

Spring 2006

# Facial analysis in video : detection and recognition

Peichung Shih

*New Jersey Institute of Technology*

Follow this and additional works at: <https://digitalcommons.njit.edu/dissertations>



Part of the [Computer Sciences Commons](#)

---

## Recommended Citation

Shih, Peichung, "Facial analysis in video : detection and recognition" (2006). *Dissertations*. 778.  
<https://digitalcommons.njit.edu/dissertations/778>

This Dissertation is brought to you for free and open access by the Theses and Dissertations at Digital Commons @ NJIT. It has been accepted for inclusion in Dissertations by an authorized administrator of Digital Commons @ NJIT. For more information, please contact [digitalcommons@njit.edu](mailto:digitalcommons@njit.edu).

## **Copyright Warning & Restrictions**

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be “used for any purpose other than private study, scholarship, or research.” If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of “fair use” that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

**Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation**

Printing note: If you do not wish to print this page, then select “Pages from: first page # to: last page #” on the print dialog screen



The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

## **ABSTRACT**

### **FACIAL ANALYSIS IN VIDEO: DETECTION AND RECOGNITION**

**by**  
**Peichung Shih**

Biometric authentication systems automatically identify or verify individuals using physiological (e.g., face, fingerprint, hand geometry, retina scan) or behavioral (e.g., speaking pattern, signature, keystroke dynamics) characteristics. Among these biometrics, facial patterns have the major advantage of being the least intrusive. Automatic face recognition systems thus have great potential in a wide spectrum of application areas. Focusing on facial analysis, this dissertation presents a face detection method and numerous feature extraction methods for face recognition.

Concerning face detection, a video-based frontal face detection method has been developed using motion analysis and color information to derive field of interests, and distribution-based distance (DBD) and support vector machine (SVM) for classification. When applied to 92 still images (containing 282 faces), this method achieves 98.2% face detection rate with two false detections, a performance comparable to the state-of-the-art face detection methods; when applied to video streams, this method detects faces reliably and efficiently.

Regarding face recognition, extensive assessments of face recognition performance in twelve color spaces have been performed, and a color feature extraction method defined by color component images across different color spaces is shown to help improve the baseline performance of the Face Recognition Grand Challenge (FRGC) problems. The experimental results show that some color configurations, such as  $YV$  in the  $YUV$  color space and  $YI$  in the  $YIQ$  color space, help improve face recognition performance. Based on these improved results, a novel feature extraction method implementing genetic algorithms (GAs) and the Fisher linear discriminant (FLD) is designed to derive the optimal discriminating features that lead to an effective image representation for face recognition.



This method noticeably improves FRGC ver1.0 Experiment 4 baseline recognition rate from 37% to 73%, and significantly elevates FRGC ver2.0 Experiment 4 baseline verification rate from 12% to 69%. Finally, four two-dimensional (2D) convolution filters are derived for feature extraction, and a 2D+3D face recognition system implementing both 2D and 3D imaging modalities is designed to address the FRGC problems. This method improves FRGC ver2.0 Experiment 3 baseline performance from 54% to 72%.

**FACIAL ANALYSIS IN VIDEO: DETECTION AND RECOGNITION**

**by  
Peichung Shih**

**A Dissertation  
Submitted to the Faculty of  
New Jersey Institute of Technology  
in Partial Fulfillment of the Requirements for the Degree of  
Doctor of Philosophy in Computer Science**

**Department of Computer Science**

**May 2006**

Copyright © 2006 by Peichung Shih

ALL RIGHTS RESERVED

## APPROVAL PAGE

### FACIAL ANALYSIS IN VIDEO: DETECTION AND RECOGNITION

**Peichung Shih**

---

Dr. Chengjun Liu, Dissertation Advisor Date  
Assistant Professor of Computer Science, New Jersey Institute of Technology

---

Dr. Joseph Y. Leung, Committee Member Date  
Distinguished Professor of Computer Science, New Jersey Institute of Technology

---

Dr. James Geller, Committee Member Date  
Professor of Computer Science, New Jersey Institute of Technology

---

Dr. Andrew Sohn, Committee Member Date  
Associate Professor of Computer Science, New Jersey Institute of Technology

---

Dr. Roy You, Committee Member Date  
Assistant Professor of Electrical and Computer Engineering, New Jersey Institute of Technology

## BIOGRAPHICAL SKETCH

**Author:** Peichung Shih  
**Degree:** Doctor of Philosophy  
**Date:** May 2006

### Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Science,  
New Jersey Institute of Technology, Newark, New Jersey, 2006
- Master of Science in Information Systems,  
New Jersey Institute of Technology, Newark, New Jersey, 2002
- Bachelor of Science in Biology,  
Tunghai University, Taichung, Taiwan, 1998

**Major:** Computer Science

### Presentations and Publications:

Peichung Shih and Chengjun Liu, "Improving the Face Recognition Grand Challenge baseline performance using color configurations across color spaces," in *Proc. International Conference on Image Processing (ICIP'06)*, Atlanta, Georgia, October 8 - 11, 2006.

Peichung Shih and Chengjun Liu, "Face detection using discriminating feature analysis and Support Vector Machine," *Pattern Recognition*, vol. 39, no. 2, pp. 260 - 276, 2006.

Peichung Shih and Chengjun Liu, "Comparative assessment of content-based face image retrieval in different color spaces," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, no. 7, pp. 873 - 894, 2005.

Peichung Shih and Chengjun Liu, "Evolving effective color features for improving FRGC baseline performance," in *Proc. IEEE Computer Vision and Pattern Recognition (CVPR'05) Workshop on Face Recognition Grand Challenge Experiments*, San Diego, California, June 21, 2005.

- Peichung Shih and Chengjun Liu, "Comparative assessment of content-based face image retrieval in different color spaces," in *Proc. Audio- and Video-based Biometric Person Authentication (AVBPA'05)*, Rye Brook, New York, July 20 - 22, 2005, pp. 1039 - 1048.
- Peichung Shih and Chengjun Liu, "Extracting efficient color features for face recognition using evolutionary computation," in *Proc. International Conference of Computational Intelligence and Multimedia Applications (ICCIMA'05)*, Las Vegas, Nevada, August 16 - 18, 2005, pp. 285 - 290.
- Peichung Shih and Chengjun Liu, "Face detection using distribution-based distance and Support Vector Machine," in *Proc. International Conference of Computational Intelligence and Multimedia Applications (ICCIMA'05)*, Las Vegas, Nevada, August 16 - 18, 2005, pp. 327 - 332.
- Peichung Shih and Chengjun Liu, "Face detection using discriminating feature analysis and Support Vector Machine in video," in *Proc. 17th International Conference on Pattern Recognition (ICPR'04)*, Cambridge, United Kingdom, August 23 - 26, 2004, pp. 407 - 410.
- "Evolving effective color features for face recognition", Face Recognition Advanced Study Workshop, Morgantown, West Virginia, November 11, 2005.
- "Evolving effective color features for improving FRGC baseline performance", IEEE Computer Vision and Pattern Recognition (CVPR'05) Workshop on Face Recognition Grand Challenge Experiments, San Diego, California, June 21, 2005.
- "Face detection using distribution-based distance and Support Vector Machine", International Conference of Computational Intelligence and Multimedia Applications (ICCIMA'05), Las Vegas, Nevada, August 17, 2005.
- "Extracting efficient color features for face recognition using evolutionary Computation", International Conference of Computational Intelligence and Multimedia Applications (ICCIMA'05), Las Vegas, Nevada, August 17, 2005.
- "Face detection in video", Graduate Student Seminar 2004 (GSSD'04), New Jersey Institute of Technology, November 23, 2004.
- "Face detection using discriminating feature analysis and Support Vector Machine in video", 17th International Conference on Pattern Recognition (ICPR'04), Cambridge, United Kingdom, August 24, 2004.

*“Behind every successful man is a woman, . . . ”*  
– Groucho Marx (1890-1977)

Although not yet successful,  
I am fortunate to have three women behind my back.

To

Mom, Irene, and Shannon

## ACKNOWLEDGMENT

Foremost, I would like to express my sincere appreciation to my dissertation advisor, Dr. Chengjun Liu, for his technical guidance, kindness, and confidence in me to bring this dissertation to fruition. During the past three years, Dr. Liu has been a valued mentor and a close friend to me. I will always be indebted to Dr. Liu for encouraging and supporting me to present our research achievements to international conferences, which significantly improves my confidence and gives me great opportunities to exchange insights with leading researchers in the world.

Secondly, I am extremely grateful to Dr. Joseph Y. Leung, Dr. James Geller, Dr. Andrew Sohn, and Dr. Roy You for serving on my committee. In addition, I would like to extend special thanks to Dr. Carol Venanzi from the Department of Chemistry at New Jersey Institute of Technology. They have provided me with academic advices inside and outside my research field. Their friendship and wit have been a tremendous source of emotional support. This dissertation would not have been possible without their invaluable advice and generous help.

I would also like to thank my fellow graduate students for their assistances and supports, in particular to Cliff Hewitt, Yu Wang, Chao-Fa Chuang, and Yan-Yu Fu. I am also thankful to many friends outside the academic program for their professional advice and welcome friendship, including Kohung Chen, Mandy Wang, Chien-Nan Chi, and Ya-Hsueh Yang.

Most importantly, I heartily appreciate my family for being my emotional anchor through not only the roughness of Ph.D. study, but my entire life. I thank my wife, Tsuiting, for her support, patience, and unwavering love upon which the past nine years of my life has been built. I thank my parents, Che-Hung and Hung-Ying, for their faith in me and allowing me to be as ambitious as I wanted. It was under their watchful eye that I gained so much strength and ability to tackle head-on challenges. Along the way, I am thankful to



my sister, Liang-Yu, for sharing her caring thoughts and providing endless encouragement. I am also grateful to my grandparents for being a model of hope and optimism. They have taught me how to love and cherish life, and have shown me the importance of family. Furthermore, my enduring gratitude goes to my uncle and aunt, Brian and Flora Jow, for taking care of me throughout my stay in New Jersey. They have helped me learn about life and live a good one. They have also given me comfort and advice whenever I needed them. Finally, I deeply appreciate my in-laws, specifically to Ching-Chih Chen, Yueh-Mei T. Chen, Chia-Wen Chen, and Kent Ho for their support and belief in me. Their wisdom and experience have been valuable assets to me.

## TABLE OF CONTENTS

Chapter	Page
1 INTRODUCTION . . . . .	1
1.1 Motivation . . . . .	1
1.2 Topics Overview . . . . .	2
2 FACE DETECTION IN STILL IMAGES AND VIDEO STREAMS . . . . .	5
2.1 Background . . . . .	6
2.2 Field of Interests Derivation . . . . .	7
2.2.1 Motion Analysis . . . . .	8
2.2.2 Skin Color Segmentation . . . . .	10
2.3 The DBD-SVM Face Detection Method . . . . .	12
2.3.1 Feature Extraction for Face Detection . . . . .	13
2.3.2 Distribution-Based Distance . . . . .	15
2.3.3 Support Vector Machine . . . . .	18
2.3.4 Classification Rules . . . . .	20
2.4 Experiments . . . . .	21
2.4.1 Training the Classifiers . . . . .	21
2.4.2 Learning the Thresholds . . . . .	22
2.4.3 Scanning Procedure . . . . .	27
2.4.4 Face Detection in Still Images . . . . .	28
2.4.5 Face Detection in Video Streams . . . . .	38
2.5 Conclusion . . . . .	41
3 FACE RECOGNITION USING COLOR CONFIGURATIONS . . . . .	46
3.1 Background . . . . .	48
3.2 Color Spaces . . . . .	49
3.3 Comparative Assessments of Face Recognition Performance . . . . .	54
3.3.1 Methodology . . . . .	55

## TABLE OF CONTENTS (Continued)

Chapter	Page
3.3.2 Experiments . . . . .	57
3.4 Feature Extraction Using Component Images Across Color Spaces . . . .	66
3.4.1 Methodology . . . . .	69
3.4.2 Experiments . . . . .	74
3.5 Conclusion . . . . .	82
4 FACE RECOGNITION USING EVOLUTIONARY COLOR FISHER FEATURES	84
4.1 Background . . . . .	85
4.2 Methodology . . . . .	86
4.2.1 The $LC_1C_2$ Color Space . . . . .	86
4.2.2 Evolutionary Computation for the Optimal $LC_1C_2$ . . . . .	88
4.2.3 Color Fisher Features . . . . .	94
4.3 Experiments . . . . .	95
4.4 Conclusion . . . . .	107
5 2D AND 3D FACE RECOGNITION USING CONVOLUTION FILTERS . . .	108
5.1 Background . . . . .	110
5.2 Convolution Filters . . . . .	111
5.2.1 Gaussian Derivative Filter . . . . .	111
5.2.2 Morlet Filter . . . . .	114
5.2.3 Complex Morlet Filter . . . . .	116
5.2.4 Complex Frequency B-Spline Filter . . . . .	119
5.3 Convolution Features and Decision Fusion . . . . .	121
5.4 Experiments . . . . .	124
5.5 Conclusion . . . . .	131
6 SUMMARY AND FUTURE RESEARCH . . . . .	132
6.1 Achievements and Contributions . . . . .	132

**TABLE OF CONTENTS**  
**(Continued)**

<b>Chapter</b>	<b>Page</b>
6.2 Future Research . . . . .	134
REFERENCES . . . . .	136

## LIST OF TABLES

Table	Page
2.1 Comparative Face Detection Performance of the Schneiderman-Kanade Method and DBD-SVM . . . . .	37
3.1 Color Configurations that Outperform Intensity Images . . . . .	72
3.2 Comparative Face Recognition Performance of the $YQ C_r$ Color Configuration and the BEE Intensity Images . . . . .	83
4.1 The Rank-One Face Recognition Rates of the FRGC ver1.0 CMC Performance	100
4.2 Face Verification Rates (at 0.1% FAR) of FRGC ver2.0 Experiment 4 Using PCA . . . . .	101
4.3 Face Verification Rates (at 0.1% FAR) of FRGC ver2.0 Experiment 4 Using FLD . . . . .	104

## LIST OF FIGURES

Figure	Page
2.1 The derivation of motion FOI. The white pixels denote the motion pixels, and black pixels are the non-motion pixels. (a) Four thresholds denoted by four yellow lines cut off insignificant motion area. (b) The resulting motion FOI.	8
2.2 Applying a $5 \times 5$ Gaussian smoothing filter eliminates undesirable false detections caused by dynamic environmental changes, such as illumination variation.	9
2.3 Example face images from the FERET database. The white strips indicate the skin color pixels acquired. For each face image, ninety skin color pixels are collected.	11
2.4 The cluster of 5,400 skin color pixels in the $RGB$ color space, and the projections onto the $R - G$ , the $R - B$ , and the $G - B$ subspaces.	12
2.5 The cluster of 5,400 skin color pixels in the $YC_bC_r$ color space, and the projections onto the $Y - C_b$ , the $Y - C_r$ , and the $C_b - C_r$ subspaces.	13
2.6 The cluster of 5,400 skin color pixels in the $L^*a^*b^*$ color space, and the projections onto the $L^* - a^*$ , the $L^* - b^*$ , and the $a^* - b^*$ subspaces.	14
2.7 (a) An elliptic skin color model. (b) Skin color segmentation in the motion FOI.	15
2.8 Three classification rules (in black diamond shapes) implementing distribution-based distance and support vector machine.	16
2.9 Some examples of the training face and non-face images. (a) Examples of the face training images. (b) Examples of the non-face training images.	23
2.10 Image representations of the mean feature vectors of the face and the non-face classes. (a) The first image is the mean face, the second and the third images are its vertical and horizontal difference images, and the last two bar graphs are its column and row amplitude projections. (b) The first image is the mean non-face, the second and the third images are its vertical and horizontal difference images, and the last two bar graphs are its column and row amplitude projections.	24
2.11 The learning of thresholds, $\tau_n$ and $\tau_f$ . (a) The relationship between the total number of non-face subimages and the threshold $\tau_n$ . (b) The threshold, $\tau_f$ , is set to be the largest distribution-based distance of all these non-face images. However, the threshold, $\tau_f$ , is applied for coarse detection, whose purpose is to reliably classify face and non-face patterns while leaving difficult patterns in the undecided class. The threshold is thus increased from $\tau_f$ to $\tau_f^*$ in order to reduce the number of false detections.	25

## LIST OF FIGURES (Continued)

Figure	Page
2.12 The learning of thresholds, $\tau_s$ and $\tau_t$ . (a) The number of missed faces increases when either $\tau_s$ or $\tau_t$ increases. (b) The number of false detections decreases when either $\tau_s$ or $\tau_t$ increases. (c) The number of total errors (missed faces + false detections) versus the thresholds ( $\tau_s$ and $\tau_t$ ). The smallest total error occurs at $\{\tau_s, \tau_t\} = \{-0.68, -0.66\}$ . . . . .	26
2.13 (a) A face detected at $q$ eliminates the region $B$ from being searched because any face appears in this region will overlap the one found at $q$ . (b) A face detected in the scale of $a \times a$ excludes the region $R$ when searching in the scale of $b \times b$ . . . . .	29
2.14 A $16 \times 16$ subimage with five labeled regions corresponding to the left eye area (A), the nose bridge area (B), the right eye area (C), the nose-cheeks area (D), and the mouth area (E). . . . .	30
2.15 Detecting multiple faces using the DBD-SVM method. . . . .	31
2.16 Detecting rotated faces using the DBD-SVM method. . . . .	32
2.17 Detecting faces that are either very large or very small using the DBD-SVM method. . . . .	33
2.18 Detecting faces in low quality images using the DBD-SVM method. . . . .	35
2.19 Detecting faces with illumination and slight pose variations using the DBD-SVM method. . . . .	36
2.20 Face detection in a video stream. The upper half of frames show the original input stream, and the lower half of frames show the motion analysis results. . . . .	39
2.21 Face detection in a video stream. The upper half of frames show the FOIs derived from both motion analysis and skin color segmentation, and the lower half of frames show the face detection results. . . . .	40
2.22 Face detection in a video stream with complex background. The upper half of frames show the original input stream, and the lower half of frames show the motion analysis results. . . . .	42
2.23 Face detection in a video stream with complex background. The upper half of frames show the FOIs derived from both motion analysis and skin color segmentation, and the lower half of frames show the face detection results. . . . .	43
2.24 Face detection in a video stream with illumination variation. The upper half of frames show the original input stream, and the lower half of frames show the motion analysis results. . . . .	44

## LIST OF FIGURES (Continued)

Figure	Page
2.25 Face detection in a video stream with illumination variation. The upper half of frames show the FOIs derived from both motion analysis and skin color segmentation, and the lower half of frames show the face detection results.	45
3.1 Color component images of twelve color spaces as defined in Section 3.2. From left to right, top to bottom, the color spaces are the $RGB$ color space, the $HSV$ color space, the $YUV$ color space, the $YC_bC_r$ color space, the $XYZ$ color space, the $YIQ$ color space, the $L^*a^*b^*$ color space, the $U^*V^*W^*$ color space, the $L^*u^*v^*$ color space, the $I_1I_2I_3$ color space, the $HSI$ color space, and the $rgb$ color space. . . . .	58
3.2 The face recognition performance using the PCA method on the intensity images derived by averaging the $R, G, B$ color components. The similarity measures applied are the $L_1$ distance measure ( $L_1$ ), the $L_2$ distance measure ( $L_2$ ), the Mahalanobis distance measure ( $Mah$ ), and the cosine similarity measure ( $cos$ ). . . . .	60
3.3 The face recognition performance of seven color configurations in the $RGB$ color space. Note that the performance curve of the intensity images (intensity) is also included for comparison (same in the following figures). . . .	61
3.4 The face recognition performance of seven color configurations in the $HSV$ color space. . . . .	62
3.5 The face recognition performance of seven color configurations in the $YUV$ color space. Note that the face recognition results in the $YC_bC_r$ color space are the same when the $Y, U$ , and $V$ color components are replaced by their counterparts $Y, C_b$ , and $C_r$ , respectively. . . . .	63
3.6 The face recognition performance of seven color configurations in the $XYZ$ color space. . . . .	64
3.7 The face recognition performance of seven color configurations in the $YIQ$ color space. . . . .	65
3.8 The face recognition performance of seven color configurations in the $L^*a^*b^*$ color space. . . . .	66
3.9 The face recognition performance of seven color configurations in the $U^*V^*W^*$ (or $L^*u^*v^*$ ) color space. . . . .	67
3.10 The face recognition performance of seven color configurations in the $I_1I_2I_3$ color space. . . . .	68
3.11 The face recognition performance of seven color configurations in the $HSI$ color space. . . . .	69



## LIST OF FIGURES (Continued)

Figure	Page
3.12 The face recognition performance of seven color configurations in the $rgb$ color space. . . . .	70
3.13 Comparative face recognition performance of the best color configurations in the $RGB$ color space, the $HSV$ color space, the $YUV/YC_bC_r$ color space, the $XYZ$ color space, the $YIQ$ color space, the $L^*a^*b^*$ color space, and the $U^*V^*W^*/L^*u^*v^*$ color space. . . . .	71
3.14 Comparative face recognition performance of seven color configurations using 456 FRGC color images. . . . .	73
3.15 Comparative face recognition performance of seven color configurations using 456 FRGC color images with eye location variations occurred in $11 \times 11$ pixel region. . . . .	74
3.16 (a) Example target, query, and training images normalized to $64 \times 64$ . The top image in either column one or column two is a target image, while the remaining two are query images. Images in the third column are training images. (b) A BEE intensity image of the size $150 \times 130$ with an oval mask applied to extract the face region. . . . .	76
3.17 Comparative face recognition performance using the $YIQ$ color configuration, the $YC_bC_r$ color configuration, and the BEE intensity images. The horizontal axis indicates the number of features used, and the vertical axis represents the rank-one face recognition rate, which is the accuracy rate for the top response being correct. . . . .	77
3.18 The CMC curves derived using the color configurations defined in the $YIQ$ color space. The BEE baseline performance is also included for comparison. Note that the curves corresponding to the color configurations and the BEE baseline performance are derived using 365 and 146 features, respectively. . . . .	79
3.19 The CMC curves derived using the color configurations defined in the $YC_bC_r$ color space. The BEE baseline performance is also included for comparison. Note that the curves corresponding to the color configurations and the BEE baseline performance are derived using 365 and 146 features, respectively. . . . .	80
3.20 The CMC curves derived using the color configurations defined across different color spaces. The BEE baseline performance is also included for comparison. Note that the curves corresponding to the color configurations and the BEE baseline performance are derived using 365 and 146 features, respectively. . . . .	81

## LIST OF FIGURES (Continued)

Figure	Page
3.21 The ROC curves derived using the $YQC_r$ color configuration and the BEE intensity images. . . . .	82
4.1 (a) The search space of $a_1, a_2, a_3$ , and (b) the search space of $a_4, a_5, a_6$ (or $a_7, a_8, a_9$ ). . . . .	88
4.2 An overview of the GA procedure. For each generation, a fixed number of individuals ( $N_s$ ) are selected by the selection operator from a population of size $N$ to reproduce offspring through the crossover and the mutation operators. The newly generated offspring are then placed back to the current population to form a temporary population of size $N + N_s$ , from which the $N_s$ least fit individuals are removed to return the population to its original size. . . . .	89
4.3 The component-wise crossover operator. Two chromosomes of length six are capable of producing three different kinds of offspring. The dashed lines indicate possible crossover points. . . . .	92
4.4 The mutation operator that guarantees the satisfaction of constraints in Equation 4.4. . . . .	93
4.5 Three component images in the $LC_1C_2$ , the $HSV$ , and the $RGB$ color spaces, respectively. The component images have a spatial resolution $64 \times 64$ . . .	96
4.6 The course of evolution using the FRGC ver1.0 dataset. The horizontal axis represents the generation, and the vertical axis denotes the rank-one face recognition accuracy derived by the best performing chromosome in each generation. . . . .	97
4.7 The rank-one face recognition rates of the PCA method using the $\mathbf{x}_{LC_1C_2}$ and the $\mathbf{x}_{BEE}$ representations. . . . .	98
4.8 The FRGC ver1.0 CMC performance of the $\mathbf{x}_{LC_1C_2}$ and the $\mathbf{x}_{BEE}$ representations. The PCA performance using the $\mathbf{x}_{BEE}$ representation is the BEE baseline performance. . . . .	99
4.9 The ROC I performance of FRGC ver2.0 Experiment 4 using the PCA method. The curve corresponding to the $\mathbf{x}_{BEE}$ representation is the BEE baseline performance. . . . .	100
4.10 The ROC II performance of FRGC ver2.0 Experiment 4 using the PCA method. The curve corresponding to the $\mathbf{x}_{BEE}$ representation is the BEE baseline performance. . . . .	101

## LIST OF FIGURES (Continued)

Figure	Page
4.11 The ROC III performance of FRGC ver2.0 Experiment 4 using the PCA method. The curve corresponding to the $\mathbf{x}_{BEE}$ representation is the BEE baseline performance. . . . .	102
4.12 The FLD performance using the $\mathbf{x}_{LC_1C_2}$ and the $\mathbf{x}_{BEE}$ representations in different PCA spaces. . . . .	103
4.13 The ROC I performance of FRGC ver2.0 Experiment 4 using FLD. . . . .	104
4.14 The ROC II performance of FRGC ver2.0 Experiment 4 using FLD. . . . .	105
4.15 The ROC III performance of FRGC ver2.0 Experiment 4 using FLD. . . . .	106
5.1 (a) An example 1D second derivative of Gaussian wavelet. (b) An example 2D Gaussian derivative convolution filter applying the parameters: $\sigma_x = 16$ , $\sigma_y = 32$ , $\mathbf{b} = (0 \ 0)^t$ and $\theta = 30^\circ$ . . . . .	113
5.2 (a) An example 1D Morlet wavelet. (b) An example 2D Morlet convolution filter applying the parameters: $\sigma = 16$ , $\theta = 0$ , and $k = 1$ . . . . .	115
5.3 An example 1D complex Morlet wavelet applying the parameters: $f_b = 2$ and $f_c = 0.8$ . The real and the imaginary parts are represented by two separate curves. . . . .	117
5.4 An example 2D complex Morlet convolution filter applying the parameters: $f_b = 2$ , $f_c = 0.8$ , $\sigma = 16$ , $\theta = 0$ , and $k = 1$ . The real and the imaginary parts are displayed in (a) and (b), respectively. . . . .	118
5.5 an example 1D complex frequency B-spline wavelet applying the parameters: $f_b = 2$ , $f_c = 0.8$ , and $m = 1$ . The real and the imaginary parts are represented by two separate curves. . . . .	119
5.6 An example 2D complex frequency B-spline convolution filter applying the parameters: $f_b = 2$ , $f_c = 0.6$ , $k = 1$ , $m = 1$ , $\sigma = 20$ , and $\theta = 0$ . The real and the imaginary parts are displayed in (a) and (b), respectively. . . . .	120
5.7 Gaussian derivative convolution filters in eight orientations and four scales. . . . .	122
5.8 Morlet convolution filters in eight orientations and four scales. . . . .	122
5.9 Complex Morlet convolution filters in eight orientations and four scales. (a) The real parts. (b) The imaginary parts. . . . .	123
5.10 Complex frequency B-spline convolution filters in eight orientations and four scales. (a) The real parts. (b) The imaginary parts. . . . .	124

