals at different levels for different purposes. For example, humans recognize objects by both seeing and touching the objects; humans also communicate by watching the talker's face and listening to his or her voice at the same time. All of these phenomena suggest that the human brain is a flexible and complicated fusion system.

Research in sensor fusion can be traced back to early 1980s [8, 9]. Sensor fusion can be applied in many ways, such as detection of the presence of an object, recognition of an object, tracking of an object, and so on. This thesis focuses on sensor fusion for verification purpose.

Information can be fused at two different levels: *feature-level* and *decision-level*. Decision-level fusion can be further divided into *abstract fusion* and *score fusion*.

1.2.1 Feature-Level Fusion

In feature-level fusion, data from different modalities are combined at the feature level before being presented to a pattern classifier [10]. One possible approach is to concatenate the feature vectors derived from different modalities [10], as illustrated in Figure 1.1. Alternatively, features vectors derived from the same modality but using different feature extraction methods can be concatenated [11]. The dimensionality of the concatenated vectors, however, is sometimes too large for a reliable estimation of classifier's parameters, a problem known as the curse of dimensionality. Although dimensionality reduction techniques such as PCA or LDA can help alleviate the problem [10, 12], these techniques rely on the condition that data from each class contain a single cluster only. Classification performance could be degraded when the data from individual classes contain multiple clusters.

Systems based on feature-level fusion are not very flexible because the system needs to be retrained whenever a new sensor is added. It is also important to synchronize different source of information in feature-level fusion, which may introduce implementation difficulty in AV biometric systems.

1.2.2 Decision-Level Fusion

Unlike feature fusion, decision fusion attempts to combine the decisions made by multiple modality-dependent classifiers (see Figure 1.2). This fusion approach solves

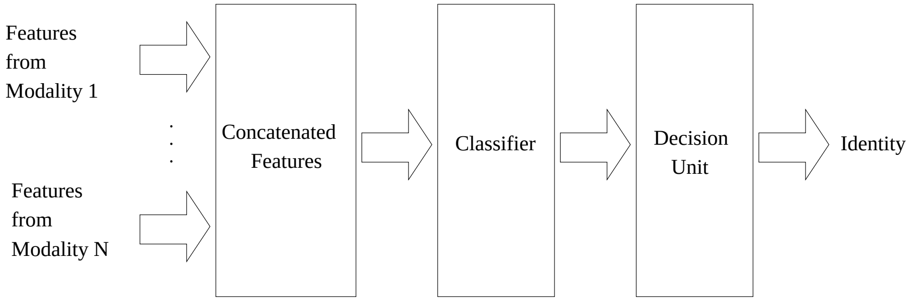


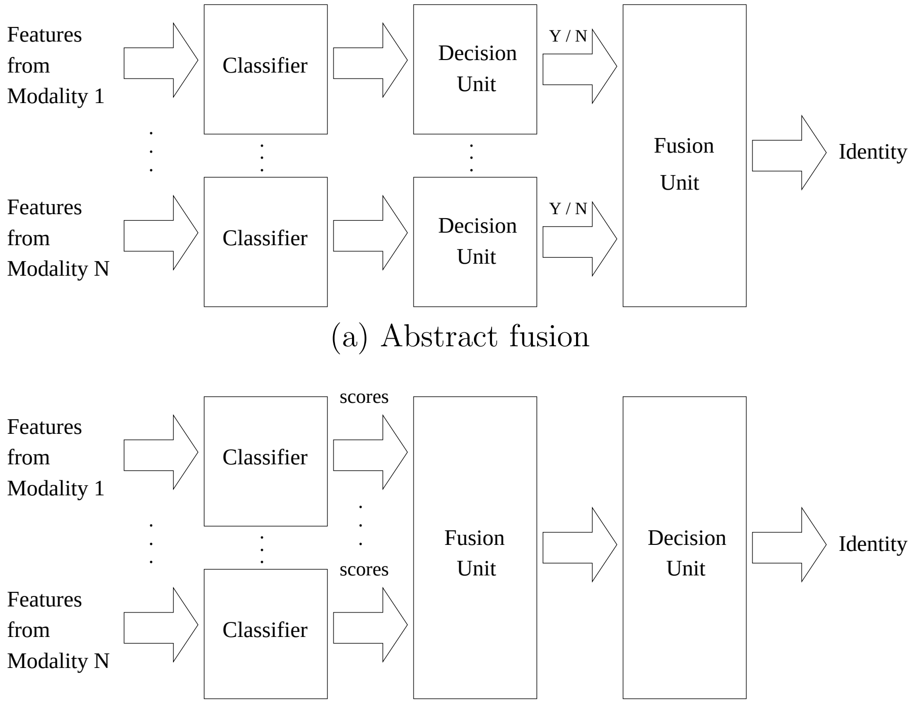
Figure 1.1: Architecture of feature-level fusion. Features are concatenated before fusion takes place.

the curse of dimensionality problem by training the modality-dependent classifiers separately. Combining the outputs of the classifiers, however, is an important issue. The classifiers can be identical but using different features (e.g., fingerprint and speech data) as input [13, 14]. Alternatively, different classifiers can work on the same features and their decisions are combined [15]. Also, there are systems that use the combination of these two types.

There are two types of decision fusions: *abstract fusion* and *score fusion*. In the former, the binary decisions made by individual classifiers are combined (see Figure 1.2(a)), whereas in the latter the scores (confidence) of the classifiers are combined (see Figure 1.2(b)).

In abstract fusion, the binary decisions can be combined by majority voting or using AND and OR operators. In majority voting [16–18], the final decision is based on the number of votes made by the individual classifiers. However, this voting method may have difficulty in making a decision when there are an even number of sensors and the decisions made by half of the classifiers do not agree with the other half.

Varshney [19] proposed to use logical AND and OR operators for fusion. In the AND fusion, the final decision is the logical AND of the classifiers' output. This type of fusion is very strict and therefore is suitable only for systems that require low false



(a) Abstract fusion

(b) Score fusion

Figure 1.2: Architecture of decision-level fusion. (a) Abstract fusion in which the Yes/No decisions made by the classifiers and decision units are combined. (b) Score fusion in which final decisions are based on the fused scores.

acceptance. However, it has difficulty when the the decisions made by different sensors are not consistent, which is a serious problem in multiclass applications. Unlike the AND fusion, the final decision in the OR fusion is the logical OR of the classifiers' output. This type of fusion is loose and is suitable only for systems that can tolerate a loose security policy (i.e., allowing high false acceptance error). The OR fusion suffers the same problem as the voting method when the decisions of individual classifiers do not agree with each other.

In score fusion, the scores of modality-specific classifiers are combined and the final score is used to make a decision (see Figure 1.2(b)). Typically, the output of modality-specific classifiers are linearly combined through a set of fusion weights [17].

The final score is obtained from

$$s = \sum_{i=1}^{K} w_i s_i, \tag{1.1}$$

where K is the number of modalities or experts, $\{w_i\}$ are a set of fusion weights, and $\{s_i\}$ are the scores obtained from the K modalities. This kind of fusion is also referred to as the *sum rule* [17, 20].

Scores can be interpreted as posteriori probabilities in the Bayesian framework. Assuming that scores from different modalities are statistically independent, the final score can be combined by using the product rule [17, 20]:

$$s = \prod_{i=1}^{K} s_i. \tag{1.2}$$

To account for the discriminative power and reliability of each modality, a set of weights can be introduced as follows:

$$s = \prod_{i=1}^{K} (s_i)^{w_i}. \tag{1.3}$$

It has been stated that the independence assumption is unrealistic in many situations. However, for some applications, this assumption does hold. For example, in audio-visual verification systems, facial and speech features are largely independent. Therefore, fusion of audio and visual data at the score level is a possible solution to increasing verification accuracy.

The fusion weights w_i can be non-adaptive and adaptive. Non-adaptive weights are learned from training data and kept fix during recognition. For example, in [15, 21, 22], the fusion weights were estimated by minimizing the misclassification error on a held-out set; in [23], the parameters of a logistic regression model are estimated from the dispersion between the means of speaker's scores and impostors' scores. The non-adaptive weights, however, may not be optimal in mismatch conditions. Adaptive weights, on the other hand, are estimated from observed data during recognition, e.g., according to the signal-to-noise ratio [24, 25], degree of voicing [12], degree of mismatch between training and testing conditions [26], and amount of estimation error presented in each modality [27]. Lau et al. [28] proposed using fuzzy logic decision fusion to account for environmental mismatches.

Another important approach to adapting the fusion weights is to use score/rank hybrids or rank order statistics extracted from observed data during recognition. For example, the score-rank couples of individual experts can be concatenated to form vectors which is to be classified by another classifier [29]. Alternatively, they can be used for computing the score dispersion over a few best recognition hypothesis [30, 31]. The key idea is that the confidence of a modality should be proportional to the score dispersion.

The linear combiners mentioned earlier assume that the combined scores obtained from different classes are linearly separable. In case the linearly separable condition cannot be met, the scores obtained from d modalities can be considered as d-dimensional vectors and nonlinear binary classifiers (e.g., support vector machines, radial basic function networks, multilayer perceptrons, binary decision trees, Fisher's linear discriminant, and Bayesian classifiers) can be trained from a held-out set to classify the vectors [32–34]. This kind of fusion strategy is also known as *learning-based fusion* [35, 36].

1.3 Research Objectives

In recent years, research has focused on using fusion techniques to improve the performance of biometric systems. One popular approach is to fuse the scores obtained from audio and visual channels. The rationale behind most audio-visual systems is that when audio signals are contaminated by acoustic noise, visual cues can help reduce the recognition errors. In many cases, the weight applied to the audio channel is dependent on the acoustic SNR. While this approach is reasonable, it is also important to point out that acoustic distortion can be minimized by channel and noise compensation techniques. Fusion will become more effective if compensation techniques are also used. The objective of this research is to develop fusion algorithms for improving the performance of multimodal verification systems. To this end, this work investigates several approaches to computing the fusion weights and proposes a fusion algorithm in which the fusion weights are not only dependent on the audio quality but also on the effectiveness of the compensation techniques.

Another problem of conventional audio-visual fusion is that the fusion weights are either determined "off-line" during training or determined "online" during recognition based on the quality of the test data. This thesis proposes a method that uses both training and recognition data for determining the fusion weights.

Besides multi-modalities fusion, fusion techniques can also be applied to fuse the decisions or scores from a single modality. Therefore, another objective of this study is to investigate the fusion of scores obtained from a single modality in order to increase verification accuracy.

1.4 Organization of the Thesis

This thesis is organized as follows. Chapter 2 gives a brief overview of speaker verification systems. Chapter 3 describes the fusion approaches for improving the performance of speaker verification systems. Chapter 4 extends the fusion approaches proposed in Chapter 3 to audio-visual biometric authentication. Finally, Chapter 5 draws a conclusion of the thesis.

Chapter 2

Speaker Verification Systems

Speaker verification is to verify a speaker's claimed identity based on his or her voice. A speaker claiming an identity is called a *claimant*, and an unregistered speaker posing as a registered speaker is an *impostor*. An ideal speaker verification system should not reject registered speakers (*false rejection*) or accept impostors as registered speakers (*false acceptance*). This chapter describes the architecture of typical speaker verification systems. Compensation of channel distortion in the feature domain will also be discussed.

2.1 Components of Speaker Verification Systems

Typically, a speaker verification system is composed of a front-end feature extractor (Section 2.2), a classifier (Section 2.3), and a decision unit. Figure 2.1 shows the architecture of a typical speaker verification system.

The spectra of speech signals encode the formants and pitch harmonics that represent the vocal-tract shape (e.g., length and cross-section area) and glottal source information of speakers. Therefore, the feature extractor extracts speakers' features

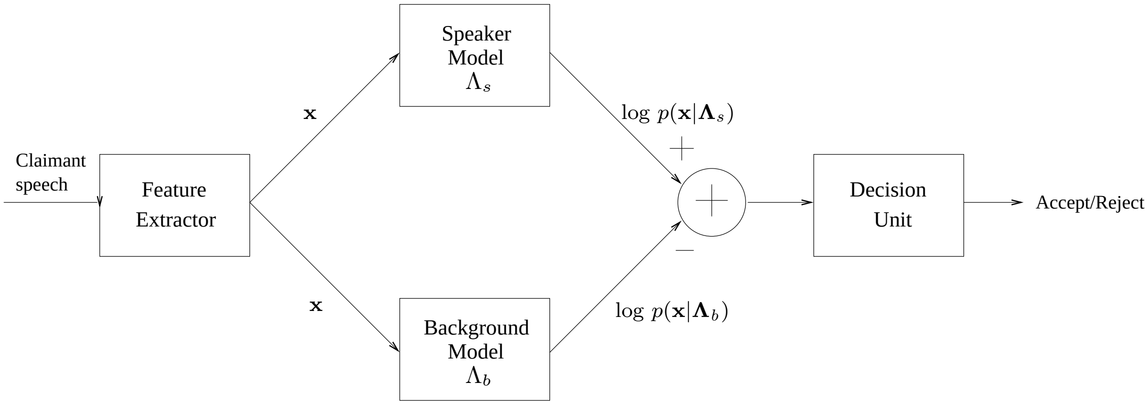

Speaker Model Λ_s
Claimant speech
Feature Extractor
$\mathbf{x}$
$\mathbf{x}$
$\log\ p(\mathbf{x}|\mathbf{\Lambda}_s)$
+
−
$\log\ p(\mathbf{x}|\mathbf{\Lambda}_b)$
Background Model Λ_b
Decision Unit
Accept/Reject

Figure 2.1: Architecture of a typical speaker verification system.

from the spectra of speech signals; the resulting feature vectors are used for training speaker-dependent models. Besides speaker models, a background model is also trained using the features extracted from the speech of a large number of speakers. The purpose of the background model is to normalize the scores of the speaker models, which has the effect of minimizing the speaker-independent variations such as background noise and channel distortion. Finally, the normalized scores are passed to a decision unit for making an accept or reject decision. The decision unit can be a simple thresholding unit. More specifically, if the normalized score is greater than a predefined threshold, the system will accept the claimant; otherwise, the system will reject the claimant. A more detailed description of speaker recognition systems can be found in Campbell [37].

2.2 Feature Extraction

Although speech signals are nonstationary, their short segments can be considered quasi-stationary. Therefore, Fourier transform can be applied to short segments of speech signals, which results in a sequence of short-time spectra. For speaker recognition, a sequence of features vectors can be extracted from the short-time spectra. One of the popular features for speaker recognition is the cepstral coefficients derived from linear prediction (LP) analysis [38, 39]. These features are commonly referred to as LP-derived cepstral coefficients (LPCCs). One positive property of LPCCs is that they can represent the envelopes of speech spectra. Because the spectral envelopes capture the resonance frequencies, length, and spatially-varying cross-section areas of the vocal tract, LPCCs can represent speaker-dependent characteristics. Another advantage of LPCCs is that they can be computed efficiently.

Apart from LPCCs, mel-frequency cepstral coefficients (MFCCs) [40] are often used in speaker verification systems. MFCCs are obtained by applying DCT to the outputs of a set of nonuniformly spaced filters. The idea comes from the fact that human perception of sound frequencies is nonlinear [37]. Research has shown that MFCCs are effective features for speaker recognition [41].

2.3 GMM classifiers

The state-of-the-art approach to text-independent speaker verification consists in using Gaussian mixture models (GMMs) for modeling client speakers and background speakers [42]. GMMs use semi-parametric techniques for approximating probability density functions (pdf). The output of a GMM is the weighted sum of M mixture densities.

Given B speaker models $\{\Lambda_{\omega_{b_i}}, i = 1, 2, \ldots, B\}$ other than the client model Λ_{ω_c}, i.e., $\Lambda_{\omega_{b_i}} \neq \Lambda_{\omega_c} \forall i$, the log-likelihood of the claimant being an impostor can be found as follows:

$$\log p(X|\Lambda_{\omega_b}) = \frac{1}{B} \sum_{i=1}^{B} \log p(X|\Lambda_{\omega_{b_i}}), \tag{2.1}$$

where

$$\log p(X|\Lambda_{\omega_{b_i}}) = \frac{1}{T}\sum_{t=1}^{T}\log p(\mathbf{x}_t|\Lambda_{\omega_{b_i}}). \tag{2.2}$$

The set of speaker models $\{\Lambda_{\omega_{b_i}}\}$ are known as the cohort set in the literature [44]. Alternatively, a universal background model (UBM) [42] can be created by using the speech of a general population. In such case, $B = 1$ and $\Lambda_{\omega_{b_1}} = \Lambda_{\omega_b}$ is a GMM representing the distribution of speaker-independent features. The universal background model is shared by every client speaker in the system. During verification, given a sequence of vectors X derived from an utterance, the normalized score is calculated as

$$s(X;\Lambda) = \log p(X|\Lambda_{\omega_c}) - \log p(X|\Lambda_{\omega_b}). \tag{2.3}$$

The UBM has the form

$$p(\mathbf{x}_t|\Lambda_{\omega_b}) = \sum_{m=1}^{M} P(\Theta_{m|b}|\omega_b)p(\mathbf{x}_t|\omega_b,\Theta_{m|b}), \tag{2.4}$$

where $p(\mathbf{x}_t|\Lambda_{\omega_b})$ is the GMM's output for input vector $\mathbf{x}_t$, $\Theta_{m|b}$ represents the parameters of the m-th mixture component, M is the number of mixture components, $p(\mathbf{x}_t|\omega_b,\Theta_{m|b}) \equiv \mathcal{N}(\mathbf{x}_t;\mu_{m|b},\Sigma_{m|b})$ is the probability density function of the m-th component, and $P(\Theta_{m|b}|\omega_b)$ is the prior probability (also called mixture coefficients) of the m-th component. Here, we assume that the component density function $p(\mathbf{x}_t|\omega_b,\Theta_{m|b})$ is Gaussian, i.e.,

$$\begin{aligned} p(\mathbf{x}_t|\omega_b,\Theta_{m|b}) &= \mathcal{N}(\mathbf{x}_t;\mu_m,\Sigma_m) \qquad (2.5)\\ &= \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma_m|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}\left[\mathbf{x}_t-\mu_m\right]^T(\Sigma_m)^{-1}\left[\mathbf{x}_t-\mu_m\right]\right\}, \end{aligned}$$

where D is the input dimension and μ_m and Σ_m are the mean vector and covariance matrix of mixture component m, respectively.

The training of UBM-GMMs can be formulated as a maximum-likelihood problem where the mean vectors $\{\mu_m\}$, covariance matrices $\{\Sigma_m\}$, and mixture coefficients $\{P(\Theta_m|\omega)\}$ are typically estimated by the expectation-maximization (EM) algorithm [43]. More specifically, the parameters of a GMM are estimated iteratively by

$$\begin{aligned}
\mu_m^{(j+1)} &= \frac{\sum_{t=1}^{T} P^{(j)}(\Theta_m|\mathbf{x}_t)\mathbf{x}_t}{\sum_{t=1}^{T} P^{(j)}(\Theta_m|\mathbf{x}_t)}, \\
\Sigma_m^{(j+1)} &= \frac{\sum_{t=1}^{T} P^{(j)}(\Theta_m|\mathbf{x}_t)[\mathbf{x}_t - \mu_m^{(j+1)}][\mathbf{x}_t - \mu_m^{(j+1)}]^T}{\sum_{t=1}^{T} P^{(j)}(\Theta_m|\mathbf{x}_t)}, \text{ and} \\
P^{(j+1)}(\Theta_m) &= \frac{\sum_{t=1}^{T} P^{(j)}(\Theta_m|\mathbf{x}_t)}{T}, \qquad (2.6)
\end{aligned}$$

where j denotes the iteration index and $P^{(j)}(\Theta_m|\mathbf{x}_t)$ is the posterior probability of the m-th mixture ($m = 1, \ldots, M$). The latter can be obtained by Bayes' theorem, yielding

$$P^{(j)}(\Theta_m|\mathbf{x}_t) = \frac{P^{(j)}(\Theta_m)p^{(j)}(\mathbf{x}_t|\Theta_m)}{\sum_{k=1}^{M} P^{(j)}(\Theta_k)p^{(j)}(\mathbf{x}_t|\Theta_k)}. \qquad (2.7)$$

Instead of creating the target speaker model Λ_{ω_c} from training patterns of the target speaker using the EM algorithm, in the GMM-UBM system, the target speaker model is obtained by adapting the parameters of the UBM using maxiumum a posteriori (MAP) estimation [42]. Given a UBM Λ_{ω_b} and T independent and identically distributed patterns $X = \{\mathbf{x}_t; t = 1, 2, \ldots, T\}$ from a target speaker, we compute the posterior probability

$$P(i|\mathbf{x}_t) = \frac{\pi_i p_i(\mathbf{x}_t)}{\sum_{j=1}^{M} \pi_j p_j(\mathbf{x}_t)}, \qquad i = 1, \ldots, M. \qquad (2.8)$$

Then, $P(i|\mathbf{x}_t)$ and $\mathbf{x}_t$ are used to compute the sufficient statistics for the mixture coefficients, mean vectors, and covariance matrices:

$$\begin{aligned} n_i &= \sum_{t=1}^{T} P(i|\mathbf{x}_t), \\ E_i(X) &= \frac{1}{n_i}\sum_{t=1}^{T} P(i|\mathbf{x}_t)\mathbf{x}_t, \text{ and} \\ E_i(X^2) &= \frac{1}{n_i}\sum_{t=1}^{T} P(i|\mathbf{x}_t)\mathbf{x}_t^2. \end{aligned} \tag{2.9}$$

These new sufficient statistics from the training data are used to update the old UBM to obtain the adapted parameters of the target speaker model:

$$\begin{aligned} \hat{\pi}_i &= \left[\alpha_i^w n_i/T + (1-\alpha_i^w)\pi_i\right]\gamma, \\ \hat{\mu}_i &= \alpha_i^m E_i(X) + (1-\alpha_i^m)\mu_i, \text{ and} \\ \hat{\sigma}_i^2 &= \alpha_i^v E_i(X^2) + (1-\alpha_i^v)(\sigma_i^2+\mu_i^2) - \hat{\mu}_i^2, \end{aligned} \tag{2.10}$$

where the scale factor, γ, is computed over all adapted mixture weights to ensure that they sum to unity and $\{\alpha_i^w, \alpha_i^m, \alpha_i^v\}$ are the adaptation coefficients controlling the balance between old and new estimates for the mixture coefficients, means and variances, respectively. These coefficients are defined as

$$\alpha_i^\rho = \frac{n_i}{n_i + r^\rho}, \qquad \rho \in \{w, m, v\}, \tag{2.11}$$

where r^ρ is a fixed relevance factor for parameter ρ. Reynold et al. [42] suggested that adapting the means only (i.e. $\alpha_i^w = \alpha_i^v = 0$) and setting α_i^m in Eq. 2.11 to 16 achieve the best performance.

2.4 Channel Compensation

One shortcoming of using LPCCs or MFCCs as features is that the system performance can be easily degraded by the mismatches between training and verification conditions. However, there are several techniques to compensate for the effects of the acoustic mismatches. The most popular one is cepstral mean subtraction (CMS)[45], which approximates a linear channel by the long-term average of distorted cepstral vectors. However, CMS fails to deal with the effect of background noise. To bring the test data closer to the training data, affine transformation [46] can be applied. Both convolutional distortion and additive noise can be compensated simultaneously by using this approach. This subsection explains two feature-based channel compensation algorithms that are based on the idea of affine transformation.

2.4.1 Stochastic Feature Transformation with Handset Detection

In most speaker verification scenarios, speech signals used for verification are codec- and handset-distorted. As a result, there is a spectral mismatch between the enrollment and verification data. A technique called stochastic matching [47] can be used during verification to minimize the mismatch. Stochastic matching is originally designed for speaker adaptation and channel compensation. Its main idea is to transform the distorted data to fit the clean speech models or to transform the clean speech models to better fit the distorted data.

