Speech and Audio Research Laboratory
School of Engineering Systems

AUTOMATIC SPEAKER RECOGNITION
UNDER ADVERSE CONDITIONS

Robert J. Vogt
B.Eng(Hons), B.InfoTech(Dist)

SUBMITTED AS A REQUIREMENT OF
THE DEGREE OF
DOCTOR OF PHILOSOPHY
AT THE
QUEENSLAND UNIVERSITY OF TECHNOLOGY
BRISBANE, QUEENSLAND
NOVEMBER 2006
Keywords

Speaker recognition, speaker verification, Bayes factor, mismatch, session variability, feature mapping, confidence measures.
Abstract

Speaker verification is the process of verifying the identity of a person by analysing their speech. There are several important applications for automatic speaker verification (ASV) technology including suspect identification, tracking terrorists and detecting a person's presence at a remote location in the surveillance domain, as well as person authentication for phone banking and credit card transactions in the private sector. Telephones and telephony networks provide a natural medium for these applications.

The aim of this work is to improve the usefulness of ASV technology for practical applications in the presence of adverse conditions. In a telephony environment, background noise, handset mismatch, channel distortions, room acoustics and restrictions on the available testing and training data are common sources of errors for ASV systems.

Two research themes were pursued to overcome these adverse conditions: Modelling mismatch and modelling uncertainty.

To directly address the performance degradation incurred through mismatched conditions it was proposed to directly model this mismatch. Feature mapping was evaluated for combating handset mismatch and was extended through the use of a blind clustering algorithm to remove the need for accurate handset labels for the training data. Mismatch modelling was then generalised by explicitly modelling the session conditions as a constrained offset of the speaker model means. This session variability modelling approach enabled the modelling of arbitrary sources of mismatch, including handset type, and halved the error rates in many cases.

Methods to model the uncertainty in speaker model estimates and verification
scores were developed to address the difficulties of limited training and testing data. The Bayes factor was introduced to account for the uncertainty of the speaker model estimates in testing by applying Bayesian theory to the verification criterion, with improved performance in matched conditions.

Modelling the uncertainty in the verification score itself met with significant success. Estimating a confidence interval for the “true” verification score enabled an order of magnitude reduction in the average quantity of speech required to make a confident verification decision based on a threshold. The confidence measures developed in this work may also have significant applications for forensic speaker verification tasks.
# Contents

**List of Tables**  
x  
**List of Figures**  
xiii  
**List of Abbreviations**  
xvii  
**Authorship**  
xxi  
**Acknowledgements**  
xxiii

## 1 Introduction

1.1 The Voice as a Biometric .......................... 1  
1.1.1 Speaker Recognition ........................... 2  
1.2 Aims and Scope .................................. 4  
1.3 Thesis Structure ................................. 6  
1.4 Original Contributions of this Thesis .......... 7  
1.4.1 Modelling Mismatch .......................... 7  
1.4.2 Modelling Uncertainty ......................... 9  
1.5 Publications .................................... 10

## 2 An Overview of Speaker Verification Technology  

2.1 Introduction .................................... 13  
2.2 Evaluation of Speaker Verification .............. 14  
2.2.1 NIST Speaker Recognition Evaluation Protocols ... 14  
2.2.2 Early Corpora for Speaker Verification ........ 19  
2.2.3 The Switchboard Series of Corpora .............. 20
3.4 Speaker Verification using Bayes Factor Scoring ........................................... 70
  3.4.1 Bayes Factor Scoring for Gaussian Mixture Models .................................. 70
  3.4.2 The Role of $\tau$ ..................................................................................... 74
  3.4.3 Test Frame Weighting .............................................................................. 74
  3.4.4 Implementation ....................................................................................... 75
3.5 Experiments ........................................................................................................ 76
  3.5.1 NIST 1999 Experiments ........................................................................... 76
  3.5.2 QUT EDT 2003 Experiments .................................................................... 79
3.6 Summary ............................................................................................................. 83

4 Handset Mismatch in Speaker Verification ............................................................. 85
  4.1 Introduction ..................................................................................................... 85
  4.2 Mismatch and Performance ........................................................................... 86
    4.2.1 Approaches to Overcoming Mismatch ..................................................... 90
  4.3 Feature Mapping ........................................................................................... 91
    4.3.1 Comparison to Speaker Model Synthesis .............................................. 92
    4.3.2 Combining Feature Mapping and Warping ............................................ 93
  4.4 Blind Feature Mapping .................................................................................. 97
    4.4.1 Clustered Feature Mapping Training ................................................... 98
    4.4.2 Experiments ......................................................................................... 100
  4.5 Summary ....................................................................................................... 107

5 Explicit Modelling of Session Variability ............................................................ 109
  5.1 Introduction ................................................................................................. 109
  5.2 Aims and Motivation .................................................................................... 111
  5.3 Modelling Session Variability .................................................................... 113
  5.4 Estimating the Speaker Model Parameters ................................................ 116
    5.4.1 Speaker Model Enrolment ................................................................. 117
    5.4.2 MAP Estimation in a GMM Mean Subspace ....................................... 118
    5.4.3 Simultaneous Relevance MAP and Subspace MAP Estimation .......... 119
    5.4.4 Gauss-Seidel Approximation Method ................................................ 126

vii
5.5 Verification .................................................. 133
  5.5.1 Top-N ELLU with Session Variability ................. 133
  5.5.2 Bayes Factors .......................................... 136
  5.5.3 Factor Analysis Likelihood Ratio ....................... 137
5.6 Training the Session Variability Subspace ............... 139
  5.6.1 Principal Components of Session Variability .......... 139
  5.6.2 Iterative Optimisation ................................ 140
5.7 Experiments ................................................. 142
  5.7.1 Switchboard-II Results ................................. 142
  5.7.2 Mixer Results .......................................... 147
  5.7.3 Session Subspace Size ................................. 150
  5.7.4 Comparison of Training Methods ....................... 152
  5.7.5 Comparison of Verification Methods .................. 157
  5.7.6 Sensitivity to the Session Variability Subspace .... 159
  5.7.7 Reduced Test Utterance Length ......................... 160
5.8 Discussion .................................................. 162
5.9 Summary .................................................... 163

6 Modelling Uncertainty in Verification Scores ............... 165
  6.1 Introduction ............................................... 165
  6.2 Confidence Measures ...................................... 166
  6.3 Background ............................................... 169
    6.3.1 Baseline Acoustic Speaker Verification System .... 169
    6.3.2 The Effect of Short Verification Utterances ....... 169
  6.4 Early Verification Decision Scoring Method .............. 171
    6.4.1 Experimental Setup ................................... 173
    6.4.2 Naive Variance Estimate .............................. 174
    6.4.3 Estimate with Correlation ............................. 180
    6.4.4 Robustly Estimating the Sample Variance .......... 184
    6.4.5 Verification Score Normalisation .................... 188
  6.5 Discussion and Future Directions ......................... 193
# List of Tables

<table>
<thead>
<tr>
<th>Table Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Summary of the corpora and conditions for the series of NIST Speaker Recognition Evaluations from 1996 to 2005</td>
<td>16</td>
</tr>
<tr>
<td>2.2</td>
<td>Standard detection cost function parameter values</td>
<td>23</td>
</tr>
<tr>
<td>3.1</td>
<td>Minimum DCF and EER of 1-side QUT EDT ’03 baseline results comparing the same handset type and different handset type performance using ELLR and Bayes factor scoring</td>
<td>82</td>
</tr>
<tr>
<td>3.2</td>
<td>Further categorising the results of Table 3.1 on the handset type used for training and testing</td>
<td>84</td>
</tr>
<tr>
<td>4.1</td>
<td>Minimum DCF and EER for standard and clustered feature mapping configurations on the QUT EDT ’03 protocol</td>
<td>101</td>
</tr>
<tr>
<td>4.2</td>
<td>The correspondence between handset and gender labels and final context membership for the randomly seeded, 8 context system</td>
<td>103</td>
</tr>
<tr>
<td>4.3</td>
<td>Minimum DCF and EER of clustered feature mapping in biased and unbiased configurations on the QUT EDT ’03 protocol</td>
<td>104</td>
</tr>
<tr>
<td>5.1</td>
<td>Minimum DCF and EER of the baseline system and session variability modelling on Switchboard-II data</td>
<td>144</td>
</tr>
<tr>
<td>5.2</td>
<td>Minimum DCF and EER of the baseline system and session variability modelling on Mixer data</td>
<td>147</td>
</tr>
<tr>
<td>5.3</td>
<td>Minimum DCF and EER results when varying the number of session subspace dimensions, $R_z$</td>
<td>151</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.4</td>
<td>Minimum DCF and EER for variations on the Gauss-Seidel training method and independent estimation of the speaker and session variables for the female subset of QUT 2004 protocol</td>
<td>153</td>
</tr>
<tr>
<td>5.5</td>
<td>Comparison of minimum DCF and EER of session modelling and baseline systems for common evaluation condition of the NIST 2005 protocol</td>
<td>157</td>
</tr>
<tr>
<td>5.6</td>
<td>Comparison of minimum DCF and EER for variations on the top-N ELLR verification scoring method for the female subset of the QUT 2004 protocol</td>
<td>158</td>
</tr>
<tr>
<td>5.7</td>
<td>Minimum DCF and EER results with varying degrees of convergence in the session variability subspace training</td>
<td>159</td>
</tr>
<tr>
<td>6.1</td>
<td>The effect of shortened utterances on speaker verification performance</td>
<td>170</td>
</tr>
<tr>
<td>6.2</td>
<td>Verification results using the naive method at the EER operating point</td>
<td>176</td>
</tr>
<tr>
<td>6.3</td>
<td>Verification results using the naive method with the minimum DCF operating point</td>
<td>180</td>
</tr>
<tr>
<td>6.4</td>
<td>Verification results comparing the naive and decorrelated methods at the EER operating point</td>
<td>182</td>
</tr>
<tr>
<td>6.5</td>
<td>Verification results comparing the naive and decorrelated methods at the minimum DCF operating point</td>
<td>184</td>
</tr>
<tr>
<td>6.6</td>
<td>Verification results incorporating a priori information in the variance estimate at the EER operating point</td>
<td>185</td>
</tr>
<tr>
<td>6.7</td>
<td>Verification results incorporating a priori information in the variance estimate at the minimum DCF operating point</td>
<td>186</td>
</tr>
<tr>
<td>6.8</td>
<td>The effect of shortened utterances on speaker verification performance using Z-Norm score normalisation</td>
<td>189</td>
</tr>
<tr>
<td>6.9</td>
<td>Verification results using the decorrelated method at the EER operating point using Z-Norm score normalisation</td>
<td>192</td>
</tr>
</tbody>
</table>
List of Figures

2.1 Example DET plot. ................................................. 23

2.2 DET plot showing contours of constant DCF value according to
the parameters in Table 2.2. ........................................ 24

2.3 Feature extraction procedure. .................................. 45

2.4 Plot of Detection Cost versus the GMM order for different param-
eterisations and adaptation approaches (1- and 3-iteration) using
all NIST 1999 male tests. ........................................... 47

2.5 Plot of EER versus the GMM order for different parameterisations
and adaptation approaches (1- and 3-iteration) using all NIST 1999
male tests. ......................................................... 48

2.6 Plot of UBM average inter-component distances versus GMM order
for different parameterisations. .................................. 49

3.1 DET plot of NIST ’99 baseline results comparing ELLR and Bayes
factor scoring ($\beta = 0.25$) for the All, Same and Different handset
type conditions. .................................................. 77

3.2 Minimum DCF values for NIST ’99 baseline results comparing
ELLR to Bayes factor scoring with varying $\beta$-values for the All,
Same and Different handset type conditions. .................. 78

3.3 EER for NIST ’99 baseline results comparing ELLR to Bayes fac-
tor scoring with varying $\beta$-values for the All, Same and Different
handset type conditions. ........................................... 79
3.4 DET plot of NIST ’99 H-Norm results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the All, Same and Different handset type conditions. ................................. 80

3.5 DET plot of NIST ’99 HT-Norm results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the All, Same and Different handset type conditions. ................................. 80

3.6 DET plot of QUT EDT ’03 baseline results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the 1-side, 3-side and 8-side training conditions. ................................. 81

3.7 DET plot of QUT EDT ’03 baseline results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the 1-side, 3-side and 8-side training conditions. ................................. 83

4.1 Example of the performance difference between matched and mismatched conditions for the baseline system on the NIST 1999 protocol. .................................................. 87

4.2 An examination of the effect of handset type combinations of the training and testing utterances for the baseline system on the QUT EDT ’03 protocol. ................................. 89

4.3 The feature extraction process incorporating feature mapping and feature warping. ................................................................. 95

4.4 DET plot of the system with feature mapping and feature warping compared to a reference system for the 1-side condition of the QUT EDT ’03 protocol development split. ................................. 96

4.5 DET plot of the standard and clustered feature mapping configurations on the QUT EDT ’03 protocol. ................................. 102

4.6 DET plot of clustered feature mapping in biased and unbiased configurations on the QUT EDT ’03 protocol. ................................. 105

4.7 DET plot of the development system for the 2004 NIST SRE. ................................. 107
5.1 Plot of the speaker mean offset supervector magnitude, $|y(s)|$, for differing optimisation techniques as it evolves over iterations of the E-M algorithm. .............................................................. 131

5.2 Plot of the session variability vector magnitude, $|z_h(s)|$, for differing optimisation techniques as it evolves over iterations of the E-M algorithm. .............................................................. 132

5.3 Plot of the expected log-likelihood of the training data for differing optimisation techniques as it evolves over iterations of the E-M algorithm. .............................................................. 132

5.4 DET plot of the 1-side training condition for the baseline system and session variability modelling on Switchboard-II data. .............. 143

5.5 DET plot of the 3-side training condition for the baseline system and session variability modelling on Switchboard-II data. .............. 145

5.6 Comparison of session variability modelling and blind feature mapping for the 1-side training condition. .............................. 146

5.7 DET plot of the 1-side training condition for the baseline system and session variability modelling on Mixer data. .............. 148

5.8 DET plot of the 3-side training condition for the baseline system and session variability modelling on Mixer data. .............. 149

5.9 DET plot of the 1-side condition when varying the number of session subspace dimensions, $R_z$ with and without ZT-Norm score normalisation .............................................................. 151

5.10 DET plot for the 1- and 3-side training conditions comparing an optimised session modelling system with a baseline GMM-UBM system with score normalisation applied to both. .............................................................. 155

5.11 DET of for the 1- and 3-side training conditions comparing an optimised session modelling system with a baseline GMM-UBM system with score normalisation applied to both for the common evaluation condition of the NIST SRE 2005 protocol. .............................................................. 156

5.12 DET plot for the 1-side training condition comparing baseline and session modelling results for short test utterance lengths. .............. 161
## List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASR</td>
<td>Automatic Speech Recognition</td>
</tr>
<tr>
<td>ASV</td>
<td>Automatic Speaker Verification</td>
</tr>
<tr>
<td>CDF</td>
<td>Cumulative Density Function</td>
</tr>
<tr>
<td>CDMA</td>
<td>Code Division Multiple Access</td>
</tr>
<tr>
<td>CLSP</td>
<td>Center for Language and Speech Processing</td>
</tr>
<tr>
<td>CMN</td>
<td>Cepstral Mean Normalisation</td>
</tr>
<tr>
<td>CMS</td>
<td>Cepstral Mean Subtraction</td>
</tr>
<tr>
<td>DARPA</td>
<td>Defense Advanced Research Projects Agency</td>
</tr>
<tr>
<td>DCF</td>
<td>Detection Cost Function</td>
</tr>
<tr>
<td>DET</td>
<td>Detection Error Trade-off plot</td>
</tr>
<tr>
<td>DETAC</td>
<td>Detection Error Trade-off Analysis Criterion</td>
</tr>
<tr>
<td>DNDT</td>
<td>Different Number, Different Type</td>
</tr>
<tr>
<td>DNST</td>
<td>Different Number, Same Type</td>
</tr>
<tr>
<td>EARS</td>
<td>Effective, Affordable, Reusable Speech-to-text project</td>
</tr>
<tr>
<td>EDT</td>
<td>Extended Data Task</td>
</tr>
<tr>
<td>EER</td>
<td>Equal Error Rate</td>
</tr>
<tr>
<td>ELLR</td>
<td>Expected Log-Likelihood Ratio</td>
</tr>
<tr>
<td>E-M</td>
<td>Expectation-Maximisation algorithm</td>
</tr>
<tr>
<td>GDF</td>
<td>Group Delay Function</td>
</tr>
<tr>
<td>GMM</td>
<td>Gaussian Mixture Model</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td>GMM-UBM</td>
<td>The GMM with UBM verification structure</td>
</tr>
<tr>
<td>GSM</td>
<td>Global System for Mobile communications</td>
</tr>
<tr>
<td>HMM</td>
<td>Hidden Markov Model</td>
</tr>
<tr>
<td>HOS</td>
<td>Higher-Order Spectra</td>
</tr>
<tr>
<td>HTIMIT</td>
<td>Handset TIMIT</td>
</tr>
<tr>
<td>iid</td>
<td>independent and identically distributed</td>
</tr>
<tr>
<td>LAR</td>
<td>Log-Area Ratio</td>
</tr>
<tr>
<td>LDC</td>
<td>Linguistic Data Consortium</td>
</tr>
<tr>
<td>LL-HDB</td>
<td>Lincoln Laboratories–Handset Database</td>
</tr>
<tr>
<td>LLR</td>
<td>Log-Likelihood Ratio</td>
</tr>
<tr>
<td>LP</td>
<td>Linear Predictor</td>
</tr>
<tr>
<td>LPCC</td>
<td>Linear Predictive Cepstral Coefficients</td>
</tr>
<tr>
<td>LRT</td>
<td>Likelihood Ratio Test</td>
</tr>
<tr>
<td>LVCSR</td>
<td>Large Vocabulary Continuous Speech Recognition</td>
</tr>
<tr>
<td>MAP</td>
<td>Maximum A Posteriori</td>
</tr>
<tr>
<td>MFCC</td>
<td>Mel-Frequency Cepstral Coefficients</td>
</tr>
<tr>
<td>ML</td>
<td>Maximum Likelihood</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean Squared Error</td>
</tr>
<tr>
<td>MSP</td>
<td>Modulation Spectrum Processing</td>
</tr>
<tr>
<td>NERF</td>
<td>Non-uniform Extraction Region Features</td>
</tr>
<tr>
<td>NIST</td>
<td>National Institute for Standards and Technology</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>pdf</td>
<td>probability density function</td>
</tr>
<tr>
<td>PLP</td>
<td>Perceptual Linear Predictive coefficients</td>
</tr>
<tr>
<td>PPCA</td>
<td>Probabilistic Principal Component Analysis</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PPRLM</td>
<td>Parallel Phonetic Recognition with Language model</td>
</tr>
<tr>
<td>PSTN</td>
<td>Public Switched Telephone Network</td>
</tr>
<tr>
<td>QUT</td>
<td>Queensland University of Technology</td>
</tr>
<tr>
<td>RASTA</td>
<td>RelAtive SpecTrA</td>
</tr>
<tr>
<td>ROC</td>
<td>Receiver Operating Characteristic</td>
</tr>
<tr>
<td>SMS</td>
<td>Speaker Model Synthesis</td>
</tr>
<tr>
<td>SNST</td>
<td>Same Number, Same Type</td>
</tr>
<tr>
<td>SPHERE</td>
<td>NIST SPeech HEader REsources audio file format</td>
</tr>
<tr>
<td>SRE</td>
<td>NIST Speaker Recognition Evaluation</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>UBM</td>
<td>Universal Background Model</td>
</tr>
<tr>
<td>VQ</td>
<td>Vector Quantisation</td>
</tr>
</tbody>
</table>
Authorship

The work contained in this thesis has not been previously submitted for a degree or diploma at any other higher education institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

Signed: ______________________

Date: ______________________
Acknowledgements

As always, there are several people who deserve to share the credit for helping me get this thesis to a completed state. My family is first and foremost on this list, particularly my parents, as they have had to put up with me during the writing process and have done so with ceaseless support and encouragement. Their efforts to keep me sane during this period are much appreciated. I particularly have to mention my Mum’s attempts to be interested and offer assistance when I was frustrated with the technical details of my research. I must thank my friends as well for assisting with my sanity and reminding me that there is life going on outside of the lab.

I also have to thank everyone with whom I shared time in the SAIVT labs. Particularly, I have to recognise the contributions of Jason Pelecanos for showing me the ropes of speaker recognition and research in general early on in my course and of Brendan Baker for his persistent threats to quit — especially during the sleep-deprived NIST evaluation periods — and for helping proof read this thesis. I must also specifically mention Chris McCool, Kit Thambiratnam, Terry Martin, Jamie Cook and Eddie Wong for their contributions to my research. There are too many other names to list here (for fear of leaving someone out) but I certainly appreciate the contributions of all the other members of the lab during my time here for their friendship and providing a constantly colourful work environment. A special mention is also warranted for those nameless saints that donated packs of cards for the lunch time 500 ritual.

Finally, I must acknowledge my supervisory team, Professors Sridha Sridharan, Miles Moody and Doctor Michael Mason for their technical direction, feedback and encouragement. I particularly thank my principal supervisor
Sridha for the opportunities that have been afforded me during my candidature.

Robbie Vogt
October 2006
Chapter 1

Introduction

1.1 The Voice as a Biometric

Security is increasingly an important topic in current affairs whether in reference to the security of information, including the wider privacy implications, the security of national borders, particularly from the threat of terrorists, or security within those borders.

With this focus on security has come a wider interest in the area of biometrics, particularly for the purpose of person authentication. In this context, biometrics refers to the automated recognition of a person based on some physiological or behavioural characteristics. A number of characteristics are being pursued for use as a biometric including fingerprint, face, iris, hand geometry, gait and voice.

The choice of biometric is very much dependent on the intended application. The proximity to the target, the acceptable level of intrusion and the amount of expected cooperation from the target, are all important factors in the decision as well as the required levels of accuracy.

Using the voice as a biometric — which is more commonly known in the literature as speaker recognition — has some desirable characteristics. Speaker recognition is a non-invasive and convenient technology that is relatively accurate as a biometric. It has the potential to be applied to a number of person authentication applications, particularly when there is a physical separation between the claimant and the biometric system. Suitable applications in the security and
surveillance domain include suspect identification by voice, combating terrorism by using the voice to locate and track terrorists, detection of a speaker’s presence at a remote location and annotation and indexing of speakers in audio data. Wired and wireless telecommunications are media of particular relevance for these applications.

In the private sector, most applications can be categorised as over-the-phone person authentication tasks including phone banking, credit card transaction verification and other forms of customer authentication. The convenience of interacting with businesses via the telephone has lead to a massive increase in the range of services offered in this medium as well as the increased use of speech technology for providing these services. The consequential demand to verify the identity of the person on the other end of the phone line has presented a security challenge ripe for speaker recognition technology.

These applications support the importance of developing and introducing the technology of automatic speaker recognition.

1.1.1 Speaker Recognition

Speaker recognition is defined as the process of recognising the identity of a person by analysing their speech signal.

Speaker recognition can be characterised in a number of ways. Firstly, a distinction is made between speaker identification and speaker verification. The identification task is to determine which one of a number of speakers produced a given utterance. In this context, an utterance is simply defined as the speech signal that is available for testing. Identification is a closed-set problem with a limited number of possible classes. The verification problem in contrast decides whether a given utterance was generated by a particular speaker. Verification is an open-set classification problem as, for any utterance, there are an arbitrary number of possible alternative but unknown speakers. These unknown classes make open-set classification an inherently more difficult problem.

A further division is made between text-dependent and text-independent recognition. Through prior knowledge of the phrase to be spoken, text-dependent
1.1. The Voice as a Biometric

recognition extracts additional information in the form of linguistic content from
the spoken utterance. For example a text-dependent system might require a user
to recite a particular phrase to perform verification. In contrast, text-independent
recognisers have no prior knowledge or requirement of this sort, typically using
conversational or spontaneous speech for recognition.

Although text-dependent recognition to date delivers better performance [99],
text-independent recognition is a more attractive technology in terms of its broad
scope of possible applications. Due to the unobtrusive nature of text-independent
recognition, it can be incorporated into many applications without imposing any
additional requirements on a user. For instance, the authentication of telephone
transactions can be performed as a background operation while the details of the
transaction are determined. In addition, many developments in text-independent
systems can be incorporated into text-dependent systems.

Extracting and modelling the information required to characterise a speaker is
a complex problem for a number of reasons. Firstly, the raw speech signal must
be reduced to a set of features that provide the required speaker information,
however, the speech signal itself is the product of several factors such as the
linguistic and semantic content of the signal, the emotional state and health of the
speaker as well as their physical characteristics. Speech is therefore a biometric
consisting of both physiological and behavioural aspects. It is possible to use all
of these factors to aid in the recognition task but most serve to obfuscate the
issue if they are not modelled appropriately.

Another complexity is introduced by the conditions under which a speech
sample is obtained. Unlike some other biometrics such as fingerprint analysis,
speaker recognition systems typically have very limited control on the equipment
used to gather samples due to the physical separation of the claimant and the
system for most applications. Due to their ubiquity and familiarity for users,
telephones and telephone networks are a natural choice of sampling device for
many applications.

The use of telephone equipment incurs a substantial amount of environmen-
tal noise and an almost infinite variety of telephone handsets and transmission
channel conditions. The complexity introduced by environmental noise and the variety in telephone equipment has been shown to impact dramatically on the error rates of speaker recognition systems, particularly in the cases of mismatches between the training and testing conditions.

1.2 Aims and Scope

The overall aim of this work is to improve the usefulness of speaker recognition technology for practical applications in the presence of adverse conditions. The most obvious means by which to fulfil this aim is to reduce error rates, but there are other factors to be considered for the practical application of this technology. These factors include the resources required for building a verification system as well as the inconvenience imposed on the end user.

The scope of the investigation of these issues will be restricted to the underlying technology of speaker recognition. That is, the pattern recognition algorithms and models will be the focus of this work as they apply to the task of identifying speakers. The practicalities of deploying a speaker verification system are beyond the scope of this research, as is the development of a speech-based user interface and the integration with existing systems that would be necessary for many user authentication applications.

The scope is also restricted to text-independent speaker verification in telephony environments. The focus on speaker verification should not be regarded as a restriction though; a technique found effective for speaker verification should at the least not be detrimental to speaker identification purposes and in most cases both will benefit. The focus on telephony data has implications for the particular types of adverse conditions encountered. As noted above, the choice of telephony environments is a natural combination with speaker verification given the number of potential applications and the ubiquity of telephones.

Two themes are pursued to achieve the aim of this research programme; modelling mismatch and modelling uncertainty.

Modelling mismatch: Mismatch is an inevitable consequence for the general
use of speaker verification system — it must be assumed that conditions will vary between training and testing sessions. Although feature extraction such as cepstral coefficients strive to capture only the relevant information and post-processing techniques such as CMS and feature warping attempt to suppress the effects of mismatch, they can not hope to remove mismatch completely.

Given this situation the solution adopted in this theme is to incorporate mismatch into our model.

This theme resulted in a focus on feature mapping methods and the explicit modelling of inter-session variability. Of these, feature mapping attempts to model specific forms of mismatch, such as handset differences, by mapping features back to a handset-neutral space. Session variability modelling generalises this to modelling arbitrary variation between different sessions of a speaker via a low dimensional subspace.

**Modelling uncertainty:** If we had an infinite amount of speech for enrolling a speaker then we could be very certain that our estimates of model parameters from this speech would be very close to the true parameters. In reality, this is not the case. Training data is limited and often we desire to lower this limit even further. The result is potentially poor model parameter estimates with a high degree of uncertainty.

To reduce this uncertainty, techniques such as Bayesian adaptation are used to incorporate prior information about the model parameters in the training procedure, giving more robust estimates.

This can be taken further by recognising that the resulting model parameter estimates can be described in terms of posterior distributions, causing scoring methods to assume that model parameters are, in fact, unknown variables with an estimated posterior distribution. This is the premise of Bayes factor scoring, which models this uncertainty in the scoring procedure.

This theme also leads to the development of confidence based verification
methods, where it is realised that the verification score for a trial is in fact an estimate of the “true” verification score with an amount of uncertainty. If the degree of this uncertainty can be determined, this information can be used to determine the confidence in a verification decision.

1.3 Thesis Structure

The remaining chapters of this thesis are composed as follows:

Chapter 2 provides an overview of existing speaker recognition research as well as the methods used to evaluate the performance of speaker verification systems. While the GMM-UBM verification structure based on short-time acoustic features garners the most attention, an overview of the emerging fields of high-level features and modern machine learning approaches is also presented.

Chapter 3 develops the Bayes factor as an alternative verification criterion to the likelihood ratio with the aim of incorporating the uncertainty in the speaker model parameter estimates into the testing procedure. The Bayes approach assumes that the estimated model parameters are random variables with a prior distribution learnt through enrolment rather than assuming they are known variables.

Chapter 4 investigates the issue of mismatch between the training and testing utterances in speaker verification. Differences in microphone transducer type are specifically highlighted as one of the greatest causes of performance degradation in speaker verification. Various methods for reducing the effect of mismatch are discussed but the particular focus is given to feature mapping. A novel blind clustering method for training the feature mapping transform is developed to remove the need for training data that is accurately labelled for handset type.

Chapter 5 extends the idea of modelling mismatch inherent in feature mapping by introducing an explicit approach to this modelling based on directly
modelling the variability between sessions of the same speaker in a low-dimensional subspace. This approach has several advantages over feature mapping including reduced labelling requirements for the training data and the ability to capture a greater variety of mismatch without specifically determining the sources or reasons for the mismatch. Modified enrolment and verification procedures are developed to take advantage of the session variability modelling approach.

Chapter 6 proposes a novel approach to estimating the confidence in the score produced by a verification trial with applications to drastically reducing the typical data requirements for producing a verification decision. The confidence estimation procedure is also extended to produce robust results with very limited and highly correlated frame scores as well as in the presence of score normalisation. Applications of this approach to forensics are also considered.

Chapter 7 concludes the dissertation with a summary of the contributions of this research and suggests further directions for continuing research in robust speaker verification.

1.4 Original Contributions of this Thesis

This research programme has resulted in contributions to the field of speaker recognition in both of the research themes identified above.

1.4.1 Modelling Mismatch

Feature mapping

This work proposes two novel enhancements for the feature mapping technique that was originally developed for modelling telephone handset mismatch issues.

Firstly a system configuration is developed to combine the benefits of feature warping with feature mapping. This configuration highlights the flexibility ad-
vantages of the feature mapping approach over similar techniques such as speaker model synthesis (SMS).

To reduce the necessity for accurate handset labelling on the feature mapping development data, a novel data-driven clustering method is introduced to refine the mapping contexts. Experimental results demonstrate that equivalent performance can be achieved by refining inaccurate data labels and can furthermore enable feature mapping to be deployed when no data labels are available.

**Session variability modelling**

A more general and powerful approach to modelling mismatch is developed by including an explicit session term in the Gaussian mixture speaker modelling framework. Under this approach the GMM that best represents the observations of a particular recording is the combination of the true speaker model with an additional session-dependent offset constrained to lie in a low-dimensional session variability subspace.

A novel and efficient model training procedure is proposed in this work to perform the simultaneous optimisation of speaker model and session variables required for speaker training. Using a similar iterative approach to Gauss-Seidel method for solving linear systems, this procedure greatly reduces the memory and computational resources required by the direct solution.

Extensive experimentation demonstrates that the explicit session modelling approach significantly improves verification performance over both a baseline system and feature mapping system and is shown to be effective on multiple corpora exhibiting different session mismatch conditions.

It is also discovered that the session modelling approach elicits excellent performance when used in conjunction with Z-Norm score normalisation, unlike feature mapping, and receives a further significant boost from T-Norm.
1.4.2 Modelling Uncertainty

Bayes factor scoring

To account for the uncertainty in the estimates of speaker model parameters, Bayes factors are applied as the verification criteria. The Bayes factor approach effectively incorporates prior information into the testing procedure and treats the test utterance as a sequence rather than as individual frames. Under the Bayesian framework, a speaker’s model parameters are considered random variables with posterior distributions determined from the available training data.

A practical and efficient solution is developed to calculate the Bayes factor for Gaussian mixture models (GMM). The solution, utilising incremental Bayesian learning theory, is designed to be a drop-in replacement for the top-$N$ expected log-likelihood ratio (ELLR) scoring typically used for GMM-based speaker verification systems.

A novel frame-weighting enhancement is proposed for the Bayes factor scoring to accommodate for the highly correlated nature of the short-time cepstral features used.

Modelling uncertainty in verification scores

This work introduces the concept of confidence measures for the frame-based verification scores, such as ELLR scores, used in speaker verification to model the uncertainty in the final verification score. The distribution of individual frame scores are analysed to produce an estimate of the confidence interval for the resultant verification score.

Several potential applications are highlighted and discussed for this information such as estimating the upper and lower bound on the probabilities of errors and estimating the confidence with which a verification score is above or below a given threshold. A particular focus is placed on using the latter information to minimise the testing data required to make a confident verification decision. This early verification decision approach empirically demonstrates a dramatic reduction in test utterance length with a minimal impact on error rates.
Several variations of the confidence interval estimation procedure are introduced to improve the robustness of the estimate especially for the short utterances and correlated feature vectors typically encountered.

1.5 Publications

Listed below are the peer-reviewed publications to result from this research programme.


Chapter 2

An Overview of Speaker Verification Technology

2.1 Introduction

Speaker recognition is the process of identifying individuals through the analysis of their speech signals. This is a non-trivial statistical pattern recognition problem that has been investigated for a number of decades.

Generally speaking, the speaker recognition problem breaks down into three processes as with other statistical pattern recognition problems: feature extraction to obtain the speaker specific data from the raw speech signal, speaker modelling to build the necessary model to uniquely identify an individual speaker from an example of his or her speech, and testing to compare an observed speech signal to a speaker model to determine whether that particular person produced the signal.

What follows is an overview of approaches pursued to perform these tasks to date, with a particular emphasis on the methods developed for Gaussian mixture speaker modelling as it is the dominant modelling process for the task of speaker verification and the topic of much of this thesis. This discussion will incorporate the many extensions and modifications to the techniques that have resulted in improved robustness of speaker recognition over the last decade or two.

Methods for evaluating speaker verification performance are first discussed in
Section 2.2 focussing on testing protocols and performance measures. Specific attention is given to the annual speaker recognition evaluations conducted by the U.S. National Institute for Standards and Technology.

Section 2.3 reviews the task of feature extraction with a review of Gaussian mixture speaker modelling in Section 2.4. An example of a standard verification system is then presented as a summary of these topics in Section 2.5; this system is used as a reference baseline throughout the remainder of the thesis.

The chapter is concluded with an overview of the other modelling and classification methods undergoing active research for speaker recognition, including approaches to utilising recent developments in machine learning in Section 2.6 and methods based on deriving a greater understanding of the information content of the speech signal in a linguistic or prosodic sense in Section 2.7.

2.2 Evaluation of Speaker Verification

Evaluating the effectiveness of a verification system is essential for both using and researching the technology.

There are two requirements for evaluating and comparing the performance of a verification system. Firstly, a testing protocol must be in place with a well defined set of verification trials. Secondly, methods for quantifying the performance are required based on the verification trials prescribed by the verification protocol. The protocols and supporting corpora as well as the performance measures used for their evaluation in this thesis are described in this section.

2.2.1 NIST Speaker Recognition Evaluation Protocols

The U.S. National Institute of Standards and Technology (NIST) Speaker Recognition Evaluations (SRE) have been held annually since 1996. The stated goals of the evaluations are “to drive the technology forward, to measure the state-of-the-art, and to find the most promising algorithmic approaches” in the field of speaker recognition [78].
For each SRE, a strict protocol is defined for the evaluation of speaker verification systems. The focus is predominantly on telephony data in text-independent scenarios usually drawing data from the Switchboard corpora. The rules of the evaluation require participating sites to provide a set of scores for verification trials matching training utterances and test utterances for unseen data with the truth information only released on the completion of the evaluation. System development and tuning are restricted to an independent data set to the evaluation corpus, typically using the previous year’s data for this purpose. These protocols have proven to be a valuable resource for the standardisation of published results as proposed techniques can often be compared directly.

The characteristics of the main evaluation conditions are summarised in Table 2.1 including the source of the evaluation data, the length of training and testing utterances, the type of channel conditions encountered as well as the number of unique speakers and verification trials specified by the protocol.

**Years 1996 to 2003**

The early evaluations (1996–1998) investigated performance differences based on test segment duration (3, 10 and 30 seconds) and the type of training data provided. While 2 minutes were provided for training, several conditions were contrasted including obtaining 2 minutes from a single conversation, 1 minute from two separate conversations using the same telephone and 1 minute from conversations recorded on different telephones. These years were also characterised by the increasing size of the evaluation, growing from 20 speakers per gender to 250.

In 1999 all training came from a single conversation, again with approximately 2 minutes of active speech, but the focus was on the level of mismatch between the training and testing conditions. Results were presented for matched (same telephone used for training and testing) to mildly mismatched (same handset type but a physically distinct telephone) and very mismatched (different handset types). Physically distinct telephones were determined via a comparison of the telephone number — a different number implied a distinct phone. From 2000 onwards all trials were restricted to be different number trials to ensure at least
Chapter 2. An Overview of Speaker Verification Technology

<table>
<thead>
<tr>
<th>Year</th>
<th>Data</th>
<th>Training</th>
<th>Testing</th>
<th>Channel</th>
<th>Speakers</th>
<th>Trials</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1996</td>
<td>Swb-I</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Land</td>
<td>310</td>
<td>96</td>
<td>All</td>
</tr>
<tr>
<td>1997</td>
<td>Swb-II p1</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Land</td>
<td>622</td>
<td>99,79</td>
<td>40,165 Land</td>
</tr>
<tr>
<td>1998</td>
<td>Swb-II p2</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Land</td>
<td>388</td>
<td>99,20</td>
<td>366 Land</td>
</tr>
<tr>
<td>1999</td>
<td>Swb-II p3</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Land</td>
<td>397</td>
<td>39,62</td>
<td>359 Land</td>
</tr>
<tr>
<td>2000</td>
<td>Swb-II p1 &amp; 2</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Land</td>
<td>117</td>
<td>49</td>
<td>88 Land</td>
</tr>
<tr>
<td>2001</td>
<td>Swb-Cell p1</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Cell</td>
<td>356</td>
<td>27</td>
<td>321 Cell</td>
</tr>
<tr>
<td>2002</td>
<td>Swb-Cell p2</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Cell</td>
<td>330</td>
<td>23</td>
<td>228 Cell</td>
</tr>
<tr>
<td>2003</td>
<td>Swb-Cell p3</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Cell</td>
<td>356</td>
<td>23</td>
<td>227 Cell</td>
</tr>
<tr>
<td>2004</td>
<td>Mixer</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Cell</td>
<td>159</td>
<td>20</td>
<td>17 Cell</td>
</tr>
<tr>
<td>2005</td>
<td>Mixer</td>
<td>2 m</td>
<td>3,10,30</td>
<td>Cell</td>
<td>159</td>
<td>20</td>
<td>17 Cell</td>
</tr>
</tbody>
</table>

Table 2.1: Summary of the corpora and conditions for the series of NIST Speaker Recognition Evaluations from 1996 to 2005.