## LIST OF FIGURES (Continued)

Figure	Page
5.11 (a) Example controlled 2D training images and their corresponding 3D recordings. (b) Example uncontrolled 2D target images and their corresponding 3D recordings. . . . .	125
5.12 Comparative performance of 2D modalities applying the four convolution filters. Face verification rate at 0.1% FAR using gray-scale images, Gaussian derivative convolution features ( <i>gaus</i> ), Morlet convolution features ( <i>mor</i> ), complex Morlet convolution features ( <i>cmor</i> ), and complex frequency B-spline convolution features ( <i>bspl</i> ). For the complex frequency B-spline, four wavelets are defined corresponding to $m = 1$ , $m = 2$ , $m = 3$ , and $m = 4$ , where $m$ is the order parameter. . . . .	126
5.13 Comparative performance of 3D modalities applying the four convolution filters. Face verification rate at 0.1% FAR using gray-scale images, Gaussian derivative convolution features ( <i>gaus</i> ), Morlet convolution features ( <i>mor</i> ), complex Morlet convolution features ( <i>cmor</i> ), and complex frequency B-spline convolution features ( <i>bspl</i> ). For the complex frequency B-spline, four wavelets are defined corresponding to $m = 1$ , $m = 2$ , $m = 3$ , and $m = 4$ , where $m$ is the order parameter. . . . .	128
5.14 FRGC ver2.0 Experiment 3 performance using different $\alpha$ . . . . .	129
5.15 FRGC ver2.0 Experiment 3 performance using convolution features and similarity fusion. BEE baseline performance is also included for comparison. .	130

## LIST OF SYMBOLS

$\mathbb{R}^d$	$d$ -dimensional Euclidean space
$A, a, B, b, \dots$	italic letters are used for scalars
$\mathbf{x}, \mathbf{y}, \dots$	lower-case boldface letters are used for column vectors
$\mathbf{A}, \mathbf{B}, \dots$	upper-case boldface letters are used for matrices (or gray-scale images)
$\mathbf{A}^{-1}$	inverse of matrix $\mathbf{A}$
$\Phi$	nonlinear mapping
$\mathcal{A}, \mathcal{B}, \dots$	vector sets
$\mathcal{E}[\cdot]$	expectation operator
$\otimes$	convolution operator
$\nabla^2$	Laplacian operator
$\mathbf{I}_N$	identity matrix of size $N \times N$ , i.e., an $N$ -by- $N$ square matrix whose diagonal elements are 1's and off-diagonal elements are 0's
$\Sigma$	covariance matrix
$\Lambda, \Delta$	diagonal eigenvalue matrix
$\Phi, \Psi$	orthogonal eigenvector matrix
$\varphi_1, \varphi_2, \dots$	eigenvectors
$\lambda_i$	eigenvalues
$p(\cdot)$	probability density density function
$P(\cdot)$	probability mass function
$K(\cdot)$	kernel function

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Motivation**

Biometric authentication systems automatically identify or verify individuals using physiological or behavioral characteristics [41]. Physiological biometrics generally refer to the measurements of human body characteristics that do not change over a person's lifetime. Face recognition [52], [57], [117], [6], fingerprint verification, [43], [48], [91], hand geometry measurement [81], and retina scan analysis [10] are examples of the utilization of physiological biometrics. Behavioral biometrics, on the other hand, measure the actions that humans take, which are difficult to copy from one person to another. Applications using behavioral biometrics include speaker identification [8], signature verification [74], and keystroke recognition [5].

Among aforementioned biometrics, facial patterns have the major advantage of being the least intrusive and private. The acquisition of facial images requires very little cooperation or modification of subjects' normal behavior. Instead of requiring people to place their hand on a reader (e.g., finger print acquisition) or precisely position their eyes in front of a scanner (e.g., iris or retina scans), face recognition systems unobtrusively take pictures or video streams containing people's faces when they enter a predefined area. Another advantage is that facial image contains less private information. Since human faces are exposed to the public everyday, a malicious breach to the stored data does not lead to a serious security threats like passwords and signatures would. As subjects do not feel like they are under surveillance and the information is less sensitive, automatic face recognition systems have great potential in a wide spectrum of application areas. Applications for which it is well suited include security access control, video surveillance, human-computer interaction (HCI), content-based face image retrieval [89], airport screening, security checkpoints, etc.

## 1.2 Topics Overview

This dissertation focuses on facial analysis and addresses various aspects concerning face detection and recognition. In particular, Chapter 2 presents a novel face detection systems, Chapter 3 empirically assesses face recognition performance in twelve well-defined color spaces, Chapter 4 designs a feature extraction method implementing genetic algorithms (GAs) and the Fisher linear discriminant (FLD) for face recognition, Chapter 5 derives four convolution filters to extract discriminating features from both 2D and 3D imaging modalities for face recognition, finally, Chapter 6 summarizes the research achievements and contributions of this dissertation and depicts the future research directions. An overview of Chapters 2, 3, 4, and 5 is given in the follows.

Chapter 2 presents a video-based frontal face detection method, the DBD-SVM method, which integrates motion analysis, skin color segmentation, distribution-based distance (DBD), and Support Vector Machine (SVM). First, in each video frame, the regions of Field Of Interests (FOI) are derived by means of temporal differencing and skin color segmentation. The FOI regions are then examined using a sliding window technique. Finally, the DBD-SVM method statistically models the face class by applying the discriminating feature vectors and defines three classification rules based on DBD and SVM to separate faces and nonfaces. Experimental results using test images from the MIT-CMU test sets and video streams show the feasibility of the DBD-SVM face detection method. In particular, when using 92 images (containing 282 faces) from the MIT-CMU test sets, the DBD-SVM method achieves 98.2% correct face detection accuracy with two false detections, a performance comparable to the state-of-the-art face detection methods, such as the Schneiderman-Kanade's method. When using video streams, the DBD-SVM method detects faces reliably with computational efficiency at more than 20 frames per second.

Chapter 3 first presents extensive assessments of face recognition performance in different color spaces, and then defines a color feature extraction method to help improve the baseline performance of the FRGC problems. The comparative assessments of face recog-

nition performance are carried out in twelve color spaces ( $RGB$ ,  $HSV$ ,  $YUV$ ,  $YC_bC_r$ ,  $XYZ$ ,  $YIQ$ ,  $L^*a^*b^*$ ,  $U^*V^*W^*$ ,  $L^*u^*v^*$ ,  $I_1I_2I_3$ ,  $HSI$ , and  $rgb$ ) by evaluating seven color configurations for every color space. A color configuration is defined by an individual or a combination of color component images. Experimental results using 600 FERET color images corresponding to 200 subjects and 456 FRGC color images of 152 subjects show that some color configurations, such as  $YV$  in the  $YUV$  color space and  $YI$  in the  $YIQ$  color space, help improve face recognition performance. Based on these results, a color feature extraction method, which uses the color configurations defined by the color component images across different color spaces, is implemented to improve the FRGC baseline performance of the Biometric Experimentation Environment (BEE) algorithm. Experimental results using an FRGC ver1.0 dataset containing 366 training images, 152 controlled gallery images, and 608 uncontrolled probe images, show that the  $YQC_r$  color configuration improves the rank-one face verification rate of the BEE baseline algorithm from 37% to 70%; when using an FRGC ver2.0 dataset including 6,660 training images, 16,028 controlled target images, and 8,014 uncontrolled query images, the  $YQC_r$  color configuration improves the face verification rate (at 0.1% false acceptance rate) of the BEE baseline algorithm from 13% to 33%.

Chapter 4 presents a novel feature extraction method for face recognition implementing genetic algorithms (GAs) and Fisher linear discriminant (FLD) to derive the optimal discriminating features that lead to an effective image representation for face recognition. A new color space,  $LC_1C_2$ , consisting of one luminance ( $L$ ) channel and two chrominance channels ( $C_1, C_2$ ) is introduced as a linear transformation of the input  $RGB$  color space. The specific transformation from the  $RGB$  color space to the  $LC_1C_2$  color space is optimized by GAs where a fitness function guides the evolution toward higher recognition accuracy. After the color feature extraction the Fisher linear discriminant (FLD) method is applied to further extract discriminating features, termed color Fisher features (CFFs). The experimental results show that CFFs improve FRGC ver1.0 Experiment 4 rank-one recog-



dition rate from 37% to 73%, and elevate FRGC ver2.0 Experiment 4 face verification rate at 0.1% false acceptance rate (FAR) from 12% to 69%.

Chapter 5 presents a feature extraction method utilizing convolution filters for 2D and 3D face recognition. Four convolution filters, Gaussian derivatives, Morlet, complex Morlet, and complex frequency B-spline, are defined, and face recognition performance derived using these four convolution filters are investigated. The experimental results applying FRGC ver2.0 Experiment 3 show that complex-valued and real-valued convolution filters may have the potential to extract discriminating features from 2D and 3D imaging modalities, respectively. Furthermore, the fusion of 2D and 3D similarity measures improves FRGC ver2.0 Experiment 3 baseline performance from 67% to 82% (ROC I), from 61% to 78% (ROC II), and from 54% to 72% (ROC III).

Chapter 6 summarizes the research achievements and contributions of this dissertation and depicts the future research direction.

## **CHAPTER 2**

### **FACE DETECTION IN STILL IMAGES AND VIDEO STREAMS**

Face detection methods learn the statistical models of face and non-face images, and apply two-class classification rules to discriminate between face and non-face patterns. As a face must be located and extracted before it can be verified or identified, face detection is the first step towards building an automated facial analysis system. In this chapter, a video-based frontal face detection method – the DBD-SVM method – is presented.

The DBD-SVM method integrates motion analysis and color information to derive field of interests (FOI), and distribution-based distance (DBD) and support vector machine (SVM) for classification. A motion FOI is first located by means of temporal differencing, and then refined by skin color segmentation. Inside the FOI, where the faces reside, the face detector searches for faces by applying three classification rules implementing DBD and SVM. The three classification rules employ a coarse-to-fine classification strategy in the sense that they are cascaded in the increasing order of detection accuracy and the decreasing order of computational efficiency. Specifically, the first rule applies DBD to classify input patterns into the face, the non-face, and the undecided classes; the second rule implements an SVM classifier to detect some faces in the undecided class; and finally, the third rule combines the classification power of both DBD and SVM to discriminate the remaining patterns in the undecided class.

Experimental results derived using both still images and video streams show the feasibility of the DBD-SVM method. In particular, when using 92 images (containing 282 faces) from the MIT-CMU test sets, the DBD-SVM method achieves 98.2% face detection rate with two false detections, a performance comparable to the Schneiderman-Kanade's method [84]. When experimenting with video streams, the DBD-SVM method detects faces reliably and efficiently.

This chapter is organized as follows. Section 2.1 briefly reviews previous research on face detection. Section 2.2 describes the derivation of FOIs using motion analysis and skin color segmentation. Section 2.3 details the DBD-SVM method including the feature extraction for face detection and the three classification rules. Section 2.4 describes the experimental procedures and presents face detection results using both still images and video streams. Finally, Section 2.5 summarizes the research in face detection.

## 2.1 Background

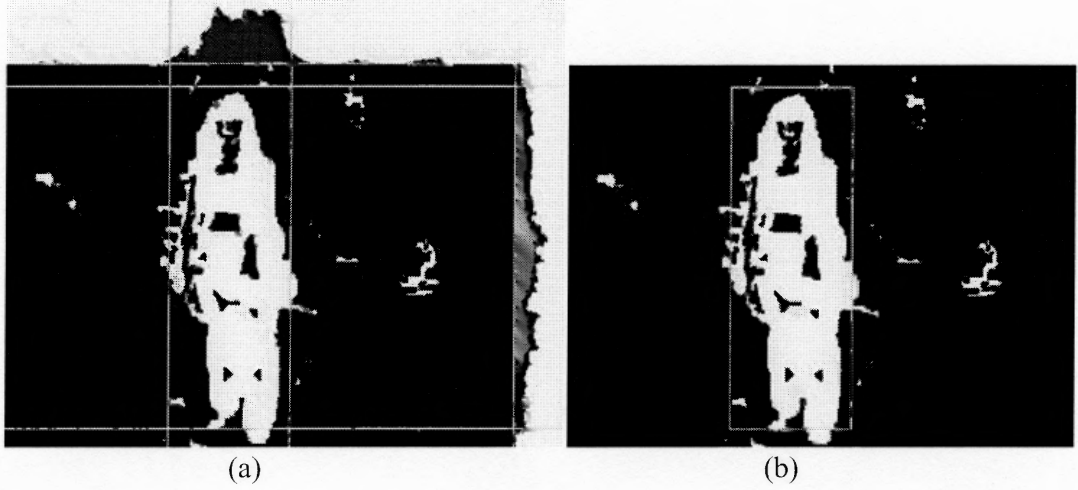
Earlier efforts of face detection research have been focused on correlation or template matching, matched filtering, sub-space methods, deformable templates, etc. [70], [116]. For comprehensive surveys of these early methods, see [93], [14], and [80]. Recent face detection approaches, however, emphasize on statistical modeling and machine learning techniques [37], [114]. Some representative methods are the probabilistic visual learning method [63], the example-based learning method [94], the neural network-based learning method [76], [77], the probabilistic modeling method [83], [84], the mixture of linear subspaces method [113], the machine learning approach using a boosted cascade of simple features [105], the statistical learning theory and SVM based methods [34], [64], [38], the Markov random field based methods [17], [75], the color-based face detection method [40], and the Bayesian discriminating feature (BDF) method [52].

Moghaddam and Pentland [63] applied unsupervised learning to estimate the density in a high-dimensional eigenspace and derived a maximum likelihood method for single face detection. Rather than using PCA for dimensionality reduction, they implemented the eigenspace decomposition as an integral part of estimating the conditional PDF in the original high-dimensional image space. Face detection is then performed by computing multi-scale saliency maps based on the maximum likelihood formulation. Sung and Poggio [94] presented an example-based learning method by means of modeling the distributions of face and non-face patterns. To cope with the variability of face images, they

empirically chose six Gaussian clusters to model the distributions for face and non-face patterns, respectively. The density functions of the distributions are then fed to a multiple layer perceptron for face detection. Rowley et al. [76] developed a neural network-based upright, frontal face detection system, which applies a retinally connected neural network to examine small windows of an image and decide whether each window contains a face. The face detector, which was trained using a large number of face and non-face examples, contains a set of neural network-based filters and an arbitrator which merges detections from individual filters and eliminates overlapping detections. In order to detect faces at any degree of rotation in the image plane, the system was extended to incorporate a separate router network, which determines the orientation of the face pattern. The pattern is then derotated back to the upright position, which can be processed by the early developed system [77]. Schneiderman and Kanade [83] proposed a face detector based on the estimation of the posterior probability function, which captures the joint statistics of local appearance and position as well as the statistics of local appearance in the visual world. To detect side views of a face, profile images are added to the training set to incorporate such statistics [84]. Viola and Jones [105] presented a machine learning approach for face detection. The novelty of their approach comes from the integration of a new image representation (integral image), a learning algorithm (based on AdaBoost), and a method for combining classifiers (cascade). Hsu et al. [40] developed a face detection method in color images by detecting skin regions first, and then generating face candidates based on some constraints, such as the spatial arrangement. The face candidates are further verified by constructing eye, mouth, and boundary maps.

## 2.2 Field of Interests Derivation

In the DBD-SVM method, the first step is to locate the field of interests (FOI) in each video frame by utilizing both temporal and color information. In particular, each individual frame is transformed from the input  $RGB$  color space to the  $YC_bC_r$  color space, where the  $Y$



**Figure 2.1** The derivation of motion FOI. The white pixels denote the motion pixels, and black pixels are the non-motion pixels. (a) Four thresholds denoted by four yellow lines cut off insignificant motion area. (b) The resulting motion FOI.

component is used to derive the motion FOI, and the  $C_b$  and the  $C_r$  components are used for skin color segmentation, which refines the motion FOI. The motion analysis and the skin color segmentation are detailed in the following subsections.

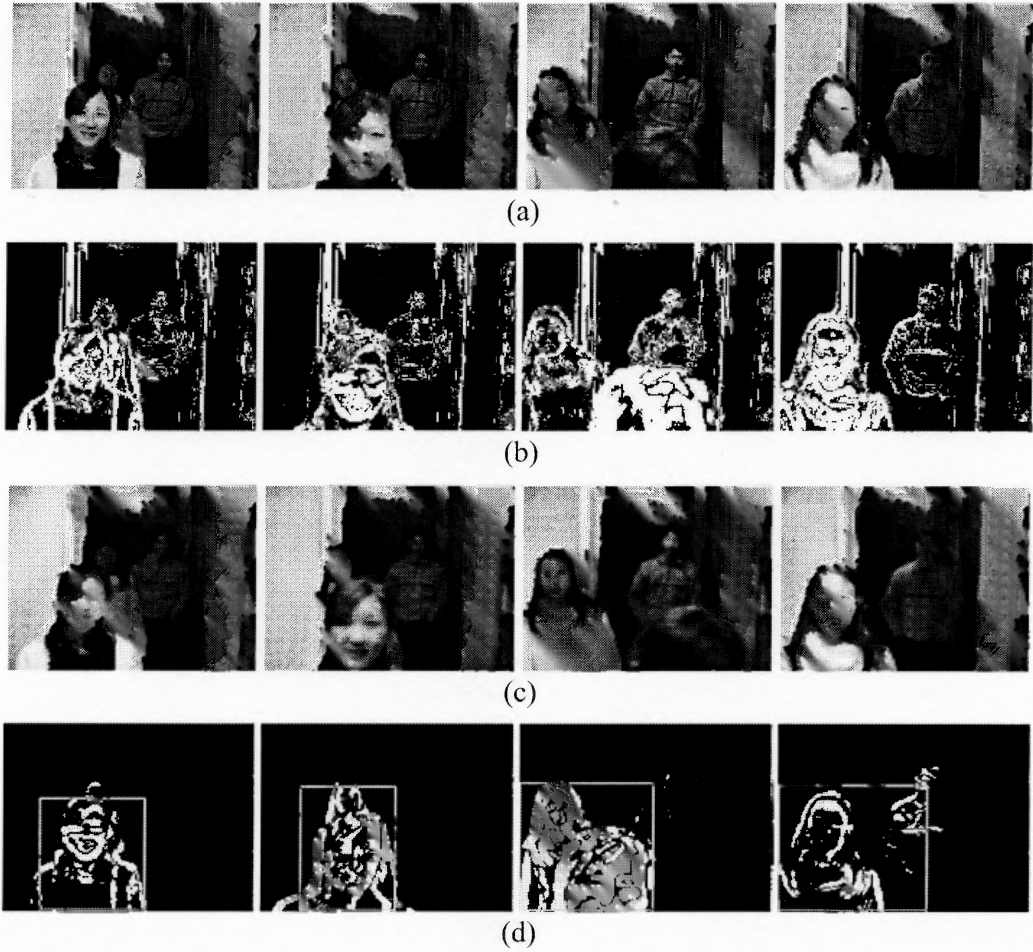
### 2.2.1 Motion Analysis

The motion analysis adopts the temporal differencing technique for the FOI derivation. Temporal differencing first calculates the absolute difference between two consecutive frames, and then applies a threshold,  $\tau$ , to classify each pixel to either motion or non-motion. Let  $\mathbf{Y}_t(i, j) \in \mathbb{R}^{m \times n}$  and  $\mathbf{Y}_{t+1}(i, j) \in \mathbb{R}^{m \times n}$  be two intensity images representing the  $Y$  components of the consecutive frames, the difference image after thresholding,  $\Delta\mathbf{Y}(i, j) \in \mathbb{R}^{m \times n}$ , is computed as follow [88]:

$$\Delta\mathbf{Y}(i, j) = \begin{cases} 1, & \text{if } |\mathbf{Y}_{t+1}(i, j) - \mathbf{Y}_t(i, j)| \geq \tau \\ 0, & \text{otherwise} \end{cases} \quad (2.1)$$

$$1 \leq i \leq m,$$

$$1 \leq j \leq n$$



**Figure 2.2** Applying a  $5 \times 5$  Gaussian smoothing filter eliminates undesirable false detections caused by dynamic environmental changes, such as illumination variation.

where  $\tau$  is the threshold. Finally, horizontal and vertical profiles of  $\Delta Y(i, j)$  are computed to define the motion FOI. Figure 2.1 (a) illustrates the derivation of horizontal and vertical profiles of the differencing image. The white pixels denote the motion pixels, and black pixels are the non-motion pixels. Four thresholds (corresponding to the four yellow lines) are set to cut off insignificant motion area. The resulting motion FOI is shown in Figure 2.1 (b) by a yellow rectangle.

Temporal differencing, however, is sensitive to dynamic environmental changes, such as illumination variation, and fails to locate a motion FOI when subject stops moving. Fig-

Figure 2.2 (a) shows four gray-scale video frames corresponding to the  $Y$  components in the  $YC_bC_r$  color space. Figure 2.2 (b) displays the difference images after thresholding. Due to unstable lightening, Figure 2.2 (b) shows many undesirable false detections caused by factors not related to motion. As a result, the motion FOI does not define a compact area, but rather incorporate many background regions. To address this sensitivity issue, a Gaussian smoothing filter is applied to each frame before calculating the difference image. Figure 2.2 (c) shows the corresponding video frames after applying a  $5 \times 5$  Gaussian smoothing filter, and Figure 2.2 (d) displays the difference images, where yellow rectangles denote motion FOIs.

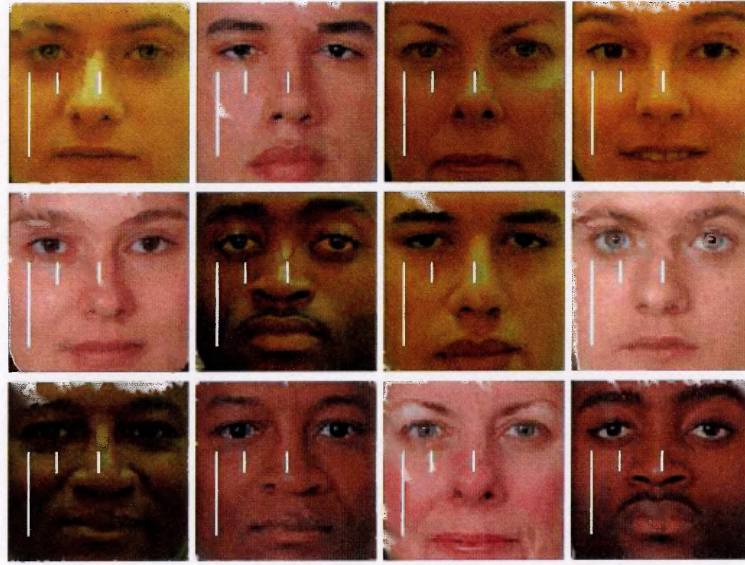
To detect a motion FOI when subject stops moving, a heuristic procedure is applied to obtain the last-known motion FOI as the current motion FOI if no significant motion is detected [88]:

$$IsMotionDetected = \begin{cases} Yes, & \text{if } \sum_{i=1}^m \sum_{j=1}^n \Delta Y(i, j) > \varepsilon \\ No, & \text{otherwise} \end{cases} \quad (2.2)$$

where  $\varepsilon$  is a threshold. As a result, when the subject stops moving or the motion is insignificant, the last-known FOI is used.

