

2011

# Feature Selection Methods for Boosted Crossspectral Face Recognition

Sirisha Boothapati  
*West Virginia University*

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

---

## Recommended Citation

Boothapati, Sirisha, "Feature Selection Methods for Boosted Crossspectral Face Recognition" (2011). *Graduate Theses, Dissertations, and Problem Reports*. 3304.  
<https://researchrepository.wvu.edu/etd/3304>

This Thesis is brought to you for free and open access by The Research Repository @ WVU. It has been accepted for inclusion in Graduate Theses, Dissertations, and Problem Reports by an authorized administrator of The Research Repository @ WVU. For more information, please contact [ian.harmon@mail.wvu.edu](mailto:ian.harmon@mail.wvu.edu).

# Feature Selection Methods for Boosted Crossspectral Face Recognition

by

Sirisha Boothapati

Thesis submitted to the  
College of Engineering and Mineral Resources  
at West Virginia University  
in partial fulfillment of the requirements  
for the degree of

Master of Science  
in  
Electrical Engineering

Donald A. Adjero, Ph.D.  
Xin Li, Ph.D.  
Natalia A. Schmid, Ph.D., Chair

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia  
2011

Keywords: Face recognition, SWIR, Gabor Wavelets, Kullback Leibler divergence, Local Binary Pattern, d-prime, Dimensionality reduction, Adaboost, Feature Selection

Copyright 2011 Sirisha Boothapati

# **Abstract**

Feature Selection Methods for Boosted Crossspectral Face Recognition

by

Sirisha Boothapati

As a successful technology in commercial as well as surveillance applications, face recognition has attracted significant attention. Wide range of applications using this technology has been a constant motivation for research developments over the past decade. Major issues with color images are illumination variation and pose change in which illumination variation has been overcome by involving Short and Long Wave Infrared imagery. However, properties of color images and Short and Long Wave Infrared images are different. Their crossmatching presents a great challenge.

In this thesis, we propose a methodology that will be able to crossmatch color face images and Short Wave Infrared face images reliably and accurately. We first adopt a recently designed Boosted Local Gabor Pattern Improved (LGPI) encoding and matching technique to encode face images in both visible and Short Wave Infrared (SWIR) spectral bands. We then apply feature selection methods to prune irrelevant information in encoded data and to improve performance of the Boosted LGPI technique. In this thesis, we propose three novel feature selection methods: (1) Genuine segment score thresholding, (2)  $d'$ -based thresholding and (3) two Adaboost inspired methods. We further compare the performance of the original Boosted LGPI face recognition method with the method involving feature selection step. Under a general parameter set up, a significant performance improvement is observed and perfect verification performance is achieved.

# Acknowledgements

A few lines of acknowledgement do not fully express my gratitude and appreciation for those who guided and supported me through the last three years.

I have been privileged to have Dr. Natalia A. Schmid as my advisor. Her attention to detail, quest for excellence, love for perfection, ingenious ideas and understanding has always motivated me to give my best time and again. She has helped me develop and fine my academic skills including research, presentation and writing skills. I am greatly indebted to her not only for making my masters a memorable experience but also for the principles and values I gathered from her.

I am grateful to Dr. Donald A. Adjero and Dr. Xin Li for their expert advice and support. This work was supported by Office of Naval Research under the contract number N00014-08-1-0895 and by Army Research Laboratory under the contract number W911NF-10-2-0109.

I thank my lab mates for their support and technical discussions which often gave me ideas for my research. I wish all of them success in their future endeavors. I also thank my friends for their support during hard times. My heartfelt thanks to my parents Seshu Kumar and Jhansi Rani, my sister Surekha, my brother Suresh Kumar and my relatives for their support, effort and sacrifice without which I would not have come this far.

To all of you I dedicate this work.

# Contents

<b>Acknowledgements</b>	<b>iii</b>
<b>List of Figures</b>	<b>vi</b>
<b>List of Tables</b>	<b>viii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Biometrics . . . . .	1
1.2 Face Recognition . . . . .	2
1.3 Multispectral Imaging . . . . .	4
1.4 Problem Description . . . . .	5
1.5 Contribution of the Thesis . . . . .	5
1.6 Organization of the thesis . . . . .	6
<b>2 Background</b>	<b>7</b>
2.1 Encoding Technique . . . . .	7
2.2 Multispectral Data-set . . . . .	7
2.3 Boosted LGPI . . . . .	8
2.3.1 Preprocessing . . . . .	8
2.3.2 Filtering . . . . .	9
2.3.3 Encoding of magnitude and phase . . . . .	9
2.3.4 Matching . . . . .	10
2.4 Introduction to Dimensionality Reduction . . . . .	10
2.5 Feature selection . . . . .	11
2.6 Proposed Feature Selection Methods . . . . .	12
<b>3 Genuine Segment Score Thresholding</b>	<b>13</b>
3.1 Motivation . . . . .	13
3.2 Verification . . . . .	14
3.2.1 Results for Verification . . . . .	15
3.3 Identification . . . . .	16
3.3.1 Results for Identification . . . . .	18

<b>4</b>	<b><math>d'</math>-based Thresholding</b>	<b>24</b>
4.1	Motivation . . . . .	24
4.2	Verification . . . . .	25
4.2.1	Results for Verification . . . . .	26
4.3	Identification . . . . .	26
4.3.1	Results for Identification . . . . .	27
4.4	Modified $d'$ . . . . .	29
4.4.1	Results for verification . . . . .	30
4.4.2	Results for Identification . . . . .	32
<b>5</b>	<b>Adaboost Inspired Methods</b>	<b>40</b>
5.1	Adaboost . . . . .	40
5.1.1	Algorithm . . . . .	40
5.2	Method 1 . . . . .	41
5.2.1	Algorithm . . . . .	41
5.2.2	Verification . . . . .	43
5.2.3	Identification . . . . .	44
5.3	Method 2 . . . . .	46
5.3.1	Algorithm . . . . .	47
5.3.2	Verification . . . . .	48
5.3.3	Identification . . . . .	52
<b>6</b>	<b>Conclusion and Future Work</b>	<b>56</b>
6.1	Conclusion . . . . .	56
6.2	Future Work . . . . .	57
	<b>References</b>	<b>58</b>

# List of Figures

1.1	Block diagram showing the stages of a basic biometric system . . . . .	2
1.2	Face images captured at different wavelengths. Wavelengths are in order: visible, 950nm, 1050nm, 1150nm, 1250nm, 1350nm, 1450nm, 1550nm, and 1650nm. . . . .	4
1.3	Block diagram showing the stages of a basic biometric system. . . . .	5
2.1	Sample images of multispectral data. First row are the visible images with neutral and talking expressions. Second row are the images captured at 1550nm with neutral and talking expressions. . . . .	8
3.1	Histogram of Genuine segment scores of training set considering 256 segments of the template. . . . .	16
3.2	Plot showing the ROCs of genuine segment scores thresholded at different thresholds. It can be observed that removal of some segments at some threshold values result in improvement of performance. This is a plot considering 256 segments and the training set being the talking images and the mask assignment based on the second rule of selecting informative segments. . . .	17
3.3	ROC showing the perfect performance on testing set at $\tau = 340$ . . . . .	18
4.1	Histogram of $d'$ segment values for the training data partitioned into 256 segments. . . . .	27
4.2	Plot showing the ROCs of $d'$ thresholding at different values of thresholds. It can be observed that by discarding segments at some threshold values we achieve improved performance. This is a plot considering 256 segments per histogram with the training set being the neutral images and with the segment selection based on the first rule of selecting informative segments. . . . .	28
4.3	ROC showing the perfect performance on testing set at $\tau = 0.2$ . . . . .	29
4.4	Histogram of $md'$ segment values for the training data partitioned into 256 segments. . . . .	34
4.5	Plot showing the ROCs of $md'$ thresholding at different values of thresholds. It can be observed that by discarding segments at some threshold values we achieve improved performance. This is a plot considering 256 segments per histogram with the training set being the talking images and with the segment selection based on the second rule of selecting informative segments. . . . .	35
4.6	ROC showing the perfect performance on testing set at $\tau = 0$ . . . . .	36

5.1	Plot showing original and improved performance using Adaboost Method 1 on training set of talking images. . . . .	44
5.2	Plot showing original and improved performance using Adaboost Method 1 on testing set. . . . .	45
5.3	Histogram of genuine and impostor scores before applying the Adaboost Method 1. . . . .	45
5.4	Histogram of normalized genuine and impostor scores after applying the Adaboost Method 1. . . . .	46
5.5	Plot showing the performance of system at different thresholds. Adaboost Method 2 is applied on training set consisting of neutral images. . . . .	50
5.6	Plot showing the ROCs of original and improved performance using Adaboost Method 2 on testing set. . . . .	50
5.7	Histogram of genuine and impostor scores before applying the Adaboost Method 2. . . . .	51
5.8	Histogram of normalized genuine and impostor scores after applying the Adaboost Method 2. . . . .	51



# List of Tables

3.1	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions. . . . .	19
3.2	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions. Masks used in this case are computed using the training set of images with neutral expressions. . . . .	20
3.3	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions. . . . .	20
3.4	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions. Masks used in this case are computed using the training set of images with talking expressions. . . . .	21
3.5	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral and talking expressions. . . . .	21

3.6	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral and talking expressions. Masks used in this case are computed using the training set of images with neutral and talking expressions. . . . .	22
3.7	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3 <sup>rd</sup> column represents the original identification rate of the training set computed as explained above. The 4 <sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2 <sup>nd</sup> column. Maximum identification rate is mentioned in the 4 <sup>th</sup> column for all the comparisons that do not result in performance improvement. . . . .	22
3.8	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3 <sup>rd</sup> column represents the original identification rate of the testing set computed as explained above. The 4 <sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2 <sup>nd</sup> column. Maximum identification rate is mentioned in the 4 <sup>th</sup> column for all the comparisons that do not result in performance improvement. . . . .	23
4.1	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions. . . . .	30
4.2	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions. Masks used in this case are computed using the training set of images with neutral expressions. . . . .	30
4.3	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions. . . . .	31

4.4	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions. Masks used in this case are computed using the training set of images with talking expressions. . . . .	31
4.5	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral and talking expressions. . . . .	32
4.6	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral and talking expressions. Masks used in this case are computed using the training set of images with neutral and talking expressions . . . . .	32
4.7	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3 <sup>rd</sup> column represents the original identification rate of the training set computed as explained above. The 4 <sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2 <sup>nd</sup> column. Maximum identification rate is mentioned in the 4 <sup>th</sup> column for all the comparisons that do not result in performance improvement. . . . .	33
4.8	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3 <sup>rd</sup> column represents the original identification rate of the testing set computed as explained above. The 4 <sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2 <sup>nd</sup> column. Maximum identification rate is mentioned in the 4 <sup>th</sup> column for all the comparisons that do not result in performance improvement. . . . .	33
4.9	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions. . . . .	34

4.10	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions. Masks used in this case are computed using the training set of images with neutral expressions. . . . .	35
4.11	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions. . . . .	36
4.12	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions. Masks used in this case are computed using the training set of images with talking expressions. . . . .	37
4.13	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral and talking expressions. . . . .	37
4.14	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral and talking expressions. Masks used in this case are computed using the training set of images with neutral and talking expressions. . . . .	38
4.15	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3 <sup>rd</sup> column represents the original identification rate of the training set computed as explained above. The 4 <sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2 <sup>nd</sup> column. Maximum identification rate is mentioned in the 4 <sup>th</sup> column for all the comparisons that do not result in performance improvement. . . . .	38

4.16	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3 <sup>rd</sup> column represents the original identification rate of the testing set computed as explained above. The 4 <sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2 <sup>nd</sup> column. Maximum identification rate is mentioned in the 4 <sup>th</sup> column for all the comparisons that do not result in performance improvement. . . . .	39
5.1	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column represents the original identification rate of the training set computed as explained above. The 3 <sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 1.	47
5.2	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column represents the original identification rate of the testing set computed as explained above. The 3 <sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 1.	47
5.3	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions using Adaboost Method2. . . . .	52
5.4	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions using Adaboost Method2. . . . .	52
5.5	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions using Adaboost Method2. . . . .	53
5.6	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions using Adaboost Method2. . . . .	53

5.7	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of one image with neutral expression and the other with talking expression using Adaboost Method2. . . . .	54
5.8	Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of one image with neutral expression and the other with talking expression using Adaboost Method2. . . . .	54
5.9	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column represents the original identification rate of the training set computed as explained above. The 3 <sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 2. . . . .	54
5.10	The 1 <sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2 <sup>nd</sup> column represents the original identification rate of the testing set computed as explained above. The 3 <sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 2. . . . .	55

# Chapter 1

## Introduction

### 1.1 Biometrics

Derived from a Greek word “bios” meaning life and “metron” meaning measure, biometrics is an automated method to recognize a person based on the person’s physiological or behavioral traits. A biometric system is a pattern recognition system with biometric sample acquired by a sensor as its input and a decision about whom the biometric sample belongs to as its output. The system extracts salient features and compares the feature set against feature sets of all individuals recorded in a database [1],[2]. Although in the past, biometrics was used in law enforcement to identify criminals, the ease, accuracy and speed of the biometric systems has paved its way in a large number of civilian applications. Any physical or behavioral traits can be used as a biometric trait to identify a human as long as it is universal, distinct, permanent and collectable. Fingerprints [2],[3], iris [4], face [5], hand geometry are a few examples of physical traits whereas voice, signature and gait [6] could be considered behavioral traits. Based on the application, a biometric system can be operated in verification or identification mode. Verification is a scenario in which a system validates a person’s identity by comparing the captured biometric with the person’s own template stored in a database [5]. Identification is to find the identity of a person by comparing the captured data with the database to find if there is any match. In general, biometric devices Fig. 1.1 can be explained with a three-step procedure:

- (1) A sensor takes an observation. The type of sensor and its observation depend on



Figure 1.1: Block diagram showing the stages of a basic biometric system

the type of biometric devices used. This observation gives us a biometric signature of the individual. A sensor can range from an Infrared camera (IR), Red Green Blue (RGB) sensor, microphones for voice capture, chemical sensors for fingerprints, to name a few.

(2) A computer algorithm normalizes the biometric signature so that it is in the same format (size, resolution, view, etc.) as enrolled signatures in the system database. The normalization of the biometric signature gives us a normalized signature of the individual.

(3) A matcher compares the normalized signature with the set (or sub-set) of normalized signatures on the system's database and provides a match score that compares the individual's normalized signature with each signature in the database set (or sub-set).

In this thesis, emphasis is on Face recognition in which facial images are used as biometric trait [5], [7].