Utterance lengths are given in seconds (s), minutes (m), and conversation sides (c).
2.2. Evaluation of Speaker Verification

a moderate amount of mismatch.

Cellular data was collected for the 2001 to 2003 evaluations. As there was very limited development data available, the first use of cellular data in 2001 was a trial evaluation, but cellular data comprised the core evaluation in 2002 and 2003. This data, drawn from the Switchboard Cellular corpora, contained few landline recordings but several challenging conditions were introduced including the effects of speech codecs and lossy transmission as well as more rapidly changing acoustic conditions due to the mobile nature of the phones.

The Extended Data Task: 2001 to 2003

The Extended Data Task (EDT) was introduced in 2001 to foster research in the area of speaker recognition using a greater understanding of the content of speech. In contrast to the previous limited-data scenarios that allowed only 2 minutes of speech for speaker model training, the EDT allowed up to an hour of speech from 16 conversation sides for training the target speaker model. Additionally, auxiliary information, such as ASR word transcriptions and pitch and energy trajectories, were provided to encourage the exploitation of idiolectal and prosodic characteristics of speakers. Significant interest in high-level features for speaker recognition has grown out of this focus (see Section 2.7).

Due to the vast data requirements of the extended training data, a jack-knife or cross-validation configuration was employed for the EDT protocols where the speakers were split into a number of distinct groups. Under this scheme, a totally independent database was not necessary for the development of background world models however it was possible to develop techniques that were more corpus-specific.

Years 2004 and 2005

The 2004 SRE saw the convergence of the core limited data task and the EDT as protocols were defined for any combination of training duration from 10 second excerpts to 16 conversation sides and test duration from 10 seconds to a whole conversation side, all using data from the Mixer corpus. A total of 18 core
combinations were defined with an additional 10 summed-channel combinations for the 2004 evaluation. The combination with 1 conversation side for both training and testing was the only compulsory condition for all participating sites with all other combinations optional.

The use of the Mixer corpus for these evaluations also introduced significantly more challenging acoustic conditions for verification systems to contend with. Previous evaluations consisted almost entirely of landline telephony data or entirely mobile data whereas Mixer incorporates both categories as well as introducing additional challenges such as a variety of languages.

**QUT Variants of the NIST Protocols**

Variants of the NIST protocols were developed at QUT to assist in the development and tuning of systems for the 2004 and 2005 NIST evaluations. These modified protocols are referred to as the QUT EDT 2003 and the QUT 2004 protocols, respectively.

The issues addressed for the 2004 evaluation are the discrepancy in training lengths from the 2003 EDT to the 2004 conditions and the difference in the proportion of target to impostor trials between the protocols. Specifically, the NIST EDT 2003 defined 4-, 8- and 16-side training conditions but did not define the 1- and 3-side conditions of the 2004 protocol. The QUT variant remaps the 4-side training in the NIST EDT 2003 to 1- and 3-side conditions.

To reflect the greater proportion of impostor trials in the NIST 2004 protocol, a number of trials were added to the QUT EDT 2003 protocol to achieve a 10:1 impostor to target trial ratio. This is important to get the most reliable results in the operating region desired for the evaluations. The total number of trials grew to around 35,130 for each training condition.

Additionally, the number of cross-validation splits in the NIST protocol were reduced from ten to three to simultaneously reduce processing requirements for world model creation and provide independent train, test and evaluation sets for system development.

---

1The QUT EDT 2003 and QUT 2004 protocols are available on request.
2.2. Evaluation of Speaker Verification

The purpose of the QUT 2004 protocol was to make greater use of the available data from the NIST 2004 protocol and facilitate system development with a similar arrangement of train, test and evaluation splits as the QUT EDT 2003 protocol. Compared to the previous evaluations, the 2004 and 2005 data presented a significant jump in the difficulty of the task due to the mismatch introduced with the Mixer collection. The QUT variant of the 2004 protocol includes a total of 139,084 trials across all splits for the 1-side training condition using only the data available in the NIST 2004 protocol — roughly a five-fold increase in trials over the official protocol.

2.2.2 Early Corpora for Speaker Verification

The YOHO corpus \[23, 18\], publicly released in 1994, was one of the first large scale collections specifically designed for speaker verification tasks. The corpus was geared towards secure access tasks with all utterances consisting of “combination lock” phrases, such as “twenty-one, nineteen, fifty-four.” The database consists of 136 utterances from each of 138 unique speakers, with a majority of 100 being male. These utterances were recorded over a period of three months in 14 separate sessions per speaker in an office environment.

Due to the mild conditions in terms of the consistency of the recording configuration and the restricted vocabulary of this database, very good verification results have been achieved. For this reason, YOHO receives limited attention today as it is considered a solved problem.

The King database \[44\] on the other hand has more in common with recent collections. Recorded over standard telephone equipment and connections and consisting of spontaneous and unrestricted speech, this database demonstrates more difficult conditions for speaker verification. This database is also of limited use today, however, as the collection is too limited in size. In total 51 speakers are represented split between two locations, 25 in one and 26 in the other. As the conditions were significantly different between the locations the use of the database has generally been restricted to identification tasks between the speakers at a single location.
2.2.3 The Switchboard Series of Corpora

The Switchboard corpora were collected by the Linguistic Data Consortium (LDC) as part of the Effective, Affordable, Reusable Speech-to-text (EARS) project, sponsored by the Defense Advanced Research Projects Agency (DARPA), predominantly for the purpose of text-independent speaker recognition research. A series of six corpora were released under the Switchboard name consisting of spontaneous 6-minute telephone conversations, the first four of these were recorded on public landline telephone networks while the remaining two focused on mobile or cellular transmissions.

The Switchboard corpora get their name from the collection method in which participants would call into a “switchboard” computer that would initiate a call to another, randomly selected participant and connect the calls. The participants were then prompted with a topic and 6 minutes of the ensuing conversation were recorded. For speaker recognition purposes, the final 5 minutes of these conversations are typically used to remove the initial introductory and off-topic part of the conversation.

The original Switchboard or Switchboard-I consists of approximately 2,400 conversations (4,800 conversation sides) from 543 unique participants recorded in the period 1990–1991. For the purposes of speech recognition research this corpus has also been fully manually transcribed to the word level. This is a landline-only corpus containing electret and carbon-button handset types.

Switchboard-II consists of three phases differing in the region of the U.S. from which the participants were drawn — mid-Atlantic, Midwest, and Southern regions respectively — recorded in the period 1996–1998. Unlike Switchboard-I, participants were mostly solicited from universities in each region resulting in a considerably younger demographic. Each phase had approximately 650 unique participants roughly balanced for gender (around 55% were female) with 3,638, 4,472 and 2,728 calls respectively.

Switchboard-II is also restricted to only landline telephones, however, participants were encouraged to make half of their calls from different phones. This was to emphasize the issue of channel variability in speaker recognition. Most
participants were involved in more than 10 conversations each.

In 1999, collection started for Switchboard Cellular with the first phase focusing on collecting examples of GSM phones. Using the same collection method as the previous corpora, 1,309 conversations were collected from 190 participants with \( \frac{3}{4} \) of all sides recorded over a GSM connection. A second Switchboard Cellular collection was conducted in 2000 which incorporated all mobile transmission types. This corpus was dominated by CDMA transmissions due to their popularity at the time. This collection consisted of 2,020 conversations and 419 participants.

The recently collected Mixer corpus \([70]\) is similar to the Switchboard corpora in collection method and intended uses. Used in NIST evaluations since 2004, Mixer has several additional mismatch issues that can potentially adversely effect verification rates. Specifically, this corpus includes a much broader variety of telephone handsets including both landline and mobile devices, using both CDMA and GSM, as well as hands-free and cordless variants. While some effort was taken to collect ground truth information about the type of handset used in each conversation, this information is incomplete and cannot be verified as it was specified by the participants at the time of the call with no supervision.

Another interesting aspect of this corpus is the focus on multi-lingual participants. Participants were encouraged to hold conversations in several languages other than English, including Spanish, if they were native speakers of that language. Thus several trials include the situation where a speaker’s model was trained with one language but they are speaking in a different language in the test segment.

### 2.2.4 Performance Measures

Several measures of performance are utilised for the speaker verification task including the equal error rate (EER), minimum detection cost function (DCF) \([71]\) and the application-independent \(C_{llr}\) \([16]\) for numerical measures and the detection error trade-off (DET) plot \([69]\) and receiver operating characteristic (ROC) curve for graphical comparison. The three measures differ in how they depict
the errors encountered in verification however all measures require a substantial number of trials to be meaningful.

For the verification problem two types of error are observed, namely false-alarm and miss detections. These categories are also referred to as false-acceptance and false-rejection errors, respectively. False-alarm errors occur when an impostor is wrongly verified as the claimant and, conversely, miss detection or false-rejection errors occur when a claimant is wrongly rejected. It is usually possible to tune the performance of a verification system to fit a particular application by adjusting the decision threshold to trade one type of error for the other. For example, it is possible to reduce the rate of false-alarm errors by increasing the threshold however this reduction causes an increase in the miss detection error rate.

Given this potential for tuning, the EER and minimum DCF measures adjust the verification threshold to satisfy a constraint. The EER measures the rate of miss detections and false alarms when the threshold is adjusted to produce an equal rate of both error types. Alternatively, the DCF incorporates a cost for each type of error and also the prior probability of a target trial, the DCF is given by

\[ C_{DET} = C_{Miss}P_{Miss|Target}P_{Target} + C_{False\,Alarm}P_{False\,Alarm|NonTarget}(1 - P_{Target}) \]

Where \( C_{Miss} \) and \( C_{False\,Alarm} \) are the error costs, \( P_{Target} \) is prior probability of a target trial and \( P_{Miss|Target} \) and \( P_{False\,Alarm|NonTarget} \) are the system-dependent miss detection and false-alarm error rates respectively. The decision threshold is then adjusted to minimise this function. By adjusting the costs associated with this function, the minimum DCF measure can be used to measure a system’s performance for a variety of specific applications. For this reason the minimum DCF measure is popular for application-dependent system comparison purposes.

For much of this thesis the values assumed for the DCF are presented in Table 2.2 as used in the NIST Speaker Recognition Evaluations.

The DET and ROC plots both graph miss detection rate as a function of false-alarm rate for varying decision thresholds, including the EER and minimum
Table 2.2: Standard detection cost function parameter values.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{\text{Miss}}$</td>
<td>10</td>
</tr>
<tr>
<td>$C_{\text{False Alarm}}$</td>
<td>1</td>
</tr>
<tr>
<td>$P_{\text{Target}}$</td>
<td>0.01</td>
</tr>
</tbody>
</table>

DCF values. In contrast to ROC curves, the DET plot uses a Gaussian deviate scaling for both axes to provide a more meaningful comparison between systems. Specifically, if the target and impostor score distributions are Gaussian the DET plot results in a straight line. For this reason the DET plot is used in favour of the ROC curve in this work, in line with accepted practice in the field. An example DET plot is given below in Figure 2.1 demonstrating two speaker verification systems with the system represented by a solid line providing far superior performance to the dashed-line system.

For the purposes of visualisation, Figure 2.2 demonstrates contours of constant
DCF value for the NIST standard DCF parameter values from Table 2.2. As can be seen from this plot, for these values the minimum DCF operating point is most likely to occur in the low false alarm region as low DCF values can be obtained with relatively high miss rates.

As the EER and DCF measures describe the performance of the system at a particular threshold or operating point they describe the performance of a speaker verification in an application-dependent fashion. There are two issues with these measures, particularly for forensic situations. Firstly, there is no indication as to the overall value of a verification system; that is, how well the system works for all operating points. Also, there is no indication as to the interpretation of the score produced by a given verification trial. This is an important point from a forensic perspective as evidence is worthless without a meaningful interpretation.

Recent work by Brümmer [15, 16] has endeavoured to address these issues
with the introduction of the information-theoretic cost function given by

\[ C_{llr} = \frac{1}{2 \log 2} \left( E_{H_0}[C_{log}(-1, \Lambda)] + E_{H_1}[C_{log}(1, \Lambda)] \right), \] (2.1)

with

\[ C_{log}(y, \Lambda) = \log(1 + e^{-y\Lambda}), \] (2.2)

where \( \Lambda \) is the score produced for a trial, \( H_1 \) and \( H_0 \) indicate a target trial and an impostor trial respectively and \( E_{H_i}[\cdot] \) is the expectation over all trials that satisfy \( H_i \).

This cost function measures the information provided by a verification system by assuming that the scores produced by the system are log likelihood ratios (LLR) — the most useful information for evaluation of evidence. This measure highlights systems that are poorly estimating LLRs by comparing the actual to the minimum possible \( C_{llr} \) cost produced by optimally mapping the output scores to real LLR values without changing the relative order of trials.

While the \( C_{llr} \) measure presents an interesting and valuable approach to measuring speaker verification performance, it will not be used for presenting results in this dissertation. The more traditional DCF and EER measures will be preferred as these are currently better accepted in the literature.

In practical situations, factors other than the rate of errors may also be relevant in the discussion of performance. The computational efficiency of the implemented verification algorithms is a performance factor that can be as relevant as error rates in the deployment and use of system. Despite its importance, computational performance will not play a big part in the analysis of methods throughout this thesis. Discussion of computational performance will be described in parts but usually in a qualitative sense and only when computing resources are likely to be restrictive. The rationale for this approach is partly the difficulties in providing accurate quantitative results and also the relatively short time span for which any such results would be relevant, given the continuing and rapid growth of computing performance.
Chapter 2. An Overview of Speaker Verification Technology

2.3 Feature Extraction

For speaker recognition, as for any classification task, feature extraction is necessary to extract the information required to determine a speaker’s identity from the raw speech signal. Desirable characteristics for the extracted features are maximising the inter-speaker variability while minimising the intra-speaker variations and to represent the relevant information in a compact form [32]. An ideal set of features would make the modelling and classification of speakers a trivial task; it would seem however that this is an unrealisable goal and a combination of sophisticated feature extraction and modelling techniques is required for acceptable performance.

Much research has centred on the problem of reliably capturing the acoustic features of speech for both speaker and speech recognition. It is commonly accepted today that cepstral-domain features based on short periods of speech provide greater robustness than both time-domain signals and frequency-domain spectra.

For the case of acoustic features used in speaker verification today, feature extraction is a three stage process consisting of frame-based speech feature processing discussed in the next section, normalisation for noise and channel effects is then applied (Section 2.3.2), and finally speech activity detection (Section 2.3.3) is applied to remove non-speech portions of the signal [95, 30, 90, 77].

2.3.1 The short-time cepstrum

Cepstral features have proven the most successful to date at capturing the useful characteristics of speech for recognising both linguistic content and speaker identity [61, 89, 27, 67, 90]. This class of features include mel-frequency cepstral coefficients [26], linear predictive cepstral coefficients [37], and perceptual linear predictive coefficients [42]. One of the main strengths of analysis in the cepstral domain is that linear time-invariant channel effects reduce to a simple additive offset of cepstral coefficients [89, 19].

All of these features are extracted from short segments of speech or frames,
2.3. Feature Extraction

typically 10-30 ms in length with a significant overlap between consecutive frames. The assumption made in using this short-time frame-based approach is that speech signals are quasi-stationary over these short periods. The choice of frame length is typically a trade-off between spectral resolution and a less stationary signal; longer frames provide higher resolution from a discrete Fourier transform, but also result in more smeared results due to the transient effects of speech production.

A windowing function is applied to the frame of speech samples before spectral analysis to provide a more consistent response across all frequencies and pitches of speech. A Hamming window is used in this work. Windowing is important due to the peaky nature of speech signals as significantly different magnitude spectra can result from small time offsets without appropriate windowing depending on the location of the peaks relative to the frame. Techniques such as pitch synchronous cepstral analysis have also been proposed to compensate for this effect by varying the speech frame rate to centre frames on the main peaks in the speech signal [123].

A compact representation of the cepstrum is then extracted from the windowed signal. The cepstrum is defined as the Fourier or cosine transform of the log-magnitude spectrum. There are two classes of cepstral features in common use in speech processing that vary in the method by which the log-magnitude spectrum is represented. Filterbank analysis describes the magnitude spectrum through the energy in the output signal of a set of bandpass filters while the linear predictor approach approximates the spectrum in an analytical form with an all-pole filter.

**Mel-scale filterbank analysis**

Although filterbank processing dates back to the early days of speech analysis where a bank of analogue bandpass filters was used for spectrogram-like analysis [89], it has remained an effective technique. Filterbank analysis refers to representing the short-time magnitude spectrum by the energy in the output signal of a set of bandpass filters spaced evenly over the frequency range of interest. As the number of filters used is typically around 20, the output energies form a
compact set of coefficients to represent the spectrum.

Mel-frequency cepstral coefficients (MFCC) are derived from a filterbank approach to speech processing but offer two enhancements to filterbank processing; the mel-frequency spacing of the bandpass filters and the decorrelating cepstral transformation.

The mel-frequency scale is a warping from perceived pitch to physical frequency derived empirically from human listeners [109]. Recently, the mel-scale has also been demonstrated from a speech production perspective [112]. The mel scale is logarithmic in the standard frequency scale and is approximated by

\[ f_{\text{mel}} = 2595 \cdot \log_{10} \left( 1 + \frac{f_{\text{Hz}}}{700} \right). \]

By spacing the filterbanks evenly according to the mel scale, the bandwidth of each filter represents a *perceptually* similar frequency range and quantity of information content. For computational efficiency reasons the filterbank is implemented in the frequency domain using a fast Fourier transform (FFT) of the speech frames.

To generate cepstral coefficients from the filterbank output, the log energies of the filters are transformed using a discrete cosine transform (DCT). The DCT has the effect of drastically reducing the correlation present in the energy output of adjacent (and usually overlapping) bandpass filters. The value of decorrelating the resultant coefficients is to allow simpler models for their analysis, such as diagonal-covariance Gaussians.

Finally, *delta* coefficients are often appended to capture some form of temporal trend information within the features. Delta coefficients approximate the instantaneous derivative of each of the cepstral coefficients by performing a least-squares linear regression fitting over a window of consecutive frames and retaining the slope coefficient. Typical window lengths are 3 to 7 frames.

**Linear predictive analysis**

Linear predictive (LP) analysis attempts to best describe the speech signal \( s_n \) at time \( n \) through a linear combination of past values of the signal plus a weighted
version of the input excitation $u_n$,

$$s_n = Gu_n - \sum_{k=1}^{p} a_k s_{n-k}$$  \hspace{1cm} (2.3)

where the set of $p$ weights, $a_k$ are the predictor coefficients [19]. The speech production model assumed in LP analysis is a glottal excitation signal filtered through the vocal tract and nasal cavity. As $u_n$ describes the excitation signal, the LP model therefore describes the response of the vocal tract with an all-pole filter defined by the set of predictor coefficients.

The predictor coefficients are estimated for a frame of speech using a minimum mean squared error (MMSE) criterion with the residual error signal assumed to be equivalent to the excitation term, $Gu_n$. While the predictor coefficients are usually the part of the model of interest in feature extraction, the residual can be useful. For example, the residual can be used to estimate the pitch of voiced speech.

The predictor coefficients form the basis of features based on linear predictor analysis but they are usually expressed in a form that is more appropriate for modelling, either by being conceptually more meaningful such as log-area ratios (LAR) or simply being numerically convenient such as reflection coefficients and arcsine reflection coefficients.

A representation based on LP analysis that has found significant use in speaker recognition [30, 95], more recently for support vector machine approaches [22], are LP cepstral coefficients (LPCC) [37]. Similarly to the cepstral features described above, LPCC features are calculated through a further Fourier or cosine transform from the log-magnitude of the spectrum, however in this case the log-magnitude of the spectrum is estimated via the frequency response of the all-pole filter defined by the predictor coefficients. LPCC features also share many of the advantages of the filterbank cepstral features in terms of representing linear channel distortion.

The perceptual linear predictive (PLP) analysis technique [42] incorporates several perceptual factors from human hearing before applying a linear predictive model. Similarly to the mel-warping for MFCC features, a Bark-scale warping is applied to the power spectrum for equalising the information content. Compen-
sation for the difference in perceived loudness for both different frequencies and power levels is also applied. As well as these enhancements, cepstral coefficients are often taken of the the resultant LP model. RASTA processing (described below) was also first designed to enhance PLP analysis [43].

Short-time phase features

The features described above are all based on the magnitude of the spectrum of short frames of the speech signal. As noted above, the motivation for analysing short periods of the speech signal is to greatly simplifying the signal processing by treating the signal as effectively stationary during each period. The motivation for utilising only the magnitude information in this processing originates from physiological studies and human perception experiments indicating that phase is less important to our understanding of speech particularly for the short frame lengths typically used for speech processing today [64].

More recently, experiments involving the reconstruction of speech signals from magnitude-only and phase-only short-time Fourier transform information have suggested that phase may have a more important role in our understanding than historically believed [84]. The results of these experiments indicate that the choice of windowing function and the proportion of analysis frame overlap play a significant role in the intelligibility of the reconstructed signal. Specifically, for phase-only signals, a rectangular windowing function is more desirable than the more common Hamming window and that the delay between successive frames should be 1/4 of the frame length or less. Under these conditions, the phase-only reconstruction was shown to contribute comparably to the magnitude-only reconstruction in terms of intelligibility.

These results indicate that incorporating phase into speech processing tasks may provide improved performance and a few features have been proposed to do just that. Examples include representations based on the the frequency derivative or group delay function (GDF) [75, 3] and features based on higher-order spectra (HOS) [25]. These representations to date have had some success in good conditions [83], however, they have struggled to match the magnitude-based fea-
2.3. Feature Extraction

tures described above for both speech [3] and speaker recognition tasks in the presence of noisy and mismatched conditions that are typically encountered in speech processing tasks.

2.3.2 Adding Robustness to Feature Extraction

It is well known that acoustic features suffer distortion from noise and channel effects [41, 90]. Many methods, such as cepstral mean subtraction [37], RASTA processing [43], feature warping [87] and feature mapping [96] have been successfully employed to reduce the effects of the distortions encountered in short-time cepstral features. These methods are briefly described below.

An alternative approach to avoiding performance degradation due to noise and distortion is to exploit higher levels of information from the speech signal for verification that are not directly dependent on acoustic representations. Recent investigations of high-level features, such as [24], have met with considerable success when combined with acoustic features. The use of high-level features is discussed in Section 2.7.

Cepstral mean subtraction

Cepstral mean subtraction (CMS) [37] is one of the more widely used methods of compensating for stationary linear channels. It is applied to a speech segment by subtracting the mean value of each cepstral feature stream from all features in that stream. This method arises from a signal processing approach, as CMS is equivalent to performing convolution of the time-signal by an estimate of the inverse of the linear channel.

While CMS is an effective method of removing channel distortion it also removes some speaker specific information, which leads to degraded performance for clean speech. Also, CMS does not account for the distortion introduced by additive noise. A common variation on CMS, sometimes referred to as cepstral mean normalisation (CMN), compensates for the compression effect of additive noise by additionally normalising the variance of the feature stream to 1.
Chapter 2. An Overview of Speaker Verification Technology

Modulation spectrum processing

Modulation spectrum processing (MSP) and in particular RelAtive SpecTrA or RASTA filtering \[43\] have proven successful methods of incorporating speech production constraints on the time trajectories of spectral and cepstral coefficients through filtering. This success has been demonstrated in numerous NIST Speaker Recognition Evaluations \[71, 76, 77\].

RASTA processing was originally designed as supplementary steps in PLP feature extraction \[43\] but has proven a useful method for other features, such as MFCCs. It applies a bandpass filter to each stream of log-filterbank or cepstral coefficients that filters out changes in the coefficients that cannot be realised due to the physical limitations of speech production. Thus, this filter is designed to suppress spectral components of the coefficients that are detrimental to speech and speaker recognition.

Data-driven methods for modulation spectrum filter design have also been successfully demonstrated in \[7, 66\]. In \[66\] a RASTA-like temporal filter was designed to optimise a phonetic variability (signal) to channel variability (noise) ratio to enhance speech recognition tasks with a similar approach also applied to speaker verification \[115, 114\].

Feature warping

In the presence of additive noise and channel distortion the distribution of log-energy based cepstral features over time undergoes a nonlinear distortion. Feature warping \[87\] was designed to compensate for this nonlinearity by remapping the distribution of a feature stream to a target “clean” distribution through cumulative distribution function matching. In the typical case of a standard normal target distribution, this can be interpreted as short-term marginal Gaussianisation of each cepstral feature stream. The short-term distribution of the original features is usually estimated over a 300 to 500 frame period — approximately 3 to 5 seconds.

The feature warping technique was originally developed as a spectral domain technique — as a response to the distortion to lower-energy parts of spectral
2.3. Feature Extraction

distributions caused by noise — however it has been most effectively applied to cepstral features such as MFCCs where its application removes the offset induced by linear channel distortions and additionally reduces the compression effect of additive noise. Data-driven approaches to optimising the feature warping approach have also been explored [121].

Even with a suboptimal normal target distribution, a significant performance improvement is attained through this mapping. This result indicates that there is more speaker specific information in the relative positions of components in a mixture model than their absolute positions.

While only CMS explicitly attempts to compensate for linear channel effects, it is interesting to note that all three techniques described above effectively compensate for these effects through removing the DC component of the cepstral features.

Feature mapping

Unlike the feature post-processing techniques described so far, feature mapping [96] explicitly models the effects of handset differences — the most significant cause of speaker verification error. It is closely related to and originally based on the speaker model synthesis (SMS) technique [111], however, as the name suggests, feature mapping works in the feature domain while SMS is applied to the speaker models directly.

For each handset type the characteristics are captured by a GMM trained on a large quantity of speech recorded on that handset type, a mapping is then defined from this context to a handset-neutral feature space using this GMM. To apply the resulting mappings to a sequence of feature vectors the handset type is first identified by scoring it against the handset models.

Both feature mapping and SMS are covered in more detail in Chapter 4.

2.3.3 Speech Activity Detection

Most speech activity detectors (SAD) used today are frame-energy based, typically with further heuristic constraints on detector decisions [77]. A common
approach is to utilise a simple two-class classifier that operates on features such as log-energy and delta coefficients of the log-energy to distinguish between speech and non-speech [45]. Typically, bi-Gaussian modelling is used as an unsupervised classifier, with logical extensions to an ergodic HMM structure [107]. With this approach the distribution of frame log-energies for a particular utterance is estimated using a mixture of two Gaussian distributions. Speech frames belonging to the higher energy Gaussian are considered speech while the lower energy frames are rejected. The potential advantages of such an approach are the ability of the detector to operate with comparatively low signal-to-noise ratio (SNR) and, with the addition of on-line refinement of the classifier, robustness to varying noise distributions.

Common heuristically derived constraints applied for speech activity detection add some temporal information into the detector. For example, a sequence of continuous frames detected as speech is commonly restricted to exceed a pre-defined minimum length [107] such as a half-second. This restriction prevents short bursts of high-energy noise, such as a door slamming, from being detected as speech. This type of heuristic can be incorporated into a HMM detector [107] implicitly through state transition probabilities or explicitly by altering the emission state topology.

2.4 Gaussian Mixture Speaker Modelling

Gaussian mixture models (GMM) [91] have to date proven to be one of the more successful structures used for modelling the statistical characteristics of a speaker.

For a $C$-component GMM, the speaker model is described in full by the mixture component weights $\omega_c$, means $\mu_c$ and covariances $\Sigma_c$, that is $\lambda = \{\omega_1, \ldots, \omega_C, \mu_1, \ldots, \mu_C, \Sigma_1, \ldots, \Sigma_C\}$.

Comparing the utterance $X = \{x_1, \ldots, x_T\}$ of length $T$ to a GMM is given by the joint density

$$p(X|\lambda) = \prod_{t=1}^{T} \sum_{c=1}^{C} \omega_c g(x_t|\mu_c, \Sigma_c).$$

(2.4)
2.4. Gaussian Mixture Speaker Modelling

The density of a sample from an $D$-dimensional multivariate Gaussian distribution is given by

$$g(x|\mu, \Sigma) = (2\pi)^{-D/2} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right). \quad (2.5)$$

Generally, the covariance matrices $\Sigma_c$ can be fully specified, however, the complexity of the model and the number of free parameters to estimate are typically reduced by adding a diagonal-covariance constraint for speaker recognition tasks.

The following sections describe the algorithms for estimating the parameters of a GMM based on the maximum likelihood and maximum a posteriori criteria. Section 2.4.3 then describes the GMM-UBM verification structure commonly used in speaker verification including a brief explanation of the score normalisation techniques used in conjunction with this structure in Section 2.4.4.

2.4.1 Maximum Likelihood Estimation

The parameters of a GMM cannot be estimated directly using a maximum likelihood criterion due to the assumptions made for mixture models. Central to the mixture model concept is the assumption that an observation was produced by only one component of the mixture, that is a single multivariate Gaussian is responsible for any given feature vector. The resulting issue in the estimation procedure is that the mixture component that produced each feature vector is unknown.

A well-known algorithm designed for this situation — where there is missing or incomplete data — is the expectation-maximisation (E-M) algorithm [28]. The idea behind the E-M algorithm is to iteratively provide an improved estimate of the model parameters by maximising the auxiliary function $Q(\lambda; \hat{\lambda})$ which is the expected log-likelihood of the complete data $Y$, given the available information $X$ and the current model $\hat{\lambda}$. The auxiliary function can be written as

$$Q(\lambda; \hat{\lambda}) = E \left[ \log p(Y|\lambda)|X, \hat{\lambda} \right]. \quad (2.6)$$

The E-M algorithm involves two steps; expectation and maximisation. In the expectation or $E$-step, the missing information is estimated by the expected value
based on the current estimate of the model parameters, $\hat{\lambda}$ and the known data $X$. The maximisation or $M$-step is then responsible for maximising the model parameters $\lambda$ using these estimates.

For the specific case of estimating a GMM the complete data $Y$ consists of the observed feature vector $X$ and the (missing) mixture component labels that produced each vector. Substituting this into (2.6) gives

$$Q(\lambda; \hat{\lambda}) = \sum_{t=1}^{T} \log \left( \sum_{c=1}^{C} P(c|\mathbf{x}_t)\omega_c g(\mathbf{x}_t|\mu_c, \Sigma_c) \right), \quad (2.7)$$

where the expected probability of the observation $x$ being produced by mixture component $c$ is estimated by

$$P(c|x) = \frac{\hat{\omega}_c g(x|\hat{\mu}_c, \hat{\Sigma}_c)}{p(x|\hat{\lambda})}. \quad (2.8)$$

Note that $P(c|x)$ depends only on known information; the observation and the current model estimate $\hat{\lambda}$.

Evaluating $P(c|x)$ forms the $E$-step of the E-M algorithm. The next step is to maximise $Q(\lambda; \hat{\lambda})$ using this information; this is the $M$-step.

Maximising $Q(\lambda; \hat{\lambda})$ directly is problematic due to the difficulty of dealing with the log of a sum. For this reason, Jensen’s inequality is invoked to produce the simpler auxiliary function

$$\tilde{Q}(\lambda; \hat{\lambda}) = \sum_{t=1}^{T} \sum_{c=1}^{C} P(c|\mathbf{x}_t) \log \omega_c g(\mathbf{x}_t|\mu_c, \Sigma_c) \quad (2.9)$$

with Jensen’s inequality ensuring that $Q(\lambda; \hat{\lambda}) \geq \tilde{Q}(\lambda; \hat{\lambda})$. It can be shown that maximising $\tilde{Q}(\lambda; \hat{\lambda})$ for the model parameters $\lambda$ then ensures that $p(X|\lambda) \geq p(X|\hat{\lambda})$, providing an improved estimate of the model [14].

Maximising (2.9) for the GMM parameters results in the estimates

$$\omega_c = \frac{n_c}{T}, \quad (2.10)$$
$$\mu_c = \frac{1}{n_c} S_{X:c}, \quad (2.11)$$
$$\Sigma_c = \frac{1}{n_c} S_{XX:c} - \mu_c \mu_c^T \quad (2.12)$$
using the statistics

\[ n_c = \sum_{t=1}^{T} P(c|x_t, \hat{\lambda}), \quad (2.13) \]

\[ S_{X,c} = \sum_{t=1}^{T} P(c|x_t, \hat{\lambda})x_t, \quad (2.14) \]

\[ S_{XX,c} = \sum_{t=1}^{T} P(c|x_t, \hat{\lambda})x_t x_t^T. \quad (2.15) \]

For the diagonal covariance case, only the elements on the diagonal of \( \Sigma_c \) are retained and the off-diagonals are set to 0.

Due to the concaveness of the GMM likelihood function being maximised, the E-M algorithm is not guaranteed to produce a globally optimal solution for the model parameters, however, it is guaranteed to converge to a local maximum after sufficient iterations.

The E-M algorithm refines the estimate but an initial estimate of the model parameters is necessary. A good initialisation method is also very desirable as it will determine the local maximum the algorithm will converge to \[73\] as well as how rapidly it will converge. One method of initialising the model is via the \( k \)-means algorithm \[89\]. The \( k \)-means algorithm is often used for clustering and vector quantisation (VQ) problems and uses an iterative approach to estimate the cluster means or code vectors as well as the samples that produced them. This information can be used to initialise the GMM parameters for the E-M algorithm. This can substantially reduce the computational expense of the E-M algorithm as far fewer iterations are then required for convergence \[86\].

Two alternative schemes for seeding the E-M algorithm are selecting random samples from the training data as initial mixture component means, and “up-mixing.” Up-mixing essentially entails doubling the number of mixture components in a GMM by splitting each component and re-estimating. This process can be repeated a number of times and is generally initialised by first estimating a single Gaussian on the entire training data. Up-mixing is a popular strategy for refining hidden Markov models (HMM) in speech recognition.

It is common practice to impose a minimum value for any element of the
covariance matrices to avoid poorly formed density functions with points of very large and potentially infinite density [100]. This can occur, for example, in limited training data situations if a mixture component is effectively modelling a single observation vector.

### 2.4.2 Maximum A Posteriori Estimation

Most commonly, speaker models are trained using maximum a posteriori (MAP) adaptation [38]. This approach, based on Bayesian estimation theory, has been shown to produce more robust models with limited training data [93] by incorporating prior knowledge about the speaker model parameters into the training procedure. Moreover, MAP estimation has enabled an order of magnitude increase in the number of mixture components compared to the ML method used in verification systems — as many as 2048 components compared to 64 or less previously — allowing significantly more detailed speaker models.

The maximum likelihood estimation described above determines an estimate for the model parameters $\lambda^{ML}$ according to the criterion

$$\lambda^{ML} = \arg \max_\lambda p(X|\lambda). \quad (2.16)$$

As can be seen from (2.16), the only information used in this estimation method is the observed samples of the distribution, $X$. This implies that the resulting model is only capable of representing events that occur in the training data. This can make verification results very much dependent on the linguistic content of the test utterance. Equally importantly, there is no protection against training poor models due to noisy or corrupted data using a ML criterion.

In contrast, the MAP criterion can incorporate prior knowledge in the form of a prior distribution for the speaker model parameters that can capture what is known about the nature of speech and what a speaker model should look like. The model parameters are constrained to satisfy the prior by training using the criterion

$$\lambda^{MAP} = \arg \max_\lambda p(\lambda|X) \quad (2.17)$$
where \( p(\lambda | X) \) is the posterior probability of the model parameters after observing the training data. Applying Bayes theorem, this is equivalent to optimising

\[
\lambda^{MAP} = \arg \max_{\lambda} p(\lambda) p(X | \lambda)
\]

which is the likelihood of the training data multiplied by the prior distribution, \( p(\lambda) \).

For MAP adaptation of Gaussian mixture models, no sufficient statistic of a fixed dimension exists for the parameter set \( \lambda \) so the optimisation problem is not straightforward. For the purpose of a simplified presentation, only the mixture component means will be adapted using prior information. Experiments presented by Reynolds, et al. [99] as well as unpublished results support the notion that the more useful parameters to adjust are the mixture component means. The MAP estimation equations for the variance and component weight parameters can be found in Gauvain et al. [38]. Given that the mixture component means will be adapted, the prior density is assumed Gaussian and is given by

\[
g(\mu_c | \Theta_c) \propto \exp \left( -\frac{\tau_c}{2} (\mu_c - m_c)^T \Sigma_c^{-1} (\mu_c - m_c) \right)
\]

(2.19)

where \( \Theta_c = \{ \tau_c, m_c \} \) are the set of hyperparameters with \( \tau_c \geq 0 \) and \( m_c \) is a \( D \)-dimensional vector. The prior distribution hyperparameters are addressed in more detail in Section 2.4.3 below. For the prior information, assuming independence between the parameters of the individual Gaussian mixture components, the joint prior density of the speaker model mean vector parameters \( \lambda \) is given by

\[
p(\lambda) = \prod_{c=1}^{C} g(\mu_c | \Theta_c).
\]

(2.20)

The MAP solution is then solved by maximising \( p(\lambda)p(X | \lambda) \). The E-M algorithm is used to maximise the joint likelihood of this function, as in the maximum likelihood case above. The auxiliary function in this case also incorporates the model prior \( p(\lambda) \) and is given by

\[
R(\lambda; \hat{\lambda}) = \log p(\lambda) + \tilde{Q}(\lambda; \hat{\lambda})
\]

\[
= \sum_{c=1}^{C} \log g(\mu_c | \Theta_c) + \sum_{t=1}^{T} \sum_{c=1}^{C} P(c | x_t) \log \omega_c g(x_t | \mu_c, \Sigma_c).
\]

(2.21)
The E-M result for the mean adaptation process is

$$\mu_c = \frac{\tau_c \mu_c + S_{Xc}}{\tau_c + n_c}$$

(2.23)

where \( n_c \) and \( S_{Xc} \) are defined in (2.13) and (2.14). It can be seen that (2.23) is equivalent to determining the maximum likelihood solution assuming an additional \( \tau_c \) samples at the \( m_c \). With the special case of \( \tau_c = 0 \), (2.23) reverts to the maximum likelihood solution; this configuration is known as a non-informative prior.

Equation (2.23) can also be written as

$$\mu_c = \alpha_c m_c + (1 - \alpha_c) \mu_c^{ML}$$

where \( \mu_c^{ML} \) is the maximum likelihood solution of (2.11) and \( \alpha_c \) is the mean adaptation coefficient \[99\] defined as

$$\alpha_c = \frac{n_c}{n_c + \tau_c}.$$ 

This formulation emphasises that the MAP estimation of the mean is a blend between the ML estimate and the prior distribution mean \( m_c \) that is controlled by the relative weightings of the prior and observed information as expressed in \( \alpha_c \). Due to this form of expressing the MAP estimation it is commonly referred to as relevance adaptation \[65\] \[58\].

As for ML estimation using the E-M algorithm, the MAP estimation procedure is iterative. The process is initialised by setting the old estimate of the model parameters \( \hat{\lambda} \) to \( \lambda_0 \). The initial model \( \lambda_0 \) is typically determined using a universal speaker model trained on a large quantity of diverse speech. This approach is one element of the GMM-UBM structure for speaker verification.