### 2.2.2 Skin Color Segmentation

To further refine the motion FOI, a skin color segmentation procedure is applied to eliminate non-skin areas. The skin color segmentation is implemented in the  $C_b - C_r$  subspace of the  $YC_bC_r$  color space, which is chosen based on an experimental analysis. In particular, the distribution of skin color pixels in the subspaces of the  $RGB$ , the  $YC_bC_r$ , and the  $L^*a^*b^*$  color spaces is assessed. Ninety skin color pixels are collected from each of 600 color images in the FERET database Batch 15 [73]. Figure 2.3 shows example images from the FERET database, where white strips indicate the skin color pixels acquired. Figures 2.4, 2.5, and 2.6 show three-dimensional clusters and two-dimensional projections of

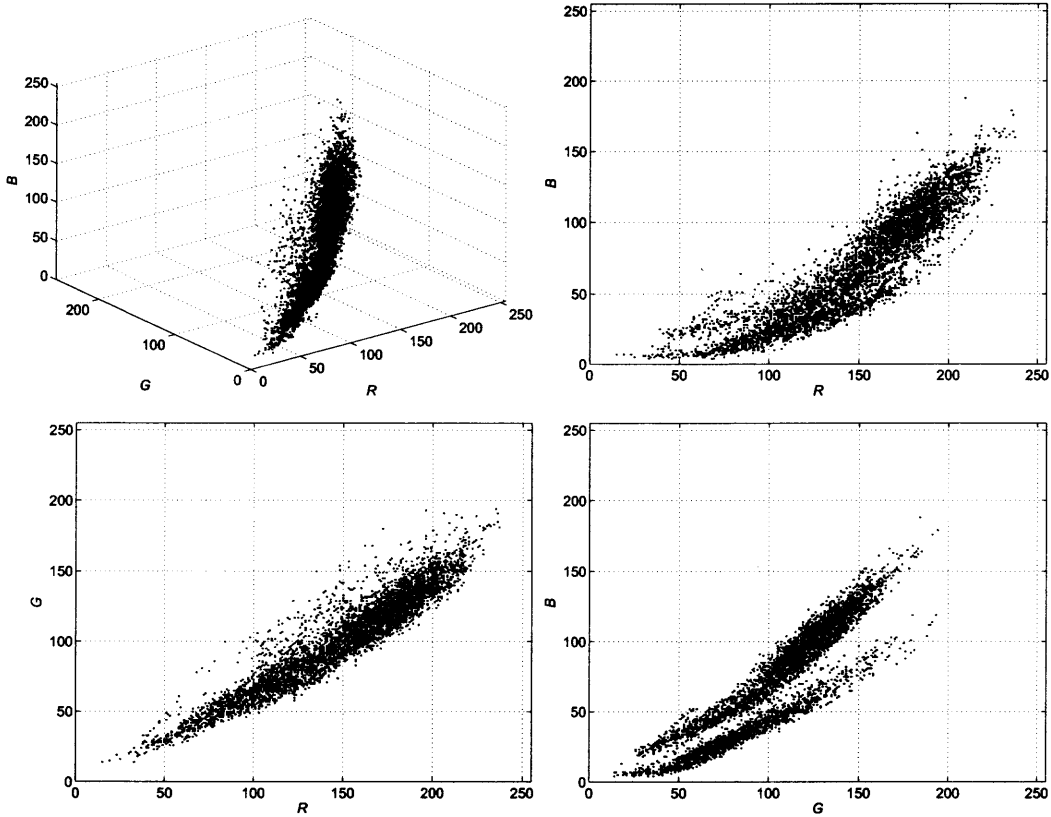


**Figure 2.3** Example face images from the FERET database. The white strips indicate the skin color pixels acquired. For each face image, ninety skin color pixels are collected.

5,400 skin color pixels in the  $RGB$ , the  $YC_bCr$ , and the  $L^*a^*b^*$  color spaces, respectively. Clearly, among the nine subspaces (i.e.,  $R - G$ ,  $R - B$ ,  $G - B$ ,  $Y - C_b$ ,  $Y - C_r$ ,  $C_b - C_r$ ,  $L^* - a^*$ ,  $L^* - b^*$ , and  $a^* - b^*$ ) assessed, the skin color pixels occupy the most compact region in the  $C_b - C_r$  subspace. The skin color segmentation is thus implemented in this subspace.

The skin color segmentation procedure consists of five steps: (i) skin color modeling, (ii) skin-non-skin classification, (iii) region filling and labeling, (iv) rectangle bounding, and (v) skin region rejection. First, an elliptic skin color model is defined due to the fact that the projection of the skin color cluster onto the  $C_b - C_r$  subspace (see Figure 2.5) approximates an ellipse. The model is then applied to distinguish foreground pixels (skin pixels) from background pixels (non-skin pixels) according to whether their  $C_b - C_r$  coordinates occur inside or outside the ellipse. After the segmentation, isolated background regions are filled with the foreground color, and each connected foreground region is labeled by a different integer. The labeled regions are then tightly bounded by rectangles. Finally, the FOIs are derived by rejecting rectangles not satisfying three geometric criteria:



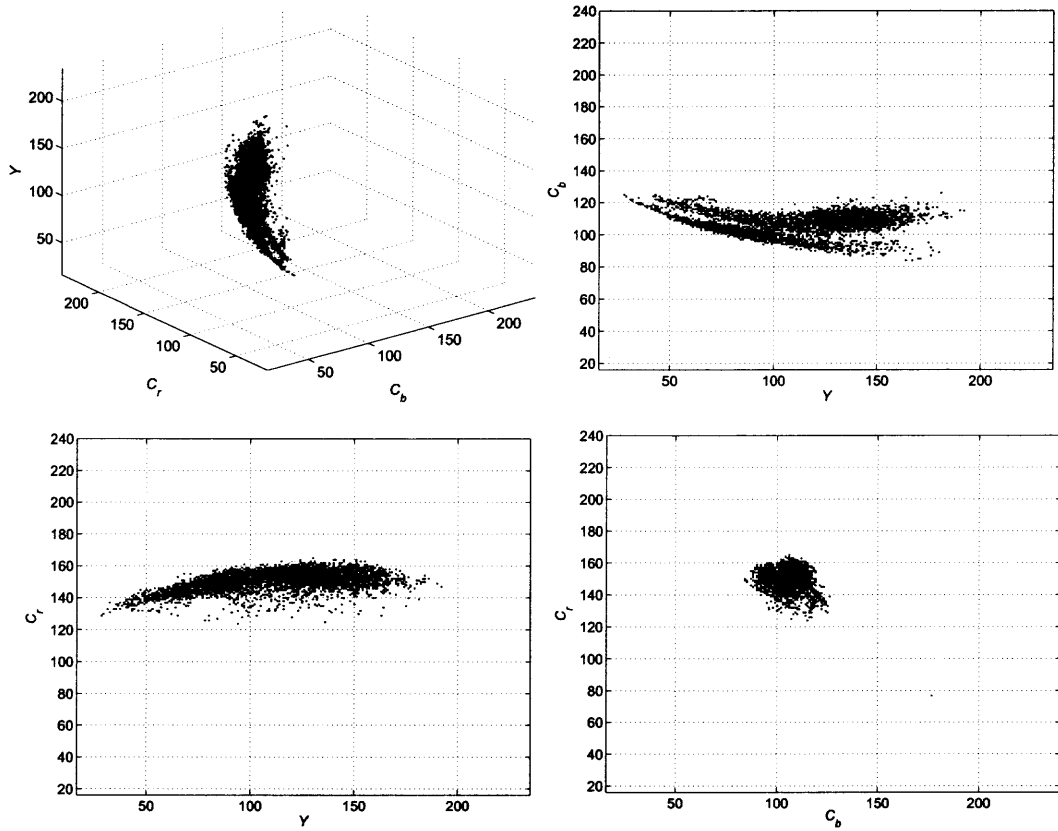


**Figure 2.4** The cluster of 5,400 skin color pixels in the  $RGB$  color space, and the projections onto the  $R - G$ , the  $R - B$ , and the  $G - B$  subspaces.

(i)  $width > 12$ , (ii)  $height > 12$ , and (iii)  $0.2 < height/width < 2$ . Figure 2.7 (b) shows the final FOIs (in red rectangles) derived from both the motion analysis (in yellow rectangles) and the skin color segmentation.

### 2.3 The DBD-SVM Face Detection Method

The DBD-SVM face detection method classifies face and non-face feature patterns by combining three classification rules. Figure 2.8 shows the three classification rules (in black diamond shapes) implementing distribution-based distance (DBD) and support vector machine (SVM). In particular, DBD is first applied to separate feature patterns into three classes: the face class, the non-face class, and the undecided class. Then an SVM classi-

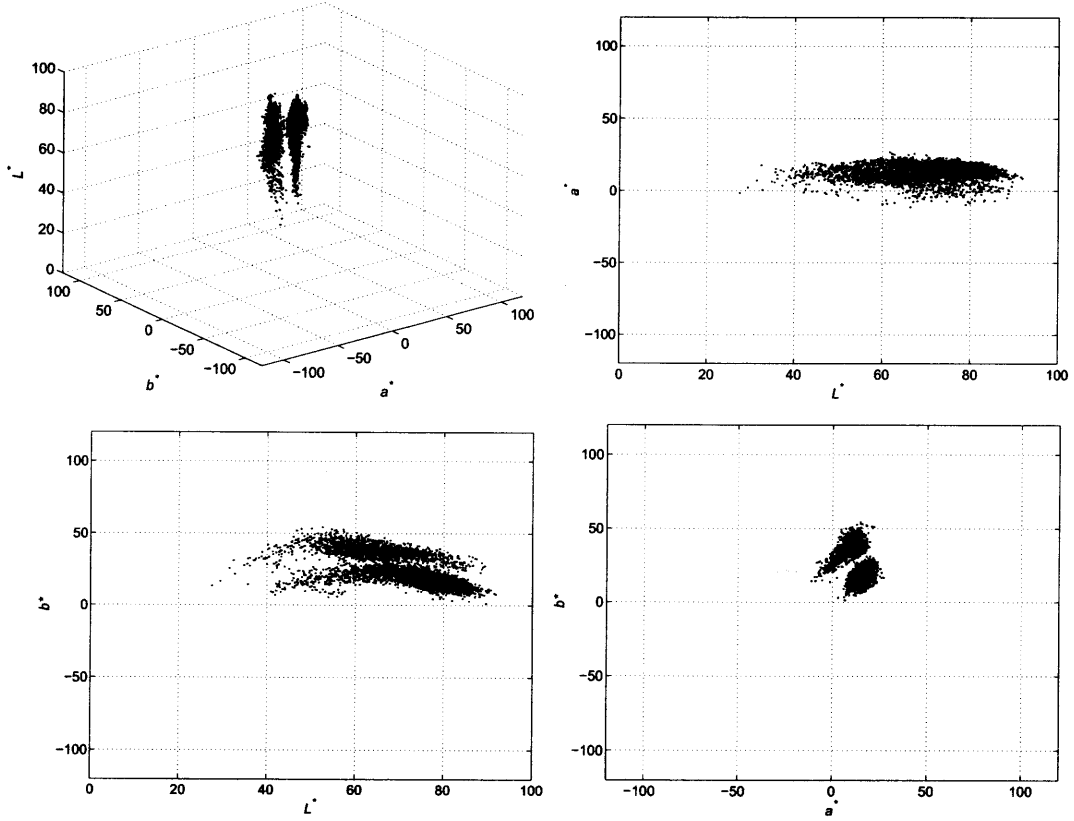


**Figure 2.5** The cluster of 5,400 skin color pixels in the  $YC_bC_r$  color space, and the projections onto the  $Y - C_b$ , the  $Y - C_r$ , and the  $C_b - C_r$  subspaces.

fier detects faces in the undecided class and passes the remaining patterns to the third rule, the DBD-SVM classification rule, which further classifies them into either the face or the non-face class. The feature extraction method, distribution-based distance, support vector machine, and the three classification rules are detailed in Subsections 2.3.1, 2.3.2, 2.3.3, and 2.3.4, respectively.

### 2.3.1 Feature Extraction for Face Detection

The discriminating features for face detection is defined by the input image, the horizontal / vertical difference images, and the row / column amplitude projections. Let  $Y(i, j) \in \mathbb{R}^{m \times n}$  represent an input image (e.g., training images for the face and the non-face classes, or



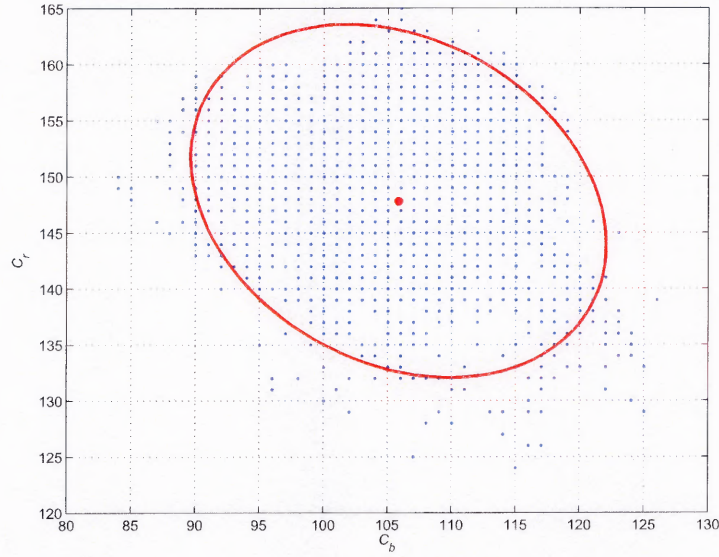
**Figure 2.6** The cluster of 5,400 skin color pixels in the  $L^*a^*b^*$  color space, and the projections onto the  $L^* - a^*$ , the  $L^* - b^*$ , and the  $a^* - b^*$  subspaces.

subimages of FOI in video frames), the vertical difference image  $\mathbf{Y}_v(i, j) \in \mathbb{R}^{m \times (n-1)}$  and the horizontal difference image  $\mathbf{Y}_h(i, j) \in \mathbb{R}^{(m-1) \times n}$  are defined by [52]:

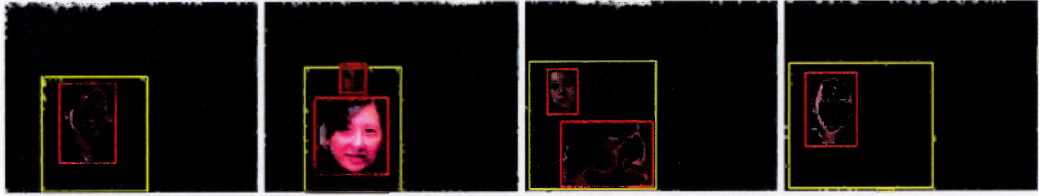
$$\begin{aligned} \mathbf{Y}_v(i, j) &= \mathbf{Y}(i, j+1) - \mathbf{Y}(i, j) & 1 \leq i \leq m, \quad 1 \leq j < n \\ \mathbf{Y}_h(i, j) &= \mathbf{Y}(i+1, j) - \mathbf{Y}(i, j) & 1 \leq i < m, \quad 1 \leq j \leq n \end{aligned} \quad (2.3)$$

The column amplitude projection  $\mathbf{x}_c \in \mathbb{R}^n$  and the row amplitude projection  $\mathbf{x}_r \in \mathbb{R}^m$  are defined by [52]:

$$\begin{aligned} \mathbf{x}_c(j) &= \sum_{i=1}^m I(i, j) & 1 \leq j \leq n \\ \mathbf{x}_r(i) &= \sum_{j=1}^n I(i, j) & 1 \leq i \leq m \end{aligned} \quad (2.4)$$



(a)



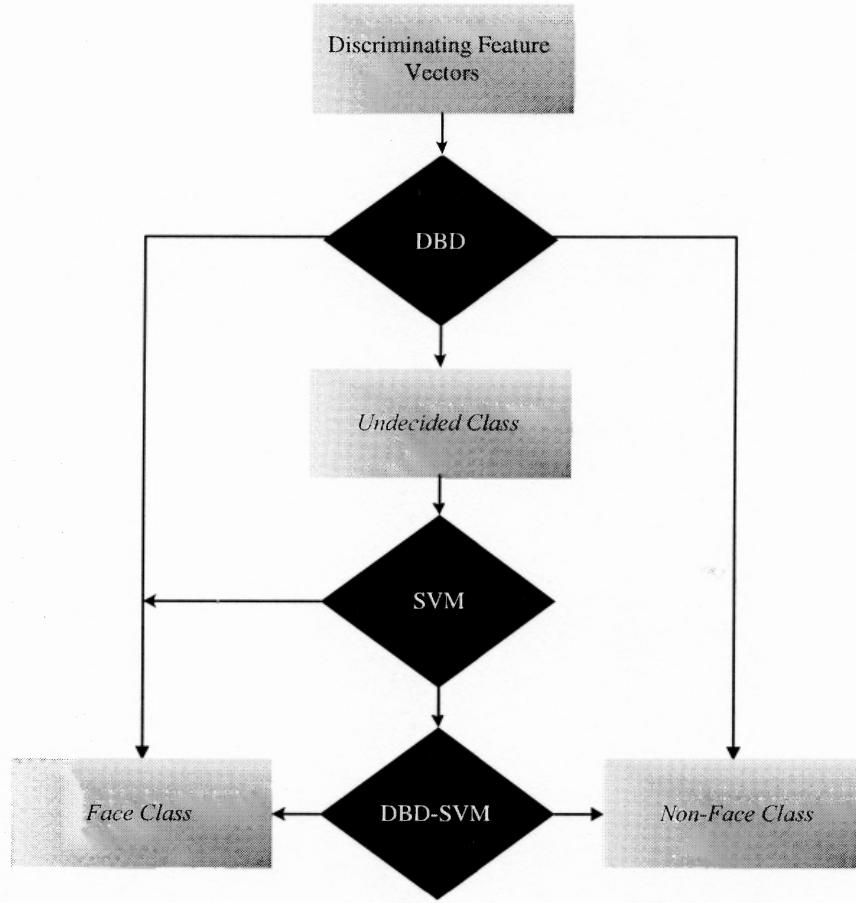
(b)

**Figure 2.7** (a) An elliptic skin color model. (b) Skin color segmentation in the motion FOI.

The three matrices:  $\mathbf{Y}(i, j)$ ,  $\mathbf{Y}_v(i, j)$ ,  $\mathbf{Y}_h(i, j)$ , and two vectors:  $\mathbf{x}_c$ ,  $\mathbf{x}_r$  are then normalized to zero mean and unit variance, respectively. Finally, the normalized matrices and vectors are concatenated to form a new feature vector  $\mathbf{y} \in \mathbb{R}^N$ ,  $N = 3mn$  to define the discriminating feature vector for face detection.

### 2.3.2 Distribution-Based Distance

Distribution-based distance is defined based on the statistical model of the face class, i.e., the probability density function (PDF) of the face class. It is reasonable to assume the PDF to be a multivariate normal distribution when only the properly aligned upright frontal faces are modeled [63], [52]. The PDF of the face class is, therefore, modeled as follows [52],



**Figure 2.8** Three classification rules (in black diamond shapes) implementing distribution-based distance and support vector machine.

[88]:

$$p(\mathbf{y}) = \frac{1}{(2\pi)^{N/2} |\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{y} - \mathbf{m})^t \Sigma^{-1} (\mathbf{y} - \mathbf{m}) \right\} \quad (2.5)$$

where  $\mathbf{y} \in \mathbb{R}^N$  is the discriminating feature vector defined in Subsection 2.3.1,  $\mathbf{m} \in \mathbb{R}^N$  and  $\Sigma \in \mathbb{R}^{N \times N}$  are the mean vector and the covariance matrix of the face class, respectively.

Taking the natural logarithm on both sides, Equation 2.5 becomes [52], [88]:

$$\ln[p(\mathbf{y})] = -\frac{1}{2} \{ (\mathbf{y} - \mathbf{m})^t \Sigma^{-1} (\mathbf{y} - \mathbf{m}) + N \ln(2\pi) + \ln|\Sigma| \} \quad (2.6)$$

The covariance matrix,  $\Sigma$ , can be factorized into the following form using principal component analysis (PCA) [25]:

$$\Sigma = \Phi \Lambda \Phi^t \text{ with } \Phi \Phi^t = \Phi^t \Phi = \mathbf{I}_N, \Lambda = \text{diag} \{ \lambda_1, \lambda_2, \dots, \lambda_N \} \quad (2.7)$$

where  $\Phi \in \mathbb{R}^{N \times N}$  is an orthogonal eigenvector matrix,  $\Lambda \in \mathbb{R}^{N \times N}$  is a diagonal eigenvalue matrix with diagonal elements (the eigenvalues) in decreasing order ( $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N$ ), and  $\mathbf{I}_N \in \mathbb{R}^{N \times N}$  is an identity matrix. The principal components are defined by the following vector,  $\mathbf{z} \in \mathbb{R}^N$  [52], [88]:

$$\mathbf{z} = \Phi^t (\mathbf{y} - \mathbf{m}) \quad (2.8)$$

where the Euclidean norm is preserved after the projection, i.e.,  $\|\mathbf{z}\|^2 = \|\mathbf{y} - \mathbf{m}\|^2$ . It then follows from Equations 2.6, 2.7, and 2.8 that [52], [88]:

$$\ln[p(\mathbf{y})] = -\frac{1}{2} \{ \mathbf{z}^t \Lambda^{-1} \mathbf{z} + N \ln(2\pi) + \ln|\Lambda| \} \quad (2.9)$$

Due to the optimal signal reconstruction property of PCA, only the first  $M$  ( $M \ll N$ ) principal components are used to estimate the probability density function. A model developed by Moghaddam and Pentland [63] is further adopted to estimate the remaining  $N - M$  eigenvalues,  $\lambda_{M+1}, \lambda_{M+2}, \dots, \lambda_N$ , using the average of those values [63]:

$$\rho = \frac{1}{N - M} \sum_{k=M+1}^N \lambda_k \quad (2.10)$$

Following from Equations 2.9 and 2.10, the probability density function is rewritten as [52], [88]:

$$\ln[p(\mathbf{y})] = -\frac{1}{2} \left\{ \sum_{i=1}^M \frac{z_i^2}{\lambda_i} + \frac{\|\mathbf{y} - \mathbf{m}\|^2 - \sum_{i=1}^M z_i^2}{\rho} + \ln \left( \prod_{i=1}^M \lambda_i \right) + (N - M) \ln \rho + N \ln(2\pi) \right\} \quad (2.11)$$

where  $z_i$ 's are the elements of  $\mathbf{z}$  defined by Equation 2.8.

Note that in this parameterization, the last three terms,  $\ln\left(\prod_{i=1}^M \lambda_i\right)$ ,  $(N - M)\ln\rho$ , and  $N\ln(2\pi)$ , are constants for all patterns, and discarded in the experimental implementations. Equation 2.11 thus defines DBD using the first  $M$  principal components, the input image, and the mean face.

### 2.3.3 Support Vector Machine

Support vector machine is a particular realization of the statistical learning theory, which describes an approach known as structural risk minimization by minimizing the risk functional in terms of both the empirical risk and the confidence interval [104]. The main idea of SVM comes from (i) nonlinear mapping of the input space to a high dimensional feature space, and (ii) optimal hyperplane design with maximal margin between the patterns of the two classes in the feature space. SVM, which displays good generalization ability, has been applied extensively for pattern classification, regression, and density estimation.

Let  $(\mathbf{x}_1, a_1), (\mathbf{x}_2, a_2), \dots, (\mathbf{x}_M, a_M)$ ,  $x_i \in \mathbb{R}^N$ , and  $a_i \in \{+1, -1\}$  be the training data in the input space, where  $a_i$  indicates the class membership of  $\mathbf{x}_i$ . Let  $\Phi$  be a nonlinear mapping between the input space ( $\mathbb{R}^N$ ) and the feature space ( $F$ ),  $\Phi : \mathbb{R}^N \rightarrow F$ , i.e.,  $x \rightarrow \Phi(x)$ . The optimal hyperplane in the feature space is defined as follows [104]:

$$\mathbf{w} \cdot \Phi(\mathbf{x}) + b = 0 \quad (2.12)$$

It can be proven [104] that the weight vector  $\mathbf{w}$  is a linear combination of support vectors that satisfy  $a_i(\mathbf{w} \cdot \Phi(\mathbf{x}_i) + b) = 1$  [104]:

$$\mathbf{w} = \sum_{\mathbf{x}_i \in \text{support vectors}} a_i \alpha_i \Phi(\mathbf{x}_i) \quad (2.13)$$

From Equations 2.12 and 2.13, a linear decision function in the feature space is defined [104]:

$$f(\mathbf{x}) = \text{sign} \left( \sum_{\mathbf{x}_i \in \text{support vectors}} a_i \alpha_i \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) + b \right) \quad (2.14)$$

where the coefficients,  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_M)^t$ , are determined by maximizing the following functional [104]:

$$W(\alpha) = \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j a_i a_j \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \quad (2.15)$$

subject to the following constraints [104]:

$$\sum_{i=1}^M \alpha_i a_i = 0, \quad \alpha_i \geq 0, \quad i = 1, 2, \dots, M. \quad (2.16)$$

Note that the decision function (see Equation 2.14) is defined on the dot products in the high dimensional feature space, where computation might be prohibitively expensive. SVM, however, manages to compute the dot products by means of kernel function [104]:

$$K(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y}) \quad (2.17)$$

Three classes of kernel functions widely used in kernel classifiers, neural networks, and support vector machines are polynomial kernels, Gaussian kernels, and sigmoid kernels [104]:

$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y})^d \quad (2.18)$$

$$K(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{y}\|^2}{2\sigma^2}\right) \quad (2.19)$$

$$K(\mathbf{x}, \mathbf{y}) = \tanh(\kappa \mathbf{x} \cdot \mathbf{y} + \vartheta) \quad (2.20)$$

where  $d \in \mathbb{N}$ ,  $\sigma > 0$ ,  $\kappa > 0$ , and  $\vartheta < 0$ .



### 2.3.4 Classification Rules

The DBD-SVM face detection method first applies DBD to detect faces that are close to the face class and exclude patterns that are far away from the face class. This classification rule thus separates all the patterns in the FOIs into three classes: the face class,  $\omega_f$ , the non-face class,  $\omega_n$ , and the undecided class,  $\omega_u$  [88]:

$$\mathbf{y} \in \begin{cases} \omega_f, & \text{if } \ln[p(\mathbf{y})] \geq \tau_f \\ \omega_u, & \text{if } \tau_n < \ln[p(\mathbf{y})] < \tau_f \\ \omega_n, & \text{otherwise} \end{cases} \quad (2.21)$$

where  $\mathbf{y}$  is the discriminating feature vector derived from an input pattern,  $\ln[p(\mathbf{y})]$  is the distribution-based distance defined by Equation 2.11, and  $\tau_f$  and  $\tau_n$  are thresholds.

The DBD-SVM face detection method then applies an SVM classifier to detect faces in the  $\omega_u$  class [88]:

$$\mathbf{y} \in \begin{cases} \omega_f, & \text{if } f(\mathbf{y}) > 0 \\ \omega_n, & \text{otherwise} \end{cases} \quad (2.22)$$

where  $f(\mathbf{y})$  is the decision function of the SVM classifier defined by Equation 2.14. The experimental results, however, show that the SVM classifier alone can not detect all the faces in the  $\omega_u$  class, and some face patterns are misclassified to the  $\omega_n$  class.