In the case of feature transformation, the channel can be represented by a cepstral bias $\mathbf{b}$, i.e., the transformed vectors is given by

$$\hat{\mathbf{x}}_t = f_\nu(\mathbf{y}_t) = \mathbf{y}_t + \mathbf{b} \tag{2.12}$$

where $\mathbf{y}_t$ is a D-dimensional distorted vector, $\nu = \{b_i\}_{i=1}^{D}$ is the set of transformation parameters, and $f_\nu(\cdot)$ denotes the transformation function. Given distorted speech $\mathbf{y}_t$, $t = 1, \ldots, T$, and an M-center Gaussian mixture model (GMM) $\Lambda_X = \{\pi_j^X, \mu_j^X, \Sigma_j^X\}_{j=1}^{M}$ with mixing coefficients π_j^X, mean vectors μ_j^X and covariance matrices Σ_j^X derived from the clean speech of several speakers (e.g., ten speakers in Mak and Kung [48]), the maximum-likelihood estimates of ν can be obtained by maximizing an auxiliary function

$$\begin{aligned} Q(\nu'|\nu) &= \sum\nolimits_{t=1}^{T} \sum\nolimits_{j=1}^{M} h_j(f_\nu(\mathbf{y}_t)) \cdot \log\left\{\pi_j^X p(\mathbf{y}_t|\mu_j^X, \Sigma_j^X, \nu')\right\} \\ &= \sum\nolimits_{t=1}^{T} \sum\nolimits_{j=1}^{M} h_j(f_\nu(\mathbf{y}_t)) \cdot \log\left\{\pi_j^X p(f_{\nu'}(\mathbf{y}_t)|\mu_j^X, \Sigma_j^X)/|J_{\nu'}(\mathbf{y}_t)|\right\} \end{aligned} \tag{2.13}$$

with respect to ν'. In Eq. 2.13, $\nu' = \{b_i'\}_{i=1}^{D}$ and $\nu = \{b_i\}_{i=1}^{D}$ represent respectively the new and current estimates of the transformation parameters, T is the number of distorted vectors, $f_\nu(\mathbf{y}_t)$ denotes the transformation, $|J_{\nu'}(\mathbf{y}_t)|$ is the determinant of the Jacobian matrix whose (r,s)-th entry is given by $J_{\nu'}(\mathbf{y}_t)_{rs} = \partial f_{\nu'}(\mathbf{y}_t)_r / \partial y_{t,s}$, and $h_j(f_\nu(\mathbf{y}_t))$ is the posterior probability given by

$$h_j(f_\nu(\mathbf{y}_t)) = \frac{\pi_j^X p(f_\nu(\mathbf{y}_t)|\mu_j^X, \Sigma_j^X)}{\sum_{l=1}^{M} \pi_l^X p(f_\nu(\mathbf{y}_t)|\mu_l^X, \Sigma_l^X)}, \tag{2.14}$$

where

$$p(f_\nu(\mathbf{y}_t)|\mu_j^X, \Sigma_j^X) = (2\pi)^{-\frac{D}{2}} |\Sigma_j^X|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(f_\nu(\mathbf{y}_t) - \mu_j^X)(\Sigma_j^X)^{-1}(f_\nu(\mathbf{y}_t) - \mu_j^X)\right\}. \tag{2.15}$$

Assuming diagonal covariance (i.e., $\Sigma_j^X = \text{diag}\{(\sigma_{j1}^X)^2, \ldots, (\sigma_{jD}^X)^2\}$), Eq. 2.13 can be written as

$$Q(\nu'|\nu) = \sum_{t=1}^{T} \sum_{j=1}^{M} h_j(f_\nu(\mathbf{y}_t)) \left\{ -\frac{1}{2} \sum_{i=1}^{D} \frac{(y_{t,i} + b'_i - \mu_{ji}^X)^2}{\left(\sigma_{ji}^X\right)^2} \right\} + C. \qquad (2.16)$$

where C is a constant independent on ν'.

In the M-step of each EM iteration, we maximize $Q(\nu'|\nu)$ with respect to ν' to obtain

$$b'_i = \frac{\sum_{t=1}^{T} \sum_{j=1}^{M} h_j(f_\nu(\mathbf{y}_t))(\sigma_{ji}^X)^{-2}(\mu_{ji}^X - y_{t,i})}{\sum_{t=1}^{T} \sum_{j=1}^{M} h_j(f_\nu(\mathbf{y}_t))(\sigma_{ji}^X)^{-2}} \qquad i = 1, \ldots, D. \qquad (2.17)$$

In this work, the feature transformation was combined with a handset selector [49, 50] for robust speaker verification. Specifically, before verification takes place, we compute one set of transformation parameters for each type of handsets that claimants are likely to use. Then, during a verification session, we identify the most likely handset that is used by the claimant and select the best set of transformation parameters accordingly. Let us denote $\Gamma_k^{(i)}$ as the GMM representing the speech obtained from the k-th handset and the i-th coder. Given an utterance, the most likely handset is selected according to

$$k^* = \arg \max_{k=1}^{H} \sum_{t=1}^{T} \log p(\mathbf{y}_t|\Gamma_k^{(i)}), \qquad (2.18)$$

where H is the number of handsets and $p(\mathbf{y}_t|\Gamma_k^{(i)})$ is the density of the distorted data given the k-th handset and the i-th coder. The transformation parameters corresponding to the k^*-th handset are used to transform the distorted vectors.

2.4.2 Blind Stochastic Feature Transformation

In speaker verification, it is important to ensure that channel variations are suppressed so that the interspeaker distinction can be enhanced. In particular, given a claimant's utterance recorded in an environment different from that during enrollment, one aims to transform the features of the utterance so that they become compatible with the enrollment environment. The stochastic feature transformation with handset detection described in the preceding subsection achieves this objective by assuming that (1) the effect of background noise is negligible, (2) the handset and channel are linear, and (3) the handset used by a claimant can be detected as one of the known handsets in a handset database. Once the handset model is identified, a priori knowledge about the identified handset can be used to shift the distorted features in the cepstral domain. Although this approach has shown to be effective for both handset- and coder-distorted speech [51, 52], it has two drawbacks. First, from the theoretical perspective, the adverse effect of the linearity assumption limits the performance improvement. Second, from the practical perspective, the requirement of handset detection gives rise to practicality problems. It will be much more practical and cost effective if handset detector-free systems are adopted.

This subsection describes a feature-based *blind* transformation approach to overcome these drawbacks. The transformation is blind in that it compensates the handset distortion without a priori information about the handset's characteristics. Hereafter, this transformation approach is referred to as blind stochastic feature transformation (BSFT) [53, 54].

Two Phases of BSFT

Figure 2.2 illustrates a speaker verification system with BSFT, whose operations are divided into two separate phases: enrollment and verification.

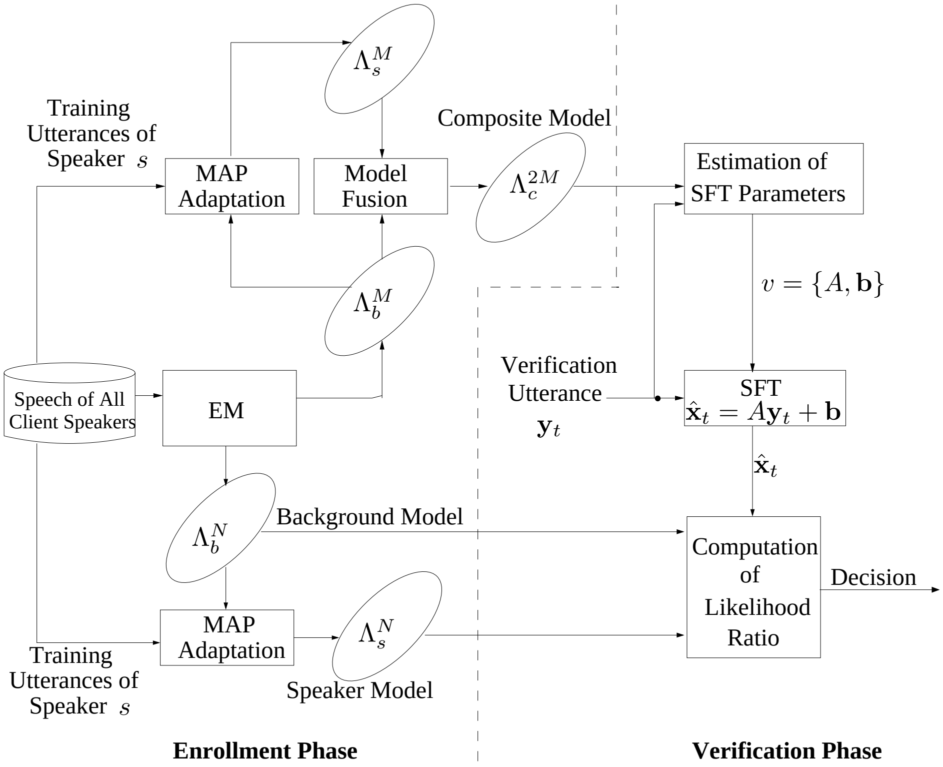


Figure 2.2: Estimation of BSFT parameters. The background model Λ_b^N, speaker model Λ_s^N, and composite model Λ_c^{2M} produced during the enrollment phase, are subsequently used for verification purposes.

Enrollment Phase. The speech of all client speakers are used to create a compact universal background model (UBM) Λ_b^M with M components. Then, for each client speaker, a compact speaker model Λ_s^M is created by adapting the UBM Λ_b^M using maximum a posteriori (MAP) adaptation [42]. Because verification decisions are based on the likelihood of the speaker model and background model, both models must be considered when the transformation parameters are computed. This can be achieved by fusing Λ_b^M and Λ_s^M to form a $2M$-component composite GMM Λ_c^{2M}. During the fusion process, the means and covariances remain unchanged but the value of each mixing coefficient is divided by 2. This step ensures that the output of the composite GMM represents a probability density function.

Verification Phase. Distorted features $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}$ extracted from a verification utterance are used to compute the transformation parameters $\nu = \{A, \mathbf{b}\}$. This is achieved by maximizing the likelihood of the composite GMM Λ_c^{2M} given the transformed features $\hat{X} = \{\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_T\}$:

$$\hat{\mathbf{x}}_t = f_\nu(\mathbf{y}_t) = A\mathbf{y}_t + \mathbf{b}, \quad t = 1, \ldots, T, \tag{2.19}$$

where A is a $D \times D$ identity matrix for zeroth-order transformation and $A = \text{diag}\{a_1, a_2, \ldots, a_D\}$ for first-order transformation, and $\mathbf{b}$ is a bias vector.[1] The transformed vectors $\hat{X}$ are then fed to a full size speaker model Λ_s^N and a full size UBM Λ_b^N for computing verification scores in terms of log-likelihood ratio:

$$s(\hat{X}) = \log p(\hat{X}|\Lambda_s^N) - \log p(\hat{X}|\Lambda_b^N).$$

The main idea of BSFT is to transform the distorted features to fit the composite GMM Λ_c^{2M}, which ensures that the transformation compensates the acoustic distortion. Compared with Eq. 2.12, the transformation in Eq. 2.19 is more flexible because the inclusion of the matrix A allows changing the variances of the transformed vectors, which is detrimental to the compensation of background noise and nonlinear convolutive distortion in the time domain (see [55] for the effect of background noise and nonlinear distortion on ceptral vectors). From the practical point of view, the algorithm is suitable for a broader scale and more economical deployment than the conventional approaches because it does not require a handset detector.

[1] see Kung, Mak, and Lin [55] for the computation of A and $\mathbf{b}$.

2.5 Performance Evaluation

Typically, verification is a two-class problem: either the claimant is the one (client) or is not the one (impostor) he or she claims to be. Therefore, the system needs to either accept or reject the claimant. For this kind of hypothesis testing problem, two types of errors can be made:

1. **False rejection (FR):** a registered client is rejected.

2. **False acceptance (FA):** an impostor is accepted.

As a result, the false rejection rate (FRR) and the false acceptance rate (FAR) can be used to evaluate the performance of speaker verification systems. These error rates are obtained as follows:

$$\mathrm{FRR} = \frac{\text{number of FRs}}{\text{number of true speaker trials}} \times 100\% \tag{2.20}$$

$$\mathrm{FAR} = \frac{\text{number of FAs}}{\text{number of impostor attempts}} \times 100\% \tag{2.21}$$

Because it might be difficult to compare the performance of systems using two numbers, the half total error rate (HTER, i.e., $\frac{\mathrm{FAR} + \mathrm{FRR}}{2}$) and the equal error rate (EER, i.e., FAR = FRR) are often used.

Because users may adjust the decision threshold to set the FRR or FAR to a desirable level, system performance is often characterized by two graphs: receiver operating characteristic (ROC) [56] and detection error trade-off (DET) [57]. Both graphs represent FRR as a function of FAR. The main difference between ROC and DET is their scale. ROC uses linear scales and DET, on the contrary, uses nonlinear scales. It is difficult to judge which method is better, but DET is more preferable when the systems to be compared have very good performance [57]. Therefore, EER and DET plots were used as a performance measure in this work.

Chapter 3

Multisample Speaker Verification

Although decision fusion is mainly applied to combine the outputs of modality-dependent classifiers, it can also be applied to fuse the decisions or scores from a single modality. The idea is to consider the multiple samples extracted from a single modality as independent but coming from the same source. The approach is commonly referred to as multisample fusion [58]. This chapter investigates the fusion of scores from multiple utterances to improve the performance of speaker verification from GSM-transcoded speech. Fusion weights for individual scores are based on the score distribution of the utterances and on the prior score statistics determined from enrollment data. We refer to this type of fusion as data-dependent decision fusion. To further enhance the robustness of the proposed fusion algorithm, the prior score statistics are adapted during verification based on the probability that the claimant is an impostor. Because the variation in handset characteristics and the encoding/decoding process introduce substantial distortion to the speech signals [59], stochastic feature transformation [48] is also applied to the feature vectors extracted from the GSM-transcoded speech before presenting them to the clean speaker model and background model.

3.1 Data-Dependent Decision Fusion

3.1.1 Architecture

Assume that K streams of features vectors (e.g., MFCCs) can be extracted from K independent utterances $\mathcal{U} = \{\mathcal{U}_1, \ldots, \mathcal{U}_K\}$. Let us denote the observation sequence corresponding to utterance $\mathcal{U}_k$ by

$$\mathcal{O}^{(k)} = \{\mathbf{o}_t^{(k)} \in \Re^D; t = 1, \ldots, T_k\}, \qquad k = 1, \ldots, K \tag{3.1}$$

where D and T_k are the dimensionality of $\mathbf{o}_t^{(k)}$ and the number of observations in $\mathcal{O}^{(k)}$, respectively. We further define a normalized score function

$$s(\mathbf{o}_t^{(k)}; \Lambda) \equiv \log p(\mathbf{o}_t^{(k)}|\Lambda_{\omega_c}) - \log p(\mathbf{o}_t^{(k)}|\Lambda_{\omega_b}), \tag{3.2}$$

where $\Lambda = \{\Lambda_{\omega_c}, \Lambda_{\omega_b}\}$ contains the Gaussian mixture models (GMMs) characterizing the client speaker (ω_c) and the background speakers (ω_b), and $\log p(\mathbf{o}_t^{(k)}|\Lambda_\omega)$ is the output of Λ_ω, $\omega \in \{\omega_c, \omega_b\}$, given observation $\mathbf{o}_t^{(k)}$.

In Mak et al. [60], frame-level fused scores are computed as

$$s(\mathbf{o}_t^{(1)}, \ldots, \mathbf{o}_t^{(K)}; \Lambda) = s(\mathbf{O}_t; \Lambda) = \sum_{k=1}^{K} \alpha_t^{(k)} s(\mathbf{o}_t^{(k)}; \Lambda) \text{ with } \sum_{k=1}^{K} \alpha_t^{(k)} = 1, \tag{3.3}$$

where $\mathbf{O}_t = \{\mathbf{o}_t^{(1)}, \ldots, \mathbf{o}_t^{(K)}\}$ contains the K observations from the K utterances at time t and $\alpha_t^{(k)} \in [0, 1]$ represents the confidence (reliability) of the observation $\mathbf{o}_t^{(k)}$. The mean fused score

$$s(\mathcal{U}; \Lambda) = \frac{1}{T} \sum_{t=1}^{T} s(\mathbf{O}_t; \Lambda) \tag{3.4}$$

is compared against a decision threshold for decision making. Figure 3.1 depicts the architecture of the fusion model.

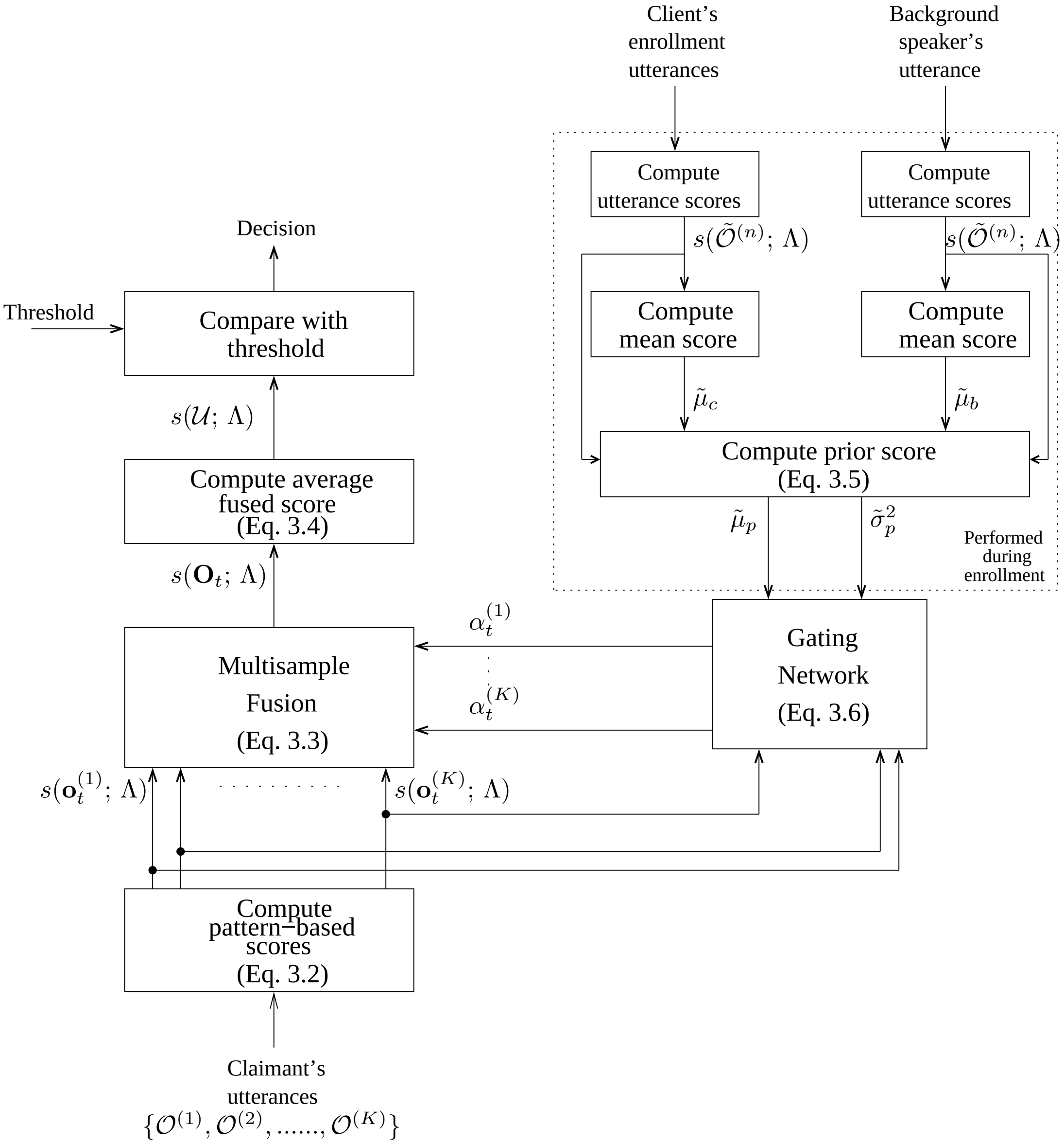

Client's enrollment utterances
Background speaker's utterance
Compute utterance scores
Compute utterance scores
$s(\tilde{\mathcal{O}}^{(n)};\,\Lambda)$
$s(\tilde{\mathcal{O}}^{(n)};\,\Lambda)$
Compute mean score
Compute mean score
$\tilde{\mu}_c$
$\tilde{\mu}_b$
Compute prior score (Eq. 3.5)
$\tilde{\mu}_p$
$\tilde{\sigma}_p^2$
Performed during enrollment
Decision
Threshold
Compare with threshold
$s(\mathcal{U};\,\Lambda)$
Compute average fused score (Eq. 3.4)
$s(\mathbf{O}_t;\,\Lambda)$
Multisample Fusion (Eq. 3.3)
$\alpha_t^{(1)}$
$\alpha_t^{(K)}$
Gating Network (Eq. 3.6)
$s(\mathbf{o}_t^{(1)};\,\Lambda)$
$s(\mathbf{o}_t^{(K)};\,\Lambda)$
Compute pattern–based scores (Eq. 3.2)
Claimant's utterances
$\{\mathcal{O}^{(1)}, \mathcal{O}^{(2)}, ..., \mathcal{O}^{(K)}\}$

Figure 3.1: Architecture of the multisample decision fusion model.

In Eq. 3.3, a larger (respectively smaller) fusion weight means a greater (respectively lesser) influence on the final decision. Fusion weights can be estimated using training data; alternatively, they can be determined purely from the observation data during recognition. Rather than using either training data or recognition data exclusively, Mak, Cheung, and Kung [60] proposed a new approach in which the fusion weights depend on both training data (prior information) and recognition data. Specifically, during enrollment, the mean score of each client speaker ($\tilde{\mu}_c$) and of the background speakers ($\tilde{\mu}_b$) are determined. Then, a prior score and a prior variance are computed as follows:

$$\tilde{\mu}_p = \frac{K_c\tilde{\mu}_c + K_b\tilde{\mu}_b}{K_c + K_b} \qquad \text{and} \qquad \tilde{\sigma}_p^2 = \frac{1}{K_c + K_b}\sum_{n=1}^{K_c+K_b}\left[s(\tilde{\mathcal{O}}^{(n)};\Lambda) - \tilde{\mu}_p\right]^2, \tag{3.5}$$

where $s(\tilde{\mathcal{O}}^{(n)};\Lambda) = \frac{1}{T_n}\sum_{t=1}^{T_n} s(\tilde{\mathbf{o}}_t^{(n)};\Lambda)$ is the mean score of the n-th training utterance and K_c and K_b are the numbers of client speaker's utterances and background speakers' utterances, repectively. Then, during verification, the claimant is asked to utter K utterances, and the data-dependent fusion weights are computed as:

$$\alpha_t^{(k)} = \frac{\exp\left\{[s_t^{(k)} - \tilde{\mu}_p]^2 \Big/ 2\tilde{\sigma}_p^2\right\}}{\sum_{l=1}^{K}\exp\left\{[s_t^{(l)} - \tilde{\mu}_p]^2 \Big/ 2\tilde{\sigma}_p^2\right\}} \qquad k = 1, \ldots, K, \tag{3.6}$$

where for ease of presentation, we have defined $s_t^{(k)} \equiv s(\mathbf{o}_t^{(k)};\Lambda)$.

Note that this method requires the K utterances to contain the same number of feature vectors, i.e., $T_k = T \;\forall\; k = 1, \ldots, K$. If not, the tail of the longer utterances can be appended to the tail of the shorter ones to make the number of vectors in all utterances equal.[1]

[1] Because it is likely that the utterances are obtained from the same speaker under the same environment in a verification session, moving feature vectors from utterances to utterances will have the same effect as partitioning a long utterance into several equal-length short utterances.