## 1.2 Face Recognition

Face recognition has attained a lot of significance in the past two decades due to its performance and ease of capturing the image. The major advantages of using face as a biometrics are (1) that no contact is required to capture face as opposed to fingerprint, hand geometry and other biometrics (2) face images can be captured at a distance. The commercial applications range from static matching of photographs on credit cards, ATM cards, passports, driver licenses, and photo ID to real-time matching with still images or video image sequences for access control. Face recognition research still faces challenges in some specific domains such as pose, illumination changes, environmental variations. Although numerous methods have been proposed to solve such problems and have demonstrated significant promise, the difficulties still remain. For these reasons, the matching performance in



current automatic face recognition is often lower compared to the performance of fingerprint and iris recognition systems. A thorough introduction to face recognition and to state-of-the-art methods for face image enhancement, encoding and matching can be found in [5], [8], [9], [10].

While many researchers working on face recognition in color range of electromagnetic spectrum, other spectra such as Near Infrared (NIR), Short Wave Infrared (SWIR), Thermal Infrared can be used for face recognition. There have been many works published on the topic of Thermal face recognition [11]. Use of NIR for face recognition has also been explored. In recent years the focus has shifted from face recognition within the same spectral band to crossspectral comparisons. As an example Kong et al. [12] show that when NIR and color images are fused and the fused image is processed with FaceIT software, the results show an improvement when compared to comparisons within the same spectral band. Local Binary Pattern (LBP) and Histogram of Gradient (HOG) features are used in [13]. Results using this encoding technique outperforms Cognitecs FaceVACS with cross spectral matching of NIR and color images. In [13] Linear Discriminant Analysis (LDA) is used for dimensionality reduction. A study of face recognition in thermal and visible spectrum using Principal Component Analysis (PCA) and FaceIt is published in [14]. They show that the performance using PCA in visible spectrum is better compared to that in thermal IR spectrum. Fusion at match score level results in performance comparable to that obtained in visible spectrum.

Crosscomparison of color face images and NIR face images can be performed using the encoding and matching techniques designed for color face images. This is because the illumination and textural properties of NIR and color images are relatively similar. Matching color images and Short or Long Wave Infrared face images is a more challenging endeavor. In this case the encoding and matching techniques designed to operate on color images do not perform well on SWIR or Long Wave Infrared (LWIR) data. This creates a need to design new encoding and matching techniques for crossmatching face images from different spectral bands.



Figure 1.2: Face images captured at different wavelengths. Wavelengths are in order: visible, 950nm, 1050nm, 1150nm, 1250nm, 1350nm, 1450nm, 1550nm, and 1650nm.

### 1.3 Multispectral Imaging

Multispectral imaging technique was first introduced for satellite imaging for geo-spatial structures. Images can be captured at wavelengths beyond the visible spectrum that provides with additional information that the human eye cannot see. Hyperspectral imaging is another technique that captures 4D image and displays it as a function of spatial location and wavelength [15]. Face images can be captured at different wavelengths ranging from 250 nm (Shortwave Ultraviolet) to 12-14  $\mu\text{m}$  LWIR.

Imaging at different wavelengths in the electromagnetic spectrum results in distinct textural information across multiple channels [16]. It can be observed from Fig. 1.2 that the information changes drastically as the wavelength increases from visible to 1650 nm.

Near Infrared (NIR) can be photographed with a photographic film or captured using a charge-coupled device (CCD) or CMOS sensor of the kind used in digital cameras and camcorders. In these situations NIR generally behaves like light. In particular it passes through glass and is reflected and refracted by conventional optics, but some image differences will be evident as a result of imaging with the longer wavelengths. Electromagnetic spectrum ranges are shown in Fig. 1.3. While Long Wave Infrared (thermal infrared) imaging senses heat radiation, NIR imaging is more like normal visible light imaging, though longer NIR is not invisible to the naked eyes. Ultraviolet radiation is harmful to the human body and



Figure 1.3: Block diagram showing the stages of a basic biometric system.

cannot be used for face recognition applications [17].

## 1.4 Problem Description

With the success of face recognition using the visible spectrum, all or most database images are captured at visible spectrum. However, the illumination problem in face images as well as the problem of performing surveillance at night are overcome by introducing Infrared imagery. The challenge lies in matching the captured IR image with visible images in the database. Attempts have been made to compare images captured at Far Infrared range (thermal Images), but SWIR face recognition has not been explored. In this thesis, we match images captured at visible spectrum to the images captured at 1550 nm in the Infrared region (SWIR).

## 1.5 Contribution of the Thesis

Face recognition using SWIR is an ongoing challenge. This spectral band is of special interest to DoD, since the light source illuminating a scene at 1550 nm is not visible to the human eye. Since the properties of images are different at different wavelengths, existing face recognition algorithms fail to match these images. An encoding technique proposed elsewhere [18] is used to create a template of any given image at any wavelength. This work is directed towards matching images captured at different wavelengths using the encoding technique and applies the proposed feature selection methods to improve the performance of the system.

Certain parameters are learned and the variation of parameters with performance is analyzed in each of the methods. Performance of feature selection methods is analyzed in

both verification and identification modes.

## 1.6 Organization of the thesis

In Chapter 2, we layout a background. We describe the encoding technique used to create the templates of images and also explain the importance of dimensionality reduction, feature selection and the necessity of applying these techniques to the encoded data and the data used in the experiments. Chapter 3, Chapter 4 and Chapter 5 describe the proposed feature selection methods. Conclusions, contributions and future work are provided in Chapter 6.

# Chapter 2

## Background

### 2.1 Encoding Technique

Images can be captured at different spectral bands of electromagnetic spectrum. Face recognition algorithms are designed to perform recognition in a certain spectral band. In surveillance applications, the images are captured at NIR spectrum but the gallery images are color images, which demands that the face recognition algorithms have to be successful in matching images at different spectral bands. One such encoding method, Boosted Local Gabor Pattern Improved (LGPI) is used to encode the face images [18]. In this chapter, we will review the encoding technique and explain the use of feature selection to this encoding technique. In Section 2.2 we will present the data-set used. In Section 2.3 we explain the different stages followed in encoding a face image.

### 2.2 Multispectral Data-set

Pre- TINDERS (Tactical Imager for Night/Day Extended-Range Surveillance) data-set [19] acquired from a distance 1.5 m is used to analyze the performance of dimensionality reduction techniques. The data-set consists of images of 48 classes captured at two wavelengths: Visible spectrum and 1550 nm (Short Wave Infrared spectrum), with 8 images per class. Four images are captured at each spectral band, in which two images have neutral expression and two images of the person talking. Sample images are shown in Fig. 2.1. Im-

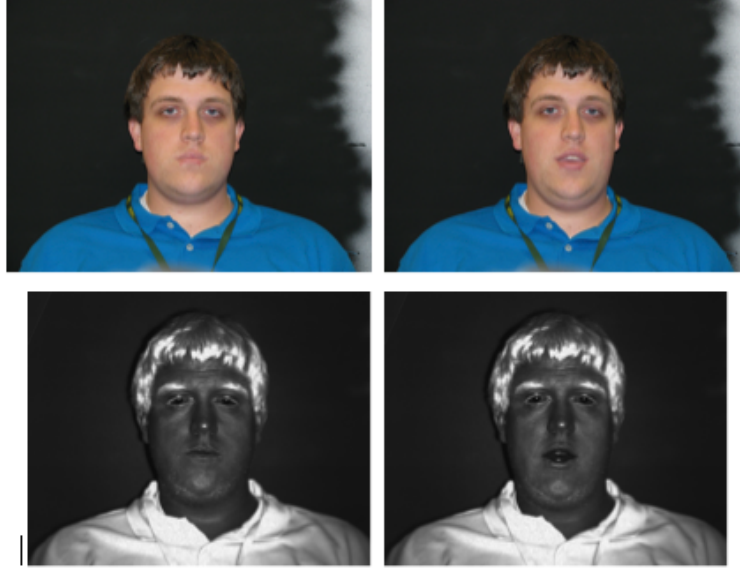


Figure 2.1: Sample images of multispectral data. First row are the visible images with neutral and talking expressions. Second row are the images captured at 1550nm with neutral and talking expressions.

ages in the first row are captured at visible spectrum with neutral and talking expressions. Images in the second row are captured at 1550 nm with neutral and talking expressions.

## 2.3 Boosted LGPI

A boosted LGPI encoding technique is used, which includes both magnitude and phase information [18]. Feature sets with better performance are used to compute the matching score between two feature vectors. All the images are mapped into a canonical representation resulting in an image of size  $128 \times 128$  pixels using the position of the eyes and nose, which are manually selected. A similarity transformation  $S$  (rotation, scaling, translation) is applied to each image such that the eyes and nose tip are in fixed positions.

### 2.3.1 Preprocessing

Color images are transformed into gray scale images while SWIR images are preprocessed using the nonlinear transformation  $\log(1+\alpha X)$  where  $X$  is the SWIR image and  $\alpha$  is a parameter estimated from a single visible image and a single SWIR image.

### 2.3.2 Filtering

A bank of Gabor filters with 4 scales and 8 orientations is applied to the cropped face image resulting in a total of 32 filter responses. The mathematical equation for the filter is as follows:

$$G(z, \theta, s) = \frac{\|K(\theta, s)\|}{\sigma^2} \exp \left[ \frac{\|K(\theta, s)\|^2 \|z\|^2}{2\sigma^2} \right] [e^{iK(\theta, s)z} - e^{-\sigma^2/2}], \quad (2.1)$$

where  $\sigma^2$  is the variance of the Gaussian kernel,  $K(\theta, s)$  is the wave vector. The magnitude and phase of the wave vector determine the scale and orientation of the oscillatory term and  $z = (x, y)$ . A normalized and preprocessed face image  $I(z)$  is convolved with the Gabor filter  $G(z, \theta, s)$  at orientation  $\theta$  and scale  $s$  resulting in the filtered image  $Y(z, \theta, s)$ .

### 2.3.3 Encoding of magnitude and phase

Both magnitude and phase responses are used to effectively encode multispectral data. Magnitude is encoded using Local Binary Pattern (LBP) technique [20] while phase is encoded using Generalized Local Binary Pattern (GLBP) operator. The defined GLBP operator is a generalization of the encoding method introduced in [21]. Both the operators consider the relationship among 12 neighbors at both radii  $r = 1, 2$ .

$$LBP_{r,12}^u(x) = \mathcal{U} \left\{ \sum_{i=0}^{11} \mathcal{I}\{x_i - x\} 2^i \right\}, \quad (2.2)$$

where  $\mathcal{U}$  is the uniform pattern mapping and  $x_i$  are the neighbors of a value  $x$  at radius  $r$  and  $\mathcal{I}(\cdot)$  is the indicator function of the event in  $\{\cdot\}$ .

Gabor phase response is encoded in the interval  $[0, 2\pi]$  as a uniform generalized binary operator defined as:

$$GLBP_{r,12,t}^u(x) = \mathcal{U} \left\{ \sum_{i=0}^{11} \mathcal{T}_t\{x_i - x\} 2^i \right\}, \quad (2.3)$$

where  $\mathcal{T}_t$  is a thresholding operator based on the threshold  $t$ .

### 2.3.4 Matching

Each encoded response is divided into non overlapping blocks of size  $8 \times 8$  resulting in 256 blocks. Histograms of the intensity values in blocks are displayed. Histograms are concatenated, and the same is repeated for all the filter responses.

Sum of two Kullback-Leibler distances [22] is used as a measure to compare the feature vectors of images captured at different wavelengths. Consider two images  $A$  and  $B$  and the feature vectors are denoted by  $H_A$ ,  $H_B$ , respectively. The symmetric Kullback-Leibler distance between the two matrices is defined as:

$$d(A, B) = \sum_{k=1}^K H_A(k) \log \frac{H_A(k)}{H_B(k)} + \sum_{k=1}^K H_B(k) \log \frac{H_B(k)}{H_A(k)}, \quad (2.4)$$

where  $K$  is the total length of the feature vectors  $H_A$  or  $H_B$ .

Since the feature vectors obtained are very long and contain long sequences of zeros, feature selection/dimensionality reduction techniques explained in Chapter 3, Chapter 4, Chapter 5 are applied to remove uninformative parts of the histogram and to boost the performance of the multispectral face recognition system.

## 2.4 Introduction to Dimensionality Reduction

Real world sensory data such as audio signals, Electro Optical (EO) sensory data (color and infrared images/video), acoustic waveforms, radar and sonar data are examples of high dimensional data. Many pattern recognition systems are designed to first reduce dimension of input data to its subset and then to extract informative features. Dimensionality reduction techniques are used to reduce the number of data dimensions to the intrinsic dimensions. Intrinsic dimensionality of data is the number of parameters needed to account for the observed properties of the data [23]. Dimensionality reduction has an important role to mitigate the curse of dimensionality.

Dimensionality reduction can be described as follows. Consider a dataset  $X$  of  $n$  data vectors each of length  $D$ . Assume that this dataset has an intrinsic dimensionality  $d$ . Dimensionality reduction techniques transform the dataset  $X$  of dimensionality  $D$  to another



dataset  $Y$  of dimensionality  $d$ . Dimensionality reduction is an ill posed problem for the reason that intrinsic dimensionality of any given dataset is unknown. Dimensionality reduction techniques can be linear, non-linear, global, local. Traditional methods like Principal Component Analysis (PCA) [24] are linear and do not perform well on complex real world data. Global Techniques such as Multidimensional Scaling (MDS) [25],[26], isomap [27], Maximum variance unfolding(MVU) [28], diffusion maps [29],[30], kernel PCA [31], multi-layer autoencoders [32],[33] suits the dimensionality reduction for real world complex data. Linear Embedding (LLE) [34], Laplacian Eigen maps [35], Heissian LLE [36], Local Tangent Space Analysis (LTSA) [37] are some of the local techniques.

## 2.5 Feature selection

Feature selection is a process of selecting a subset of features so that the feature set is optimally reduced. High dimensional data which contains thousands of features can contain a lot of irrelevant features. Feature selection methods are used to filter such irrelevant data. Feature selection methods are broadly classified into two categories, filter model [38] and wrapper model [39],[40]. Filter model depends on the general characteristics of training data without involving any kind of learning. Wrapper models requires an algorithm to select the features and then evaluate its response. Other model is hybrid model [41],[39],[42] of algorithms combine both filter and wrapper models.

Filter models can further be divided into two kinds. Subset selection and feature weighing. Relief [43] is a good example for feature weighing. In this algorithm, the features are weighed according to the relevance. However, redundant features cannot be completely removed. Subset search algorithms removes the irrelevant data and selects only a subset of features. The search stops after an optimal subset is found. Consistency measure [44] and correlation measure [45],[46] are two examples of subset search algorithms. In contrast to the dimensionality reduction techniques such as PCA, in which the data is projected on to a subspace, feature selection methods do not alter the dimensionality but selects useful subset of the data.

In this thesis, Chapter 3, Chapter 4 can be considered as subset search algorithms whereas

methods in Chapter 5 can be considered as feature weighing algorithm.