### 2.4.3 The GMM-UBM Verification System

The GMM-UBM structure first proposed by Reynolds \[93\] has rapidly become the standard approach to text-independent speaker verification by realising a significant improvement in performance. The central advance introduced with the GMM-UBM approach is the extensive use of a universal background model
2.4. Gaussian Mixture Speaker Modelling

(UBM). The UBM is a high-order GMM trained on a large quantity of speech obtained from a wide sample of the speaker population of interest.

As described above, the UBM is used to provide the initial estimate of the speaker model parameters for MAP adaptation training but it also plays an important role in providing the prior distribution hyperparameters $\Theta$ in (2.20). Thus, the prior distribution means are set to the UBM component means giving $m_c = \mu_{UBM}^c$. Additionally, $\tau_c = \tau$ for all mixture components where $\tau$ is known as the relevance factor and controls the weighting toward the UBM in MAP adaptation.

Using this prior distribution results in a fully-coupled model adaptation system. A significant advantage of fully-coupled adaptation is that the estimate of a mixture component mean will revert to the corresponding UBM component mean when there is no appropriate adaptation data to provide a better estimate. This ensures a robust, but not necessarily accurate, estimate of the speaker’s pdf. On the other hand, when ample training data is available, the MAP adapted estimate will asymptotically approach the ML estimate.

The third use of the UBM, as the name suggests, is to represent the distribution of the null hypothesis or background speaker population in the expected log-likelihood ratio (ELLR) scoring.

$$\Lambda(s) = E \left[ \log \frac{p(x_t|\lambda_S)}{p(x_t|\lambda_{UBM})} \right] = \frac{1}{T} \sum_{t=1}^{T} \left( \log p(x_t|\lambda_S) - \log p(x_t|\lambda_{UBM}) \right)$$

The base verification score used in the GMM-UBM structure is the ratio of the speaker model likelihood to the UBM likelihood for each of the test utterance frames. The expectation of the frame log-likelihood ratios is taken to allow scores from different length test utterances to be compared, as in (2.24).

The fully-coupled adaptation used in the GMM-UBM structure is particularly effective in conjunction with ELLR scoring. Scoring unadapted mixture components of the speaker model results in a log-likelihood ratio of 0 as the speaker model and UBM will produce the same likelihood. This result supports neither
hypothesis. This implies that only components for which there is relevant adaptation data will contribute to the overall verification score.

The relationship between the target speaker models and the UBM also facilitates a very efficient method for calculating the ELLR score, known as top-$N$ ELLR scoring. Specifically, this scoring exploits the property that component $c$ in a target speaker GMM will correspond to component $c$ in the UBM due to the fully-coupled adaptation. Under top-$N$ ELLR scoring only the $N$ mixture components that contribute most to the overall likelihood scores for the target speakers and the UBM are calculated and compared for any given frame. While it is still necessary to score every component in the UBM to determine the top $N$ components, the number of target speaker components scored is drastically reduced; often by a factor of 100 or more. This is particularly useful when many target speakers must be scored at once, such as with T-Norm score normalisation described below, as the cost for scoring additional models is very low.

An interesting question that has been raised about the GMM-UBM structure is whether it derives its ability to discriminate between speakers from accurately estimating the probability distributions of speakers or by highlighting the differences between models. There is some evidence to suggest that producing more accurate models does not lead to better verification performance but that keeping a tight relationship between corresponding components of the speaker models and the UBM is more important. This is a recurring topic throughout this work and is discussed in subsequent chapters.

2.4.4 Score Normalisation

In a verification system, the final task of the system is to produce an accept or reject decision based on a threshold test. Reliably selecting thresholds is a difficult problem as distributions of system scores have been shown to be a function of many variables including the quantity of data available in training, the length of the test utterance and specifics of the recorded utterance such as handset type, background noise, transmission channel and linguistic content. Several score normalisation techniques have been proposed to both enhance performance
2.4. Gaussian Mixture Speaker Modelling

and provide a more stable operating point.

Z-Norm attempts to compensate for the training conditions of a model by mapping the distribution of impostor trial scores for each model to the standard normal distribution \([6]\). This is achieved by scoring a set of impostor trials against a speaker’s model after training and recording the mean \(\mu_Z(s)\) and standard deviation \(\sigma_Z(s)\) of these scores. All subsequent trials are normalised by

\[
\Lambda_Z(s) = \Lambda(s) - \mu_Z(s) \frac{1}{\sigma_Z(s)}
\]  

(2.25)

where \(\Lambda(s)\) is the unnormalised verifier output.

Noting the effect that handset transducer type has on the performance and specifically the score distributions of telephony speaker verification systems, H-Norm or handset normalisation was developed as an extension to Z-Norm that models the impostor distribution for carbon-button and electret handsets separately \([93, 99]\). A verification score is then normalised according to (2.25) using either the carbon-button (\(\mu_{\text{carb}}(s)\) and \(\sigma_{\text{carb}}(s)\)) or electret (\(\mu_{\text{elec}}(s)\) and \(\sigma_{\text{elec}}(s)\)) statistics depending on the type of handset used to record the test utterance. This method was found to dramatically improve performance in NIST evaluations of the late 1990’s \([30]\). H-Norm was later extended to cover any number of discrete contexts, rather than simply handset type, and rebadged as C-Norm or context normalisation.

It has also been noted that the score distribution produced by a model is highly dependent on the distance a model has been adapted from the UBM using MAP estimation. D-Norm or distance normalisation was developed to take advantage of this trend by dividing the original score by the estimated Kullback-Leibler distance between the target model and the UBM \([12]\). D-Norm seems to provide similar benefits to Z-Norm with the added advantage that no additional data is required to estimate the impostor distribution.

While Z-Norm, H-Norm and D-Norm all effectively compensate for the response of the target model, the test utterance is also responsible for introducing undesirable variation into the verification score. While this has no effect in an identification scenario, as all scores are affectively relative, it can be detrimental
to verification performance and the interpretability of scores.

T-Norm or test utterance normalisation was introduced to combat this issue \[\text{[6]}\]. Essentially the same approach was adopted for T-Norm as for Z- and H-Norm, that is estimating the distribution of impostor trials and normalising the scores by the mean and variance however in this case the test utterance is scored against a population of impostor *models*, hence the characteristics of the test utterance are captured. T-Norm is very similar to modelling the background distribution with a cohort of impostor speakers (and, in fact, the UBM score actually cancels out of the final score) except that the score is also divided by the standard deviation of these scores.

It is interesting to note that the overall effect of applying T-Norm has typically been a counter-clockwise rotation of a system’s DET curve as well as an overall reduction in the error rates. Navrátil, *et al.* investigated this phenomenon to discover that the cause of this counter-clockwise rotation was a score distribution that better fits the Gaussian assumptions of the DET plot rather than the expected reduction in the ratio of target to impostor score variances \[\text{[82]}\].

Today it is quite unusual to see a speaker verification system that does not utilise one of these score normalisation schemes and most systems submitted to NIST evaluations exploit more than one. The most successful combination has been H-Norm followed by T-Norm which is often referred to as HT-Norm for obvious reasons; interestingly, reversing the order of normalisation is considerably less effective.

### 2.5 A Baseline Speaker Verification System

Through out this thesis, proposed techniques are compared to the reference or baseline verification system detailed in this section. This verification system represents the state-of-the-art in text-independent speaker verification for telephony environments circa 2001 and incorporates many of the techniques described in the previous sections. It is briefly described by Pelecanos *et al.* in \[\text{[87]}\] and is the culmination of several years of tuning and development.
2.5. A Baseline Speaker Verification System

The feature extraction procedure, depicted in Figure 2.3, incorporates feature warping into the standard extraction of 12 mel-filterbank cepstral coefficients with the bandwidth limited to that of telephone channels at 300–3200 Hz. Delta coefficients are also appended to form a 24-dimensional feature vector. With a frame advance of 10 ms this system produces approximately 100 feature vectors per second of active speech.

The baseline system utilises fully coupled GMM-UBM modelling using iterative MAP adaptation. An adaptation relevance factor of $\tau = 8$ and 512-component models are used throughout. Unless otherwise stated, convergence of the speaker model adaptation was assumed after 10 iterations of the E-M MAP procedure. Top-$N$ ELLR scoring is used as the base verification score with $N = 5$. Score normalisation is also generally applied.
2.5.1 Interaction of Feature Extraction and Modelling Techniques

This section presents a brief study on the relationship between feature post-processing and speaker modelling techniques. A typical fully-coupled GMM-UBM verification structure is used to contrast the iterative and single-iteration formulations of MAP speaker model adaptation for different feature post-processing techniques. Three post-processing techniques for cepstral features are considered; feature warping, CMS and RASTA processing. It is shown that the advantage gained through iterative MAP adaptation is somewhat dependent on the parameterisation technique used. Reasons for this dependency are discussed. This section also highlights the difficulty in assessing performance trends due to the complex interactions between the various components of a speaker recognition system.

There are a number of factors to consider in contrasting one of the standard MAP approaches to its iterative form. The standard MAP technique is simply a single iteration of (2.23) while the E-M based result is iterative. The iterative version of this result allows for the variation of mixture component means to become dependent not only on previous iterations but also on other components to further refine the MAP estimate. Alternatively, the single-iteration approach assumes that the mixture component means vary in a completely independent manner, thus only a single iteration is required to find the MAP solution. This assumption is not always beneficial and the sparsity of the features used may determine the appropriateness of either MAP technique.

Figure 2.4 presents the effect of the number of mixture components, parameterisation method and the type of MAP adaptation on speaker recognition performance according to the minimum DCF criterion for the 1999 NIST evaluation. It can be seen that the DCF error rates significantly improve when the multiple-iteration MAP is performed instead of the basic algorithm for the NIST 1999 speech corpus. In addition, the extended MAP procedure tends to reach an optimal error rate using fewer Gaussian mixture components. In this evaluation,
2.5. A Baseline Speaker Verification System

Figure 2.4: Plot of Detection Cost versus the GMM order for different parameterisations and adaptation approaches (1- and 3-iteration) using all NIST 1999 male tests.

Feature warping is an improvement on both RASTA and CMS channel compensation techniques.

Following from this discussion, Figure 2.5 shows the same systems evaluated at the EER operating region. Interestingly, the performance of the multi-iteration MAP approach is sub-optimal to the standard algorithm for RASTA and CMS processing for 128–256 mixture components and above. In contrast to this result, the feature warping technique introduces an improvement across the range of model orders for multiple MAP iterations. This presents the issue of why, for multiple iterations, feature warping improves in performance at the EER operating point while RASTA and CMS degrade. Possible reasons for this result may be model over-training, the coupled target and background model nature of the GMM-UBM system and the sparseness of the speaker feature space attributed to the type of parameterisation.

In addressing the issue of the sparseness of the feature space, the average inter-component distance of each background model was measured using the Bhattacharyya [36] and Kullback-Leibler [103] distances (Figure 2.6). It was observed that the Bhattacharyya distance for feature warped background models was significantly smaller than either the RASTA or CMS feature processing models.
Consequently, for feature warping, if the UBM mixture component distributions are more overlapped, the use of multiple MAP iterations becomes essential to accommodate for the mixture component interactions. The Kullback-Leibler distance indicated a similar trend with feature warping producing more overlapping distributions than the other techniques.

To summarise, these experiments indicate that iterative MAP adaptation can be an effective method for improving speaker recognition performance. In particular, DCF error rates improved with feature warping, RASTA processing and cepstral mean subtraction. The equal error rate results were less conclusive with improved feature warping and degraded RASTA and CMS results at higher mixture orders. It was hypothesised that feature warping, using multiple MAP iterations, improved in a consistent manner because of the tightly clustered nature of the Gaussian modes represented in the background model as iterative MAP adaptation, within the E-M algorithm theory, accounts for the mixture component interactions while single-step adaptation assumes sparse, independent, Gaussian clusters.

This example, as with many more in this thesis, indicates that the fully-coupled GMM-UBM structure for speaker verification is not always enhanced by

![Plot of EER versus the GMM order for different parameterisations and adaptation approaches (1- and 3-iteration) using all NIST 1999 male tests.](image)
Figure 2.6: Plot of UBM average inter-component (a) Bhattacharyya and (b) Kullback-Leibler distances versus GMM order for different parameterisations.

providing *theoretically* more accurate speaker models or scoring schemes: The performance is very much dependent on the relationship and differences between the speaker models and the UBM. Due to these intricacies, empirical evidence is usually necessary to confirm a conclusion. For example, improved performance of a system on unnormalised scores may be reversed with normalisation applied (this is demonstrated in Section 5.7.3).

2.6 Modern Machine Learning Approaches

Much of the speaker verification research described in the previous sections has evolved from very traditional ideas in pattern recognition and classification such as probabilistic modelling through selecting an appropriate parametric model (mixtures of Gaussians in this case), estimating those parameters as accurately as possible with the limited data that is available for training and applying Bayesian decision logic to make classification decisions. This type of modelling is known as *generative* modelling as the idea is to determine the process or distribution that generated the observed data.
While there is nothing inherently wrong with this approach, machine learning as a field has evolved significantly and introduced many new ideas and methods for approaching pattern recognition problems. Specifically, methods such as neural networks, boosting and maximum boundary classifiers like support vector machines have proven very successful at tackling a variety of classification problems that are difficult from a classical pattern recognition perspective.

The central idea behind many of these new methods that differentiate them from the generative approach is to directly discriminate between classes by learning from examples of both classes as in the case of a binary classification problem, such as verification. For this reason these techniques are collectively referred to as discriminative methods.

The recent history of speaker recognition is, however, still dominated by generative models, particularly GMMs for the text-independent task. Apart from the current — but diminishing — advantage in performance, there have been several factors governing this dominance compared to more discriminative methods such as superior robustness to mismatch, suitability for score post-processing and a straightforward method of dealing with sequential data.

Due to the objective of estimating speakers’ speech distributions the generative models produced have no explicit decision boundary. While this can be interpreted as a disadvantage it has lead to improved robustness to mismatch compared to discriminative approaches that optimise a decision boundary based on the presented training data. In the case of mismatch, the true decision boundary will be transformed from that determined through the enrolment procedure, usually in a non-trivial manner, thus causing the trained decision boundary to suboptimal. Especially in the case of hard decisions or very abrupt decision boundaries this can be a significant source of errors.

In the generative case, the “soft” scores produced by the likelihood ratio decision criterion have demonstrated suitability for score post-processing, such as H- and T-Norm, that have provided significant performance improvements. In contrast, neural networks tend to produce much less suitable score distributions that are dominated by scores close to the extremes of -1 and 1 (with a sigmoid
activation function). This situation may change, however, as research into discriminative methods progresses.

As the generative method produces a well-understood probabilistic score for each frame, it is straightforward to deal with sequential data as a series of independent trial and combine the scores to produce the joint probability or likelihood for the entire sequence under the assumption that all feature vectors in the sequence belong to the same class. While this approach may not be optimal, as consecutive acoustic feature vectors are far from independent and a sequential model is therefore more appropriate, it has nonetheless proven to be successful. On the other hand, discriminative methods are typically designed to deal with single observations and combining the results from multiple observations is problematic as the produced scores are not usually probabilistic.

There are also disadvantages to the generative model approach, particularly that the models tend not to utilise all information available during training to produce a model that effectively discriminates between classes. In the verification case, there is generally a large amount of non-target or background information available at enrolment time which could potentially be used to produce a more discriminating model by providing negative examples to the modelling procedure. In a GMM-UBM system this may form the training data for the UBM or be a disjoint dataset. As the objective of generative techniques is to optimally estimate the distribution of speech, this information is generally ignored. This leads to a model trained only on data from one of the classes it is utilised to discriminate between. Consequently, the criteria typically used for training speaker models make no explicit attempt to discriminate between classes.

It can also be argued that the generative models spend significant effort and resources (in terms of free model parameters) in modelling parts of the probability distribution that have little or no value in terms of performing classification as these areas are very similar or identical across classes. This can lead to the situation where the poorly modelled tails of distributions contribute most to an overall verification score.

Recent work in speaker verification has focussed on approaches to combining
generative and discriminatory methods in a bid to get the best of both approaches. To date, three distinct approaches have met with success, these are utilising discriminative criteria for generative model training, combining generative and discriminatory models in hybrid systems and designing discriminative methods specifically for sequential data. These approaches are described in the following sections.

2.6.1 Discriminative Optimisation

This approach to adding discrimination to speaker modelling focuses on replacing or augmenting current generative training optimisation criteria with discriminative criteria. Current methods for training Gaussian mixture speaker models optimise for either a ML or MAP criterion. Both criteria maximise the likelihood of the speaker model producing the observed training data while the additional constraint of a prior distribution on the model parameters is considered in the MAP case.

Recent approaches such as in [122] use an additional discriminative criterion as well. In this work, a “figure of merit” optimisation step adapts speaker model parameters using a gradient descent algorithm to directly improve the system’s DET curve. This method utilises both target and impostor trials to improve the system performance. The use of this “figure of merit” criterion demonstrated significant reductions in DCF for both matched and mismatched conditions compared to ML training.

Navrátil, et al. [81] introduced the detection error trade-off analysis criterion (DETAC) as an effective criterion for training discriminatively to enhance speaker verification performance. Assuming that a verification system produces score distributions that are roughly Gaussian, the system performance will be described by a straight line on a DET plot. The DETAC aims to optimise both the offset from the origin and the slope of a system’s DET curve. The slope of this line is determined by the sigma ratio or $\sigma$-ratio, $\frac{\sigma_{\text{impostor}}}{\sigma_{\text{target}}}$, while the offset is governed by the delta-term, $\frac{\mu_{\text{impostor}} - \mu_{\text{target}}}{\sigma_{\text{target}}}$. In [81] the DETAC was applied at both the feature space (fDETAC) and system score combination (pDETAC) levels with
results that generalised significantly better than for alternative criteria, such as logistic regression.

### 2.6.2 Hybrid Systems

An alternative approach is to utilise both generative and discriminative techniques in a hybrid system. In this configuration generative models are used to estimate the pdf of a person’s speech as usual for a generative system with the addition of a discriminative classifier in the testing phase to produce an accept or reject decision. Recent research has indicated that support vector machines (SVM) \[17\] may be an appropriate discriminative structure to utilise for this purpose in speaker verification \[31, 63, 35, 119, 59\].

In several other fields, including image processing, support vector machines have proven very successful in various verification problems \[17\]. This success is due to the discriminatory nature of SVM as the machines are trained to find an optimal boundary to discriminate between classes in a high-dimensional feature space defined by a kernel. The most important task in applying SVMs in this way is the development of an appropriate kernel function. There are two distinct approaches to kernels for use in hybrid systems and both have demonstrated potential; frame-based and whole-utterance sequence kernels.

Frame-based scoring is very much akin to the generative approach, combining independent verification results from each individual frame. The difficulty with this approach is that frame scores must be converted to probabilities or likelihoods, which poses some difficulty as SVM classifiers, as with other discriminative classifiers, are typically not calibrated to produce probabilistic output. An example of this approach is given in \[13\] and \[31\].

The utterance level or sequence scoring is a more natural use of a discriminative classifier (such as SVM), the challenge with this method is to develop a mapping from a variable length sequence of speech frames to a single high-dimensional feature vector, this mapping is known as a sequence kernel. A recently developed method showing significant promise is the Fisher kernel \[46\] that utilises partial derivatives of the log-likelihood of an utterance with respect to each of the...
generative model parameters,

\[ U_\lambda(x) = \nabla_\lambda \log p(x|\lambda) \]

This method has been demonstrated to be effective for speaker verification based on GMM speaker models particularly with a normalised likelihood ratio version of the Fisher kernel [119].

Boosting techniques [74] for speaker verification based on short-time acoustic features. Boosting is a relatively recent approach in machine learning and pattern classification that can optimally combine a number of weak classifiers that perform only marginally better than chance to produce an ensemble classifier with arbitrarily good accuracy. The work of Li, et al. [62] uses pairs of components of trained Gaussian mixtures as weak learners.

2.6.3 Sequence Kernels

Campbell [20] proposed an alternative sequence kernel for speaker verification with support vector machines that does not rely on a generative model to define a feature space. The generalised linear discriminant sequence (GLDS) kernel compares two sequences of feature vectors \( X \) and \( Y \) as if \( Y \) was scored against a generalised linear discriminant model trained on \( X \) [22]. This kernel is then used to train a support vector machine. A polynomial or simple monomial linear discriminant is typically used.

The recent success of this technique in NIST evaluations, particularly in conjunction with a standard GMM-UBM system in a fused approach, has motivated the development of channel normalisation techniques specific to the GLDS SVM structure [106].

Undoubtedly discriminative techniques based on advances in machine learning will continue to receive significant attention in speaker recognition research for the foreseeable future.
2.7 Verification using High-Level Features

The majority of speaker recognition research has focussed on the use of short-time acoustic features for speaker characterisation and classification as described in the previous sections. Capturing the information of the speech signal in this fashion has proven the most successful to date but it is well known that the performance levels achieved depend heavily on the acoustic conditions of the recording. An order of magnitude difference in error rates is not unusual when comparing favourable to adverse acoustic conditions.

Motivated by the shortcoming of acoustic features, other sources of speaker specific information have recently been investigated to enhance speaker recognition performance and robustness. One of the notable investigations was a recent Center for Language and Speech Processing (CLSP) Summer Workshop on speaker recognition [24, 97] that focussed specifically on exploiting high-level features. High-level in this case refers to speaker-specific information such as linguistic content, pronunciation idiosyncrasy, idiolectal word usage, prosody and speaking style. This information is theoretically less susceptible to varying acoustic conditions.

Pilot studies have been conducted to capture high-level idiosyncratic information of speakers. These include word N-gram probabilities for capturing a speaker’s idiolect [29], refracted phonetic level N-gram statistics for capturing pronunciation idiosyncrasies [5], pitch, fundamental frequency and energy dynamics tracking [2] and conversational speech/pause timing [34] for capturing speaking style. Accurate extraction of each of these sources of speaker related information can assist in providing better informed speaker recognition decisions.

Individually these high-level features do not perform as well as short-time acoustic features and are not expected to. This comparatively poor performance is attributed to various factors including the difficulty of accurately extracting high-level features and typically high intra-speaker variation. High intra-speaker variation is particularly understandable considering the nature of the features in question; to communicate effectively we deliberately manipulate many of the
characteristics described above to convey meaning and emotion with our speech.

The purpose of these features then is to provide complementary information that is valuable in combination with each other and especially with short-time features. Specifically, high-level features are intended to add robustness to traditional features in mismatched acoustic conditions. Also, by adding independent sources of information these features potentially provide a more difficult task for mimicry in both deliberate and unintentional situations. To realise the benefits of complementary approaches system fusion is also a major focus of this area of research.

The CLSP workshop and many subsequent publications demonstrate that combining a number of high-level features could match the performance of a state-of-the-art short-time parameterisation system. In addition, results demonstrate a marked reduction in classification error after the decision statistics of both types of systems were fused.

Presented below are some of the more prominent areas of research in high-level features for speaker recognition.

### 2.7.1 Idiolect Word-Level Language Modelling

The recent revival of interest in higher levels of information for speaker recognition can be attributed to the experiments of Doddington \[29\] and the subsequent introduction of the Extended Data Task to the NIST SRE in 2001.

Doddington investigated the use of N-gram language models to capture speakers’ idiolects — the use of language unique to an individual — and subsequent use of these idiolect models to distinguish between speakers based on manual transcriptions of Switchboard conversations. A so-called “bag-of-N-grams” classifier was used in these experiments in which the speaker model consisted of the probabilities (in a relative frequencies sense) of word sequences of length N occurring in a person’s speech. These models were estimated for a speaker using transcriptions from a large number of utterances. For a given utterance the verification score was then calculated as the expected log likelihood ratio of the N-gram word
sequences that occur in the utterance,

\[ \Lambda_s = \frac{1}{N} \sum_i \log p_s(x_i) - \log p_0(x_i) \]

where \( N \) is the total number of tokens, \( x_i \) is the \( i \)th N-gram token, and \( p_s(\cdot) \) and \( p_0(\cdot) \) are the speaker and background probabilities, respectively.

Due to the large number of possible sequences of words and limited training data available for training a speaker-specific model only uni-gram and bi-gram models were assessed. Some techniques were also investigated to reduce the issue of model sparsity including discounting, where the contribution of a particular token in testing can be restricted regardless of the number of times it occurs, and a N-gram threshold count, to avoid modelling and scoring particularly infrequent tokens that will tend to have very poorly estimated probabilities with limited training.

With large numbers of training conversations these techniques were capable of performance approaching 5% EER, but provided only modest capability with more restricted training.

The techniques were subsequently investigated for the Switchboard-II corpus using transcripts produced by a large vocabulary continuous speech recognition (LVCSR) system in [120]. While the performance of the idiolect system was found to be quite modest in this situation, the potential for providing complementary classification information to a traditional GMM system was demonstrated, even with errorful transcripts.

Recently, a more disciplined approach to countering model sparsity issues was introduced by Baker, et al. [8]. Taking the cue from GMM speaker modelling approaches, a MAP estimation procedure was introduced for determining the speaker-specific language models. In this scheme, a background model trained on much larger quantities of transcribed speech was used to add robustness to speaker models. This technique provided significant performance improvements for idiolect system performance for all quantities of speaker model training data, highlighting the sparsity issues even for relatively high training quantities. Furthermore, the MAP technique demonstrated superior suitability for fusion with
a traditional GMM system providing significant gains with only a single training conversation \[9\].

### 2.7.2 Phone Information and Capturing Pronunciation Idiosyncrasy

It has been hypothesised that there is significant discriminatory information in the way a speaker realises or pronounces particular phonemes or sequences of phonemes. One difficulty in effectively eliciting this information is determining the actual phoneme the speaker is realising and a second issue is how best to model the idiosyncratic differences in the realisation.

These difficulties have been addressed by instead capturing the idiosyncrasies in the observed phone sequences recognised by an automatic phone recognition system, such as the approach taken by Andrews, et al. \[4, 5\]. The research described in \[5\] uses a parallel phonetic recognition with language model (PPRLM) structure previously used for high-performance, extensible, language identification \[124\]. This system comprised of a set of six parallel systems that each output a stream of recognised phones. Each parallel system provided phonetic transcriptions specific to one of the six system-internal languages based on the OGI database. A bag-of-N-grams classifier is then used to capture the speaker characteristics in each of these languages in a similar fashion to the word-level modelling above. It is anticipated that the speaker-specific pronunciation characteristics will be refracted through modelling this information in multiple languages \[5\].

Several extensions and variations have been proposed for this approach. As for the idiolect modelling case, introducing MAP adaptation for training the bag-of-N-grams models has been shown to increase the robustness and performance of the approach significantly while reducing the training data requirements \[8\].

Other approaches have attempted to link the speaker information across languages in a more effective manner than the simple score fusion used in Andrews’ approach. Jin, et al. \[49\] applied the bag-of-N-gram modelling technique across the languages used for recognition by aligning the phone transcriptions and con-
structing a token from the set of phones that were simultaneously recognised by the set of recognisers. Alternatively, the recognised phone sequences can be matched to the actual phoneme sequence that has been estimated from the canonical phonemic form of the word transcription of a LVCSR system [60].

Alternative modelling schemes have also been applied to this problem including support vector machines [21] and binary decision trees [80].

### 2.7.3 Prosodic Information

Prosody is another important source of speaker related information. The pitch, intonation and pause information in speech can be an indicator of speaker identity. Recently, pitch and energy contour extraction [2] and speaking rates and timing [34] for speaker recognition were shown to be beneficial for improving recognition performance. This work was combined and expanded in [1] by modelling these prosodic speaker characteristics in the context of broad phonetic categories. An earlier paper [108] examined the use of dynamic prosody statistics by tracking the fundamental frequency component of voiced speech and approximating its short-term trajectory through a linear piecewise estimate.

A framework has recently been developed to address the challenges of modelling prosodic data [51]. The non-uniform extraction region features (NERFS) framework can be used to model prosodic features — usually duration, pitch and energy statistics — over regions of speech that coincide with or are bounded by events of interest. For example, the regions may be defined as the speech between pauses and some example features may be the mean of the stylised F0, or the average phone duration in the region. For each extraction region a feature vector is extracted consisting of all the prosodic features of interest. In [51] these feature vectors were then modelled with a GMM. As some features are not meaningful in all extraction regions (there is no F0 in an unvoiced region for example) the GMM training and scoring methods were adjusted to account for this.

The NERF framework has led to the syllable NERF N-gram or SNERF-gram technique [105]. The extraction region under SNERF-gram modelling is defined by the time alignment of syllables based on automatic word transcriptions for
a segment. Similar prosodic features are also extracted under this framework however the features from several consecutive regions are concatenated to make an “N-gram” feature. Additionally, to allow for the greater dimensionality of the input features, the modelling and classifying utilises an SVM approach.

The advantages of SVM modelling were exploited using the SNERF-gram technique to gain some insight into the relative usefulness of a variety of prosodic features [104]. Under this approach pitch features contributed most to performance especially in the form of long-term trends captured by higher-order n-grams. While this scheme is quite complex, requiring a full LVCSR system plus several prosodic feature extraction algorithms and subsequent SVM processing, SNERF-grams appear to provide the most benefit from prosodic features to date and fuse well with standard acoustic approaches.

2.7.4 Constrained Speaker Recognition using Details of High-Level Features

It has been shown that some phonetic classes have higher speaker distinguishing capabilities than others [33, 50]. For example, extracted vowel steady states appear to contain more speaker specific information than transitions. This idea has been extended to constraining GMM-based speaker recognition techniques to events of interest such as specific words, phones or syllables.

One system, used in the NIST 2002 extended training speaker recognition evaluation [77], performed text-independent speaker recognition using speech constrained to a set of keywords [110]. This approach searched the input speech for special keywords and extracted features only from these regions in using a standard GMM-UBM speaker classification structure. The keywords or sounds were commonly used English words with high speaker discrimination. Although this system used short-term spectral features for use in performing speaker classification, this research used information based on high-level constraints.

Two difficulties arise with the use of a keyword-constrained system. Firstly it can be very expensive to perform a full LVCSR process over the test utterance
before speaker verification can begin. In some applications this isn’t an issue as an LVCSR system may be required for other reasons, such as for an interactive voice response (IVR) system. Secondly, there is usually no guarantee that a speaker will produce enough instances of the keywords for training or testing.

To overcome these issues, a framework for constrained recognition was recently developed based on a syllable-length unit [10]. Originally applied to a language identification task [68], the framework constrains recognition to triplets of recognised broad phone classes (for example all vowels and diphthongs are grouped as one broad class, as are all fricatives). While these triplets, or pseudo-syllables, do not necessarily represent actual syllables, they do tend to have similar typical durations.

The framework has shown promise in early investigations for speaker recognition using a GMM-UBM classifier structure for each pseudo-syllable type [10] but the intention is to use this framework in an effort to capture temporal information for speaker verification using a HMM approach and also for modelling prosodic features in a contextual way.

## 2.8 Summary

A review of the current status of text-independent speaker verification research was presented in this chapter. Much of this review was devoted to the traditional statistical pattern recognition approach using Gaussian mixture speaker modelling of features extracted from short-time analysis of the spectral content of speech signals but recent developments in the use of higher levels of information contained in the speech signal as well as modern machine learning approaches were also explored.

The issue of performance evaluation was initially addressed considering the databases, protocols and performance measures in common use for speaker verification and used throughout this thesis. The role of the NIST Speaker Recognition Evaluations was highlighted in this discussion.

Short-time cepstral analysis was presented as the dominant approach to ex-
tract speaker-specific information from a speech signal. Mel-filterbank cepstral coefficient (MFCC) features provide an efficient representation of the spectral content of speech in a manner that has the advantage of linear time invariant channel effects reducing to additive biases. Several techniques for increasing the robustness of these features to adverse conditions such as noise and channel mismatch were also discussed.

The central ideas and techniques for modelling speakers with Gaussian mixture models were discussed. Maximum *a posteriori* estimation of parameters, particularly in a fully-coupled, iterative adaptation scenario, provided a robust and efficient method for modelling speakers. MAP adaptation was presented as an element of the GMM-UBM verification structure that additionally employs a universal background model for non-target speaker modelling and uses the expected frame-based log-likelihood ratio between the target model and UBM as the verification score. Score normalisation for enhancing the robustness of verification decisions was also discussed.

Finally, the details of a complete speaker verification system were specified. This system, based on the GMM-UBM structure, forms the baseline reference system for comparison purposes in the subsequent chapters.
Chapter 3

Modelling Uncertainty in Speaker Model Estimates

3.1 Introduction

One of the major developments in the history of speaker verification was the introduction of the universal background model. The UBM generally serves two distinctly different roles in a typical speaker verification system. Firstly, as the name suggests, as a background model representing all other speakers other than the claimant during a verification trial. Secondly, and more importantly, the UBM provides the information used to define the prior distribution of speaker model parameters for MAP adaptation training.

As already noted in Section 2.4, it was this incorporation of prior information into the speaker model training procedure that realised a significant step forward in the performance and utility of speaker recognition technology. This prior information built into speaker recognition systems the knowledge of what speech is expected to “look” like and constrained the model of a speaker to adhere to this expectation, providing significantly more robust speaker models with less data than was previously possible.

With the advent of MAP adaptation, there was an implicit and subtle shift in the understanding of the nature of model parameters. Maximum likelihood training assumes that there is a correct value for each model parameter; the role of
the training method is to determine these values as best it can given the training observations. The best estimate was determined to be the one that maximised the likelihood of these observations.

Under the MAP framework the model parameters are instead considered to be random variables drawn from a distribution. This is the essence of Bayesian theory and consequently MAP adaptation is alternately known as Bayesian adaptation. The role of training in this situation is to find the estimate of the model parameters that optimally represents both the knowledge gained from the prior distribution and the observed training data. This set of parameter values represent the maximum point of the parameter distribution after observing the training data, that is the maximum of the posterior distribution.

Typically, finding this set of parameter values is the end of the story; they are retained as the “best” values for the speaker model and assumed to be fixed from then on. All knowledge of the posterior distribution that these parameter values maximise is effectively lost in the transition from enrolment to testing. This has the disadvantage of ignoring the uncertainty of the resulting parameter estimates which, in the case of limited training observations, can be considerable.

This chapter presents the Bayes factor as a replacement scoring technique for speaker verification that extends the Bayesian philosophy exploited by MAP adaptation into the testing domain. Specifically, the Bayesian approach is extended to the testing procedure by treating speaker model parameters as random variables coming from the posterior distribution estimated through training. This extension provides the ability to model the uncertainty present in the model parameters that result from training.

As this shift has implications from the nature of verification onwards, Section 3.3 presents speaker verification (and the verification problem in general) in terms of a statistical hypothesis test, proceeding to develop the decision criterion for verification under a Bayesian framework. This development results in the Bayes factor. Also considered is the role of the null hypothesis under this Bayesian framework and how the Bayes factor relates to the more familiar likelihood ratio.

In Section 3.4 Bayes factor scoring of Gaussian mixture models is derived and
3.2 Relation to Previous Work

The work presented herein was motivated by the application of Bayes factor scoring to speaker verification championed by Jiang [47] and while it adopts their central theme several significant implementation choices differentiate this work from its predecessor.

It will be shown that some approximations are necessary to realise Bayes factor scoring for GMM speaker verification. To this end, an incremental Bayes learning approach is used for calculating Bayes factors for GMMs in this work instead of a Viterbi approximation method favoured by Jiang.

Jiang also describes a method that implies significant changes to the entire speaker verification process including an extensively modified enrolment procedure. The method presented in this chapter is more suited to current state-of-the-art systems based on a GMM-UBM approach and MAP adaptation; it is effectively a drop-in replacement scoring method.

This work also introduces a novel frame-weighted adaptation variant of Bayes factor scoring to compensate for the highly correlated acoustic features commonly used in speaker verification.
3.3 Bayes Factors

To apply Bayesian methods to the speaker verification procedure it is first necessary to understand the nature of verification and describe what it is exactly that we are trying to evaluate.

Speaker verification, and verification problems generally, can be considered in the framework of statistical hypothesis testing. In the case of speaker verification, the hypothesis under scrutiny, $H_1$, is that an utterance was produced by the claimant speaker. The null hypothesis, $H_0$, is simply that the utterance was produced by another speaker. Under this scenario, Bayesian decision theory suggests that the appropriate statistic for testing the hypotheses is the posterior odds of $H_1$ given by

$$\frac{P(H_1|D)}{P(H_0|D)}$$

(3.1)

where $D$ is the available evidence and $P(H_k|D)$ is the a posteriori probability of the hypothesis $H_k$ given this evidence.

It is often difficult or impossible to directly determine the posterior probabilities required to calculate these odds so an equivalent simplification is substituted. Applying Bayes theorem to the numerator and denominator, (3.1) becomes

$$\frac{P(H_1|D)}{P(H_0|D)} = \frac{P(H_1)}{P(H_0)} \times \frac{P(D|H_1)}{P(D|H_0)}$$

(3.2)

It can be readily seen that the posterior odds are the prior odds scaled by a factor dependent on the evidence. This scaling factor is the Bayes factor \[33\], denoted $B_{10}$ or simply $B$,

$$B_{10} = \frac{P(D|H_1)}{P(D|H_0)}$$

(3.3)

The Bayes factor can be used directly as a decision criterion for verification, with an easily interpreted threshold under the assumption that the prior odds are equal. This is in fact the information required for the presentation of forensic evidence; it is not the role of the expert witness in these situations to infer prior odds on the evidence.

In the case of speaker verification the available evidence $D$ consists of the test utterance, $Y$, and training data for the claimant, represented by $X$. While this
definition could be extended to include all available speech data, the train and test utterances are the only evidence relevant to this discussion. Incorporating this data, the Bayes factor becomes

\[ B_{10} = \frac{P(Y, X|H_1)}{P(Y, X|H_0)}. \]  

That is, the ratio of the conditional probabilities of the observed training and test utterances given that both were produced by the claimant and that only the training utterance was produced by the claimant.

The particular concern of this work is the solution of (3.4) incorporating a parametric model structure to represent a class or more specifically a speaker. Gaussian mixture models are an obvious choice for the model structure that will form the basis of this work, however, the current development is more general.

With the introduction of a parametric model the Bayesian method diverges from the familiar likelihood ratio method. Under a Bayesian framework, the model parameters are considered unknown random variables which themselves have a probability density distribution. This assumption allows for the case of incomplete data and uncertainty in parameter estimates.

The consequence of assuming that model parameters are in fact random variables is that every possible value of these parameters must be considered. Thus to calculate each \( P(D|H_k) \) in (3.3), we must integrate the densities \( p(D|\lambda, H_k) \), representing the likelihood of the evidence, over the entire model parameter space,

\[ P(D|H_k) = \int p(D|\lambda, H_k)p(\lambda|H_k) \, d\lambda \]  

where \( \lambda \) is the vector of unknown parameters for the model representing the claimant and \( p(\lambda|H_k) \) is the prior probability density of the set of model parameters. This contrasts with the usual practice of effectively determining parameter estimates that maximise the conditional density.