To further improve the face detection performance, a new classification rule, the DBD-SVM classification rule, is designed to check the patterns in the  $\omega_u$  class [88]:

$$\mathbf{y} \in \begin{cases} \omega_f, & \text{if } (g(\mathbf{y}) > \tau_s) \text{ and } (\ln[p(\mathbf{y})] + cg(\mathbf{y}) > \tau_t) \\ \omega_n, & \text{otherwise} \end{cases} \quad (2.23)$$

where  $c$  is a constant controlling parameter,  $\tau_s$  and  $\tau_t$  are thresholds, and  $g(\mathbf{y})$  is the decision function of the SVM classifier without the sign function (see Equation 2.14) [88]:

$$g(\mathbf{y}) = \sum_{\mathbf{x}_i \in \substack{\text{support} \\ \text{vectors}}} a_i \alpha_i \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{y}) + b \quad (2.24)$$

Note that  $\tau_s$  in Equation 2.23 is a negative constant, and the functionality of the first term,  $g(\mathbf{y}) > \tau_s$ , is to select face candidates misclassified to the  $\omega_n$  class by the SVM classifier (see Equation 2.22). The second term,  $\ln[p(\mathbf{y})] + cg(\mathbf{y}) > \tau_i$ , then further determines the true faces from the face candidates by applying both classifiers.

The three classification rules detailed above actually apply a coarse-to-fine classification strategy in the sense that they are cascaded in an increasing order of detection accuracy and decreasing order of computational efficiency. In terms of detection accuracy, the third classification rule detects faces more accurately than the first two rules do because it combines the classification power of the distribution-based distance and SVM. In terms of computational efficiency, the first classification rule runs faster than the second and the third ones due to the fact that the term  $\ln[p(\mathbf{y})]$  is evaluated by estimating the first  $M$  principal components ( $M \ll N$ ) of the input vector  $\mathbf{y}$ , while the term  $g(\mathbf{y})$  is obtained directly in the original input space,  $\mathbb{R}^N$ .

## 2.4 Experiments

This section details the training of classifiers, the learning of thresholds, and the scanning procedure, and presents the face detection results in both still images and video streams.

### 2.4.1 Training the Classifiers

The training of the DBD-SVM method includes three steps: (i) face class modeling, (ii) non-face images deriving, and (iii) SVM learning. First, the DBD-SVM method models the face class ( $\omega_f$ ) as a multivariate normal distribution and defines the distribution-based distance (see Subsection 2.3.2) using 600 frontal face images from the FERET database Batch 15 [73]. Note that the mirror images of the training data set are also included, hence the total training images for the face class modeling is 1,200. Second, non-face training images are derived by choosing subimages from 14 natural scene images that do not contain any face at all. Those subimages that lie close to the face class are chosen as

non-face training images [88]:

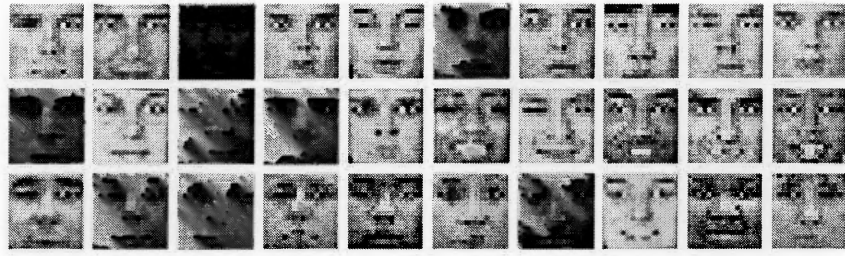
$$\ln[p(\mathbf{y})] > \tau_n \quad (2.25)$$

where  $\ln[p(\mathbf{y})]$  is DBD defined by Equation 2.11 and  $\tau_n$  is the threshold in Equation 2.21. In particular, 3,813 non-face training images are chosen from more than two million subimages of the 14 natural scene images to form the training data set for the non-face class. Note that both face and non-face images are normalized to a standard spatial resolution of  $16 \times 16$ . Figure 2.9 (a) shows example face training images that are normalized to  $16 \times 16$ , and Figure 2.9 (b) shows example non-face training images derived from a natural scene image. Note that the non-face images in Figure 2.9 (b) display different sizes, which correspond to different scales of the original natural scene image when the non-face images are derived. The spatial resolution of all the non-face images, however, is the same,  $16 \times 16$ . Finally, the training data, 1,200 faces and 3,813 non-faces, are applied to train an SVM classifier with a polynomial kernel of degree two for face detection.

Figure 2.10 shows the image representations of the mean feature vectors of the face and the non-face classes. In particular, the first images in Figure 2.10 (a) and Figure 2.10 (b) are the mean face and the mean non-face, respectively. The second images are their vertical difference images, the third images are their horizontal difference images, and the last two bar graphs are column and row amplitude projections. Note that the images and projections in Figure 2.10 (b) resemble their counterparts in Figure 2.10 (a) because the 3,813 non-face training images lie close to the face class.

#### 2.4.2 Learning the Thresholds

Thresholds play important roles in the DBD-SVM face detection method as a change of a single threshold may affect the detection performance significantly. This section describes a general procedure to fine-tune the four thresholds,  $\tau_n$ ,  $\tau_f$ ,  $\tau_s$ , and  $\tau_t$ , defined in Section 2.3.



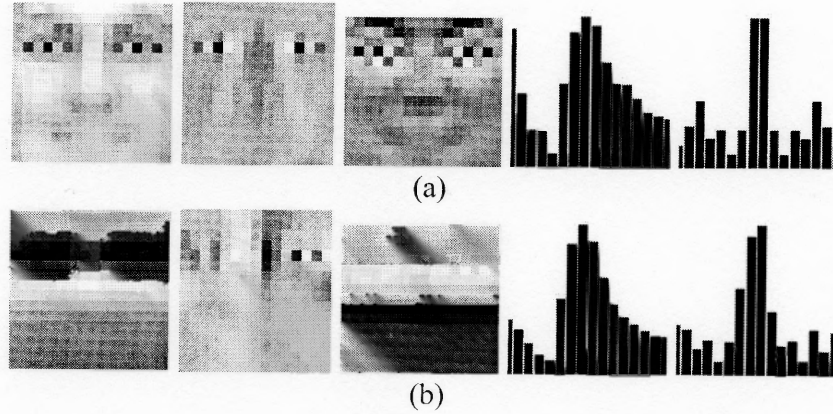
(a)



(b)

**Figure 2.9** Some examples of the training face and non-face images. (a) Examples of the face training images. (b) Examples of the non-face training images.

The learning of thresholds starts with  $\tau_n$  (see Equation 2.25) as it directly determines how many non-face training images will be derived from the 14 natural scene images using distribution-based distance. Figure 2.11 (a) plots the relationship between the total number of non-face subimages and the threshold  $\tau_n$ , where the horizontal axis indicates the value of  $\tau_n$  and the vertical axis is the total number of non-face subimages derived from all the 14 natural scene images used in the experiments. Note that in order to prevent one scene image from unduly dominating the others in generating the non-face subimages, the number of

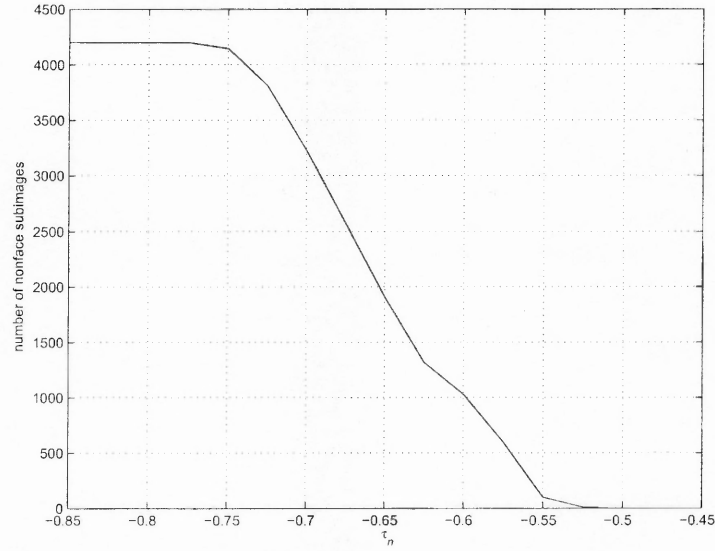


**Figure 2.10** Image representations of the mean feature vectors of the face and the non-face classes. (a) The first image is the mean face, the second and the third images are its vertical and horizontal difference images, and the last two bar graphs are its column and row amplitude projections. (b) The first image is the mean non-face, the second and the third images are its vertical and horizontal difference images, and the last two bar graphs are its column and row amplitude projections.

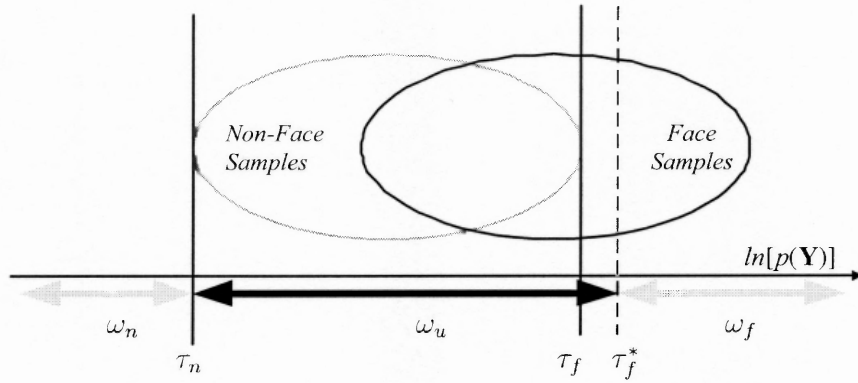
non-face subimages for each scene is capped at 300. As a result, the maximum number of non-face subimages derived from all the scene images should be 4,200, which is shown in Figure 2.11 (a) by the flat curve segment. The choice of  $\tau_n$  is thus close to the threshold value that leads to this flat curve segment. In particular, the chosen threshold  $\tau_n$  generates 3,813 non-face subimages from the 14 natural scene images, and these subimages form the non-face training set for the DBD-SVM face detection method.

After generating the non-face images, a reasonable choice of the threshold,  $\tau_f$ , is the largest distribution-based distance of all these non-face images (see Figure 2.11 (b)). However, Equation 2.21 is applied for coarse detection, whose purpose is to reliably classify face and non-face patterns while leaving difficult patterns in the undecided class. The threshold is thus increased from  $\tau_f$  to  $\tau_f^*$  (see Figure 2.11 (b)) in order to reduce the number of false detections.

The thresholds,  $\tau_s$  and  $\tau_t$ , in Equation 2.23 are used for the fine detection, whose functionality is to classify the undecided patterns into either the face class or the non-face class. The values of these two thresholds are determined through a numerical analysis. In



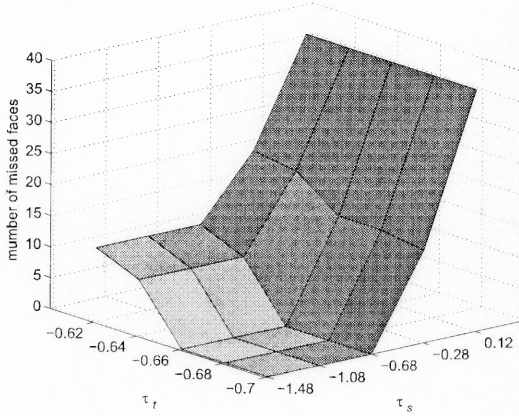
(a)



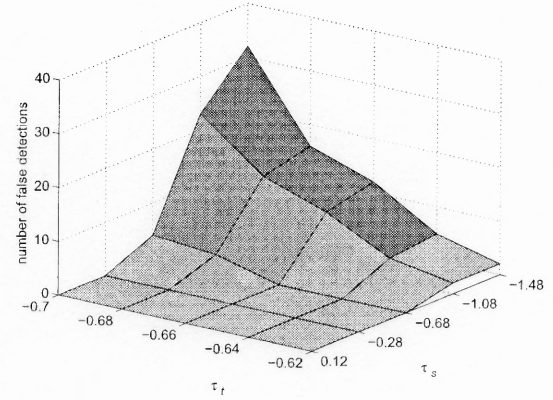
(b)

**Figure 2.11** The learning of thresholds,  $\tau_n$  and  $\tau_f$ . (a) The relationship between the total number of non-face subimages and the threshold  $\tau_n$ . (b) The threshold,  $\tau_f$ , is set to be the largest distribution-based distance of all these non-face images. However, the threshold,  $\tau_f$ , is applied for coarse detection, whose purpose is to reliably classify face and non-face patterns while leaving difficult patterns in the undecided class. The threshold is thus increased from  $\tau_f$  to  $\tau_f^*$  in order to reduce the number of false detections.

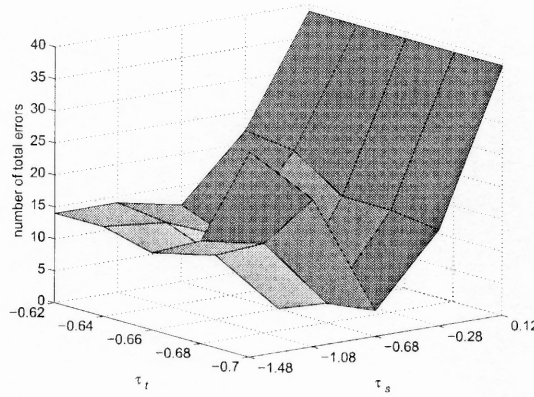
particular, let  $\mathcal{S}$  be a set of values of  $\tau_s$  and  $\mathcal{T}$  be a set of values of  $\tau_t$ , face detection performance of each combination of  $\mathcal{S} \times \mathcal{T}$  is evaluated using a subset (24 images containing 54 faces) of the MIT-CMU test sets. Note that the constant,  $c$ , defined in Equation 2.23 is fixed at 0.05 throughout the experiments.



(a)



(b)



(c)

**Figure 2.12** The learning of thresholds,  $\tau_s$  and  $\tau_t$ . (a) The number of missed faces increases when either  $\tau_s$  or  $\tau_t$  increases. (b) The number of false detections decreases when either  $\tau_s$  or  $\tau_t$  increases. (c) The number of total errors (missed faces + false detections) versus the thresholds ( $\tau_s$  and  $\tau_t$ ). The smallest total error occurs at  $\{\tau_s, \tau_t\} = \{-0.68, -0.66\}$ .

Figures 2.12 (a), (b), and (c) show the number of missed faces versus the thresholds ( $\tau_s$  and  $\tau_t$ ), the number of false detections versus the thresholds, and the number of total errors (missed faces + false detections) versus the thresholds, respectively. Figure 2.12 (a) shows that the number of missed face increases when either  $\tau_s$  or  $\tau_t$  increases. Figure 2.12 (b), however, shows that the number of false detections decreases when either  $\tau_s$  or  $\tau_t$

increases. Finally, Figure 2.12 (c) shows that the smallest total error occurs at  $\{\tau_s, \tau_t\} = \{-0.68, -0.66\}$ . The thresholds,  $\tau_s$  and  $\tau_t$ , are therefore set to be  $-0.68$  and  $-0.66$ , respectively.

### 2.4.3 Scanning Procedure

The basic search procedure of the DBD-SVM method is sliding a  $16 \times 16$  detection window across possible locations, scales, and orientations in a test image. The detection window is shifted pixel-by-pixel and the incremental scale is a factor of 1.125. Note that because the DBD-SVM method is trained using only upright frontal faces, the rotated faces are detected by rotating test images to predefined degrees, such as  $\pm 5^\circ$ ,  $\pm 10^\circ$ ,  $\pm 15^\circ$ ,  $\pm 20^\circ$ , and  $\pm 25^\circ$ .

To ease the computational burden of the scanning procedure, a heuristic is defined such that if a face is detected at a certain location, the region occupied by the face should be eliminated from being searched again. Let the search order be top to bottom then left to right, and the upper left pixel represent a  $16 \times 16$  subimage. Figure 2.13 (a) shows the idea of this heuristic. Suppose a face is detected at a point  $p$ , its  $7 \times 7$  neighborhood is searched to find a face that lies closest to the face class. Note that due to the predefined searching order, half of these neighbors have already been visited, and the remaining unvisited neighbors are the pixels inside the region  $A$ . Note that each of these twenty-four neighbors in region  $A$  defines a  $16 \times 16$  subimage, which could be a face. Suppose  $q$  defines a face that lies closest to the face class, then the 542 pixels inside the region  $B$  should not be searched because any face appears in this region will overlap the one found at  $q$ .

This heuristic also benefits face detection when the detector scans the image in multiple scales. Figure 2.13 (b) shows the idea of applying this heuristic to multi-scale detections. Suppose a face is detected at the scale  $a \times a$ . When detecting faces in another scale  $b \times b$ , the region  $R$  should not be searched because any face appears in this region will overlap the one found at the scale  $a \times a$ . Note that the area  $R$  should be shrunk by one or two pixels in order to detect closely adjacent or slightly overlapping faces as shown in



Figure 2.19 (d). Also note that the multi-scale searching in the DBD-SVM method always starts with the largest search scale and then goes toward smaller ones to eliminate as many regions as possible.

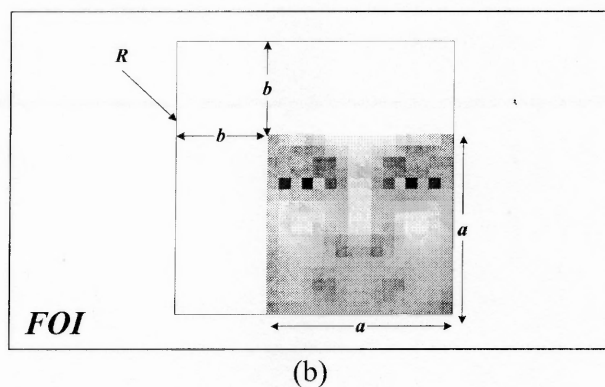
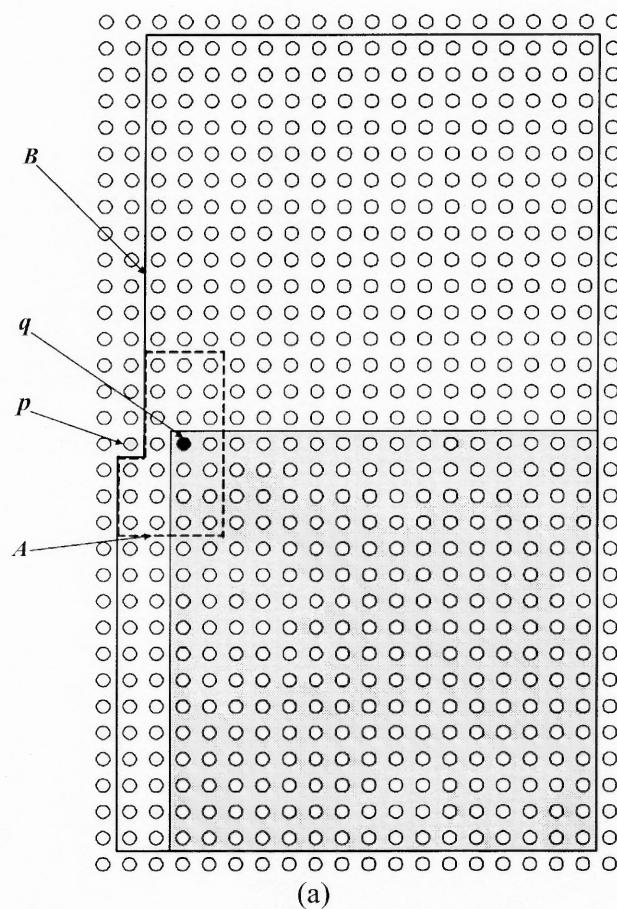
To further improve the computational efficiency, several heuristics are applied to exclude a non-face pattern in an early stage. Figure 2.14 shows a  $16 \times 16$  subimage with five labeled regions corresponding to the left eye area (A), the nose bridge area (B), the right eye area (C), the nose-cheeks area (D), and the mouth area (E). The first criterion calculates the variances in region D and E, respectively, and excludes the subimage as a face candidate if either variance is smaller than its predefined threshold. If the subimage is not excluded so far, the second criterion calculates the mean values in the region A, B, and C. Then two average values,  $m_A$  and  $m_C$ , are derived by averaging the intensity values of the pixels whose intensity values are smaller than the mean values of the regions A and C, respectively; and another average value,  $m_B$ , is computed by taking the average of the intensity values of the pixels whose intensity values are larger than the mean value of the region B. Finally, the second criterion eliminates the subimage as a face candidate if  $m_A < \kappa m_B$  or  $m_C < \kappa m_B$ , where  $\kappa$  is a control factor.

#### 2.4.4 Face Detection in Still Images

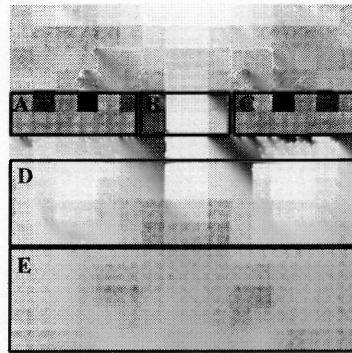
The data used to test the DBD-SVM method for face detection comes from the MIT-CMU test sets [76]. Specifically, the test set includes 92 images containing 282 faces<sup>1</sup>. As the

---

<sup>1</sup>The 92 images are listed as follows: *aerosmith-double.gif*, *albert.gif*, *Argentina.gif*, *audrey1.gif*, *audrey2.gif*, *audrybt1.gif*, *baseball.gif*, *bksomels.gif*, *blues-double.gif*, *brian.gif*, *bt301.gif*, *bwolen.gif*, *cfb.gif*, *churchill-downs.gif*, *class57.gif*, *cluttered-tahoe.gif*, *cnn1085.gif*, *cnn1160.gif*, *cnn1260.gif*, *cnn1630.gif*, *cnn1714.gif*, *cnn2020.gif*, *cnn2221.gif*, *cnn2600.gif*, *cpd.gif*, *crimson.gif*, *ds9.gif*, *ew-courtney-david.gif*, *ew-friends.gif*, *fleetwood-mac.gif*, *frisbee.gif*, *Germany.gif*, *giant-panda.gif*, *gigi.gif*, *gpripe.gif*, *harvard.gif*, *hendrix2.gif*, *henry.gif*, *jackson.gif*, *john.coltrane.gif*, *judybats.gif*, *kaari-stef.gif*, *kaari1.gif*, *kaari2.gif*, *karen-and-rob.gif*, *knex0.gif*, *knex37.gif*, *kymberly.gif*, *lacrosse.gif*, *larroquette.gif*, *madaboutyou.gif*, *married.gif*, *me.gif*, *mom-baby.gif*, *monalisa.gif*, *music-groups-double.gif*, *natalie1.gif*, *nens.gif*, *newsradio.gif*, *oksana1.gif*, *original1.gif*, *original2.gif*, *pittsburgh-park.gif*, *police.gif*, *sarah4.gif*, *sarah.live.2.gif*, *seinfeld.gif*, *shumeet.gif*, *soccer.gif*, *speed.gif*, *tahoe-and-rich.gif*, *tammy.gif*, *tommyrw.gif*, *tori-crucify.gif*, *tori-entweekly.gif*, *tori-live3.gif*, *torrance.gif*, *tp-reza-girosi.gif*, *tp.gif*, *tree-roots.gif*, *trek-trio.gif*, *trekcolr.gif*, *tress-*



**Figure 2.13** (a) A face detected at  $q$  eliminates the region  $B$  from being searched because any face appears in this region will overlap the one found at  $q$ . (b) A face detected in the scale of  $a \times a$  excludes the region  $R$  when searching in the scale of  $b \times b$ .

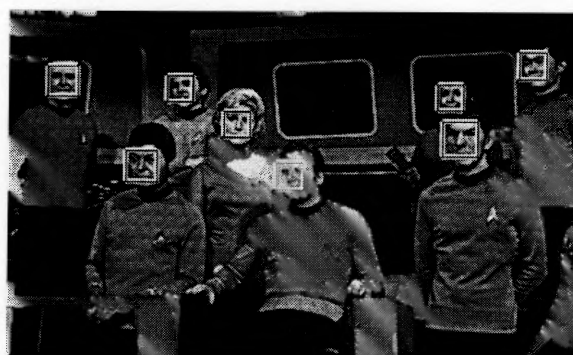


**Figure 2.14** A  $16 \times 16$  subimage with five labeled regions corresponding to the left eye area (A), the nose bridge area (B), the right eye area (C), the nose-cheeks area (D), and the mouth area (E).

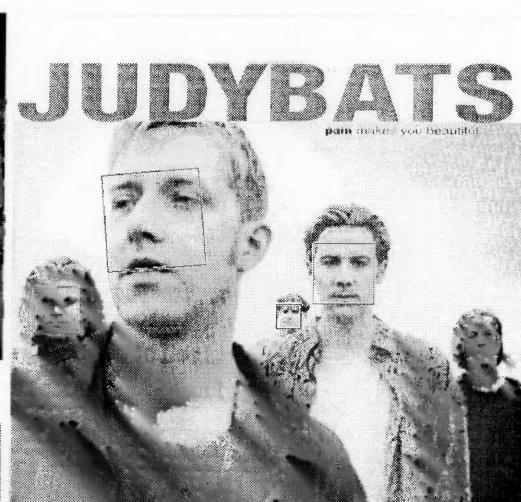
DBD-SVM method addresses detection of frontal and real human faces, those MIT-CMU test images that contain large pose-angled face, line-drawn face, poker face, masked face, or cartoon face are not included in this test set. Note that the test images used in the experiments are from diverse image sources, while the training images are from only one database, i.e., the frontal face images of the FERET Batch 15 [73]. The test data is thus able to test the generalization performance of the DBD-SVM method.

Figure 2.15 (a) shows the detection results of searching from  $12 \times 12$  to  $120 \times 120$ . Note that even though all faces appear in Figure 2.15 (a) are smaller than  $20 \times 20$ , the DBD-SVM method searches a wide spectrum of scales without leading to any false detection. Similarly, Figure 2.15 (b) through (e) display some other examples of multiple face detection in different scales. Note that Figures 2.15 (b), (c), and (e) show face detection performance of the DBD-SVM method in low contrast images, while Figure 2.15 (d) shows face detection performance with some slightly pose-angled faces. All the faces in Figure 2.15 are successfully detected by the DBD-SVM method.