## 2.6 Proposed Feature Selection Methods

The data used in our experiments in the later chapters are one-dimensional. They are concatenated histograms. Feature selection is performed on the data to prune or weigh the features based on the algorithms that are explained in the next three chapters. Three feature selection methods proposed in this thesis are: 1. Genuine segment score thresholding 2.  $d'$  thresholding, and 3. Adaboost inspired methods: Method 1 and Method 2. The first two methods are used to prune the data. Unwanted data are discarded. The Adaboost inspired methods emphasize on weighing the features based on the information provided by them. To select informative portions of the histogram, it is partitioned into  $n$  nonoverlapping equal in length segments. Feature selection is performed with varying number of segments: 256, 768, and 1280.

## Chapter 3

# Genuine Segment Score Thresholding

### 3.1 Motivation

Genuine segment score thresholding operates on the segment scores calculated as discussed in the previous section. The algorithm is designed to discard all the segments which are considered non informative based on a threshold. Only genuine scores are used for the thresholding method. Since distance measure is used, ideal genuine segment scores have to be low and impostor segment scores have to be high. But in practice, some of the genuine segments have higher values and the impostor segments have lower values due to noise. This method focuses on discarding the noise and retaining the useful segments while calculating a match score.

The algorithm requires training with a set of genuine segment scores. A binary mask is obtained during the training phase, in which value 1 represents informative segment and value 0 represents a non-informative segment. Value of the binary mask is based on the genuine segment score and the threshold whose value is a design parameter and can be set to any value in the range of the genuine segment scores obtained using the training set.

Assume that histograms are partitioned into  $n$  segments. Let  $m_i = 1, \dots, n$  be the value of the mask at the location of the  $i^{th}$  segment. Let  $\tau$  be the threshold and  $g_i$  denotes the genuine segment score for the  $i^{th}$  segment. Then the bits in each mask are assigned.

The first rule of selecting informative segments is as follows:

Set

$$\begin{aligned}
m_i &= 1, \text{ if } g_i > \tau, i = 1, \dots, n \\
&= 0, \text{ otherwise}
\end{aligned}$$

The second rule of selecting informative segments is as follows:

Set

$$\begin{aligned}
m_i &= 1, \text{ if } g_i < \tau, i = 1, \dots, n \\
&= 0, \text{ otherwise}
\end{aligned}$$

This method can be used in two modes: 1. Verification 2. Identification.

## 3.2 Verification

Verification is a one to one match between the probe and gallery to verify if the user is who he claims to be. Each user in the gallery has a specific mask designed from training data. Each user in the database has 4 images captured at visible spectrum and 4 images captured at 1550 nm, which also include different expressions. Training set is formed to include half of the images in visible spectrum and half of the images from 1550nm. The remaining data are used for performance testing.

Genuine segment scores of the training set are calculated. A histogram of genuine segment scores is plotted. Each user has 4 vectors of segment scores. Each vector has  $n$  components, which is equal to the number of segments. A mean genuine segment score is computed for each user. This mean segment score is used to find a unique mask for each class. Threshold range is the range of the training genuine segment scores. For each value of threshold, mask is obtained by comparing  $g_i$  with the threshold as explained above. User mask is created for each threshold value.

To compute the match score, final mask is obtained as an “or” operation between the two masks. Match score is obtained as the dot product of the final mask and the score matrix for the two templates. For each threshold value the performance using the feature selection method is compared with the original performance. The thresholds that result in

performance improvement are noted and are compared with the range of thresholds obtained on training set.

### 3.2.1 Results for Verification

Each user has 8 images, 4 captured at visible spectrum and 4 captured at 1550nm. Half of the data (2 images at visible spectrum and 2 images at 1550nm) is used for training and the other half is used for testing. Three sets of training data are considered. The first training set includes the images with neutral expression of all the users captured at visible spectrum and 1550 nm. A unique mask is derived for each user using the training set. This mask is applied to the testing set which contains the images with talking expressions in the first case.

In the second case, training set includes all the images with talking expression and training set includes all the images with neutral expression. The third case combines both the first and second case, in which the training set includes one image of neutral expression and one image of talking expression for each spectrum. Testing set includes the images that are not used for training.

Results in this section follow the second case in which talking images are used for training and neutral images are used for testing. Fig. 3.1 shows the histogram of genuine segment scores used for training with  $n=256$ . Threshold can be set to any value in the range of genuine segment scores shown. Fig. 3.2 shows the performance of training set at different thresholds when 256 segments are considered. Improvement in performance is measured based on the Equal Error Rate (EER). Fig. 3.3 shows one threshold value at which improvement of performance is observed. The same experiment is conducted by changing the training and testing sets and the rule of selecting informative segments. Values of thresholds resulting in performance improvement are specified in the 2<sup>nd</sup> column of Tables 3.1-3.6. Values of thresholds resulting in perfect performance are specified in the 3<sup>rd</sup> column of Tables 3.1-3.6. It can be observed from the tables that the range of values in training and testing overlap.

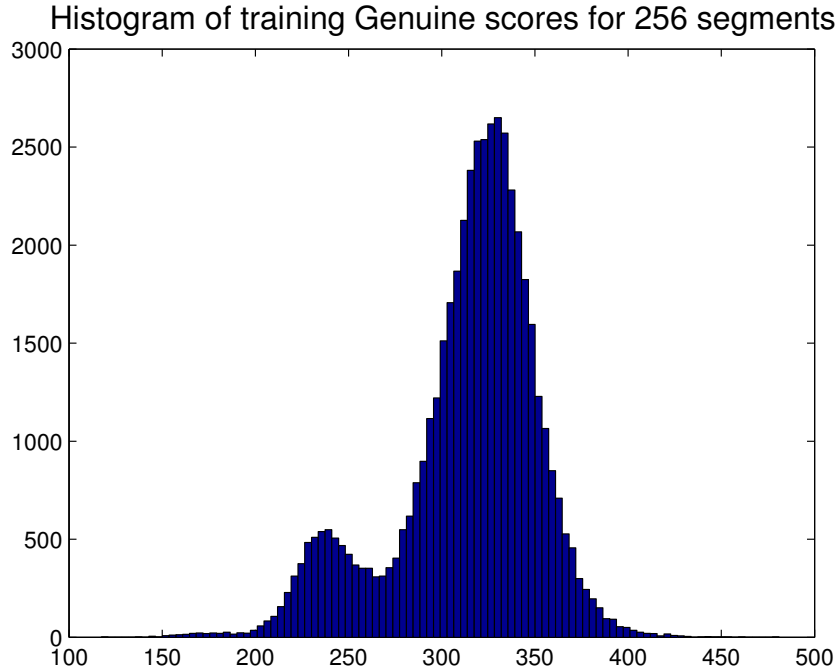


Figure 3.1: Histogram of Genuine segment scores of training set considering 256 segments of the template.

### 3.3 Identification

Identification is a one to many comparison of the query with the gallery templates to find the identity of a person. The algorithm is trained on a training set, resulting in a single mask that can be applied to any image. Training set contains all images of 24 classes selected at random. Each comparison has  $n$  segment scores. A mean genuine segment score set is formed from all the genuine comparisons in the training set. The above algorithm is implemented with varying values of the threshold, resulting in different masks. The final masks obtained at various thresholds are used for every comparison. Confusion matrix is used as a measure to see any performance improvement. In the testing phase, all the images of the classes not included in training set are used along with the mask found during the training phase. Range of thresholds resulting in improved performance is compared in training and testing phases. Matching score between two templates is computed as the dot product of the final mask and individual segment scores.

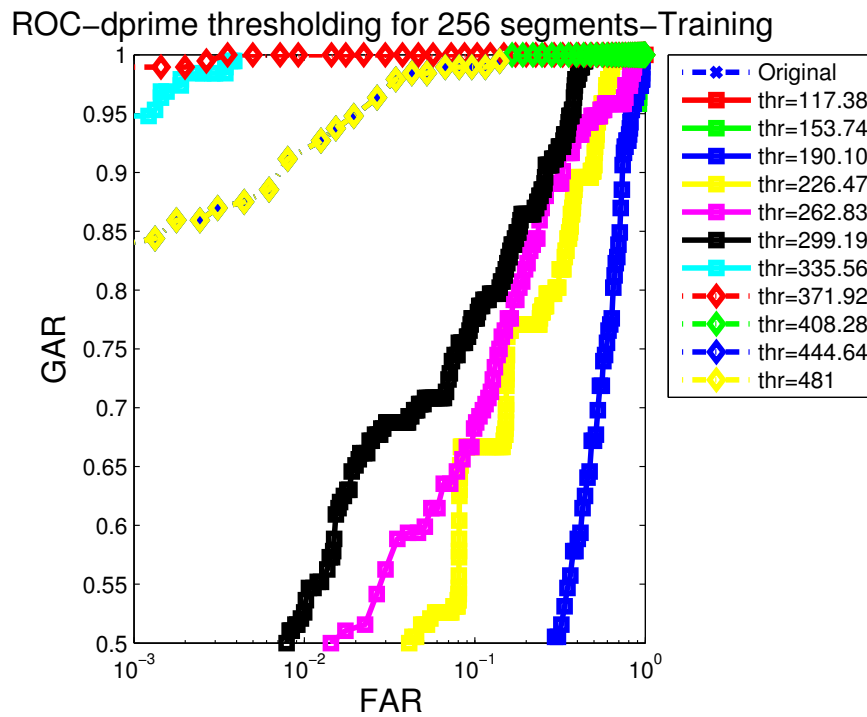


Figure 3.2: Plot showing the ROCs of genuine segment scores thresholded at different thresholds. It can be observed that removal of some segments at some threshold values result in improvement of performance. This is a plot considering 256 segments and the training set being the talking images and the mask assignment based on the second rule of selecting informative segments.

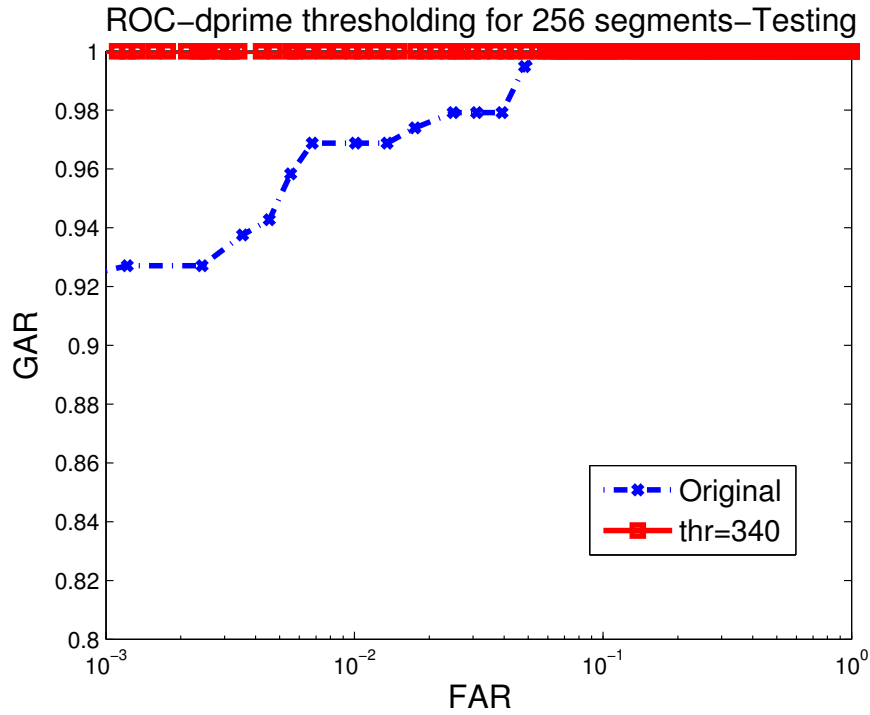


Figure 3.3: ROC showing the perfect performance on testing set at  $\tau = 340$ .

### 3.3.1 Results for Identification

Experiments are conducted with training set in which 24 classes are selected at random. All images of the selected classes are used for training. Testing set includes all the classes that are not used for testing. The same procedure is followed by changing the training and testing sets. Each trial results in an identification rate as explained below. An average of all the trials is considered. The 3<sup>rd</sup> column of Table 3.7 and Table 3.8 specifies the average identification rate without applying the feature selection method. The 4<sup>th</sup> column of Table 3.7 and Table 3.8 denotes the average identification rate obtained at different thresholds specified in the 2<sup>nd</sup> column. All images captured at visible spectrum are considered as gallery images. All images captured at 1550 nm are considered as query images. A confusion matrix is computed based on the first 4 ranks (since each user has 4 images captured at 1550nm). Identification rate is a measure that indicates how many classes are accurately identified. If  $C$  is the confusion matrix, identification rate is computed as follows:

$$\text{Identification Rate} = \frac{\text{sum}(\text{diag}(C))}{\text{sum}(C)} \times 100 \quad (3.1)$$



	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $g > \tau$	(167-169),(181-203),(212-236), (251-257)	-
256 segments: $g < \tau$	324-401	331-378.3
768 segments: $g > \tau$	(52-55),(73-212),(312-327), (384-420),(453-480),(506-513)	-
768 segments: $g < \tau$	(236-313),(444-633),(645-688)	(241-270),(461-487),(532-607)
1280 segments: $g > \tau$	(43-96),(126-261),(306-507)	384-412
1280 segments: $g < \tau$	(173-247),(348-445),(471-637), (646-686)	(180-226),(360-415),(499-583)

Table 3.1: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions.