Under this framework, (3.4) can be expressed as

\[ B_{10} = \frac{\int p(Y, X|\lambda)p(\lambda) \, d\lambda}{\int p(Y|\lambda_2)p(\lambda_2) \, d\lambda_2 \cdot \int p(X|\lambda_1)p(\lambda_1) \, d\lambda_1} \]  

where the numerator evaluates the likelihood of the evidence (\( Y \) and \( X \)) coming from a single class, while the denominator evaluates the likelihood of \( Y \) coming
from a different class to that of \( \mathbf{X} \) (these distinct classes are emphasised by the use of subscripts for the distinct sets of model parameters, \( \lambda \)).

Fortunately this equation can be simplified somewhat to remove the integration over the training data. Assuming independence of the training and test data and utilising Bayesian incremental learning \[32\], (3.6) can be expressed as

\[
B_{10} = \int \frac{p(\mathbf{Y}|\lambda_2)p(\lambda_2|\mathbf{X})}{p(\mathbf{Y}|\lambda_2)p(\lambda_2)} \cdot \int \frac{p(\mathbf{X}|\lambda_1)p(\lambda_1)}{p(\mathbf{X}|\lambda_1)p(\lambda_1)} d\lambda_2 \cdot \int \frac{p(\mathbf{Y}|\lambda)p(\lambda)}{p(\mathbf{Y}|\lambda)p(\lambda)} d\lambda_1
\]

(3.7)

In this work, the Bayes factor described in (3.7) is used as the criterion for verification. Although this Bayes factor requires integration over the entire parameter space (comprising thousands of dimensions in the high-order GMM case), a method for efficiently calculating an approximation is presented in Section 3.4.1.

### 3.3.1 Modelling the Null Hypothesis

From (3.6) it can be seen that we are in fact evaluating a ratio of likelihoods as our verification criterion although it is not the familiar likelihood ratio commonly used in speaker verification systems. Of particular note is the difference in the modelling of the null hypothesis.

The Bayes factor approach outlined above elegantly removes the issue of modelling the background population that has been a significant issue in the history of speaker verification research. Early in this history the background population, represented in the denominator of the likelihood ratio, was ignored and verification decisions were based solely on the likelihood of the claimant’s model producing the test utterance; the particular words spoken and the acoustic environment of the recording were significant sources of unwanted variability in these scores.

To reduce these dependencies, a cohort of background speakers were introduced and combined to model a background population in the denominator \[102\]. This approach raised the question of choosing an appropriate set of speakers to form this cohort: Should the cohorts be near, far or evenly distributed? How do we in fact determine whether a cohort speaker is near or far? How many cohort
speakers are required to adequately represent all speakers? How can we efficiently score against all of these models? And finally, how do we best combine the scores from the cohort to produce a representative denominator?

The introduction of the UBM and Bayesian adaptive model estimation allowed for more detailed and robust models while replacing the background cohort with a single model. The UBM in this approach plays a dual role by providing a prior distribution for the claimant model parameters and a “rest of the world” model as the denominator of the LRT.

The Bayes factor approach presented goes a step further by removing this dual role of the UBM as it is used solely for providing a prior distribution for model parameters. It is simply unnecessary under this approach to provide a model for “all other speakers;” the denominator of the ratio in (3.6) evaluates the likelihood of a different model to the claimant producing the test utterance. In this way the Bayes factor is capable of evaluating the evidence in favour of the null hypothesis, rather than introducing a model to represent a background population.

3.3.2 The Likelihood Ratio: A Special Case

Under the assumption that both hypotheses are represented by probability distributions with no free parameters, (3.4) resolves to the familiar likelihood ratio—this is known as the “simple-versus-simple” case. Additionally, under the strong condition that the probability distributions are exactly known, the Neyman-Pearson Lemma suggests that the likelihood ratio is in fact the most powerful criterion.

These conditions are equivalent to setting

\[ p(\lambda|\mathbf{X}) = \delta(\lambda - \lambda_X), \]  
\[ p(\lambda) = \delta(\lambda - \lambda_0) \]

where \( \delta(\cdot) \) is the Dirac delta function and \( \lambda_X \) and \( \lambda_0 \) represent the known target and background model parameters, respectively.
Therefore, (3.7) becomes
\[ B_{10} = \frac{\int_{\lambda} p(Y|\lambda)\delta(\lambda - \lambda_X) \, d\lambda}{\int_{\lambda} p(Y|\lambda)\delta(\lambda - \lambda_0) \, d\lambda} = \frac{p(Y|\lambda_X)}{p(Y|\lambda_0)} = \Lambda_{10}. \tag{3.10} \]

In practice, \( \lambda_X \) and \( \lambda_0 \) must be estimated.

It follows from these conditions that by using the likelihood ratio we are assuming that our model parameters are known and estimated perfectly.

\section*{3.4 Speaker Verification using Bayes Factor Scoring}

The discussion of the Bayes factor approach so far has been general to all verification problems or at least to situations where parametric models are used to represent the classes. In this form it is still a long way from practical application. The greatest hurdle to overcome for practical use is to determine the form of the integrals over the entire model space and to develop a method of efficiently calculating the value of this integral.

This section describes the incorporation of Bayes factor scoring into an existing speaker verification system \cite{87} based on the GMM-UBM structure \cite{93}. Section 3.4.1 derives the Bayes factor scoring criteria for Gaussian mixture models and Section 3.4.3 extends this derivation to compensate for a highly correlated feature set. Section 3.4.4 describes some of the practical implementation issues and the efficiency improvements used in this research.

\subsection*{3.4.1 Bayes Factor Scoring for Gaussian Mixture Models}

For speaker verification to employ Bayes factor scoring a solution for Gaussian mixture models must be determined. To evaluate Bayes factors for GMMs it is necessary to evaluate the Bayesian predictive density (3.5) that is of the form
\[ P(X|H) = \int p(X|\lambda)p(\lambda) \, d\lambda \tag{3.11} \]
where \( X \) is a sequence of observations and \( p(X|\lambda) \) is the likelihood of these observations given the model parameters \( \lambda \). In the GMM case the likelihood
3.4. Speaker Verification using Bayes Factor Scoring

function is given by

\[ p(X|\lambda) = \prod_{t=1}^{T} \sum_{c=1}^{C} \omega_c g(x_t|\mu_c, \Sigma_c) \]  

(3.12)

where \( g(\cdot) \) is the standard Gaussian density. Additionally constraining all Gaussian covariance matrices to be diagonal for the individual Gaussian components,

\[ g(x|\mu_c, \Sigma_c) = \prod_{d=1}^{D} \frac{1}{\sqrt{2\pi\sigma_{cd}^2}} \exp \left\{ -\frac{(x_d - \mu_{cd})^2}{2\sigma_{cd}^2} \right\}. \]

(3.13)

The integral also depends on the prior probability distribution of the model parameters, \( p(\lambda) \) in (3.11). The form of this prior is known to be a Normal-Wishart distribution for the Gaussian components and the Dirichlet distribution for the component weights [38]. This is complex prior over which to integrate.

Following from common practice in speaker recognition for MAP adaptation of GMMs and also from supporting experimental evidence, only the component Gaussian means are considered for adaptation in this work. Consequently the prior distribution for \( \lambda = \{\mu_1, \mu_2 \ldots \mu_C\} \) is

\[ p(\lambda) = \prod_{c=1}^{C} g(\mu_c|\Theta_c) \]

(3.14)

where \( \Theta_c = \{\tau_c, m_c\} \) are the set of hyperparameters of the prior distribution with \( \tau_c > 0 \) and \( m_c \) is a \( D \)-dimensional vector and \( g(\mu_c|\Theta_c) \) is also from the Gaussian family and given by

\[ g(\mu_c|\Theta_c) = g(\mu_c|m_c, \tau_c^{-1}\Sigma_c) = \prod_{d=1}^{D} \frac{\tau_c}{2\pi\sigma_{cd}^2} \exp \left\{ -\frac{\tau_c (\mu_{cd} - m_{cd})^2}{2\sigma_{cd}^2} \right\}. \]

(3.15)

A closed form solution to the integral in (3.11) unfortunately does not exist. Essentially this is due to the weighted sum that is central to the mixture of Gaussians and the incomplete information in the form of unknown mixture component allocation, as was the case for the E-M algorithm described in Section 2.4.

It is possible, given this difficulty to approximate the missing information in an approach analogous to expectation step of the Expectation-Maximisation algorithm for GMM estimation or the Baum-Welch algorithm used in HMM speech recognition, that is to estimate a “soft” component allocation based on the posterior probability of each component. This approach is feasible for a single observation vector \( x \) as demonstrated below however it falls down when a sequence of
observations is considered as all possible sequences of mixture components must be considered. Simplification is necessary to produce a practical result.

Jiang, et al. [47] approximate the solution of (3.11) by performing the Viterbi approximation described in [48]. Applying this approximation effectively assigns each observation sample to a single component Gaussian, potentially losing the benefits of the “soft” alignment used in the E-M algorithm for GMM estimation.

Under this approximation only one component of the mixture is considered responsible for an observation so the weighted sum of Gaussians in the GMM likelihood disappears, leaving only a single (weighted) Gaussian per observation. For the purposes of scoring the whole sequence this simplifies things greatly as the total likelihood degenerates to a product of Gaussians. This is much easier to deal with when taking the log as it avoids the log-of-a-sum issues that are otherwise present. The definite integral thus becomes feasible [47].

In contrast, this work adopts an incremental approach by updating the model prior density after each observation using incremental Bayesian learning. Hence, (3.11) simplifies to the iterative evaluation of

$$P(X|H) = \prod_{t=1}^{T} \int p(x_t|\lambda)p(\lambda|X^{(t-1)}) \, d\lambda$$

(3.16)

where $X^{(t-1)} = \{x_1, x_2 \ldots x_{t-1}\}$ is the set of observation vectors preceding $x_t$.

In this way the problem is broken down into two feasible problems; calculating the predictive density of a single observation and re-evaluating the prior density of the model parameters as observations are presented.

When only dealing with a single observation, $\int p(x_t|\lambda)p(\lambda|X^{(t-1)})d\lambda$ simplifies to a weighted sum of independent integrals over the component Gaussians,

$$\int p(x|\lambda)p(\lambda|X) \, d\lambda = \sum_{c=1}^{C} \omega_c \int p(x|\mu_c)p(\mu_c|X) \, d\mu_c$$

$$= \sum_{c=1}^{C} \omega_c \int p(x|\mu_c)p(\mu_c|X) \, d\mu_c. \quad (3.17)$$

While there is no closed form solution to the indefinite integrals in (3.17), the definite integral over the entire space is known and can be derived with the
assistance of tables of integrals, such as [40]. The result for a single mixture component is given by
\[
\int p(x|\mu_c)p(\mu_c|X) d\mu_c = \prod_{d=1}^{D} \sqrt{\frac{\tau_c}{2\pi\sigma^2_{cd}(\tau_c + 1)}} \exp \left\{ -\frac{\tau_c (x_d - m_{cd})^2}{2(\tau_c + 1)\sigma^2_{cd}} \right\}. \tag{3.18}
\]
This solution is also a Gaussian with mean \( m_c \) and a variance that has inflated over the original variance by a factor of \((\tau_c + 1)/\tau_c\).

The prior distribution \( p(\lambda|X^{(t-1)}) \) can be determined with an incremental update approach. The update equations for the prior distribution hyperparameters are equivalent to the equations for MAP adaptation of a GMM but for a single observation,
\[
\tau'_c = \tau_c + P(i|x) \tag{3.19}
\]
\[
m'_c = \frac{\tau_c m_c + P(c|x)x}{\tau_c + P(c|x)} \tag{3.20}
\]
where \( \tau'_c \) and \( m'_c \) are the updated hyperparameters after observing \( x \) and
\[
P(c|x) = \frac{\omega_c g(x|\mu_c, \Sigma_c)}{p(x|\lambda)} \tag{3.21}
\]
is the posterior probability of mixture component \( c \) producing the observation.

From the above equations, it can be seen that Bayes factor scoring can in fact be implemented as incremental MAP adaptation while scoring with adjusted variances to compensate for uncertainty in the component means. It should be noted that both hypotheses are evaluated in this fashion.

Theoretically, this incremental Bayesian learning transformation produces exactly equivalent results to the desired predictive density however some approximations violate this equivalence. Specifically, the incremental update approach to re-estimating the prior distribution after each observation causes some discrepancy due to the posterior probability estimation of the mixture component allocation in (3.21). A more accurate result would be obtained by evaluating the component allocation for all previous observations after receiving each observation. The root of this problem is again that the mixture allocations are missing data and must be estimated.

Further approximations are also made to improve the efficiency of evaluating the Bayes factor that are described in Section 3.4.4.
3.4.2 The Role of $\tau_c$

The description of the hyperparameters $\tau_c$ provided above obscurely states that they are larger than 0. What exactly is the role of these hyperparameters?

One way to interpret these values is from their role in the MAP estimation of the Gaussian means. As previously described, the prior distribution for a Gaussian mean is described by $\tau$ and the prior mean $m$. The resulting estimation equation for the Gaussian mean is a “blend” of the sample mean and the prior mean with the rate of this blend controlled by $\tau$ and the number of samples used to calculate the sample mean. The effect of the prior distribution in this equation is literally to update the sample mean as if there were $\tau$ additional samples located at the prior mean, $m$. Hence the role of $\tau$ in this instance is to specify the number of samples the prior is worth.

This is also the role of $\tau_c$ in the development of the GMM Bayes factor equations above. For (3.19) and (3.20) above this is clearly seen as these follow directly from the MAP adaptation equations ($\tau_c$ is increased by the $P(c|x)$ to signify that there was a $P(c|x)$ probability that the observation $x$ was produced by this mixture component).

As for (3.18) this role has a slightly different intuitive interpretation. As noted this predictive density has an inflated variance by a factor of $(\tau_c + 1)/\tau_c$ over the standard component density. Therefore $\tau_c$ controls the amount by which this variance is increased indicating the level of uncertainty in the mean estimate. That is, as $\tau_c$ increases the more confident we are in the mean estimate, as we have effectively more samples from which we have estimated it and the inflation of the variance therefore diminishes.

3.4.3 Test Frame Weighting

Acoustic features commonly used for speaker verification, such as MFCCs, exhibit high levels of correlation between consecutive observation frames. This is essentially by definition, considering that the short-time spectra and cepstra typically calculated for consecutive frames share two-thirds of their waveform samples and
that delta cepstra explicitly average over a number of frames.

This correlation obviously voids the commonly cited assumption of statistically independent and identically distributed (iid) feature vectors.

Although not stated explicitly, much of the preceding discussion also invokes this assumption of iid features leading to overly confident adaptation during the Bayes factor scoring process. This can be seen from (3.19), (3.20) and (3.21) which combined treat each incoming observation as completely independent. Particularly in the case of extreme mismatch, such as mismatched telephone handset types, this ultimately leads to degraded performance.

To prevent over confident adaptation during scoring a frame weighted adaptation can be employed. Adding a weighting factor $\beta$ to the update equations (3.19) and (3.20) produces

$$\tau'_c = \tau_c + \beta P(c|x)$$

$$m'_c = \frac{\tau_c m_c + \beta P(c|x)x}{\tau_c + \beta P(c|x)}$$

where typically $0 < \beta \leq 1$. Intuitively, $\beta$ represents how dependent each observation vector is from its predecessor; a value of 1 implies statistical independence and reducing values indicate increasing correlation (and, consequently, less information).

### 3.4.4 Implementation

Several issues remain with respect to the practical implementation of Bayes factor scoring within a speaker verification system.

Firstly, the discussion above does not mention the initial values for the prior distribution hyperparameters, $\Theta = \{m_c, \tau_c|c = 1, \ldots, C\}$. For all models the initial values of the hyperparameters are the same; the prior means are derived from the UBM (as is the case with MAP adaptation) and all $\tau_c$ are set to the MAP adaptation “relevance factor,” $\tau$. For the numerator, these values are then updated as a result of the speaker enrolment/training procedure; the prior means become the MAP adapted means and $\tau_c$ is the sum of the relevance factor and the probabilistic count for mixture component $c$ for the enrolment data. As
a practical note, the probabilistic counts determined from model training must therefore be retained.

Under this scheme, the speaker enrolment procedure consequently has a slightly different interpretation as it adapts the prior distribution hyperparameters to be speaker dependent rather than estimating a speaker model directly.

For the denominator, the prior distribution hyperparameters are left as their initial speaker independent values. An interpretation of this is that, at the start of a test utterance the denominator effectively represents no speaker in contrast to the usual interpretation of representing many unknown speakers with a UBM. To be verified, a claimant speaker model has to be more like the test utterance than no speaker as the speaker independent prior distribution will adapt more rapidly toward the test utterance than the speaker dependent prior.

Secondly, for efficient evaluation of the Bayes factor, a top-$N$ scoring strategy is employed that works similarly to top-$N$ ELLR scoring [93]. This also implies that only the $N$ highest contributing components of a model have their prior distributions updated by an observation; a positive side-effect of this is the reduced potential for numerical accuracy issues in the prior distribution update step. All experiments in this study use $N = 10$. It should be noted that, even with top-$N$ scoring, Bayes factor scoring is more computationally expensive than ELLR scoring due to the extra effort in incrementally adapting the prior distributions.

3.5 Experiments

The baseline recognition system used in this study is described in Section 2.5 on page 44.

3.5.1 NIST 1999 Experiments

For this evaluation, the NIST 1999 Speaker Recognition Evaluation database was used. This database is an excerpt of the Switchboard-II Phase 3 telephone speech corpus with over 500 target speakers. Approximately 2 minutes of enrolment speech is provided with typically 30-second test utterances. For further details of
3.5. Experiments

Figure 3.1: DET plot of NIST ’99 baseline results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the All, Same and Different handset type conditions.

Of particular interest with this database is the emphasis placed on the levels of mismatch represented. As well as overall performance, our results are categorised into two subsets distinguished by the level of mismatch; *Same* and *Different* handset type trials. In this corpus, the telephone handset transducer type is either electret or carbon-button. A trial is categorised as *Same* type if the training and testing segments were both recorded on the same telephone type; representing moderate mismatch. *Different* type trials are significantly more mismatched with consequently poorer system performance.

The *Different* category corresponds directly to the *DNDT* condition commonly used for the NIST ’99 corpus, however the *Same* condition combines the *SNST* and *DNST* conditions. This approach was chosen to improve the clarity of plots and the meaningfulness of the results presented as the impostor trials in the original *SNST* and *DNST* conditions were an identical set.

Figure 3.1 compares the DET curves of Bayes factor and ELLR scoring for the NIST ’99 data with the EER and minimum DCF presented in Figures 3.2 and 3.3.
Chapter 3. Modelling Uncertainty in Speaker Model Estimates

Figure 3.2: Minimum DCF values for NIST ’99 baseline results comparing ELLR to Bayes factor scoring with varying $\beta$-values for the All, Same and Different handset type conditions.

respectively. Improved performance in the low false alarm region is attained with the Bayes factor method, with reductions in the observed DCF for all conditions; up to 19% in the Same case and 6% overall. Mixed results were observed at the EER operating point with improvements in the Same condition and degradations in the All and Different cases.

The DET plots demonstrate a trend of a counter-clockwise rotation of the Bayes factor curves compared to ELLR scoring. Assuming Gaussian output score distributions, the observed reduction in DET curve slope would indicate a proportional reduction in the ratio of standard deviations of impostor to target trial score distributions, termed the $\sigma$-ratio \[82\]. This was indeed observed with the Bayes factor scoring reducing the $\sigma$-ratio by 5% overall.

It is also noted that the results indicate a reducing effectiveness of Bayes factor scoring as mismatch increases, resulting in worse performance in the Different case compared to standard ELLR. It is hypothesised that while the Bayes scoring method is more effective than ELLR scoring at discriminating between speaker classes, it is more adversely affected by mismatched features. Figures 3.2 and 3.3.
3.5. Experiments

Figure 3.3: EER for NIST ’99 baseline results comparing ELLR to Bayes factor scoring with varying $\beta$-values for the All, Same and Different handset type conditions.

do, however, indicate the positive effect of incorporating frame-weighted Bayes factor scoring particularly for the mismatched case when compared to the un-weighted version with $\beta = 1$. The configuration with $\beta = 0.125$ gives the best Bayes factor results for both DCF and EER in the Different case. Overall a $\beta$ value of 0.25 gave the most consistent results.

Figures 3.4 and 3.5 depict DET performance incorporating H-Norm and T-Norm [6]. H-Norm provides a significant boost for the Bayes factor method with an overall DCF improvement of 12% and EER improvement of 3% in favour of the proposed method. The use of HT-Norm (Figure 3.5) almost nullifies the differences between the methods, however the Bayes factor approach has a small overall advantage in both DCF and EER.

3.5.2 QUT EDT 2003 Experiments

The Bayes factor approach was further evaluated and compared using data from the NIST 2003 Speaker Recognition Evaluation EDT [78] described in Section 2.2.1. The evaluation data is a subset of the Switchboard-II Phase 2 and 3
Figure 3.4: DET plot of NIST ’99 H-Norm results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the All, Same and Different handset type conditions.

Figure 3.5: DET plot of NIST ’99 HT-Norm results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the All, Same and Different handset type conditions.
3.5. Experiments

The results for this task, presented in Figure 3.6, support the results for the NIST ’99 corpus. The Bayesian scoring method provided improved performance as measured by the minimum DCF with 6%, 9% and 2% relative improvement in the 1-, 3- and 8-side training conditions with degraded results at the EER operating point. These plots also confirm the trend of counter-clockwise DET curve rotation observed in the previous section.

It is clear however that the Bayes factor approach shows decreasing usefulness as the length of training data is increased. Undoubtedly this is related to the increased confidence in the model parameter estimates that can be expected from these extended quantities of training data.

This observation again raises questions about the evaluation of the denominator, or the null hypothesis likelihood. As the quantity of training data increases so too does the confidence in the model parameter estimates. This should then...
Table 3.1: Minimum DCF and EER of 1-side QUT EDT ’03 baseline results comparing the same handset type and different handset type performance using ELLR and Bayes factor scoring.

<table>
<thead>
<tr>
<th>Condition</th>
<th>Min. DCF</th>
<th>EER</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ELLR</td>
<td>Bayes</td>
</tr>
<tr>
<td>All</td>
<td>.0442</td>
<td>.0414</td>
</tr>
<tr>
<td>Same Type</td>
<td>.0303</td>
<td>.0249</td>
</tr>
<tr>
<td>Different Type</td>
<td>.0635</td>
<td>.0682</td>
</tr>
</tbody>
</table>

lead to the Bayes factor approach having diminishing effect. In the limit as the quantity of training data goes to infinity (for both classes) the Bayes factor will converge to the likelihood ratio, as explained in Section 3.3.2. This is not the case with the configuration described in this work. This discrepancy is caused by the denominator of the Bayes factor, in the method described here it will always start from the speaker independent prior.

Tables 3.1 and 3.2 investigate the impart of handset type on the usefulness of the Bayes factor scoring technique for the 1-side training condition. Table 3.1 indicates a similar trend to experiments on the NIST ’99 protocol with matched handset conditions providing the improved performance at the minimum DCF operating point while the mismatched case does not compare favourably with existing practice. Figure 3.7 depicts this conclusion graphically with the ELLR scoring consistently ahead for the Different curves.

Looking specifically at the combinations of training and testing handset types reveals an interesting result. Table 3.2 again indicates better results in matched conditions with the Carb—Carb and Elec—Elec results showing good DCF improvements although the EER results are mixed. Interestingly, for the mismatched conditions it seems that the train on electret, test on carbon-button combination (Elec—Carb) shows particularly poor results for the Bayes factor approach with 16% and 33% worse results at the DCF and EER operating points respectively. Notably, training speakers with data recorded on electret handsets produces both the greatest improvements in the matched case and the worst
3.6 Summary

This chapter reviewed the verification problem as a statistical hypothesis test and developed the Bayes factor as the optimal decision criterion under a Bayesian framework. The ability of the Bayesian approach to incorporate prior information into the scoring process and to allow for uncertainty in speaker model parameter estimates was highlighted. The likelihood ratio test was related to the Bayes factor as a special case under the “simple-versus-simple” conditions.

The Bayes factor was then applied in the context of a speaker verification system following the GMM-UBM structure. A novel approximation of the Bayes factor specific to GMMs was derived using an incremental learning approach to overcome the difficulties of the missing component occupancy information. This degradations in the mismatched case. This result highlights the issue of handset mismatch in speaker verification in general, as investigated in the next chapter.

Figure 3.7: DET plot of QUT EDT ’03 baseline results comparing ELLR and Bayes factor scoring ($\beta = 0.25$) for the 1-side, 3-side and 8-side training conditions.
Table 3.2: Further categorising the results of Table 3.1 on the handset type used for training and testing. Carb indicates carbon-button and Elec indicates electret.

<table>
<thead>
<tr>
<th>Handset</th>
<th>Min. DCF</th>
<th>EER</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ELLR</td>
<td>Bayes</td>
</tr>
<tr>
<td>Carb—Carb</td>
<td>.0552</td>
<td>.0468</td>
</tr>
<tr>
<td>Carb—Elec</td>
<td>.0736</td>
<td>.0723</td>
</tr>
<tr>
<td>Elec—Carb</td>
<td>.0551</td>
<td>.0638</td>
</tr>
<tr>
<td>Elec—Elec</td>
<td>.0230</td>
<td>.0183</td>
</tr>
</tbody>
</table>

derivation resulted in a drop-in replacement for standard ELLR scoring.

A novel frame-weighting factor was introduced to the derived Bayes factor to compensate for the highly correlated nature of the acoustic features used in this work.

Experiments conducted on the 1999 NIST Speaker Recognition Evaluation corpus and an extended 2003 NIST corpus demonstrated generally improved performance of Bayes factor scoring over ELLR scoring for better matched conditions particularly in the low false alarm operating region. Performance in handset mismatched conditions, however, deteriorated using Bayes factor scoring.
Chapter 4

Handset Mismatch in Speaker Verification

4.1 Introduction

The last decade of speaker recognition research particularly in the context of telephone environments has identified mismatch as a fundamental cause of verification errors. Mismatch refers to the differences between the conditions under which the training and testing utterances were recorded. This issue was also exemplified by the results of the previous chapter.

This chapter investigates the impact of mismatch caused by differences in microphone transducers used in telephone handsets on automatic speaker verification. This mismatch is simply referred to as handset mismatch. Methods for reducing the impact of handset mismatch are discussed with particular emphasis on the feature mapping technique originally develop by Reynolds [96].

Feature mapping, as the name suggests, is a normalisation approach based on a set of feature-vector-space transformations from handset-specific contexts to a handset-neutral space — in this way feature mapping directly addresses handset mismatch. Several novel extensions are proposed to extend the utility of feature mapping including an effective method for combining its use with feature warping [87] and a clustering approach for training feature mapping where accurate handset labels for the development data are not available.
Chapter 4. Handset Mismatch in Speaker Verification

The chapter starts with a brief review of handset mismatch in speaker recognition research with an investigation of the impact on performance and a discussion of the techniques to combat it. Section 4.3 describes feature mapping in detail, as proposed in [96], and an approach to incorporating feature mapping and warping in the same system is also presented. Finally, the blind, clustering variant of feature mapping is presented and compared to the original method in Section 4.4.

4.2 Mismatch and Performance

Since the move from controlled laboratory environments to public switched telephone networks (PSTN), speaker recognition research and experiments have highlighted the detrimental effect incurred due to mismatch between the training and testing conditions. The issue of mismatch has overwhelmed the degradation in performance due to background noise and poor quality transmission channels.

Published results on the King database demonstrated mismatch as a major issue with the severe drop in performance for trials that cross the “great divide” [90]. Even though the handsets used for all recordings in the King database were identical, some seemingly innocuous change in the recording apparatus caused the first half of the recorded sessions to differ significantly from the second half. In the case of features without channel compensation such as CMS, this lead to as much as a 60% absolute drop in identification rate and the drop remained in the 10–20% range with channel compensation applied.

Experiments on the Switchboard corpus highlighted handset mismatch specifically as a major cause of performance degradation [92]. For Switchboard, participants were encouraged to use a number of different telephones when making recordings which lead to examples of a wide variety of channels and handset types. Among the different handsets used, a distinct difference was discovered between those that used relatively high quality electret microphone transducers and those that used inferior carbon-button microphones. The results in [92] indicate a four-fold increase in the error rates due to handset mismatched conditions.

These studies lead to the development of databases such HTIMIT and LL-
4.2. Mismatch and Performance

Figure 4.1: Example of the performance difference between matched and mismatched conditions for the baseline system on the NIST 1999 protocol.

HDB [94] which were designed to enable the direct comparison of a variety of different handset types and the impact this has on the acoustic features used for speaker verification.

The handset mismatch was investigated further in the NIST 1999 Speaker Recognition Evaluation with results plotted based on increasing levels of mismatch. The \textit{SNST} (same number, same type) subset represents the most matched conditions with training and testing utterances for all true trials collected on the same telephone number, implying exactly the same handset was used. The \textit{DNST} (different number, same type) subset exhibits a higher degree of mismatch as the physical handsets used are different, implied by a different telephone number, but the transducer types are at least matched. The \textit{DNDT} (different number, different type) has the added effect of handset transducer type mismatch.

Several sources have demonstrated the impact of the increasing levels of mismatch on this Evaluation, particularly in the official results [71, 30]. The added
difficulty of the mismatch is also evident in the results from the previous chapter where handset type mismatch also reduced the effectiveness of the Bayes factor approach. The ELLR results from those experiments are plotted for the SNST, DNST and DNDT conditions in Figure 4.1 for both raw system scores and with HT-Norm applied. The results presented in Figure 4.1 also support the Switchboard results from [92] with the EER for the mismatched DNDT condition four times worse than the matched SNST case. The minimum DCF was also around three times worse. A notable observation is that the application of HT-Norm score normalisation has no effect on the relative performance of the matched and mismatched conditions as the error rates for the SNST and DNDT conditions have reduced by similar proportions. Evidently, these normalisation approaches are not removing the artefacts of handset mismatch although normalisation has reduced the error rates throughout.

Analysing the results of the reference system (see Section 2.5) on the 1-side training condition of the QUT EDT ’03 protocol reveals a similar discrepancy between the same handset type performance levels and that of using different handset types for training and testing as depicted in Figure 4.2. Furthermore, splitting these results based on the transducer type for training and testing reveals significantly poorer results for conditions trained with utterances collected on lower quality carbon-button handsets (the Carb—Carb and Carb—Elec subsets). As the quantity of data available for training and testing are equivalent, the difference in performance between the Carb—Elec and Elec—Carb conditions are somewhat surprising. One possible explanation for the difference could be the relative importance of training with high-quality recordings, giving the train-on-electret condition (Elec—Carb) an advantage based on the higher quality of electret transducers in general.

For reasons of clarity, this discussion has focussed on the type of handset mismatch encountered in landline telephony scenarios, however, the issues highlighted generalise to environments such as mobile telephony and combinations of mobile and landline as evidenced by recent NIST Evaluations.
### 4.2. Mismatch and Performance

<table>
<thead>
<tr>
<th>False Alarm probability (in %)</th>
<th>Miss probability (in %)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>Carb−−Carb</td>
</tr>
<tr>
<td>0.2</td>
<td>Carb−−Elec</td>
</tr>
<tr>
<td>0.5</td>
<td>Elec−−Carb</td>
</tr>
<tr>
<td>1</td>
<td>Elec−−Elec</td>
</tr>
<tr>
<td>2</td>
<td>Same Type</td>
</tr>
<tr>
<td>5</td>
<td>Different Type</td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.2: An examination of the effect of handset type combinations of the training and testing utterances for the baseline system on the QUT EDT ’03 protocol.
4.2.1 Approaches to Overcoming Mismatch

Given the well-known nature of the problem posed by mismatch, it is not surprising that many of the techniques developed for speaker recognition — and also for speech recognition — attempt to address the issue. This is indeed the motivation behind many of the enhancements described in Chapter 2.

It is interesting to note the distinct approaches used to combat the issue of mismatch and the chronological trend of the developed techniques.

Early approaches focussed on suppressing the artefacts of mismatch in the feature extraction process through filtering applied to the raw features, such as cepstral coefficients. CMS and RASTA processing are familiar examples of this approach where both can be described as filters in the time dimension; RASTA is defined as a bandpass filter and CMS is equivalent to a highpass filter that removes the DC component (this interpretation has been taken quite literally by some on-line versions of this algorithm [85]). This approach is very much a traditional signal processing approach — suppress the noise, enhance the signal — as exemplified by the data driven approach describe in [66].

Feature filtering significantly improved the ability of speaker verification systems to perform in mismatched conditions yet significant mismatch issues were still evident, leading to attempts to compensate for these issues in later stages of the verification process. Score normalisation, and particularly H-Norm, attempted to nullify the mismatch issue at the output verification score stage. Although H-Norm identified handset differences as a primary cause of mismatch, the results in the figure above indicate that the scheme was unable to effectively deal with handset mismatch even though significant reductions in error rates were achieved.

Recently the trend has been towards addressing the handset mismatch issue in earlier stages of the verification process so that the modelling of the speaker can be more accurate. This has lead to speaker model synthesis (SMS) [111], feature warping [87] and feature mapping [98] with the distinguishing feature that these techniques attempt to “fix” the damage done to the extracted features through non-linear transformations. While feature warping describes this
transformation without specific knowledge of the context, both SMS and feature mapping utilise knowledge of the handset type to cater the transformation to the particular conditions of an utterance.

The next section will describe feature mapping as described by Reynolds and its relation to SMS as well as the extensions necessary for combined use with feature warping.

4.3 Feature Mapping

Feature mapping is a context normalisation technique that learns a set of non-linear transformations for mapping a context-dependent feature space to a context-neutral feature space \[96\]. The transformation is applied to the extracted feature vectors in both the enrolment and verification phases.

The non-linear transform is defined by the relative differences between a set of GMMs representing the recording contexts of interest, such as different handset types, and a context-neutral root GMM. The root GMM is first trained using standard ML training based on example feature vectors from all available contexts. A GMM for each context is then adapted from the root model using data specific to that context via a MAP criterion.

It is essentially the adaptation relationship between the context-specific and root models that is exploited to transform the feature space. To map an observed feature vector \( \mathbf{x} \) from the context \( h \) to the context-neutral space, the mapping

\[
\mathbf{x}' = \Sigma_c^0 (\Sigma_c^h)^{-1} (\mathbf{x} - \mu_c^h) + \mu_c^0 \tag{4.1}
\]

is applied where \( \mu_c \) and \( \Sigma_c \) are the Gaussian mean and variance parameters of mixture component \( c \) from the context specific GMM (denoted with an \( h \) superscript) or the root GMM (with a 0 superscript). The component \( c \) used for this transformation is determined to be the component of the context-specific GMM with the highest likelihood of producing the observed vector \( \mathbf{x} \).

The transformation described in (4.1) maps \( \mathbf{x} \) to the same position relative to component \( c \) of the root model to that of the corresponding component in the
context-dependent model for handset $h$. That is $\mathbf{x}$ is mapped from $\mathcal{N}(\mu^h, \Sigma^h)$ to $\mathcal{N}(\mu^0, \Sigma^0)$.

This transformation is piecewise-linear; within the region where the transform is defined by a particular component $c$ the transform will be linear and can be expressed in the form

$$\mathbf{x}' = A^h_c \mathbf{x} + b^h_c,$$

where $A^h_c = \Sigma^0_c \left( \Sigma^h_c \right)^{-1}$ and $b^h_c = \mu^0_c - A^h_c \mu^h_c$. There are, however, discontinuities on the boundaries of these regions.

A variant of the transform described above that avoids discontinuities uses a soft component allocation for the feature vectors instead of the discrete allocation to the most likely component. In this scheme the transform is comprised of a weighted sum over all components,

$$\mathbf{x}' = \sum_{c=1}^C P_h(c|\mathbf{x}) \left( A^h_c \mathbf{x} + b^h_c \right)$$

where $P_h(c|\mathbf{x})$ is the posterior probability of component $c$ producing the observation $\mathbf{x}$.

In the absence of handset labels for the enrolment and verification utterances, the context-specific models are also used to estimate the recording context from which to map. This will typically be determined as the context with the highest likelihood although it may also be appropriate to estimate the prior probability for each handset type.

For a landline task, the contexts represented would typically include female and male variants of carbon-button and electret handsets while a cellular task may represent contexts such as analogue, GSM and CDMA transmission types.

### 4.3.1 Comparison to Speaker Model Synthesis

Feature mapping was inspired by, and has much in common with, speaker model synthesis \[\text{[111]}\]. Both approaches seek to find a mapping to allow verification trials to occur in matched conditions but the core difference between the two is the domain in which this mapping is applied. As the name suggests, SMS works in the
model domain and attempts to *synthesize* speaker models for all contexts other than the one represented by the training utterance. For example, if the enrolment utterance for a speaker was recorded on an electret handset, a transformation would be applied to the GMM parameters to synthesise an additional model for testing against carbon-button recordings. Similarly to feature mapping, this transformation is derived from a set context-specific models adapted from a root model.

Despite this similarity, and indeed similar verification performance \[96\], feature mapping has a couple of advantages.

Feature mapping tends to be better suited to on-line tasks where it is impractical to wait for the entire utterance to be captured. In this type of application the context of the utterance can be identified based on short windows or segments such as a second or even on a frame-by-frame basis. A frame-by-frame approach was in fact used successfully in the 2005 NIST SRE \[52\]. This short-term analysis is also beneficial for multi-speaker situations where more than one context may be represented in a single utterance due to the handsets used on either end of a telephone conversation.

The most compelling advantage, however, is the possibility of entirely separating the feature mapping process from the modelling aspects of the system and treating it as an independent feature post-processing step. This separation allows simple differences in configuration such as using higher orders of GMM for speakers than for the mapping contexts, the use of alternate modelling paradigms such as SVMs \[52\] or even in combination with hidden Markov models for speech recognition tasks. This separation will be exploited in the next section with the combination of feature mapping and feature warping.

### 4.3.2 Combining Feature Mapping and Warping

The QUT speaker verification system has used feature *warping* for several years with significant success and feature warping is an integral part of the reference system described in Section \[2.5\], however combining feature mapping with this system is a challenging task. The difficulty arises due to feature warping actually
Chapter 4. Handset Mismatch in Speaker Verification

suppressing much of the information used by feature mapping to determine the context of a given utterance. Hence, applying feature mapping after warping produces too many context classification errors and consequently erroneous mappings. The net result of this increased rate of context classification error is worse overall performance with mapping than without.

Since feature warping has been shown to outperform a number of other feature normalisation techniques such as CMS and modulation spectrum processing [85], it is desirable to develop a configuration to maintain the use of feature warping with the introduction of feature mapping.

The resulting configuration applies feature mapping prior to warping, but to remove the static (linear) channel issues CMS was also applied before feature mapping. For the reference system CMS is not used as feature warping usually makes it irrelevant as warping relies solely on the relative ordering or ranking of observations which CMS does not effect. The resulting feature extraction process is presented in Figure 4.3.