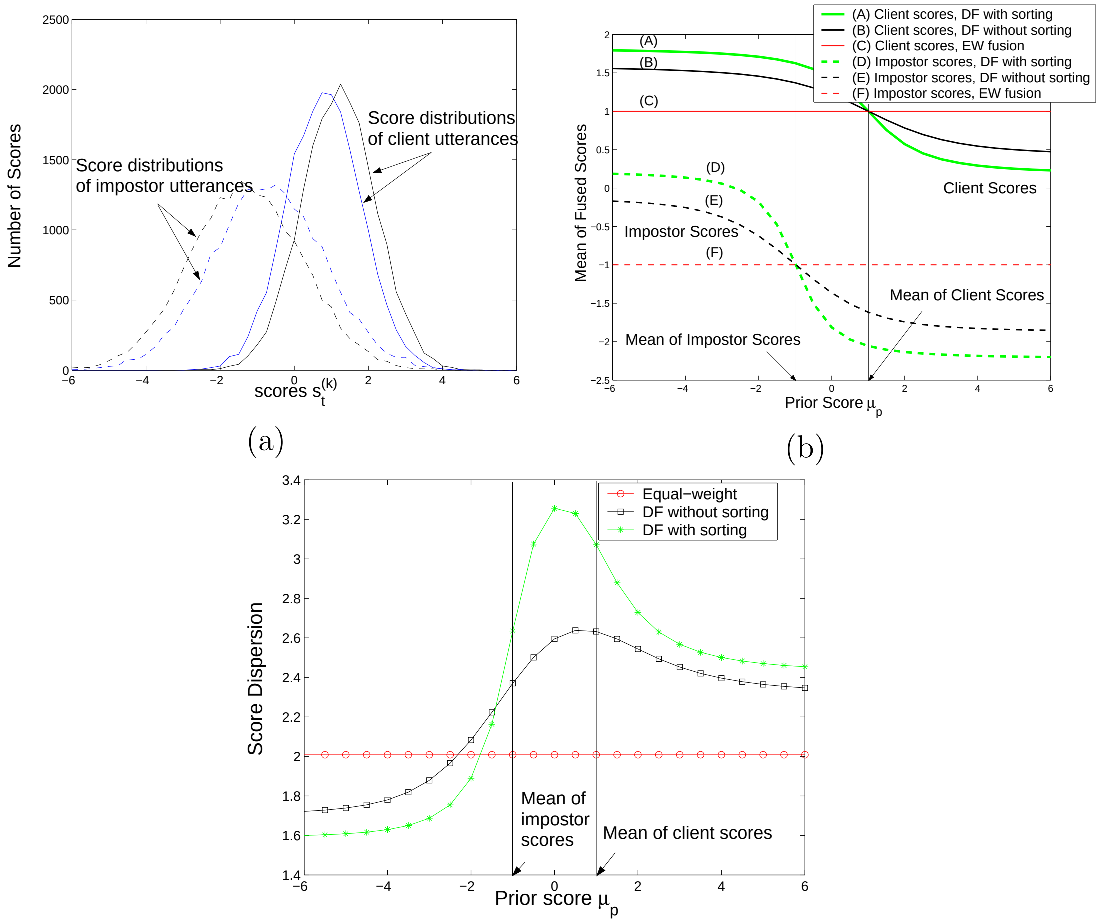
3.1.2 Gaussian Example

Figure 3.2 illustrates an example in which the distributions of the client-speaker scores and impostor scores are assumed to be Gaussian. It is also assumed that both the client and the impostor utter two utterances. The client speaker's mean scores for the first and second utterances are equal to 1.2 and 0.8, respectively. Likewise, the impostor's mean scores for the two utterances are equal to -1.3 and -0.7. Obviously, equal-weight fusion will produce a mean speaker score of 1.0 and a mean impostor score of -1.0, resulting in a score dispersion of 2.0. These two mean scores (-1.0 and 1.0) are indicated by the two vertical lines in Figure 3.2(b). We can see from Figures 3.2(b) and 3.2(c) that when the prior score $\tilde{\mu}_p$ is set to a value between these two means (i.e., between the vertical lines), the data-dependent fusion algorithm can produce a score dispersion larger than 2.0. Because the mean of fused score is used to make the final decision, increasing the score dispersion can decrease speaker verification error rates.

3.2 Theoretical Analysis

This section provides a theoretical analysis of the fusion algorithm. The analysis aims to explain how and why the fusion algorithm achieves better performance than the equal-weight fusion approach. The reason behind the increase in the score dispersion in Figure 3.2 is also explained.

We consider the case where the score sequences of two independent utterances are fused, i.e., $K = 2$ in Eq. 3.3. The extension to multiple sequences is trivial. Because the two utterances are independent, their scores $s_t^{(1)}$ and $s_t^{(2)}$ are also independent. Differentiating both side of Eq. 3.6 with respect to $s_t^{(k)}$ and using the independence between $s_t^{(1)}$ and $s_t^{(2)}$, we obtain



(a)

(b)

(c)

Figure 3.2: (a) Distributions of client scores and impostor scores as a result of four utterances: two from a client speaker and another two from an impostor. The means of client scores are 0.8 and 1.2, and the means of impostor scores are −1.3 and −0.7. (b) The mean of fused client scores and the mean of fused impostor scores versus the prior score $\tilde{\mu}_p$. (c) Difference between the mean of fused client scores and the mean of fused impostor scores under different values of prior scores $\tilde{\mu}_p$'s based on equal-weight (EW) fusion and data-dependent fusion (DF) with and without score sorting.

$$
\begin{aligned}
\frac{\partial \alpha_t^{(k)}}{\partial s_t^{(k)}} &= \frac{\left(\sum_l e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right) \frac{s_t^{(k)}-\tilde{\mu}_p}{\tilde{\sigma}_p^2} \left(e^{\{(s_t^{(k)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)}{\left(\sum_l e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)^2} \\
&\quad - \frac{\left(e^{\{(s_t^{(k)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right) \frac{s_t^{(k)}-\tilde{\mu}_p}{\tilde{\sigma}_p^2} \left(e^{\{(s_t^{(k)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)}{\left(\sum_l e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)^2} \\
&= \frac{s_t^{(k)}-\tilde{\mu}_p}{\tilde{\sigma}_p^2} e^{\{(s_t^{(k)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} \left[\frac{\left(\sum_l e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right) - \left(e^{\{(s_t^{(k)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)}{\left(\sum_l e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)^2}\right] \\
&= \frac{s_t^{(k)}-\tilde{\mu}_p}{\tilde{\sigma}_p^2} e^{\{(s_t^{(k)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} \left[\frac{\left(\sum_{l \neq k} e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)}{\left(\sum_l e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)^2}\right] \\
&= \frac{C\left(s_t^{(k)}-\tilde{\mu}_p\right)}{\tilde{\sigma}_p^2} \left[\frac{e^{\{(s_t^{(k)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}}{\left(\sum_{l=1}^{2} e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right)^2}\right] \qquad k = 1, 2. \qquad (3.7)
\end{aligned}
$$

where $C = \sum_{l \neq k} e^{\{(s_t^{(l)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} > 0$. Eq. 3.7 suggests that when $s_t^{(k)} > \tilde{\mu}_p$, $\partial \alpha_t^{(k)}/\partial s_t^{(k)} > 0$, and vice versa for $s_t^{(k)} < \tilde{\mu}_p$.

Let us consider two scenarios:

Scenario A: $\tilde{\mu}_p < \mu$, where μ is the mean score of the two utterances. In this scenario (see Figure 3.3), the claimant is more likely to be a client speaker than an impostor because the two utterances produce many large pattern-based scores to make $\mu > \tilde{\mu}_p$; for example, $\mu = 1$ for the two client utterances in Figure 3.2(a). Because the majority of the pattern-based scores ($s_t^{(k)}$ and $s_t^{(l)}$, which are any two scores from the verification utterance l and k, respectively.) are large, we have the following conditions:

Condition A-1 (Figure 3.3(a))**:**

$$P(s_t^{(k)} > \tilde{\mu}_p) > P(s_t^{(k)} < \tilde{\mu}_p) \quad k \in \{1, 2\}$$

Condition A-2 (Figure 3.3(b))**:**

$$
\begin{aligned}
& P(\text{emphasizing large scores}) \\
= \; & P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3) \\
= \; & P(\{(s_t^{(k)} > \tilde{\mu}_p) \, \cap \, (s_t^{(l)} > \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \, \cap \, (s_t^{(l)} < \tilde{\mu}_p) \, \cap \, (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \, \cap \, (s_t^{(l)} > \tilde{\mu}_p) \, \cap \, (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\}) \\
= \; & P(\{(s_t^{(k)} > \tilde{\mu}_p) \, \cap \, (s_t^{(l)} > \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \, \cap \, (s_t^{(l)} < \tilde{\mu}_p) \, \cap \, (s_t^{(k)} + s_t^{(l)} > 2\tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \, \cap \, (s_t^{(l)} > \tilde{\mu}_p) \, \cap \, (2\tilde{\mu}_p < s_t^{(k)} + s_t^{(l)})\}) \\
> \; & P(\{(s_t^{(k)} < \tilde{\mu}_p) \, \cap \, (s_t^{(l)} < \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \, \cap \, (s_t^{(l)} < \tilde{\mu}_p) \, \cap \, (s_t^{(k)} + s_t^{(l)} < 2\tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \, \cap \, (s_t^{(l)} > \tilde{\mu}_p) \, \cap \, (2\tilde{\mu}_p > s_t^{(k)} + s_t^{(l)})\}) \\
& \quad \text{from Condition A-1} \\
= \; & P(\{(s_t^{(k)} < \tilde{\mu}_p) \, \cap \, (s_t^{(l)} < \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \, \cap \, (s_t^{(l)} < \tilde{\mu}_p) \, \cap \, (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \, \cap \, (s_t^{(l)} > \tilde{\mu}_p) \, \cap \, (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}) \\
= \; & P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6) \\
= \; & P(\text{emphasizing small scores})
\end{aligned}
$$

$P(\mathcal{S})$ in Condition A-2 stands for the probability of having the scores fall on the set $\mathcal{S}$, and its value can be obtained by integrating the 2-D Gaussian density function over the region defined by $\mathcal{S}$. Because the peak of the 2-D density function falls on $\mathcal{S}1$ (see Figure 3.3(b)), the volume under $\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3$ should be larger than that under $\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6$. This observation agrees with the inequality in Condition A-2.

The above argument shows that $P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3) > P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6)$. Here, we explain why $P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3)$ is the probability of emphasizing large scores and $P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6)$ is the probability of emphasizing small scores.[2] In $\mathcal{S}1$, because both $s_t^{(k)}$ and $s_t^{(l)}$ are larger than the prior score $\tilde{\mu}_p$, Eq. 3.6 will emphasize the larger score only. In $\mathcal{S}2$, although $s_t^{(l)}$ is smaller than the prior score $\tilde{\mu}_p$, Eq. 3.6 still emphasizes the larger score (i.e., $s_t^{(k)}$ in this set) because the difference between $s_t^{(k)}$ and $\tilde{\mu}_p$ is larger than that between $s_t^{(l)}$ and $\tilde{\mu}_p$. The situation in $\mathcal{S}3$ is similar to that in $\mathcal{S}2$, except that the large score is $s_t^{(l)}$ and the small score is $s_t^{(k)}$; again the larger score $s_t^{(l)}$ is emphasized because it is further away from $\tilde{\mu}_p$ than $s_t^{(k)}$ is. Therefore, by merging these three sets together, we can obtain the probability of emphasizing large scores. Similar arguments can be applied to $\mathcal{S}4$, $\mathcal{S}5$, and $\mathcal{S}6$ to obtain the probability of emphasizing the small scores.

Scenario A suggests that when the majority of scores are greater than the prior score $\tilde{\mu}_p$, the fusion algorithm has a higher chance of emphasizing large scores. Meanwhile, Eq. 3.7 suggests that if the score $s_t^{(k)}$ increases, the fusion weight $\alpha_t^{(k)}$ for $s_t^{(k)}$ will also increase (because $\partial\alpha_t^{(k)}/\partial s_t^{(k)} > 0$). These two observations suggest that the mean fused score is more likely to be larger than the mean scores of the two utterances.[3]

Scenario B: $\tilde{\mu}_p > \mu$, where μ is the mean score of the two utterances. In this scenario (see Figure 3.4), the claimant is more likely to be an impostor because the two utterances produce many small pattern-based scores to make $\mu < \tilde{\mu}_p$;

[2] Emphasizing large scores means that larger weights are assigned to the large scores, and emphasizing small scores means that larger weights are assigned to the small scores.

[3] For a proof, please refer to Appendix A.

for example, $\mu = -1$ for the two impostor utterances in Figure 3.2(a). Because the majority of the pattern-based scores ($s_t^{(k)}$ and $s_t^{(l)}$) are small, we have the following conditions:

Condition B-1 (Figure 3.4(a))**:**

$$P(s_t^{(k)} < \tilde{\mu}_p) > P(s_t^{(k)} > \tilde{\mu}_p) \quad k \in \{1, 2\}$$

Condition B-2 (Figure 3.4(b))**:**

$$\begin{aligned}
& P(\text{emphasizing small scores}) \\
= \; & P(\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6) \\
= \; & P(\{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}) \\
> \; & P(\{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\} \cup \\
& \quad \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\} \cup \\
& \quad \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\}) \\
& \quad \text{from Condition B-1} \\
= \; & P(\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3) \\
= \; & P(\text{emphasizing large scores})
\end{aligned}$$

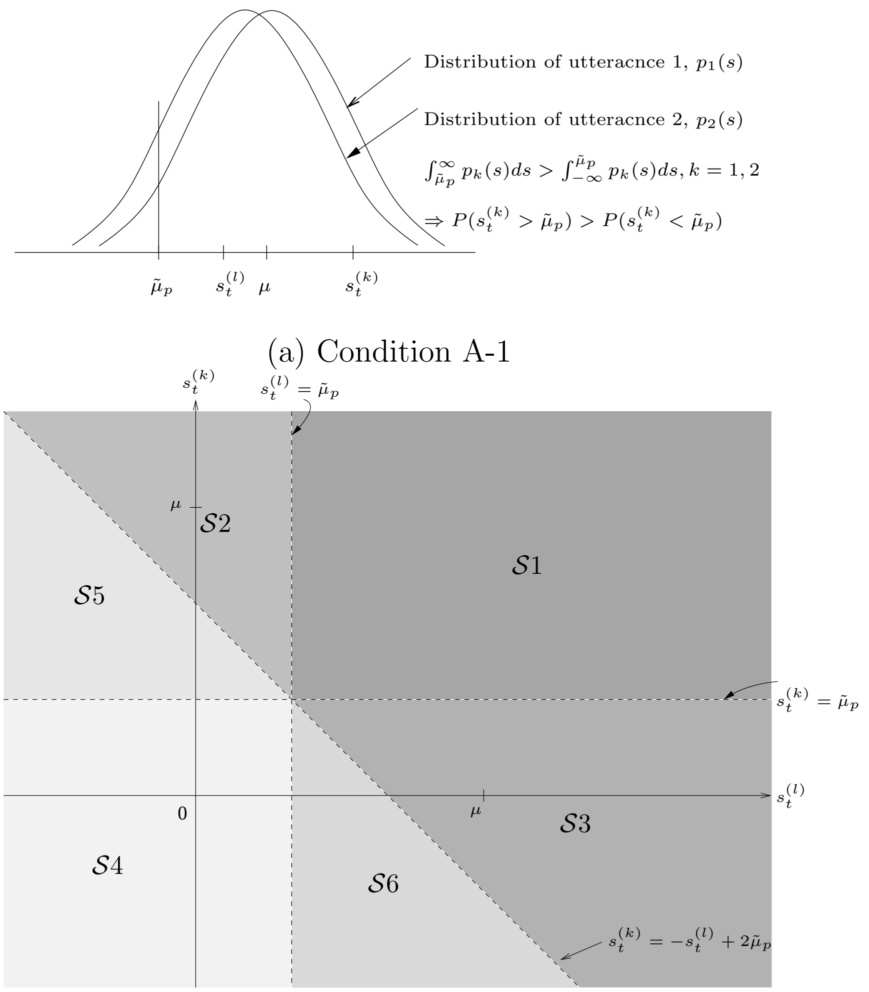
In Condition B-2, $P(\mathcal{S})$ stands for the probability of having the scores fall on the set $\mathcal{S}$, and its value can be obtained by integrating the 2-D Gaussian density function over the region defined by $\mathcal{S}$. Because the peak of the 2-D density function falls on $\mathcal{S}4$ (see Figure 3.4(b)), the volume under $\mathcal{S}4 \cup \mathcal{S}5 \cup \mathcal{S}6$ should be larger than that under $\mathcal{S}1 \cup \mathcal{S}2 \cup \mathcal{S}3$. This observation agrees with the inequality in Condition B-2.

Scenario B suggests that when the majority of scores are smaller than the prior score $\tilde{\mu}_p$, the fusion algorithm has a higher chance of emphasizing small scores. We can also observe from Eq. 3.7 that the score $s_t^{(k)}$ decreases, the fusion weight $\alpha_t^{(k)}$ for $s_t^{(k)}$ will increase (because $\partial\alpha_t^{(k)}/\partial s_t^{(k)} < 0$). These two observations lead to the conclusion that the mean fused score of the two utterances is more likely to be smaller than the utterances' mean score.

The preceding analysis suggests that if the claimant is more likely to be a client speaker, the fusion algorithm will increase his or her mean fused score and vice versa if he or she is an impostor. This has the effect of increasing the score dispersion, as demonstrated in Figure 3.2(c).

The regions of emphasizing large scores and small scores illustrated in Figures 3.3 and 3.4 can be highlighted by plotting $\alpha_t^{(k)}$ against $s_t^{(k)}$ and $s_t^{(l)}$. Figure 3.5(a) illustrates the fusion weights $\alpha_t^{(1)}$ as a function of $s_t^{(1)}$ and $s_t^{(2)}$ where $s_t^{(k)} \in [-12, 12]$, $k \in \{1, 2\}$, $\tilde{\mu}_p = -2$, and $\tilde{\sigma}_p = 3.5$. A closer look at Figure 3.5(a) reveals that scores falling on the upper-right region of the dashed line L will be increased by the fusion function Eq. 3.3. This is because in that region, for $s_t^{(1)} > s_t^{(2)}$, $\alpha_t^{(1)} \approx 1$ and $\alpha_t^{(2)} \approx 0$; moreover, for $s_t^{(1)} < s_t^{(2)}$, $\alpha_t^{(1)} \approx 0$ and $\alpha_t^{(2)} \approx 1$. Both of these conditions make Eq. 3.3 to emphasize the larger score. On the other hand, the fusion algorithm will put more emphasis on the small scores if they fall on the lower-left region of the dashed line L. The effect of the fusion weights on the scores is depicted in Figure 3.5(b). Evidently, the fusion weights will favor large scores if they fall on the upper-right region, whereas the fused scores will be close to the small scores if they fall on the lower-left region.

The rationale behind this fusion approach is the observation that most of the

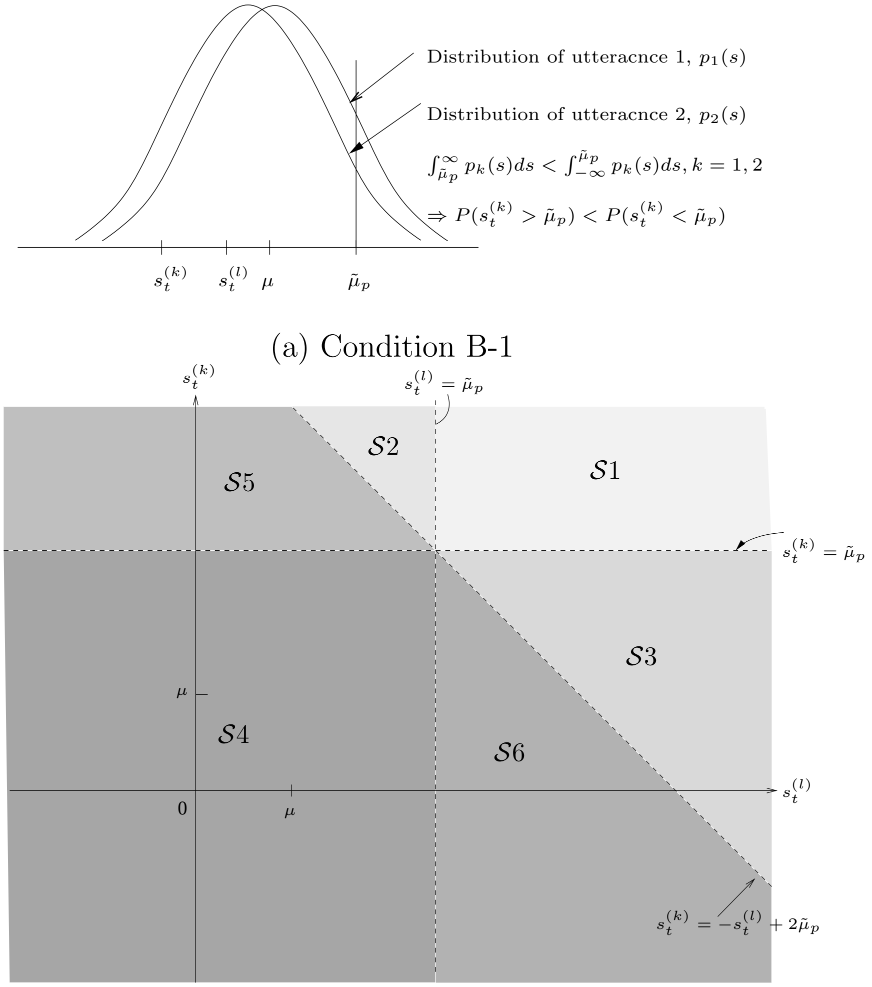

Distribution of utteracnce 1, $p_1(s)$
Distribution of utteracnce 2, $p_2(s)$
$\int_{\tilde{\mu}_p}^{\infty} p_k(s)ds > \int_{-\infty}^{\tilde{\mu}_p} p_k(s)ds, k = 1, 2$
$\Rightarrow P(s_t^{(k)} > \tilde{\mu}_p) > P(s_t^{(k)} < \tilde{\mu}_p)$
$\tilde{\mu}_p$
$s_t^{(l)}$
μ
$s_t^{(k)}$
$s_t^{(k)}$
$s_t^{(l)} = \tilde{\mu}_p$
μ
$\mathcal{S}2$
$\mathcal{S}1$
$\mathcal{S}5$
$s_t^{(k)} = \tilde{\mu}_p$
$s_t^{(l)}$
0
μ
$\mathcal{S}3$
$\mathcal{S}4$
$\mathcal{S}6$
$s_t^{(k)} = -s_t^{(l)} + 2\tilde{\mu}_p$

(a) Condition A-1

$$
\begin{aligned}
\mathcal{S}1 &= \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\} \\
\mathcal{S}2 &= \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\} \\
\mathcal{S}3 &= \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\} \\
\mathcal{S}4 &= \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\} \\
\mathcal{S}5 &= \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\} \\
\mathcal{S}6 &= \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}
\end{aligned}
$$

(b) Condition A-2

Figure 3.3: Illustration of the two conditions in Scenario A ($\tilde{\mu}_p < \mu$). (a) $P(s_t^{(k)} > \tilde{\mu}_p) > P(s_t^{(k)} < \tilde{\mu}_p)$; (b) P(emphasizing large scores) > P(emphasizing small scores).

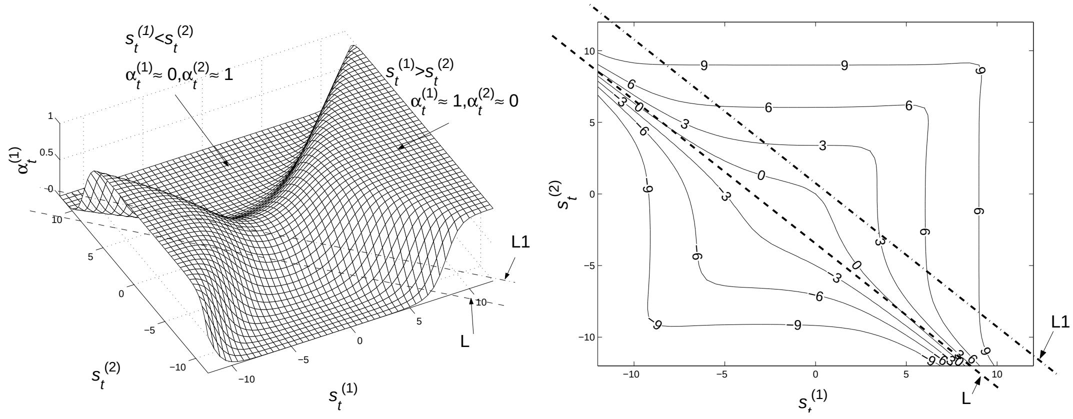

Distribution of utteracnce 1, $p_1(s)$
Distribution of utteracnce 2, $p_2(s)$
$\int_{\tilde{\mu}_p}^{\infty} p_k(s)ds < \int_{-\infty}^{\tilde{\mu}_p} p_k(s)ds, k = 1, 2$
$\Rightarrow P(s_t^{(k)} > \tilde{\mu}_p) < P(s_t^{(k)} < \tilde{\mu}_p)$
$s_t^{(k)}$
$s_t^{(l)}$
μ
$\tilde{\mu}_p$
$s_t^{(k)}$
$s_t^{(l)} = \tilde{\mu}_p$
$\mathcal{S}2$
$\mathcal{S}1$
$\mathcal{S}5$
$s_t^{(k)} = \tilde{\mu}_p$
$\mathcal{S}3$
μ
$\mathcal{S}4$
$\mathcal{S}6$
$s_t^{(l)}$
0
μ
$s_t^{(k)} = -s_t^{(l)} + 2\tilde{\mu}_p$

(a) Condition B-1

$\mathcal{S}1 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p)\}$
$\mathcal{S}2 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p > \tilde{\mu}_p - s_t^{(l)})\}$
$\mathcal{S}3 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} < s_t^{(l)} - \tilde{\mu}_p)\}$
$\mathcal{S}4 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p)\}$
$\mathcal{S}5 = \{(s_t^{(k)} > \tilde{\mu}_p) \cap (s_t^{(l)} < \tilde{\mu}_p) \cap (s_t^{(k)} - \tilde{\mu}_p < \tilde{\mu}_p - s_t^{(l)})\}$
$\mathcal{S}6 = \{(s_t^{(k)} < \tilde{\mu}_p) \cap (s_t^{(l)} > \tilde{\mu}_p) \cap (\tilde{\mu}_p - s_t^{(k)} > s_t^{(l)} - \tilde{\mu}_p)\}$

(b) Condition B-2

Figure 3.4: Illustration of the two conditions in Scenario B ($\tilde{\mu}_p > \mu$). (a) $P(s_t^{(k)} > \tilde{\mu}_p) < P(s_t^{(k)} < \tilde{\mu}_p)$; (b) P(emphasizing small scores) $> P$(emphasizing large scores).