The same procedure is followed for different number of segments and with different rules of selecting informative segments.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $g > \tau$	(166-173),(178-278)	-
256 segments: $g < \tau$	327-425	341-363
768 segments: $g > \tau$	(69-222), (265-549)	-
768 segments: $g < \tau$	(236-270),(305-315),(444-662)	(243-260),(575-578)
1280 segments: $g > \tau$	(65-282),(292-533)	353-422
1280 segments: $g < \tau$	(173-240),(348-630)	(188-197),(370-393),(507-578)

Table 3.2: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions. Masks used in this case are computed using the training set of images with neutral expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $g > \tau$	(156-241),(282-285)	-
256 segments: $g < \tau$	324-406	339-366
768 segments: $g > \tau$	(71-303),(365-555)	-
768 segments: $g < \tau$	(234-342),(353-380),(406-653)	(248-265),(456-492),(511-591)
1280 segments: $g > \tau$	(103-264),(293-547)	(417-427),(454-472)
1280 segments: $g < \tau$	(166-257),(320-630)	(377-431),(474-574)

Table 3.3: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $g > \tau$	(156-176),(207-243),(262-287)	-
256 segments: $g < \tau$	327-422	341-363
768 segments: $g > \tau$	(78-234),(272-298),(344-545)	415-442
768 segments: $g < \tau$	(233-325),(432-665)	456-596
1280 segments: $g > \tau$	(42-92),(137-257),(282-535)	352-479
1280 segments: $g < \tau$	(175-253),(349-640)	(191-212), (377-413),(485-583)

Table 3.4: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions. Masks used in this case are computed using the training set of images with talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $g > \tau$	(164-243),(270-275)	-
256 segments: $g < \tau$	324-400	338-360
768 segments: $g > \tau$	(75-229),(266-316),(376-509)	-
768 segments: $g < \tau$	(236-316),(435-668)	(240-268),(454-479),(488-502), (520-598)
1280 segments: $g > \tau$	(61-81),(137-262),(315-516)	(366-430),(440-465)
1280 segments: $g < \tau$	(173-250),(313-325),(347-670)	(185-205),(369-421),(481-584)

Table 3.5: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral and talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $g > \tau$	(174-207),(216-244),(255-274)	-
256 segments: $g < \tau$	323-416	339-354
768 segments: $g > \tau$	(48-55),(75-131),(192-239), (266-344),(362-558)	-
768 segments: $g < \tau$	(229-319),(380-387),(421-640)	(243-264),(454-470),(545-594)
1280 segments: $g > \tau$	(44-81),(141-274),(300-548)	(391-413),(442-446),(484-490)
1280 segments: $g < \tau$	(171-247),(305-633)	(185-203),(374-414),(486-569)

Table 3.6: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral and talking expressions. Masks used in this case are computed using the training set of images with neutral and talking expressions.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using genuine thresholding
256 segments: $g > \tau$	(211-235),(246-257)	97.9688	97.9998
256 segments: $g < \tau$	307-360	98.0208	98.1748
768 segments: $g > \tau$	108-141	96.8750	96.9271
768 segments: $g < \tau$	(546-552),(635-644)	97.3958	97.4219
1280 segments: $g > \tau$	-	96.7969	96.7969
1280 segments: $g < \tau$	503-614	97.3958	97.4803

Table 3.7: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3<sup>rd</sup> column represents the original identification rate of the training set computed as explained above. The 4<sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2<sup>nd</sup> column. Maximum identification rate is mentioned in the 4<sup>th</sup> column for all the comparisons that do not result in performance improvement.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using genuine thresholding
256 segments: $g > \tau$	180-188	97.6302	97.6563
256 segments: $g < \tau$	-	97.2135	97.3099
768 segments: $g > \tau$	106-146	96.9010	96.9776
768 segments: $g < \tau$	581-635	96.5365	96.5817
1280 segments: $g > \tau$	71-167	96.9531	97.0262
1280 segments: $g < \tau$	-	96.2500	96.2500

Table 3.8: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3<sup>rd</sup> column represents the original identification rate of the testing set computed as explained above. The 4<sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2<sup>nd</sup> column. Maximum identification rate is mentioned in the 4<sup>th</sup> column for all the comparisons that do not result in performance improvement.

# Chapter 4

## $d'$ -based Thresholding

### 4.1 Motivation

D-prime thresholding operates on the  $d'$  values calculated for each segment based on the training set. The difference between this method and the genuine segment score thresholding is that this method involves the impostor segment scores together with genuine segment scores.  $d'$  is a statistical measure of how well a system can distinguish between two probability density functions based on their two first moments, mean and variance, where  $\mu_g$  is the mean of genuine scores,  $\mu_i$  is the mean of impostor scores,  $\sigma_g^2$  is variance of genuine scores and  $\sigma_i^2$  is variance of impostor scores.  $d'$  in biometrics is defined as [47]

$$d' = \frac{|\mu_g - \mu_i|}{\sqrt{\sigma_g^2 + \sigma_i^2}}. \quad (4.1)$$

The larger the  $d'$  value is, the better the performance of the biometric system in recognizing individuals is. The algorithm suggests discarding the segments with  $d'$  values greater or smaller than a threshold. Threshold can take values within the  $d'$  range of the training set. The algorithm requires training using  $d'$  segment values calculated from a set of genuine and impostor segment scores. A binary mask is obtained during the training phase, in which value 1 represents informative segment and value 0 represents a non-informative segment. An alternative dimensionality reduction method based on d-prime keeps  $d'$  with values below threshold. Value of the binary mask is based on the  $d'$  segment value and the threshold whose range is the range of the training set. The first rule of selecting informative segments

is as follows:

Set

$$\begin{aligned} m_i &= 1, \text{ if } d' > \tau, i = 1, \dots, n \\ &= 0, \text{ otherwise} \end{aligned}$$

The second rule of selecting informative segments is as follows: Set

$$\begin{aligned} m_i &= 1, \text{ if } d' < \tau, i = 1, \dots, n \\ &= 0, \text{ otherwise} \end{aligned}$$

This method is operated in two modes. 1. Verification 2. Identification.

## 4.2 Verification

Verification is a one to one match between the probe and gallery to verify if the user is who he claims to be. Training set is selected the same way as explained in 3.2. Each user in the gallery has a specific mask. Genuine and impostor segment scores of the training set are calculated.  $d'$  values for each user are calculated using the class genuine and impostor segment scores. Histogram of  $d'$  values is plotted.  $d'$  values obtained are used to find a unique mask for each class. Threshold range is the range of  $d'$  values of the entire training set. For each value of threshold, mask is obtained by comparing  $d'$  with the threshold as explained in the previous algorithm. Each threshold value has a mask for each user. Comparison can be greater or lower than the threshold.

To compute the match score, final mask is obtained as an “or” operation between the two masks. Match score is obtained as the dot product of final mask and segment scores between the two templates. For each threshold value the performance using the feature selection method is compared with the original performance. The entire threshold range that result in performance improvement is noted and is compared with the range of thresholds obtained on training set.

### 4.2.1 Results for Verification

Each user has 8 images, 4 captured at visible spectrum and 4 captured at 1550nm. Half of the data (2 images at visible spectrum and 2 images at 1550nm) is used for training and the other half is used for testing. Three sets of training data are considered. The first training set includes the images with neutral expression of all the users captured at visible spectrum and 1550 nm. A unique mask is derived for each user using the training set. This mask is applied to the testing set which contains the images with talking expressions in the first case. In the second case, training set includes all the images with talking expression and training set includes all the images with neutral expression. The third case combines both the first and second case, in which the training set includes one image of neutral expression and one image of talking expression for each spectrum. Testing set includes the images that are not used for training. Results in this section follow the second case in which neutral images are used for training and talking images are used for testing. Fig. 4.1 shows the histogram of  $d'$ -prime values used for training for 256 segments. Threshold can be set to any value in the range of  $d'$ -prime segment scores shown. Fig. 4.2 shows the performance of training set at different thresholds when 256 segments are considered. Improvement in performance is measured based on the EER. Fig. 4.3 shows one threshold value at which improvement of performance is observed. The same experiment is conducted by changing the training and testing sets and the rule of selecting informative segments. Values of thresholds resulting in improvement in performance are specified in the 2<sup>nd</sup> column of Tables 4.1- 4.6. Values of thresholds resulting perfect performance are specified in the 3<sup>rd</sup> column of Tables 4.1- 4.6. It can be observed from the tables that the range of values in training and testing overlap.

## 4.3 Identification

Identification is a one to many comparison of the query with the gallery templates to find the identity of a person. The algorithm is trained on a training set, resulting in a single mask that can be applied to any image. Training set contains all images of randomly selected classes. Each comparison has  $n$  segment scores. A mean genuine and impostor segment score



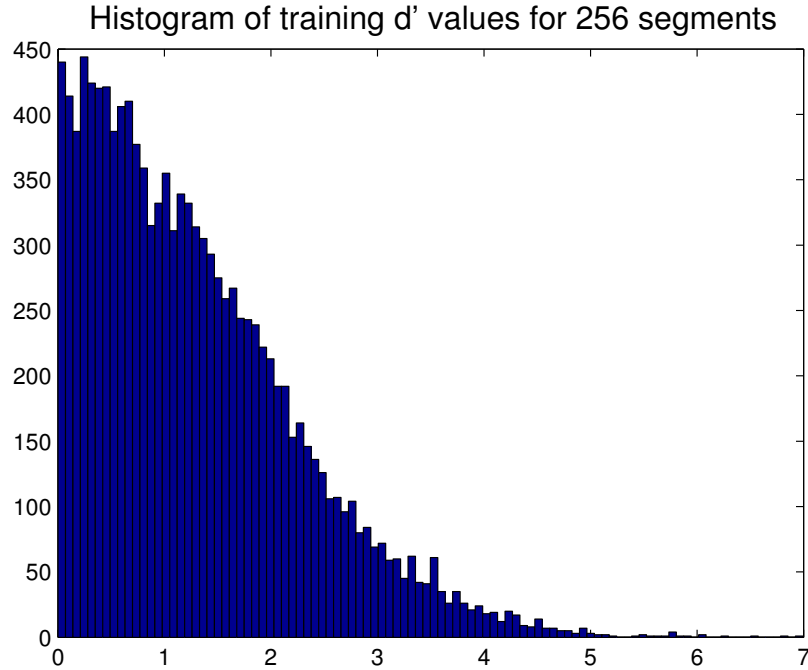


Figure 4.1: Histogram of  $d'$  segment values for the training data partitioned into 256 segments.

set is formed from all the genuine and impostor comparisons in the training set. The above algorithm is implemented with varying thresholds, resulting in different masks. The final masks obtained at various thresholds are used for every comparison. Confusion matrix is used as a measure to see any performance improvement. In the testing phase, all the images of other classes are used along with the mask found during the training phase. Matching score between two templates is computed as the dot product of the final mask and individual segment scores.

### 4.3.1 Results for Identification

Experiments are conducted with training set in which 24 classes are selected at random. All images of the selected classes are used for training. Testing set includes all the classes that are not used for testing. The same procedure is followed by changing the training and testing sets. Each trial results in an identification rate as explained below. An average of all the trials is considered. The 3<sup>rd</sup> column of Table 4.7 and Table 4.8 specifies the average

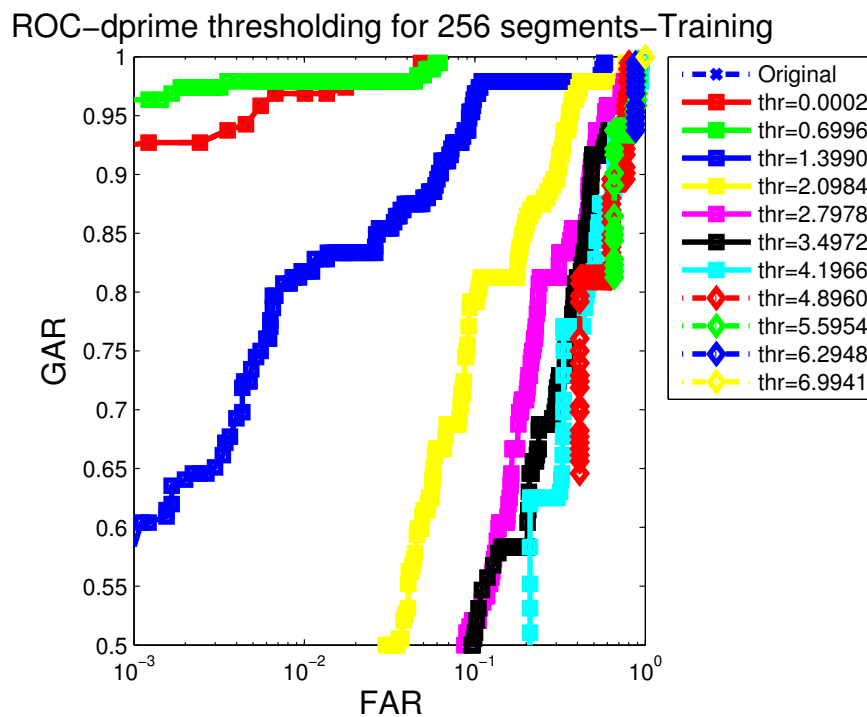


Figure 4.2: Plot showing the ROCs of  $d'$  thresholding at different values of thresholds. It can be observed that by discarding segments at some threshold values we achieve improved performance. This is a plot considering 256 segments per histogram with the training set being the neutral images and with the segment selection based on the first rule of selecting informative segments.

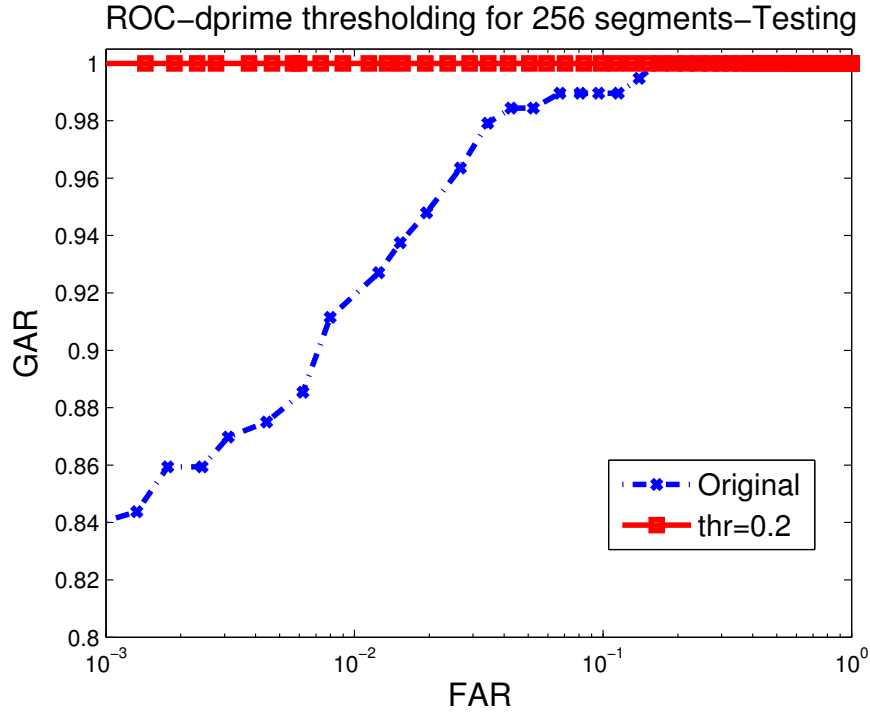


Figure 4.3: ROC showing the perfect performance on testing set at  $\tau = 0.2$ .

identification rate without applying the feature selection method. The 4<sup>th</sup> column of Table 4.7 and Table 4.8 denotes the average identification rate obtained at different thresholds specified in the 2<sup>nd</sup> column. All images captured at visible spectrum are considered as gallery images. All images captured at 1550 nm are considered as query images. A confusion matrix is computed based on the first 4 ranks (since each user has 4 images captured at 1550nm). Identification rate is a measure that indicates how many classes are accurately identified. If  $C$  is the confusion matrix, identification rate is computed as follows:

$$\text{Identification Rate} = \frac{\text{sum}(\text{diag}(C))}{\text{sum}(C)} \times 100. \quad (4.2)$$

The same procedure is followed for different number of segments and with different rules of selecting informative segments.