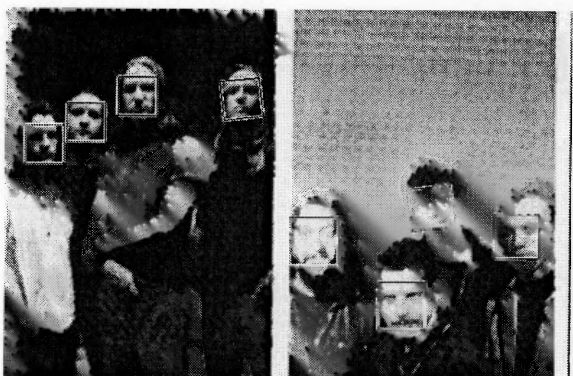
Figure 2.16 shows some examples of detecting multiple frontal faces across different rotation angles. All the faces in Figure 2.16 are correctly detected, but there is one false detection in Figure 2.16 (a). The false detection occurs because the configuration of that



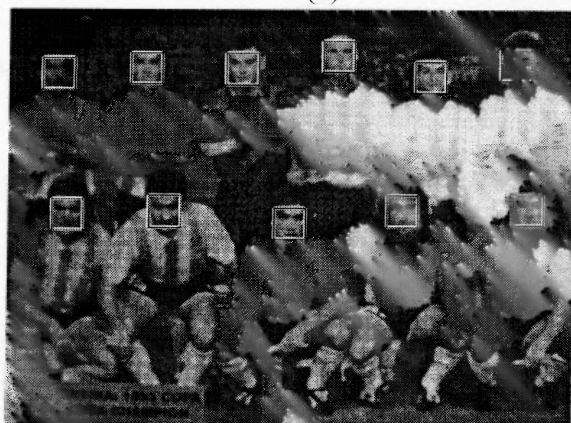
(a)



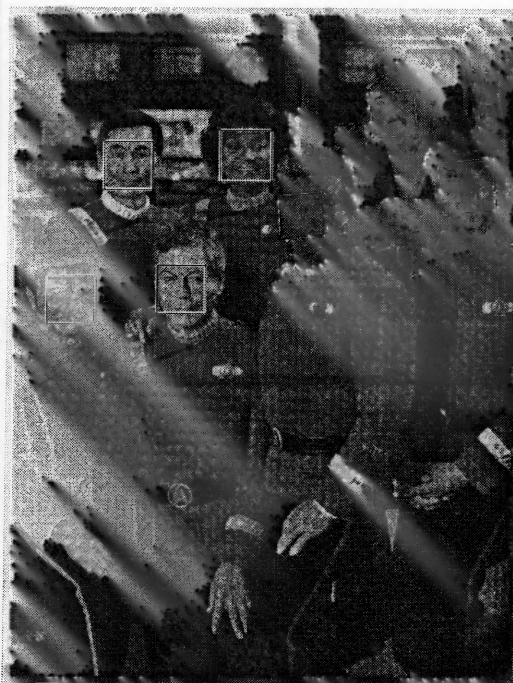
(d)



(b)

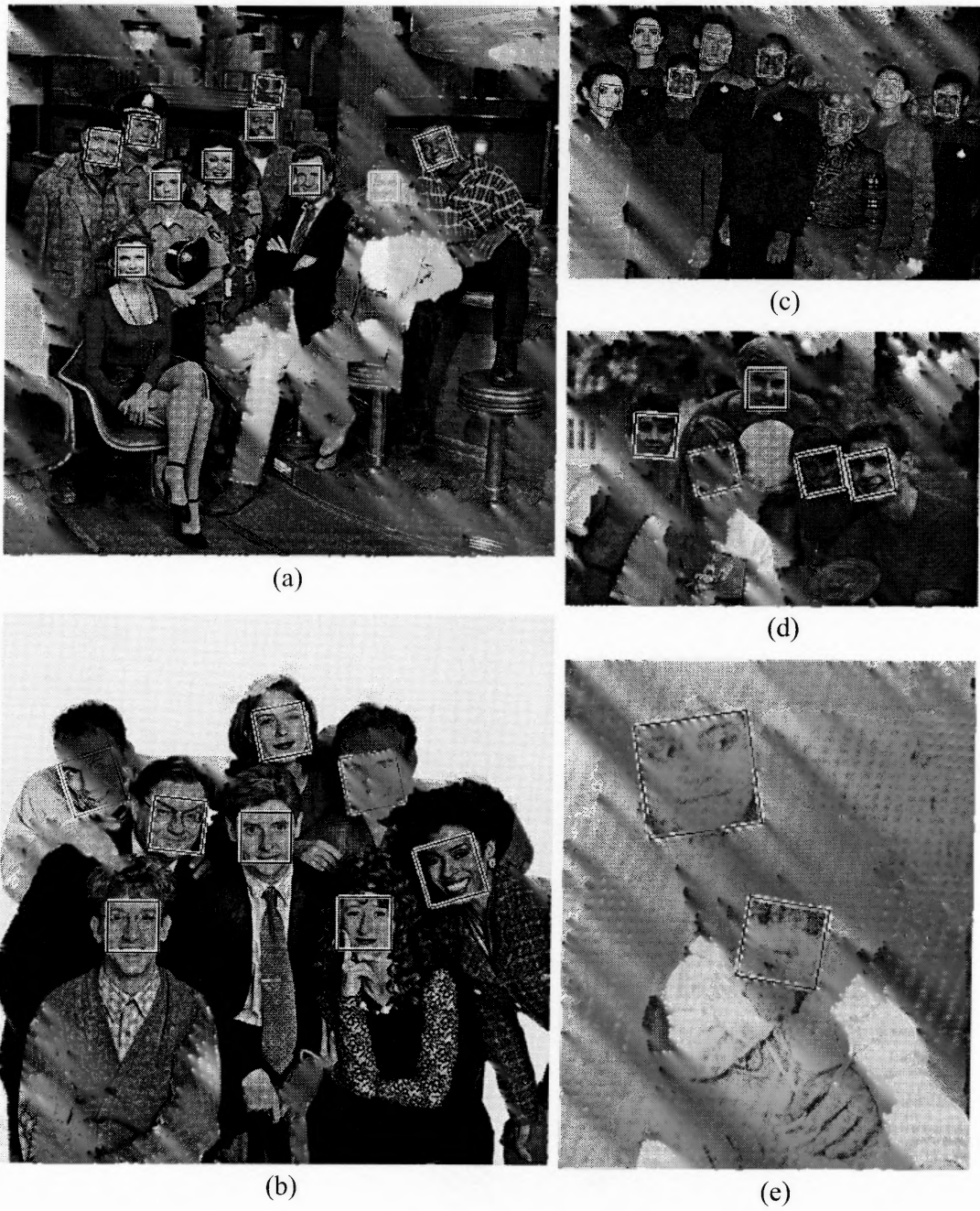


(c)



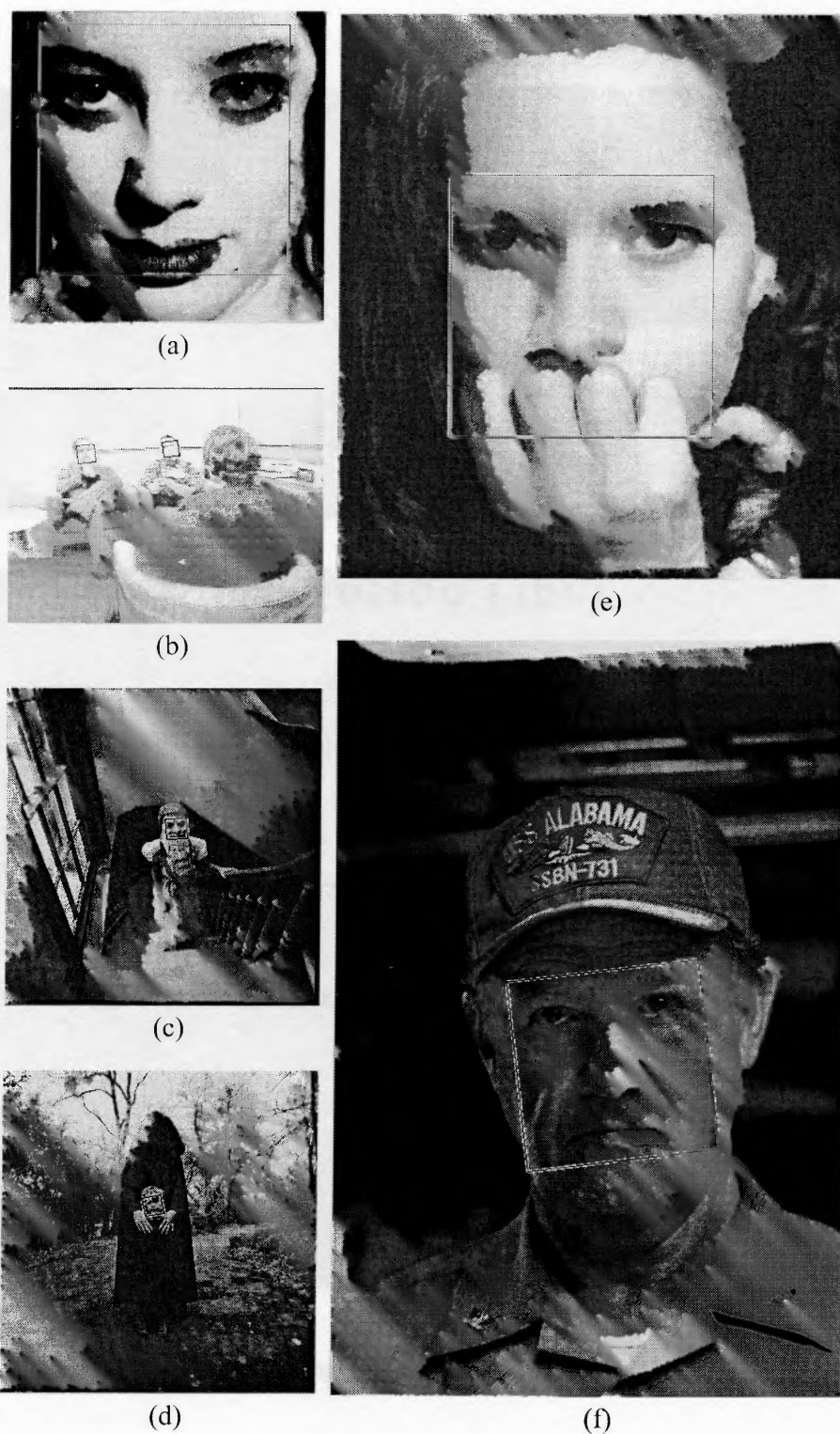
(e)

**Figure 2.15** Examples of detecting multiple faces using the DBD-SVM method.



**Figure 2.16** Examples of detecting rotated faces using the DBD-SVM method.





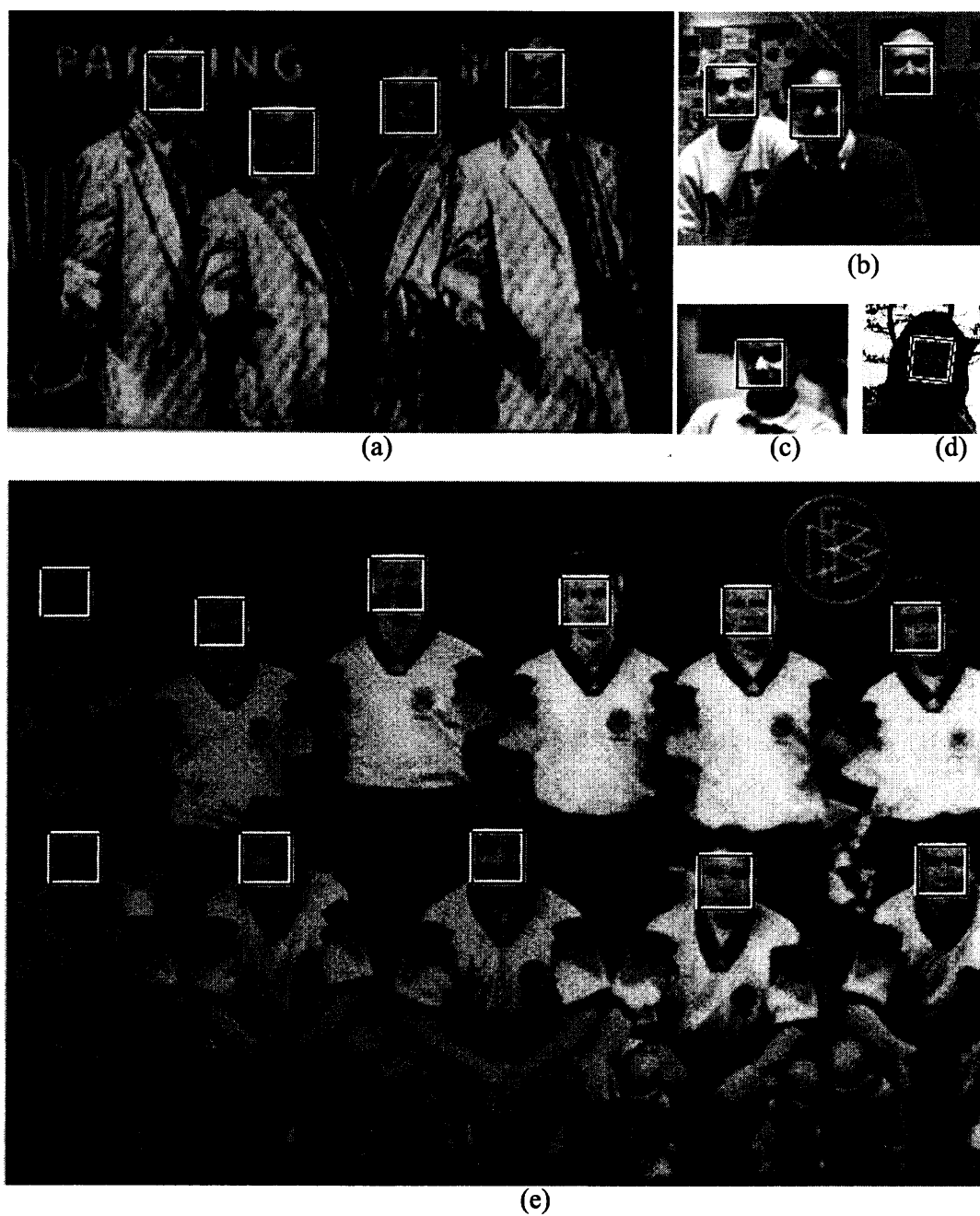
**Figure 2.17** Examples of detecting faces that are either very large or very small using the DBD-SVM method.

pattern resembles a human face. Note that because the DBD-SVM face detection method is trained using only upright frontal faces, the rotated faces are detected by rotating test images to predefined degrees, such as  $\pm 5^\circ$ ,  $\pm 10^\circ$ ,  $\pm 15^\circ$ ,  $\pm 20^\circ$ , and  $\pm 25^\circ$ . For example, in Figure 2.16 (a), the DBD-SVM face detection method searches from  $21 \times 21$  through  $210 \times 210$  with  $15^\circ$  and  $-25^\circ$  rotation angles.

Figure 2.17 shows examples of detecting faces that are either very large or very small. Figure 2.16 (a), for example, displays the detection of the largest face in the test set ( $360 \times 360$ ) and Figure 2.17 (b) shows the detection of the smallest face in the test set ( $13 \times 13$ ). Faces in Figure 2.17 (c) through (f) are detected at  $16 \times 16$ ,  $16 \times 16$ ,  $290 \times 290$ , and  $210 \times 210$ , respectively.

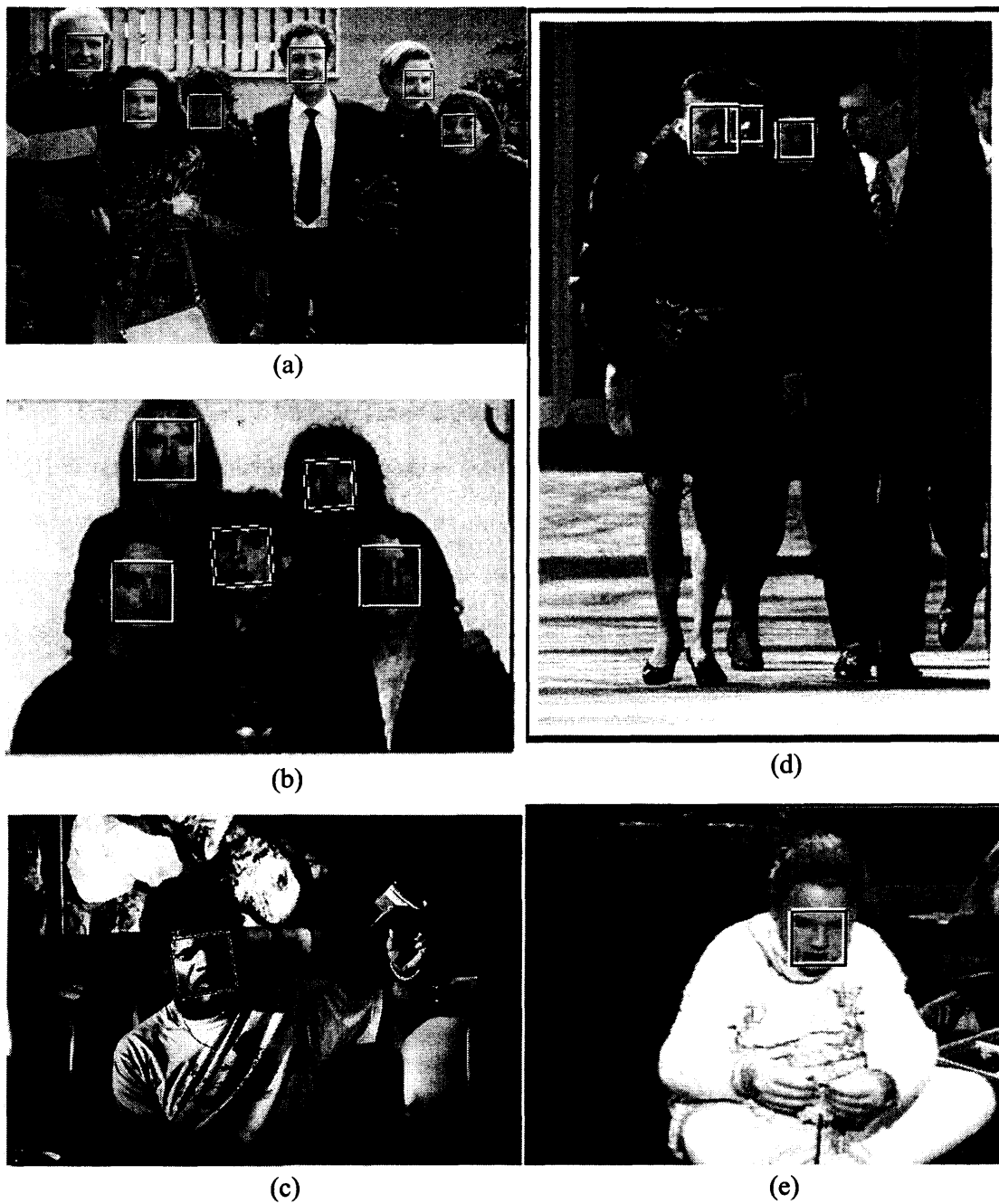
Figure 2.18 shows examples of detecting faces in low quality images. Again, all the faces are correctly detected by the DBD-SVM method. In particular, Figure 2.18 (a) displays detection of faces with dark glasses; Figure 2.18 (b) through (e) show detection of faces with blurred facial features due to the poor quality of the images.

Figure 2.19 shows examples of detecting faces with illumination and slight pose variations. Specifically, Figures 2.19 (a), (b), and (e) show images containing slightly pose-angled faces, and the DBD-SVM method successfully detects all the faces in these images. However, the DBD-SVM method can not detect faces with large pose variations, such as the large pose-angled face in Figure 2.19 (d). The reason of such missed detection is that the DBD-SVM method is trained only on upright frontal face images, i.e., 600 FERET frontal face images, which do not include any pose-angled faces. Figure 2.19 (c) shows an image with uneven lighting, and the face in this image displays one side brighter than the other side. Still, the face is successfully detected by the DBD-SVM method. The experimental results shown in Figure 2.19 demonstrate that the DBD-SVM method, which is trained on a simple image set yet works on much more complex images, displays robust generalization performance.



**Figure 2.18** Examples of detecting faces in low quality images using the DBD-SVM method.





**Figure 2.19** Examples of detecting faces with illumination and slight pose variations using the DBD-SVM method.

**Table 2.1** Comparative Face Detection Performance of the Schneiderman-Kanade Method and DBD-SVM

method	faces detected	false detections	detection rate
Schneiderman-Kanade method (1.0, 1.0)	271	41	96.1%
Schneiderman-Kanade method (2.0, 2.0)	264	5	93.6%
Schneiderman-Kanade method (3.0, 3.0)	255	1	90.4%
the DBD-SVM method	277	2	98.2%

Among the state-of-the-art face detection methods, the Schneiderman-Kanade method [84] is publicly available: <http://vasc.ri.cmu.edu/cgi-bin/demos/fndface.cgi>. The detection performance of DBD-SVM is thus compared against this method. Note that the Schneiderman-Kanade method has two thresholds, the frontal detection threshold and the profile detection threshold, which control the number of faces detected and the number of false detections. Table 2.1 shows the comparative face detection performance of the Schneiderman-Kanade method and the DBD-SVM method. Note that the two numbers in the parentheses correspond to the frontal detection threshold and the profile detection threshold, respectively. Experimental results show that the Schneiderman-Kanade method achieves 96.1% face detection rate with 41 false detections when the thresholds are set to be 1.0. Note that the number of false detections of the Schneiderman-Kanade method counted here only refers to the frontal face false detections, and it does not include the false detections caused by profile face detection. Also note the face detection rate of the Schneiderman-Kanade method decreases when the thresholds get larger. The DBD-SVM method, achieving 98.2% face detection rate with two false detections, thus compares favorably against the Schneiderman-Kanade method [84].

### 2.4.5 Face Detection in Video Streams

The testing set for face detection in video contains video streams captured by a DFW-VL500 digital camera, which uses IEEE 1394 interface and is capable of capturing full motion images at 30 frames per second. To demonstrate the real-world application, the DBD-SVM face detection method is implemented to search for faces in six predefined scales:  $75 \times 75$ ,  $62 \times 62$ ,  $50 \times 50$ ,  $40 \times 40$ ,  $30 \times 30$ , and  $25 \times 25$ . The faces detected in one scale are propagated to the successive search scales to eliminate the detected regions from being searched again. The scanning procedure is detailed in Subsection 2.4.3.

Figures 2.20 and 2.21 show face detection results in a video stream. The upper half of frames in Figure 2.20 show the original input stream, and the lower half of frames display the corresponding motion analysis results. Yellow rectangles denote motion FOIs that bound motion pixels. Within the motion FOIs, skin color segmentation is applied to eliminate non-skin color pixels and derives final FOIs. Figure 2.21 shows the skin segmentation results in the upper half of frames where skin pixels are displayed in color while non-skin pixels are in black. The final FOIs derived by both the motion analysis and the skin color segmentation are displayed in red rectangles. Finally, the lower half of frames in Figure 2.21 show face detection results. All faces whose sizes are within the search scales are successfully detected.

Figures 2.22 and 2.23 show face detection results in a video stream with complex background. Similarly, the upper half of frames in Figure 2.22 show the original input stream, the lower half of frames in Figure 2.22 display the motion analysis results in yellow rectangles, the upper half of frames in Figure 2.23 show the final FOI derived by both the motion analysis and the skin color segmentation in red rectangles, and the lower half of frames in Figure 2.23 show face detection results. Again, all faces whose sizes are among the predefined search scales are successfully detected. Note that in this video stream, the subjects wear clothes with more local structures (such as stripes) than the subjects appeared in Figure 2.20. As a results, more motion pixels are detected here than in the previous



**Figure 2.20** Face detection in a video stream. The upper half of frames show the original input stream, and the lower half of frames show the motion analysis results.





**Figure 2.21** Face detection in a video stream. The upper half of frames show the FOIs derived from both motion analysis and skin color segmentation, and the lower half of frames show the face detection results.

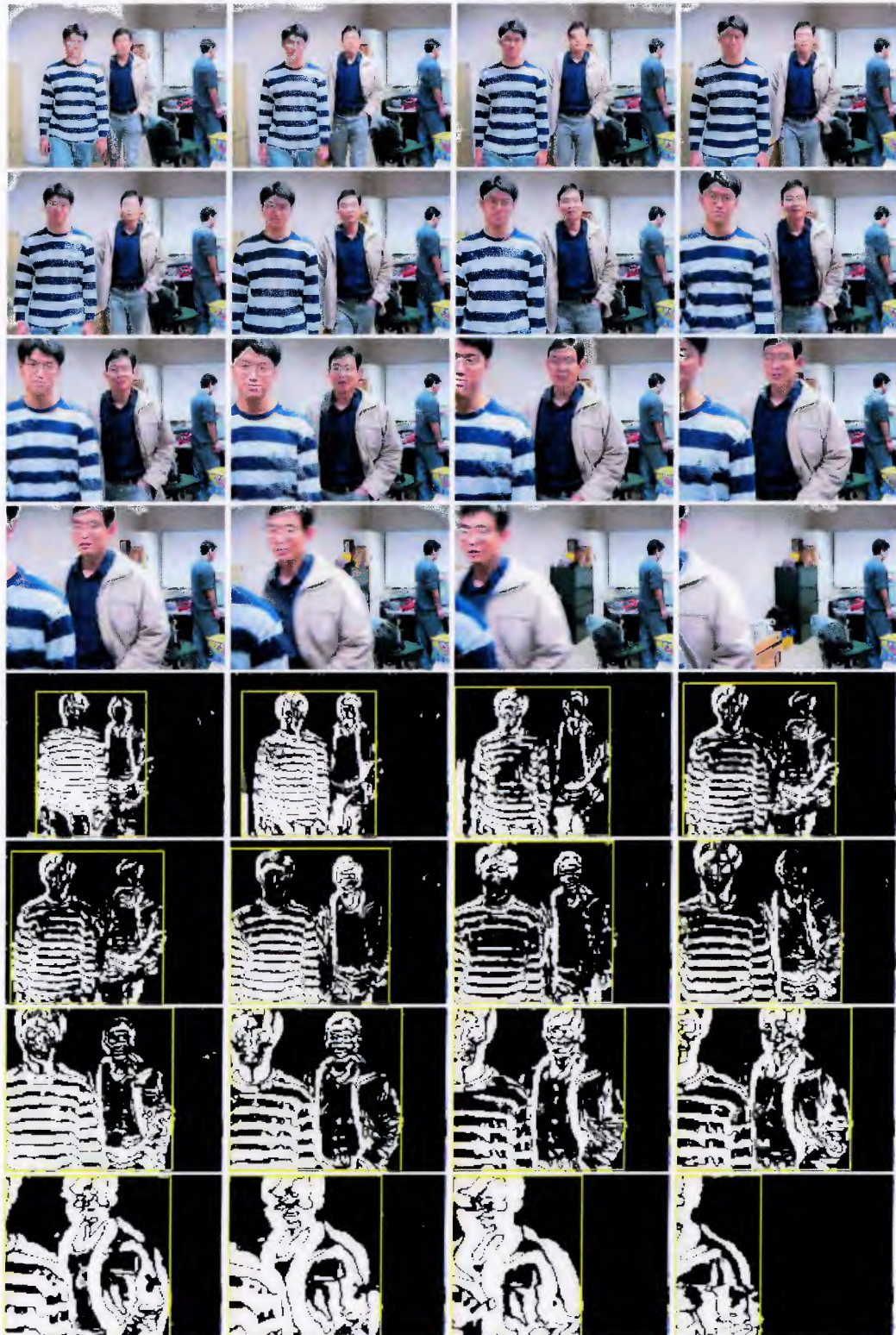
video stream. Also note that the motion analysis eliminates small motion caused by people standing far away from the camera.