This feature extraction process also implies that the root GMM used for feature mapping must be distinct from the background models used for verification as the features for the root and context specific models do not yet have feature warping applied. This approach is impossible with SMS as the root model and background model are inseparable. As a side benefit, this configuration permits the use of gender-dependent UBMs which are well known to give a marginal performance increase over a single, gender-independent UBM as used in [96].

Figure 4.4 shows the performance of this configuration for the development split of the QUT EDT '03 protocol (Section 2.2.1). The utterances used to train the feature mapping models and the system UBMs were selected as a balanced set of 150 utterances each from four gender/handset contexts to provide 600 utterances for background training. The set of utterances used were selected from Switchboard-II, Phases 2 & 3 and were independent of the evaluation set.

Overall, the feature mapping approach delivers a substantial improvement compared to the reference system with relative improvements of 16% and 19% in the minimum DCF and EER respectively. The DET curves for the matched
Figure 4.3: The feature extraction process incorporating feature mapping and feature warping.
Figure 4.4: DET plot of the system with feature mapping and feature warping compared to a reference system for the 1-side condition of the QUT EDT ’03 protocol development split.
and mismatched handset types are also included in Figure 4.4. It can be seen that feature mapping achieves the bulk of the performance gains through the mismatched handset condition with feature mapping consistently ahead for all operating regions.

Contrary to expectations, there are also some gains when the same handset type is used for both training and testing, particularly in the low false alarm region. One possible explanation for this observation is that the background models are less polluted with handset differences in the feature mapping case as all background data has been mapped to be handset neutral, rather than containing a combination of both electret and carbon-button information. This will allow the background models to more accurately represent the background speaker population.

4.4 Blind Feature Mapping

To this point of the discussion of feature mapping, an assumption has been made on the availability of a substantial amount of development data labelled for handset type. It will often be the case that handset type labels are not available for the development data or that the given labels are inaccurate.

This is a consequence of the difficulty in obtaining and auditing ground truth information about handset types in any data collection of a reasonable size that attempts to capture a realistic representation of telephone network conditions at large.

For example, Switchboard and Switchboard-II both contain only automatically detected labels for carbon-button and electret handset types based on a detector trained using the synthetic HTIMIT and the small LLHDB corpora. The accuracy of these labels is also debatable, given the differences in the original labels (contained in the SPHERE audio file headers) and a later set of labels made available for the 2003 NIST SRE (for details see the NIST SRE website [78]). Both sets of labels were produced by MIT Lincoln Laboratories yet they disagree in roughly 20% of cases.
For the Mixer corpus this labelling was left to the participants, who were asked a series of questions relating to the equipment they were using [70]. This information has subsequently been considered ground truth but there is no way of accurately auditing this information.

Blind, data-driven feature mapping is presented in this section as an alternative method of training the context-specific models that define the feature-space transform which does not assume the availability of accurate context labels. An iterative clustering method is used to refine the context membership of the development data to overcome the limitations of inaccurate labels. Experiments demonstrate the ability of this clustering method to perform at least as effectively as the technique proposed by Reynolds [96]. A number of scenarios are considered that vary in the type of labelling that is available, ranging from context labels that are potentially suspect to having no labels available of any kind.

4.4.1 Clustered Feature Mapping Training

The idea behind the clustering is to group together acoustically similar utterances under the assumption that utterances that are acoustically similar will be from similar contexts or, more specifically, similar handset types.

The clustering algorithm chosen for this work is essentially the \( k \)-means algorithm with the exception that each cluster is represented by a GMM trained on all data belonging to that cluster instead of simply a mean vector. The similarity measure chosen for this work is therefore the log-likelihood of an utterance given the “centroid” GMM representing each cluster. Under this scheme the iterative clustering approach improves the context membership of the training data by repeatedly re-classifying the training utterances and updating the context memberships. The training of the feature mapping models in its entirety is described in Algorithm 1.

As with any iterative refinement algorithm, the loop should also be guarded by a condition on the maximum number of iterations, however it may be noted that during the following experiments the loop always converged in no more than 20 iterations and usually closer to 5.
4.4. Blind Feature Mapping

**Algorithm 1 Clustered Context Model Training**

1: Select a set of utterances \( X_1, \ldots, X_N \) to represent the universe of interest.
2: Train the root GMM, \( \lambda^0 \), from all \( X_n \) to represent the neutral context.
3: Initialise the context labels \( y_1, \ldots, y_N \) for each utterance.
4: repeat
5: \hspace{1em} for \( i = 1 \) to \( H \) do
6: \hspace{2em} Train a context GMM \( \lambda^i \) by MAP adapting from \( \lambda^0 \) using the set of context utterances \( \{ X_n \mid y_n = i \} \).
7: \hspace{1em} end for
8: Update each utterance’s context membership using 
   \( y_n \leftarrow \text{arg max}_i p(X_n | \lambda^i) \).
9: until the context membership \( y_1, \ldots, y_n \) is unchanged.

The assumption that acoustically similar utterances originate from similar contexts may seem unreasonable as several instances of the same speaker, or of similar speakers, may be expected to have more similarities than very different speakers on a the same handset type. It is argued, however, that, if the number of utterances in each cluster is kept high, broader characteristics are more likely to be captured. For this reason there is also a minimum context membership condition place on Algorithm 1. Furthermore, there is no particular guarantee that the final contexts represented correspond to handset types; as this is a data driven method, the data will determine the most salient characteristics in differentiating the clusters.

There is a sizeable body of work considering clustering issues for speech data in the field of speaker clustering and segmentation and many more elaborate clustering schemes are possible using more sophisticated distances measures. There is a strong argument to use a log-likelihood comparison based on cluster GMMs, however, for consistency with the eventual use of the clusters in feature mapping: As noted in Section 4.3, the first step of feature mapping is to identify the context of an utterance via a log-likelihood comparison to the context-dependent GMMs.

Several variations are possible with this algorithm that will often be dictated by the availability of labelled data. Specifically, the way in which the context-specific sets are initialised on line 3 and the initial selection of training utterances on line 1 are the subject of the experiments that follow.
4.4.2 Experiments

The effectiveness of the clustering approach to training the feature mapping transform were empirically investigated in comparison to both a reference system and the original feature mapping approach as described by Reynolds. It should be noted that for these experiments a combination of feature mapping and warping was used in all situations, as described in Section 4.3.2 above. The 1-side condition of the QUT EDT ’03 protocol was used for these experiments. Results are presented for the development split with the feature mapping and UBM training data selected from the remaining splits.

Comparison to standard feature mapping

The initial experiment involved the replication of Reynolds’ feature mapping method, with the addition of feature warping, and comparing this with an iterative refinement of the context models using the algorithm described above. The purpose of this configuration is to model the situation in which context labels, that is handset labels, are available but are not necessarily accurate. For this experiment the utterances used for training the UBM and feature mapping models were selected as a balanced set of approximately 600 utterances equally representing the four known contexts in the data.

In addition to replicating Reynolds’ feature mapping approach the first experiment investigated seeding the training of the contexts randomly under the assumption that no useful context labels are available. In addition to determining whether randomly seeded training can automatically generate effective contexts, the effect of varying the number of clustered contexts was investigated. The same selection of background utterances were used for the random seeding experiment as for the refinement experiment.

Table 4.1 details the minimum DCF value and EER for the each of the systems investigated. Evidently, refining the context membership using the proposed method is essentially equivalent to the labelled method, comparing the Labelled and Label Seeded results. This indicates that the clustering refinement at least maintains the abilities of the labels it attempts to refine. In this case it would
Table 4.1: Minimum DCF and EER for standard and clustered feature mapping configurations on the QUT EDT ’03 protocol.

<table>
<thead>
<tr>
<th>System</th>
<th># Contexts</th>
<th>Min. DCF</th>
<th>EER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>4</td>
<td>.0465</td>
<td>14.0%</td>
</tr>
<tr>
<td>Labelled</td>
<td>4</td>
<td>.0392</td>
<td>11.3%</td>
</tr>
<tr>
<td>Label Seeded</td>
<td>4</td>
<td>.0391</td>
<td>11.3%</td>
</tr>
<tr>
<td>Random Seeded</td>
<td>4</td>
<td>.0422</td>
<td>12.3%</td>
</tr>
<tr>
<td>Random Seeded</td>
<td>6</td>
<td>.0387</td>
<td>11.1%</td>
</tr>
<tr>
<td>Random Seeded</td>
<td>8</td>
<td>.0383</td>
<td>10.9%</td>
</tr>
</tbody>
</table>

It seems that the labels are in fact acceptably accurate.

The results for the randomly seeded systems indicate that equivalent or marginally improved performance can be achieved without the benefit of useful context labels when the number of clusters is increased. A marginal gain was observed in both the minimum DCF and EER criteria. These results demonstrate that the clustering algorithm does in fact converge to represent useful feature mapping contexts rather than similar speakers.

In this experiment, the randomly seeded clustering produced inferior results to labelled data when the same number of contexts was used, although this is still an improvement on the reference system. This difference in performance may be due to the rudimentary clustering algorithm converging to a local optimum. It is possible that a more advanced clustering algorithm would rectify this problem, which is an avenue for further research in this area.

Figure 4.5 presents the DET curves for these systems. The performance of each of the feature mapping systems are difficult to separate. From these figures no significant difference can be found between the different feature mapping systems although it is apparent that all variants provided a significant boost over the baseline system.

The results for 6 and 8 clusters are interesting as they seem to be capturing characteristics beyond the four handset type and gender combinations known to be represented in the training utterances. Table 4.2 investigates the cluster
Figure 4.5: DET plot of the standard and clustered feature mapping configurations on the QUT EDT ’03 protocol.
4.4. Blind Feature Mapping

Table 4.2: The correspondence between handset and gender labels and final context membership for the randomly seeded, 8 context system.

<table>
<thead>
<tr>
<th>Context</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
<th>C6</th>
<th>C7</th>
<th>C8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Carbon-Button</td>
<td>24</td>
<td>1</td>
<td>85</td>
<td>.</td>
<td>.</td>
<td>27</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Electret</td>
<td>121</td>
<td>.</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>.</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>Male</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Carbon-Button</td>
<td>.</td>
<td>97</td>
<td>4</td>
<td>14</td>
<td>13</td>
<td>9</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Electret</td>
<td>2</td>
<td>1</td>
<td>.</td>
<td>67</td>
<td>31</td>
<td>30</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>Total</td>
<td>147</td>
<td>99</td>
<td>95</td>
<td>83</td>
<td>46</td>
<td>39</td>
<td>34</td>
<td>17</td>
</tr>
</tbody>
</table>

membership of the training utterances for the 8-cluster system compared to the available handset and gender labels. In this table it can be seen that the four most populous classes are each dominated by one of known categories with some leakage into the near-by categories. Interestingly, the predominantly electret classes (C1 and C4) also contain a significant number of carbon-button utterances from the same gender while the reverse is not true of the clusters dominated by carbon-button recordings (C2 and C3). This observation raises the question whether this is an artefact of the clustering algorithm or inaccuracies in the provided automatic handset labels. This question is obviously difficult to answer given that ground truth is not available.

Of the smaller clusters, there seems to be less of a significant trend with more evenly distributed occupancies. It may be interesting to investigate further the main characteristics of these smaller clusters and the impact these characteristics have on performance. Looking at the size of the smaller clusters, particularly C8, may explain the negligible difference in performance compared to the 6-cluster system as this cluster is approaching the minimum size condition described above.

The experimental results presented demonstrate that the simple clustering algorithm proposed in this work is capable of producing clusters representing contexts that are useful for the purposes of feature mapping and produce equivalent performance with the same data to a system utilising labels.
Table 4.3: Minimum DCF and EER of clustered feature mapping in biased and unbiased configurations on the QUT EDT ’03 protocol.

<table>
<thead>
<tr>
<th>System</th>
<th># Contexts</th>
<th>Min. DCF</th>
<th>EER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td></td>
<td>.0465</td>
<td>14.0%</td>
</tr>
<tr>
<td>Labelled</td>
<td>4</td>
<td>.0392</td>
<td>11.3%</td>
</tr>
<tr>
<td>Biased</td>
<td>8</td>
<td>.0383</td>
<td>10.9%</td>
</tr>
<tr>
<td>Gender Biased</td>
<td>8</td>
<td>.0390</td>
<td>11.0%</td>
</tr>
<tr>
<td>Unbiased</td>
<td>8</td>
<td>.0401</td>
<td>10.8%</td>
</tr>
</tbody>
</table>

Unbiased cluster training

For the purposes of a fair comparison of techniques the previous experiment used the same selection of utterances for training the feature mapping context models in all cases. The selection of training utterances was made with knowledge of the detected handset labels to represent each context equally in the training data according to Reynolds’ feature mapping method \cite{96}.

A consequence of this is that the investigation of random clustering is biased by this initial utterance selection in a way that would not be possible without handset labels.

This second experiment addresses this bias by investigating random feature mapping contexts when the training utterances for the UBM and the feature mapping models are selected blindly from a set of utterances independent of the evaluation set. Two situations are considered in this experiment. Firstly, that although handset labels are not available for selecting utterances, it is assumed that gender information is, allowing for a gender-balanced selection. This situation is quite common in practice. The situation in which gender information is not available will also be investigated. This second scenario incurs the added limitation that gender-specific background models can not be used which have been shown to provide a small but consistent improvement in performance. As for the previous experiment, approximately 600 background utterances were selected from Switchboard-II phases 2 & 3.

Figure 4.6 and Table 4.3 present the results comparing the systems that used
4.4. Blind Feature Mapping

Figure 4.6: DET plot of clustered feature mapping in biased and unbiased configurations on the QUT EDT ’03 protocol.
the biased background data to the blindly selected (unbiased) background data for the clustered feature mapping method. The Biased system in row 3 of Table 4.3 corresponds to the last row of Table 4.1.

Comparing the results for the system with only gender information available (Gender Biased) to the systems with biased utterance selection indicates that the blind selection of training data does not cause a significant drop in performance. Further removing the gender labels produces a less than significant degradation in performance according to the detection cost function. This degradation is very much in line with the expected difference between the use of a single UBM and using a pair of gender-dependent UBMs. Somewhat surprisingly, the totally unbiased system demonstrated the best overall EER although the difference was minimal.

2004 NIST SRE development system

Feature mapping formed an integral part of the QUT submission for the 2004 NIST SRE [72]. As described in Section 2.2.1, this evaluation included both landline and cellular data drawn from the Mixer collection, and posed a significant new challenge compared to previous years.

For development and tuning purposes, the 2003 EDT protocol was used in combination with data from the cellular conditions of 2001 and 2002 as the EDT consisted only of landline data. Figure 4.7 demonstrates the performance of the submitted system on the development protocol.

For this system, eight contexts were modelled in the feature mapping configuration with female and male variants of electret, carbon-button, CDMA and GSM handset types. These contexts were then refined using the clustering algorithm described in this chapter. Modelling these eight contexts actually compared favourably for the development protocol to modelling only the four landline contexts represented in the data.

This system was most notable, however, for the amount of reuse of background data. Specifically, the limited quantity of cellular data available meant that all of this data was necessary for training the feature mapping contexts. This left
Figure 4.7: DET plot of the development system for the 2004 NIST SRE.

As can be seen from Figure 4.7, both the feature mapping and normalisation configurations were successful in this case. The combined normalisation techniques reduced the minimum DCF point from .0384 to .0274 or a relative reduction of 28%, with a 15% reduction in the EER. This system was among the most competitive in the 2004 Evaluation.

4.5 Summary

The translation of speaker recognition technology from the laboratory to public telephone systems has consistently highlighted the adverse affects of mismatch, particularly in the form of handset type mismatch. Handset type mismatch originally referred to differences in the type of microphone transducer used but more
recently has broadened to include differences such as the speech coding and trans-
mission processes used in wireless and digital environments.

Although a number of methods have been proposed to counter the degradation
imposed by handset mismatch, the performance for mismatched trials regularly
lags that of matched conditions by a factor of four in terms of equal error rate.

The recent introduction of the feature mapping technique has shown promise
in directly addressing the impact of handset mismatch by mapping feature vec-
tors extracted from different handset contexts to a neutral, handset-independent
space. This mapping is a non-linear transformation defined by the differences
between a GMM representing the neutral space and a set of adapted GMMs
representing each context.

Two extensions to feature mapping were proposed and investigated in this
chapter to expand the situations in which feature mapping can be successfully
applied. A method was proposed for combining the use of feature mapping with
feature warping to enhance the performance of the reference system without losing
the benefits of either technique.

Also presented was a method for adapting the original feature mapping
method to allow for effective training of feature mapping models in the absence
of context labels for the background data, as is often the case for practical ap-
lications. The experiments presented demonstrated the performance of systems
incorporating data-driven feature mapping models and found that the perfor-
ance provided with blindly selected background data was comparable to sys-
tems utilising the traditional approach to feature mapping training with fully
labelled data.
Chapter 5

Explicit Modelling of Session Variability

5.1 Introduction

The previous chapter described handset mismatch as the major cause of errors in speaker verification. While handset mismatch may be the biggest single cause of errors, this appraisal is a somewhat naïve and narrow minded description of the problem: Mismatch is not restricted to differences in handset type as there are a myriad of possible causes of mismatch. Other examples of mismatch in a telephony environment include a number of environmental factors such as nearby sources of noise (other people, cars, TV and music) and differing room acoustics (compare a stairwell to a park) — even holding a phone with your shoulder can cause significant mismatch due to differences in microphone position relative to the mouth. This list doesn’t even include many of the potential sources of mismatch introduced by the claimants themselves. All of these sources of mismatch have the potential to increase the rate of errors for a speaker verification system.

With this overwhelming number of possible variables the techniques presented to date cannot possibly hope to generalise well to all situations. For example, H-Norm and speaker model synthesis both require labelled training data, so not only is transcription necessary — and this is likely to be prohibitively expensive or
impossible in many cases — but the requirements for this data escalate rapidly with the variety of conditions modelled. Even though a successful method for implementing feature mapping without labelled training data was presented in the previous chapter it still suffers from similar issues with scaling as more variables are modelled. Other techniques such as T-Norm and feature warping attempt to suppress the types of effects that mismatch causes (to output scores in the case of T-Norm, to cepstral features in the case of feature warping) but do not actually know anything about the mismatch encountered.

This chapter proposes an approach to address the issue of mismatch in GMM-based speaker verification by explicitly modelling session variability in both the training and testing procedures and learning from the mismatch encountered. By directly modelling the mismatch between sessions in a constrained subspace of the GMM speaker model means, the proposed technique replaces the discrete categorisation of techniques such as feature mapping and H-Norm with a continuous vector-valued representation of the session conditions. The training methods used also remove the need for labelling the training data for particular conditions.

The motivation and aims of the approach are discussed in the next section followed by the proposed model for achieving these aims. Section 5.4 develops the tools and methods required for simultaneously estimating the session and speaker variables of the proposed model culminating in a novel and practical iterative approximation method based on the Gauss-Seidel method for solving linear systems.

Approaches to verification scoring using the proposed model are presented in Section 5.5 followed by the procedure for learning the characteristics of session variability from a background population of speakers.

The proposed approach to modelling session variability is empirically evaluated and compared to the classical GMM-UBM approach as well as blind feature mapping in Section 5.7. Results are presented for both Switchboard-II and Mixer conversational telephony corpora and the effects of several configuration options are explored.

Finally, the results and future directions for the proposed technique are dis-
5.2 Aims and Motivation

The aim of this work is to explicitly model the mismatch between different recorded sessions of the same speaker.

This mismatch between sessions of the same speaker is often referred to as *intra-speaker variability* but, as this term emphasises the differences in the performance of the speaker rather than the conditions of the session, the term *inter-session variability* or simply *session variability* will be preferred in this work as it affects a more accurate connotation.

The term session variability is defined to be very general and cover a wide variety of phenomena; specifically, any phenomenon that causes an observable difference from one recorded session to another for a given speaker. In a telephony environment, a far from exhaustive list includes environmental conditions such as background noise and room acoustics, the microphone transducer type and position relative to the mouth, transmission channel characteristics including coding artefacts in digital transmissions and factors introduced by the speaker, such as linguistic and emotional content of the session and health issues.

A number of techniques have been proposed to compensate for various aspects of session variability at almost every stage in the verification process with some success; a state of the art verification system will often incorporate a number of these techniques. An example system [72] from the NIST Speaker Recognition Evaluation might include feature warping [87] and mapping [96] to produce more robust features as well as score compensation techniques such as H- and T-Norm [99, 6].

These techniques fail to meet the goal stated above for different reasons, but they can be grouped into two major deficiencies.

The most common failing is only considering specific classes or sources of session mismatch; feature mapping, SMS, and H-Norm fall into this group. These techniques all have a common theme in that they all attempt to address some form
of categorical phenomena such as handset type. Assuming that the mismatch falls into categories greatly simplifies dealing with the issue but has several negative consequences.

Giving session conditions discrete labels does not generalise well. Apart from the issue that some characteristics are very difficult to describe in a discrete fashion, this can be demonstrated in general by noting that the only way to improve the representation of the mismatch encountered with these techniques is to add more variables to describe the mismatch. Modelling additional variables leads to an exponential growth in the number of categories. For example, adding a Boolean condition, such as whether the speaker is talking hands-free, will double the number of category labels. As is true for the methods listed above, doubling the number of categories also doubles the data required to train the method; hence the data requirements also grow exponentially.

Such categorical methods usually require base truth information on the characteristics they model. Accurate truth information is often impossible to acquire after the fact and will certainly be expensive if hand transcription is necessary. Techniques such as blind feature mapping can reduce the need for accurately labelled data as “truth” in this sense is simply defined by set membership for the training data rather than any specific real-world trait (although the sets may be designed to approximate real-world traits). This still causes issues for the test utterance as automatically detecting the appropriate set is a necessary and error-prone process, potentially causing errors in the verification process by applying an inappropriate normalisation.

The second major deficiency is not actually modelling the effects of session variability but simply attempting to quash them. Feature warping, T-Norm and Z-Norm [6] fit into this category. These methods have no knowledge of the particular conditions encountered in a recording but use some a priori knowledge of the effects that session conditions could have. As an example, feature warping was developed due to observing the non-linear compressing effect that additive noise has on cepstral features [85]. Rather than attempting to explicitly model this effect and learn how the cepstral features have been distorted for a specific session,
feature warping attempts to warp every utterance back to the same (standard normal) distribution, thus loosing any knowledge of the actual distortion encountered.

While it is obvious from the performance improvements that there is merit in the approaches used to date in speaker verification, the aim of this chapter is to tackle the problem of session variability by modelling it explicitly. Apart from overcoming the deficiencies of previous techniques there is some further motivation behind this approach. Specifically, modelling the prevalent conditions of a number of sessions from a particular speaker will provide an opportunity to learn more about the speaker. The goal here is specifically to provide better accuracy by learning more from the combination of the training and test utterances as a pair. Another goal is to more accurately estimate speaker parameters in the situation where multiple enrolment utterances are available. The argument here is that, knowing that there are multiple sessions with differing conditions, these conditions can be learnt separately and the speaker characteristics can be more easily observed and modelled by knowing the conditions under which they are being observed. This is in contrast to simply agglomerating multiple sessions together for enrolment and, in effect, averaging the session conditions into the speaker model estimate.

The following section describes the conceptual model used to realise these goals.

### 5.3 Modelling Session Variability

The approach used in this work is to model the effect of session variability in the GMM speaker model space. More specifically, the particular conditions of a recording session are assumed to result in an offset to each of the GMM component mean vectors. In other words, the Gaussian mixture model that best represents the acoustic observations of a particular recording is the combination of a session-independent speaker model with an additional session-dependent offset of the model means. This can be represented for speaker $s$ and session $h$ in terms of
the \( CD \times 1 \) concatenated GMM component means supervectors as

\[
\mu_h(s) = \mu(s) + U z_h(s), \tag{5.1}
\]

where the GMM is of order \( C \) and dimension \( D \).

Here, the speaker \( s \) is represented by the mean supervector \( \mu(s) \) which consists of the concatenated mixture component means, that is \( \mu(s) = [\mu_1(s)^T \cdots \mu_C(s)^T]^T \). To represent the conditions of the particular recording, designated with the subscript \( h \), an additional offset of \( U z_h(s) \) is introduced where \( z_h(s) \) is a low-dimensional representation of the conditions in the recording and \( U \) is the low-rank transformation matrix.

The presence of the term \( U z_h(s) \) fulfils the objective of explicitly modelling the session conditions stated above. Also, the issues related to using a categorical approach described in the previous section are addressed by using a continuous multi-dimensional variable \( z_h(s) \) to express this model.

Further, as the observed feature vectors are assumed to be conditional on both an explicit session-dependent part and a session-independent speaker part, this model also differs from the suppressive methods such as feature warping and T-Norm.

The likelihood function in this model is ostensibly identical to the standard GMM likelihood function, that is

\[
p(X_h|\mu_h(s)) = \prod_{t=1}^{T} \sum_{c=1}^{C} \omega_c g(x_t|\mu_{h,c}(s), \Sigma_c)
\]

where \( \mu_{h,c}(s) \) is portion of the supervector \( \mu_h(s) \) corresponding to component \( c \) and likewise for the component covariance matrix \( \Sigma_c \) and

\[
g(x|\mu, \Sigma) = (2\pi)^{-D/2}|\Sigma|^{-1/2} \exp\left(-\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu)\right)
\]

is the standard Gaussian kernel, as in (2.4) and (2.5).

One of the central assumptions in this formulation is that the majority of session variability can be described in a low-dimensional, linear subspace of the concatenated GMM mean vectors. In (5.1) this subspace is defined by the transform \( U \) from the constrained session variability subspace of dimension \( R_z \) to the
GMM mean supervector space of dimension $CD$ where $R_z \ll CD$; consequently all $z_h(s)$ are $R_z \times 1$ column vectors.

By knowing the effect that the session conditions can have on a speaker model, in the form of a session variability subspace, it is possible to distinguish between true characteristics of a speaker and spurious session artefacts. Assuming that the session subspace $U$ is appropriately trained and constrained to capture only the most significant session effects then any characteristics that can be explained in the subspace will be heavily dominated by session effects and hold a minimum of reliable speaker information.

An important aspect to the session variability modelling approach is that the subspace defined by $U$ is determined in an entirely data-driven manner using a corpus of background speakers without any requirements for labelling the session conditions. By observing the actual differences in component means for multiple recordings of the same speaker under a variety of session conditions, $U$ can be estimated without any knowledge of the specific session characteristics actually captured. While the corpus should reflect the anticipated deployment conditions and is preferably quite large, the composition of the corpus does not need to be as carefully balanced as is required for H-Norm or feature mapping; this makes much better use of the available training corpus as much less is potentially wasted due to balancing issues.

Returning to the model described in (5.1), it simply states that the true speaker characteristics are described by the concatenated mean supervector $\mu(s)$. There are a number of possibilities for how this supervector is estimated but there is one important restriction: Adaptation must be used for a subspace to describe the relationships of the component means, that is the speaker mean should comprise a shared speaker-independent mean plus a speaker-dependent offset. That is

$$\mu(s) = m + d(s)$$

where $m$ is the speaker-independent UBM mean and $d(s)$ is the speaker offset supervector. This requirement is necessary to ensure that the component means’ relationships modelled in $U$ hold between distinct speaker models. (This will
not be the case for instance with standard ML training of GMMs using the E-M algorithm.)

Classical relevance MAP adaptation is an example that fulfils this requirement, and this is the primary configuration used in this work. Another possibility is to also introduce a speaker variability subspace defined by the low rank transform matrix $V$ and adapt within that subspace, giving $\mu(s) = m + Vx(s)$, such as described for speaker verification by Lucey and Chen [65]. As this model contains far fewer variables than relevance MAP it potentially requires far less data to train but has the disadvantage of not asymptotically converging with an ML estimate. Kenny, et al. use a combination of classical relevance MAP and subspace adaptation [58] in a bid to get the best of both approaches, giving $\mu(s) = m + d(s) + Vx(s)$.

Ideally, the enrolment and verification algorithms will be able to accurately discern the session-independent speaker model $\mu(s)$ in the presence of session variability. These topics will be discussed in Sections 5.4 and 5.5 respectively. This will be followed by a description of the algorithm for training the session variability transform in Section 5.6.

### 5.4 Estimating the Speaker Model Parameters

The meaning of speaker enrolment and the overall process of speaker model training will be described in the next section as well as the criteria to be optimised. While MAP adaptation of GMM component means, referred to as relevance MAP, was addressed in Section 2.4 on page 38, MAP adaptation in a GMM mean subspace will be developed in Section 5.4.2. This method has also been referred to as probabilistic principal components analysis (PPCA) [65]. This result will be necessary for estimating the session and speaker factors.

This will be followed by a description of the joint estimation of the speaker and session variables of the proposed model, all adhering to MAP criteria. Due to the complexity of this procedure, a novel iterative approximation method will be presented in Section 5.4.4 based on the Gauss-Seidel method for solving si-
multaneous linear equations.

### 5.4.1 Speaker Model Enrolment

The goal of the enrolment process is to get the best possible representation of a speaker. According to the model described in (5.1) this information is contained in the concatenated GMM mean supervector $\mu(s)$ but this task is complicated by the prevalent conditions in the recording or recordings used for enrolment, represented by $z_h(s)$. Therefore the purpose of enrolment is to maximise the set of parameters $\lambda_s = \{ \mu(s), z_1(s), \ldots, z_H(s) \}$ to best fit the training data, but it is only necessary to retain the true speaker mean $\mu(s)$. That is, the posterior likelihood

$$ p(\lambda_s|X_1(s), \ldots, X_H(s)) = p(\mu(s), z_1(s), \ldots, z_H(s)|\lambda_s) p(\lambda_s) $$

$$ = p(\mu(s)) \prod_{h=1}^H p(z_h(s)) p(X_h(s)|z_h(s), \mu(s)) \quad (5.2) $$

must be maximised for all parameters. This is therefore a simultaneous optimisation problem.

The likelihood function of the observation data, $p(X_h(s)|z_h(s), \mu(s))$ is the standard GMM likelihood with the component means given by (5.1) and covariance matrices $\Sigma$. It can be seen that the speaker mean supervector $\mu(s)$ is optimised according to the MAP criterion often used in speaker verification systems [93]. The prior distribution $p(\mu(s))$ in this case is derived from a UBM, as previously described in Section 2.4.2.

The MAP criterion is also employed for optimising each of the session variability vectors $z_h(s)$. As described by Kenny et al. [58] the prior distribution in this case is set to be a standard normal distribution in the subspace defined by the transformation matrix $U$. The optimisation of such a criterion has previously been described for speaker recognition problems [58] [65].

Using the model described by (5.1) there are an infinite number of possible representations of any given value of $\mu_h(s)$ as the range of $Uz_h(s)$ is a subset of the range of $\mu(s)$. This is not an issue, however as the MAP criteria ensure that
there is not a “race condition” between the simultaneous optimisation criteria as the constraint imposed by the prior information ensures convergence to an optimum.

An E-M algorithm is used to optimise this model as there is no sufficient statistics for mixtures of Gaussians due to the missing information of mixture component occupancy of each observation. The following sections (5.4.2 to 5.4.4) only discuss the maximisation of the model parameters given an estimate of the mixture component occupancy statistics thus this is only the $M$ step of the full E-M algorithm. A full estimation procedure using these results will be an iterative approach also including the estimation part, which is identical to the $E$ step described in Section 2.4.

The following sections develop the tools necessary for speaker enrolment under the session variability modelling framework, concluding with a practical approximation method in Section 5.4.4.

5.4.2 MAP Estimation in a GMM Mean Subspace

Suppose we wish to estimate a GMM speaker model where the concatenated mean vectors are constrained to lie in a low-dimensional subspace. The model in this situation is

$$\mu = m + U z,$$

where $\mu$ is the $CD \times 1$ concatenated supervector of the GMM component means, $m$ is the prior mean, $z$ is the low-dimensional, $R_z \times 1$ vector variable to optimise and $U$ is a $CD \times R_z$ transformation matrix. For MAP estimation of this model the task is to estimate the variable $z$ which is assumed to have a standard normal distribution with zero mean and identity covariance, that is

$$z \sim N(0, I).$$

With this model it can be shown that $\mu$ has a covariance of $UU^T$ and is restricted to lie within the space defined by $U$.

Given this model and the prior distribution hyperparameters $\{m, U\}$ the
5.4. Estimating the Speaker Model Parameters

MAP estimate maximises

\[ p(X | \mu) p(\mu | m, U) = p(X | z, m, U) g(z | 0, I) \] (5.3)

where \( X = \{x_1, x_2, \ldots, x_t\} \) is the set of observation vectors and \( g(z | 0, I) \) refers to evaluating the standard Gaussian kernel at \( z \).

As with relevance adaptation, there is the missing information of which mixture component produced which observation. For this reason an iterative E-M approximation is used to optimise this model. The statistics required from the expectation step using this approach are the component occupancy count \( N_c \) and sample sum vector \( S_{Xc} \) for each mixture component \( c \), as defined in Section 2.4.2. Further, define \( S_X \) as the \( CD \times 1 \) concatenation of all \( S_{Xc} \) and \( N \) as the \( CD \times CD \) diagonal matrix consisting of \( C \) blocks along the diagonal of \( N_c = n_c I \) where \( I \) is the \( D \times D \) identity matrix.

With these quantities it can be shown that maximising the MAP criterion is equivalent to solving

\[ (I + U^T \Sigma^{-1} NU)z = U^T \Sigma^{-1} S_{X|m} \] (5.4)

for \( z \) where \( S_{X|m} = S_X - Nm \) is the first order statistics centralised on \( m \). This can be expressed in the conventional linear algebra form of

\[ Az = b \]

where \( A \) is a \( R_z \times R_z \) matrix and \( b \) is a \( R_z \times 1 \) column vector, given by

\[ A = I + U^T \Sigma^{-1} NU \] (5.5)

\[ b = U^T \Sigma^{-1} S_{X|m}. \] (5.6)

As \( A \) is a positive definite matrix this can be straightforwardly solved for \( z \) using the Cholesky decomposition method.

5.4.3 Simultaneous Relevance MAP and Subspace MAP Estimation

Before presenting the solution to simultaneous relevance and subspace MAP estimation, it is helpful to present relevance adaptation in a similar form to subspace
estimation using a standard normal prior. This result will be combined with the result of the previous section to simultaneously optimise in both a subspace and the full \( CD \)-sized speaker model space. Finally the solution of optimising with multiple sessions will be examined.

**Relevance MAP revisited**

The relevance MAP described in Section \[2.4.2\] can be expressed in the form

\[
\mu = m + Dy
\]  \hspace{1cm} (5.7)

where \( \mu \) and \( m \) have the same meaning as in the previous section and we are optimising the \( CD \times 1 \) vector \( y \) to maximise the same MAP criterion as the previous section also with a standard normal prior distribution. For equivalence with the previous development of relevance adaptation, the \( CD \times CD \) matrix \( D \) is set to be the diagonal matrix satisfying

\[
I = \tau D^T \Sigma^{-1} D
\]  \hspace{1cm} (5.8)

where \( \tau \) is the relevance factor.

According to the solution above, this can also be formed into a standard linear system of equations, \( Ay = b \), with

\[
A = I + D^T \Sigma^{-1} ND
\]

\[
= \tau D^T \Sigma^{-1} D + D^T \Sigma^{-1} ND
\]

\[
= D^T \Sigma^{-1} (\tau I + N) D
\]  \hspace{1cm} (5.9)

\[
b = D^T \Sigma^{-1} S_{X|m}.
\]  \hspace{1cm} (5.10)

Substituting back in and removing \( D^T \Sigma^{-1} \) from both sides,

\[
(\tau I + N) Dy = S_{X|m},
\]  \hspace{1cm} (5.11)

\[
y' = (\tau I + N)^{-1} S_{X|m},
\]  \hspace{1cm} (5.12)

where \( y' = Dy \) is the offset in the concatenated GMM mean space. It can be readily seen that \( (5.11) \) has a trivial solution as \( (\tau I + N) \) is a diagonal matrix and that it is equivalent to the relevance MAP adaptation solution presented in Section \[2.4.2\].
5.4. Estimating the Speaker Model Parameters

Optimising \( y \) and \( z \)

Having shown the equivalence of relevance MAP and subspace MAP estimation techniques given the appropriate transformation matrix \( D \), we can extend the result to optimise both \( y \) and \( z \).

Let \( z \) be the \((R_z + CD) \times 1\) column vector that is the concatenation of \( z \) and \( y \), and similarly let \( U \) be a \( CD \times (R_z + CD) \) concatenation of \( U \) and \( D \),

\[
U = \begin{bmatrix} U & D \end{bmatrix}.
\] (5.13)

With this notation it is then straightforward to formulate \( A \bar{z} = b \) in an analogous way to (5.5) and (5.6),

\[
A = I + U^T \Sigma^{-1} NU
\] (5.14)

\[
b = U^T \Sigma^{-1} S_{X|m}.
\] (5.15)

Unfortunately evaluating the solution to this equation directly is less than practical; it involves the decomposition of \( A \) which in this case is a \((R_z + CD) \times (R_z + CD)\) matrix. With the typical values of these dimensions this is a large task, especially as this matrix is not diagonal. It is, however, still positive definite.

There is still some sparsity to exploit as the lower right part will be diagonal.

Expressing \( A \) in terms of the blocks that it comprises

\[
A = \begin{bmatrix} I + U^T \Sigma^{-1} NU & U^T \Sigma^{-1} ND \\ D^T \Sigma^{-1} NU & I + D^T \Sigma^{-1} ND \end{bmatrix},
\] (5.16)

it can be seen that the \( CD \times CD \) block in the lower right region is given by \( I + D^T \Sigma^{-1} ND \) which has non-zero elements only on the diagonal. This trait can be exploited to solve the system using the identity for symmetric positive definite matrices [54],

\[
\begin{bmatrix} \alpha & \beta \\ \beta^T & \gamma \end{bmatrix}^{-1} = \begin{bmatrix} \zeta^{-1} & -\zeta^{-1} \beta \gamma^{-1} \\ -\gamma^{-1} \beta^T \zeta^{-1} & \gamma^{-1} + \gamma^{-1} \beta^T \zeta^{-1} \beta \gamma^{-1} \end{bmatrix}
\] (5.17)

where

\[
\zeta = \alpha - \beta \gamma^{-1} \beta^T.
\] (5.18)
Substituting this identity into the expression for \( A \) we have,

\[
\alpha = I + U^T \Sigma^{-1} NU \\
\beta = U^T \Sigma^{-1} ND \\
\gamma = I + D^T \Sigma^{-1} ND.
\]

Therefore the inverse of \( A \) can be determined by inverting the \( R_z \times R_z \) matrix \( \zeta \) and inverting \( \gamma \) which, while large, is diagonal.