## 4.4 Modified $d'$

The feature selection method is also applied on the parameter defined by

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $d' > \tau$	0.023-1.0260	0.0702-0.4665
256 segments: $d' < \tau$	3.0775-3.7070	-
768 segments: $d' > \tau$	0.0262-0.7609	0.0525-0.6297
768 segments: $d' < \tau$	0.1312-4.3817	-
1280 segments: $d' > \tau$	0.0264-0.7387	0.0264-0.6859
1280 segments: $d' < \tau$	0.0528-4.6168	0.2638-0.4221

Table 4.1: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $d' > \tau$	0.0235-0.9094	0.0702-0.3732
256 segments: $d' < \tau$	2.8910-6.9941	-
768 segments: $d' > \tau$	0.0262-0.8921	0.0262-0.5510
768 segments: $d' < \tau$	0.0787-5.7460	-
1280 segments: $d' > \tau$	0.0264-1.0289	0.0264-0.6332
1280 segments: $d' < \tau$	0.0264-5.0125	0.2638-0.5540

Table 4.2: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions. Masks used in this case are computed using the training set of images with neutral expressions.

$$md' = \frac{\mu_g - \mu_i}{\sqrt{\sigma_g^2 + \sigma_i^2}}. \quad (4.3)$$

This is a modified d-prime measure with the direct difference between the means.

#### 4.4.1 Results for verification

Experiments conducted are the same as in Section 4.2. Results in this section follow the second case in which neutral images are used for training and talking images are used for testing. Fig. 4.4 shows the histogram of modified d-prime values used for training for 256

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $d' > \tau$	0-0.8952	0.0230-0.3902
256 segments: $d' < \tau$	2.2035-6.3580	-
768 segments: $d' > \tau$	0.0191-0.9320	0.0191-0.4945
768 segments: $d' < \tau$	0.0951-4.6027	-
1280 segments: $d' > \tau$	0-0.9938	0.0188-0.5625
1280 segments: $d' < \tau$	0.0375-4.7438	-

Table 4.3: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $d' > \tau$	0-0.8263	0.0230-0.4361
256 segments: $d' < \tau$	2.1576-4.9578	-
768 segments: $d' > \tau$	0.0191-0.6657	0.0191-0.5136
768 segments: $d' < \tau$	0.1142-5.2113	-
1280 segments: $d' > \tau$	0.0188-0.6938	0.0188-0.6
1280 segments: $d' < \tau$	0.0563-4.0688	0.3750-2.0813

Table 4.4: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions. Masks used in this case are computed using the training set of images with talking expressions.

segments. Threshold can be set to any value in the range of  $d'$ -prime segment scores shown. Fig. 4.5 shows the performance of training set at different thresholds when 256 segments are considered. Fig. 4.6 shows one threshold value at which improvement of performance is observed. The same experiment is conducted by changing the training and testing sets and the rule of selecting informative segments. Values of thresholds resulting in improved performance are specified in the 2<sup>nd</sup> column of Tables 4.9- 4.14. Values of thresholds resulting in perfect performance are specified in the 3<sup>rd</sup> column of Table 4.9- 4.14. It can be observed from the tables that the range of values in training and testing overlap.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $d' > \tau$	0.0220-0.7930	0.0441-0.3965
256 segments: $d' < \tau$	2.7094-4.7139	-
768 segments: $d' > \tau$	0-0.6415	0.0377-0.5471
768 segments: $d' < \tau$	0.0943-5.0751	-
1280 segments: $d' > \tau$	0-0.7488	0.0170-0.5956
1280 segments: $d' < \tau$	0.0681-5.1055	0.1362-0.2042

Table 4.5: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral and talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $d' > \tau$	0-0.9031	0.0661-0.3745
256 segments: $d' < \tau$	2.2468-5.7933	-
768 segments: $d' > \tau$	0.0189-0.8867	0.0377-0.5094
768 segments: $d' < \tau$	0.0566-5.0751	-
1280 segments: $d' > \tau$	0.0170-1.0551	0.0340-0.5610
1280 segments: $d' < \tau$	0.0340-4.6290	0.1021-0.2042

Table 4.6: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral and talking expressions. Masks used in this case are computed using the training set of images with neutral and talking expressions .

#### 4.4.2 Results for Identification

Experiments conducted are the same as in Section 4.3. The 3<sup>rd</sup> column of Table 4.15 and Table 4.16 specifies the average identification rate without applying the feature selection method. The 4<sup>th</sup> column of Table 4.15 and Table 4.16 denotes the average identification rate obtained at different thresholds specified in the 2<sup>nd</sup> column.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using $d'$ thresholding
256 segments: $d' > \tau$	0.0532-2.7317	97.0313	98.2499
256 segments: $d' < \tau$	-	97.8385	97.8385
768 segments: $d' > \tau$	0.0351-2.5228	97.0313	98.6566
768 segments: $d' < \tau$	-	96.4063	96.4063
1280 segments: $d' > \tau$	0.0311-2.1876	97.0052	98.4533
1280 segments: $d' < \tau$	-	96.9010	96.9010

Table 4.7: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3<sup>rd</sup> column represents the original identification rate of the training set computed as explained above. The 4<sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2<sup>nd</sup> column. Maximum identification rate is mentioned in the 4<sup>th</sup> column for all the comparisons that do not result in performance improvement.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using $d'$ thresholding
256 segments: $d' > \tau$	0.0710-2.4479	97.9167	98.7591
256 segments: $d' < \tau$	-	97.5260	97.5260
768 segments: $d' > \tau$	0.0175-2.5929	96.7448	98.2413
768 segments: $d' < \tau$	-	97.7865	97.7865
1280 segments: $d' > \tau$	0.0155-2.0014	96.7969	98.3846
1280 segments: $d' < \tau$	-	96.9010	96.9010

Table 4.8: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3<sup>rd</sup> column represents the original identification rate of the testing set computed as explained above. The 4<sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2<sup>nd</sup> column. Maximum identification rate is mentioned in the 4<sup>th</sup> column for all the comparisons that do not result in performance improvement.

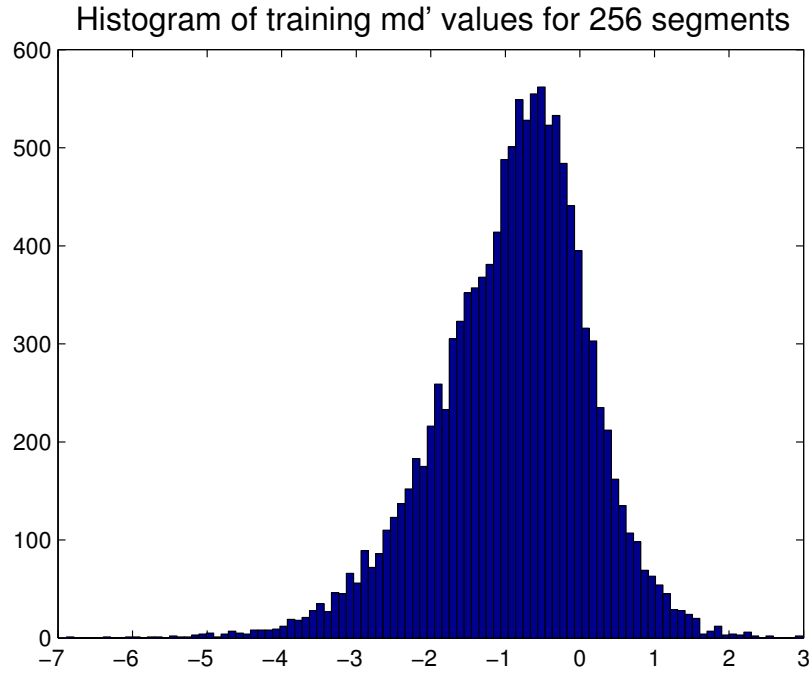


Figure 4.4: Histogram of  $md'$  segment values for the training data partitioned into 256 segments.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $md' > \tau$	-3.6946 to -3.0760	-
256 segments: $md' < \tau$	-0.8763 to 2.8357	-0.1889 to 1.3234
768 segments: $md' > \tau$	-4.3789 to -2.3670	-
768 segments: $md' < \tau$	-0.4690 to 3.5169	-0.2792 to 1.5050
1280 segments: $md' > \tau$	-4.3277 to -1.9365	-
1280 segments: $md' < \tau$	-0.5018 to 3.3639	-0.3823 to 1.4908

Table 4.9: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions.



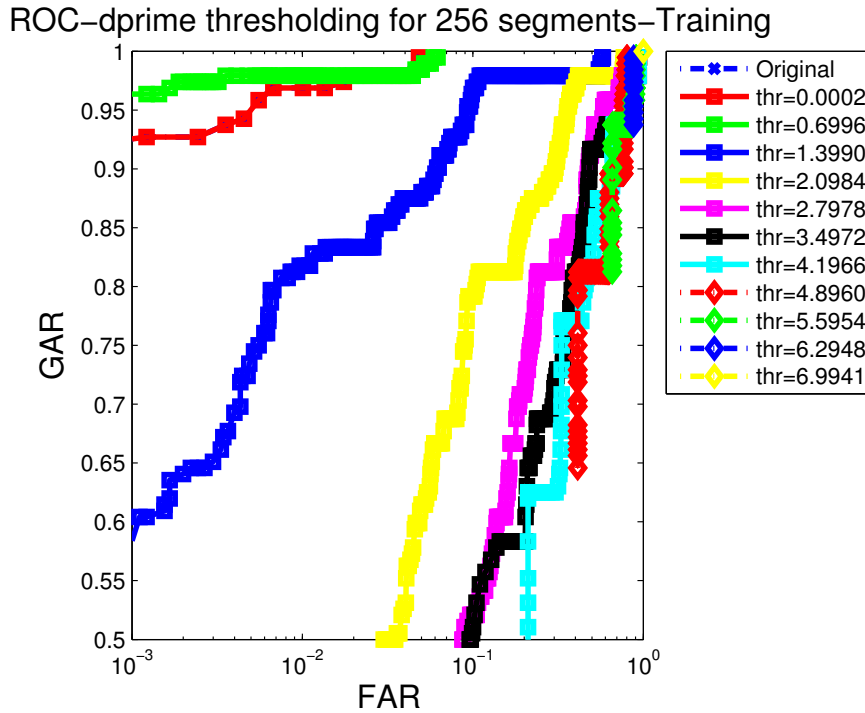


Figure 4.5: Plot showing the ROCs of  $md'$  thresholding at different values of thresholds. It can be observed that by discarding segments at some threshold values we achieve improved performance. This is a plot considering 256 segments per histogram with the training set being the talking images and with the segment selection based on the second rule of selecting informative segments.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $md' > \tau$	-6.9941 to -2.9729	-
256 segments: $md' < \tau$	-0.7388 to 3.1794	0.0861 to 1.1172
768 segments: $md' > \tau$	-5.7455 to -1.9115	-
768 segments: $md' < \tau$	-0.5449 to 2.6438	-0.2033 to 1.2013
1280 segments: $md' > \tau$	-5.0052 to 0.1757	-
1280 segments: $md' < \tau$	-0.7410 to 3.3240	-0.2229 to 1.3314

Table 4.10: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions. Masks used in this case are computed using the training set of images with neutral expressions.

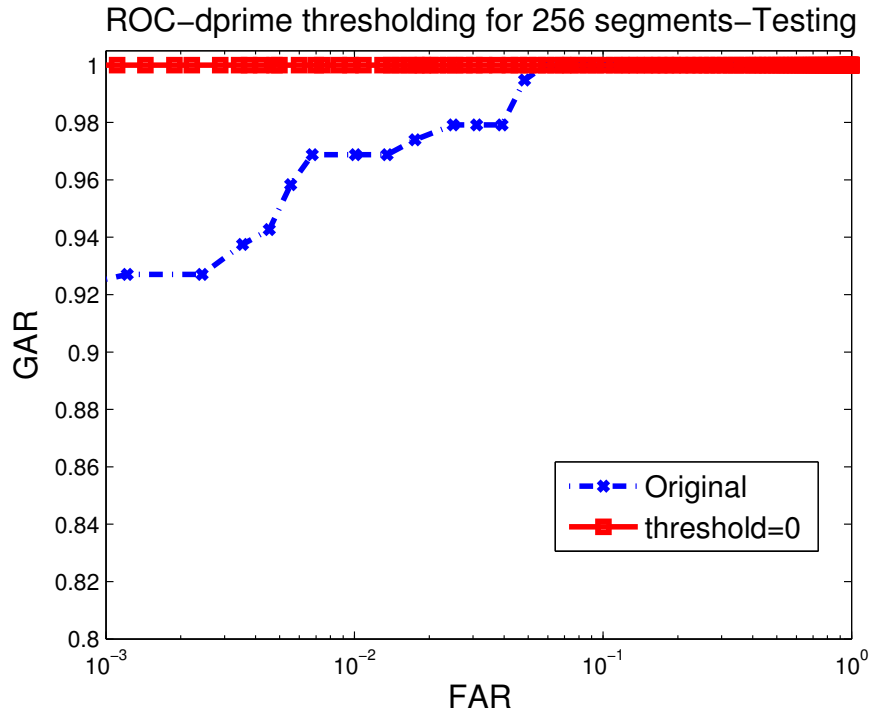


Figure 4.6: ROC showing the perfect performance on testing set at  $\tau = 0$ .