Finally, Figures 2.24 and 2.25 display face detection results in a video stream with illumination variation. The subjects far away from the camera are under very dim lighting condition, and subsequently deteriorate the motion analysis and the skin color segmentation. When subjects approach the camera, however, the DBD-SVM face detection method reliably detects all faces.

## 2.5 Conclusion

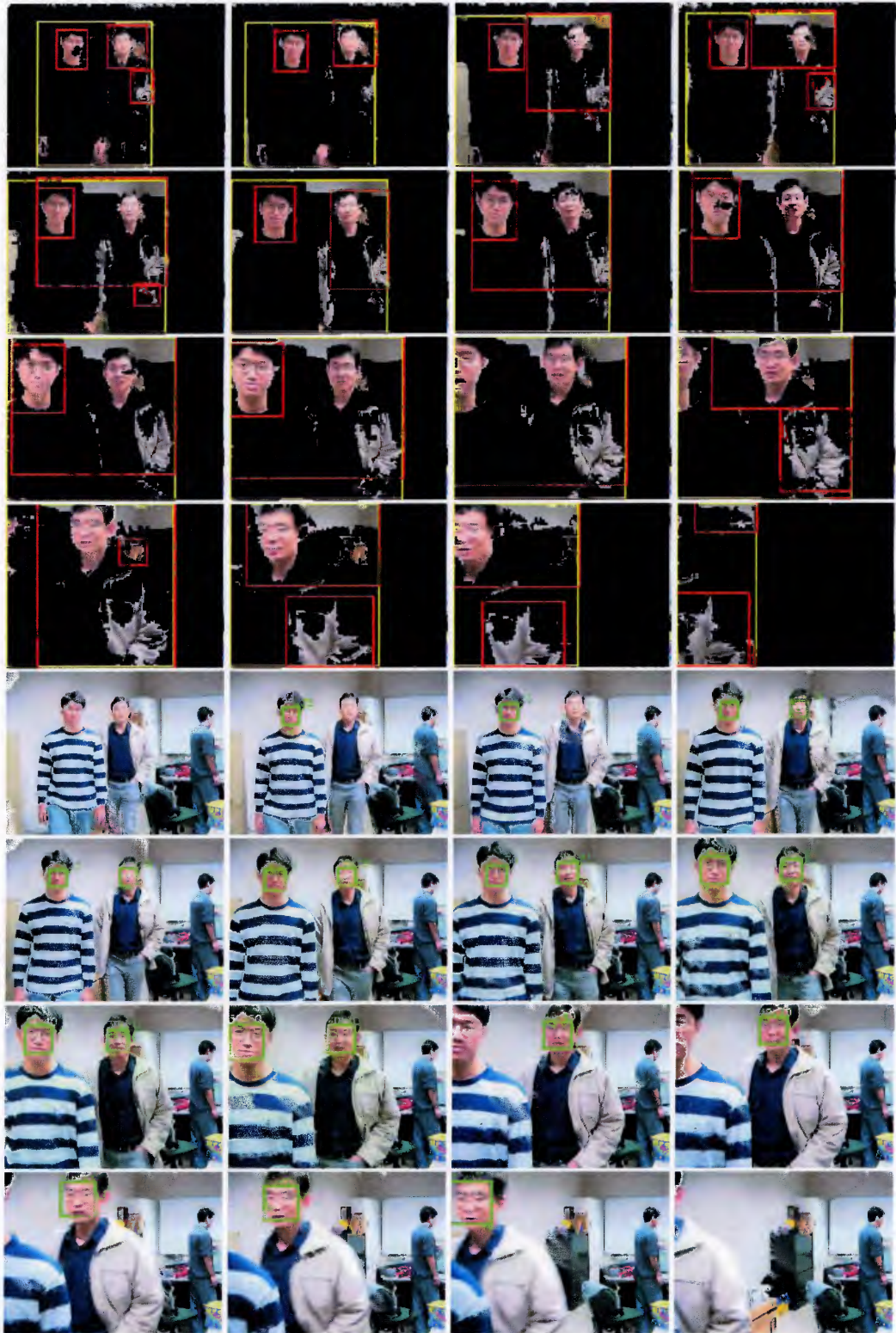
This chapter presents the DBD-SVM face detection method, whose novelty comes from the integration of motion analysis, skin color segmentation, distribution-based distance, and support vector machine. Experimental results using test images from the MIT-CMU test sets and video streams show the feasibility of the DBD-SVM face detection method. In particular, when using 92 images (containing 282 faces) from the MIT-CMU test sets, the DBD-SVM method achieves 98.2% correct face detection rate with two false detections, a performance comparable to the Schneiderman-Kanade's method. When using video streams, the DBD-SVM method detects faces reliably and efficiently.





**Figure 2.22** Face detection in a video stream with complex background. The upper half of frames show the original input stream, and the lower half of frames show the motion analysis results.



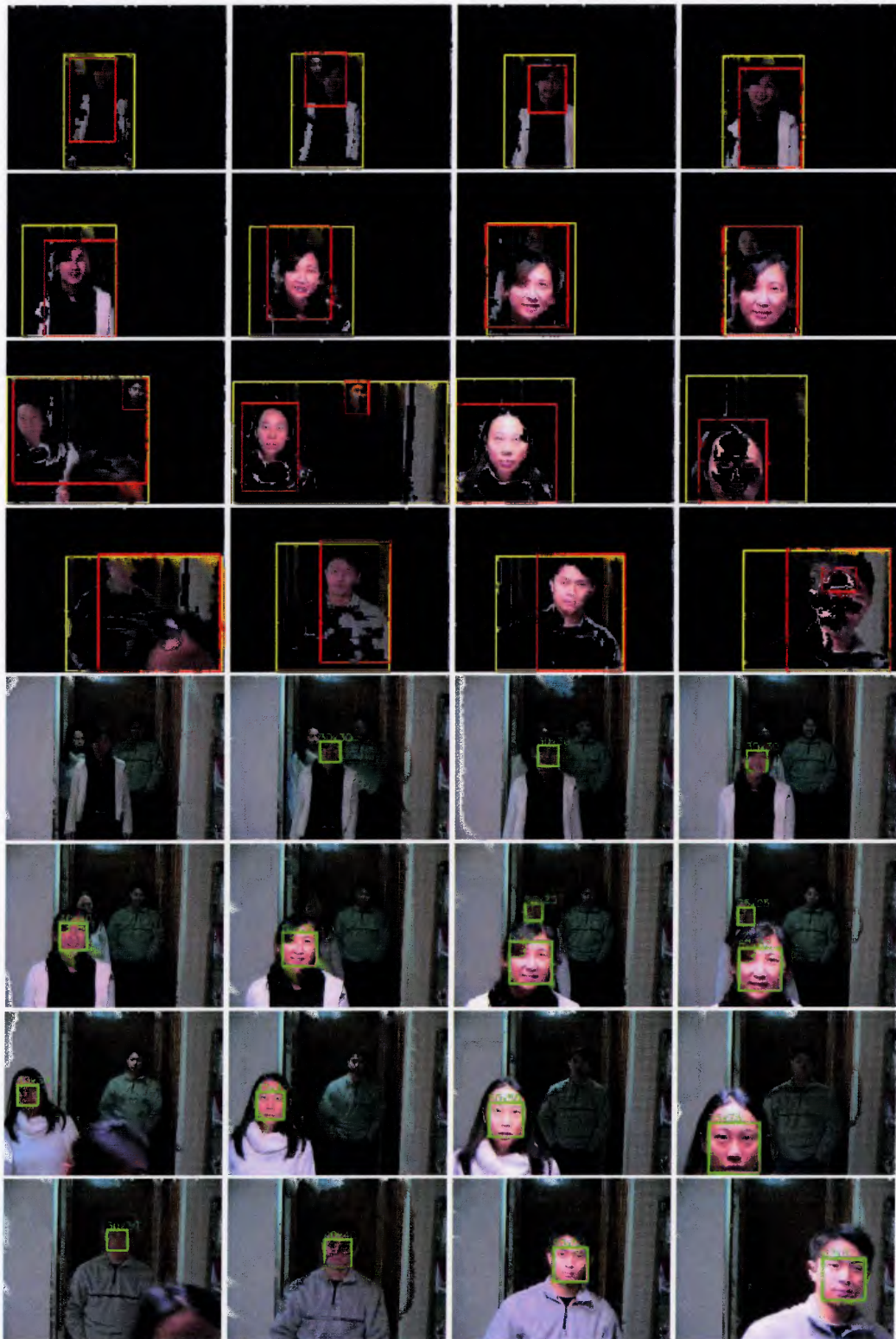


**Figure 2.23** Face detection in a video stream with complex background. The upper half of frames show the FOIs derived from both motion analysis and skin color segmentation, and the lower half of frames show the face detection results.





**Figure 2.24** Face detection in a video stream with illumination variation. The upper half of frames show the original input stream, and the lower half of frames show the motion analysis results.



**Figure 2.25** Face detection in a video stream with illumination variation. The upper half of frames show the FOIs derived from both motion analysis and skin color segmentation, and the lower half of frames show the face detection results.

## CHAPTER 3

### FACE RECOGNITION USING COLOR CONFIGURATIONS

Face recognition identifies individuals from a collection of face images based on the geometric or statistical features automatically derived from these images. An efficient algorithm generally requires a robust feature extraction method that has the ability to derive discriminating features to distinguish one face from another. Among a wide range of statistical features, color is considered an effective descriptor that often simplifies object identification [30]. Color spaces, defined by means of transformations from the original  $RGB$  color space, thus display different color properties and provide useful information for different visual tasks. The  $YC_bC_r$ , the  $YIQ$ , and the  $YUV$  color spaces, for example, have wide applications in color clustering and quantization of skin color regions [26], [111], [32]. The  $HSV$ , the  $HSI$ , and the  $HLS$  color spaces, on the other hand, are often applied to locating and extracting facial features [92], [26]. In the face recognition research field, however, most models are implemented in the luminance domain. To improve existing face recognition algorithms through the utilization of color information, this chapter first presents extensive assessments of face recognition performance in different color spaces, and then defines a color feature extraction method to help improve the baseline performance of the Face Recognition Grand Challenge (FRGC) problems.

The comparative assessments of face recognition performance are implemented in twelve color spaces ( $RGB$ ,  $HSV$ ,  $YUV$ ,  $YC_bC_r$ ,  $XYZ$ ,  $YIQ$ ,  $L^*a^*b^*$ ,  $U^*V^*W^*$ ,  $L^*u^*v^*$ ,  $I_1I_2I_3$ ,  $HSI$ , and  $rgb$ ) by evaluating seven color configurations for every color space. A color configuration is defined by an individual or a combination of color component images. Take the  $RGB$  color space as an example, possible color configurations are  $R$ ,  $G$ ,  $B$ ,  $RG$ ,  $RB$ ,  $GB$ , and  $RGB$ . Experimental results using 600 FERET color images corresponding to 200 subjects and 456 FRGC color images of 152 subjects show that some



color configurations, such as  $YV$  in the  $YUV$  color space and  $YI$  in the  $YIQ$  color space, help improve face recognition performance.

The color feature extraction method, which uses the color configurations defined by the color component images across different color spaces, is implemented to improve the FRGC baseline performance of the Biometric Experimentation Environment (BEE) algorithm. The FRGC databases are designed to address “face recognition problems that are harder, as defined by the image sets in the experiments and the performance by a control algorithm [72].” The control algorithm of the FRGC databases is known as the BEE baseline algorithm, which is a principal component analysis (PCA) algorithm optimized for large scale problems [66]. The distance measure used in the nearest neighbor classifier of the BEE baseline algorithm is whiten cosine [66]. Experimental results using an FRGC ver1.0 dataset containing 366 training images, 152 controlled gallery images, and 608 uncontrolled probe images, show that the  $YQC_r$  color configuration improves the rank-one face recognition rate of the BEE baseline algorithm from 37% to 70%; when using an FRGC ver2.0 dataset including 6,660 training images, 16,028 controlled target images, and 8,014 uncontrolled query images, the  $YQC_r$  color configuration improves the face verification rate (at 0.1% false acceptance rate) of the BEE baseline algorithm from 13% to 33%.

The remainder of this chapter is organized as follows. Section 3.1 reviews various computer vision applications implementing color information. Section 3.2 gives the definition of twelve color spaces in which face recognition performance is assessed. Section 3.3 details the comparative assessments of face recognition performance in twelve color spaces. Section 3.4 presents the color feature extraction method for improving the FRGC baseline performance. Finally, Section 3.5 summarizes the research in face recognition.

### 3.1 Background

Face recognition is a very active research area as evidenced by the large number of publications in the journals and conferences of computer vision and pattern recognition [6], [42], [54], [52], [72], [19]. Appearance-based approaches, such as eigenfaces [47], [102], also known as principal component analysis (PCA), and Fisherfaces [22], [3], [118], also known as Fisher Linear Discriminant (FLD) or Linear Discriminant Analysis (LDA), have been proved to be effective in the face recognition tasks. Based on these algorithms, recent research has focused on extracting discriminating features to enhance the classification ability of PCA and FLD. These applications, such as [56], [57], have been shown to be less sensitive to variations in illumination and viewpoint. A comprehensive survey of face recognition can be found in [117].

Color provides useful information for object detection, indexing and retrieval [112], [40]. Color histograms and color invariant moments provide robust object recognition against image variations such as illumination [33], [23]. Swain and Ballard [89] developed a color indexing system which applies color histogram for image inquiry from a large image database [95]. The system separates the chrominance from the luminance and the color information derived is invariant to illumination variations.

In general, different color spaces, which are defined by means of transformations from the original *RGB* (red, green, blue) color space, display different color properties. The *HSV* (hue, saturation, value) color space and its variants, such as the *HSI* (hue, saturation, intensity) color space and the *HLS* (hue, lightness, saturation) color space, are often applied in locating and extracting facial features [92], [26]. The *YCbCr* (luminance, chrominance-blue, chrominance-red) color space, the *YIQ* (luminance, in-phase, quadrature) color space, and the *YUV* color space have wide applications in color clustering and quantization for skin color regions [26], [111], [78], [32]. The perceptually uniform color spaces, such as the *CIE- $U^*V^*W^*$*  color space, the *CIE- $L^*u^*v^*$*  color space, and the *CIE- $L^*a^*b^*$*  color space have general and ubiquitous applications [109], [118], [59], [99].

The color information provided by the different color spaces has been applied to different visual tasks. Skin color, for example, is used for face detection [40], [37], [99]. Yang et al. [112] suggested that the luminance account for most of the skin color variation and showed that the color histogram based on normalized red and green color components occupies a small cluster in the histogram. Sobottka et al. [92], on the other hand, showed that The *HSV* color space and its variations have the advantages in providing large variance among facial features and are suitable for locating and extracting facial features. Hsu et al. [40] applied a nonlinear color transformation to detect skin patches in order to find human faces in an image. Torres et al. [100], however, showed that color information does not improve the face recognition performance in comparison with the intensity information.

### 3.2 Color Spaces

This section details twelve color spaces assessed: the *RGB* color space, the *rgb* color space, the  $I_1I_2I_3$  (decorrelated RGB) color space, the CIE-*XYZ* color space, the human perceptual color spaces: *HSV* and *HSI*, the video transmission efficiency color spaces: *YIQ*, *YUV* and  $YC_bC_r$ , and the CIE perceptually uniform color spaces: CIE- $U^*V^*W^*$ , CIE- $L^*u^*v^*$ , and CIE- $L^*a^*b^*$ .

The *RGB* images are sensitive to luminance, surface orientation, and other photographic conditions. To minimize such sensitivity, one can project the  $R, G, B$  values onto the  $R = G = B = \max\{R, G, B\}$  plane. The projection spans a normalized *rgb* chromaticity triangle. The transformation is defined as follows [95]:

$$\begin{aligned} r &= \frac{R}{R+G+B} \\ g &= \frac{G}{R+G+B} \\ b &= \frac{B}{R+G+B} \end{aligned} \tag{3.1}$$

Another approach to stabilize RGB image is to decorrelate the  $RGB$  components. The  $I_1I_2I_3$  color space proposed by Ohta et al. [69] applies a Karhunen-Loeve transformation to decorrelate the  $RGB$  components. The linear transformation based on Ohta's experimental model is defined as follows [69]:

$$\begin{aligned} I_1 &= \frac{R+G+B}{3} \\ I_2 &= \frac{R-B}{2} \\ I_3 &= \frac{2G-R-B}{2} \end{aligned} \quad (3.2)$$

The definition of the  $HSV$  and the  $HSI$  color spaces are based by human vision system in a sense that human describes color by means of hue, saturation, and brightness. Hue and saturation define chrominance, while intensity or value specifies luminance [30]. The  $HSV$  color space is defined as follows [90]:

$$\begin{aligned} \text{Let } \begin{cases} MAX = \max(R, G, B) \\ MIN = \min(R, G, B) \\ \delta = MAX - MIN \end{cases} \\ V &= MAX \\ S &= \begin{cases} \frac{\delta}{MAX} & \text{if } MAX \neq 0 \\ 0 & \text{if } MAX = 0 \end{cases} \\ H &= \begin{cases} 60(\frac{G-B}{\delta}) & \text{if } MAX = R \\ 60(\frac{B-R}{\delta} + 2) & \text{if } MAX = G \\ 60(\frac{R-G}{\delta} + 4) & \text{if } MAX = B \\ \text{not defined} & \text{if } MAX = 0 \end{cases} \end{aligned} \quad (3.3)$$

In order to confine  $H$  within the range of  $[0,360]$

$$H = H + 360 \quad \text{if } H < 0$$

The *HSI* color space is specified as follows [30]:

$$\begin{aligned}
 I &= (R + G + B)/3 \\
 S &= 1 - I[\min(R, G, B)] \\
 H &= \begin{cases} \theta & \text{if } B \leq G \\ 360 - \theta & \text{otherwise} \end{cases}
 \end{aligned} \tag{3.4}$$

where

$$\theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)]^{\frac{1}{2}}} \right\}$$

Note that in both Equation 3.3 and Equation 3.4, the  $R, G, B$  values are scaled to  $[0,1]$ .

The *YUV* and the *YIQ* color spaces are commonly used in video for transmission efficiency. The *YIQ* color space is adopted by the NTSC (National Television System Committee) video standard in reference to *RGB NTSC*, while the *YUV* color space is used by PAL (Phase Alternation by Line) and SECAM (System Electronique Couleur Avec Memoire). The *YUV* color space is specified as follows [9]:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.2990 & 0.5870 & 0.1140 \\ -0.1471 & -0.2888 & 0.4359 \\ 0.6148 & -0.5148 & -0.1000 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.5}$$

The  $I$  and  $Q$  components are derived from their counterparts,  $U$  and  $V$ , via a clockwise rotation ( $33^\circ$ ), and the *YIQ* color space is defined as follows [9]:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.2990 & 0.5870 & 0.1140 \\ 0.5957 & -0.2745 & -0.3213 \\ 0.2115 & -0.5226 & 0.3111 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{3.6}$$



The  $YC_bC_r$  color space is developed as a part of the ITU-R Recommendation B.T. 601 [9] for digital video standard and television transmissions. It is a scaled and offset version of the  $YUV$  color space. In  $YC_bC_r$ , the  $RGB$  components are separated into luminance ( $Y$ ), chrominance blue ( $C_b$ ) and chrominance red ( $C_r$ ). The  $Y$  component has 220 levels ranging from 16 to 235, while the  $C_b, C_r$  components have 225 levels ranging from 16 to 240 [9]:

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.4810 & 128.5530 & 24.9660 \\ -37.7745 & -74.1592 & 111.9337 \\ 111.9581 & -93.7509 & -18.2072 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.7)$$

where the  $R, G, B$  values are scaled to  $[0,1]$ .

The Commission Internationale de l'Éclairage (CIE) defined perceptually uniform color spaces, such as  $U^*V^*W^*$ ,  $L^*u^*v^*$ , and  $L^*a^*b^*$ , based on the  $XYZ$  tristimulus [46]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.116 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.8)$$

Note that the  $Y$  component defined here is consistent with the luminance defined in Equations 3.5, 3.6, 3.7. In addition, a chromaticity diagram can be derived via the chromaticity coordinates  $x, y$ , which are specified by the  $X, Y, Z$  tristimulus. This CIE chromaticity diagram, however, is not perceptually uniform [46], i.e., areas of the least perceptible differences on the diagram are distorted to ellipses (known as MacAdam ellipse) rather than circles. To overcome such a shortcoming, the CIE- $uv$  chromaticity diagram was proposed [46]:

$$\begin{aligned} u &= \frac{4x}{-2x+12y+3} \quad \text{or} \quad \frac{4X}{X+15Y+3Z} \\ v &= \frac{6y}{-2x+12y+3} \quad \text{or} \quad \frac{6Y}{X+15Y+3Z} \end{aligned} \quad (3.9)$$

Based on this uniform chromaticity scale (UCS), a CIE uniform color space  $U^*V^*W^*$  was proposed. The  $W^*$  component corresponds to luminance, while the  $U^*$ ,  $V^*$  components correspond to chrominance [46]:

$$W^* = \begin{cases} 116\left(\frac{Y}{Y_o}\right)^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Y_o} > 0.008856 \\ 903.3\left(\frac{Y}{Y_o}\right) & \text{otherwise} \end{cases} \quad (3.10)$$

$$U^* = 13W^*(u - u_o)$$

$$V^* = 13W^*(v - v_o)$$

where the  $u_o$  and  $v_o$  are derived from the reference white stimulus.

Although the CIE- $uv$  diagram is perceptually uniform, it has its own deficiency in representing yellow-red colors as the area of yellow-red in the diagram is relatively small [11]. To improve this deficiency, a new  $uv$  diagram was proposed [11]:

$$\begin{aligned} u' &= u \\ v' &= \frac{3}{2}v \end{aligned} \quad (3.11)$$

Based on the  $u'v'$  coordinate system, two CIE uniform color spaces were defined, namely the CIE- $L^*u^*v^*$  color space and the CIE- $L^*a^*b^*$  color space [11]. The CIE- $L^*u^*v^*$  color space is proposed to obsolete the  $U^*V^*W^*$  color space [11]:

$$L^* = \begin{cases} 116\left(\frac{Y}{Y_o}\right)^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Y_o} > 0.008856 \\ 903.3\left(\frac{Y}{Y_o}\right) & \text{otherwise} \end{cases} \quad (3.12)$$

$$u^* = 13L^*(u' - u'_o)$$

$$v^* = 13L^*(v' - v'_o)$$

where the  $u'_o$  and  $v'_o$  components are derived from the reference white stimulus.

The  $L^*a^*b^*$  color space is one of the most commonly used color spaces and is modeled based on human vision system. The  $L^*$  component in the  $L^*a^*b^*$  color space corresponds to brightness ranging from 0 (black) to 100 (white), the  $a^*$  component corresponds to the measurement of redness (positive values) or greenness (negative values), and the  $b^*$  component corresponds to the measurement of yellowness (positive values) or blueness (negative values). The  $L^*a^*b^*$  color space is defined based on the  $XYZ$  tristimulus [11]:

$$\begin{aligned} L^* &= 116f\left(\frac{Y}{Y_o}\right) - 16 \\ a^* &= 500\left[f\left(\frac{X}{X_o}\right) - f\left(\frac{Y}{Y_o}\right)\right] \\ b^* &= 200\left[f\left(\frac{Y}{Y_o}\right) - f\left(\frac{Z}{Z_o}\right)\right] \end{aligned} \quad (3.13)$$

where

$$f(x) = \begin{cases} x^{\frac{1}{3}} & \text{if } x > 0.008856 \\ 7.787x + \frac{16}{116} & \text{otherwise} \end{cases}$$

### 3.3 Comparative Assessments of Face Recognition Performance

Color spaces are defined for special advantages related to devices that produce color (e.g.,  $RGB$  for color monitors and color video cameras,  $CMY$  for color printing) or related to human perception (e.g.,  $HSV$  and  $HSI$  for human describing and interpreting color,  $YUV/YC_bC_r$  for image/video compression in JPEG and MPEG). As a result, the appropriateness of color spaces for pattern classification is usually assessed by experimental studies. For example, Geusebroek et al. [27] experimentally investigated the invariance and discriminative power of the color invariants; Terrillon et al. [98] performed comparative analysis of nine different color spaces for face detection; and based on this analysis Hsu et al. [40] chose the  $YC_bC_r$  color space to develop their face detection algorithm.