The solution to the maximisation of our model parameters is thus given by

\[
z = A^{-1} b
= \begin{bmatrix}
\zeta^{-1} & -\zeta^{-1} \beta \gamma^{-1} \\
-\gamma^{-1} \beta^T \zeta^{-1} & \gamma^{-1} + \gamma^{-1} \beta^T \zeta^{-1} \beta \gamma^{-1}
\end{bmatrix}
\begin{bmatrix}
U^T \\
D^T
\end{bmatrix}
\begin{bmatrix}
\Sigma^{-1} S_{X|m}
\end{bmatrix}
\]

This results in the solution

\[
z = \zeta^{-1} (U^T - \beta \gamma^{-1} D^T) \Sigma^{-1} S_{X|m},
\]

and

\[
y = \gamma^{-1} D^T \Sigma^{-1} S_{X|m} - \gamma^{-1} \beta^T z
= \gamma^{-1} D^T \Sigma^{-1} (S_{X|m} - NU z).
\]

Comparing (5.11) and (5.23) and setting \( z = 0 \), it can be seen that the simultaneous solution for \( y \) is identical to (5.11) since \( \gamma^{-1} D^T \Sigma^{-1} = D^{-1}(\tau I + N)^{-1} \). With \( z \neq 0 \) the solution for \( y \) also includes a subtractive term that is a weighted version of the solution to the subspace variable \( z \); this is the contribution explained by the subspace variable. Evidently, the two variables are competing to describe the observations represented by the statistic \( S_{X|m} \).
5.4. Estimating the Speaker Model Parameters

While there is still a quite significant processing requirement to evaluate the simultaneous solution to \( y \) and \( z \), it is certainly feasible. Furthermore, there is one non-trivial matrix inversion required of \( \zeta \), which is only an \( R_z \times R_z \) matrix.

**Simultaneous solution with multiple sessions**

So far a simultaneous solution for \( y \) and \( z \) has been presented but this does not yet cover the case of most interest in this work. The proposed model attempts to capture the session conditions of each session in a session variability subspace to learn a more accurate representation of the speaker. Therefore it is necessary to find a MAP solution to the model in (5.1) repeated here for convenience,

\[
\mu_h = \mu + U z_h, \tag{5.24}
\]

where

\[
\mu = m + D y \tag{5.25}
\]

over all observed sessions \( X_h; h = 1, \ldots, H \). (The speaker label \( s \) has been dropped in this section for clarity as we are only dealing with a single speaker at this point.)

Our set of variables in this instance is similar to the previous section with the exception that there are \( H \) subspace variables to estimate,

\[
\begin{bmatrix}
  z_1 \\
  \vdots \\
  z_H \\
  y
\end{bmatrix}, \tag{5.26}
\]

which is a \((HR_z + CD) \times 1\) column vector variable. On the other hand the combined \( HCD \times (HR_z + CD) \) transformation matrix takes a more complicated form,

\[
U = \begin{bmatrix}
  U & \quad D \\
  \vdots & \vdots \\
  U & \quad D
\end{bmatrix}, \tag{5.27}
\]
due to the complication of multiple sessions. This is also true for the statistic $S_X$ defined as

$$S_X = \begin{bmatrix} S_{X,1} \\ \vdots \\ S_{X,H} \end{bmatrix}$$  \hspace{1cm} (5.28)

to allow the statistics of each session to be available independently. Similar definitions of the component occupancy statistics matrix $N$ and $\Sigma$ are also required, producing a $HCD \times HCD$ diagonal matrices. $N$ is simply the concatenation of all available $N_h$ while $\Sigma$ consists of $H$ repeats of $\Sigma$ along the diagonal. It will also be convenient to define $S_X = \sum_{h=1}^{H} S_{X,h}$ and $N = \sum_{h=1}^{H} N_h$.

It can be seen from these definitions that, for example, the product of $U \Sigma^{-1} S_X$ is a $(HR_C + CD) \times 1$ column vector.

Given these definitions, essentially the same $Az = b$ formulation of the optimisation problem as in (5.14) and (5.15) can be stated, that is

$$A = I + UT \Sigma^{-1} NU$$  \hspace{1cm} (5.29)

$$b = U^T \Sigma^{-1} S_{X|m}.$$  \hspace{1cm} (5.30)

Obviously the processing and memory requirement issues involved in solving this set of equations that were encountered in the previous section have increased with this formulation including multiple recording sessions. A parallel development of a practical solution will be followed in this section, taking into account the increased complexity.

Using the same identity to find $A^{-1}$,

$$\alpha = \begin{bmatrix} I + UT \Sigma^{-1} N_1 U \\ \vdots \\ I + UT \Sigma^{-1} N_H U \end{bmatrix}$$  \hspace{1cm} (5.31)

$$\beta = \begin{bmatrix} UT \Sigma^{-1} N_1 D \\ \vdots \\ UT \Sigma^{-1} N_H D \end{bmatrix}$$  \hspace{1cm} (5.32)
\[ \gamma = I + D^T \Sigma^{-1} ND \]  

(5.33)

and recall from (5.18)

\[ \zeta = \alpha - \beta \gamma^{-1} \beta^T. \]

This method requires inverting \( \zeta \) which in this case is an \( HR_z \times HR_z \) symmetric positive definite matrix. While inverting this matrix will be much faster than inverting \( A \) directly, the cost of this operation is \( O(H^3 R_z^3) \). This cost is therefore very sensitive to both the number of sessions and size of the session subspace; both of which can potentially limit the feasibility of this model. Fortunately both of these factors tend to be quite reasonable in this work.

Similarly to the result for a single session,

\[
\begin{bmatrix}
\zeta^{-1} & -\zeta^{-1} \beta \gamma^{-1} \\
-\gamma^{-1} \beta^T \zeta^{-1} & \gamma^{-1} + \gamma^{-1} \beta^T \zeta^{-1} \beta \gamma^{-1}
\end{bmatrix}
\begin{bmatrix}
U^T \Sigma^{-1} S_{X,1|m} \\
\vdots \\
U^T \Sigma^{-1} S_{X,H|m} \\
D^T \Sigma^{-1} S_{X|m}
\end{bmatrix}
\]

giving

\[
\begin{bmatrix}
z_1 \\
\vdots \\
z_H
\end{bmatrix}
= \zeta^{-1}
\begin{bmatrix}
U^T \Sigma^{-1} (S_{X,1|m} - N_1 \delta) \\
\vdots \\
U^T \Sigma^{-1} (S_{X,H|m} - N_H \delta)
\end{bmatrix},
\]  

(5.34)

where

\[ \delta = D \gamma^{-1} D^T \Sigma^{-1} S_{X|m} \]  

(5.35)

and

\[
y = \gamma^{-1} D^T \Sigma^{-1} S_{X|m} - \gamma^{-1} \beta^T z_{1,...,H} \\
= \gamma^{-1} D^T \Sigma^{-1} \left( S_{X|m} - \sum_{h=1}^H N_h U z_h \right).
\]  

(5.36)
In the case of classical relevance adaptation with $D$ satisfying (5.8), it simplifies these solutions with

$$\delta = (\tau I + N)^{-1} S_{X|m},$$

and

$$y = D^{-1}(\tau I + N)^{-1}\left(S_{X|m} - \sum_{h=1}^{H} N_h U z_h\right)$$

### 5.4.4 Gauss-Seidel Approximation Method

While a practical solution to the simultaneous MAP estimation of multiple session variables and the speaker mean offset was presented in the previous section, the solution is still very expensive in terms of processing requirements. In fact it is still so expensive as to become impractical if a reasonable number of speakers, each with a reasonable number of sessions, are to be estimated — such as is the case for a NIST evaluation that typically involves training thousands of models. Also it is worth noting that the solutions above are merely for the maximisation step of an E-M algorithm where multiple iterations are required before an accurate estimate of the missing mixture component occupancy information is realised.

An approximation method with more modest processing requirements is desirable.

This section presents such a method inspired by the iterative Gauss-Seidel method for solving linear systems of equations [11].

Iterative methods for solving linear systems of equations are often preferred for solving very large systems where direct solutions would be prohibitively expensive to calculate. Iterative methods can also be used to improve the accuracy of a direct solution where floating point precision issues have incurred rounding and accumulation errors.

The Gauss-Seidel method specifically is one of the simplest iterative methods for solving linear equations that comes from the family of stationary iterative methods and is a slight modification of the Jacobi method.

Using the Jacobi method for solving the linear system $Ax = b$ a succession
of improved estimates of each element $x_i$ of $\mathbf{x}$ are given by

$$x_i^{(k)} = a_{ii}^{-1} \left( b_i - \sum_{j \neq i} a_{ij} x_j^{(k-1)} \right)$$

where the superscript $(k)$ refers to the current iteration and $(k-1)$ the previous. It is also assumed that some initial guess of the solution is available to initialise the algorithm for the first iteration — this is often set to be $\mathbf{x} = \mathbf{0}$ if no informative guess is available. Essentially the trivial solution for $x_i$ is found by setting the value for all of the other variables $x_j; j \neq i$ to their previous estimate.

The improvement on this made for the Gauss-Seidel method is to use the most current available estimate for the other variables. Assuming that $\mathbf{x}$ is estimated in the order $x_1, x_2, \ldots$ then to estimate $x_2^{(k)}$ the value $x_1^{(k)}$ is used rather than the previous estimate $x_1^{(k-1)}$. This gives the iterative update equation

$$x_i^{(k)} = a_{ii}^{-1} \left( b_i - \sum_{j < i} a_{ij} x_j^{(k)} - \sum_{j > i} a_{ij} x_j^{(k-1)} \right) .$$

As the elements of $\mathbf{x}$ are re-estimated in order, the new estimates are used as soon as they are available to enhance the accuracy of estimating subsequent elements. In comparison to the Jacobi method, using the new estimates in this way instead of using only the old estimates provides improved convergence rates.

Extending the idea of iterative approximation to the simultaneous solution of the speaker model with session variability, the speaker mean offset and each of the session condition variables are solved assuming the estimate of all other variables is fixed. In this way, the speaker mean offset $y$ can be estimated with the usual relevance MAP adaptation equations assuming that the session conditions $z_h$ are all known. Similarly, the session variables $z_h$ for $h = 1, \ldots, H$ can each be estimated assuming that $y$ is known. This estimation process for each of the variables is repeated until the result converges on the optimal solution.

As with the direct solution presented in the previous section, this is only the solution to maximising the MAP criterion and forms only the $M$ step of an E-M algorithm. Due to the missing information of the mixture component allocations of the training data, an iterative algorithm is also required on this level to converge on the optimal result. The complete algorithm for estimating the speaker model and the session condition variables is presented in Algorithm 2.
Algorithm 2 Speaker Model Estimation

1: \( y \leftarrow 0; z_h \leftarrow 0; h = 1, \ldots, H \)

2: for \( i = 1 \) to No. E-M iterations do

3: \hspace{0.5cm} \textit{E Step}:

4: \hspace{1cm} for \( h = 1 \) to \( H \) do

5: \hspace{1.5cm} Calculate \( N_h \) and \( S_{X,h} \) for session \( X_h \) where \( \mu_h = m + Dy + Uz_h \)

6: \hspace{1cm} end for

7: \hspace{1cm} \( N \leftarrow \sum_{h=1}^{H} N_h \)

8: \hspace{1cm} \( S_X \leftarrow \sum_{h=1}^{H} S_{X,h} \)

9: \hspace{0.5cm} \textit{M Step}:

10: \hspace{1cm} for \( j = 1 \) to No. Gauss-Seidel iterations do

11: \hspace{1.5cm} for \( h = 1 \) to \( H \) do

12: \hspace{2cm} \( z_h \leftarrow A_h^{-1} b_h \)

where \( A_h = I + U^T \Sigma^{-1} N_h U \)

and \( b_h = U^T \Sigma^{-1} \left( S_{X,h|m} - N_h Dy \right) \)

13: \hspace{1.5cm} end for

14: \hspace{1.5cm} \( y \leftarrow A_y^{-1} b_y \)

where \( A_y = I + D^T \Sigma^{-1} ND \)

and \( b_y = D^T \Sigma^{-1} \left( S_{X|m} - \sum_{h=1}^{H} N_h U z_h \right) \)

15: \hspace{1.5cm} end for

16: \hspace{1cm} end for

17: return \( y \)
5.4. Estimating the Speaker Model Parameters

In this algorithm, the expectation or $E$ step is essentially the same as for standard E-M algorithm for GMM training with the caveat that the session statistics are gathered separately and the Gaussian means also include a session-dependent offset.

The maximisation or $M$ step uses an iterative solution. Following the iterative method, each variable is optimised assuming the values of all other variables is known, as described above. The resulting solutions are given by

$$z_h = (I + U^T \Sigma^{-1} N_h U)^{-1} U^T \Sigma^{-1} (S_{X|h|m} - N_h D y), \quad (5.37)$$

$$y = \gamma^{-1} D^T \Sigma^{-1} \left( S_{X|m} - \sum_{h=1}^{H} N_h U z_h \right). \quad (5.38)$$

These are, respectively, the subspace MAP and relevance MAP solutions with compensated $b$ vectors, as emphasised on Lines 12 and 14.

Comparing these solutions with the direct solutions for multiple sessions, the solution for the speaker mean offset $y$ takes an identical form ((5.36) and (5.38)) that is dependent on the solution to $z_h$. This is somewhat misleading as the actual resulting values are potentially quite different due to the differing solutions for the session variables. As can be seen in (5.34) and (5.37) the direct solution for the session variables is significantly more involved, requiring the inversion of a larger matrix and also coupling the results from all of the session variables together. On the other hand, the iterative approximation is independent of the other sessions and relies solely on the most recent approximation of $y$.

The initial guesses of all variables in this algorithm is chosen to be 0. Given that the aim is to optimise a MAP criterion for each variable this is a reasonable assumption as the standard normal prior distribution is also assumed to have a zero mean. After the first iteration of the E-M algorithm, the initial guess for the Gauss-Seidel maximisation part of the algorithm will be initialised with the results of the previous iteration which will be a much better guess than 0, leading to better convergence rates in subsequent iterations. This refinement of previous estimates is a strength of an iterative approximation method.

The processing requirements for this algorithm grow linearly with the number of sessions used for training, which was the goal of this method, and only $H$
matrix decompositions are necessary of size $R_z \times R_z$. A large value for $R_z$ would be required before these decompositions start to dominate the processing time; for the values used in this study the algorithm is dominated by the $E$ step of calculating the statistics $N_h$ and $S_{X,h}$ for each session (Line 5).

**Behaviour of the Gauss-Seidel approximation**

There are several interesting aspects to this algorithm which deserve some exploration.

Given that the E-M algorithm for Gaussian mixture models generally converges to a *local* optimum, it is possible for different solutions to occur for the same data with different initialisation for each iteration. The implication for the approximation method described in Algorithm 2 is the potential to converge to a different local optimum to the direct solution method of Section 5.4.3. While this will not happen with a fully converged G-S solution as it will match the direct solution at the end of each E-M iteration, it can occur if full convergence is not achieved.

So the relevant question to arise from this observation is, how many iterations of the Gauss-Seidel method are necessary for convergence? Or, more practically, how many iterations are necessary for optimal verification performance?

These questions are complicated by the fact that changing the order of evaluating the estimates in the Gauss-Seidel method will effect the intermediate approximations of the variables. The algorithm described above estimates the session variables first but could just as easily be formulated to estimate the speaker first. This should not effect the final converged result to the system of linear equations but does impact on the rate of convergence and the intermediate estimates.

Figures 5.1, 5.2 and 5.3 demonstrate the effect of using only one iteration of the Gauss-Seidel approximation with estimating the session variables first (as described in Algorithm 2) compared to estimating the speaker offset first. Both variants are compared to a fully converged G-S estimate and estimating the session and speaker variables independently of each other for each E-M iteration. (While the magnitudes graphed in these plots cannot be directly used to assess
5.4. Estimating the Speaker Model Parameters

The most significant point of these figures for the current discussion is the similarity between the single iteration, session first method and the fully converged result. These results are so similar that they are almost indistinguishable in all figures. For the speaker first method this is also true of the speaker vector, $y(s)$ from around 14 iterations of the E-M algorithm but the session vectors do not share this similarity. It would seem, however, that in the case of this example all of these methods will eventually converge to the same result.

Interestingly, the independent estimation method seems to have little in common with any of the Gauss-Seidel variants and seems unlikely to converge to the same result; the session variables seem to stabilise after only a few iterations to a very different result to the other methods while the estimate of the speaker variables is larger in magnitude than the standard MAP adaptation. These factors indicate that this method will indeed converge to a different local minimum to the fully converged G-S approximation.
Figure 5.2: Plot of the session variability vector magnitude, $|z_h(s)|$, for differing optimisation techniques as it evolves over iterations of the E-M algorithm.

Figure 5.3: Plot of the expected log-likelihood of the training data for differing optimisation techniques as it evolves over iterations of the E-M algorithm.
5.5 Verification

It is not possible to draw conclusive statements based on the single example depicted above although the single iteration, session first estimate appears to be a close approximation to the fully converged estimate. This may allow for more efficient speaker enrolment procedures for equivalent verification performance. This possibility will be investigated further in Section 5.7.4 as will the effect on performance of the other variants described in this comparison.

5.5 Verification

The previous section developed the procedure for enrolling a speaker with a model incorporating session variability using simultaneous optimisation of speaker and session variables. This section extends this treatment to the verification stage of the system. To this end, the session variation introduced in the verification utterance must also be considered.

There are a number of possible methods of implementing verification in a session variability modelling scenario to make full use of the proposed model and they vary considerably in complexity and sophistication. The discussion of these candidate methods begins with proposed variants of common top-$N$ ELLR scoring, moving on to an extension of the Bayes factor approach described in Chapter 3. This section will then be concluded with a discussion of the factor analysis likelihood ratio championed by Kenny, et al. [58].

5.5.1 Top-$N$ ELLR with Session Variability

An expected log likelihood ratio (ELLR) score takes the general form

$$\Lambda_s(X_v) = \frac{1}{T} \log \frac{\ell_s(X_v)}{\ell_0(X_v)} \quad (5.39)$$

where the $X_v$ is the set of verification trial observations, $T$ is the number of observation vectors, $\ell_s(\cdot)$ is the likelihood score for the speaker $s$ and $\ell_0(\cdot)$ is the background likelihood based on the UBM. (To aid clarity, the parameterisation by $X_v$ will be omitted for the rest of this section where it is obvious due to context.)
The simplest approach to verification under the session variability framework is to continue to use ELLR scoring as is traditionally used with GMM-UBM verification systems. By taking this approach the conditions of the verification session are completely ignored but performance gains are still possible over standard GMM-UBM systems assuming that the training procedure produced a speaker model that more accurately represents the speaker. This is a reasonable assumption given that the point of the training procedure was to separate the speaker and session contributions as separate variables rather than modelling a combination of speaker and session conditions; particularly with multiple training sessions to distinguish between speaker and session effects this should be the case. According to the model proposed in (5.1) this is equivalent to calculating the ratio of likelihoods

\[ \ell_s = p(X_v | \mu_v(s) = \mu(s)) \]

(5.40)

where the session variable has been set to \( z = 0 \).

Using standard scoring methods with the improved training can only ever hope to address half of the mismatch issue; it may be possible to determine the speaker characteristics sans session effects but comparing this to a verification trial with session effects still entails mismatched conditions.

One possible way of dealing with the mismatch introduced by the verification utterance is to estimate the session variable \( z_v(s) \) of the utterance for each speaker prior to performing standard top-\( N \) ELLR scoring. Under this approach the likelihood score for a speaker is given by

\[ \ell_s = \max_z p(X_v | \mu_v(s) = \mu(s) + Uz) g(z|0, I) \]

(5.41)

This likelihood is essentially the MAP criterion used in Section 5.4.2 however in this case the evaluation of the likelihood is the desired result rather than determining the argument \( z \) that maximises it, although \( z \) is a necessary by-product.

The estimation procedure for \( z \) is similar to that described in Section 5.4.2 with a few differences. These differences are due to the context in which this estimation occurs. Often (5.41) must be evaluated for several models for the same
5.5. Verification

verification trial — at least the target and background model but many more if T-Norm score normalisation is to be used — so efficiency is very important.

To substantially reduce the processing required, a simplification is made in that the mixture component occupancy statistics for the observations are calculated based on the UBM rather than independently for each model to be scored. This allows for a solution that calls for only one additional pass of the verification utterance than standard top-N ELLR scoring and implies that only one matrix decomposition is necessary, regardless of the number of speakers being tested. Also, only a single adaptation step is used as, without re-aligning the observation vectors, more iterations would not produce a different result.

It is interesting to note the role of the prior distribution of $z$ in (5.41). While its presence is necessary to mirror the MAP criterion used for estimating the session variables in the training algorithm, the effect it has is to penalise models that require a large session compensation offset compared to those that are “closer” to matching the recording. In practice the presence of the prior is insignificant in terms of verification performance as its contribution to the overall score is dwarfed by that of the observation vectors.

An obvious extension to the likelihood function described in (5.41) is to also consider the speaker mean as a variable in the verification process, rather than as a value considered “known” after estimation during enrolment. Under this circumstance, the training algorithm is seen as estimating the posterior distribution of the speaker mean after observing the training data rather than estimating its value directly, in a similar way to the Bayes factor approach in Chapter 3. This leads to the formulation

$$
\ell_s = \max_{\mu, z} p\left(X_v|\mu_v(s) = \mu(s) + D(s)y + Uz\right) g(z, y|0, I) \quad (5.42)
$$

where $D(s) = \left(D^{-2} + \Sigma^{-1}N\right)^{-\frac{1}{2}}$ is the speaker-dependent transform after observing the enrolment data. Here $y$ is an additional offset to the speaker supervector mean $\mu(s) = m + Dy(s)$ to find the optimal fit to the combination of the training and testing data. The formulation of $D(s)$ reflects the posterior covariance of $Dy(s)$ which is approximated by $D^T \left(I + \Sigma^{-1}N\right) D$, ignoring the
cross correlation between $y(s)$ and the session variables $z_h(s); h = 1, \ldots, H(s)$.

Again, this is essentially MAP estimation of the model parameters followed by standard top-$N$ ELLR scoring with the prior of the additional speaker offset parameter adjusted according to the observed training data. The methods developed for enrolment can be effectively used for this estimation procedure but for efficiency reasons the same approximations used to evaluate (5.41) also apply.

This method is fundamentally different to the Bayes factor approach in that it finds the maximum of the posterior likelihood of the verification observations and the model parameters rather than finding the marginal likelihood of the verification observations by integrating over the entire space of the model parameters, as described in Chapter 3.

These methods will be empirically examined to determine the usefulness of the added complexity of embedding MAP estimation of the session and speaker variables in the verification scoring process.

### 5.5.2 Bayes Factors

The Bayes factor verification score introduced in Chapter 3 is designed to account for the uncertainty in the estimates of speaker model parameters. The conclusions drawn in that chapter indicated that there was merit to the approach in well matched conditions but it was apparently more adversely effected by mismatch than ELLR scoring.

Intuitively, this conclusion suggests that Bayes factors may work particularly well when used in conjunction with explicit modelling of the session mismatch through the framework introduced in this chapter as the issue of mismatch should be greatly reduced.

Following the previous development of the Bayes factor for speaker verification, the challenge is to evaluate the Bayesian predictive density of the available evidence conditional on each hypothesis. This essentially involves integrating the likelihood of the evidence over all possible values for the model parameters. In Chapter 3, the model parameters simply consisted of the speaker model mean but this chapter extends the model by incorporating the session variable, thus giving
the desired predictive density of the form

$$P_s(X_v) = \int \int p(X_v|y, z, \lambda_s) \ p(y|\lambda_s) \ p(z) \ dy \ dz. \quad (5.43)$$

As in the Chapter 3, there is no closed form solution to this value due to the issues raised by the weighted sum in the GMM likelihood function.

It may be possible however to evaluate this integral for a single observation at a time, thus allowing an incremental learning approach to be applied as described in 3.4.1. Several issues need to be resolved for this however. Firstly, the closed form solution of (5.43) for a single observation must be determined. This may not be straightforward as the solution in (3.17) relied on separating the problem into independent integrations over single variables which in this case is not possible as the session variables and the speaker model mean are directly linked through the subspace transform $U$. Secondly, the incremental update to the model parameter posterior distributions is significantly more involved; this will require at least solving a system of equations for the session variables after every observed frame of speech. This is likely to be a very expensive operation.

Finding a practical solution to these issues therefore is left as a future direction of this research.

5.5.3 Factor Analysis Likelihood Ratio

Kenny, et al. describe another alternative to providing a verification score with similar intent to the Bayes factor approach [58, 54, 55]. Here the intention is to evaluate the ratio of likelihoods given by (5.43) as for the Bayes factor method however the value $p(X_v|y, z)$ is very different; as stated in [56],

$$\log p(X_v|y, z) = \sum_{c=1}^{C} \left( n_c \log(2\pi)^{-D/2} |\Sigma_c|^{-1/2} \right. \\
- \frac{1}{2} \sum_{t_c} \left( x_{t_c} - \mu_{v,c}(s) \right)^T \Sigma_c^{-1} \left( x_{t_c} - \mu_{v,c}(s) \right)$$

where $t_c$ ranges over the observations allocated to component $c$. Assuming a hard alignment of observed frames to components this is equivalent to the likelihood
function
\[ p(X_v|y, z) = \prod_{c=1}^{C} \prod_{t_c} g(x_{t_c}|\mu_{v,c}(s), \Sigma_c) \]

where \( t_c \) has the same range as above. This function is obviously different to the normal likelihood function for mixtures of Gaussians but is significantly easier to evaluate the required integrals of [5.43], as demonstrated by comparing the resulting closed-form solutions in [58] to the incremental approach for evaluating the Bayes factor adopted in Chapter 3 and proposed above. This difference is essentially due to the difficulty of separating variables in the Bayes factor case where a product of sums is involved (over observed frames and mixture components, respectively).

Extending this to a soft alignment, where the probability of an observation being produced by each component is estimated,

\[
\log p(X_v|y, z) = \sum_{c=1}^{C} \left( n_c \log(2\pi)^{-D/2}|\Sigma_c|^{-1/2} - \frac{1}{2} \sum_{t=1}^{T} P(c|x_{t})(x_{t} - \mu_{v,c}(s))^T \Sigma_{c}^{-1}(x_{t} - \mu_{v,c}(s)) \right)
\]

where (presumably, although this is unclear from the available literature)

\[ P(c|x) = \frac{\omega_c g(x|\mu_{v,c}(s), \Sigma_c)}{\sum_{d=1}^{C} \omega_d g(x|\mu_{v,d}(s), \Sigma_d)} \]

The equivalent likelihood function is therefore

\[ p(X_v|y, z) = \prod_{t=1}^{T} \prod_{c=1}^{C} g(x_{t}|\mu_{v,c}(s), \Sigma_c)^{P(c|x_{t})}. \]

This is even further from the normal understanding of the GMM likelihood function than the hard alignment case, but is similarly straightforward to integrate.

It is interesting to note that these functions are the forms that are \textit{actually} maximised in the \( M \) step of the E-M algorithm [14]: They are much easier to differentiate and deal with than the actual likelihood function but are not necessarily maximised by the same values as the true GMM likelihood function, as described in Section 2.4.

This class of verification score will not be further considered in this work as they are not based on the GMM likelihood. They are presented here for comparison purposes.
5.6 Training the Session Variability Subspace

For the session variation modelling described in this chapter to be effective, the constrained session variability subspace described by the transformation matrix $U$ must represent the types of intra-speaker variations expected between sessions. To this end, the subspace is trained on a database containing a large number of speakers each with several independently recorded sessions. Preferably this training database will include a variety of channels, handset types and environmental conditions that closely resembles the conditions on which the eventual system is to be used.

This section describes the procedure for optimising the session transform matrix $U$ for a population of speakers by building on the results of Sections 5.4. Firstly, a straightforward method of estimating the transform using a principal components approach is described. An E-M algorithm is then presented that fully optimises the $U$ for all of the available data.

5.6.1 Principal Components of Session Variability

The simplest method of estimating the session variability transform is to observe the differences of models trained for the same speaker from different recordings for a group of speakers and determine the principal components of this variation.

Given a set of recordings $X_h(s); h = 1, \ldots, H(s)$ for a group of speakers $s = 1, \ldots, S$, a model is first estimated for each recording using classical relevance MAP adaptation. This gives a set of adapted GMM mean supervectors $\mu_h(s)$. This set of mean supervectors, minus the UBM mean $m$ from which they were adapted, then form the samples of a standard principal components analysis (PCA).

The within-class scatter matrix for this analysis is given by

$$S_W = \frac{1}{R} \sum_{s=1}^{S} \sum_{h=1}^{H(s)} (\mu_h(s) - \bar{\mu}(s)) (\mu_h(s) - \bar{\mu}(s))^T$$

where

$$\bar{\mu}(s) = \frac{1}{H(s)} \sum_{h=1}^{H(s)} \mu_h(s)$$
is the mean of the mean supervectors for speaker $s$ and $R = \sum_{s=1}^{S} H(s)$.

As $S_W$ is a large $CD \times CD$ matrix, it is typically too large to directly perform eigenvalue analysis but it usually has significantly lower rank, with a maximum possible rank of $R$. Thus an equivalent eigenvalue problem can be constructed with an $R \times R$ matrix as described in [36] (pages 35–37).

Taking the eigenvalue decomposition of the scatter matrix gives the form

$$S_W = X\Lambda X^T$$

where $\Lambda$ is the diagonal matrix of eigenvalues and $X$ is the matrix with the corresponding eigenvectors as its columns. Using $X$ as a transform therefore diagonalises the observed within class scatter resulting in the matrix $\Lambda$. The desired behaviour for the transform $U$ is to whiten this scatter matrix in order to use the standard normal distribution with covariance $I$ as the prior distribution of the session variable $z$, therefore the desired decomposition is

$$S_W = UIU^T.$$ 

Simple manipulation results in the expression

$$U = X\Lambda^{-\frac{1}{2}}. \quad (5.44)$$

The number of (non-zero) columns of $U$ is at most $R$ and is determined by the rank of the scatter matrix but in practice only the columns corresponding to the largest eigenvalues are retained.

### 5.6.2 Iterative Optimisation

Estimating the principal components of the variation observed in speaker model training provides a starting point for estimating the session subspace but, as it does not use the same simultaneous estimation training method as described in Section [5.4] it will not provide optimal results.

To most accurately model the speaker and the session variability the session subspace must be found that maximises the total *a posteriori* likelihood of all
segments in the training database by training a model for each speaker represented using the procedure in section 5.4. That is, $U$ must satisfy

$$U = \arg \max_U \prod_{s=1}^S p(\lambda_s | X_1(s), \ldots, X_{H(s)}(s)). \quad (5.45)$$

As the speaker and corresponding session variables are hidden in this optimisation procedure, another E-M algorithm is used. This procedure is described in detail in [56], with the caveat that a modified speaker model training procedure was used.

Briefly, the iterative optimisation of the subspace proceeds as follows: Firstly, an initial estimate of $U$ is used to bootstrap the optimisation. The PCA estimate described above is appropriate for this task as the better the initial estimate the more quickly the iterative method will converge. Then for the following iterations there are successive estimation and maximisation steps.

The $E$-step in this algorithm involves estimating the parameter set $\lambda_s = \{y(s), z_1(s), \ldots, z_{H(s)}(s)\}$ for each speaker $s$ in the training database using the current estimate of the session subspace transform $U$. This estimation follows the speaker enrolment procedure described in Section 5.4 above.

The $M$-step then involves maximising (5.45) given the expected values for $\lambda_s$. Using the notation of Section 5.4.4, this maximisation is equivalent to solving the system of equations

$$\sum_{s=1}^S \sum_{h=1}^{H(s)} N_h(s) U (z_h(s) z_h(s)^T + A_h^{-1}(s)) = \sum_{s=1}^S \sum_{h=1}^{H(s)} (S_{X,h|m} - N_h(s) D y(s)) z_h(s)^T \quad (5.46)$$

for $U$. Using the notation $U_c$ to represent the rows of $U$ corresponding to the $c$th mixture component — that is rows $cD+1$ to $(c+1)D$ — and similarly for the other variables, this can be rewritten as

$$U_c A_c = B_c \quad (5.47)$$
where

\[ A_c = \sum_{s=1}^{S} \sum_{h=1}^{H(s)} n_{c,h}(s) \left( z_h(s)z_h(s)^T + A_h^{-1}(s) \right) \]  \hspace{1cm} (5.48)

\[ B_c = \sum_{s=1}^{S} A_h(s) = \sum_{h=1}^{H(s)} \left( S_{X,c,h|m} - n_{c,h}(s)D_c y_c(s) \right) z_h(s)^T \]  \hspace{1cm} (5.49)

which is a straightforward system of equations that can be solved in the usual way for \( U_c \).

As stated in [58] this optimisation converges quite slowly and requires significant processing resources, however, empirical experiences with the process indicate that there is little improvement in verification performance to be gained with a fully converged algorithm; 10 iterations of the E-M algorithm proved to be more than sufficient. Indeed it has been argued that the principal components analysis used to seed the E-M algorithm may provide the required performance [57]. The sensitivity of this approach to the quality of session transformation will be further investigated empirically in terms of verification performance in Section 5.7.6.

5.7 Experiments

The baseline recognition system used in this study is described in Section 2.5 on page 44.

5.7.1 Switchboard-II Results

The proposed session variability modelling technique was initially evaluated on data from the Switchboard-II conversational telephony corpus. By design, this corpus exhibits a wide variety of session conditions including a variety of landline handset types used over PSTN channels in a number of locations. As participants in the collection were encouraged to use different telephones on different numbers throughout the collection, this corpus is well suited for evaluating the suitability of the session modelling methods and also training the required session subspace.

The QUT EDT 2003 protocol was used for these experiments (Section 2.2.1). Specifically, results are presented for the Development split of this protocol and
the evaluation splits were used as background data for training the UBM, session variability subspace transform $U$ and score normalisation techniques.

Figures 5.4 and 5.5 show DET plots comparing systems with and without session variability modelling for the 1- and 3-side training conditions respectively. Table 5.1 presents the minimum DCF and EER performance corresponding to these DET plots.

With no score normalisation applied, the session modelling technique provided a 32% reduction in DCF for the 1-side condition and a 54% reduction in the 3-side condition with similar trends in EER. While the improvement in the 3-side training condition is very substantial, the 1-side result is at least as interesting and, in many ways, more surprising and encouraging: In the 1-side condition, there was not multiple sessions from which to gain a good estimate of the true speaker characteristics by factoring out the session variations, however, the technique successfully factored out the variations between the training and testing
Table 5.1: Minimum DCF and EER of the baseline system and session variability modelling on Switchboard-II data.

<table>
<thead>
<tr>
<th>System</th>
<th>Raw Scores</th>
<th>Z-Norm</th>
<th>ZT-Norm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min. DCF</td>
<td>EER</td>
<td>Min. DCF</td>
</tr>
<tr>
<td><strong>1-Side</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0458</td>
<td>13.6</td>
<td>.0415</td>
</tr>
<tr>
<td>Session Modelling</td>
<td>.0311</td>
<td>9.0</td>
<td>.0251</td>
</tr>
<tr>
<td><strong>3-Side</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0243</td>
<td>5.9</td>
<td>.0252</td>
</tr>
<tr>
<td>Session Modelling</td>
<td>.0110</td>
<td>2.8</td>
<td>.0089</td>
</tr>
</tbody>
</table>

sessions.

Also presented are results with normalisation applied to all systems. The normalisations applied were Z-Norm to characterise the response of each speaker model to a variety of (impostor) test segments followed by T-Norm to compensate for the variations of the testing segments, such as duration and linguistic content. Again the proposed technique outperforms the baseline system, but also in fact gains more from this normalisation process than the baseline system with the improvements in DCF growing to 48% and 68% respectively for the 1- and 3-side conditions.

The benefits gained with Z-Norm score normalisation, particularly in the 1-side case, seem to imply that a model produced with the proposed technique exhibits a more consistent response to a variety of test segments from differing session conditions. In contrast, the baseline system improved little with Z-Norm while it is well known that H-Norm — utilising extra handset type labels — is more effective.\(^1\) This difference indicates that the session modelling technique is successfully compensating for session differences such as handset type.

At the same time, the Z-Norm result indicates that there is significant discrep-\(^1\)As H-Norm is known to be more effective than Z-Norm for the baseline system it is relevant to question why H-Norm was not used for this comparison. One of the focuses of this chapter is alleviating the need for labelled corpora for training the normalisation techniques and for this purpose Z-Norm is more suitable since H-Norm requires its normalisation data to be accurately labelled for handset types.
Figure 5.5: DET plot of the 3-side training condition for the baseline system and session variability modelling on Switchboard-II data.
Figure 5.6: Comparison of session variability modelling and blind feature mapping for the 1-side training condition.

Figure 5.6 compares the performance of the presented technique to a feature mapping system trained with data-driven clustering as described in Chapter 4 on equivalent development data (similar results can be achieved with standard feature mapping as described in [96]). Again, it can be seen that the session variation modelling technique has a clear advantage with a 19% improvement at the minimum DCF operating point, and similarly for the EER.

With score normalisation applied, the advantage of the session modelling method increases as Z-Norm is largely ineffective for feature mapping. Following the logic above, this indicates that feature mapping is less effective in compensating for the encountered session effects.
Table 5.2: Minimum DCF and EER of the baseline system and session variability modelling on Mixer data.

<table>
<thead>
<tr>
<th>System</th>
<th>1-Side</th>
<th>3-Side</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Raw Scores</td>
<td>Z-Norm</td>
</tr>
<tr>
<td></td>
<td>Min. DCF EER</td>
<td>Min. DCF EER</td>
</tr>
<tr>
<td>Baseline</td>
<td>.0389 10.6</td>
<td>.0339 9.2</td>
</tr>
<tr>
<td>Session Modelling</td>
<td>.0358 8.7</td>
<td>.0242 6.0</td>
</tr>
<tr>
<td>Baseline</td>
<td>.0183 4.2</td>
<td>.0183 3.8</td>
</tr>
<tr>
<td>Session Modelling</td>
<td>.0119 2.8</td>
<td>.0108 2.2</td>
</tr>
</tbody>
</table>

5.7.2 Mixer Results

The results presented so far indicate that session modelling can produce significant gains in speaker verification performance for the conversational telephony data of Switchboard-II. This section presents results of the same system for the Mixer corpus [70] to demonstrate that this method is not exploiting hidden characteristics of Switchboard. Furthermore, the increased variety of channel conditions present — including a variety of mobile transmission types, hands-free and cordless handsets as well as cross-lingual trials — represents a significantly more challenging situation for the proposed session modelling approach to tackle.

Figures 5.7 and 5.8 and Table 5.2 present results for Mixer data using the QUT 2004 protocol (Section 2.2.1) analogous to the results presented above.

Due to the limited number of speakers in this database the background data was supplemented with Switchboard-II data. The UBM and session transform were trained on a combination of Switchboard-II and Mixer data with approximately equal proportions. In contrast, the background data used for Z-Norm and T-Norm statistics were restricted to Mixer. Results for all three splits are combined in these results.

Overall the advantage gained through session modelling for this data is less than for the Switchboard-II case. Relative improvements over the reference GMM-UBM system are approximately 30% and 36% at the minimum DCF ope-
Figure 5.7: DET plot of the 1-side training condition for the baseline system and session variability modelling on Mixer data.
Figure 5.8: DET plot of the 3-side training condition for the baseline system and session variability modelling on Mixer data.
iating point for the 1-side and 3-side conditions, respectively, and 40% reduction in EER for both conditions when full score normalisation is applied. This performance is still a significant step forward and confirms the usefulness of explicitly modelling session variability.