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $md' > \tau$	-6.3592 to -2.2113	-
256 segments: $md' < \tau$	-0.6641 to 2.9899	-0.0716 to 0.5539
768 segments: $md' > \tau$	-4.6149 to -0.5940	-
768 segments: $md' < \tau$	-0.7810 to 3.0529	-0.2199 to 1.2139
1280 segments: $md' > \tau$	-4.7269 to -0.2060	-
1280 segments: $md' < \tau$	-0.8 to 3.0	-0.3056 to 1.5243

Table 4.11: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $md' > \tau$	-4.9437 to -2.1784	-
256 segments: $md' < \tau$	-0.5654 to 2.8912	-0.1045 to 0.4222
768 segments: $md' > \tau$	-5.2071 to -1.6849	-
768 segments: $md' < \tau$	-0.4381 to 3.6451	-0.2511 to 1.0268
1280 segments: $md' > \tau$	-4.0682 to -0.2360	-2.0922 to -1.7329
1280 segments: $md' < \tau$	-0.4062 to -0.1736	-0.3625 to 1.3927

Table 4.12: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions. Masks used in this case are computed using the training set of images with talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $md' > \tau$	-4.7137 to -2.7548	-
256 segments: $md' < \tau$	-0.6675 to 1.8373	-0.1858 to 0.4886
768 segments: $md' > \tau$	-5.0538 to -1.9018	-
768 segments: $md' < \tau$	-0.3560 to 3.4325	-0.2045 to 0.9775
1280 segments: $md' > \tau$	-5.1055 to -0.0185	-
1280 segments: $md' < \tau$	-0.2127 to 4.2342	-0.2761 to 1.5637

Table 4.13: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral and talking expressions.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments: $md' > \tau$	-5.7734 to 2.4337	-
256 segments: $md' < \tau$	-0.6996 to 2.5759	-0.0252 to 0.4886
768 segments: $md' > \tau$	-5.0538 to -0.1136	-
768 segments: $md' < \tau$	-0.7197 to 2.7960	-0.2045 to 0.9472
1280 segments: $md' > \tau$	-4.6152 to 0.4105	-
1280 segments: $md' < \tau$	-0.7623 to 2.9434	-0.4231 to 1.3432

Table 4.14: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral and talking expressions. Masks used in this case are computed using the training set of images with neutral and talking expressions.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using $md'$ thresholding
256 segments: $md' > \tau$	-	97.2135	97.2135
256 segments: $md' < \tau$	-2.9295 to 0.6274	97.3438	98.4496
768 segments: $md' > \tau$	-	97.3438	97.3438
768 segments: $md' < \tau$	-2.3185 to 1.3326	96.8750	98.0524
1280 segments: $md' > \tau$	-	96.7188	96.7188
1280 segments: $md' < \tau$	-2.3006 to 0.8331	97.0573	98.1891

Table 4.15: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3<sup>rd</sup> column represents the original identification rate of the training set computed as explained above. The 4<sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2<sup>nd</sup> column. Maximum identification rate is mentioned in the 4<sup>th</sup> column for all the comparisons that do not result in performance improvement.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using $md'$ thresholding
256 segments: $md' > \tau$	-	97.5521	97.5521
256 segments: $md' < \tau$	-2.7813 to 0.0345	97.5781	98.3412
768 segments: $md' > \tau$	-	96.5625	96.5625
768 segments: $md' < \tau$	-2.1636 to 0.0713	97.2396	98.5128
1280 segments: $md' > \tau$	-	96.8490	96.8490
1280 segments: $md' < \tau$	-2.2763 to 0.2015	96.4844	97.9468

Table 4.16: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column summarizes the range of thresholds, which result in improved performance compared to the original performance. The 3<sup>rd</sup> column represents the original identification rate of the testing set computed as explained above. The 4<sup>th</sup> column represents the average identification rate obtained at each threshold value mentioned in the 2<sup>nd</sup> column. Maximum identification rate is mentioned in the 4<sup>th</sup> column for all the comparisons that do not result in performance improvement.

# Chapter 5

## Adaboost Inspired Methods

### 5.1 Adaboost

Adaboost [48],[49] is an algorithm for building a strong classifier by linearly combining a set of weak classifiers. The decision of the classifier is based on the sum of weighted outputs of all the weak classifiers. The classifier output is weighed based on the error produced by the classifier evaluated on the training set. Patterns correctly classified have lower weights than misclassified patterns. The steps of an Adaboost algorithm are summarized below. The algorithm considers a training set  $(x^i, y_i), \dots, (x^m, y_m)$  where  $x^i$  is a pattern which belongs to domain  $X$ ,  $y_i$  is the class label of  $x^i$ .  $Y$  contains all the class labels and  $y_i$  takes one value in  $Y$ . For a two class problem, we assume,  $Y = \{-1, +1\}$ . Adaboost is an iterative algorithm. The value of  $k$  denotes the number of iterations. Weight of the distribution of a sample  $i$  at iteration  $k$  is denoted by  $W_k(i)$ . Initially all the weights are set equal, but on each iteration weights of incorrectly classified patterns increase so that the weak learner is focused to learn on incorrectly classified patterns. The task of the weak classifiers,  $C_k$ , task is to find a weak hypothesis  $h_k$  appropriate for the distribution  $W_k$ . Weak hypothesis is measured by its error  $E_k$ .  $Z_k$  is a normalizing constant to ensure that  $W_k$  is a distribution.  $\alpha_k$  intuitively measures the importance of  $h_k$ .

#### 5.1.1 Algorithm

1 begin initialization  $D = \{x^1, y_1, x^2, y_2, \dots, x^m, y_m\}, k_{max}, W_1(i) = \frac{1}{m}, i = 1, \dots, m$

```

2       $k \leftarrow 0$ 
3      do  $k \leftarrow k + 1$ 
4          Train weak learner  $C_k$  using  $D$  sampled according to distribution  $W_k(i)$ 
5           $E_k \leftarrow$  Training error of  $C_k$  measured on  $D$  using  $W_k(i)$ 
6           $\alpha_k \leftarrow \frac{1}{2} \ln[(1 - E_k)/E_k]$ 
7           $W_{k+1}(i) \leftarrow \frac{W_k(i)}{Z_k} \times \begin{cases} e^{-\alpha_k}, & \text{if } h(x^i) = y_i \text{ (correctly classified)} \\ e^{\alpha_k}, & \text{if } h(x^i) \neq y_i \text{ (incorrectly classified)} \end{cases}$ 
8      until  $k = k_{max}$ 
9      return  $C_k$  and  $\alpha_k$  for  $k = 1$  to  $k_{max}$  (ensemble of classifiers with weights)
10 end

```

Output of the final classifier is

$$H(x) = \text{sign} \left( \sum_{k=1}^T \alpha_k h_k(x) \right), \quad (5.1)$$

where  $T$  is the number of weak learners.

The main principles of this algorithm can be used to improve performance of the face classifier by weighing the segments based on the information provided by that segment. The basic Adaboost algorithm can be reconstructed in a way that the segments can be weighed based on the information. Two dimensionality reduction algorithms are proposed based on the basic Adaboost algorithm.

## 5.2 Method 1

This method focuses on minimizing the overlap between genuine and impostor scores of a training set. These genuine and impostor scores are computed as the weighted sum of segment scores. Weight for each segment is computed based on its location in the distribution of genuine and impostor scores.

### 5.2.1 Algorithm

Each feature vector in the training set is divided into  $m$  segments. Histogram of genuine and impostor matching scores is plotted.

Step 1: Maximum value of genuine ( $max_g$ ) and minimum value of impostor scores ( $min_i$ ) are stored as the initial upper and lower thresholds for each user. The goal of this algorithm is to minimize the overlap between genuine and impostor matching scores.

Step 2: Initialize segment weights. Let the weights be  $\alpha_i$ ,  $i = 1, \dots, m$ . Each segment has the same weight equal to  $1/m$  on the first iteration:

$$\alpha_1 = \alpha_2 = \dots \alpha_m = \frac{1}{m}. \quad (5.2)$$

Step 3: Each user has an average genuine segment scores and average impostor segment scores. Matching scores are obtained as the weighted sum of individual segment scores using the weights obtained:

$$GS = \sum_{i=1}^m \alpha_i g_i \quad (5.3)$$

$$IS = \sum_{i=1}^m \alpha_i im_i \quad (5.4)$$

where  $GS$  is the genuine score,  $IS$  is the impostor score,  $g_i$  is the genuine segment score of segment  $i$  and  $im_i$  is the impostor segment score of segment  $i$ .

Step 4: If  $GS > min_i$  or  $IS < max_g$ , update weights by comparing each segment score. segment index varies from 1 to  $m$

$$\text{if } g_j > min_{ij}, \alpha_j = \alpha_j \times \frac{min_{ij}}{g_j} \quad (5.5)$$

$$\text{else, } \alpha_j = \alpha_j \times \frac{im_j}{max_{gj}} \quad (5.6)$$

Renormalize the weights.

Step 5: Find  $max_g$  and  $min_i$  from the users segment score using updated weights.

If  $\frac{max_g}{min_i} \geq 1$ , goto Step 3 and continue.

We used this algorithm in two scenarios: 1. Verification and 2. Identification.



### 5.2.2 Verification

Training set is selected the same way as it was selected in Section 3.2. Each user in the gallery has a specific mask. Genuine and impostor scores for all the users in the training set are computed. Mean genuine and impostor segment scores for each user are considered to initialize the weight of the segment. Since distance scores are used, for an ideal system maximum of genuine scores have to be lower than the minimum of impostor scores. In practice, a certain overlap exists between genuine and impostor scores. Maximum of genuine and minimum of impostor scores are set as two reference points according to which the entire algorithm is based on. Weight of each segment for each user is updated as per the algorithm and the reference points are updated for each iteration until genuine scores are separated from the impostor scores. To compute the match score, final weight for each segment is obtained as the maximum weight of the two weights. Match score is obtained as the dot product of final segment weights and segment scores between the two templates.

### Results for Verification

Each user has 8 images, 4 captured at visible spectrum and 4 captured at 1550nm. Half of the data (2 images at visible spectrum and 2 images at 1550nm) is used for training and the other half is used for testing. Three sets of training data are considered. The first training set includes the images with neutral expression of all the users captured at visible spectrum and 1550 nm. A unique mask is derived for each user using the training set. This mask is applied on the testing set which contains the images with talking expressions in the first case. In the second case, training set includes all the images with talking expression and training set includes all the images with neutral expression. The third case combines both the first and second case, in which the training set includes one image of neutral expression and one image of talking expression for each spectrum. Testing set includes the images that are not used for training. Results in this section follow the second case in which talking images are used for training and neutral images are used for testing. Improvement in performance is measured based on the EER. Fig. 5.1 shows the performance improvement in the training stage. Fig. 5.2 shows the performance improvement in the testing stage. Fig. 5.3 shows

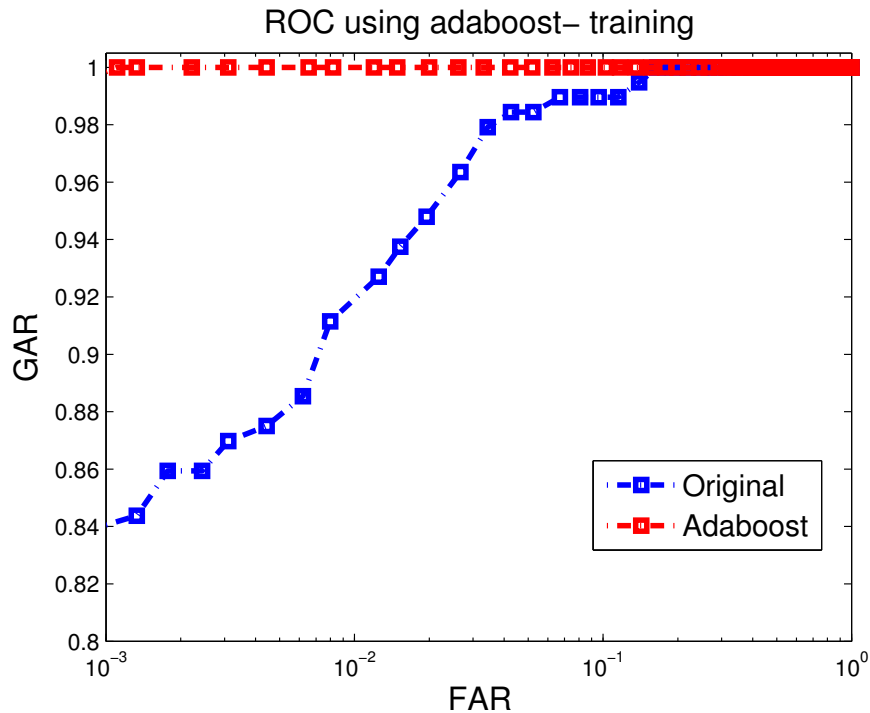


Figure 5.1: Plot showing original and improved performance using Adaboost Method 1 on training set of talking images.

the overlap between genuine and impostor scores. Fig. 5.4 shows the separation between genuine and impostor scores when the mask is used to compute the match scores. The same experiment is conducted by changing the training and testing sets.

### 5.2.3 Identification

A single weight vector is used for all the images that are found using the mean genuine and impostor scores of the training set. Confusion matrix is used to display the performance.

#### Results for Identification

Experiments are conducted with training set in which 24 classes are selected at random. All images of the selected classes are used for training. Testing set includes all the classes that are not used for testing. The same procedure is followed by changing the training and testing sets. Each trial results in an identification rate as explained below. An average of all the trials is considered. The 2nd column of Table 5.1 and Table 5.2 specifies the

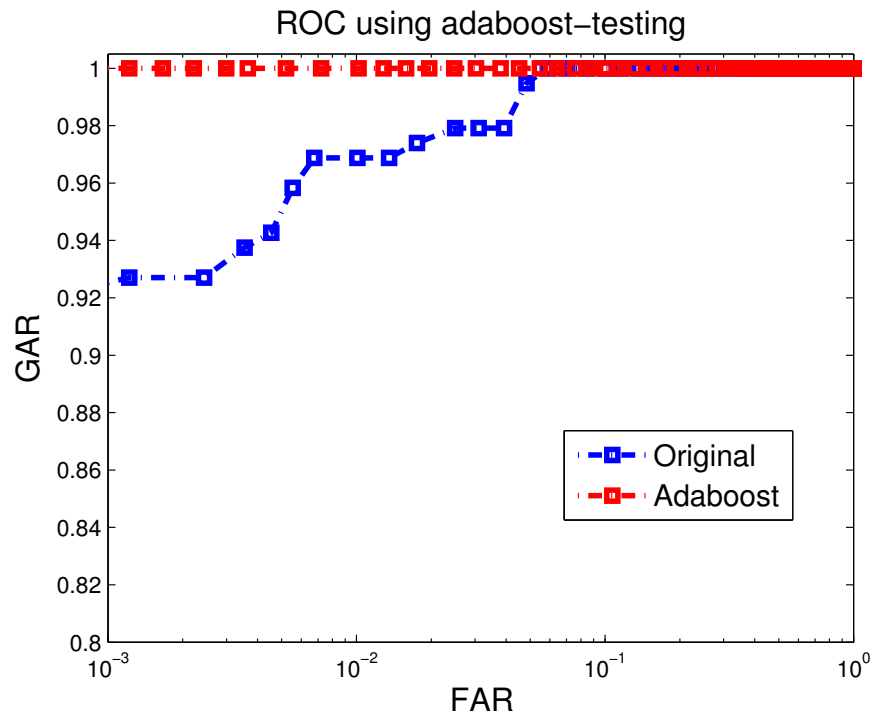


Figure 5.2: Plot showing original and improved performance using Adaboost Method 1 on testing set.

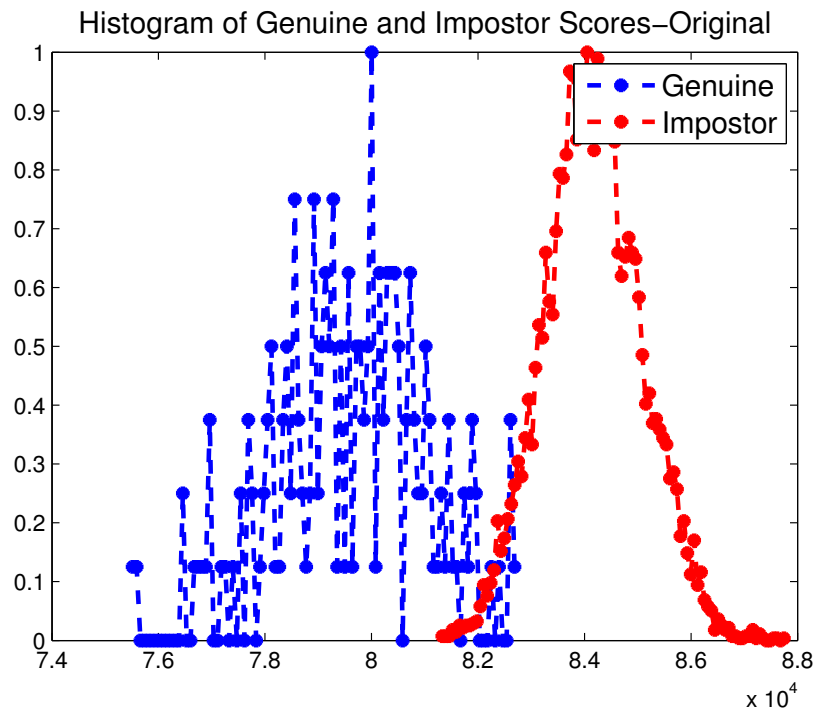


Figure 5.3: Histogram of genuine and impostor scores before applying the Adaboost Method 1.