This section first details the experimental procedure for empirical assessments of face recognition performance in twelve different color spaces and then presents experimental re-

sults derived using 600 FERET color images corresponding to 200 subjects and 456 FRGC color images of 152 subjects. Specifically, the recognition performance of twelve color spaces ( $RGB$ ,  $HSV$ ,  $YUV$ ,  $YC_bC_r$ ,  $XYZ$ ,  $YIQ$ ,  $L^*a^*b^*$ ,  $U^*V^*W^*$ ,  $L^*u^*v^*$ ,  $I_1I_2I_3$ ,  $HSI$ , and  $rgb$ ) is assessed by evaluating seven color configurations for every single color space. A color configuration is defined by an individual or a combination of color component images. Take the  $RGB$  color space as an example, possible color configurations are  $R$ ,  $G$ ,  $B$ ,  $RG$ ,  $RB$ ,  $GB$ , and  $RGB$ . The experimental results show that some color configurations, such as  $YV$  in the  $YUV$  color space and  $YI$  in the  $YIQ$  color space, help improve face recognition performance.

### 3.3.1 Methodology

Principal component analysis (PCA) has been widely used to perform dimensionality reduction for face recognition and indexing. In particular, PCA is the method behind the Eigenfaces coding scheme [102] whose primary goal is to project the similarity judgment for face recognition into a low-dimensional space. This space defines a feature space, or a “face space,” which drastically reduces the dimensionality of the original space, and face detection and identification are performed in this reduced face space.

Let  $\mathbf{x} \in \mathbb{R}^N$  be a random vector representing a color configuration, where  $N$  is the dimensionality of the corresponding input space. The vector is formed by concatenating the rows or the columns of the color component images which has been normalized to have zero mean and unit variance. The covariance matrix of  $\mathbf{x}$  is defined as follows [25]:

$$\Sigma = \mathcal{E}\{[\mathbf{x} - \mathcal{E}(\mathbf{x})][\mathbf{x} - \mathcal{E}(\mathbf{x})]^t\} \quad (3.14)$$

where  $\mathcal{E}[\cdot]$  is the expectation operator,  $t$  denotes the transpose operator, and  $\Sigma \in \mathbb{R}^{N \times N}$  is the covariance matrix of  $\mathbf{x}$ . The PCA procedure factorizes the covariance matrix  $\Sigma$  into the following form [25]:

$$\Sigma = \Phi \Lambda \Phi^t \quad (3.15)$$

where  $\Phi = [\varphi_1 \varphi_2 \dots \varphi_N] \in \mathbb{R}^{N \times N}$  is an orthogonal eigenvector matrix and  $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_N\} \in \mathbb{R}^{N \times N}$  a diagonal eigenvalue matrix with diagonal elements in decreasing order ( $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_N$ ) [25].

An important property of PCA is decorrelation, i.e., the components of the transformed data,  $\mathbf{y} = \Phi^t \mathbf{x}$ , are decorrelated since the covariance matrix of  $\mathbf{y}$  is diagonal,  $\Sigma_{\mathbf{y}} = \Lambda$ , and the diagonal elements are the variances of the corresponding components. Another important property of PCA is its optimal signal reconstruction in the sense of minimum mean square error (MSE) when only a subset of principal components is used to represent the original signal. Following this property, an immediate application of PCA is the dimensionality reduction [54], [86]:

$$\mathbf{z} = \mathcal{P}^t \mathbf{x} \quad (3.16)$$

where  $\mathcal{P} = [\varphi_1 \varphi_2 \dots \varphi_m]$ ,  $m < N$ , and  $\mathcal{P} \in \mathbb{R}^{N \times m}$ . The lower dimensional vector  $\mathbf{z} \in \mathbb{R}^m$  captures the most expressive features of the original data  $\mathbf{x}$ .

After dimensionality reduction by Equation 3.16, feature vectors are compared and classified by the nearest neighbor (to the mean) rule using a similarity (distance) measure  $\delta$  [54], [86]:

$$\delta(\mathbf{z}, \mathbf{m}_k) = \min_j \delta(\mathbf{z}, \mathbf{m}_j) \longrightarrow \mathbf{z} \in \omega_k \quad (3.17)$$

where  $\mathbf{z}$  is a testing feature vector and  $\mathbf{m}_k$ ,  $k = 1, 2, \dots, L$  is the mean of the training samples for the class  $\omega_k$ . The testing feature vector,  $\mathbf{z}$ , is classified as belonging to the class of the closest mean,  $\mathbf{m}_k$ , using the similarity measure  $\delta$ .

The similarity measures used in the experiments to evaluate the efficiency of different representation and recognition methods include the  $L_1$  distance measure,  $\delta_{L_1}$ , the  $L_2$  distance measure,  $\delta_{L_2}$ , the Mahalanobis distance measure,  $\delta_{Mah}$ , and the cosine similarity measure,  $\delta_{cos}$ , which are defined as follows [54], [86]:

$$\delta_{L_1}(\mathbf{a}, \mathbf{b}) = \sum_i |a_i - b_i| \quad (3.18)$$

$$\delta_{L_2}(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^t(\mathbf{a} - \mathbf{b}) \quad (3.19)$$

$$\delta_{Mah}(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^t \Sigma^{-1}(\mathbf{a} - \mathbf{b}) \quad (3.20)$$

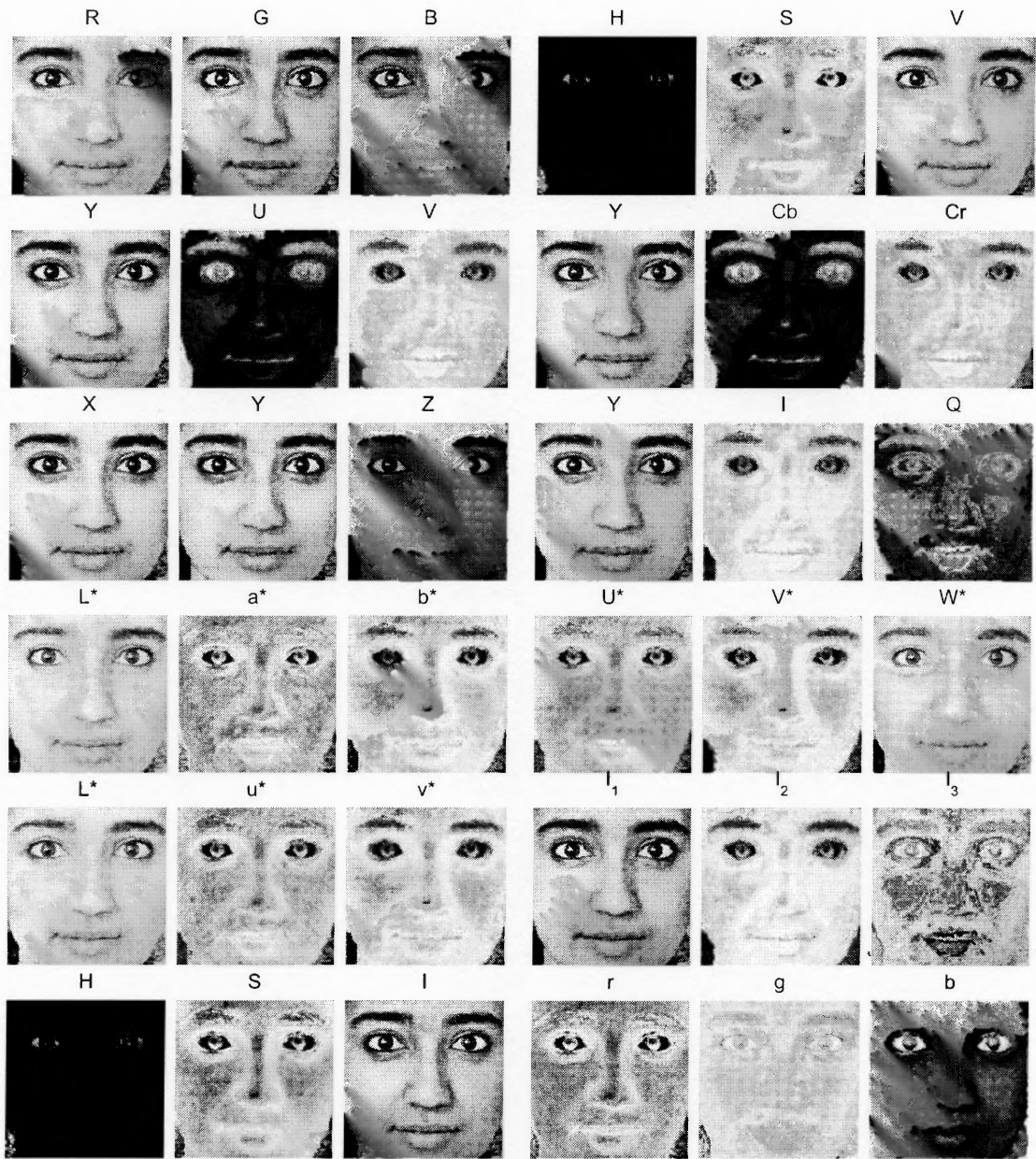
$$\delta_{cos}(\mathbf{a}, \mathbf{b}) = \frac{-\mathbf{a}^t \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\|} \quad (3.21)$$

where  $\Sigma$  is the covariance matrix, and  $\|\cdot\|$  denotes the norm operator. Note that the cosine similarity measure includes a minus sign in Equation 3.21, because the nearest neighbor (to the mean) rule of Equation 3.17 applies minimum (distance) measure rather than maximum similarity measure [65], [54], [86].

### 3.3.2 Experiments

The experiments of the assessments of the face recognition in twelve color spaces is carried out using 600 FERET color images corresponding to 200 subjects and 456 FRGC (Face Recognition Grand Challenge) color images of 152 subjects. The 600 FERET images are from the FERET database [73], which has become the de facto standard for evaluating face recognition technologies. The images correspond to 200 subjects such that each subject has three images. The 456 FRGC color images are from the FRGC database [6], and they correspond to 152 subjects such that there are three images for each subject. Among the three images, one is a controlled image with good image quality, and the remaining two are uncontrolled images with challenging image quality in terms of illumination, resolution, blurring, etc. [6].

The face images used in the experiments are normalized to extract facial regions that contain only faces, so that the face recognition performance is not affected by the factors not related to face, such as hair style. Specifically, the normalization consists of



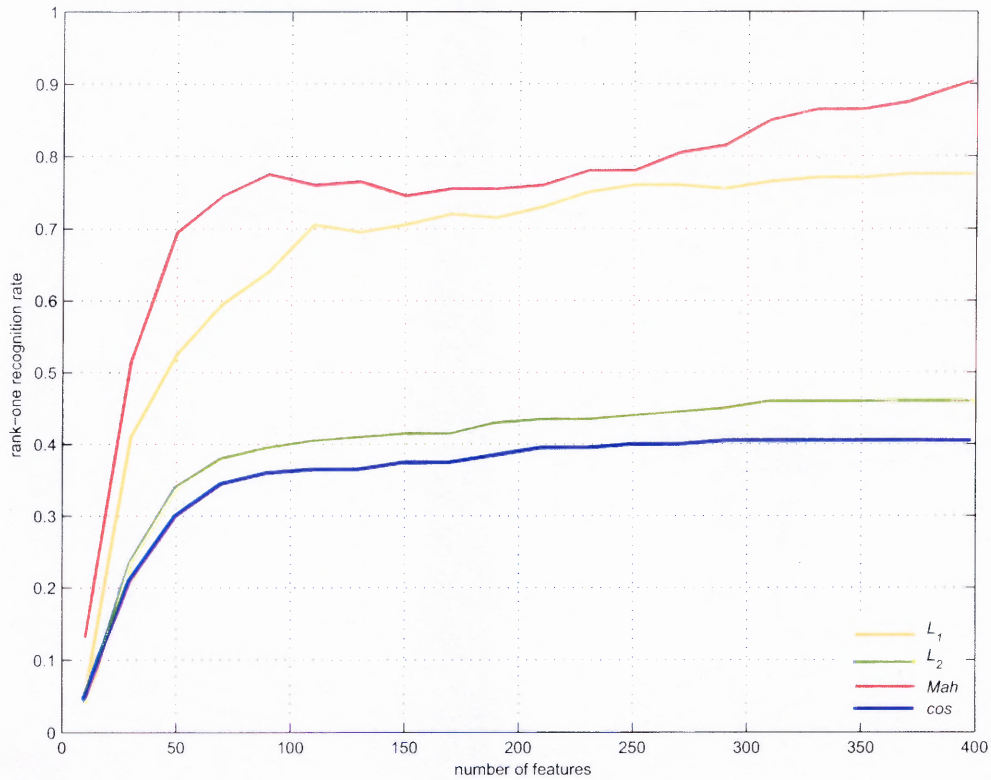
**Figure 3.1** Color component images of twelve color spaces as defined in Section 3.2. From left to right, top to bottom, the color spaces are the  $RGB$  color space, the  $HSV$  color space, the  $YUV$  color space, the  $YC_bC_r$  color space, the  $XYZ$  color space, the  $YIQ$  color space, the  $L^*a^*b^*$  color space, the  $U^*V^*W^*$  color space, the  $L^*u^*v^*$  color space, the  $I_1I_2I_3$  color space, the  $HSI$  color space, and the  $rgb$  color space.

the following procedures: first, manual annotation detects the centers of the eyes; second, rotation and scaling transformations align the centers of the eyes to the predefined locations and fix the interocular distance; finally, a subimage procedure crops the face image to the size of  $128 \times 128$  to extract the facial region. Figure 3.1 shows some example images used in the experiments that are already cropped to the size of  $128 \times 128$ . These images are the individual color component images of twelve color spaces as defined in Section 3.2. From left to right, top to bottom, the color spaces are the *RGB* color space, the *HSV* color space, the *YUV* color space, the *YCbCr* color space, the *XYZ* color space, the *YIQ* color space, the *L\*a\*b\** color space, the *U\*V\*W\** color space, the *L\*u\*v\** color space, the *I<sub>1</sub>I<sub>2</sub>I<sub>3</sub>* color space, the *HSI* color space, and the *rgb* color space. Note that both the FERET and the FRGC images were acquired during different photo sessions, they display different illumination characteristics and facial expressions. For the FERET images, as there are three images for each subject, two of them are randomly chosen for training, while the remaining one (unseen during training) is used for testing; similarly, for the FRGC images, two images of each subject are randomly chosen for training, while the remaining one is used for testing.

The face recognition performance of different color spaces is assessed by evaluating seven color configurations for every single color space. A color configuration is defined by an individual or a combination of color component images. Take the *RGB* color space as an example, possible color configurations are *R*, *G*, *B*, *RG*, *RB*, *GB*, and *RGB*. Note that when two or three color component images are used to define a color configuration, each color component image is first normalized to zero mean and unit variance, and then the normalized color component images are concatenated to form an augmented vector representing the color configuration.

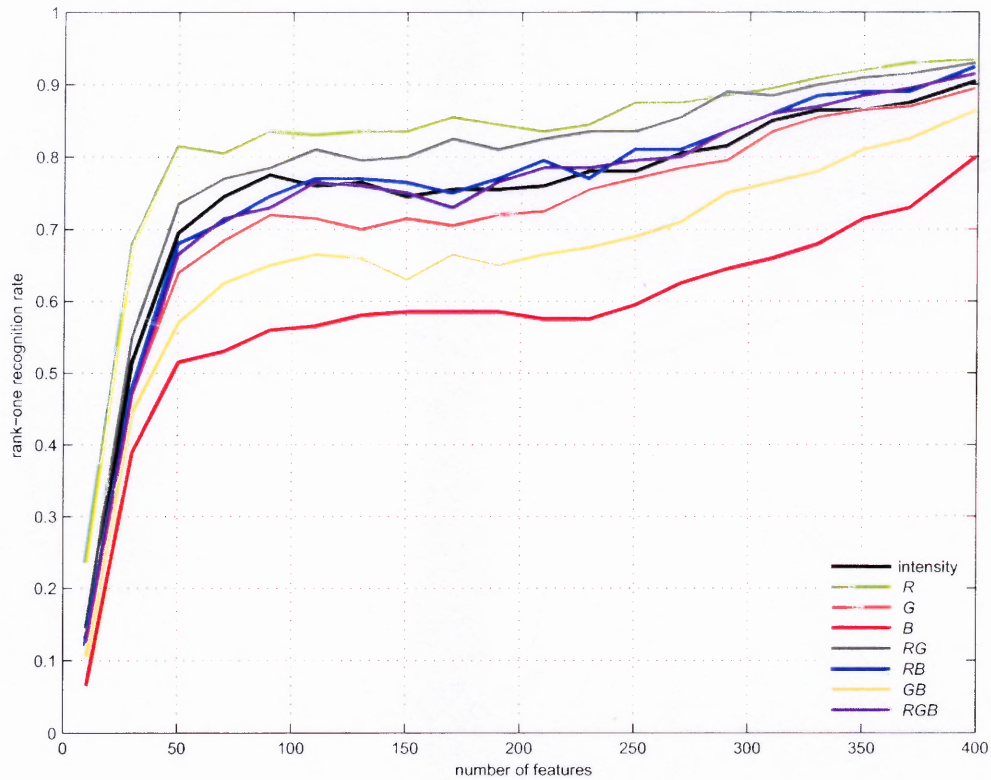
To provide a baseline performance for comparison, the first set of experiments applies different similarity measures as defined in Subsection 3.3.1 on the intensity images derived by averaging the *R*, *G*, *B* color component images. Figure 3.2 shows the face recognition





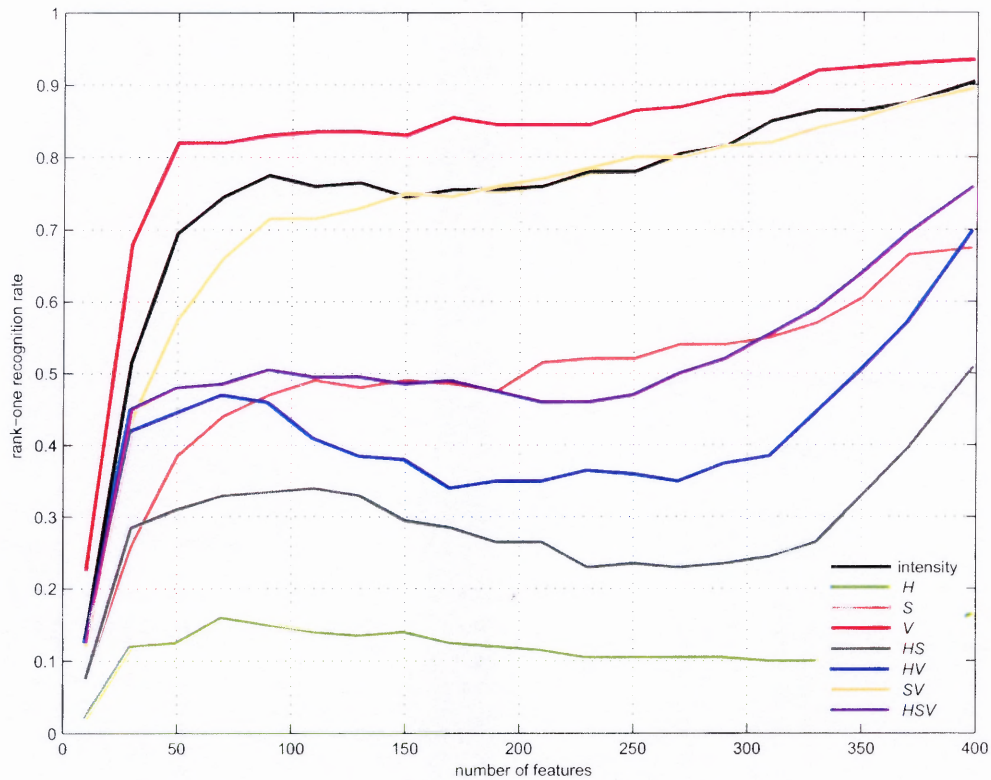
**Figure 3.2** The face recognition performance using the PCA method on the intensity images derived by averaging the  $R$ ,  $G$ ,  $B$  color components. The similarity measures applied are the  $L_1$  distance measure ( $L_1$ ), the  $L_2$  distance measure ( $L_2$ ), the Mahalanobis distance measure ( $Mah$ ), and the cosine similarity measure ( $cos$ ).

performance using the PCA method. The horizontal axis indicates the number of features used, and the vertical axis represents the rank-one face recognition rate, which is the accuracy rate for the top response being correct. Figure 3.2 shows that the Mahalanobis distance measure performs the best, followed in order by the  $L_1$  distance measure, the  $L_2$  distance measure, and the cosine similarity measure. The experimental results provide a baseline face recognition performance based on the intensity images, and suggest that one should use the Mahalanobis distance measure for the comparative assessments in different color spaces. Note that the Mahalanobis distance measure also performs the best in all twelve color spaces detailed in Section 3.2.



**Figure 3.3** The face recognition performance of seven color configurations in the  $RGB$  color space. Note that the performance curve of the intensity images (intensity) is also included for comparison (same in the following figures).

The next set of experiments assesses the following color spaces:  $RGB$ ,  $HSV$ ,  $YUV$ ,  $YC_bC_r$ ,  $XYZ$ ,  $YIQ$ ,  $L^*a^*b^*$ ,  $U^*V^*W^*$ , and  $L^*u^*v^*$ . The face recognition performance of seven color configurations in these color spaces is shown in Figure 3.3, Figure 3.4, Figure 3.5, Figure 3.6, Figure 3.7, Figure 3.8, and Figure 3.9, respectively. Note that the  $YUV$  and the  $YC_bC_r$  color spaces (as well as the  $U^*V^*W^*$  and the  $L^*u^*v^*$  color spaces) have identical face recognition performance due to their definitions (see Section 3.2). The color configurations, which perform better than the intensity images for face recognition are summarized in Table 3.1. In particular, Figure 3.3 shows that the  $R$  and the  $RG$  color configurations perform better than the intensity images. Figure 3.4 shows that the  $V$  color configuration outperforms the intensity images. Figure 3.5 shows that the  $Y$  and  $YV$  color

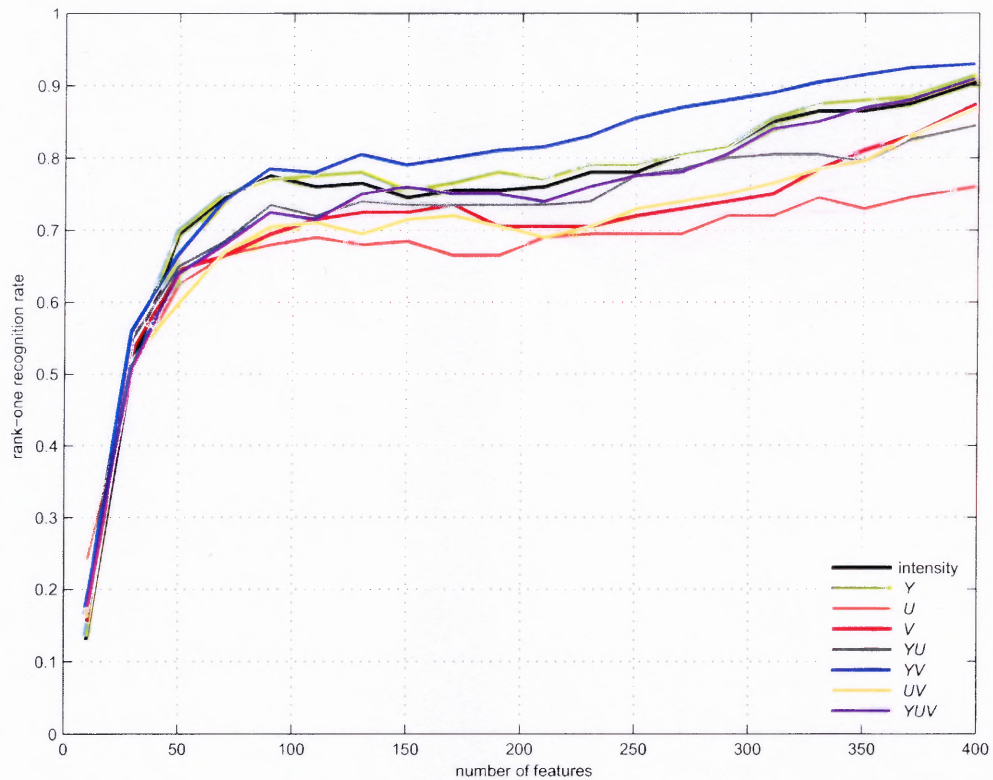


**Figure 3.4** The face recognition performance of seven color configurations in the  $HSV$  color space.

configurations in the  $YUV$  color space or the  $Y$  and  $YC_r$  color configurations in the  $YC_bC_r$  color space have better face recognition performance than the intensity images. Figure 3.6 shows the  $X$ ,  $Y$ , and  $XY$  color configurations perform better than the intensity images. Figure 3.7 shows that the  $Y$  and  $YI$  color configurations outperform the intensity images. Figure 3.8 shows that the  $L^*$ ,  $L^*a^*$ ,  $L^*b^*$ , and  $L^*a^*b^*$  color configurations are better than the intensity images for face recognition. Figure 3.9 shows that the  $W^*$  and  $U^*W^*$  color configurations in the  $U^*V^*W^*$  color space or the  $L^*$  and  $L^*u^*$  color configurations in the  $L^*u^*v^*$  color space perform better than the intensity images.