(a) (b)

Figure 3.5: (a) Fusion weights $\alpha_t^{(1)}$ as a function of scores $s_t^{(1)}$ and $s_t^{(2)}$. (b) Contour plot of fused scores based on the fusion formula Eq. 3.3 and the fusion weights in (a).

client-speaker scores are larger than the prior score, but most of the impostor scores are smaller than the prior score. As a result, if the claimant is a client speaker, the fusion algorithm will assign large weights to the large scores; on the other hand, the algorithm will weight small scores more heavily if the claimant is an impostor. This has the effect of reducing the overlapping area of the score distribution of the client speakers and the impostors, thus reducing the error rate.

3.3 Fusion of Sorted Scores

Because the proposed fusion algorithm depends on the pattern-based scores of individual utterances, the positions of scores in the score sequence also affect the final fused scores. Moreover, as illustrated in Section 3.2, the emphasis of large speaker scores under Scenario A and the de-emphasis of small impostor scores under Scenario B are probabilistic, i.e., there is no guarantee that these situations will always occur. To overcome this limitation, Cheung, Mak, and Kung [61] proposed to sort the scores before fusion so that small scores will always be fused with large scores.

3.3.1 Theoretical Analysis

Here, we provide a theoretical analysis to explain the benefit of sorting the scores before fusion. We assume that there are two sorted score sequences ($s_t^{(1)}$ and $s_t^{(2)}$) with equal mean (μ), $s_t^{(1)}$ being arranged in ascending order and $s_t^{(2)}$ in descending order. We further assume that the scores in the sequences follow a Gaussian distribution. If the numbers of scores in the sequences are sufficiently large, we can obtain the following relationship:[4]

$$\mu - s_t^{(1)} \approx s_t^{(2)} - \mu, \quad \text{i.e.,} \quad s_t^{(2)} \approx 2\mu - s_t^{(1)}, \tag{3.8}$$

where $s_t^{(1)}$ and $s_t^{(2)}$ represent the scores less than and greater than the score mean μ, respectively. When $\mu = 1$, Eq. 3.8 represents the straight line $L1$ in Figure 3.5(b). Evidently, Line $L1$ lies in the region where large scores are emphasized. As a result, an increase in the mean fused score can be guaranteed.

Without loss of generality, we denote the smaller score as $s_t^{(1)}$ and the larger one as $s_t^{(2)}$, i.e., $s_t^{(1)} < s_t^{(2)}$. Substituting Eq. 3.8 into Eq. 3.6, the fusion weight for the small scores $s_t^{(1)}$ can be expressed as

$$\begin{aligned} \alpha_t^{(1)} &= \frac{\exp\{(s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}{\sum_{l=1}^{2}\exp\{(s_t^{(l)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} \\ &= \frac{\exp\{(s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}{\exp\{(s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\} + \exp\{(2\mu - s_t^{(1)} - \tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}. \end{aligned} \tag{3.9}$$

[4] The relationship is obtained by assuming that the two utterances have the same mean and variance.

Differentiate both side of Eq. 3.9 with respect to $s_t^{(1)}$, we obtain

$$
\begin{aligned}
\frac{\partial \alpha_t^{(1)}}{\partial s_t^{(1)}} &= \frac{\left[e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} + e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right] \frac{s_t^{(1)}-\tilde{\mu}_p}{\tilde{\sigma}_p^2} e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}}{\left[e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} + e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right]^2} \\
&\quad - \frac{e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} \left[\frac{s_t^{(1)}-\tilde{\mu}_p}{\tilde{\sigma}_p^2} e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} - \frac{2\mu-s_t^{(1)}-\tilde{\mu}_p}{\tilde{\sigma}_p^2} e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right]}{\left[e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} + e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right]^2} \\
&= \frac{\frac{2(\mu-\tilde{\mu}_p)}{\tilde{\sigma}_p^2} e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}}{\left[e^{\{(s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}} + e^{\{(2\mu-s_t^{(1)}-\tilde{\mu}_p)^2/2\tilde{\sigma}_p^2\}}\right]^2}.
\end{aligned}
$$

Therefore, we have

$$
\frac{\partial \alpha_t^{(1)}}{\partial s_t^{(1)}} \begin{cases} < 0 \text{ when } \tilde{\mu}_p > \mu \\ = 0 \text{ when } \tilde{\mu}_p = \mu \\ > 0 \text{ when } \tilde{\mu}_p < \mu. \end{cases} \tag{3.10}
$$

Similarly, we can show that

$$
\frac{\partial \alpha_t^{(2)}}{\partial s_t^{(2)}} \begin{cases} < 0 \text{ when } \tilde{\mu}_p > \mu \\ = 0 \text{ when } \tilde{\mu}_p = \mu \\ > 0 \text{ when } \tilde{\mu}_p < \mu. \end{cases} \tag{3.11}
$$

Eqs. 3.10 and 3.11 suggest that when $\tilde{\mu}_p > \mu$ (i.e., the prior score $\tilde{\mu}_p$ is larger than most of the pattern-based scores), the fusion weights for small scores $\alpha_t^{(1)}$ increase when $s_t^{(1)}$ decreases, and the fusion weights for large scores $\alpha_t^{(2)}$ decrease when $s_t^{(2)}$ increases. Therefore, Eqs. 3.3 and 3.6 will assign larger weights to the small scores and thus decrease the mean fused score. In Figure 3.2(b), the right vertical line represents the mean of client scores and the left vertical line the mean of impostor scores, i.e., they represent the means of equal-weight fused scores. The figure shows that both the mean of the fused client scores and that of the fused imposter scores obtained by data-dependent fusion are smaller than that of their respective equal-weight fused scores when the prior score $\tilde{\mu}_p$ is greater than the respective score mean of the two utterances, i.e., $\tilde{\mu}_p > 1.0$ for the client scores and $\tilde{\mu}_p > -1.0$ for the impostor scores.

When $\tilde{\mu}_p < \mu$ (i.e., the prior score $\tilde{\mu}_p$ is smaller than most of the pattern-based scores), Eqs. 3.10 and 3.11 suggest that the fusion weights for small scores $\alpha_t^{(1)}$ decrease when $s_t^{(1)}$ decreases, and the fusion weights for large scores $\alpha_t^{(2)}$ increase when $s_t^{(2)}$ increases. As a result, the proposed fusion algorithm (Eqs. 3.3 and 3.6) emphasizes larger scores when $\tilde{\mu}_p < \mu$, which has the effect of increasing the mean fused scores. Again, Figure 3.2(b) shows that both the mean of the fused client scores and that of the fused imposter scores are greater than that of their respective equal-weight fused scores when the prior score $\tilde{\mu}_p$ is smaller than the respective score mean of the two utterances, i.e., $\tilde{\mu}_p < 1.0$ for the client scores and $\tilde{\mu}_p < -1.0$ for the impostor scores.

When $\tilde{\mu}_p = \mu$, data-dependent fusion will be equivalent to equal-weight fusion. This can be observed from Figure 3.2(b) where the curves intersect each other when the prior score $\tilde{\mu}_p$ is equal to the mean of client scores or impostor scores. This suggests that the mean of fused scores is equal regardless of the fusion algorithm used. Figure 3.2(b) also shows that when the prior score is set between the means of client scores and impostor scores (i.e., between the two vertical lines), theoretically the mean of fused client scores increases and the mean of fused impostor scores decreases. This has the effect of increasing the difference between the means of fused client scores and that of the fused impostor scores, as demonstrated in Figure 3.2(c). Because the mean of fused scores is used to make the final decision, increasing the score dispersion can decrease the speaker verification error rate.

To conclude, the fusion algorithm will either increase or decrease the mean of fused scores depending on the value of the prior score $\tilde{\mu}_p$ and the score mean μ before fusion. Because the increase or decrease in the mean fused scores is guaranteed rather

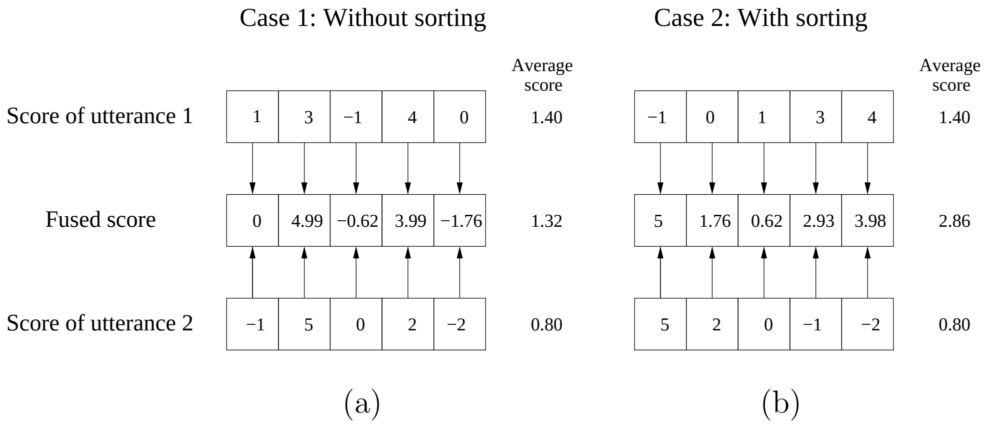


Figure 3.6: Fused scores derived from (a) unsorted and (b) sorted score sequences obtained from a client speaker. The assumption is that $\tilde{\mu}_p = 0$ and $\tilde{\sigma}_p^2 = 1$ in Eq. 3.6.

than probabilistic, the degree of increase or decrease for the fusion of sorted scores is larger than that for the fusion of unsorted scores, as demonstrated in Figure 3.2(b).

3.3.2 Numerical Example

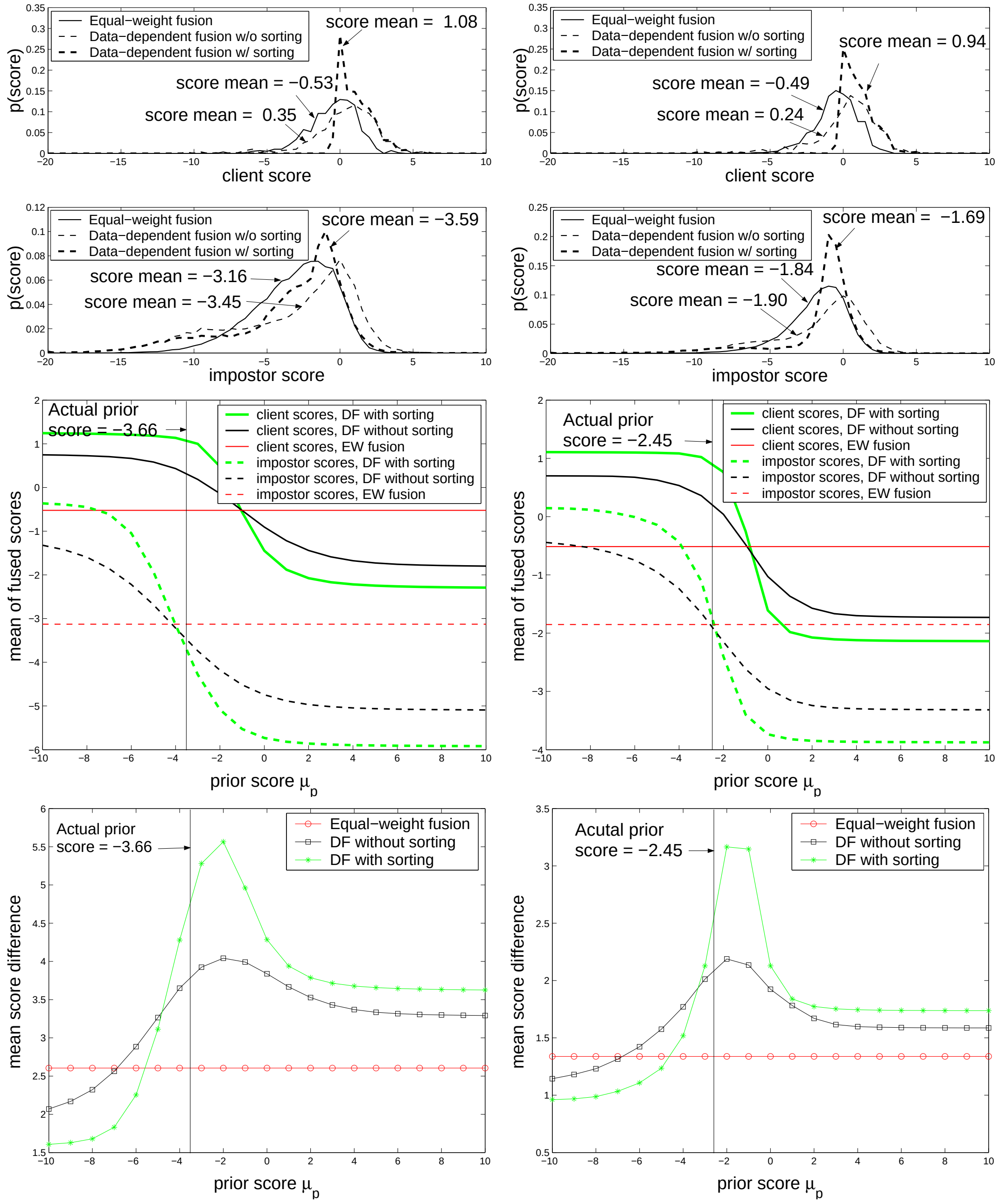
In the previous subsection, we argue that the fusion of sorted score sequences increases the score dispersion. Here, we compare the fusion of unsorted scores with the fusion of sorted scores via a numerical example. Figure 3.6 shows a hypothetical situation in which the scores were obtained from two client utterances. For client utterances, Eq. 3.6 should emphasize large scores and de-emphasize small scores. However, Figure 3.6(a) illustrates the situation in which a very small score (5th score of utterance 2—i.e., -2) is fused with a relatively large score (5th score of utterance 1—i.e., 0.0). In this case, $\alpha_t^{(1)} = \frac{e^0}{e^0+e^2} \approx 0.0$ and $\alpha_t^{(2)} = \frac{e^2}{e^0+e^2} \approx 1.0$, which means the fifth fused score (-1.76) is dominated by the fifth score of utterance 2. This is undesirable for client utterances.

The influence of these extremely small client scores on the final mean fused score can be reduced by sorting the scores of the two utterances in opposite order before fusion such that small scores will always be fused with large scores. With this arrangement, the contribution of some extremely small client scores in one utterance can be compensated by the large scores of another utterance. As a result, the mean of the fused client scores will be increased. Figure 3.6(b) shows that the mean of fused scores increases from 1.32 to 2.86 after sorting the scores. Likewise, if this sorting approach is applied to the scores of impostor utterances with a proper prior score $\tilde{\mu}_p$ (i.e., greater than the mean of impostor scores, see Figure 3.2(b)), the contribution of some extremely large impostor scores in one utterance can be greatly reduced by the small scores in another utterance, which has the net effect of minimizing the mean of the fused impostor scores. Therefore, this score sorting approach can further increase the dispersion between client scores and impostor scores, resulting in a lower error rate. This is demonstrated in Figure 3.2(c) where the score dispersion achieved by data-dependent fusion with score sorting is significantly larger than that without score sorting.

3.3.3 Effect on Real Speech Data

To further demonstrate this phenomenon, we select two client speakers (faem0 and mdac0) from the HTIMIT corpus [62] and plot the distributions of the fused speaker scores and fused impostor scores in Figure 3.7. In Eq. 3.5, we use the overall mean $\tilde{\mu}_p$ as the prior score. However, because the number of background speakers' utterances is usually much larger than that of client speaker's utterances during the training phase, the overall mean is very close to the mean score of background speakers, i.e., $\tilde{\mu}_p \approx \tilde{\mu}_b$. According to Figure 3.2(b) and the third row of Figure 3.7, when $\tilde{\mu}_p \approx \tilde{\mu}_b$, the mean of fused impostor scores are almost identical for all fusion algorithms under investigation. However, the same $\tilde{\mu}_p$ will increase the mean of fused client scores significantly, especially when the client scores were sorted before fusion.

Figure 3.7(a) shows that the mean of client scores increases from 0.35 to 1.08 and



(a) client speaker "mdac0" (b) client speaker "faem0"

Figure 3.7: Distributions of pattern-by-pattern client scores (*top row*) and impostor scores (*second row*), the mean of fused client scores and the mean of fused impostor scores (*third row*), and difference between the mean of fused client scores and the mean of fused impostor scores (*bottom row*) based on equal-weight fusion (score averaging) and data-dependent fusion with and without score sorting. The means of speaker scores and impostor scores obtained by both fusion approaches are also shown.

the mean of impostor scores decreases from -3.45 to -3.59 after sorting the score sequences.[5] Therefore, the dispersion between the mean client score and the mean impostor score increases from 3.80 to 4.67. We can notice from Figure 3.7(b) (top and second figure) that after sorting the scores, both the mean of client scores and the mean of impostor scores increase. This is because the means of impostor scores obtained from verification utterances are greater than the prior score $\tilde{\mu}_p$. This causes the increase in the mean of fused impostor scores. However, because the increase in the mean client scores is still greater than the increase in the mean impostor scores, there is still a net increase in the score dispersion. Specifically, the dispersion in Figure 3.7(b) increases from $2.14(= 0.24 - (-1.90))$ to $2.63(= 0.94 - (-1.69))$. Because verification decision is based on the mean scores, the wider the dispersion between the mean client scores and the mean impostor scores, the lower the error rate.

3.3.4 Experiments and Results

Experiments Based on Telephone Speech

Speech Corpora and Feature Extraction. The HTIMIT corpus [62] was used in this part of the experiments. HTIMIT was obtained by playing a subset of the TIMIT corpus through nine telephone handsets and a Sennheizer head-mounted microphone. We also used a GSM speech coder [64] to transcode the HTIMIT corpus and applied the data-dependent fusion techniques described in Section 3.1 to the resynthesized coded speech. In the sequel, we denote the two corpora as HTIMIT and GSM-HTIMIT.

Speakers in HTIMIT were divided into: a speaker set, an impostor set, and two pseudo-impostor sets (PI-20 and PI-100). The speaker identities of these sets are

[5] The decrease in the mean of fused impostor scores is caused by the prior score $\tilde{\mu}_p$ being greater than the mean of the unfused impostor scores—see the fourth figure of Figure 3.7(a).

Table 3.1: Speaker identities in the speaker set, impostor set and pseudo-impostor sets. The speakers in these sets are arranged alphabetically.

speaker set (50 female and 50 male)	**impostor set** (25 female and 25 male)	**pseudo-impostor set PI-20** (10 female and 10 male)	**pseudo-impostor set PI-100** (50 female and 50 male)
fadg0,faem0,. . .,fdxw0 mabw0,majc0,. . .,mfgk0 mjls0,mjma0,mjmd0 mjmm0,mpdf0	feac0,fear0,. . .,fjem0 mfxv0,mgaw0,. . .,mjlg1	fjen0,fjhk0,. . .,fjrp1 mjpm1,mjrh0,. . .,mpgr1	fjen0,fjhk0,. . .,fmbg0 mjpm1,mjrh0,. . .,msrr0

shown in Table 3.1. Similar to the speakers in HTIMIT, speakers in GSM-HTIMIT were also divided into three sets and the speaker identities of these sets are identical to those in HTIMIT.

12th-order mel-frequency cepstrum coefficients (MFCC) [40] were extracted from the utterances in the corpora. The extracted feature vectors were computed using a Hamming window of 28ms at a frame rate of 14ms.

Enrollment Procedures. For each speaker in the speaker set, we used the SA and SX utterances from handset "senh" of HTIMIT to create a 32-center GMM speaker model. A 64-center universal GMM background model [42] was also created based on the speech of 100 client speakers in the speaker set. The background model was shared among all client speakers in subsequent verification sessions. Besides the speaker models and background model, a 2-center GMM (Λ_X in Section 2.4.1) was also created using the uncoded HTIMIT utterances (from handset senh) of 10 speakers. A set of transformation parameters ($\mathbf{b}$'s in Eq. 2.12) were then estimated using this model and the handset- and codec-distorted speech (see Section 2.4.1).

We fed the SA and SX utterances of all speakers in the speaker set to the background model and each of the speaker models to obtain the speaker scores corresponding to the enrollment data. For each speaker, the SA and SX utterances of all other speakers in the speaker set were also fed to the background model and the corresponding speaker model to obtain the pseudo-impostor scores. These scores were used to compute the client score mean $\tilde{\mu}_c$ and background speakers' score mean $\tilde{\mu}_b$, as illustrated in Figure 3.1. These score means were then used to compute the prior score $\tilde{\mu}_p$ and prior variance $\tilde{\sigma}_p^2$ according to Eq. 3.5.

Verification Procedures. The SI sentences in the corpora were used for verification. We assume that a claimant will be asked to utter two sentences during a verification session, i.e., $K = 2$ in Eq. 3.1. To obtain the scores of the claimant's utterances, the utterances were first fed to a handset detector [48–50] to determine the set of feature transformation parameters to be used. The features were transformed and then fed to the claimed speaker model and the background model to obtain two streams of normalized scores (Eq. 3.2), one for each utterance. Then, the proposed fusion algorithms were applied to fuse the two streams of scores.

The fusion algorithm (Eqs. 3.3 and 3.4) requires that the two streams of scores have identical length. This can be achieved by appending the tail of the longer score sequence to the shorter one.[6] In this work, this length was calculated according to

$$L = \lfloor \frac{L_1 + L_2}{2} \rfloor, \tag{3.12}$$

where L is a positive integer and L_1 and L_2 represent the length of the first and second utterance, respectively. Figure 3.8 illustrates how the length-equalization procedure is performed. In the figure, there are 105 and 138 feature vectors in the first and second utterance, respectively. According to Eq. 3.12, the equal length is 121 ($= \lfloor (105 + 138)/2 \rfloor$). After finding the equal length, the remaining scores in the longer score stream (Utterance 2 in this case) were appended to the end of the shorter

[6] We cut out the tail of the longer utterance and appended it to the shorter utterance because we want to maximize the numbers of fused scores. Dynamic programming algorithms such as dynamic time warping was not used because the acoustic contents of different utterances may not be the same.

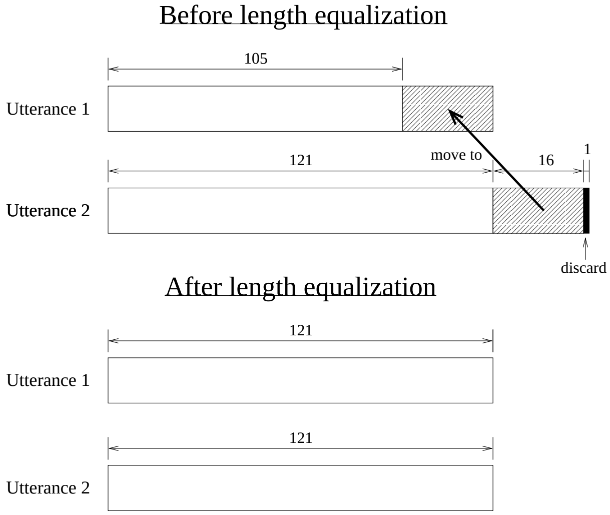


Figure 3.8: Procedure for making equal-length score streams. The tail of the longer stream is appended to the tail of the shorter one.

one (Utterance 1 in this case). In this example, one extra score in Utterance 2 was discarded to make the score length equal.