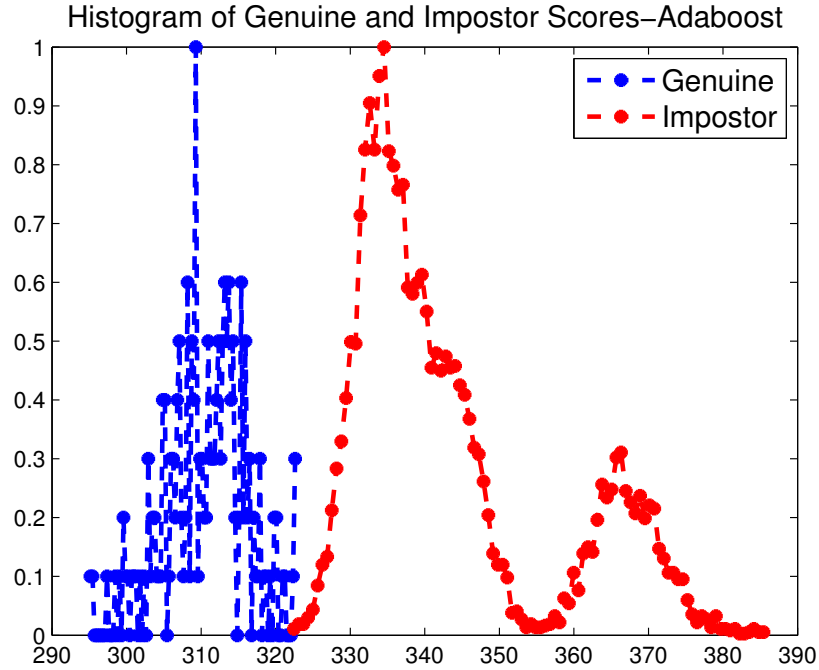


Figure 5.4: Histogram of normalized genuine and impostor scores after applying the Adaboost Method 1.

average identification rate without applying the feature selection method. The 3rd column of Table 5.1 and Table 5.2 denotes the average identification rate obtained after applying the Adaboost method. All images captured at visible spectrum are considered as gallery images. All images captured at 1550 nm are considered as query images. A confusion matrix is computed based on the first 4 ranks (since each user has 4 images captured at 1550nm). Identification rate is a measure that indicates how many classes are accurately identified. If  $C$  is the confusion matrix, identification rate is computed as follows:

$$\text{Identification Rate} = \frac{\text{sum}(\text{diag}(C))}{\text{sum}(C)} \times 100 \quad (5.7)$$

The same procedure is followed for different number of segments.

### 5.3 Method 2

Motivation for introducing this method is as follows. Since we use a distance score for comparison of templates, genuine scores have to be lower than impostor scores. Since the

	Original Identification rate(%)	Improved Identification rate (%) using Adaboost method1
256 segments	97.6563	98.9583
768 segments	96.4410	98.6111
1280 segments	97.1788	98.8715

Table 5.1: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column represents the original identification rate of the training set computed as explained above. The 3<sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 1.

	Original Identification rate(%)	Improved Identification rate (%) using Adaboost method1
256 segments	97.1788	97.7431
768 segments	96.7448	97.1788
1280 segments	96.8750	97.7865

Table 5.2: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column represents the original identification rate of the testing set computed as explained above. The 3<sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 1.

distance score is computed as the combination of segment scores, minimum score is obtained only when the segment with minimum score is considered and rest of them are discarded. This algorithm not only focuses on the minimum value but also looks for a range of values when considered results in performance improvement. This method focuses on weighing the segments based on the difference between the segment score and the dot product of weights and segment scores. Only genuine scores are used in this method to compute the weights.

### 5.3.1 Algorithm

Each feature vector in the training set is divided into  $m$  segments. Histogram of genuine and impostor matching scores is plotted.

Step 1: Maximum value of genuine ( $max_g$ ) and minimum value of impostor scores ( $min_i$ ) are stored as the initial upper and lower thresholds for each user. The goal of this algorithm is to minimize genuine score.

Step 2: Initialize segment weights. Let the weights be  $\alpha_i$ ,  $i = 1, \dots, m$ . Each segment has

the same weight equal to  $1/m$  on the first iteration.

$$\alpha_1 = \alpha_2 = \dots \alpha_m = \frac{1}{m} \quad (5.8)$$

Step 3: Each user has an average genuine segment scores and average impostor segment scores. Matching scores are obtained as the weighted sum of individual segment scores using the weights obtained in Step 5:

$$GS = \sum_{i=1}^m \alpha_i g_i \quad (5.9)$$

where  $GS$  is the genuine score and  $g_i$  is the genuine segment score of segment  $i$ .

Step 4: If  $GS > \tau$  ; ( $\tau$  ranges from minimum to maximum value of genuine segment scores)

goto Step 5

else

goto Step 6

Step 5: Update weights by comparing each segment score with the obtained genuine score and sum of all segment scores.

$$\alpha_i = \alpha_i \times \left[ \sum_{j=1}^m g_j - \Delta g_i \right], \quad (5.10)$$

where

$$\Delta g_i = g_i - \sum_{j=1}^m \alpha_j g_j, \quad (5.11)$$

Renormalize the weights. goto Step 3

Step 6: Exit

We used this algorithm in two scenarios: 1. Verification and 2. Identification.

### 5.3.2 Verification

Training set is selected the same way as it was selected in Section 3.2. Each user in the gallery has a specific mask. From the training set of scores, each user has a single set

of genuine segment scores obtained as the mean of the training set. The above algorithm is used to find the final weight vector for each user. Each user has a separate weight vector. To compute the match score, final weight for each segment is obtained as the maximum weight of the two weights. Match score is obtained as the dot product of final segment weights and the corresponding segment scores between the two templates. A range of thresholds for which improved performance is obtained is recorded and is reported in the section below.

## Results for Verification

Each user has 8 images, 4 captured at visible spectrum and 4 captured at 1550nm. Half of the data (2 images at visible spectrum and 2 images at 1550nm) is used for training and the other half is used for testing. Three sets of training data are considered. The first training set includes the images with neutral expression of all the users captured at visible spectrum and 1550 nm. A unique mask is derived for each user using the training set. This mask is applied on the testing set which contains the images with talking expressions in the first case. In the second case, training set includes all the images with talking expression and training set includes all the images with neutral expression. The third case combines both the first and second case, in which the training set includes one image of neutral expression and one image of talking expression for each spectrum. Testing set includes the images that are not used for training. Results in this section follow the second case in which talking images are used for training and neutral images are used for testing. Fig. 5.5 shows the performance at different thresholds in the training stage considering 256 segments. Fig. 5.6 shows the performance improvement in the testing stage considering 256 segments. Fig. 5.7 shows the overlap between genuine and impostor scores. Fig. 5.8 shows the separation between genuine and impostor scores when the mask is used to compute the match scores. The same experiment is conducted by changing the training and testing sets.

Values of thresholds resulting in performance improvement are specified in the 2nd column of Tables 5.3- 5.8. Values of thresholds resulting perfect performance are specified in the 3rd column of Tables 5.3- 5.8. It can be observed from the tables that the range of values in training and testing overlap.

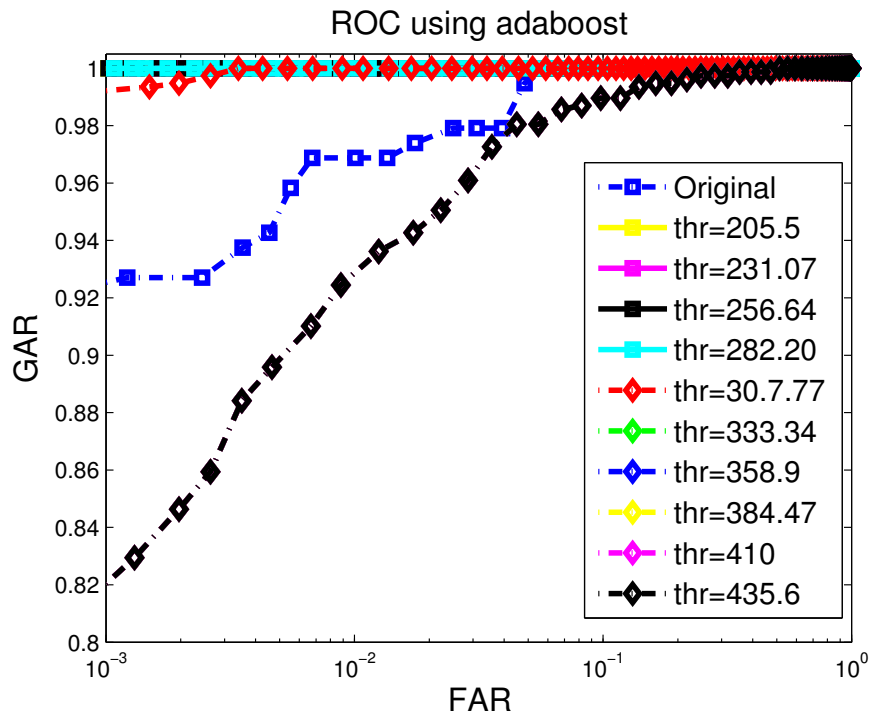


Figure 5.5: Plot showing the performance of system at different thresholds. Adaboost Method 2 is applied on training set consisting of neutral images.

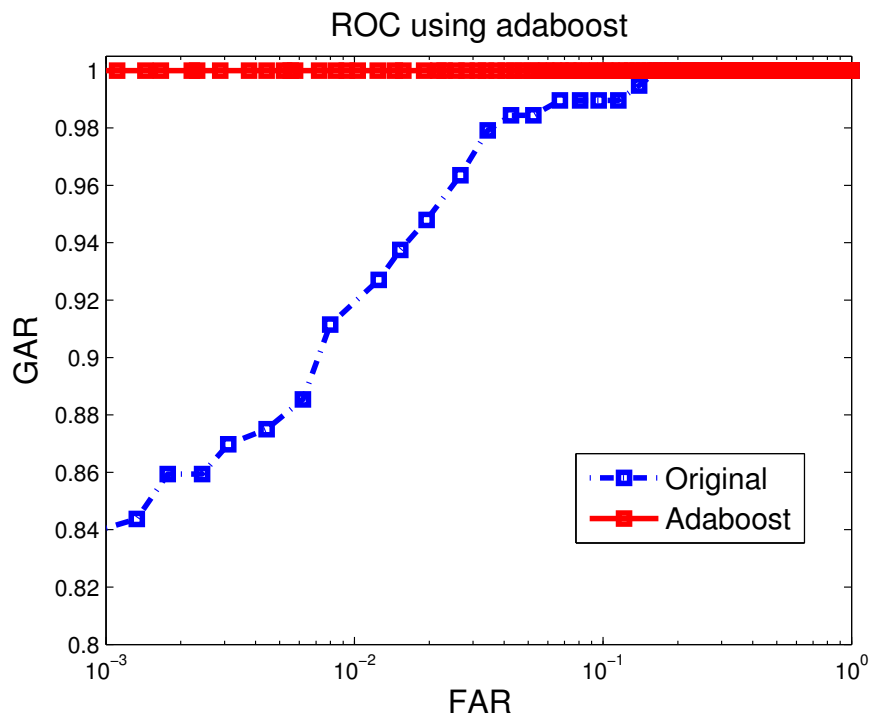


Figure 5.6: Plot showing the ROCs of original and improved performance using Adaboost Method 2 on testing set.



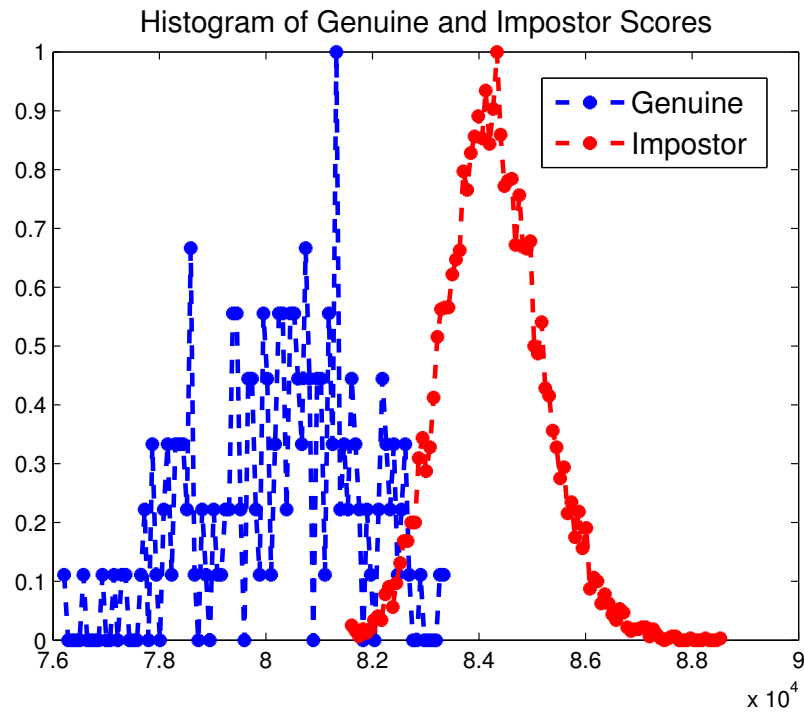


Figure 5.7: Histogram of genuine and impostor scores before applying the Adaboost Method 2.