Interestingly, the session modelling results are actually quite consistent across the different databases, with the absolute error rates and detection costs being very similar across the corpora both with and without score normalisation. It would seem that the reduced relative improvement gained with the session modelling is actually a result of better baseline performance. This is somewhat surprising due to the stated intention of the Mixer project to produce a more challenging dataset with a wider variety of mismatch [70].

The relatively modest improvements experienced in the 3-side training condition for Mixer data (36% minimum DCF improvement compared to 68% for Switchboard-II) combined with the known increase in the variety of channel conditions suggests that the session subspace may be saturated by the observed session variability for this data. Increasing the variation captured in the subspace may lead to further performance gains.

5.7.3 Session Subspace Size

All results so far have assumed a session variability subspace of dimension $R_z = 20$. Presented in Table 5.3 are results obtained by varying the dimension of the session variability subspace for the 1- and 3-side training conditions of the QUT 2004 protocol.

In [116] the importance of severely constraining the dimension of the session variability subspace was noted, citing degrading performance comparing results from the $R_z = 50$ and $R_z = 20$ cases in the 1-side condition with no score normalisation. Further experiments revealed this to not necessarily be the case. As Table 5.3 shows, increasing $R_z$ from 20 to 50 results in worse performance based on the raw output scores but after normalisation is applied the situation has reversed, with $R_z = 50$ giving both superior minimum DCF and EER. The DET curves associated with these systems is depicted in Figure 5.9.
Table 5.3: Minimum DCF and EER results when varying the number of session subspace dimensions, $R_z$.

<table>
<thead>
<tr>
<th>System</th>
<th>Raw Scores</th>
<th>ZT-Norm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min. DCF</td>
<td>EER</td>
</tr>
<tr>
<td>1-Side</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0389</td>
<td>10.6</td>
</tr>
<tr>
<td>$R_z = 10$</td>
<td>.0355</td>
<td>8.8</td>
</tr>
<tr>
<td>$R_z = 20$</td>
<td><strong>.0358</strong></td>
<td><strong>8.7</strong></td>
</tr>
<tr>
<td>$R_z = 50$</td>
<td>.0391</td>
<td>9.4</td>
</tr>
<tr>
<td>3-Side</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0183</td>
<td>4.2</td>
</tr>
<tr>
<td>$R_z = 10$</td>
<td>.0128</td>
<td>3.1</td>
</tr>
<tr>
<td>$R_z = 20$</td>
<td>.0119</td>
<td>2.8</td>
</tr>
<tr>
<td>$R_z = 50$</td>
<td><strong>.0104</strong></td>
<td><strong>2.5</strong></td>
</tr>
</tbody>
</table>

Figure 5.9: DET plot of the 1-side condition when varying the number of session subspace dimensions, $R_z$ with and without ZT-Norm score normalisation.
For the 3-side condition the advantage of increasing the subspace size is clear as improved performance is gained for both measures with or without score normalisation.

The implications of this result are that increasing the power of the system’s ability to model session variability can provide improved performance but score normalisation may be required to realise these benefits. This leads to the conclusion that the session variability modelling method produces inherently less calibrated raw scores than the reference GMM-UBM system with standard top-$N$ ELLR scoring, particularly as $R_z$ is increased.

It is also apparent that it is not always possible to make accurate conclusions about the comparative performance of different configurations after normalisation based on raw system scores alone.

### 5.7.4 Comparison of Training Methods

As noted in Section 5.4.4 there are several possibilities for the algorithm used to simultaneously optimise the set of variables $\{\mathbf{y}(s); \mathbf{z}_h(s), h = 1, \ldots, H(s)\}$ during speaker enrolment. Results comparing several configurations for the female portion of the QUT 2004 protocol are presented in Table 5.4.

The configurable parameters of interest in this experiment are the number of iterations required in training for both the E-M algorithm and the Gauss-Seidel optimisation part of this algorithm. It is advantageous from a processing time perspective to keep both of these to a minimum.

The number of E-M iterations is given in the second column of Table 5.4. As noted in Section 5.4.4 estimating the speaker vector does not converge quickly and is seemingly far from converging even after 20 iterations in the sense of finding a final optimal speaker offset, as shown in Figures 5.1 to 5.3. For comparison purposes it was therefore impractical to wait for full convergence and a maximum of five iterations was selected based on empirical knowledge from standard GMM-UBM systems (designated Baseline in the table above).

Interestingly, dropping back to only one iteration of the E-M procedure gives much better performance than using more iterations across the board for all
Table 5.4: Minimum DCF and EER for variations on the Gauss-Seidel training method and independent estimation of the speaker and session variables for the female subset of QUT 2004 protocol.

<table>
<thead>
<tr>
<th>Systems</th>
<th>E-M Iterations</th>
<th>Raw Scores</th>
<th>ZT-Norm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Min. DCF</td>
<td>EER</td>
</tr>
<tr>
<td><strong>1-Side</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>5</td>
<td>.0380</td>
<td>9.9</td>
</tr>
<tr>
<td>Independent</td>
<td>5</td>
<td>.0404</td>
<td>9.6</td>
</tr>
<tr>
<td>Gauss-Seidel</td>
<td>5</td>
<td>.0389</td>
<td>9.3</td>
</tr>
<tr>
<td>Converged G-S</td>
<td>5</td>
<td>.0389</td>
<td>9.3</td>
</tr>
<tr>
<td>Speaker First G-S</td>
<td>5</td>
<td>.0392</td>
<td>9.4</td>
</tr>
<tr>
<td>Baseline</td>
<td>1</td>
<td>.0319</td>
<td>8.5</td>
</tr>
<tr>
<td>Independent</td>
<td>1</td>
<td>.0221</td>
<td>5.4</td>
</tr>
<tr>
<td>Gauss-Seidel</td>
<td>1</td>
<td><strong>.0219</strong></td>
<td>5.1</td>
</tr>
<tr>
<td>Converged G-S</td>
<td>1</td>
<td><strong>.0219</strong></td>
<td>5.1</td>
</tr>
<tr>
<td><strong>3-Side</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>5</td>
<td>.0165</td>
<td>4.0</td>
</tr>
<tr>
<td>Independent</td>
<td>5</td>
<td>.0112</td>
<td>3.2</td>
</tr>
<tr>
<td>Gauss-Seidel</td>
<td>5</td>
<td>.0091</td>
<td>2.5</td>
</tr>
<tr>
<td>Converged G-S</td>
<td>5</td>
<td>.0092</td>
<td>2.5</td>
</tr>
<tr>
<td>Speaker First G-S</td>
<td>5</td>
<td>.0093</td>
<td>2.6</td>
</tr>
<tr>
<td>Baseline</td>
<td>1</td>
<td>.0169</td>
<td>3.6</td>
</tr>
<tr>
<td>Independent</td>
<td>1</td>
<td>.0058</td>
<td>1.6</td>
</tr>
<tr>
<td>Gauss-Seidel</td>
<td>1</td>
<td><strong>.0054</strong></td>
<td>1.6</td>
</tr>
<tr>
<td>Converged G-S</td>
<td>1</td>
<td><strong>.0054</strong></td>
<td>1.6</td>
</tr>
</tbody>
</table>
session modelling variants; more than 40% reductions in both minimum DCF and EER were observed comparing the best five-iteration system to best one-iteration system based on unnormalised scores for the 1-side training condition. Similar results were observed for the 3-side condition. While single iteration training remained ahead after score normalisation was applied, the margin was significantly reduced.

The one-iteration result is quite interesting for two reasons. Firstly this result reverses the usual trend of improved and more consistent performance from multiple-iteration MAP adaptation seen in standard GMM-UBM systems [88, 117]. Secondly, at least in the case of only a single Gauss-Seidel iteration, the speaker mean supervector is effectively trained on variability that can not be explained in the session subspace as the session variables are estimated before the speaker. This may indicate that it is better to fully optimise the session variables independently of the speaker variable and then determine the speaker parameters on what is effectively the residual variability after removing the channel effects and other forms of session variability.

This single-iteration result also reinforces the hypothesis that the overall performance of a GMM-UBM verification system is more about the differences between the target speaker models and the background model rather than ensuring the target models accurately represent the probability distribution of the target’s speech, as discussed in Section 2.4.3. To investigate this issue it may be interesting to remove some of the biases toward the UBM in the scoring process. For example, how much effect does the top-$N$ scoring procedure have on this analysis if the top components are not determined via the UBM? This issue is beyond the scope of this discussion and is left as a direction for future research.

The impact of the order in which the speaker and session variables are estimated seems to make minimal difference to overall system performance as shown by comparing the results labelled Gauss-Seidel and Speaker First G-S above, which both use only a single iteration of Gauss-Seidel optimisation. Ensuring this optimisation has properly converged (Converged G-S above) also seems irrelevant; there is virtually nothing to separate the fully converged estimate and
5.7. Experiments

Finally, enrolment using independent optimisation of the speaker and session variables results in only a small degradation in performance compared to the Gauss-Seidel methods, as can be seen by observing the results for the systems labelled Independent in Table 5.4. (It should be mentioned that the results for Speaker First G-S with one iteration are intentionally absent as this configuration will produce identical results to the single-iteration Independent system as the estimates of the session variables do not have an opportunity to feed back into the speaker variable estimate.)

Using the results of this section, Figure 5.10 compares the performance of an optimised system using the session modelling techniques of this chapter to the baseline system for the QUT 2004 protocol for both the 1- and 3-side training conditions. With a minimum DCF of .0158 and EER of 4.2% for the 1-side condition, this translates to relative reductions of 47% and 53% compared to the

Figure 5.10: DET plot for the 1- and 3-side training conditions comparing an optimised session modelling system with a baseline GMM-UBM system with score normalisation applied to both.
baseline system. The performance improvements in the 3-side condition are more impressive with 56% and 58% reductions in detection cost and EER respectively with absolute values of .0064 and 1.5%.

Figure 5.11 and Table 5.5 demonstrate the performance of this system for the common evaluation condition of the NIST SRE 2005 protocol (Section 2.2.1). Relative improvements in minimum DCF were achieved for this protocol that are very similar to the QUT 2004 results in both the 1- and 3-side conditions. The reductions in EER were also large although slightly less than for QUT 2004. This system is believed to be the best performing individual system submitted to NIST for evaluation in the 2005 SRE.

\footnote{This claim cannot be substantiated as not all sites reported the results for the individual systems that were combined for final submission, however, few sites produced fused results with comparable or better performance.}
Table 5.5: Comparison of minimum DCF and EER of session modelling and baseline systems for common evaluation condition of the NIST 2005 protocol.

<table>
<thead>
<tr>
<th>Systems</th>
<th>ZT-Norm</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min. DCF</td>
<td>EER</td>
</tr>
<tr>
<td><strong>1-Side</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0352</td>
<td>9.5</td>
</tr>
<tr>
<td>Session Modelling</td>
<td>.0197</td>
<td>6.1</td>
</tr>
<tr>
<td><strong>3-Side</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0267</td>
<td>6.6</td>
</tr>
<tr>
<td>Session Modelling</td>
<td>.0110</td>
<td>3.4</td>
</tr>
</tbody>
</table>

5.7.5 Comparison of Verification Methods

Section 5.5.1 described variations on top-$N$ ELLR verification scoring with potential applications for the session modelling. Compared to standard top-$N$ ELLR scoring, described in (5.40), the first variation in (5.41) attempts to estimate the session conditions of the verification utterance as well as during enrolment by maximising $z$ for the utterance. In (5.42), the speaker vector $y(s)$ is additionally considered a variable with a posterior distribution that must be maximised for the verification utterance as well as the enrolment utterances.

Table 5.6 compares the performance of these variations for the single-iteration session modelling system described above (results for a Baseline system excluding session modelling are also included).

The value of estimating the session conditions is apparent by comparing the results for standard ELLR scoring to additionally maximising for the verification utterance session vector. This result agrees with the stated intention of this scoring method to address mismatch in the verification phase as well as during enrolment. As noted for the enrolment procedure, score normalisation has a greater impact for the more sophisticated scoring method incorporating session modelling as the performance difference between the standard ELLR and maximised session variable increases in every instance with score normalisation applied.

The standard ELLR results demonstrate the improved quality and robustness
Table 5.6: Comparison of minimum DCF and EER for variations on the top-N ELLR verification scoring method for the female subset of the QUT 2004 protocol.

<table>
<thead>
<tr>
<th>Systems</th>
<th>Raw Scores</th>
<th></th>
<th>ZT-Norm</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Raw Scores</td>
<td>ZT-Norm</td>
<td>Raw Scores</td>
<td>ZT-Norm</td>
</tr>
<tr>
<td></td>
<td>Min. DCF</td>
<td>EER</td>
<td>Min. DCF</td>
<td>EER</td>
</tr>
<tr>
<td><strong>1-Side</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0380</td>
<td>9.9</td>
<td>.0266</td>
<td>7.9</td>
</tr>
<tr>
<td>Standard ELLR</td>
<td>.0221</td>
<td>5.4</td>
<td>.0165</td>
<td>4.7</td>
</tr>
<tr>
<td>Maximised Session</td>
<td><strong>.0219</strong></td>
<td><strong>5.1</strong></td>
<td><strong>.0138</strong></td>
<td><strong>4.0</strong></td>
</tr>
<tr>
<td>Max. Session &amp; Speaker</td>
<td>.1000</td>
<td>37.9</td>
<td>.0277</td>
<td>8.4</td>
</tr>
<tr>
<td><strong>3-Side</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0165</td>
<td>4.0</td>
<td>.0114</td>
<td>3.2</td>
</tr>
<tr>
<td>Standard ELLR</td>
<td>.0073</td>
<td>1.8</td>
<td>.0057</td>
<td>1.6</td>
</tr>
<tr>
<td>Maximised Session</td>
<td><strong>.0054</strong></td>
<td><strong>1.6</strong></td>
<td><strong>.0040</strong></td>
<td><strong>1.2</strong></td>
</tr>
<tr>
<td>Max. Session &amp; Speaker</td>
<td>.0992</td>
<td>33.2</td>
<td>.0126</td>
<td>3.8</td>
</tr>
</tbody>
</table>

of the speaker models produced with the session modelling approach to enrolment as the enrolment process is the only difference to the baseline system. The effect of the improved enrolment process is particularly evident for the 3-side case where a 50% reduction in both measures is observed using identical scorings methods.

Adding the extra complexity of refining the speaker model estimate during enrolment has a dramatic effect on performance, particularly prior to score normalisation; in the 1-side case without score normalisation the system is in fact adding no more value than a system that simply rejects all trials, according to the NIST detection cost function. While normalisation improves the performance of this system significantly, it still lags the performance of the baseline system.

Based on the extremely poor performance of the raw system scores and the dramatic improvement provided by score normalisation, it would seem that further refining the speaker model has caused additional issues with the calibration of the raw scores across different models and different verification utterances. This approach may deserve further investigation but the issue of badly calibrated raw scores will need to be addressed for any advantage to be gained over the simpler method of assuming the speaker model is known; particularly, the role of
Table 5.7: Minimum DCF and EER results with varying degrees of convergence in the session variability subspace training.

<table>
<thead>
<tr>
<th>System</th>
<th>Raw Scores</th>
<th>ZT-Norm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min. DCF</td>
<td>EER</td>
</tr>
<tr>
<td>1-Side</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0389</td>
<td>10.6</td>
</tr>
<tr>
<td>Switchboard-II</td>
<td>.0257</td>
<td>6.7</td>
</tr>
<tr>
<td>1 iteration</td>
<td>.0247</td>
<td>6.1</td>
</tr>
<tr>
<td>2 iterations</td>
<td>.0238</td>
<td>5.7</td>
</tr>
<tr>
<td>5 iterations</td>
<td>.0226</td>
<td>5.3</td>
</tr>
<tr>
<td>10 iterations</td>
<td>.0219</td>
<td>5.1</td>
</tr>
<tr>
<td>20 iterations</td>
<td>.0213</td>
<td>5.1</td>
</tr>
<tr>
<td>3-Side</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Baseline</td>
<td>.0183</td>
<td>4.2</td>
</tr>
<tr>
<td>Switchboard-II</td>
<td>.0089</td>
<td>2.3</td>
</tr>
<tr>
<td>1 iteration</td>
<td>.0076</td>
<td>2.0</td>
</tr>
<tr>
<td>2 iterations</td>
<td>.0071</td>
<td>1.9</td>
</tr>
<tr>
<td>5 iterations</td>
<td>.0059</td>
<td>1.6</td>
</tr>
<tr>
<td>10 iterations</td>
<td>.0054</td>
<td>1.6</td>
</tr>
<tr>
<td>20 iterations</td>
<td>.0054</td>
<td>1.6</td>
</tr>
</tbody>
</table>

the UBM will need to be addressed.

5.7.6 Sensitivity to the Session Variability Subspace

Two aspects of performance sensitivity to the training of the session variability subspace transform $U$ are of practical interest. Firstly, the impact of the number of E-M iterations will be investigated as the E-M training algorithm is very computationally expensive and also appears to converge quite slowly. Also, the issue of database mismatch is an important consideration as the training database for an application does not typically match the situation it is applied to. The results of these experiments are summarised in Table 5.7.

Contrary to the conclusions drawn in [57], the proposed method gains signif-
significantly from allowing the E-M algorithm for training the subspace to converge, especially in the 1-side training condition. Furthermore, there does appear to be considerable sensitivity to the nature of the data used to train the subspace transform as the results using the transform trained solely on Switchboard-II data demonstrated degraded performance compared to using Mixer data (comparing the system labelled \textit{Switchboard-II} in Table 5.7 to the other systems). Using Switchboard data still performs favourably compared to the reference system with no session variability modelling, again demonstrating the utility of the method.

The results in Table 5.7 also demonstrate diminishing returns with more than 10 iterations of the E-M algorithm.

### 5.7.7 Reduced Test Utterance Length

An important part of the session modelling method is estimating the session vector $z$ for the test utterance. While this is a low-dimensional variable, estimating it accurately will require a sufficient quantity of speech. This experiment aims to determine the minimum requirements for extracting improved results from session modelling.

Figure 5.12 shows the impact of reducing the test utterance length for both the session variability modelling method and standard GMM-UBM modelling with test utterance lengths of 5, 10 and 20 seconds of active speech.

These results indicate that approximately 10 seconds of speech are required to estimate the session factors sufficiently accurately to produce improved results over standard modelling and scoring practice, while 20-second trials produce advances in performance approaching those experienced with full-length testing utterances, with relative improvements of over 20\% in both minimum DCF and EER. In fact the 20-second session modelling results out perform the baseline system using full verification utterances with an average of more than 100 seconds of active speech.
Figure 5.12: DET plot for the 1-side training condition comparing baseline and session modelling results for short test utterance lengths.
Chapter 5. Explicit Modelling of Session Variability

5.8 Discussion

One of the major advantages of the approach presented in this chapter is the more relaxed requirements for training corpus labelling. This technique removes the necessity of labelling databases for channel, handset type and other forms of session variability, which is often difficult, error prone and expensive if not impossible.

The benefits gained with score normalisation, particularly Z-Norm, seem to imply that a model produced with the proposed technique exhibits much more uniform response to a variety of test segments from different conditions. In contrast, the baseline system improved little with Z-Norm while it is well known that H-Norm — which models score distributions separately for different handset types — is more effective. This difference apparently indicates that the session modelling techniques are indeed successfully compensating for session differences such as handset type.

More sophisticated verification techniques are possible with the session variability modelling approach. Future research will investigate the effectiveness of Bayes factor techniques in conjunction with modelling session variability in a similar approach to [118]. Under this approach the speaker model parameters are not assumed to be known at testing time, but rather to have posterior distributions that have been refined by the training procedure.

The results presented in this chapter indicate that the multiple training side conditions of the official NIST protocols do not show as significant an improvement as the results for the QUT versions of the protocols on the same data. The main reason for this is the additional constraints NIST chose to put on the selection of utterances for enrolment and verification purposes. Specifically, as stated in Section 4.3 of the 2005 evaluation specification [79], all verification trials were different-number trials meaning that the enrolment utterance and verification utterance were recorded on different telephone lines; this is not necessarily true for the QUT protocols where this constraint was not enforced. Discussion at the follow-up workshop for the 2005 SRE additionally revealed that NIST inter-
interpreted this constraint to imply that all sessions used to enrol a 3-side or 8-side model were recorded from the same number if possible [113]. Again this was not a constraint applied in developing the QUT protocols. The net result is that the multiple-side training conditions of the NIST protocols exhibit significantly less variety in the available training data than the equivalent QUT protocol and potentially more mismatch between enrolment and verification conditions as there are likely to be same number trials in the latter.

The question to come out of this observation is whether one protocol is more appropriate or representative than another. Of course the answer to this question depends on the eventual application of the technology, but what type of application always has the situation prescribed in the NIST protocol? It seems that this protocol actually discourages techniques that attempt to exploit the mismatch in conditions in the training sessions to better understand the mismatch when transferring to verification trials.

5.9 Summary

In this chapter a technique was proposed to compensate for mismatch experienced in text-independent speaker verification due to session-to-session variability. Explicit modelling of the prevalent conditions in training and verification sessions was introduced by adding a session-dependent variable to the speaker modelling process that was constrained to lie in a session variation subspace.

Techniques were developed to incorporate this augmented model into both the speaker enrolment and verification phases of a verification system.

The enrolment process involved a sophisticated simultaneous optimisation of the speaker mean vector and additional session vectors for each session available for enrolment via an expectation-maximisation algorithm. Maximum a posteriori criteria were used in all cases; a classical relevance MAP approach for the speaker mean and MAP with a standard normal prior in the session variability subspace for each of the session variables.

Due to the model complexity, a direct solution to the maximisation step of
the E-M algorithm was shown to be very computationally expensive to the point of being impractical for large verification trials, such as a NIST evaluation. To avoid this issue, an iterative approximation method was proposed based on the Gauss-Seidel method for solving linear systems.

Of the verification techniques discussed, the most successful was derived from the top-$N$ expected log-likelihood ratio scoring used in standard GMM-UBM speaker verification systems. It was empirically shown that estimating the session conditions of the verification utterance was beneficial and successfully addressed mismatch from both the enrolment and verification perspectives.

Methods for training the session variability subspace based on a database of background speakers were also described. The sensitivity of verification performance to insufficient convergence of this training was empirically investigated, as was the issue of mismatched conditions between training and testing databases.

Experiments on conversational telephony data demonstrated the effectiveness of the technique for both single and multiple training session conditions with up to 68% reduction in detection cost over a standard GMM-UBM system and significant improvements over a system utilising feature mapping. It was also observed that the session variability modelling responds particularly well to score normalisation with the Z-Norm and T-Nrom approaches.
Chapter 6

Modelling Uncertainty in Verification Scores

6.1 Introduction

Deploying a speaker verification system is a difficult task for several reasons. Typically these difficulties involve determining system parameters such as the required amount of speech for sufficiently accurate enrolment and for sufficiently accurate verification trials. Additionally, there is always the problem of estimating a threshold for acceptance and rejection and the eventual error rate to expect from such a threshold. This is particularly difficult in the presence of a significant mismatch between the development database and the anticipated deployment conditions. This set of related issues is also notably absent from published research in speaker recognition.

This chapter presents an initial attempt to address some of the problems raised above, particularly concerning the quantity of speech required for sufficiently accurate verification results, by employing confidence measures on the verification score. This also leads to some implications for determining and applying thresholds to the verification problem.

The following section presents the concept of confidence measures for the speaker verification score. Several potential uses for confidence measures in verification systems are also discussed and the implications of these uses on the
nature of the confidence measure.

Section 6.3 presents some relevant background material to the development of the confidence-based verification methods. Specifically this includes the assumptions made on the nature of the score produced by current speaker verification systems and the effect on this score of reducing the available data with which to make a decision.

To account for the specific issues encountered in speaker verification several methods of estimating the verification score variance are then developed in Section 6.4. This variance statistic provides the fundamental tools required to estimate the confidence of a verification decision. Experimental evaluation of these estimates are also presented for the application of making confident verification decisions with as little data as possible.

The chapter is concluded with some possible directions for future developing this very promising approach.

6.2 Confidence Measures

Ideally, we would like to produce a verification confidence from a trial, as this is the most useful and usable result from a system designer perspective: Knowing that there is a 96% probability that an utterance was produced by speaker s makes it easy for a designer to employ Bayesian logic to produce the best possible system. This, however, is not practical. Firstly, to do this requires accurately estimating the prior probability of a true trial; this is impossible under most circumstances considering that the non-target class potentially includes the entire population of the world. In a forensic situation, deductive logic and other evidence may help in this regard. Secondly, assuming accurate priors are available, producing a verification confidence also requires that verification scores produced by a system are in fact accurate likelihoods (or likelihood ratios). This is very rarely the case, considering that rudimentary statistical models are usually used to represent speakers and the inherent difficulties in representing every other speaker in a non-target model. Add to this that score normalisation is usually applied and
the resulting scores often have little resemblance of true likelihoods.

Some work has investigated the issue of producing accurate likelihood ratio scores as this is the explicit goal of forensic applications of speaker recognition \[101, 39\]. The analysis and evaluation of speaker verification systems based on the accuracy of output likelihood ratios is an emerging topic of recent interest \[15, 16\], but speaker verification systems do not in general produce scores that should be interpreted as likelihood ratios. Specifically, a system can be calibrated to approximate likelihood ratios for a particular database but this may not correspond to accurate likelihoods after moving to the environment in which the system is to be deployed.

Given these difficulties with determining an accurate verification confidence, an alternative approach is to determine a method by which one can state that the verification score for this trial lies within the interval $\Lambda_S = a \pm b$ at the 99% confidence level.

While not ideal, this information is still very useful for the deployment and application of a speaker verification system. It provides the capability to:

1. Estimate upper and lower bounds on probability of errors for a verification trial at a particular confidence level based on a development database.
2. Estimate the level of confidence at which the verification score is above or below a particular threshold.
3. Shortcut a verification trial when we are confident that the “true” verification score lies within a particular interval of the current estimate.
4. Shortcut a verification trial when we are confident that the “true” verification score is above or below particular thresholds.

Here the “true” verification score is defined as the score that the verification system would produce given an infinite quantity of testing speech.

Assuming a verification score is a random variable drawn from a Gaussian distribution with a mean of the “true” verification score and known variance, it
is straightforward to formulate \(1 \) and \(2 \) above. The main difficulty arises because the variance is unknown and must be estimated. The variance of a trial score distribution is usually dependent on many factors including whether a trial is a genuine or impostor trial (which we obviously do not know \textit{a priori}), the length of a particular verification utterance and the noise levels and other environmental conditions of the recording. These factors lead to the conclusion that the variance must be estimated for each trial \textit{individually}. This estimation forms the basis of the techniques presented and will be addressed in Section \(6.4\).

In the case of \(3 \) and \(4 \) above, further assumptions are made on the form of the verification score. Specifically, it is assumed that the verification score is a random \textit{process} that evolves over time. According to the structure of the speaker verification system in use and the assumptions it is built on, it was assumed that this random process was Gaussian at time \( t \), had a fixed mean ("true" score) and a time-dependent standard deviation, that is

\[
\Lambda_S(t) \sim \mathcal{N}(\mu_S, \sigma_S(t))
\]

Making these assumptions, \(3 \) and \(4 \) above can be treated as essentially incremental versions of \(1 \) and \(2 \) where at time \( t \) the value of \( \sigma_S(t) \) must be estimated and stopping criteria assessed. This again will be addressed in Section \(6.4\).

All of the capabilities outlined above have useful applications. Items \(1 \) and \(2 \) are particularly applicable to forensic tasks where the goal is to evaluate the strength of the available evidence. Items \(3 \) and \(4 \) are more applicable to verification for access purposes, for example user authentication for telephone transactions. Item \(4 \) will be the primary focus of this study as it provides the ability to require the least amount of speech to perform a successful verification trial and consequently the least inconvenience to the end user of the technology.
6.3 Background

6.3.1 Baseline Acoustic Speaker Verification System

The verification system used in this study is described in Section 2.5. The verification score used for this system is the expected log-likelihood ratio of the target speaker to the UBM. The expectation is taken over the individual frame-based log-likelihood ratios for the test utterance,

\[ \Lambda_s = \frac{1}{T} \sum_{t=1}^{T} \ell_s(t) = \frac{1}{T} \sum_{t=1}^{T} \log \left( \frac{p(x_t|\lambda_s)}{p(x_t|\lambda_{ubm})} \right) \]  

(6.2)

where, in the case of Gaussian mixtures,

\[ p(x|\lambda) = \sum_{c=1}^{C} w_c g(x|\mu_c, \Sigma_c) \]  

(6.3)

where \( w_c \) is the mixing factor and \( g(x|\mu_c, \Sigma_c) \) denotes the multivariate Gaussian density with mean \( \mu_c \) and variance matrix \( \Sigma_c \) for mixture component \( c \).

6.3.2 The Effect of Short Verification Utterances

From a researcher’s perspective it is preferable to have as much speech as possible available for each verification to make the most accurate decision. This is the exact opposite of a system designer’s preference to put the least possible demand on the end user. Compromise is necessary. To this end, it is important to have an understanding of the impact of limiting the verification utterance length. The impact of restricted utterances for a typical GMM-UBM system is presented in Table 6.1 and Figure 6.1.

These results demonstrate that utterance length, predictably, has a significant effect on overall system performance in the 10-second and shorter range, which is typically of interest for a system designer, as previously observed [71]. One positive is that performance apparently degrades gracefully at least down to 2-second length without catastrophic error rates.

It is also evident from the DET plot that the performance degrades consistently across a wide range of operating points. This can be viewed as both an
Table 6.1: The effect of shortened utterances on speaker verification performance.

<table>
<thead>
<tr>
<th>System</th>
<th>EER</th>
<th>Min. DCF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>13.5%</td>
<td>.0413</td>
</tr>
<tr>
<td>2 sec</td>
<td>20.4%</td>
<td>.0656</td>
</tr>
<tr>
<td>5 sec</td>
<td>17.1%</td>
<td>.0543</td>
</tr>
<tr>
<td>10 sec</td>
<td>15.5%</td>
<td>.0490</td>
</tr>
<tr>
<td>20 sec</td>
<td>14.5%</td>
<td>.0454</td>
</tr>
</tbody>
</table>

Figure 6.1: DET plot of the effects of shortened utterances on speaker verification performance.
advantage and an inefficiency. Consistency and predictability are useful properties for increasing the flexibility of a system, allowing it to be used in a variety of situations. However, for a particular application where the scenario is well defined and the desired operating point is known, only the performance at that point is relevant; devoting resources to improving performance at other operating points is wasteful. This work presents a method to trade the performance in these regions for decreased verification utterance lengths.

### 6.4 Early Verification Decision Scoring Method

As described in Section 6.2 above, the most immediate and appealing use of the confidence based verification methods is to provide a verification decision with minimal speech. This is achieved by making a verification decision as soon as we are confident the “true” verification score is above or below the specified threshold based on the confidence interval of the current estimate of the “true” score.

An example of this process is presented in Figure 6.2. In this figure, the samples used to estimate the distribution are represented as dots, the mean evolving verification score estimate is shown as a thick red line with the 99% confidence interval of this estimate depicted with dashed lines above and below the estimate. The verification threshold of $-0.1$ is shown as a horizontal line through the centre of the figure. After two seconds of the trial (Figure 6.2(a)) the estimate of the verification score is quite erratic, which is reflected in the wide confidence interval, but looks to be converging to a point below the threshold. By four seconds (Figure 6.2(b)) the estimate seems to be more stable as more samples become available and the width of the confidence interval has narrowed to be entirely below the threshold. At this point, after only four seconds, we can be confident that the verification score will continue to lie below the threshold and thus make a reject decision for this trial. Figure 6.2(c) confirms that the verification score does in fact continue to be below the threshold and the confidence interval continues to narrow.

The crux of confidence-based methods for verification is the ability to estimate
Chapter 6. Modelling Uncertainty in Verification Scores

Figure 6.2: Example verification trial using the early decision method.
6.4. Early Verification Decision Scoring Method

confidence intervals based on the observed sequence of frame scores. This ability in turn relies on estimating the variance of the mean estimate distribution from the sequence of frame scores, which is the focus of this section. Presented below are a number of techniques for calculating this estimate with an increasing degree of sophistication to combat issues encountered with real data. These methods are experimentally evaluated for their effectiveness.

6.4.1 Experimental Setup

Experiments were conducted on the QUT 2004 protocol (Section 2.2.1) using conversational telephony speech drawn from the Mixer corpus. The focus of these results is primarily on the 1-side training condition of this corpus.

Several details of the experimental setup were influenced by implementation details of the methods presented, as described below.

The shortcutting or early-decision criteria were checked after scoring each block of 100 consecutive observation frames, roughly corresponding to 1 second of active speech. This is due to two constraints. Testing the stopping criteria does take some additional processing, so it was deemed unnecessarily wasteful to perform these checks after every frame as testing the criterion more often will not have a significant impact on the final results. Additionally, for the decorrelated method for estimating the frame score variance, multiple frames are required before an update to the variance estimate is possible (Section 6.4.3). This number of frames is controlled by the configuration variable $N$ in (6.6). Since the largest value of $N$ investigated was 100, this was a natural choice for the granularity of testing the stopping criteria to remain consistent across all experiments.

A minimum verification length of 2 sec, or more accurately 200 frames, was also imposed. This limit is also a result of the maximum $N$-value of 100 used in these experiments. This is due to the theoretical limit of variance estimates requiring at least 2 samples. Again this was retained for all systems for consistency.

The 2 sec minimum trial limit does unfortunately have a noticeable effect on some results. Specifically, the median length of verification trials is regularly
pushing this lower limit.

### 6.4.2 Naïve Variance Estimate

As can be seen from (6.2), the verification score is a sum of the log-likelihood ratios of individual frames. The central limit theorem states that a sum of random variables (such as this) will exhibit a Gaussian distribution. Furthermore it is a commonly stated assumption that the feature vectors \( \mathbf{x}_t \) and, by consequence, the frame log-likelihood ratios \( \ell_S(t) \) are independent and identically distributed (iid) random variables. Thus, if \( \ell_S(t) \) has sample mean \( m_\ell \) and variance \( s_\ell^2 \), the ELLR verification score will have a mean and variance approximated by

\[
\mu_S = m_\ell \\
\sigma^2_S = \frac{s_\ell^2}{T - 1}
\]

Thus, for any sequence of frames \( \mathbf{X} \) it is possible estimate the mean and variance of the ELLR score.

Using these estimates of the ELLR score statistics, a confidence interval for the “true” score can be calculated using a confidence level and the Gaussian cumulative density function (CDF).

### Results

Figure 6.3 shows the performance of a system employing early decision scoring using the naïve frame-based estimate in (6.5) with the threshold set for the equal error rate operating point at three confidence levels, 90%, 99% and 99.9%. These confidence levels are the minimum confidence that the “true” verification score is above or below the EER threshold required by the system to make an early verification decision. Also shown is the DET curve for the baseline reference system using all available speech and a system using a fixed 2-second utterance length (dotted curve) as a “worst case” system, given the minimum length constraints discussed in the previous section.

As can be seen in Figure 6.3, there is a significant drop in performance compared to the reference system due to the shortcut stopping criterion however there
are some interesting aspects to this plot. First, the degradation in performance is actually quite modest as the reference system used at least 10 times the amount of speech to make a verification decision, as described in Table 6.2. This point will be addressed further below.

Second, the performance of the system improves using higher confidence levels providing a better EER; this observation is backed by Table 6.2 with the Na"ive 99.9% system showing an EER 2.6% lower than at the 90% confidence level.

Third, and more interestingly, the DET curve for these systems veers away from the reference system the farther it is from the EER operating point, this is particularly evident in the low false alarm region. The performance curves of the early decision systems drop back to meet the 2-second worst-case system in these areas. This characteristic is a direct consequence of the shortcut method as the system is only interested in the performance at the specified threshold and essentially trades performance in other areas for shorter test utterances. In the ideal case the system would only provide performance at the threshold and trade
Table 6.2: Verification results using the naïve method at the EER operating point.

<table>
<thead>
<tr>
<th>System</th>
<th>EER</th>
<th>Trial Length</th>
<th>Shortcut Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EER</td>
<td>Median</td>
<td>Mean</td>
</tr>
<tr>
<td>Reference</td>
<td>13.5%</td>
<td>110.2</td>
<td>109.6</td>
</tr>
<tr>
<td>Naïve at 90%</td>
<td>17.5%</td>
<td>2</td>
<td>2.8</td>
</tr>
<tr>
<td>Naïve at 99%</td>
<td>15.4%</td>
<td>2</td>
<td>5.9</td>
</tr>
<tr>
<td>Naïve at 99.9%</td>
<td>14.9%</td>
<td>3</td>
<td>10.0</td>
</tr>
</tbody>
</table>

By comparing the Tables 6.1 and 6.2 it can be seen that the shortcut method is effective in trading performance at a specific operating point for shorter trials. Table 6.2 shows that the confidence levels presented roughly correspond in terms of mean trial length to the short utterances in Table 6.1 (page 170) but demonstrate considerably less degradation in EER compared to the reference system. Comparing the 5 sec results to the 99% confidence results the EER improves from 17.1% to 15.4%, almost halving the gap to the reference, with similar average test utterance lengths.

Additionally, the mean test utterance lengths are dominated by a relatively small number of long trials with the majority of trials providing a result within 2 seconds, as indicated by the median trial lengths.

This last point has an astonishing implication: For most trials a text-independent speaker verification system will produce the same decision with only 2 seconds of speech that it will with 2 minutes of speech.

A better understanding of the distribution of trials lengths can be taken from the histogram in Figure 6.5.

Also presented in the two rightmost columns of Table 6.2 are the rates of errors introduced by the early stopping criteria for target and impostor trials, respectively. These represent the trials that are accepted as above the threshold.
6.4. Early Verification Decision Scoring Method

Figure 6.4: DET plot of the ideal early verification decision scoring system.

Figure 6.5: Histogram of the test utterance length using the naïve variance estimate method with the EER operating point.
according to the stopping criteria but produce a negative result according to the reference system using all available speech, and vice-versa. This is the loss introduced by the shortcut methods and, if the distribution assumptions and estimates are accurate, should closely match the confidence levels specified.

Two points can be made based on these numbers; the error rates do not match the specified confidence levels well and, also, there are marginally more errors for impostor trials than target trials.

The fact that the error rates don’t reflect the desired confidence levels suggests two possible issues. Firstly, that the ELLR variance estimates are not sufficiently accurate particularly when based on a small number of frames and are thus causing the scoring process to be terminated prematurely. This issue will be addressed in the following sections with the use of better estimates and estimation methods.

The second possible issue is that the actual distribution of the frame scores, and by extension of the ELLR scores, does not fit well with the assumed Gaussian shape on which the confidence level thresholds are based. Observations of frame score distributions show that this is in fact a valid assertion as they exhibit significant third and fourth order statistics, however, the law of large numbers states that the ELLR score will tend towards normality. The issue, then, is that for very short utterances there is not a “large number” of frame scores. Since the degree to which the shortcut performance approaches the reference system is typically more important than the accuracy of the confidence levels, this issue will not constitute a large portion of this discussion.