To compare with the score averaging approach proposed in [58], we also fused the utterances' scores using equal-weight fusion, i.e., $\alpha_t^{(1)} = \alpha_t^{(2)} = 0.5 \; \forall t = 1, \ldots, L$.

Results of Fusion of Sorted and Unsorted Scores. Figure 3.9 depicts the detection error tradeoff (DET) curves based on 100 client speakers and 50 impostors using utterances from handset "cb1" for verification. Figure 3.9 shows that (1) with feature transformation, data-dependent fusion can reduce the error rates significantly; and (2) sorting the scores before fusion can reduce the error rate further. However, without feature transformation, the performance of data-dependent fusion with score sorting is not significantly better than that of the equal-weight fusion. This is caused by the mismatch between the prior scores $\tilde{\mu}_p$'s in Eq. 3.6 and the scores of the distorted features. Therefore, it is very important to use feature transformation to reduce the mismatch between the enrollment data and verification data.

Figure 3.10 shows the detection error trade-off curves based on 100 client speakers

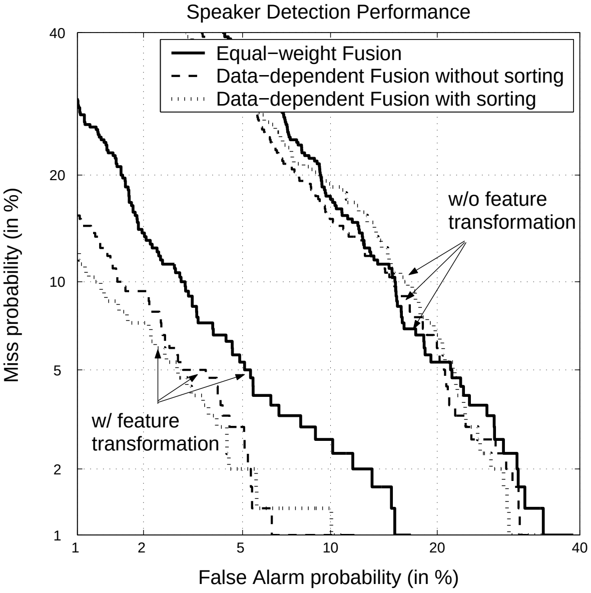


Figure 3.9: DET curves for equal-weight fusion (score averaging) and data-dependent fusion with and without score sorting. The curves were obtained by using the utterances of handset "cb1" as verification speech.

and 50 impostors using the scores of ten handsets. It shows that data-dependent fusion with score sorting outperforms equal-weight fusion for all operating points and by 23% in terms of equal error rate.

Table 3.2 shows the speaker detection performance of 100 speakers and 50 impostors for the equal-weight fusion approach and the proposed fusion approach with and without sorting the score sequences. Table 3.2 clearly shows that the proposed fusion approach outperforms the equal-weight fusion one. In particular, after sorting the score sequences, the equal error rates are further reduced. Compared to the unsorted cases, an 11% error reduction has been achieved.

Experiments Based on Cellular Phone Speech

Speech Corpus and Feature Extraction. To further demonstrate the capability of data-dependent fusion under practical situations, evaluations were performed

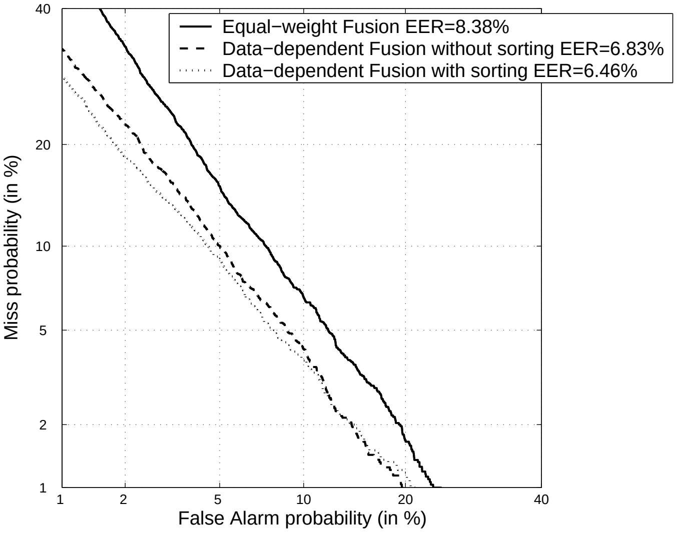


Figure 3.10: DET curves for equal-weight fusion (score averaging) and data-dependent fusion with and without score sorting. The curves were obtained by concatenating the scores of ten handsets.

Table 3.2: Equal error rates achieved by different fusion approaches, using utterances from 10 different handsets for verification. Each figure is based on the average of 100 speakers, each impersonated by 50 impostors. DF stands for data-dependent fusion. "No fusion" means that the verification results were obtained from using a single utterance per verification session. "Average" denotes the average EER of 10 handsets.

	Equal Error Rate (%)										
Fusion Method	cb1	cb2	cb3	cb4	el1	el2	el3	el4	pt1	senh	**Average**
No fusion	6.31	6.36	19.96	15.35	5.89	10.83	11.79	8.18	9.38	4.90	**9.90**
Equal-weight fusion	5.11	4.33	19.15	12.89	4.42	8.31	9.96	6.29	7.57	2.99	**8.10**
DF w/o sorting	4.01	3.27	15.92	10.55	3.04	6.51	8.67	4.75	7.51	2.32	**6.67**
DF w/ sorting	3.60	2.86	15.30	9.91	3.49	4.65	6.81	4.02	6.59	1.99	**5.92**

using cellular phone speech: the 2001 NIST speaker recognition evaluation set [65]. The 2001 NIST evaluation set contains cellular phone speech of 174 target speakers (clients) with 74 male and 100 female. Another set of 60 speakers is also available for development. Each target speaker has 2 minutes of speech for training, and the duration of test utterances varies from 15 to 45 seconds. The evaluation set provides a total of 20,380 gender-matched verification trials. The ratio between target and impostor trials is roughly one to ten.

In the experiments, 12 MFCCs and their delta coefficients were extracted from the utterances at a rate of 71Hz using a 28ms Hamming window. Cepstral mean subtraction (CMS) [66] was applied to all MFCCs before they were appended to the delta MFCCs.

Enrollment Procedures. A 1024-component universal background model (UBM) was created based on the speech of 60 speakers in the development set of the corpus. Then, for each client speaker, a speaker-dependent GMM was created by adapting the UBM using maximum a posteriori (MAP) adaptation [42]. The speaker-dependent prior scores $\tilde{\mu}_p$ and variances $\tilde{\sigma}_p^2$ in Eq. 3.6 were then estimated from all the testing speech in the development set of the corpus. This was achieved by considering the testing speech in the development set as produced by pseudo-impostors and by presenting their feature vectors to the speaker and background models to obtain a sequence of speaker-dependent pseudo-impostor scores. The utterances used for training the speaker models were also presented to the speaker and background models to obtain a sequence of client scores. The averages of these scores were then used to estimate the prior scores and variances as in Eqs. 3.5 and 3.6. The dotted box in Figure 3.1 illustrates the process of estimating the prior scores and variances.

Verification Procedures. For each verification session, the feature sequence Y obtained from a claimant's utterance was transformed by blind stochastic feature transformation (BSFT) [53, 54] to form a sequence of transformed vectors Y_ν.

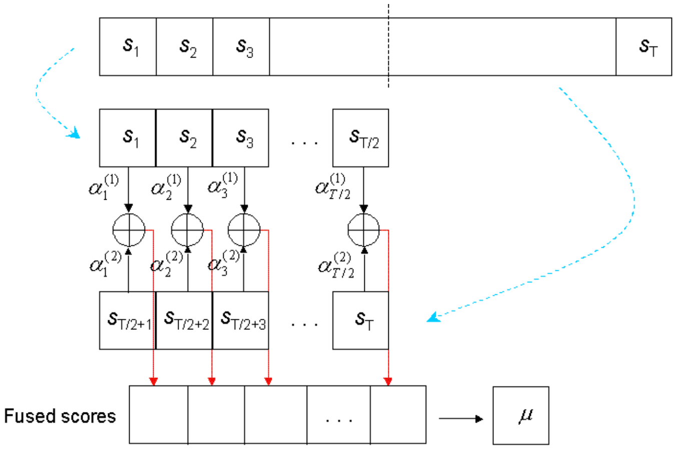


Figure 3.11: Process of fusion of single utterance.

First-order BSFT with 64 components ($M = 64$ in Section 2.4.2) was used. The transformed vectors were then fed to a 1024-component speaker model (Λ_s^{1024}) and a 1024-component UBM (Λ_b^{1024}) to obtain a sequence of normalized scores

$$S(\mathbf{y}_\nu) = \log p(\mathbf{y}_\nu|\Lambda_s^{1024}) - \log p(\mathbf{y}_\nu|\Lambda_b^{1024}), \;\; \mathbf{y}_\nu \in Y.$$

The average of these scores was used for decision making.

The proposed fusion algorithm was applied to compute the weight of each score. Because there is only one test utterance in each verification session, the utterance was split into two segments and then fusion was performed. Figure 3.11 illustrates the procedure of splitting a utterance for fusion. In this case, the length was calculated according to

$$T = \lfloor \frac{L}{2} \rfloor,$$

where T is a positive integer and L represents the length of the test utterance.

Performance Measures. We used detection error tradeoff (DET) curves and detection cost function (DCF) as performance measures. DCF is defined as follows:

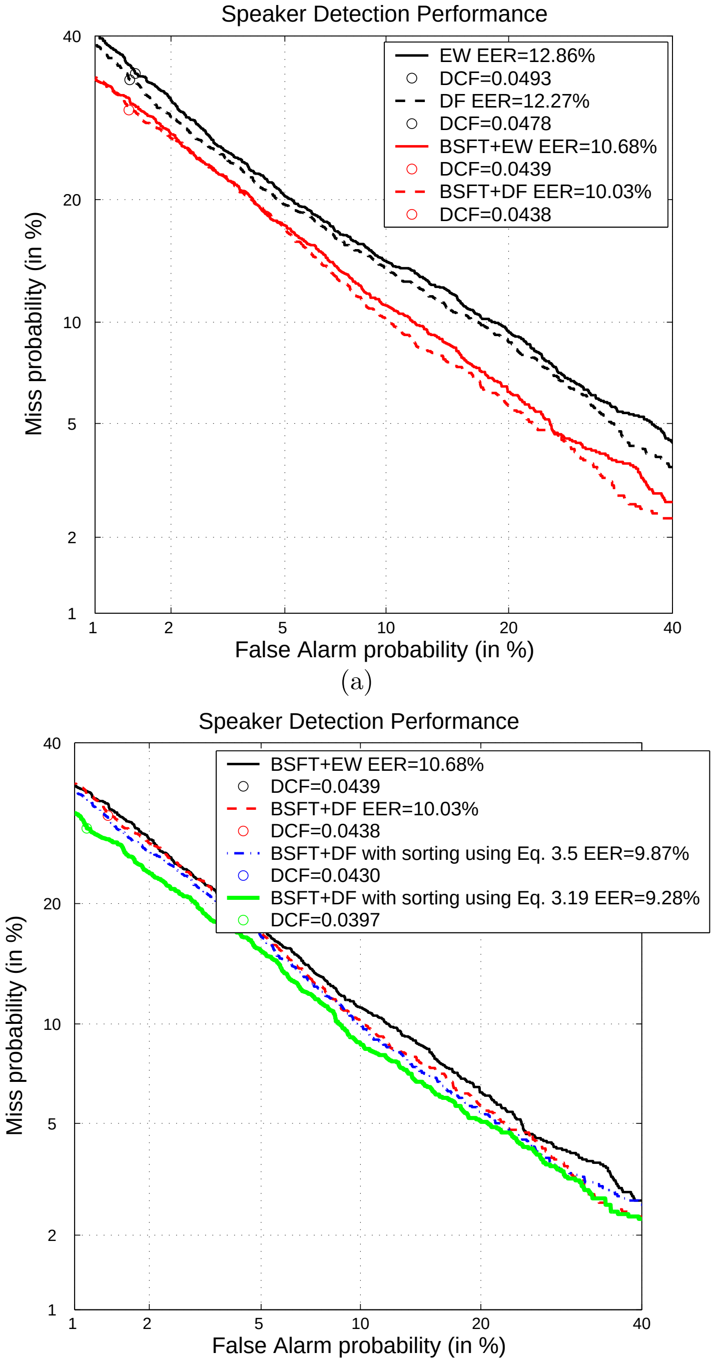
$$\text{DCF} = C_{FA}Pr(FA|I)Pr(I) + C_{FR}Pr(FR|T)Pr(T), \tag{3.13}$$

where $Pr(I)$ and $Pr(T)$ are the prior probability of impostors and target speakers, respectively, and C_{FA} and C_{FR} are the costs of false alarm and false rejection, respectively. In this work, we set $Pr(I) = 0.99$, $Pr(T) = 0.01$, $C_{FA} = 1$ and $C_{FR} = 10$, as suggested in Przybocki and Martin [67].

Results and Discussions. Figure 3.12 compares the speaker detection performance of equal-weight fusion and data-dependent fusion. Figure 3.12(a) shows that multisample data-dependent fusion performs better than equal-weight fusion. In particular, the equal error rate (EER) achieved by data-dependent fusion (without BSFT) is 12.27%. When compared to equal-weight fusion (which achieves an EER of 12.86%), a relative error reduction of 5% was obtained. Data-dependent fusion also achieves a 3% improvement in terms of minimum DCF as compared with the baseline. With BSFT, data-dependent fusion has a 22% relative error reduction as compared to the baseline. Regarding minimum DCF, a 11% improvement is achieved as compared to the baseline. The results suggest that BSFT is important to both equal-weight and data-dependent fusions.

Figure 3.12(b) shows that in terms of error rates, data-dependent fusion with score sorting performs better than that without score sorting. However, it has just a slightly improvement. This may be due to the increase in the sensitivity of prior scores. To reduce the influence of prior scores on the fusion weights, we modified the fusion equation (Eq. 3.6) to the following:

$$\alpha_t^{(k)} = \frac{\exp\{(|s_t^{(k)} - \tilde{\mu}_p|)/2\tilde{\sigma}_p^2\}}{\sum_{l=1}^{K} \exp\{(|s_t^{(l)} - \tilde{\mu}_p|)/2\tilde{\sigma}_p^2\}}. \tag{3.14}$$



(a)

(b)

Figure 3.12: Speaker detection performance for data-dependent (DF) fusion with and without score sorting and equal-weight (EW) fusion. BSFT stands for blind stochastic feature transformation. (a) Highlighting the importance of BSFT in equal-weight and data-dependent fusion. (b) Highlighting the benefit of score sorting. For ease of comparison, the labels in the legend are arranged in decreasing EERs.

Figure 3.12(b) shows that after replacing Eq. 3.6 by Eq. 3.14, data-dependent fusion with score sorting performs better than that without score sorting in terms of both EERs and minimum DCF. Specifically, a relative error reduction of 7.5% was achieved. Regarding minimum DCF, a 9% improvement was obtained.

3.4 Adaptation of Prior Scores

3.4.1 Rationale for Adapting Prior Scores

Eq. 3.6 uses the overall mean $\tilde{\mu}_p$ as the prior score. However, because the number of background speaker's utterances is much larger than that of speaker's utterances during the training phase, the overall mean is very close to the mean score of the background speakers, i.e., $\tilde{\mu}_p \approx \tilde{\mu}_b$. This causes most of the client-speaker scores larger than the prior score; moreover, only part of the impostor scores will be smaller than the prior score. An example of this situation is illustrated in Figure 3.13 in which the distributions of the pattern-by-pattern speaker scores and impostor scores corresponding to speaker "mdac0" are shown. Figures 3.5 and 3.13 show that if the claimant is a client speaker, the fusion algorithm will emphasize large scores because most of the speaker scores in Figure 3.13 are larger than the prior score. On the other hand, the algorithm will have equal preference on both small and large scores if the claimant is an impostor because the prior score lies on the middle of the impostor scores distribution.

The ultimate goal of Eq. 3.3 is to apply a larger weight to the large scores to make the fused score larger when the claimant is a client speaker; on the other hand, if the claimant is an impostor, Eq. 3.3 should apply a smaller weight to the small scores to keep the fused score small. In other words, the goal is to increase the separation between client speaker's scores and impostors' scores. As a result, when the claimant

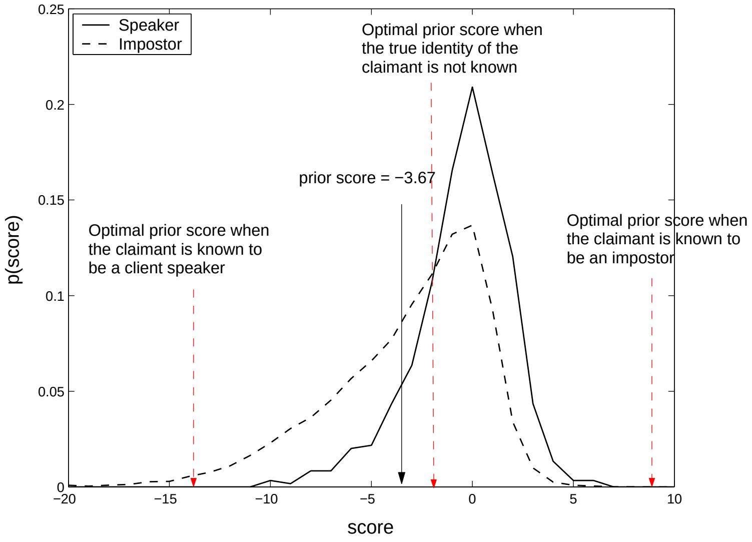


Figure 3.13: Distribution of pattern-by-pattern speaker scores and impostor scores corresponding to speaker "mdac0". *Note*: Most of the speaker scores are larger than the prior score and that only some of the impostor scores are smaller than the prior score.

is a true speaker, the prior score should be smaller than all possible speaker scores so that Eq. 3.3 only emphasizes larger scores; on the other hand, when the claimant is an impostor, the prior score should be larger than all possible impostor scores so that Eq. 3.3 only favors smaller scores. However, satisfying these two conditions simultaneously is almost impossible in practice because the true identity of the claimant is never known. Therefore, the optimal prior score should be equal to the intersection point of the speaker score distribution and impostor score distribution (see Figure 3.13). At that point, the number of speaker scores smaller than the prior score plus the number of impostor scores larger than the prior score is kept to a minimum.

3.4.2 Adaptation Algorithm

Here, we propose a method to adapt the prior score during verification to achieve the goal mentioned in Section 3.4.1. For each client speaker, a one-dimensional GMM score model $\Omega_s = \{\pi_j, \mu_j, \Sigma_j\}_{j=1}^{M}$ with output

$$p(\tilde{s}; \Omega_s) = \sum_{j=1}^{M} \pi_j p(\tilde{s}|j) = \sum_{j=1}^{M} \pi_j \mathcal{N}(\tilde{s}; \mu_j, \Sigma_j) \tag{3.15}$$

is trained to represent the distribution of the utterance-based background speakers' scores $\tilde{s} \equiv \frac{1}{T} \sum_{t=1}^{T} s(\mathbf{o}_t; \Lambda)$ obtained during the training phase. During verification, a claimant will be asked to utter K utterances $\mathcal{U} = \{\mathcal{U}_1, \ldots, \mathcal{U}_K\}$. For the k-th utterance, the normalized likelihood that the claimant is an impostor is computed:

$$\zeta = \frac{p(\tilde{s}_k|\Omega_s)}{\max_{-\infty \le s \le \infty} p(s|\Omega_s)} \tag{3.16}$$

where

$$\tilde{s}_k \equiv \frac{1}{T_k} \sum_{t=1}^{T_k} s(\mathbf{o}_t; \Lambda) \tag{3.17}$$

is the score of the k-th utterance. The purpose of the denominator in Eq. 3.16 is to determine the maximum value of the likelihood that the claimant is an impostor. By having this term, we can limit ζ to fall within the range [0,1]. In Eq. 3.16, when the normalized likelihood ζ is close to 1, the claimant is likely to be an impostor. Therefore, the prior score should be increased such that the large scores of this claimant are deemphasized so that the system can reject this claimant. On the other hand, when the normalized likelihood ζ is close to 0, the claimant is likely to be the true speaker. As a result, the prior score should remain unchanged such that the large scores of this claimant can be emphasized. Figure 3.9 gives an example of the relationship between the adapted prior score and normalized likelihood. When the value of the normalized likelihood is close to zero, the adapted prior score will be close to the original value, i.e., -5. However, when the value of the normalized likelihood is close to one, the prior score is adapted to a maximum value.

The prior score is then adapted according to

$$\widehat{\mu}_p^{(k)} = \tilde{\mu}_p + f(\zeta), \tag{3.18}$$

where $\widehat{\mu}_p^{(k)}$ is the adapted prior score and $f(\zeta)$ is a monotonic increasing function controlling the amount of adaption. Then, we replace $\tilde{\mu}_p$ in Eq. 3.6 by $\widehat{\mu}_p^{(k)}$ to compute the fusion weights. The reason of using a monotonic increasing function for $f(\zeta)$ is that if the normalized likelihood of a claimant is large, he or she is likely to be an impostor. As a result, the prior score should be increased by a large degree to de-emphasize the large scores of this claimant. With the large scores being de-emphasized, the claimant's mean fused score will become smaller, thus increasing the chance of rejecting this potential impostor.

Notice that Eq. 3.18 increases the prior score rather than decreasing it because we found that the optimal prior score is always greater than the unadapted prior score $\tilde{\mu}_p^{(k)}$ [63]. A monotonic function of the form

$$f(\zeta) = \frac{a(e^{b\zeta} - 1)}{e^b - 1} \tag{3.19}$$

was suggested, where a represents the maximum amount of adaptation and b controls the rate of increase of $f(\zeta)$ with respect to ζ. Figure 3.14 shows the effect of varying a and b on the adapted prior scores.

The values of a and b should not be too large. If a is too large (e.g., $a = 20$ in Figure 3.14), Eq. 3.18 will lead to a large adapted prior score $\widehat{\mu}_p^{(k)}$ even for utterances with low normalized likelihood ζ. As a result, almost all of the claimant scores will become smaller than the prior score, which is undesirable. Moreover, a large a will result in poor verification performance when there is a large mismatch between the pseudo-impostor score model and verification scores. This is because in a severe mismatch situation, Eq. 3.16 will produce incorrect normalized likelihood. This will

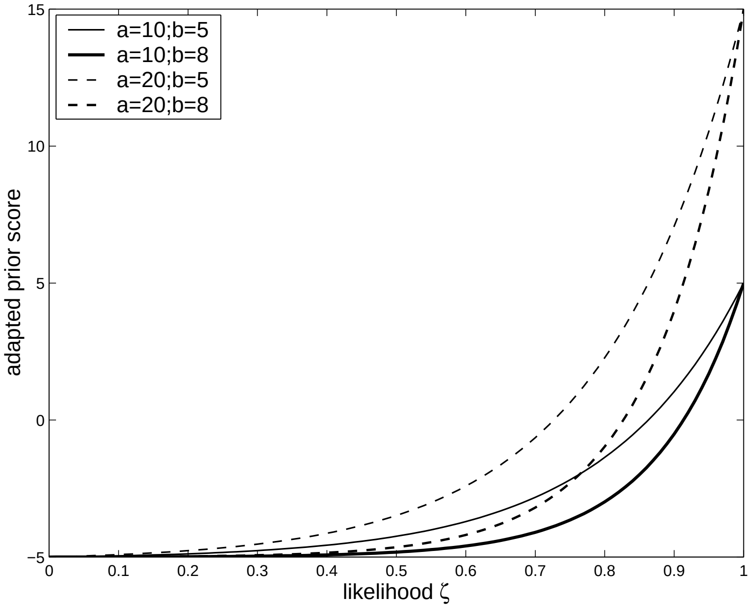


Figure 3.14: The influence of varying a and b of Eq. 3.19 on the adapted prior scores $\widehat{\mu}_p^{(k)}$, assuming $\tilde{\mu}_p = -5$.

lead to incorrect adaptation of the prior score, which in turn results in incorrect fusion weights. On the other hand, if b is too large, there will be a sharp bend in the adaptation curves, as illustrated in Figure 3.14. As a result, the prior score will be adapted only when the verification utterances give high likelihood in Eq. 3.16. We found that the best range for parameter a is 5 to 20 and that for b is 5 to 8.