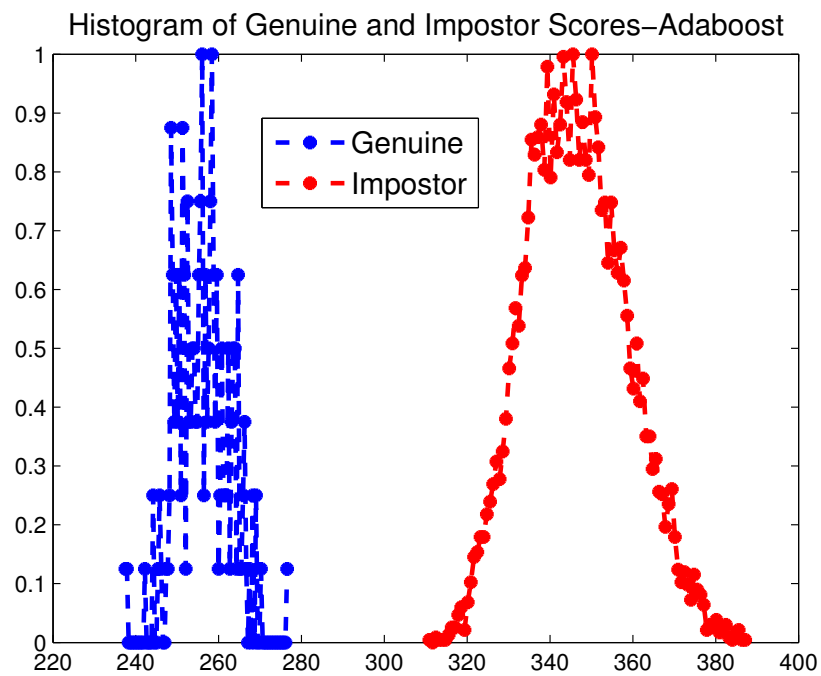


Figure 5.8: Histogram of normalized genuine and impostor scores after applying the Adaboost Method 2.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments	202-454	202-302
768 segments	207-743	207-335
1280 segments	197-718	197-356

Table 5.3: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with neutral expressions using Adaboost Method2.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments	203-321	208-306
768 segments	207-396	207-335
1280 segments	197-407	197-356

Table 5.4: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with talking expressions using Adaboost Method2.

### 5.3.3 Identification

Training set is selected the same way as it was selected in Section 3.3. Mean genuine segment score is computed from the training set. Weight for each segment is computed by applying the algorithm to mean genuine segment score. Match score is computed as the dot product of the vector of weights with segment scores. The range of thresholds that result in performance improvement is analyzed.

### Results for Identification

Experiments are conducted with training set in which 24 classes are selected at random. All images of the selected classes are used for training. Testing set includes all the classes that are not used for testing. The same procedure is followed by changing the training and testing sets. Each trial results in an identification rate as explained below. An average of all the trials is considered. The 3rd column of Table 5.9 and Table 5.10 specifies the

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments	208-481	208-309
768 segments	203-741	203-373
1280 segments	187-692	187-389

Table 5.5: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of images with talking expressions using Adaboost Method2.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments	208-319	211-306
768 segments	203-403	203-379
1280 segments	187-395	187-384

Table 5.6: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of images with neutral expressions using Adaboost Method2.

average identification rate without applying the feature selection method. The 4th column of Table 5.9 and Table 5.10 denotes the average identification rate obtained at different thresholds specified in the 2nd column.

All images captured at visible spectrum are considered as gallery images. All images captured at 1550 nm are considered as query images. A confusion matrix is computed based on the first 4 ranks (since each user has 4 images captured at 1550nm). Identification rate is a measure that indicates how many classes are accurately identified. If  $C$  is the confusion matrix, identification rate is computed as follows:

$$\text{Identification Rate} = \frac{\text{sum}(\text{diag}(C))}{\text{sum}(C)} \times 100. \quad (5.12)$$

The same procedure is followed for different number of segments.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments	203-461	208-306
768 segments	236-717	236-363
1280 segments	218-751	218-379

Table 5.7: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the training set composed of one image with neutral expression and the other with talking expression using Adaboost Method2.

	$\tau$ range with performance improvement	$\tau$ range with perfect performance
256 segments	203-320	211-306
768 segments	236-399	236-369
1280 segments	218-408	218-384

Table 5.8: Improved performance is observed for a range of thresholds specified in the second column of the table. Thresholds resulting in perfect performance are listed in the third column. The first column specifies the selection rule and the total number of segments in a single histogram. The results are provided for the testing set composed of one image with neutral expression and the other with talking expression using Adaboost Method2.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using genuine thresholding
256 segments	-	98.0469	98.0469
768 segments	-	96.5885	96.5885
1280 segments	353-413	96.8229	96.9401

Table 5.9: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column represents the original identification rate of the training set computed as explained above. The 3<sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 2.

	$\tau$ range with performance improvement	Original Identification rate(%)	Improved Identification rate (%) using genuine thresholding
256 segments	-	97.1354	97.1354
768 segments	-	97.0313	97.0313
1280 segments	375-413	96.9271	96.9866

Table 5.10: The 1<sup>st</sup> column specifies the rule for dimensionality reduction and the number of segments per histogram. The 2<sup>nd</sup> column represents the original identification rate of the testing set computed as explained above. The 3<sup>rd</sup> column represents the identification rate obtained after applying Adaboost Method 2.

# Chapter 6

## Conclusion and Future Work

### 6.1 Conclusion

A traditional face recognition system considers a color image as an input, compares the input with the color images in the database and outputs a decision. In this thesis, we compare color images with SWIR images and evaluate the performance of the system in both verification and identification scenarios.

We used the Boosted LGPI technique to encode the face image into histogram with magnitude and phase response computed using LBP and GLBP operators. The histograms are observed to have long runs of zeros, which motivates the application of feature selection methods.

We designed three feature selection methods to consider only the relevant features based on two parameters: segment size, threshold. In the first method, Genuine segment score thresholding, we discarded the unimportant data using two rules of selection of segments based on the threshold. Performance of the system is observed to improve at certain threshold range. The second feature selection method, d-prime thresholding uses the  $d'$  values to discard the segments with little or no information. This method also is based on the threshold and two rules of selecting useful segments. We also experimented with a modified  $d'$  values. The performance of the system is studied in all the cases.

The third feature selection approach is based on Adaboost algorithm. Two method are derived. The first method is iterative and is designed to minimize the overlap between

genuine and impostor scores. The second method is designed to update the weights based on the difference between the segment score and the weighted sum of segment scores. A threshold is introduced which has the range of genuine segment scores.

Threshold ranges resulting in improvement of performance based on Equal Error Rate(EER) are presented in the form of tables. The performance in case of identification is analyzed by extending the results of verification.

## 6.2 Future Work

1. In this thesis, we used all the frontal face images. In surveillance applications, however, the face images captured may not be perfect frontal images. In the future, we will investigate how the encoding and dimensionality reduction methods are affected by the view angle.

2. In the case of verification, each user has a specific mask. However training each subject in the database depends on the number of images per class. In our future work we would work on designing a single mask which works well for all the users in the database. By doing this, the images of any new class can also use the same mask without having to train all the classes again.

3. In identification scenario, we assumed that there is a match in the database. In our future work, we propose to work on a watch list scenario.

4. In this thesis, we concentrated on the genuine segment scores. In our future work, we propose to use both the genuine and impostor segment scores to prune the data.

5. We experimented only on 3 different segment ranges. In our future work, we propose to work on different segment size and to establish a relation between the segment size and the threshold range.

# References

- [1] A. K. Jain, R. Bolle, and S. Pankanti, *Biometrics, Personal Identification in Networked Society: Personal Identification in Networked Society*, Kluwer Academic Publishers, Norwell, MA, USA, 1998.
- [2] A. K. Jain, P. Flynn, and A. A. Ross, *Handbook of Biometrics*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.
- [3] A. K. Jain and D. Maltoni, *Handbook of Fingerprint Recognition*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2003.
- [4] R.P. Wildes, “Iris recognition: an emerging biometric technology,” *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1348–1363, sep 1997.
- [5] A. K. Jain and S. Z. Li, *Handbook of Face Recognition*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
- [6] A.K. Jain, A. Ross, and S. Prabhakar, “An introduction to biometric recognition,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 14, no. 1, pp. 4–20, jan. 2004.
- [7] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition: A literature survey,” *ACM Comput. Surv.*, vol. 35, pp. 399–458, December 2003.
- [8] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition: A literature survey,” *ACM Comput. Surv.*, vol. 35, pp. 399–458, December 2003.
- [9] L. Sirovich and M. Meytlis, “Symmetry, probability, and recognition in face space,” *Proceedings of the National Academy of Sciences*, vol. 106, no. 17, pp. 6895–6899, Apr. 2009.
- [10] P. Sinha, B. Balas, Y. Ostrovsky, and R. Russell, “Face recognition by humans: Nineteen results all computer vision researchers should know about,” *Proceedings of the IEEE*, vol. 94, no. 11, pp. 1948–1962, nov. 2006.
- [11] S. G. Kong, J. Heo, B. R. Abidi, J. Paik, and M. A. Abidi, “Recent advances in visual and infrared face recognition: a review,” *Comput. Vis. Image Underst.*, vol. 97, no. 1, pp. 103–135, Jan. 2005.



- [12] S. G. Kong, J. Heo, F. Boughorbel, Y. Zheng, B. R. Abidi, A. Koschan, M. Yi, and M. A. Abidi, "Multiscale fusion of visible and thermal ir images for illumination-invariant face recognition," *Int. J. Comput. Vision*, vol. 71, pp. 215–233, February 2007.
- [13] B. Klare and A. K. Jain, "Heterogeneous face recognition: Matching nir to visible light images," in *Proceedings of the 2010 20th International Conference on Pattern Recognition*, Washington, DC, USA, 2010, ICPR '10, pp. 1513–1516, IEEE Computer Society.
- [14] X. Chen, P. J. Flynn, and K. W. Bowyer, "Ir and visible light face recognition," *Comput. Vis. Image Underst.*, vol. 99, pp. 332–358, September 2005.
- [15] "Special issue on hyperspectral imaging," January 2002.
- [16] A. Richards, *Alien Vision: Exploring the Electromagnetic Spectrum with Imaging Technology*, SPIE Press, , Bellingham, WA, USA, 2011.
- [17] S. Z. Li, R. Chu, S. Liao, and L. Zhang, "Illumination invariant face recognition using near-infrared images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 627–639, April 2007.
- [18] F. Nicolo and N.A. Schmid, "A method for robust multispectral face recognition," in *International Conf. on Image Analysis and recognition (ICIAR 2011)*, Burnaby, BC, Canada, June 22-24,2011.
- [19] WVHTCF, "Tactical imager for night/day extended-range surveillance," .
- [20] T. Ojala, M. Pietikainen, and D. Harwood, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," in *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision Image Processing, Proceedings of the 12th IAPR International Conference on*, oct 1994, vol. 1, pp. 582 –585 vol.1.
- [21] Y. Guo and Z. Xu, "Local gabor phase difference pattern for face recognition," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, dec. 2008, pp. 1 –4.
- [22] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 49–86, 1951.
- [23] K. Fukunaga, *Introduction to statistical pattern recognition (2nd ed.)*, Academic Press Professional, Inc., San Diego, CA, USA, 1990.
- [24] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine*, vol. 2, no. 6, pp. 559–572, 1901.
- [25] T. Cox and M. Cox, *Multidimensional Scaling*, Chapman & Hall, London, 1994.
- [26] J. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, March 1964.

- [27] J. B. Tenenbaum, “Mapping a manifold of perceptual observations,” in *Proceedings of the 1997 conference on Advances in neural information processing systems 10*, Cambridge, MA, USA, 1998, NIPS '97, pp. 682–688, MIT Press.
- [28] K. Q. Weinberger, F. Sha, and L. K. Saul, “Learning a kernel matrix for nonlinear dimensionality reduction,” in *Proceedings of the twenty-first international conference on Machine learning*, New York, NY, USA, 2004, ICML '04, pp. 106–, ACM.
- [29] S. Lafon and A.B. Lee, “Diffusion maps and coarse-graining: a unified framework for dimensionality reduction, graph partitioning, and data set parameterization,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 9, pp. 1393–1403, sept. 2006.
- [30] B. Nadler, S. Lafon, R. Coifman, and I. Kevrekidis, “Diffusion maps, spectral clustering and reaction coordinates of dynamical systems,” *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 113–127, 2006.
- [31] B. Schölkopf, A. Smola, and K. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Comput.*, vol. 10, pp. 1299–1319, July 1998.
- [32] M.H.C. Law and A.K. Jain, “Incremental nonlinear dimensionality reduction by manifold learning,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 3, pp. 377–391, march 2006.
- [33] G. E. Hinton and R. R. Salakhutdinov, “Reducing the Dimensionality of Data with Neural Networks,” *Science*, vol. 313, no. 5786, pp. 504–507, July 2006.
- [34] S. T. Roweis and L. K. Saul, “Nonlinear Dimensionality Reduction by Locally Linear Embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.
- [35] M. Belkin and P. Niyogi, “Laplacian eigenmaps and spectral techniques for embedding and clustering,” 2002.
- [36] D. L. Donoho and C. Grimes, “Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, no. 10, pp. 5591–5596, May 2003.
- [37] Z. Zhang and H. Zha, “Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment,” *SIAM Journal of Scientific Computing*, vol. 26, no. 1, pp. 313–338, Dec. 2002.
- [38] P. Langley, “Selection of relevant features in machine learning,” in *In Proceedings of the AAAI Fall symposium on relevance*. 1994, pp. 140–144, AAAI Press.
- [39] S. Das, “Filters, wrappers and a boosting-based hybrid for feature selection,” in *Proceedings of the Eighteenth International Conference on Machine Learning*, San Francisco, CA, USA, 2001, ICML '01, pp. 74–81, Morgan Kaufmann Publishers Inc.

- [40] R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artif. Intell.*, vol. 97, pp. 273–324, December 1997.
- [41] A. Y. Ng, “On feature selection: Learning with exponentially many irrelevant features as training examples,” in *Proceedings of the Fifteenth International Conference on Machine Learning*. 1998, pp. 404–412, Morgan Kaufmann.
- [42] E. P. Xing, M. I. Jordan, and R. M. Karp, “Feature selection for high-dimensional genomic microarray data,” in *In Proceedings of the Eighteenth International Conference on Machine Learning*. 2001, pp. 601–608, Morgan Kaufmann.
- [43] K. Kira and L. A. Rendell, “The feature selection problem: traditional methods and a new algorithm,” in *Proceedings of the tenth national conference on Artificial intelligence*. 1992, AAAI’92, pp. 129–134, AAAI Press.
- [44] M. Dash, H. Liu, and H. Motoda, “Consistency based feature selection,” in *Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Applications*, London, UK, 2000, PADKK ’00, pp. 98–109, Springer-Verlag.
- [45] M. A. Hall, “Correlation-based feature selection for discrete and numeric class machine learning,” in *Proceedings of the Seventeenth International Conference on Machine Learning*, San Francisco, CA, USA, 2000, ICML ’00, pp. 359–366, Morgan Kaufmann Publishers Inc.
- [46] M. Hall, *Correlation-based Feature Selection for Machine Learning*, Ph.D. thesis, University of Waikato, 1999.
- [47] T. D. Wickens, *Elementary Signal Detection Theory*, Oxford University Press, New York, 1 edition, Oct. 2001.
- [48] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *J. Comput. Syst. Sci.*, vol. 55, pp. 119–139, August 1997.
- [49] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd Edition)*, Wiley-Interscience, 2 edition, Nov. 2001.