Operating at the minimum detection cost threshold

As this is a threshold-based algorithm, it can in theory be used at any operating point as required by the application. Figure 6.6 and Table 6.3 describe the performance of the shortcut method at the NIST minimum detection cost function operating point. As can be seen, many of the characteristics of this performance closely resemble the performance at the EER operating point, specifically the DET curve produces the best performance at the desired minimum DCF operat-
6.4. Early Verification Decision Scoring Method

Figure 6.6: DET plot using the naïve method with the minimum DCF operating point.

ing point and drops away in all other operating regions, and the higher confidence levels produce results closer to the reference system.

Unlike with an EER threshold, the errors introduced by the early decision method are not evenly distributed between the target and impostor trials, with the target trial errors far outweighing the low rate of impostor trial errors. From this observation it is hypothesised that this discrepancy is due to the threshold lying much closer to the centre of the target trial score distribution (at approximately 35% miss rate) compared to near the tail of the impostor scores distribution (approximately 1% false alarms). Hence it is simpler to dismiss a larger proportion of the impostor trials due to the increased distance of the score to the threshold.

It is also evident from Table 6.3 that even less speech was required to produce the minimum DCF results than for the EER threshold case, as a median trial length of 2 sec is used for all confidence levels, and the mean length only reaches
Table 6.3: Verification results using the naïve method with the minimum DCF operating point.

<table>
<thead>
<tr>
<th>System</th>
<th>Min. DCF</th>
<th>Trial Length</th>
<th>Shortcut Errors</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>.0413</td>
<td>110.2</td>
<td>109.6</td>
<td>–</td>
</tr>
<tr>
<td>Naïve at 90%</td>
<td>.0608</td>
<td>2</td>
<td>2.2</td>
<td>14.9% 2.0%</td>
</tr>
<tr>
<td>Naïve at 99%</td>
<td>.0519</td>
<td>2</td>
<td>2.8</td>
<td>11.5% 1.0%</td>
</tr>
<tr>
<td>Naïve at 99.9%</td>
<td>.0472</td>
<td>2</td>
<td>3.6</td>
<td>8.3% 0.6%</td>
</tr>
</tbody>
</table>

3.6 sec.

6.4.3 Estimate with Correlation

Unfortunately, acoustic features commonly used for speaker verification, such as MFCC features, exhibit high levels of correlation between consecutive observation frames. This is essentially by definition, considering that the short-term spectra and cepstra typically calculated for consecutive frames share two-thirds of their waveform samples and that delta cepstra explicitly average over a number of frames. This is also due to the characteristics of the mechanics of speech production as there are limits on the rate at which vocal tract shape can change, this is a fact exploited by techniques such as RASTA filtering [43]. This correlation obviously voids the commonly cited assumption of statistically iid feature vectors.

Due to this invalidity of the iid assumption, the estimated ELLR variance is invalid and empirical evidence shows that it is often underestimated, particularly with short sequences. For this reason, it is necessary to develop an alternative estimate to reduce the effect of this correlation.

In this research a transformation approach was adopted to reduce the correlation by producing a series of ELLR estimates $y_S$ from short, fixed-length, non-overlapping frame sequences,

$$y_S(i) = \frac{1}{N} \sum_{t=N_i}^{N(t+1)-1} \ell_S(t)$$

(6.6)
where \( N \) is the length of the short frame sequences. If \( N \) is sufficiently large, the correlation between successive \( y_S(i) \) drops to a negligible level.

From \( y_S \), it is then possible to estimate the overall ELLR mean and variance as

\[
\mu_S = m_y \\
\sigma^2_S = \frac{s^2_y}{T/N - 1}
\]

where \( m_y \) and \( s^2_y \) are the sample mean and sample variance of \( y_S \) respectively.

**Results**

Table 6.4 presents the performance of the shortcut scoring method using the decorrelated distribution estimates from (6.7) and (6.8). A range of short frame sequence length values, \( N \), are assessed with the longer sequences reducing the degree of correlation in the samples used to estimate the ELLR score distribution. A value of \( N = 1 \) is equivalent to the naïve, frame-based estimate described previously. With a typical frame rate of 100 frames per second, a value of \( N = 100 \) averages the frame scores over the period of a whole second of active speech.

It can be seen from these results that decorrelating the samples used to estimate the ELLR score distribution does in fact reduce the proportion of errors introduced by the shortcut scoring method (the two rightmost columns of Table 6.4), producing performance closer to that of the reference system. The best performing configuration in Table 6.4 drops only 0.4% at the EER operating point.

It is also apparent that the choice of short sequence length \( N \) is a trade-off between conflicting concerns, as also demonstrated by Figure 6.7. If the sequences are too short the system will not benefit from the decorrelating effect. If the sequences are too long, such as the case with \( N = 100 \), the samples are decorrelated however the number of samples with which to estimate the ELLR variance are severely limited. For example after 2 sec of active speech there will only be two samples from which to estimate the variance in the \( N = 100 \) case, this will clearly not be a reliable estimate. A value of \( N = 10 \) seems to provide a good balance.
Table 6.4: Verification results comparing the naïve and decorrelated methods at the EER operating point.

<table>
<thead>
<tr>
<th>System</th>
<th>EER</th>
<th>Trial Length</th>
<th>Shortcut Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Median</td>
<td>Mean</td>
</tr>
<tr>
<td>Reference</td>
<td>13.5%</td>
<td>110.2</td>
<td>109.6</td>
</tr>
<tr>
<td><strong>90% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N = 1 (Naïve)</td>
<td>17.5%</td>
<td>2</td>
<td>2.8</td>
</tr>
<tr>
<td>N = 10</td>
<td>15.9%</td>
<td>2</td>
<td>5.1</td>
</tr>
<tr>
<td>N = 100</td>
<td>16.4%</td>
<td>2</td>
<td>4.9</td>
</tr>
<tr>
<td><strong>99% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N = 1 (Naïve)</td>
<td>15.4%</td>
<td>2</td>
<td>5.9</td>
</tr>
<tr>
<td>N = 10</td>
<td>14.4%</td>
<td>3</td>
<td>13.9</td>
</tr>
<tr>
<td>N = 100</td>
<td>15.2%</td>
<td>4</td>
<td>14.7</td>
</tr>
<tr>
<td><strong>99.9% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N = 1 (Naïve)</td>
<td>14.9%</td>
<td>3</td>
<td>10.0</td>
</tr>
<tr>
<td>N = 10</td>
<td>13.9%</td>
<td>5</td>
<td>21.5</td>
</tr>
<tr>
<td>N = 100</td>
<td>14.8%</td>
<td>7</td>
<td>22.8</td>
</tr>
</tbody>
</table>
6.4. Early Verification Decision Scoring Method

![Figure 6.7: DET plot comparing the na"ıve and decorrelated methods at the EER operating point using a 99\% confidence level.](image)

and demonstrates clearly superior performance to the other configurations tested.

There is unfortunately an increase in both the mean and median utterance length associated with the decorrelated estimation method, however, despite this increase the median utterance lengths required are still extremely short at around 2–5 seconds.

These outcomes are also relevant at the minimum DCF operating point, with the best configuration, with $N = 10$, giving away only .0020 to the reference system. Table [6.5] summarises the results for the minimum DCF threshold with $N = 10$. These results also suggest that at this operating point the median utterance length is still extremely low, staying at the 2 sec minimum throughout.
Table 6.5: Verification results comparing the naïve and decorrelated methods at the minimum DCF operating point.

<table>
<thead>
<tr>
<th>System</th>
<th>Min. DCF</th>
<th>Trial Length</th>
<th>Shortcut Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>.0413</td>
<td>Median 110.2</td>
<td>Target –</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Mean 109.6</td>
<td>Impostor –</td>
</tr>
<tr>
<td>$N = 10$ at 90%</td>
<td>.0537</td>
<td>2</td>
<td>12.2% 1.2%</td>
</tr>
<tr>
<td>$N = 10$ at 99%</td>
<td>.0454</td>
<td>2</td>
<td>5.6% 0.3%</td>
</tr>
<tr>
<td>$N = 10$ at 99.9%</td>
<td>.0433</td>
<td>2</td>
<td>3.1% 0.1%</td>
</tr>
</tbody>
</table>

6.4.4 Robustly Estimating the Sample Variance

For these techniques to be effective, it is important to robustly estimate the variance of the frame log-likelihood ratios with a very limited number of samples. This issue is also exacerbated by the correlated nature of these scores. One possible method to produce a more robust estimate of this variance is to introduce a priori information, with the resulting estimate given by

$$s^2 = \frac{\tau \kappa^2 + (M - 1)s^2}{\tau + (M - 1)},$$  \hspace{1cm} (6.9)

where $s^2$ is unbiased sample variance from $M$ samples and $\kappa^2$ and $\tau$ are hyper-parameters of the prior distribution, which takes the form of a Dirichlet distribution \[38\].

This estimate can then be used to produce more robust estimates of the ELLR variance, as estimated in (6.5) and (6.8) above.

Results

By incorporating a priori information in the variance estimate it is possible to reduce the performance discrepancy between the reference system and the early decision version. This improved performance unfortunately comes at the cost of longer verification utterances both in terms of the mean and median length statistics, as presented in Tables 6.6 and 6.7 above.

It is interesting to note the effect of varying both the prior hyperparameter $\kappa^2$ and the target confidence level as both can be tuned to produce similar levels
Table 6.6: Verification results incorporating *a priori* information in the variance estimate at the EER operating point.

<table>
<thead>
<tr>
<th>System</th>
<th>EER</th>
<th>Trial Length</th>
<th>Shortcut Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Median</td>
<td>Mean</td>
</tr>
<tr>
<td>Reference</td>
<td>13.5%</td>
<td>110.2</td>
<td>109.6</td>
</tr>
<tr>
<td><strong>90% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No prior</td>
<td>15.9%</td>
<td>2</td>
<td>5.1</td>
</tr>
<tr>
<td>$\kappa^2 = 1$</td>
<td>15.1%</td>
<td>2</td>
<td>6.3</td>
</tr>
<tr>
<td>$\kappa^2 = 2$</td>
<td>14.8%</td>
<td>3</td>
<td>7.8</td>
</tr>
<tr>
<td>$\kappa^2 = 4$</td>
<td>14.5%</td>
<td>4</td>
<td>10.0</td>
</tr>
<tr>
<td><strong>99% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No prior</td>
<td>14.4%</td>
<td>3</td>
<td>13.9</td>
</tr>
<tr>
<td>$\kappa^2 = 1$</td>
<td>13.9%</td>
<td>6</td>
<td>17.1</td>
</tr>
<tr>
<td>$\kappa^2 = 2$</td>
<td>13.8%</td>
<td>7</td>
<td>19.2</td>
</tr>
<tr>
<td>$\kappa^2 = 4$</td>
<td>13.8%</td>
<td>9</td>
<td>22.2</td>
</tr>
<tr>
<td><strong>99.9% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No prior</td>
<td>13.9%</td>
<td>5</td>
<td>21.5</td>
</tr>
<tr>
<td>$\kappa^2 = 1$</td>
<td>13.7%</td>
<td>9</td>
<td>25.2</td>
</tr>
<tr>
<td>$\kappa^2 = 2$</td>
<td>13.7%</td>
<td>11</td>
<td>27.4</td>
</tr>
<tr>
<td>$\kappa^2 = 4$</td>
<td>13.6%</td>
<td>14</td>
<td>30.4</td>
</tr>
</tbody>
</table>
Table 6.7: Verification results incorporating *a priori* information in the variance estimate at the minimum DCF operating point.

<table>
<thead>
<tr>
<th>System</th>
<th>Min. DCF</th>
<th>Trial Length</th>
<th>Shortcut Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Median</td>
<td>Mean</td>
</tr>
<tr>
<td>Reference</td>
<td>0.0413</td>
<td>110.2</td>
<td>109.6</td>
</tr>
<tr>
<td><strong>90% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No prior</td>
<td>0.0537</td>
<td>2</td>
<td>2.8</td>
</tr>
<tr>
<td>$\kappa^2 = 1$</td>
<td>0.0505</td>
<td>2</td>
<td>3.0</td>
</tr>
<tr>
<td>$\kappa^2 = 2$</td>
<td>0.0474</td>
<td>2</td>
<td>3.4</td>
</tr>
<tr>
<td>$\kappa^2 = 4$</td>
<td>0.0450</td>
<td>2</td>
<td>4.0</td>
</tr>
<tr>
<td><strong>99% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No prior</td>
<td>0.0454</td>
<td>2</td>
<td>4.8</td>
</tr>
<tr>
<td>$\kappa^2 = 1$</td>
<td>0.0436</td>
<td>2</td>
<td>5.5</td>
</tr>
<tr>
<td>$\kappa^2 = 2$</td>
<td>0.0435</td>
<td>3</td>
<td>6.3</td>
</tr>
<tr>
<td>$\kappa^2 = 4$</td>
<td>0.0430</td>
<td>4</td>
<td>7.6</td>
</tr>
<tr>
<td><strong>99.9% Confidence</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No prior</td>
<td>0.0433</td>
<td>2</td>
<td>6.9</td>
</tr>
<tr>
<td>$\kappa^2 = 1$</td>
<td>0.0428</td>
<td>4</td>
<td>7.7</td>
</tr>
<tr>
<td>$\kappa^2 = 2$</td>
<td>0.0427</td>
<td>5</td>
<td>8.9</td>
</tr>
<tr>
<td>$\kappa^2 = 4$</td>
<td>0.0421</td>
<td>6</td>
<td>10.6</td>
</tr>
</tbody>
</table>
Figure 6.8: DET plot of the variance estimation with prior method at the minimum DCF operating point.
of performance (the hyperparameter $\tau$ was set to 1 for these experiments). For example, from Table 6.6 an EER of approximately 14.5% can be achieved at a 90% confidence level with $\kappa^2 = 4$ and at a 99% confidence level with no prior ($\kappa^2 = 0, \tau = 0$). While these configurations produce similar error rates, they have different utterance length characteristics, specifically, the configuration with no prior has a 1-second shorter median utterance length of 3 seconds but the mean utterance length is almost 4 seconds longer.

6.4.5 Verification Score Normalisation

Typically, raw scores output by speaker verification systems are further processed to normalise for factors such as the quality of the trained speaker model, mismatch between the training and testing conditions and the linguistic content in the test utterance. Z-Norm [6] is an example of a score normalisation technique that normalises the verification score by the mean and variance of the speaker model’s response to a set of impostor trials. H-Norm is a similar technique that additionally characterises the speaker models response to utterances from each different type of telephone handset [93].

It is straightforward to apply Z-Norm to the applications described above as it can be characterised as a simple linear transform of the frame-based scores. If the Z-Norm statistics are given by $\mu_Z$ and $\sigma_Z$ then the normalised ELLR score is given by,

$$\Lambda_Z(s) = \frac{\Lambda(s) - \mu_Z(s)}{\sigma_Z(s)}$$

$$= a\Lambda(s) + b$$

(6.10)

where $a = 1/\sigma_Z(s)$ and $b = -\mu_Z(s)/\sigma_Z(s)$. As the ELLR score is a scaled sum of the frame scores, this transform can alternatively be applied directly to the individual frame scores,

$$\Lambda_Z(s) = \frac{1}{T} \sum_{t=1}^{T} \ell'_S(t);$$

(6.11)

$$\ell'_S(t) = a\ell_S(t) + b.$$  

(6.12)
Table 6.8: The effect of shortened utterances on speaker verification performance using Z-Norm score normalisation.

<table>
<thead>
<tr>
<th>System</th>
<th>EER</th>
<th>Min. DCF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>6.6%</td>
<td>.0266</td>
</tr>
<tr>
<td>2 sec</td>
<td>20.1%</td>
<td>.0713</td>
</tr>
<tr>
<td>5 sec</td>
<td>14.1%</td>
<td>.0533</td>
</tr>
<tr>
<td>10 sec</td>
<td>10.8%</td>
<td>.0426</td>
</tr>
<tr>
<td>20 sec</td>
<td>8.8%</td>
<td>.0340</td>
</tr>
</tbody>
</table>

Hence, the same linear transform applies to the distribution of the estimated ELLR score. From the na"ive estimate, (6.4) and (6.5) become

\[
\mu_{S|Z} = a\mu_S + b = am_\ell + b, \tag{6.13}
\]

\[
\sigma_{S|Z}^2 = a^2\sigma_S^2 = \frac{a^2s^2_\ell}{T-1}. \tag{6.14}
\]

The same applies to H-Norm and C-Norm [99], which can both be considered as extensions of Z-Norm. T-Norm [6] is a more difficult prospect (see Section 6.5.3).

Results

For comparison purposes, Figure 6.9 shows the performance of a reference system using Z-Norm score normalisation. While it can be seen that the score normalisation dramatically improves the performance of the reference system, shortening these trials degrades the performance more substantially than for a system without normalisation, as confirmed in Table 6.8.

Applying the early verification decision method to this Z-Norm system produces results analogous to systems without score normalisation, as demonstrated in Figure 6.10. Due to the substantial drop in performance with short utterances the early decision method has a particularly dramatic effect on the DET curves.
Figure 6.9: DET plot of the effects of shortened utterances on speaker verification performance using Z-Norm score normalisation.
Figure 6.10: DET plot of the variance estimation with prior method at the EER operating point using Z-Norm score normalisation.
Table 6.9: Verification results using the decorrelated method at the EER operating point using Z-Norm score normalisation.

<table>
<thead>
<tr>
<th>System</th>
<th>EER</th>
<th>Trial Length</th>
<th>Shortcut Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Median</td>
<td>Mean</td>
</tr>
<tr>
<td>Reference</td>
<td>6.6%</td>
<td>108.5</td>
<td>107.6</td>
</tr>
<tr>
<td>N = 10 at 90%</td>
<td>12.0%</td>
<td>3</td>
<td>6.6</td>
</tr>
<tr>
<td>N = 10 at 99%</td>
<td>7.6%</td>
<td>8</td>
<td>20.6</td>
</tr>
<tr>
<td>N = 10 at 99.9%</td>
<td>6.8%</td>
<td>17</td>
<td>32.3</td>
</tr>
</tbody>
</table>

It was anticipated that the early decision method would produce decisions at least as quickly with Z-Norm applied as with no normalisation and, since the true and false score distributions are better separated, that shorter utterances may be used. The experimental evidence indicates that this was not the case, however, as shown in the mean and median trial lengths from Table 6.9 compared to the trial lengths in Table 6.4. The increase in both the mean and median trials lengths indicates that the task of making a verification decision was in fact complicated by the use of Z-Norm. It is hypothesised that this outcome may be the result of the offset introduced by $b = -\mu_Z(s)/\sigma_Z(s)$ into the score estimate. With a small number of samples the “true” score estimate is initially dominated by the speaker-dependent value of $b$ causing a greater number of samples to prove this prior information incorrect.

The lengthier trials in this case can be viewed as a positive outcome in two ways. Firstly the utterance lengths are still considerably shorter than using the entire utterance and provide far superior performance when compared to specifying a fixed short utterance length. For example, looking at the 99% confidence level results in Table 6.9 there is only a 1% drop in EER compared to the reference system, with at least half the trials taking 8 seconds or less. This compares to the 3.4% drop using a fixed 10 second utterance length. Comparing the mean trial length of just over 20 seconds, there is a 2.1% drop incurred when using a fixed 20 second system.

Also, the lengthier trials indicate that the early decision method is in fact ac-
6.5 Discussion and Future Directions

There is significant opportunity to improve the confidence measures presented in this work and described below are just a few potential future directions.

6.5.1 Optimising System Performance

Many experiments were presented in the previous section investigating the effect of techniques for estimating the verification scores’ variance, each with configurable parameters. Additionally, the results of these systems were analysed on a number of criteria, particularly the drop in performance relative to a reference system and the mean and median verification lengths.

Given this array of possibilities, a structured method is desirable to determine the best configuration for any particular scenario. The meaning of the word...
“best” here is also dependent on the performance or time constraints of the scenario.

Two classes of scenario are likely to be of interest for this technique, achieving the best performance with a given average verification utterance length and using the shortest possible verification length to achieve a given performance level.

It is anticipated that to perform either optimisation will require an experimental refinement approach, although given the number of parameters to configure a disciplined and efficient procedure for performing this optimisation is definitely desirable.

### 6.5.2 Compensating for Skewed Distributions

Empirical evidence suggests that the frame-based log-likelihood ratio scores produced by GMM-UBM speaker verification systems are not necessarily normally distributed. Specifically, they can exhibit highly skewed distributions. This is particularly true of genuine target trials that often exhibit significant positive skew as depicted in Figure 6.12. While for large numbers of frames, the distribution of their expectation will still be Gaussian, this is not true for smaller quantities. Particularly for trying to make early verification decisions, this skew has the potential to be a significant source of error in our confidence measures.

It may be advantageous to use a generalised Gaussian distribution that incorporates higher order moments to better represent the observed frame scores. This may cause some issues, however, in accurately and easily estimating the distribution of the ELLR score, which would no longer be Gaussian for short utterances and is unlikely to be of the same form as the generalised distribution chosen.

Another possibility is to use a distribution estimation method that does not assume a parametric representation. One possibility is to use Parzen window methods to estimate the frame score distribution. Translating the frame score distribution to the distribution of the eventual ELLR score remains a difficulty with such an approach.
6.5. Discussion and Future Directions

6.5.3 Extension for T-Norm Score Normalisation

Using Z-Norm score normalisation involved a relatively straightforward modification to the early verification decision method. In contrast, T-Norm [6] poses a more difficult problem. Recall from Section 6.4.5 that Z-Norm can be effectively represented as a linear transformation of the verification score, this is also true of T-Norm which can be described similarly through linear transformation coefficients $a_T$ and $b_T$. The difficulties arise from the calculation of these coefficients.

Unlike Z-Norm, the T-Norm coefficients describe the characteristics of the test utterance using the verification score distribution from a set of impostor models. A consequence of this is that these characteristics become dependent on the number of observations actually scored from the test utterance and they therefore evolve over time.

There are several possible approaches to implementing T-Norm with the confidence based techniques. One option is to perform T-Norm on a frame-by-frame basis. Using this arrangement each frame score is normalised individually based on the T-Norm models’ scores for the frame. This approach would work seamlessly with the confidence-based methods but has some disadvantages. Firstly, empirical evidence shows that using T-Norm on a frame-by-frame basis has infe-
rior performance to normalisation based on full utterances. Secondly, Z-Norm is usually more effective when applied before T-Norm; while this is not impossible, it is certainly more difficult with this arrangement.

Another possible approach is to treat the distribution of T-Norm model scores as a random process similar to that of the target model. It is straightforward and well understood to estimate the distribution of all the T-Norm model scores, and estimates of both the mean and variance are possible. Again assuming that the score from each T-Norm model is random variable with a Gaussian distribution then the mean of these variables is also normally distributed. Subtracting this mean from the target model score also produces a Gaussian distribution. The main difficulty arises in this approach with the division by the standard deviation of the T-Norm model scores. Determining the resulting distribution is not straightforward and certainly introduces complications in calculating the cumulative distribution functions to determine the confidence level cut-off values.

6.5.4 Application to Forensic Tasks

The experiments presented in this chapter utilise the confidence-based verification methods to use the minimum speech to make a verification decision. As noted in Section 6.2, these techniques at least theoretically have valuable properties for use in forensic tasks, particularly to help evaluate the strength of evidence.

Information such as the upper and lower bounds of the verification score on a given verification trial can provide a wealth of information in a forensic scenario to indicate the quality of the evidence. For example, knowing that the odds in favour of an hypothesis are in the range 50:1 to 2:1 provides more information than simply providing 10:1 as the best estimate. Also a range of 50:1 to 2:1 is very different to a range of 12:1 and 9:1 although they may have an equivalent best estimate. Thus the size of the confidence interval on a verification score may be an indicator as to whether the verification score actually gives a legitimate result.

One potential avenue for this work is to investigate whether the confidence measures developed in this work can be utilised to turn verification scores into calibrated likelihood ratios. This could be achieved by translating a verification
score into a distance from a threshold normalised by the estimated variance, either in terms of cumulative probability density or standard deviations. This may have further applications for score normalisation and fusion tasks as well.

6.6 Summary

A user of a system may prefer to know the confidence with which a verification decision can be made, however, there are several theoretical and practical difficulties with this goal.

Alternatively, knowing the confidence interval around a given verification score is a more practical option with a number of possible uses, including the ability to estimate the error bounds on a verification decision compared to a background population and estimating the confidence with which a verification score is above or below a given threshold.

The development of a novel method for estimating the confidence interval for the expected log-likelihood ratio scoring method used in speaker verification was presented in this chapter based on estimating the variance of individual frame scores. Several enhancements to this estimate were proposed to increase its robustness and accuracy for the peculiarities of GMM-based speaker verification.

One particular application for this information was explored to determine the minimum number of frames required to confidently make a verification decision based on a given threshold. This early verification decision method demonstrated that as little as 2–5 seconds of active speech on average was able to produce verification results approaching that of using an average of over 100 seconds of speech. Moreover, the performance loss incurred by making an early decision can be controlled by adjusting the confidence required in the resultant decision.

A number of directions for further developing the confidence-based methods developed in this chapter were also proposed.
Chapter 7

Conclusions and Future Directions

7.1 Introduction

This chapter provides a summary of the work presented in this dissertation and the conclusions drawn. The summary follows the two main research themes and areas of contribution identified in Chapter 1 — that is, modelling mismatch and modelling uncertainty.

7.2 Modelling Mismatch

As speaker recognition technology moves from the laboratory to public telephone systems and other hostile environments the detrimental effects of mismatch between the enrolment and verification conditions are consistently highlighted. The significance of mismatch is evident throughout this work; Sections 2.3.2 and 2.4.4 cite mismatch as the motivation for a variety of feature and score normalisation techniques, the results of Chapter 3 highlight the performance impact of mismatch and Chapters 4 and 5 are both devoted to addressing mismatch.
Chapter 7. Conclusions and Future Directions

7.2.1 Handset Mismatch and Feature Mapping

Chapter 4 specifically addressed the issue of handset type mismatch in telephony environments. The term handset type mismatch originally referred to differences in the type of microphone transducer used but more recently has expanded to include differences such as the speech coding and transmission processes used in wireless and digital environments. Even with the techniques proposed to alleviate handset mismatch, the performance of mismatched trials lags that of matched conditions by a factor of four in terms of equal error rate.

The recent introduction of feature mapping has shown promise in directly addressing the impact of handset mismatch by mapping feature vectors extracted from different handset contexts to a neutral feature space. This mapping is a non-linear transformation defined by the differences between a GMM representing the neutral space and a set of adapted GMMs representing each context.

Original Contributions

Two extensions to feature mapping were proposed and investigated to expand the situations in which feature mapping can be successfully applied.

- A configuration was proposed for combining the use of feature mapping with feature warping to enhance the performance of the reference system without losing the benefits of either technique. This configuration applies feature mapping prior to feature warping to avoid context identification errors in the feature mapping front-end. By separating the neutral context model and the UBM, this configuration also successfully demonstrates a theoretical advantage of feature mapping over the similar SMS technique.

- A method was proposed and evaluated for adapting the original feature mapping method to allow for effective training of feature mapping contexts in the common case of absent or inaccurate context labels for the background corpus.
This method utilised a data-driven clustering method to determine the contexts present in the available background data.

The experiments presented demonstrated the performance of systems incorporating the data-driven approach to feature mapping in a variety of scenarios from refining existing context labels to a total absence of label information. The performance provided with blindly selected background data was comparable to systems utilising the traditional approach to feature mapping training.

**Future directions**

More sophisticated clustering algorithms could provide increased utility and performance to the blind feature mapping approach. A method for automatically determining the most appropriate number of clusters would be useful when very limited training resources are available while measures to avoid local optima in the clustering procedure should ensure more consistent results with less clusters.

### 7.2.2 Explicit Modelling of Session Variability

Chapter 5 broadened the scope of the investigation of mismatch beyond handset mismatch to modelling all types of variability that can occur between sessions of the same speaker.

To this end, a technique was proposed to explicitly model the prevalent conditions in the training and verification sessions. This was achieved by expressing the GMM that best represented the acoustic observations of an utterance as the session-independent characteristics of the speaker with an additional mean offset due to the session conditions. This session variable was additionally constrained to lie in a low-dimensional session variability subspace trained on a background population.
Original Contributions

Techniques were developed to incorporate this augmented model into both the speaker enrolment and verification processes of a verification system.

- A practical and efficient method was developed to simultaneously estimate the speaker model and session variables using a similar approach to the Gauss-Seidel iterative solver.

The enrolment process involved a complex simultaneous optimisation of the speaker mean vector and session vectors using maximum \textit{a posteriori} criteria in all cases. A direct solution to this optimisation was presented but was shown to be very computationally expensive to the point of being impractical for large verification trials.

To avoid this issue, an iterative approximation method was proposed based on the Gauss-Seidel method for solving linear systems. This method was demonstrated to be both efficient and successful.

- Several verification scoring methods derived from top-$N$ ELLR scoring used in standard GMM-UBM speaker verification systems were proposed and evaluated.

It was empirically shown that estimating the session conditions of the verification utterance was beneficial and successfully addressed mismatch from both the enrolment and verification perspectives.

Alternatives to ELLR-derived scoring for session variability modelling were also discussed, such as Bayes factor scoring as described in Chapter 3 and the factor analysis likelihood ratio, proposed by Kenny, \textit{et al}.

- Score normalisation was discovered to combine particularly well with the session variability modelling approach, particularly Z-Norm followed by T-Norm.

- The role of the session variability subspace in overall system performance was investigated.
The iterative E-M method for training the session variability subspace based on a database of background speakers was presented. As this is a computationally expensive process, the sensitivity of verification performance to insufficient convergence of this training was empirically investigated as was the issue of mismatched conditions between background and evaluation databases.

The impact of subspace size was also investigated. This investigation further demonstrated that score normalisation plays an important role in this approach as it allows performance benefits to be gained from a higher dimensional representation of the session conditions.

- Significant performance improvements were also demonstrated for verification utterances with as little as 10 seconds of active speech.

Future directions

The main avenue for further developing the session modelling approach is in the verification scoring method used. The combination of session variability modelling and Bayes factor scoring in a similar approach to that described in Chapter 3 particularly holds significant promise of improved verification performance.

7.2.3 Conclusions

Experiments on conversational telephony data demonstrated the effectiveness of the both the feature mapping and explicit session variability modelling techniques. Both techniques were also effective without the need for training data labelled for the conditions of interest.

Explicit session variability modelling produced superior results overall with up to a 68% reduction in detection cost over the reference system and consistent reductions of a similar magnitude across corpora exhibiting different types of mismatch.
Chapter 7. Conclusions and Future Directions

Feature mapping, however, has the advantage of requiring no labelling of its training data whatsoever with the clustering enhancement proposed in this work whereas modelling session variability still requires training data with speaker labels.

7.3 Modelling Uncertainty

With an infinite quantity of speech for enrolling or verifying a speaker it is possible to be very certain that our verification decisions closely match the truth. In reality, this is not the case. Training and testing data is limited and uncertainty in model and verification score estimates result.

Section 2.4.2 reviewed the use of MAP adaptation under the Bayesian framework for reducing this uncertainty in the speaker model estimation procedure to increase the robustness of verification systems. This approach was taken further with the introduction of Bayes factor scoring in Chapter 3 by modelling the remaining uncertainty of the model estimates in the scoring procedure.

Chapter 6 introduced confidence based verification methods by considering the verification score for a trial as an estimate of the “true” verification score with an amount of uncertainty. By measuring the degree of this uncertainty, this information can be used to determine the confidence in a verification decision.

7.3.1 Modelling Uncertainty in Speaker Model Estimates

Chapter 3 reviewed the verification problem as a statistical hypothesis test and developed the Bayes factor as the optimal decision criterion under a Bayesian framework. The ability of the Bayesian approach to incorporate prior information into the scoring process and to allow for uncertainty in speaker model parameters was highlighted.

Original Contributions

The Bayes factor was applied in the context of speaker verification system following the GMM-UBM structure. This lead to the following research contributions.
7.3. Modelling Uncertainty

- A novel approximation of the Bayes factor specific to GMMs was derived using an incremental learning approach to overcome the difficulties of the missing component occupancy information.

  This derivation was suitable for use in standard GMM-UBM systems as a direct replacement for ELLR scoring. The implementation consisted of scoring and incremental MAP adaptation steps on each consecutive frame of the test utterance with a number efficiency improvements also included.

- A frame-weighting factor was introduced in the incremental adaptation step to compensate for the high levels of correlation present in short-time acoustic features.

  This weighting factor reduced the rate of adaptation as the original derivation invokes the inherently untrue assumption of independently, identically distributed feature vectors.

Future directions

While this approach demonstrated some performance improvements in matched conditions, handset and other types of mismatch pose an even greater challenge than for standard ELLR scoring. It was therefore concluded that the most promising avenues for further development of the Bayes factor approach is in conjunction with methods that address and reduce mismatch. Specifically, the combination of Bayes factor scoring and explicit session variability modelling may provide a potent combination.

7.3.2 Modelling Uncertainty in Verification Scores

While the ultimate goal of producing an accurate confidence in a verification decision seems unlikely to be obtained for several theoretical reasons, modelling the uncertainty in the score produced by a verification system provides information with considerable practical use. Knowing the confidence interval around a given verification score allows the ability to estimate the error bounds on a verification decision compared to a background population and estimating the confidence
with which a verification score is above or below a given threshold. Chapter 6 investigated the estimation and use of such confidence measures in the context of speaker verification system producing ELLR scores.

Original Contributions

- A method for estimating the confidence interval of expectation-based scores was proposed based on the estimated variance of the underlying frame score distribution.

  This derivation follows as the distribution resulting from the sum of random variables is Gaussian, as stated by the central limit theorem.

- Enhancements were proposed to improve the accuracy and robustness of the confidence interval to compensate for the peculiarities of ELLR scoring for GMM-based speaker verification.

  To reduce the impact of highly correlated consecutive frame scores resulting from correlated feature vectors, groups of frames were combined and treated as individual, but decorrelated, observations. A prior distribution was also introduced into the frame score estimation procedure to increase the robustness of this estimate when very few observations are available. This is necessary as the central limit theorem is designed for large numbers of observations.

- The confidence measure estimation procedure was extended to incorporate Z-Norm and H-Norm score normalisation techniques by interpreting these normalisations as linear transformations of the score space.

- The application of producing confident verification decisions given a threshold with the minimum number of frames was investigated to demonstrate the effectiveness of the confidence measures.

  This early verification decision method demonstrated that as little as 2–5 seconds of active speech on average was able to produce verification re-
7.3. Modelling Uncertainty

...results approaching those of using an average of over 100 seconds of speech. Moreover, the performance loss incurred by making an early decision can be controlled by adjusting the confidence required in the resultant decision.

Future directions

Several opportunities exist for further developing these confidence-based techniques.

A number of configuration parameters were included into the variance estimation procedure. As these configuration variables interact in a complex fashion and are often introduced for competing goals, it is not a straightforward task to determine the optimal values for these parameters for a given application. A systematic procedure for optimising these parameters would further increase the utility of the approach.

While some characteristics of the frame score distribution were addressed others, such as the skew that is common in speaker verification scores, were not. There is an opportunity to provide more accurate estimations of the final score confidences through the use of a distribution model that is more appropriate than the current Gaussian assumption, leading to better overall accuracies with less observations.

It is also highly desirable to extend this work to accommodate T-Norm score normalisation. This is likely to be important in practice as T-Norm represents a main-stay of modern high-performance speaker verification systems but poses some theoretical difficulties in combination with this work.

The use of these techniques in forensic tasks is also a very promising area of research, however, it falls outside the scope of this work.

7.3.3 Conclusions

Introducing Bayes factor scoring to the GMM-UBM speaker verification structure as an alternative to ELLR scoring provided a means of modelling the uncertainty in the speaker model estimates in the verification process. By doing so it extended the use of the Bayesian framework of MAP adaptation form enrolment to
verification.

Modelling the speaker model uncertainty with Bayes factor scoring provided mixed success with gains in matched conditions but resulted in degraded performance in mismatched conditions.

Modelling uncertainty in the final verification score, however, provided substantial practical benefits. Modelling score uncertainty in the form of confidence intervals was capable of reducing the quantity of active speech required by more than a factor of 10 on average for verification trials with minimal difference in overall performance. Moreover, this is only one possible application for confidence measures in speaker verification with the potential to impact in other areas such as forensic investigations.

7.4 Summary

The overall aim of this work was to enhance the usefulness of speaker recognition technology in realistic, adverse conditions with a specific focus on telephony environments. This aim was to be achieved through reducing error rates in these conditions and removing barriers to adoption, such as a lack of labelled data.

Two main research themes were pursued: Modelling mismatch and modelling uncertainty.

To directly address the performance degradation incurred through mismatched conditions it was proposed to directly model this mismatch. Feature mapping was evaluated for the specific purpose of combating handset mismatch and was extended through the use of a blind clustering algorithm to remove the need for accurate handset labels for the background training data.

This mismatch modelling approach was then generalised through explicitly modelling the conditions present in a session as a constrained offset of the speaker model means. This session variability modelling approach enabled the modelling of arbitrary sources of mismatch, including handset type, and provided a halving of the error rates in many cases.

Methods to model the uncertainty in speaker model estimates and verification
scores were developed to address the difficulties of limited data in both training and testing. The Bayes factor was introduced to account for the uncertainty of the speaker model parameter estimates in testing by extending the Bayesian approach of MAP adaptation to the verification criterion. This produced some success overall and in matched conditions but had a negative effect in mismatched conditions.

Modelling the uncertainty in the verification score itself, however, met with significant success by providing confidence intervals on the final verification score. This enabled an order of magnitude reduction in the average speech required to make a confident verification decision based on a threshold. The confidence measures developed in this work may also have significant applications for forensic speaker verification tasks.
Bibliography


mation of HMM,” in Automatic Speech and Speaker Recognition: Advanced
Topics (C.-H. Lee, F. Soong, and K. Paliwal, eds.), pp. 83–107, Boston,

and J. Ortega-Garcia, “Robust estimation, interpretation and assessment of
likelihood ratios in forensic speaker recognition,” Computer Speech & Lan-

[40] I. S. Gradshteyn and I. M. Ryzhik, Table of Integrals, Series, and Products.

[41] L. Heck and M. Weintraub, “Handset-dependent background models for ro-
bust text-independent speaker recognition,” in IEEE International Confer-
ence on Acoustics, Speech, and Signal Processing, vol. 2, pp. 1071–1074,
1997.

[42] H. Hermansky, “Perceptual linear predictive (PLP) analysis of speech,” The
Journal of the Acoustical Society of America, vol. 87, no. 4, pp. 1738–1752,
1990.

[43] H. Hermansky and N. Morgan, “RASTA processing of speech,” IEEE Trans-


[45] X. Huang, A. Acero, and H.-W. Hon, Spoken language processing: a guide
to theory, algorithm, and system development. Upper Saddle River, NJ:

classifiers,” in Advances in Neural Information Processing Systems II,