3.4.3 Effect on Score Distributions

To demonstrate the effect of the proposed prior score adaptation on fused score distributions, we have selected arbitrarily a client speaker (mdac0) from GSM-transcoded HTIMIT [59] and plot the distributions of the fused speaker scores and fused impostor scores in Figure 3.15, using equal-weight fusion ($\alpha_t^{(1)} = \alpha_t^{(2)} = 0.5\ \forall t$), data-dependent fusion without prior score adaptation (Eq. 3.6), and data-dependent fusion with prior score adaptation (Eq. 3.18). Evidently, the upper part of Figure 3.15 shows that the number of large client-speaker scores is larger in data-dependent fusion. However, the distribution of client-speaker pattern-based scores are almost identical before and after prior score adaptation. This is because the pseudo-impostors' score model gives low likelihood for client speaker's utterances. Therefore, the normalized likelihood ζ for client speaker's utterances will be close to zero. As a result, the prior score remains unchanged or only increases slightly. Therefore, the client score distribution remains unchanged. The lower part of Figure 3.15 shows that there are more small impostor scores in data-dependent fusion than in equal-weight fusion. More smaller scores are emphasized after applying data-dependent fusion with prior score adaptation. The figures also show that data-dependent fusion with prior score adaptation outperforms the other two fusion approaches.

Further evidence demonstrating the advantage of prior score adaptation can be found in Table 3.3, which shows that the dispersion between the mean client score and the mean impostor score increases from 2.63 $(= -0.53 - (-3.16))$ to 4.04 $(= 0.25 - (-3.79))$ without prior score adaptation and to 4.61 $(= 0.22 - (-4.39))$ with prior score adaptation. Because verification decisions are based on the mean scores, the wider the dispersion between the mean client scores and the mean impostor scores, the lower the error rate. We can also notice from Table 3.3 that with prior score adaptation, both the mean speaker score and the mean impostor score are reduced. The reason for a reduction in the mean speaker score is that there is a small increase in the prior score when the claimant is a true speaker, which increases the number of scores that are smaller than the prior score. In other words, more small speaker scores are emphasized by Eq. 3.6, which lowers the fused speaker scores as well as the mean speaker score.

Table 3.3: The mean fused speaker scores and fused impostor scores of speaker mdac0 obtained by different fusion approaches. EW and DF represent equal-weight and data-dependent fusion, respectively. PS stands for prior score.

	EW	DF without PS adaptation	DF with PS adaptation
prior score	N/A	-3.67	Vary for different utterances
mean speaker score	-0.53	0.25	0.22
mean impostor score	-3.16	-3.79	-4.39
score dispersion	2.63	4.04	4.61

3.4.4 Experiments and Results

Experiments Based on Telephone Speech

Enrollment Procedures. The enrollment procedures were the same as the one described in Section 3.3.4. However, in this part of the experiments, two sets of scores were collected. For the first set, we fed the SA and SX utterances of all speakers in the speaker set to the background model and each of the speaker models to obtain the speaker scores corresponding to the enrollment data. For the second set, we fed the utterances of all speakers in the pseudo-impostor sets to the background model and each of the speaker models to obtain the pseudo-impostor scores corresponding to "unseen" impostor data. One pseudo-impostor score was obtained from each pseudo-impostor utterance. These two sets of scores were used to compute the client score mean μ_c and background speakers' score mean μ_b, as illustrated in Figure 3.1. These score means were then used to compute the prior score $\tilde{\mu}_p$ and prior variance $\tilde{\sigma}_p^2$ according to Eq. 3.5. The resulting utterance-based pseudo-impostor scores were also used to create a 2-center 1-D GMM pseudo-impostor score model (Ω_s in Eq. 3.15).

Results of Adaptation of Prior Scores. In the experiments, we set the parameters a and b in Eq. 3.18 to 10 and 5, respectively. Figure 3.16 depicts the speaker

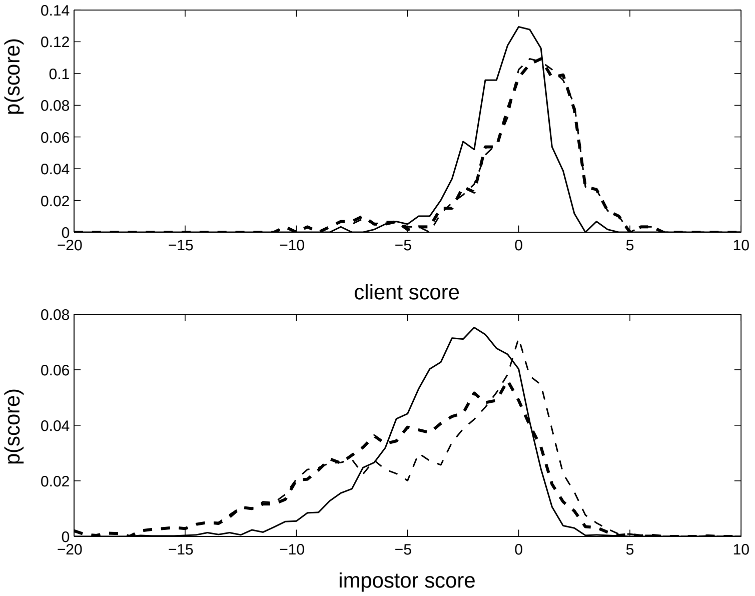


Figure 3.15: Distribution of pattern-by-pattern speaker scores (upper figure) and impostor scores (lower figure) based on equal-weight fusion (score averaging) and data-dependent fusion. *Solid*: equal-weight fusion. *Thin Dashed*: Data-dependent fusion without prior score adaptation. *Thick Dashed*: Data-dependent fusion with prior score adaptation.

detection performance based on 100 speakers and 50 impostors for equal-weight fusion and data-dependent fusion with and without prior score adaptation. Figure 3.16 shows that with feature transformation, data-dependent fusion with prior score adaptation achieves the lowest error rates. In particular, the equal error rate (EER) achieved by data-dependent fusion with prior score adaptation is 4.01%. When compared to equal-weight fusion (which achieves an EER of 5.11%), a relative error reduction of 22% was obtained.

In Figure 3.16, we used the speakers in the speaker set to train the pseudo-impostor score model. It is of interest to see the distribution of pseudo-impostor scores obtained from unseen data. These distributions are illustrated in Figure 3.17. A comparison between Figures 3.17(a) and 3.17(c) reveals that there is a mismatch between the

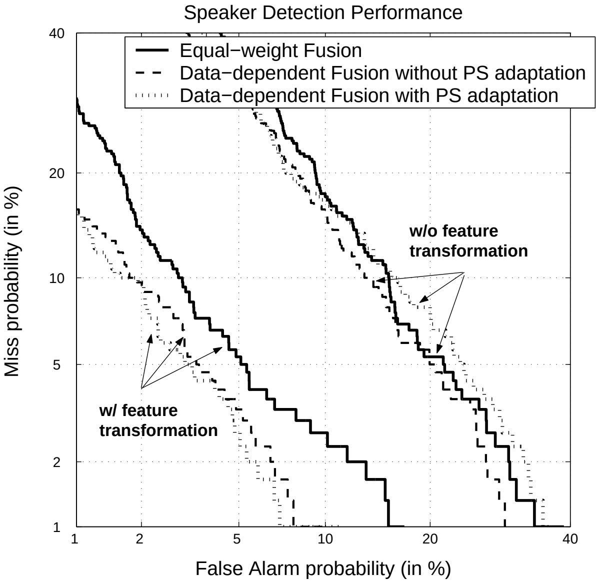


Figure 3.16: Speaker detection performance for equal-weight fusion (score averaging) and data-dependent fusion with and without prior score (PS) adaptation.

pseudo-impostor score distribution and the impostor score distribution. In particular, there are more small scores in Figure 3.17(a) than in Figure 3.17(c). This mismatch affects the adaptation of prior scores because the likelihood ζ in Eq. 3.16 is no longer accurate. This is evident in Table 3.4 where the equal error rates using different numbers of pseudo-impostors with speech extracting from different corpora are shown. In particular, the second column of Table 3.4 shows that the improvement after adapting the prior scores is just 3.14% (from 4.14% to 4.01%).

To create more stable and reliable pseudo-impostor score models, we used another pseudo-impostor set (namely PI-100) that contains 100 pseudo-impostors to train the pseudo-impostor score models. Also, we used the GSM-transcoded speech instead of HTIMIT speech to train the model in an attempt to create a verification environment as close to the real one as possible. We also transformed the GSM-transcoded speech before calculating the scores. Figures 3.17(b) and 3.17(c) show that the scores

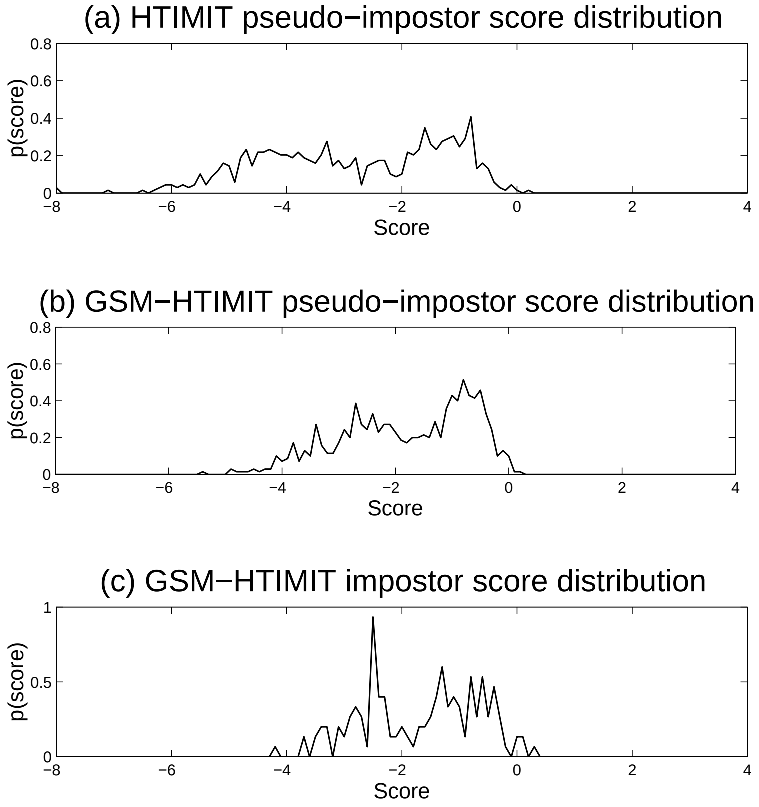


Figure 3.17: Distribution of pseudo-impostor scores for speaker "fadg0" obtained by using (a) HTIMIT speech in the speaker set and (b) GSM-transcoded speech in the pseudo-impostor set. (c) Distribution of impostor scores for speaker "fadg0" during verification using GSM-transcoded speech.

distributions are much closer after using the transformed features. The improvement after adapting the prior score is 17% (from 3.52% to 2.92%), as demonstrated in the second column (PI-100 GSM) of Table 3.4.

Table 3.4 indicates that even for pseudo-impostor set containing as little as 20 speakers (PI-20 GSM), a lower equal error rate (3.01%) can still be obtained by using the GSM-transcoded speech to train the pseudo-impostor score models. Error rates can be further reduced (from 3.01% to 2.55%) by applying the prior score adaptation together with score sorting.

Table 3.4: EERs achieved by data-dependent fusion approaches. The values inside the parentheses are the EERs obtained by using the score sorting approach. The EERs achieved by "No Fusion" (single utterance) and "EW Fusion" are 6.31% and 5.11%, respectively. Each figure is based on the average of 100 speakers, each impersonated by 50 impostors. DF stands for data-dependent fusion.

	PI-20 GSM	PI-100 GSM	PI-100 HTIMIT	SS-100 HTIMIT
DF w/o PS adaptation	3.61% (2.81%)	3.52% (2.67%)	3.92% (3.80%)	4.14% (3.60%)
DF w/ PS adaptation (a=10)	3.01% (2.55%)	2.92% (2.25%)	3.68% (3.51%)	4.01% (3.60%)
DF w/ PS adaptation (a=20)	2.80% (2.78%)	2.57% (2.71%)	4.22% (4.23%)	4.05% (3.69%)

The abbreviations in the first row of the table denote the data used for creating the pseudo-impostor score models. They are detailed as follows:

- PI-20 GSM: Pseudo-impostor set containing 20 speakers, with GSM-transcoded speech from handset "cb1"
- PI-100 GSM: Pseudo-impostor set containing 100 speakers, with GSM-transcoded speech from handset "cb1"
- PI-100 HTIMIT: Pseudo-impostor set containing 100 speakers, with HTIMIT speech from handset "senh"
- SS-100 HTIMIT: Speaker set containing 100 speakers, with HTIMIT speech from handset "senh"

By using GSM-transcoded speech, real verification environments can be modeled so that parameter a can be increased to obtain a much lower EER (from 3.01% to 2.80% when using 20 pseudo-impostors and from 2.92% to 2.57% when using 100 pseudo-impostors). However, the error rates increase from 3.68% to 4.22% and from 4.01% to 4.05% when there is a mismatch between the pseudo-impostor score distributions and the pseudo-impostor score models. Because the fusion weights are an exponential function of the distance between the individual scores and the prior score, Eq. 3.6 puts more emphasis on small scores when the prior score increases. Therefore, if the claimant is an impostor, the mean impostor score will become smaller. On the other hand, if the prior score is wrongly adapted, the mean speaker score will also become smaller because there are more scores smaller than the prior score.

Figure 3.14 shows that for the same likelihood, the increase in prior scores is larger when parameter a is increased from 10 to 20. In other words, even if inaccurate likelihood is obtained, its influence on the system performance is small when parameter a is small. Therefore, if stable and reliable pseudo-impostor score distributions can be obtained, parameter a can be increased to further improve the verification performance. This is because if an utterance gives high likelihood in Eq. 3.16, the claimant is more likely to be an impostor; Eq. 3.6 can then be made to emphasize small scores only. On the other hand, when an utterance gives low likelihood in Eq. 3.16, the prior score should be slightly increased to make it close to the optimal prior score (see Figure 3.13) because impostors' utterances can also give low likelihood. The parameter a should be kept small so that system performance is not degraded when stable and reliable pseudo-impostor score models cannot be obtained.

Experiments Based on Cellular Phone Speech

Enrollment Procedures. The enrollment procedures have been detailed in Section 3.3.4. However, in this part of the experiments, the testing speech in the development set was also used to create a 2-center 1-D GMM pseudo-impostor score model (Ω_s in Eq. 3.15).

Results and Discussions. Figure 3.18 shows that data-dependent fusion with prior score adaptation performed better than data-dependent fusion without prior score adaptation and equal-weight fusion. In addition, it suggests that data-dependent fusion (based on Eq. 3.14) with prior score adaptation and score sorting attains the best performance. Specifically, it achieves 9.14% EER and 0.0397 minimum DCF, which represent a 29% and 19% relative reduction in terms of EERs and minimum DCF when compared to the baseline. Table 3.5 summarized the results obtained using equal-weight fusion and data-dependent fusion with and without sorting score

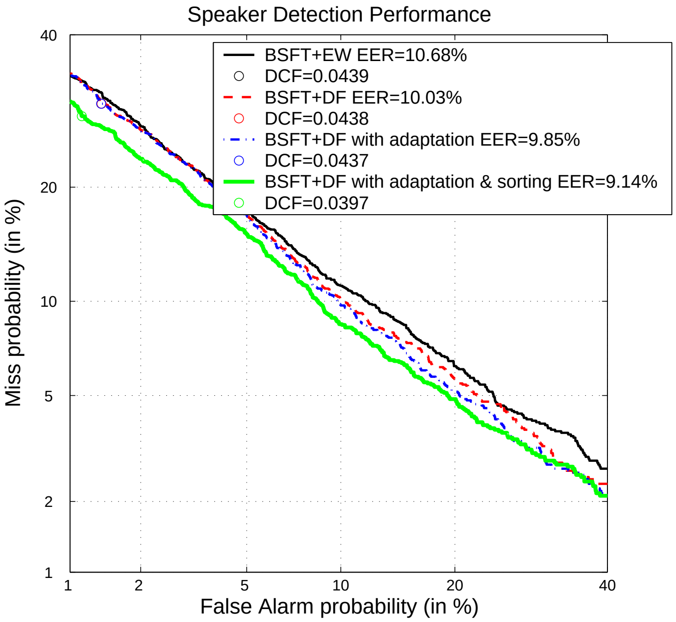


Figure 3.18: Speaker detection performance for data-dependent (DF) fusion and equal-weight (EW) fusion. BSFT stands for blind stochastic feature transformation. For ease of comparison, the labels in the legend are arranged in decreasing EERs.

and prior score adaptation.

Because the NIST 2001 has a standard evaluation protocol, the results reported in this subsection can be compared with those of the short-time Gaussianization approach proposed by Xiang et al. [68].[7] The work of Xiang et al. focuses on feature-level processing to minimize channel distortions. The approach proposed here, on the other hand, first reduces channel distortions at the feature level and then weighs more heavily on those useful scores at the score level. This two-level optimization approach helps reduce the error rate further: 9.14% EER (our fusion approach) versus 10.84% EER (short-time Gaussianization). In addition, data-dependent fusion can achieve a slightly lower minimum detection cost (0.0397) than that of short-time Gaussianization (0.0440).

[7] The short-time Gaussianization approach used 512 mixtures to model the speakers but we used 1024 mixtures in our experiments.

Table 3.5: Speaker detection performance for data-dependent (DF) fusion and equal-weight (EW) fusion. BSFT stands for blind stochastic feature transformation.

Method	**EER(%)**	**Minimum DCF**
EW	12.86	0.0493
DF	12.27	0.0478
BSFT+EW	10.68	0.0439
BSFT+DF	10.03	0.0438
BSFT+DF with sorting using Eq. 3.6	9.87	0.0430
BSFT+DF with sorting using Eq. 3.14	9.28	0.0397
BSFT+DF with adaptation only	9.85	0.0437
BSFT+DF with adaptation + sorting using Eq. 3.6	9.64	0.0429
BSFT+DF with adaptation + sorting using Eq. 3.14	9.14	0.0397

3.5 Concluding Remarks

This chapter has presented a score fusion algorithm that makes use of the prior score statistics and distribution of recognition data. The statistics of pseudo-impostor scores are used to adapt the prior scores in the fusion algorithm. The fusion algorithm was combined with feature transformation for speaker verification. Results based on the HTIMIT corpus and the 2001 NIST evaluation set show that the proposed fusion algorithm outperforms equal-weight fusion. It was found that system performance can be further improved by sorting the scores before fusion and by adapting the prior scores.

Chapter 4

Audio-Visual Biometric Authentication

Various biometric studies have suggested that no single modality can provide an adequate solution for high-security applications. These studies agree that it is vital to use multiple modalities such as visual, infrared, acoustic, chemical sensors, and so on to improve the robustness of biometric systems.

To cope with the limitations of individual biometrics, researchers have proposed using multiple biometric traits concurrently for verification. Such systems are commonly known as multimodal biometric systems [69]. By using multiple biometric traits, systems gain more immunity to intruder attack. For example, it is more difficult for an impostor to impersonate another person using both audio and visual information simultaneously. Multimodal biometrics also helps improve system reliability. For instance, while background noise has a detrimental effect on the performance of voice biometrics, it does not have any influence on face biometrics. Conversely, although the performance of face recognition systems depends on lighting conditions, lighting does not have any effect on voice quality. As a result, audio and visual (AV) biometrics has attracted a great deal of attention in recent years.

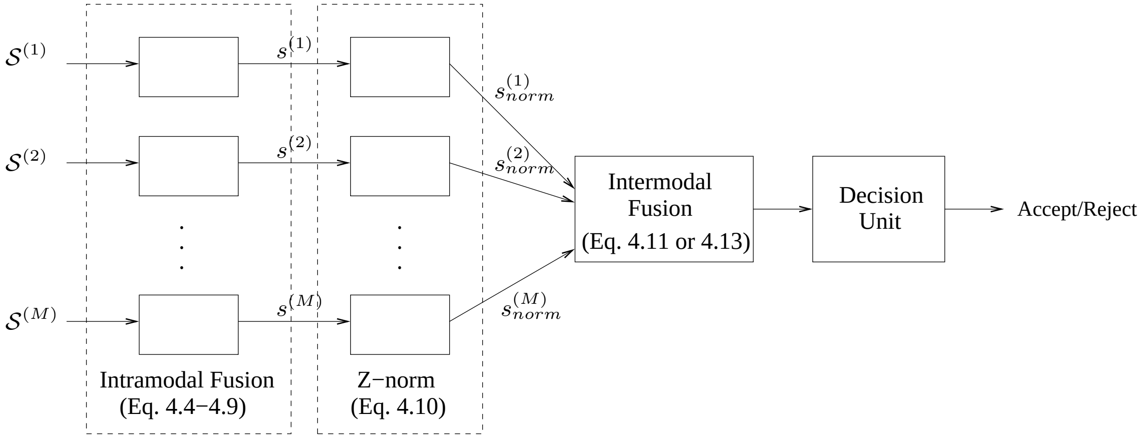


Figure 4.1: Architecture of multi-source multi-sample fusion. *N*ote: $\mathcal{S}^{(m)}$ contains multiple samples obtained from the m-th modality, i.e., $\mathcal{S}^{(m)} = \bigcup_k \mathcal{S}^{(m,k)}$.

In Chapter 3, we have shown that decision fusion techniques can be easily adapted to fuse the scores of a single modality to improve performance. Built on the success of this fusion technique, this chapter proposes a multi-source multi-sample fusion approach to identity verification. Specifically, fusion is performed at two levels: intramodal and intermodal. In intramodal fusion, the scores of multiple samples (e.g., utterances or video shots) obtained from the same modality are linearly combined, where the combination weights are dependent on the difference between the score values and a client-dependent reference score obtained during enrollment. This is followed by intermodal fusion in which the means of intramodal fused scores obtained from different modalities are fused. The final fused score is then used for decision making. Figure 4.1 depicts the architecture of multi-source multi-sample fusion. It was found that making the frame-based intramodal fusion weights data-dependent helps the subsequent intermodal fusion to lower the error rates.

4.1 Intramodal Fusion

4.1.1 Equal-Weight Fusion

Assume that in each verification session, $T^{(m)}$ scores can be obtained from the m-th modality, that is

$$\mathcal{S}^{(m)} = \{s_t^{(m)} \in \Re; t = 1, \ldots, T^{(m)}\}, \tag{4.1}$$

where $s_t^{(m)}$ is the score of frame t. For the special case where $\mathcal{S}^{(m)}$ contains speakers' scores, we have

$$s_t^{(m)} = \log p(\mathbf{x}_t^{(m)}|\Lambda_c^{(m)}) - \log p(\mathbf{x}_t^{(m)}|\Lambda_b^{(m)}), \tag{4.2}$$

where $\Lambda_c^{(m)}$ and $\Lambda_b^{(m)}$ represent the client and background models, respectively [42], and $\mathbf{x}_t^{(m)}$ is the t-th feature vector obtained from the audio channel.

In the *equal-weight* fusion approach [58], the mean score

$$\bar{s}^{(m)} = \frac{1}{T^{(m)}} \sum_{t=1}^{T^{(m)}} s_t^{(m)} \tag{4.3}$$

is used for decision making. The factor $1/T^{(m)}$ in Eq. 4.3 can be considered as a constant weight applying to all scores. In other words, all scores are weighted equally.

4.1.2 Data-Dependent Fusion

Instead of assigning an equal weight to all scores, different weights can be assigned to different scores. This can be achieved by using the data-dependent fusion explained in Mak et al. [60]. Specifically, the approach splits a score sequence $\mathcal{S}^{(m)}$ into K subsequences:

$$\mathcal{S}^{(m,k)} = \{s_t^{(m,k)} \in \Re; t = 1, \ldots, T^{(m)}/K\} \quad k = 1, \ldots, K. \tag{4.4}$$

The frame-level fused scores are then computed as

$$\hat{s}_t^{(m)} = \sum\nolimits_{k=1}^{K} \alpha_t^{(m,k)} s_t^{(m,k)}, \tag{4.5}$$

where $t = 1, \ldots, T^{(m)}/K$, and $\alpha_t^{(m,k)} \in [0, 1]$ represents the confidence (reliability) of the score $s_t^{(m,k)}$. The fusion weights $\alpha_t^{(m,k)}$ are made dependent on both the training data (prior information) and recognition data (scores):

$$\alpha_t^{(m,k)} = \frac{\exp\{(s_t^{(m,k)} - \tilde{\mu}_p^{(m)})^2/2(\tilde{\sigma}_p^{(m)})^2\}}{\sum\nolimits_{l=1}^{K} \exp\{(s_t^{(m,l)} - \tilde{\mu}_p^{(m)})^2/2(\tilde{\sigma}_p^{(m)})^2\}}, \tag{4.6}$$

where $t = 1, \ldots, T^{(m)}/K$ and $k = 1, \ldots, K$. By using enrollment data, the user-dependent prior score $\tilde{\mu}_p^{(m)}$ and prior variance $(\tilde{\sigma}_p^{(m)})^2$ are computed as follows:

$$\tilde{\mu}_p^{(m)} = \frac{K_c \tilde{\mu}_c^{(m)} + K_b \tilde{\mu}_b^{(m)}}{K_c + K_b} \tag{4.7}$$

and

$$(\tilde{\sigma}_p^{(m)})^2 = \frac{1}{K_c + K_b} \sum_{k=1}^{K_c+K_b} \left[\bar{s}^{(m,k)} - \tilde{\mu}_p^{(m)}\right]^2, \tag{4.8}$$

where K_c and K_b are respectively the numbers of client's enrollment samples and pseudo-impostors' samples, $\tilde{\mu}_c^{(m)}$ and $\tilde{\mu}_b^{(m)}$ are respectively the score means of client's and pseudo-impostors' samples, and $\bar{s}^{(m,k)}$ denotes the mean score of the k-th enrollment sample. Finally, the intramodal fused score

$$\hat{s}^{(m)} = \frac{K}{T^{(m)}} \sum\nolimits_{t=1}^{T^{(m)}/K} \hat{s}_t^{(m)} \tag{4.9}$$

is computed for decision making.

4.2 Zero-Normalization

A system that uses a single client-independent decision threshold must ensure that all client and impostor scores have values comparable to the threshold. This requirement can be fulfilled by normalizing the scores so that they fall into a predefined range. One possible approach (called Z-norm [70]) shifts and scales the impostor scores so that their mean and variance become zero and unity, respectively. More specifically, the claimant's score $\bar{s}^{(m)}$ in Eq. 4.3 or $\hat{s}^{(m)}$ in Eq. 4.9 is normalized:

$$s_{norm}^{(m)} = \frac{s^{(m)} - \mu_b^{(m)}}{\sigma_b^{(m)}} \qquad s^{(m)} \in \{\bar{s}^{(m)}, \hat{s}^{(m)}\}, \tag{4.10}$$

where $\mu_b^{(m)}$ and $\sigma_b^{(m)}$ are the mean and standard deviation of pseudo-impostor scores, respectively. These scores can be obtained during training by testing a client model against nontarget observations.

4.3 Intermodal Fusion

Instead of asking the individual modalities to make decisions based on the Z-norm scores, these modality-dependent scores can be further fused to produce more reliable scores. There are many ways to fuse the modality-dependent scores. Typical approaches include (1) the sum rule and product rule in rule-based fusion; and (2) support vector machines, multilayer perceptrons, and binary decision trees in learning-based fusion. Research has shown that the sum rule and support vector machines are generally superior [17, 32, 34, 35].

4.3.1 Sum Rule

Given the intramodal fused scores $s_{norm}^{(m)}$, the intermodal fused score can be computed by the sum rule:

$$s = \sum_{m=1}^{M} \beta_m s_{norm}^{(m)}, \tag{4.11}$$

where β_m is the fusion weight for the m-th modality.

For audio-visual biometrics, the audio-visual score s is obtained by linearly combining the audio score $s^{(A)}$ and visual score $s^{(V)}$:

$$s = \beta s_{norm}^{(A)} + (1 - \beta) s_{norm}^{(V)}, \tag{4.12}$$

where $s_{norm}^{(m)}, m \in \{A, V\}$, is the Z-norm score (Eq. 4.10) and β is a combination weight that can be computed using training data or made dependent on the quality of audio or visual data [12, 24, 26]. The audio and visual scores must have the same range for the fusion to be meaningful. This can be achieved by normalizing the scores, as in Eq. 4.10.

4.3.2 Support Vector Machines

A support vector machine (SVM) [71, 72] is a binary classifier that maps input patterns $\mathbf{x}_i \in \Re^D$ to output labels $y_i \in \{-1, 1\}$, where $i = 1, \ldots, l$, and l is the number of patterns. Generally, an SVM has the form

$$f(\mathbf{x}) = \sum_{j \in \Omega} \alpha_j y_j K(\mathbf{x}, \mathbf{x}_j) + b, \tag{4.13}$$

where α_j are the Lagrange multipliers, Ω contains the indexes to the support vectors for which $\alpha_j \neq 0$, b is a bias term, $\mathbf{x}$ is an input vector to be classified, and $K(\mathbf{x}, \mathbf{x}_j)$ is a kernel function. The most common kernels are:

- Polynomial:

$$K(\mathbf{x}, \mathbf{x}_j) = (\mathbf{x} \cdot \mathbf{x}_j + 1)^p, \quad p > 0; \tag{4.14}$$

- Radial Basis Function:

$$K(\mathbf{x}, \mathbf{x}_j) = \exp(-\|\mathbf{x} - \mathbf{x}_j\|^2 / 2\sigma^2). \tag{4.15}$$

For audio-visual biometrics, $\mathbf{x}$ is typically composed of audio and visual scores (i.e., $\mathbf{x} = [s^{(A)}_{norm} \;\; s^{(V)}_{norm}]^T$) and decisions are based on whether the value $f(\mathbf{x})$ is above or below a threshold. Research has found that the polynomial kernel is superior to the RBF kernel for audio-visual fusion [32].

4.4 Experimental Evaluations

This section explains how the intramodal and intermodal fusion techniques described in the previous sections can be applied to audio-visual biometric authentication.

4.4.1 Audio-Visual Feature Extraction

Audio-Visual Data Sets

The XM2VTSDB corpus [73, 74] was used in the evaluations. XM2VTSDB is an audio-visual corpus designed for biometric research. The corpus consists of the audio and video recordings of 295 subjects taken over a period of four months. We adopted Configuration II as specified by Luettin and Maitre [74] in the evaluation. More precisely, the database was divided into 200 clients, 70 impostors (part of the 95 impostors in DVD003b) for testing, and 25 pseudo-impostors (the remaining impostors in DVD003b) for finding decision thresholds or other system parameters. For each client, the first two sessions were used for training, and the last session was used for testing. Each client was impersonated by 70 impostors using the audio and video data of the four sessions.

Preprocessing of Audio Files

Because the original audio files were captured in a quiet, controlled environment using a high-quality microphone, the error rate using the audio data alone is very low (about 0.7% half-total error rate); as a result, performing audio-visual fusion was unnecessary. Therefore, coder distortion and factory noise were introduced to the sound files in an attempt to simulate a more realistic acoustic environment.

The audio files in the corpus were down-sampled from 32kHz to 8kHz. The down-sampled PCM files were then transcoded by a GSM codec [64]. Factory noise ("factory1.wav" of the NOISE92 database [75]) was added to the test files at a signal-to-noise ratio of 1dB. Therefore, training utterances were distorted by downsampling and GSM transcoding, whereas test utterances were not only distorted by downsampling and transcoding but also contaminated by factory noise. This introduces acoustic mismatch between the training and testing utterances. Nineteen MFCCs and their time derivative (delta MFCCs) were extracted from the files using a 28ms Hamming window at a rate of 71Hz.

We used the training sessions of 200 client speakers in the speaker set to create a 128-center background model. The background model was then adapted to speaker models using MAP adaptation [42]. As defined in Configuration II of XM2VTSDB, two sessions (i.e., four utterances) per speaker were used for model training. Cepstral mean subtraction (CMS) was performed on all MFCCs before they were used for training, testing, and evaluation.

Preprocessing of Video Files

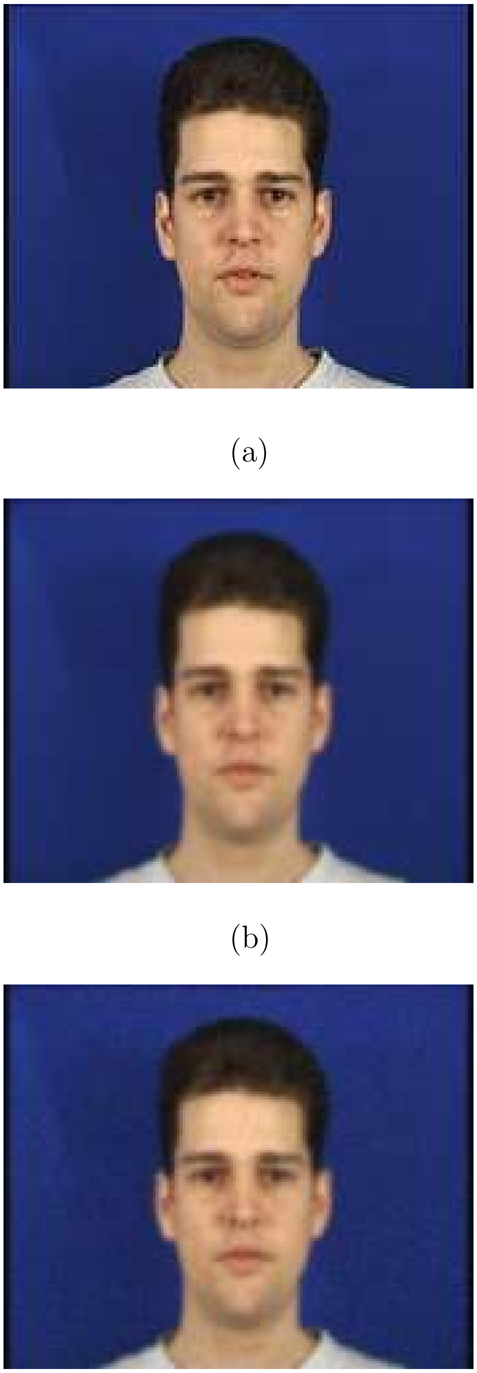
Similar to audio files, the quality of video files in the corpus is also very good, making audio-visual fusion unnecessary (because face verification on the original video data already approaches 0% half-total error rate). As a result, distortion was introduced to the video sequences using PhotoShop Version 7.0 as follows. First, each of the AVI files in the corpus was converted into a sequence of high-quality JPEG files with 720 $\times$ 576 pixels. Second, the frame rate was reduced to one frame per second; for each frame, the JPEG image was downsampled to 176 $\times$ 144 pixels. Third, the image was blurred by setting the "Gaussian Blur" of PhotoShop to 1.0. Finally, Gaussian noise was added to the image by setting the "Gaussian Noise" of PhotoShop to 3.0. Figure 4.2 shows the effect of these operations on the downsampled images.

Both enrollment and verification used the blurred, noise-added images (see Figure 4.2(c)).[1] During verification, a blurred and noise-added image sequence of a claimant was input to a face verification system based on the Identix's Face Verification SDK (FaceIT) [76] to locate the head and compute the scores. The scores range from 0 to 10; the higher the score, the more likely the claimant is genuine.

4.4.2 Audio-Visual Fusion

We assumed that one utterance and one video shot can be obtained from the claimant in a verification session. The utterance and the video shot were divided into two equal-length subutterances and two equal-length subvideo shots, i.e., $K = 2$ and $m \in \{A, V\}$ in Eq. 4.4, where A represents the audio channel and V the video channel. Feeding these subutterances and subvideo shots to the speaker verification system and the face verification system gives two streams of audio scores and two streams of visual scores. We applied intramodal fusion to the two audio score streams

[1] Here we assumed that the same camera was used for both enrollment and verification. Therefore, the constructions of noisy images for both enrollment and verification are the same.



(a)

(b)

(c)

Figure 4.2: Example of preprocessed images. (a) Downsampled image; (b) after Gaussian blurring; (c) after adding Gaussian noise.

and also to the two visual score streams independently to obtain the mean of the fused audio scores, $\hat{s}^{(A)}$, and the mean of the fused visual scores, $\hat{s}^{(V)}$. These scores were further normalized according to Eq. 4.10 to ensure that they have the same range. The client-dependent fusion parameters, including the prior scores and prior variances ($\tilde{\mu}_p^{(m)}$,$(\tilde{\sigma}_p^{(m)})^2$; $m \in \{A, V\}$), were obtained by feeding the utterances and video shots of 25 pseudo-impostors to the client and background models.

The normalized scores of every clients and of the 25 pseudo-impostors were used for training a second-degree polynomial SVM, i.e., $p = 2$ in Eq. 4.14. We used the training data of all clients and all pseudo-impostors to obtain 400 clients scores (2×200) and 40,000 pseudo-impostors scores ($8 \times 25 \times 200$) and used these scores to train a client-independent SVM. Because a client-independent SVM was used, Z-norm was applied to all of the audio and visual scores to ensure that the decision threshold is appropriate for all clients.

During verification, a total of 400 client trials (200 clients × 2 utterances per client) and 120,000 impostor attempts (200 clients × 75 impostors per client × 8 utterances per impostor) were used to test the system.

4.5 Results and Discussions

4.5.1 Intramodal Fusion

Tables 4.1 and 4.2 show the error rates of speaker verification and face verification using different types of intramodal fusion techniques described in Section 4.1, and Figure 4.3 plots the corresponding DET curves. The false acceptance rate (FAR), false rejection rate (FRR), and half-total error rate (HTER) were obtained using a posteriori thresholds.[2] The results show that (1) data-dependent fusion generally

[2] Because of the low resolution in FRR (1/400), the FARs and FRRs are not identical even using posteriori thresholds.

performs better than equal-weight fusion and (2) Z-norm helps lower the HTER of both equal-weight fusion and data-dependent fusion.

4.5.2 Cascading Intramodal and Intermodal Fusion

Figure 4.4 shows the HTERs of two-level audio-visual fusion for different values of the combination weight β in Eq. 4.12. It shows that when β is equal to 0.6, both equal-weight fusion and data-dependent fusion achieve the lowest HTER. However, obtaining this optimum value requires the true identities of claimants to be known prior to verification, which is unrealistic in practice. A better approach is to determine the value of β using training data. Therefore, similar to SVM-based intermodal fusion, the normalized scores of every clients and of the 25 pseudo-impostors were used for determining the value of β. More specifically, the value of β that produces minimum HTER in the training data was used for fusing the audio and visual scores during verification. It was found that $\beta = 0.7$ gives the lowest HTER in the training data set.

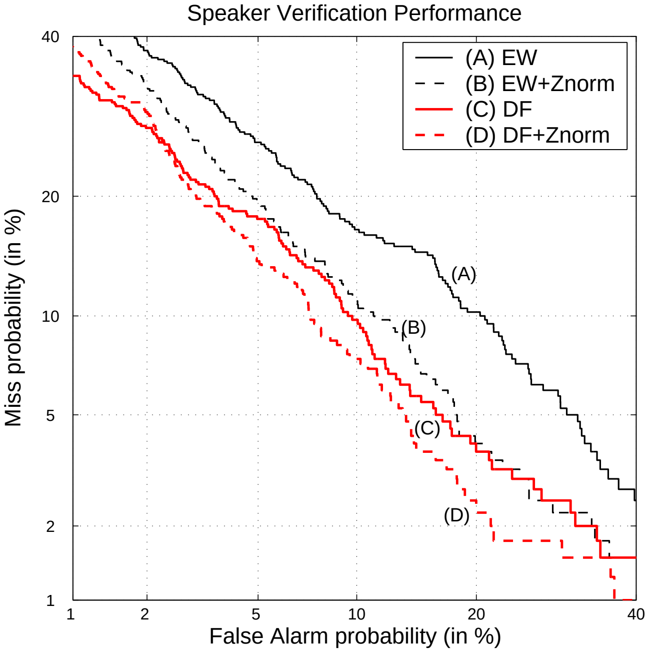
Table 4.3 summarizes the error rates obtained by applying the two-level audio-visual fusion with β in Eq. 4.12 set to 0.7, and Figure 4.5 plots the corresponding DET curves. The results show that data-dependent fusion always performs better than equal-weight fusion. The results also show that applying intermodal fusion (either linear combination or SVMs) on intramodal fused scores can further reduce HTERs. Figure 4.6 plots the decision boundary created by the sum rule and the second-degree polynomial SVM. It shows that the SVM can create a nonlinear boundary to separate the client scores from the impostor scores, which results in lower error rates.

4.5.3 Compared with Related Fusion Approaches

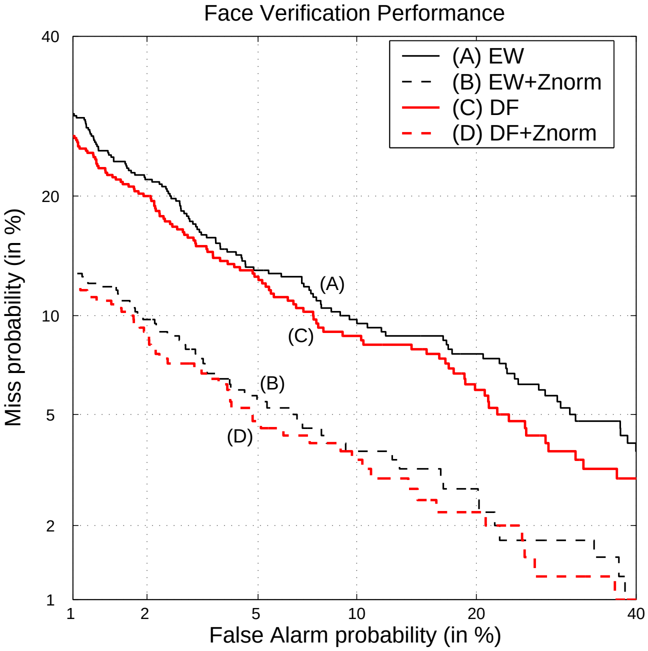
It is of interest to compare the proposed two-level fusion approach with the multi-frame–multi-expert system proposed by Czyz et al. [15]. In their system, two different face verification algorithms were applied to the same facial sequence to create two sets of scores. The frame-based scores were then fused using the minimum rule to obtain two minimum scores, which were subsequently fused using an SVM. The multi-frame part of [15] can be considered as a special case of our data-dependent intramodal fusion because when $\tilde{\mu}_p^{(m)}$ in Eq. 4.6 is large, $\alpha_t^{(m,k)}$ in Eq. 4.5 will be close to 1 for small scores and close to 0 for large scores, which has the effect similar to the minimum rule.

4.6 Concluding Remarks

This chapter has presented an audio-visual biometric authentication system. A novel two-level fusion technique that fuses the scores obtained from speaker and face models was detailed. The proposed technique is general and is applicable to multi-modal biometric systems. This is evident by promising experimental results on the XM2VTSDB audio-visual database. It was found that an error rate reduction of up to 71% can be achieved when the proposed fusion technique is applied to fuse the scores derived from speaker models and face models.



(a)



(b)

Figure 4.3: DET plots of different intramodal fusion techniques in (a) speaker verification and (b) face verification. *EW* stands for equal-weight fusion and *DF* stands for data-dependent fusion. *EW+Znorm* means that Z-norm was performed on the mean fused scores of equal-weight fusion. A similar definition applies to *DF+Znorm*.

Intramodal Fusion Method	**Error Rate**			**Rel. Red. of HTER**
(Applied to voice only)	**FAR**	**FRR**	**HTER**	**(w.r.t. EW fusion)**
EW	14.29%	14.75%	14.52%	N.A.
EW+Znorm	10.08%	10.50%	10.29%	29.13%
DF	9.70%	9.75%	9.73%	32.99%
DF+Znorm	8.39%	8.50%	8.45%	41.80%

Table 4.1: Speaker verification error rates and relative reduction (Rel. Red.) of half-total error rate (HTER) with respective to equal-weight fusion achieved by the speaker verification systems using data-dependent intramodal fusion. *Note*: Fusion takes place only within the audio and visual scores, not between them. *EW+Znorm* (*DF+Znorm*) means that Z-norm was performed on the mean fused scores of equal-weight intramodal fusion (data-dependent intramodal fusion). *EW* and *DF* stand for equal-weight and data-dependent intramodal fusion, respectively.

Intramodal Fusion Method	**Error Rate**			**Rel. Red. of HTER**
(Applied to face only)	**FAR**	**FRR**	**HTER**	**(w.r.t. EW fusion)**
EW	9.53%	9.75%	9.64%	N.A.
EW+Znorm	4.96%	5.50%	5.23%	45.75%
DF	8.01%	9.00%	8.51%	11.72%
DF+Znorm	4.80%	4.75%	4.77%	50.52%

Table 4.2: Face verification error rates and relative reduction (Rel. Red.) of half-total error rate (HTER) with respective to equal-weight fusion achieved by the face verification systems using data-dependent intramodal fusion. *Note*: Fusion takes place only within the audio and visual scores, not between them. *EW+Znorm* (*DF+Znorm*) means that Z-norm was performed on the mean fused scores of equal-weight intramodal fusion (data-dependent intramodal fusion). *EW* and *DF* stand for equal-weight and data-dependent intramodal fusion, respectively.

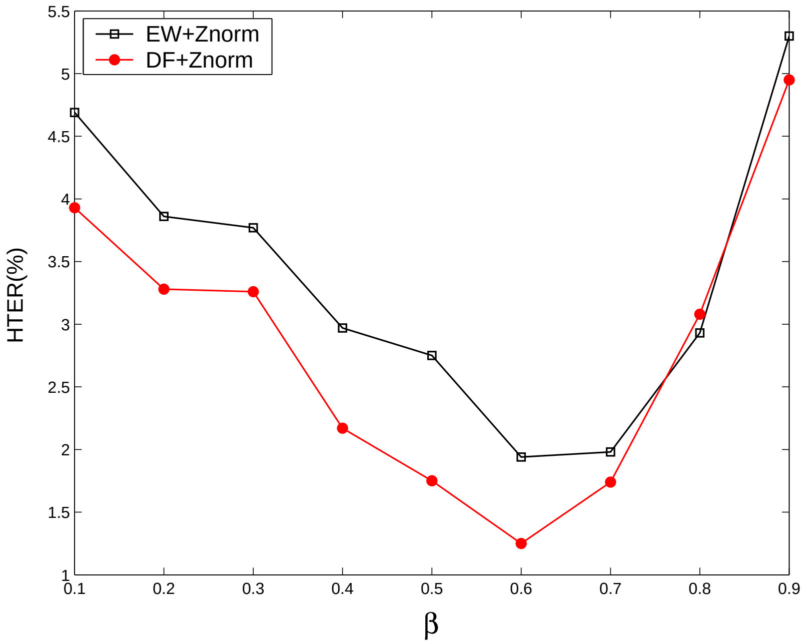


Figure 4.4: HTERs of two-level audio-visual fusion for different values of the combination weight β in Eq. 4.12. *EW* and *DF* stand for equal-weight and data-dependent intramodal fusion, respectively.

Intramodal Fusion Type	**Intermodal Fusion Type**	**Error Rate**			**Rel. Red. of HTER (w.r.t. face only)**
		FAR	**FRR**	**HTER**	
EW+Znorm	Sum Rule	1.95%	2.00%	1.98%	62.14%
DF+Znorm	Sum Rule	1.73%	1.75%	1.74%	66.73%
EW+Znorm	SVM	1.85%	2.00%	1.93%	63.10%
DF+Znorm	SVM	1.53%	1.50%	1.51%	71.13%

Table 4.3: Voice+Face verification error rates and relative error reduction with respect to the HTER of face verification (Table 4.2) obtained by linearly combining the means of intramodal fused scores and by polynomial SVMs. The combination weight β in Eq. 4.12 was set to 0.7. *EW* and *DF* stand for equal-weight and data-dependent intramodal fusion, respectively.

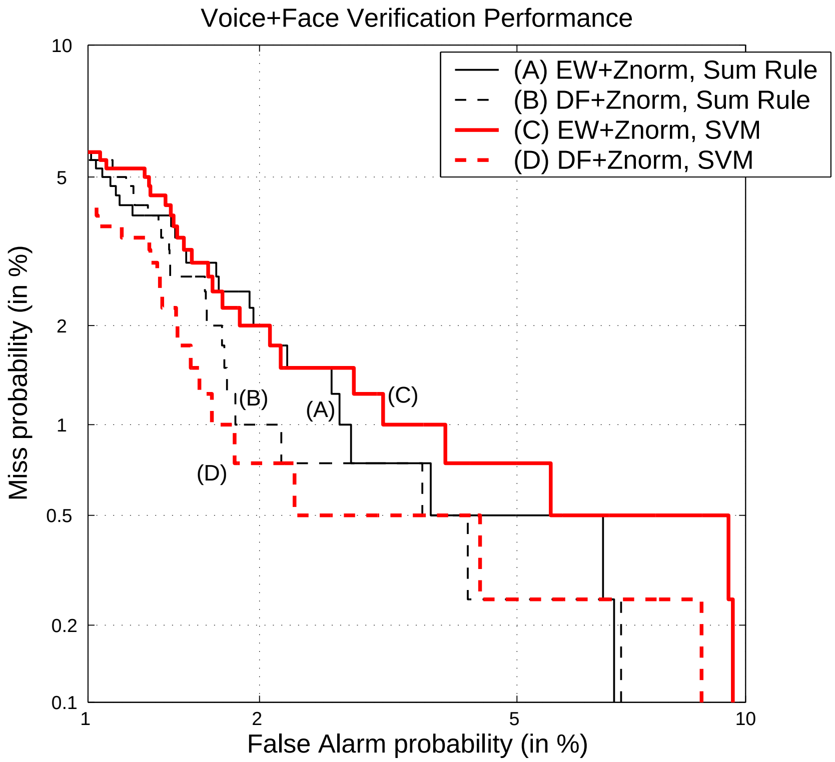


Figure 4.5: DET plots showing the performance of the sum rule and SVMs in fusing the audio and visual scores. *EW* stands for equal-weight intramodal fusion and *DF* stands for data-dependent intramodal fusion. *EW+Znorm* means that Z-norm was performed on the mean fused scores of equal-weight intramodal fusion. A similar definition applies to *DF+Znorm*.

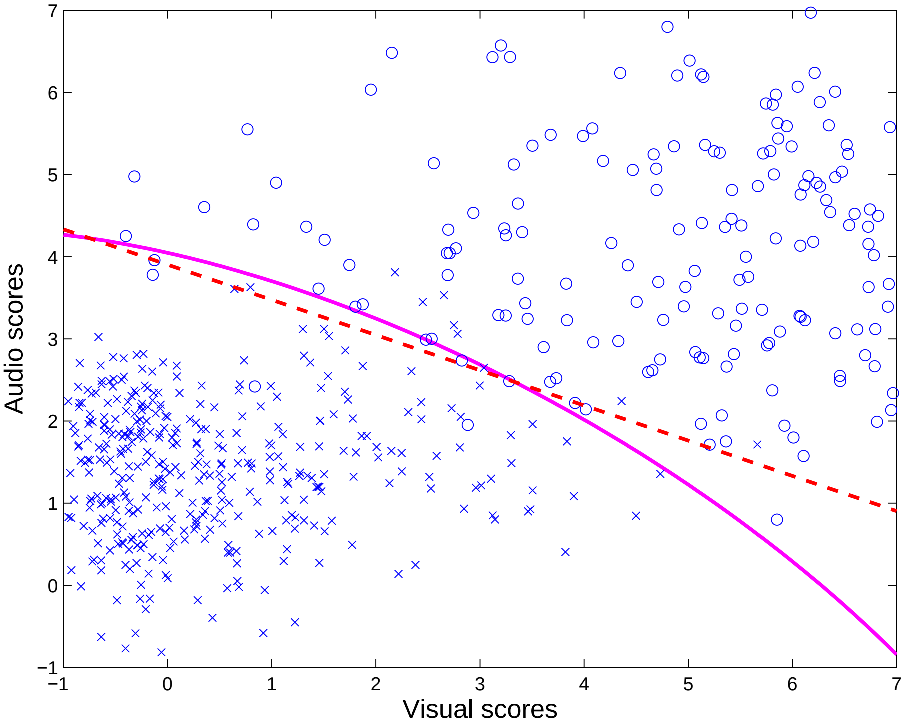


Figure 4.6: Decision boundary created by the sum rule (dashed) and a second-degree polynomial SVM (solid). Circles (○) and Crosses (×) represent clients' and impostors' attempts, respectively.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

This work has investigated several fusion techniques for biometric authentication. Most of today's fusion techniques focus on combining the information from different experts. However, combining information from a single expert is also worth studying. To this end, this work proposed a new fusion approach that combines the scores of a speaker verification system in an attempt to improve its performance. Traditionally, the scores of multiple samples are averaged and the average score is compared against a threshold for decision making. The average score, however, may not be optimal because the score distribution is ignored. To addresses this limitation, a new fusion model was proposed; the model incorporates the score distribution by making the fusion weights dependent on the dispersion between the frame-based scores and the prior score statistics obtained from training data. Because the fusion weights depend on verification data, the positions of scores in the score sequences are detrimental to the final fused scores. Therefore, a score sorting method was introduced to enhance the fusion model. Besides sorting the verification scores, adaptation of prior scores was also applied. During verification, the prior score is adapted based on the likelihood of the claimant being an impostor. The fusion model was evaluated on a speaker verification task using a GSM-transcoded corpus and the NIST2001 evaluation set. Experimental results show that the proposed fusion approaches improve the system performance significantly.

Build on the success of the data-dependent fusion model, this work extended the modal to audio-visual biometric authentication. The technique is different from the conventional ones in that it divides the fusion process into two stages. In the first stage, the method assigns a larger weight to the more reliable scores in a frame-by-frame basis. This weight assignment process is performed on the audio and visual modalities independently. In the second stage, the weighted sum of the frame-based scores from the audio and visual modalities are further fused by either the sum rule or a support vector machine. The proposed technique is general and is applicable to multimodal biometric systems. This is evident by the encouraging experimental results based on the XM2VTSDB corpus.

5.2 Future Work

The data-dependent decision fusion uses prior score statistics to compute intramodal fusion weights without taking the type of sound into account. Knowing the type of sound, however, may lead to a new dimension of determining the fusion weights. For example, we may make use of the fact that certain sounds (e.g., high-energy nasals and vowels) contain more speaker-dependent information than the others and assign larger weights to these sounds accordingly [77, 78]. In such case, the frame-based fusion weights become phoneme-based fusion weights. Another possibility is to weight individual frames according to their degree of voicing, i.e., assign large weights to those frames with a high degree of voicing. The degree of voicing can be easily obtained from the pitch gain parameter of a code-excited linear predictive encoder

[79]. The reliability of individual frames can also be obtained from the posterior probability of observing a particular articulatory feature [80]. In this approach, a high probability of observing an articulatory features suggests that the corresponding frames contain significant speaker-dependent information.

The performance of face-based verification systems is affected by the accuracy of face detection. Therefore, the scores obtained from a face detector can also be used to determine fusion weights.

Bibliography

[1] P. Tikkanen, S. Puolitaival, and I. Kansala, "Capabilities of biometrics for authentication in wireless devices," in *AVBPA 2003*, 2003, pp. 796–804.

[2] C. Bernasconi, "On instantaneous and transitional spectral information for text-dependent speaker verification," *Speech Communication*, vol. 9, pp. 129–139, 1990.

[3] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, 1995.

[4] Y. Adini, Y. Moses, and S. Ullman, "Face recognition: The problem of compensating for changes in illumination direction," *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp. 721–732, 1997.

[5] J. Kittler, G. Matas, K. Jonsson, and M. Sánchez, "Combining evidence in personal identity verification systems," *Pattern Recognition Letters*, vol. 18, no. 9, pp. 845–852, Sept. 1997.

[6] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, pp. 6–23, 1997.

[7] B. V. Dasarathy, "Sensor fusion potential exploitaiton-innovative architectures and illustrative applications," *Proceedings of the IEEE*, pp. 24–38, 1997.

[8] Y. Barniv and D. Casasent, "Multisensor image registration: Experimental verification," in *Proceedings of the SPIE*, 1981, vol. 292, pp. 160–171.

[9] R. R. Tenney and N. R. Sandell, "Detection with distributed sensors," *IEEE Trans. Aerospace Electron. Syst.*, vol. 17, pp. 98–101, 1981.

[10] C. C. Chibelushi, J. S. D. Mason, and F. Deravi, "Feature-level data fusion for bimodal person recognition," in *Proc. 6th Int. Conf. on Image Processing and Its Applications*, 1997, vol. 1, pp. 399–403.

[11] K. Chen, "Speaker modeling with various speech representations," in *ICBA'04*, 2004, pp. 592–599.

[12] C. Neti et al., "Audio-visual speech recognition," in *Final Workshop 2000 Report*, Center for Language and Speech Processing, The Johns Hopkins University, Baltimore, 2000.

[13] K. A. Toh, X. D. Jiang, and W. Y. Yau, "A hyperbolic function model for mutiple biometrics decision fusion," in *ICBA'04*, 2004, pp. 655–662.

[14] K. A. Toh and W. Y. Yau, "Combination of hyperbolic functions for multimodal biometrics data fusion," *IEEE Transactions on System, Man, and Cybernetics—Part B: Cybernetics*, vol. 34, no. 2, Sept. 2004.

[15] J. Czyz, M. Sadeghi, J. Kittler, and L. Vandendorpe, "Decision fusion for face authentication," in *ICBA'04*, 2004, pp. 686–693.

[16] D. Genoud, G. Gravier, F. Bimbot, and G. Chollet, "Combining methods to improve speaker verification decision," in *Proc. ICSLP '96*, 1996, vol. 3, pp. 1756–1759.

[17] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. on Pattern Anal. Machine Intell.*, vol. 20, no. 3, pp. 226–239, 1998.

[18] C. Y. Zheng and Y. H. Yan, "Fusion based speech segmentation in DARPA SPINE2 task," in *Proc. IEEE ICASSP'04*, 2004, pp. I885–I888.

[19] P. K. Varshney, *Handbook of Multisensor Data Fusion*, Springer-Verlag New York, 1997.

[20] L. A. Alexandre, A. C. Campilho, and M. Kamel, "On combining classifers using sum and products rules," *Pattern Recognition Letters*, vol. 22, pp. 1283–1289, 2001.

[21] C. Sanderson and K. K. Paliwal, "Joint cohort normalization in a multi-feature speaker verification system," in *The 10th IEEE International Conference on Fuzzy Systems, 2001*, 2001, vol. 1, pp. 232 –235.

[22] G. Potamianos and C. Neti, "Stream confidence estimate for audio-visual speech recognition," in *Proc. ICSLP'2000*, 2000, vol. 3, pp. 746–749.

[23] S. Pigeon, P. Druyts, and P. Verlinde, "Applying logistic regression to the fusion of the NIST'99 1-speaker submisions," *Digital Signal Processing*, vol. 10, pp. 237–248, 2000.

[24] U. Meier, W. Hurst, and P. Duchnowski, "Adaptive bimodal sensor fusion for automatic speech reading," in *Proc. ICASSP'96*, 1996, pp. 833–836.

[25] M. Chu, M. Yeung, L. H. Liang, and X. X. Liu, "Environment-adpative multi-channel biometrics," in *Proc. IEEE ICASSP'03*, 2003, pp. V788–V791.

[26] C. Sanderson and K. K. Paliwal, "Noise compensation in a person verification system using face and multiple speech features," *Pattern Recognition*, vol. 36, pp. 293–302, 2003.

[27] T. Wark and S. Sridharan, "Adaptive fusion of speech and lip information for robust speaker identification," *Digital Signal Processing*, vol. 11, pp. 169–186, 2001.

[28] C. W. Lau, B. Ma, M. L. Meng, Y. S. Moon, and Y. Yam, "Fuzzy logic decision fusion in a multimodal biometric system," in *ICSLP'04*, 2004.

[29] R. Brunelli and D. Falavigna, "Person identification using multiple cues," *IEEE Trans. on Pattern Anal. Machine Intell.*, vol. 17, no. 10, pp. 955–966, 1995.

[30] B. Maison, C. Neti, and A. Senior, "Audio-visual speaker recognition for video broadcast news: some fusion technques," in *Proc. IEEE 3rd Workshop on Multimedia Signal Processing*, 1999, pp. 161–167.

[31] A. Rogozan and P. Deleglise, "Adaptive fusion of acoustic and visual sources for automatic speech recognition," *Speech Communications*, vol. 26, pp. 149–161, 1998.

[32] S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz, "Fusion of face and speech data for person identity verification," *IEEE Trans. on Neural Networks*, vol. 10, no. 5, pp. 1065–1074, 1999.

[33] J. Czyz, S. Bengio, C. Marcel, and L. Vandendorpe, "Scalability analysis of audio-visual person identity verification," in *AVBPA'03*, 2003, pp. 752–760.

[34] V. Chatzis, A. G. Bors, and I. Pitas, "Multimodal decision-level fusion for person authentication," *IEEE Trans. on Systems, Man and Cybernetics-Part A: Systems and Humans*, vol. 29, no. 6, pp. 674–680, 1999.

[35] J. A. Fierrez, J. G. Ortega, D. R. Garcia, and J. R. Gonzalez, "A comparative evaluation of fusion strategies for multimodal biometric verification," in *AVBPA 2003*, 2003, pp. 830–836.

[36] J. A. Fierrez, D. R. Garcia, J. G. Ortega, and J. R. Gonzalez, "Exploiting general knowledge in user-dependent fusion strategies for multimodal biometric verification," in *Proc. IEEE ICASSP'04*, 2004, pp. V617–V620.

[37] J. P. Campbell, "Speaker recognition: A tutorial," *Proceedings of the IEEE*, vol. 85, pp. 1437–1461, 1997.

[38] J. Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE*, vol. 63, pp. 561–580, 1975.

[39] K. T. Assaleh and R. J. Mammone, "New LP-derived features for speaker identification," *IEEE Trans. on Speech and Audio Processing*, vol. 2, no. 4, pp. 630–638, 1994.

[40] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. on ASSP*, vol. 28, no. 4, pp. 357–366, August 1980.

[41] A. Reynolds, "Speaker identification and verification using gaussiam mixture speaker models," *Speech Communication*, vol. 17, pp. 91–108, 1995.

[42] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19–41, 2000.

[43] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. of Royal Statistical Soc., Ser. B.*, vol. 39, no. 1, pp. 1–38, 1977.

[44] A. E. Rosenberg, J. DeLong, C. H. Lee, B. H. Juang, and F. K. Soong, "The use of cohort normalized scores for speaker verification," in *Proc. ICSLP'92*, 1992, vol. 2, pp. 599–602.

[45] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker for automatic speaker identification and verification," *J. Acoust. Soc. Am.*, vol. 55, no. 6, pp. 1304–1312, 1974.

[46] R. J. Mammone, X. Zhang, and R. P. Ramachandran, "Robust speaker recognition," *IEEE Signal Processing Magazine*, pp. 58–71, September 1996.

[47] A. Sankar and C. H. Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 4, no. 3, pp. 190–202, 1996.

[48] M. W. Mak and S. Y. Kung, "Combining stochastic feature transformation and handset identification for telephone-based speaker verification," in *Proc. ICASSP'02*, 2002, pp. I701–I704.

[49] C.L. Tsang, M. W. Mak, and S.Y. Kung, "Divergence-based out-of-class rejection for telephone handset identification," in *Proc. ICSLP'02*, 2002, pp. 2329–2332.

[50] M. W. Mak, C. L. Tsang, and S. Y. Kung, "Stochastic feature transformation with divergence-based out-of-handset rejection for robust speaker verification," *J. on Applied Signal Processing*, vol. 2004, no. 4, April 2004.

[51] Eric W.M. Yu, M. W. Mak, C. H. Sit, and S.Y. Kung, "Speaker verification based on G.729 and G.723.1 coder parameters and handset mismatch compensation," in *Eurospeech'03*, 2003, pp. 1681–1684.

[52] M. W. Mak, C. H. Sit, and S. Y. Kung, "Extraction of speaker features from different stages of DSR front-ends for distributed speaker verification," *International Journal of Speech Technology*, to appear.

[53] K. K. Yiu, M. W. Mak, M. C. Cheung, and S. Y. Kung, "A new approach to channel robust speaker verification via constrained stochastic feature transformation," in *Proc. ICSLP'04*, 2004.

[54] K. K. Yiu, M. W. Mak, M. C. Cheung, and S. Y. Kung, "Blind stochastic feature transformation for channel robust speaker verification," *J. of VLSI Singal Processing*, to appear.

[55] S.Y. Kung, M.W. Mak, and S.H. Lin, *Biometric Authentication: A Machine Learning Approach*, Prentice Hall, 2004.

[56] J. Egan and P. James, *Signal detection theory and ROC analysis*, Academic Press, 1975.

[57] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection task performance," in *Proc. Eurospeech '97*, 1997, pp. 1895–1898.

[58] N. Poh, S. Bengio, and J. Korczak, "A multi-sample multi-source model for biometric authentication," in *Proc. IEEE 12th Workshop on Neural Networks for Signal Processing*, 2002, pp. 375–384.

[59] Eric W.M. Yu, M. W. Mak, and S.Y. Kung, "Speaker verification from coded telephone speech using stochastic feature transformation and handset identification," in *The 3rd IEEE Pacific-Rim Conference on Multimedia 2002*, 2002, pp. 598–606.

[60] M. W. Mak, M. C. Cheung, and S. Y. Kung, "Robust speaker verification from GSM-transcoded speech based on decision fusion and feature transformation," in *Proc. IEEE ICASSP'03*, 2003, pp. II745–II748.

[61] M. C. Cheung, M. W. Mak, and S. Y. Kung, "Multi-sample data-dependent fusion of sorted score sequences for biometric verification," in *Proc. IEEE ICASSP'04*, 2004, pp. V681–V684.

[62] D. A. Reynolds, "HTIMIT and LLHDB: speech corpora for the study of handset transducer effects," in *ICASSP'97*, 1997, vol. 2, pp. 1535–1538.

[63] M. C. Cheung, M. W. Mak, and S. Y. Kung, "Adaptive decision fusion for multi-sample speaker verification over GSM networks," in *Eurospeech'03*, 2003, pp. 2969–2972.

[64] European Telecommunication Standards Institute, *European digital telecommunications system (Phase 2); Full rate speech; Part 2: Transcoding (GSM 06.10 version 4.1.1)*, 1998.

[65] "The NIST year 2001 speaker recognition evaluation plan," in *http://www.nist.gov/speech/tests/spk/2001/*.

[66] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. ASSP-29, no. 2, pp. 254–272, 1981.

[67] M. Przybocki and A. Martin, "NIST's assessment of text independent speaker recognition performance 2002," in *The Advent of Biometircs on the Internet, A COST 275 Workshop*, Rome, Italy, Nov. 2002.

[68] B. Xiang, U. Chaudhari, J. Navratil, G. Ramaswamy, and R. Gopinath, "Short-time Gaussianization for robust speaker verification," in *Proc. IEEE ICASSP'02*, 2002, vol. 1, pp. 681–684.

[69] J. Kittler, G. Matas, K. Jonsson, and M. Sánchez, "Combining evidence in personal identity verification systems," *Pattern Recognition Letters*, vol. 18, no. 9, pp. 845–852, Sept. 1997.

[70] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for text-independent speaker verification systems," *Digital Signal Processing*, vol. 10, pp. 42–54, 2000.

[71] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.

[72] T. Joachims, "Making large-scale SVM learning practical," in *Advances in Kernel Methods – Support Vector Learning*, B. Scholkopf, C. Burges, and A. Smola, Eds. MIT-Press, 1999.

[73] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre, "XM2VTSDB: The extended M2VTS database," in *AVBPA 1999*, Washington D.C., 1999.

[74] J. Luettin and G. Maitre, "Evaluation protocol for the extended M2VTS database," Tech. Rep., IDIAP, Martigny, Valais, Switzerland, Oct. 1998.

[75] http://spib.rice.edu/spib/select_noise.html

[76] *SDK Developer's Guide FaceIT Verification*, 2003.

[77] S. J. Ahn, S. M. Kang, and H. S. Ko, "On effective speaker verification based on subword model," in *ICSLP'02*, 2002, pp. 1361–1364.

[78] S. M. Chan and M. H. Siu, "Discrimination power weighted subword-based speaker verification," in *Proc. IEEE ICASSP'04*, 2004, pp. I45–I48.

[79] J. P. Campbell, T. E. Tremain, and V. C. Welch, "The federal standard 1016 4800 bps CELP voice coder," *Digital Signal Processing*, vol. 1, pp. 145–155, 1991.

[80] K. Y. Leung, M. W. Mak, and S. Y. Kung, "Articulatory feature-based conditional pronunciation modeling for speaker verification," in *Proc. ICSLP'04*, 2004, pp. 516–519.

Appendix A

The mean fused score can only be proved to be probabilistically larger than the mean scores of the two utterances. Below is a proof:

The mean fused score is given by

$$\begin{aligned}
s(\mathcal{U};\Lambda) &= \frac{1}{T}\sum_{t=1}^{T} s(\mathbf{O}_t;\Lambda) \\
&= \frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K} \alpha_t^{(k)} s_t^{(k)} \\
&= \frac{1}{T}\sum_{t=1}^{T}(\alpha_t^{(1)} s_t^{(1)} + \alpha_t^{(2)} s_t^{(2)}), \quad \text{for } K=2 \\
&= \frac{1}{T}\sum_{t=1}^{T}(\alpha_t^{(1)} s_t^{(1)} + \alpha_t^{(2)} s_t^{(2)}) + \frac{1}{2}\left[\frac{1}{T}\sum_{t=1}^{T}(s_t^{(1)} + s_t^{(2)})\right] - \frac{1}{2}\left[\frac{1}{T}\sum_{t=1}^{T}(s_t^{(1)} + s_t^{(2)})\right] \\
&= \frac{1}{T}\sum_{t=1}^{T}\left[\alpha_t^{(1)} s_t^{(1)} + (1-\alpha_t^{(1)}) s_t^{(2)}\right] + \mu - \frac{1}{T}\sum_{t=1}^{T}\left[\frac{1}{2}(s_t^{(1)} + s_t^{(2)})\right] \\
&= \mu + \frac{1}{T}\sum_{t=1}^{T}\left[(\alpha_t^{(1)} - \frac{1}{2})(s_t^{(1)} - s_t^{(2)})\right], \qquad \text{(A.1)}
\end{aligned}$$

where

$$(\alpha_t^{(1)} - \frac{1}{2})(s_t^{(1)} - s_t^{(2)}) \begin{cases} > 0 \text{ when } s_t^{(1)} > s_t^{(2)} > \tilde{\mu}_p \text{ or } s_t^{(2)} > s_t^{(1)} > \tilde{\mu}_p \\ < 0 \text{ when } s_t^{(1)} < s_t^{(2)} < \tilde{\mu}_p \text{ or } s_t^{(2)} < s_t^{(1)} < \tilde{\mu}_p \end{cases} . \qquad \text{(A.2)}$$

According to Figure 3.5(a), when $s_t^{(1)} > s_t^{(2)} > \tilde{\mu}_p$, $\alpha_t^{(1)}$ will be larger than 0.5. Therefore, $(\alpha_t^{(1)} - \frac{1}{2})(s_t^{(1)} - s_t^{(2)})$ will be larger than 0. Similarly, when $s_t^{(2)} > s_t^{(1)} > \tilde{\mu}_p$, $\alpha_t^{(1)}$ will be smaller than 0.5. This also makes $(\alpha_t^{(1)} - \frac{1}{2})(s_t^{(1)} - s_t^{(2)})$ larger than 0. A similar proof can be applied to the cases where smaller scores are emphasized.

By Condition A-2, the probability of emphasizing large scores is larger than the probability of emphasizing small scores. This leads to

$$P\left(\frac{1}{T}\sum_{t=1}^{T}\left[(\alpha_t^{(1)} - \frac{1}{2})(s_t^{(1)} - s_t^{(2)})\right] > 0\right) > P\left(\frac{1}{T}\sum_{t=1}^{T}\left[(\alpha_t^{(1)} - \frac{1}{2})(s_t^{(1)} - s_t^{(2)})\right] < 0\right).$$

$$\because \text{Eq. A.2}$$

As a result, we have

$$P(s(\mathcal{U};\Lambda) > \mu) > P(s(\mathcal{U};\Lambda) < \mu). \quad \because \text{Eq. A.1}$$