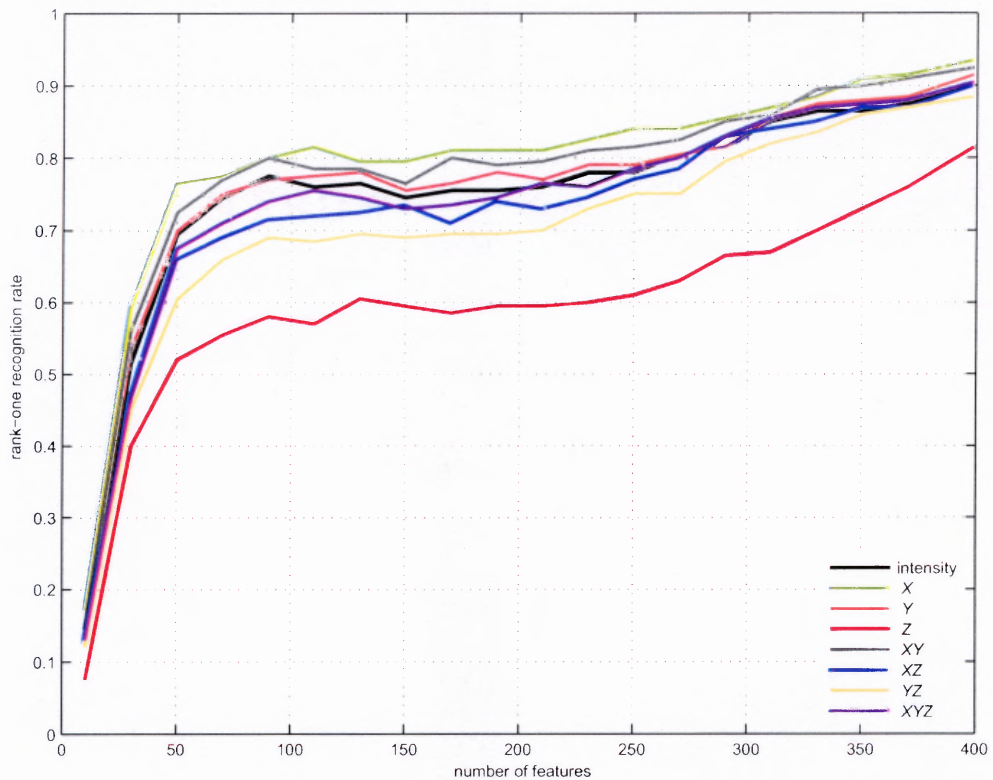
Figure 3.13 compares the best color configurations in the  $RGB$ ,  $HSV$ ,  $YUV$ ,  $YC_bC_r$ ,  $XYZ$ ,  $YIQ$ ,  $L^*a^*b^*$ ,  $U^*V^*W^*$ , and  $L^*u^*v^*$  color spaces. Note that those color configurations with better face recognition performance shown in Figure 3.13 all share one common





**Figure 3.5** The face recognition performance of seven color configurations in the  $YUV$  color space. Note that the face recognition results in the  $YC_bC_r$  color space are the same when the  $Y$ ,  $U$ , and  $V$  color components are replaced by their counterparts  $Y$ ,  $C_b$ , and  $C_r$ , respectively.

characteristic: they contain both chromatic and achromatic (intensity) components. The pure chromatic color configurations, however, all display worse (than the intensity images) face recognition performance. Specifically, these pure chromatic color configurations include the  $HS$  in Figure 3.4, the  $UV$  and the  $C_bC_r$  in Figure 3.5, the  $IQ$  in Figure 3.7, the  $a^*b^*$  in Figure 3.8, and the  $U^*V^*$  and the  $u^*v^*$  in Figure 3.9. Note also that simply applying all the color components does not necessarily achieve the best face recognition performance. The experimental results derived using the  $I_1I_2I_3$ , the  $HSI$ , and the  $rgb$  color spaces, as shown in Figure 3.10, Figure 3.11, and Figure 3.12, respectively, suggest

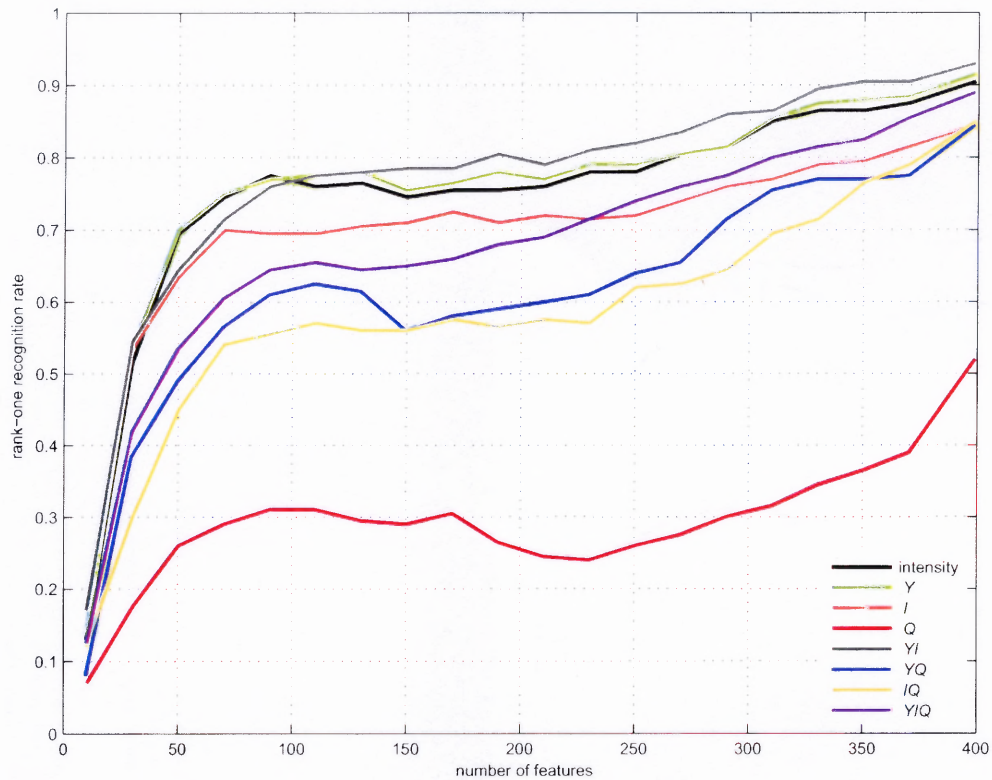


**Figure 3.6** The face recognition performance of seven color configurations in the  $XYZ$  color space.

that the color configurations in these color spaces do not improve face recognition performance.

The robustness of the seven color configurations shown in Figure 3.13 is further assessed using a new dataset from the Face Recognition Grand Challenge (FRGC) database [71]. In particular, this new testing set includes 456 color images of 152 people, and two thirds of the images are taken in uncontrolled environments with challenging image quality in terms of illumination, resolution, blurring, etc. As three images are available for each subject, two of them are randomly chosen for training, while the remaining one is used for testing.

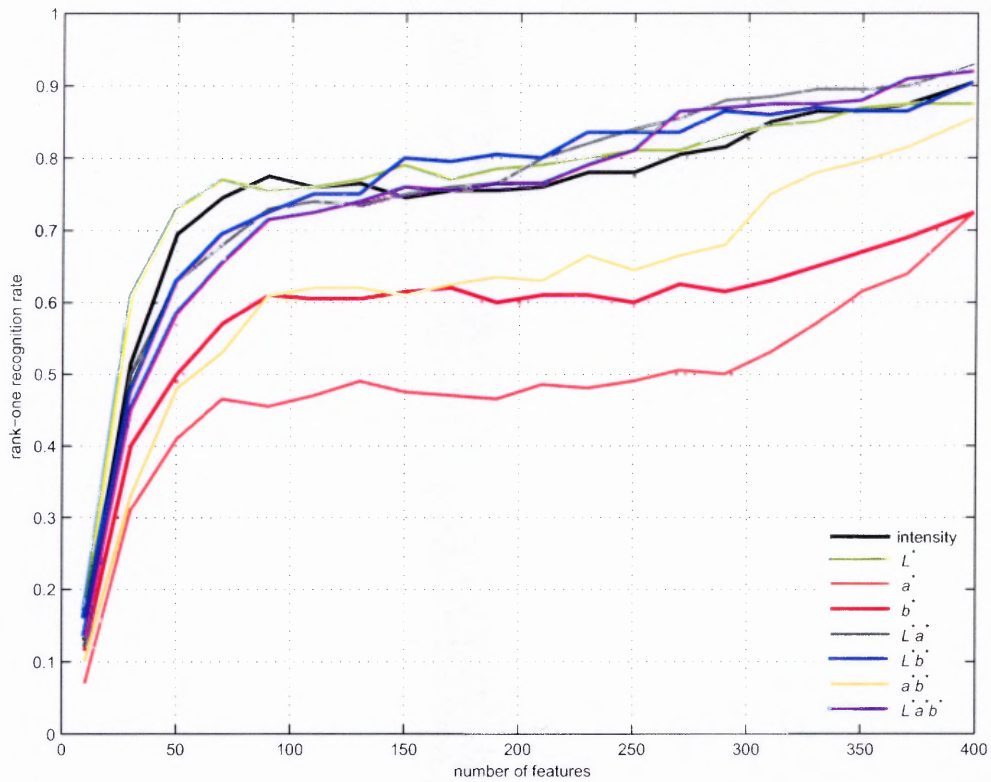
The face recognition performance of the seven color configurations is shown in Figure 3.14. Again, all these seven color configurations have better face recognition perfor-



**Figure 3.7** The face recognition performance of seven color configurations in the  $YIQ$  color space.

mance than the intensity images, even though the overall face recognition rates are lower than those shown in Figure 3.13. The reason for the lower recognition rates is mainly due to the FRGC dataset, which contains two thirds of images taken in uncontrolled environments with challenging image quality.

The last set of experiments assesses the face recognition performance with image alignment variations in terms of varying the eye locations to assess the robustness of face recognition performance using the color configurations shown in Figure 3.14 and Figure 3.13. The experimental results show that (i) small variations in eye locations, such as in  $5 \times 5$  pixel region, do not change the face recognition performance much; and (ii) larger variations, such as  $11 \times 11$  pixel region, can significantly deteriorate the face recog-



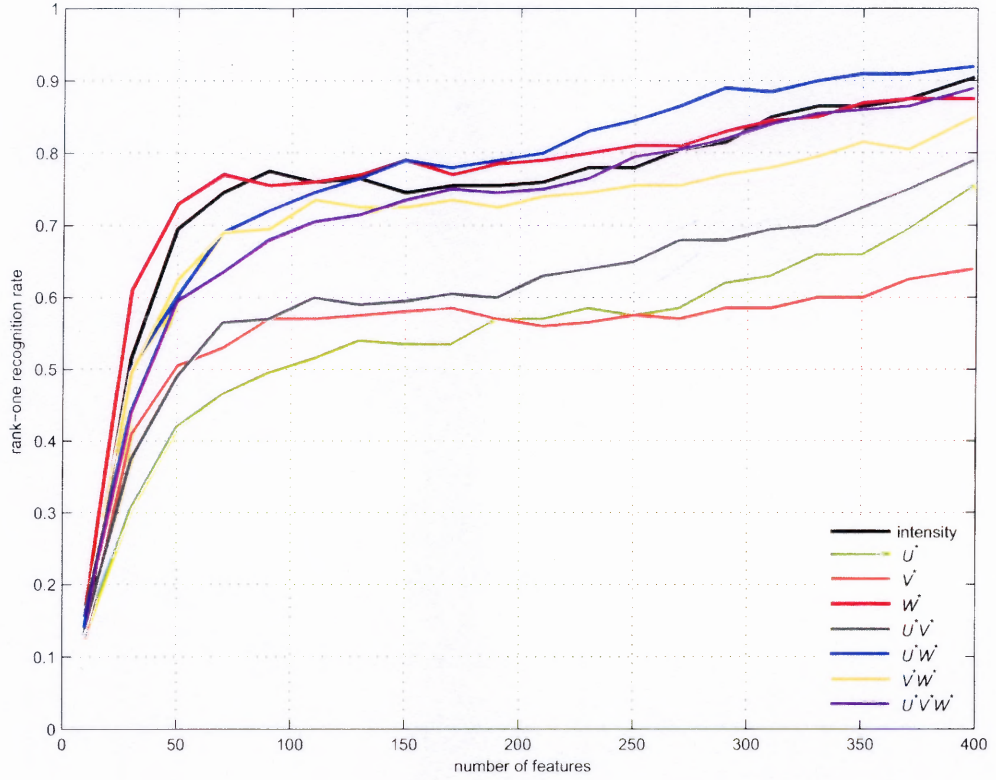
**Figure 3.8** The face recognition performance of seven color configurations in the  $L^*a^*b^*$  color space.

inition performance. Figure 3.15 shows the deteriorated face recognition performance when the eye location variations occur in  $11 \times 11$  pixel region using the FRGC dataset.

### 3.4 Feature Extraction Using Component Images Across Color Spaces

Face recognition has received significant attention during the past decade. Recently, Phillips and Newton [72] suggest that researchers should concentrate on “face recognition problems that are harder, as defined by the image sets in the experiments and the performance by a control algorithm” rather than work on problems that have already been solved. To this end, the Face Recognition Grand Challenge (FRGC) databases are designed, and a control algorithm, known as the Biometric Experimentation Environment (BEE) baseline algo-



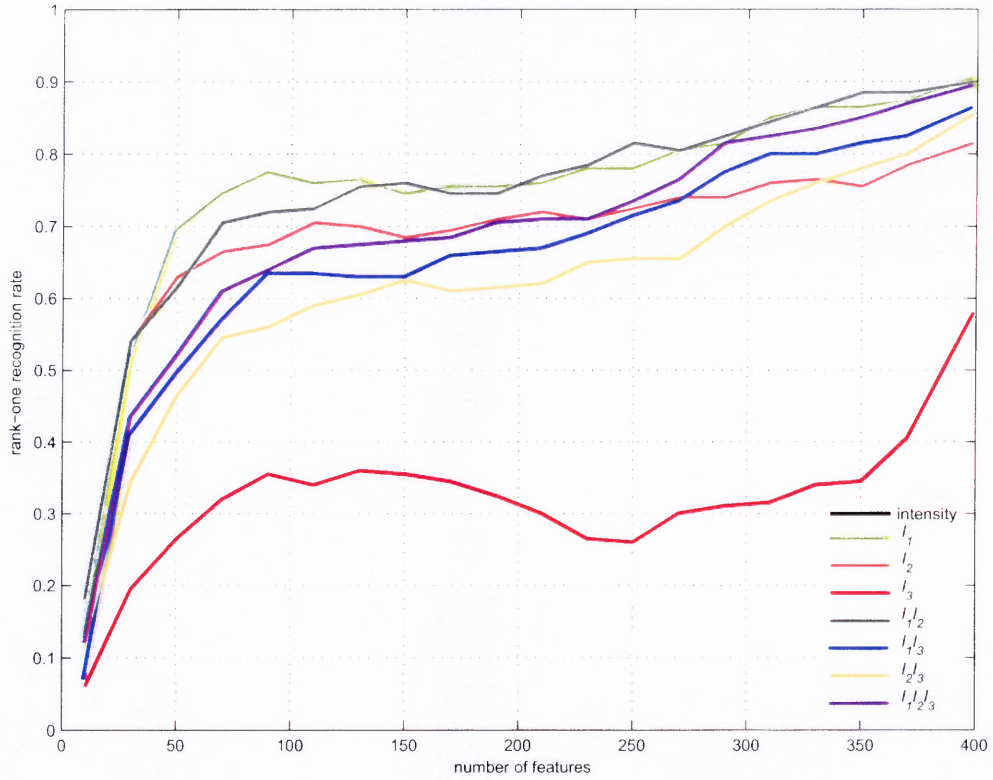


**Figure 3.9** The face recognition performance of seven color configurations in the  $U^*V^*W^*$  (or  $L^*u^*v^*$ ) color space.

rithm [66], is developed. Due to its large scale and standardized protocols, FRGC has been considered the de facto benchmark to evaluate face recognition algorithms. This section focuses on extending the color research to improve the BEE baseline performance.

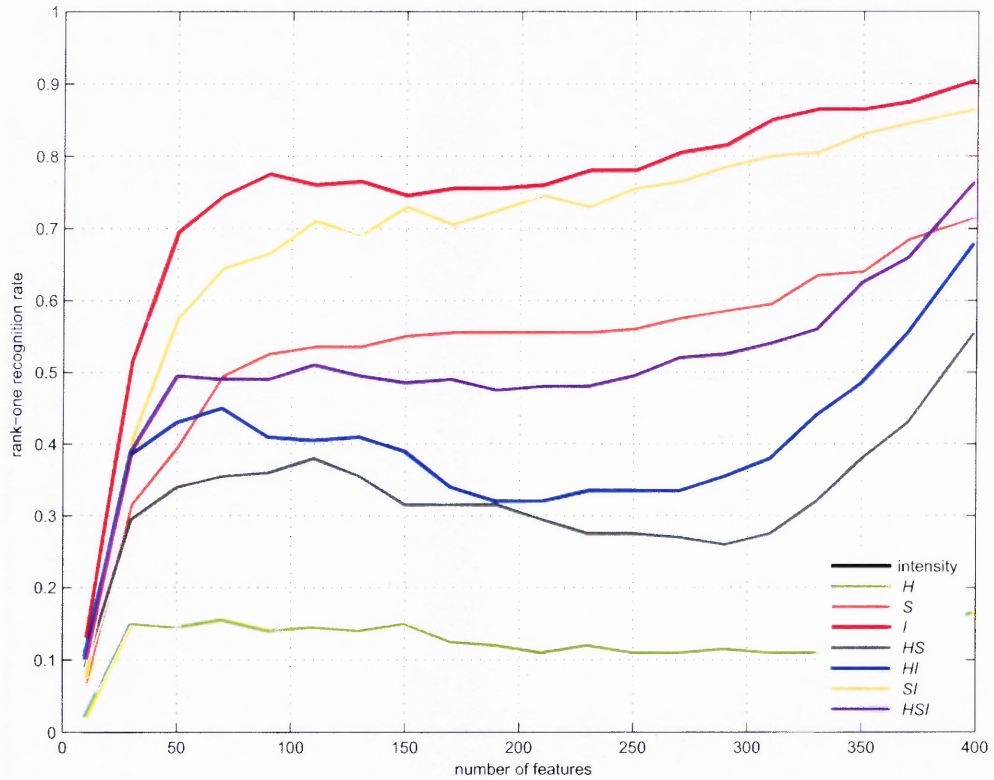
Based on the experimental results presented in Subsection 3.3.2, the color configurations defined by the pure achromatic component (such as  $Y$ ) or the pure chromatic components (such as  $I$ ,  $Q$ ,  $C_b$ ,  $C_r$ ,  $IQ$ , or  $C_bC_r$ ) do not improve face recognition performance. This section thus concentrates on color configurations that comprise both achromatic and chromatic component images. In particular, the face recognition performance is empirically assessed using the  $YI$ , the  $YQ$ , the  $YIQ$  color configurations in the  $YIQ$  color space and the  $YC_b$ , the  $YC_r$ , the  $YC_bC_r$  color configurations in the  $YC_bC_r$  color space. The color configurations defined by combining color component images across different color





**Figure 3.10** The face recognition performance of seven color configurations in the  $I_1I_2I_3$  color space.

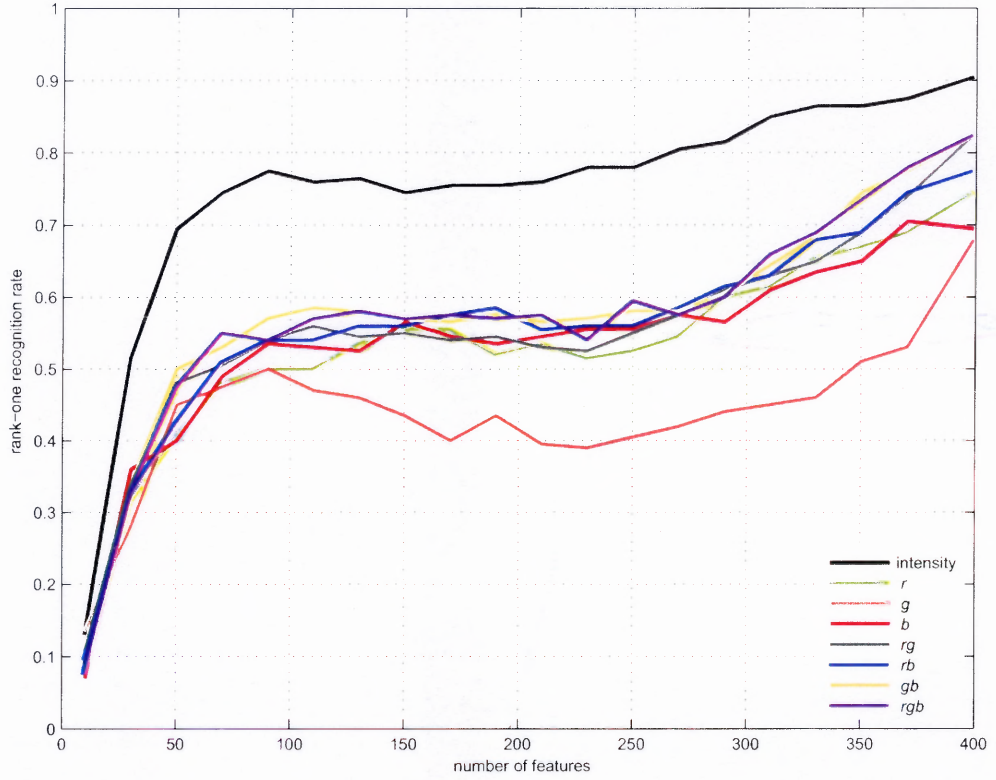
spaces are investigated as well. Specifically, the  $YIC_r$ , the  $YIC_b$ , the  $YQC_r$ , and the  $YQC_b$  color configurations defined by the color component images from both the  $YIQ$  and the  $YC_bC_r$  color spaces are assessed. The experimental results show that some color configurations, such as the  $YQC_r$  color configuration, achieves superior face recognition performance. In particular, when using an FRGC ver1.0 dataset containing 366 training images, 152 controlled gallery images, and 608 uncontrolled probe images, the  $YQC_r$  color configuration improves the rank-one face recognition rate of the BEE baseline algorithm from 37% to 70%; when using an FRGC ver2.0 dataset including 6,660 training images, 16,028 controlled target images, and 8,014 uncontrolled query images, the  $YQC_r$  color configuration improves the face verification rate (at 0.1% false acceptance rate) of the BEE baseline algorithm from 13% to 33%.



**Figure 3.11** The face recognition performance of seven color configurations in the  $HSI$  color space.

### 3.4.1 Methodology

Face recognition challenge problems refer to the “face recognition problems that are harder, as defined by the image sets in the experiments and the performance by a control algorithm” [72]. Two versions of FRGC, namely the ver1.0 and the ver2.0, are distributed [71]. In particular, the ver1.0 is designed to introduce the FRGC challenge problem format and the supporting infrastructure to the face recognition community, while the FRGC ver2.0 is designed to challenge researchers to meet the FRGC performance goal [71]. There are 7,544 and 50,000 recordings in the ver1.0 and the ver2.0, respectively, and each of them is divided into three subsets: the *training* set, the *target* set, and the *query* set. The images from the training set are used to learn specific parameters or settings of a particular

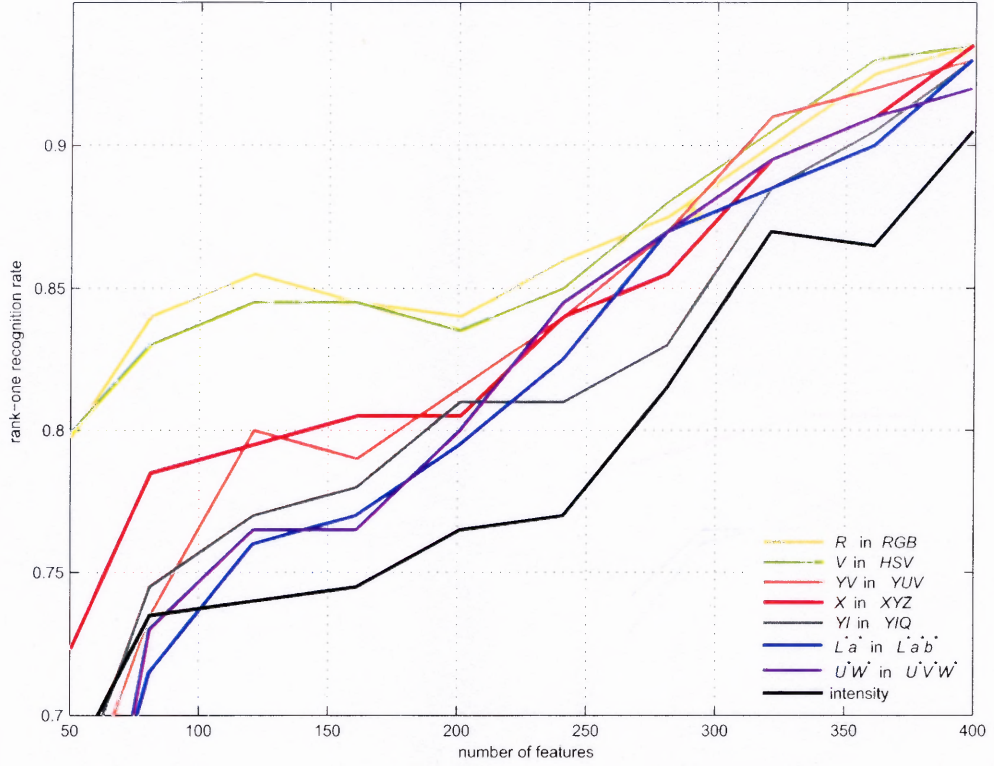


**Figure 3.12** The face recognition performance of seven color configurations in the *rgb* color space.

recognition algorithm, and the target and the query sets are applied to validate the trained system.

The BEE baseline algorithm is a principal component analysis (PCA) algorithm, which has been optimized for large scale problems [66]. Specifically, in the FRGC databases, the eigenvectors are derived using the images from the training set following the procedure described in Subsection 3.3.1. After the training, the target and the query sets are applied to derive the face recognition performance using a nearest neighbor classifier. First, the query feature set  $Q$  of size  $|Q|$  is obtained by projecting the query pattern vectors onto the eigenvectors of PCA (see Equation 3.15). The target feature set  $T$  of size  $|T|$  is produced in a similar manner. Then a similarity matrix  $S$  of size  $|Q| \times |T|$  is derived by calculating a similarity score between every query-target pair. Let  $\mathbf{q}_i$  represents a query feature vector in





**Figure 3.13** Comparative face recognition performance of the best color configurations in the *RGB* color space, the *HSV* color space, the *YUV/YC<sub>b</sub>C<sub>r</sub>* color space, the *XYZ* color space, the *YIQ* color space, the *L\*a\*b\** color space, and the *U\*V\*W\*/L\*u\*v\** color space.

$\mathcal{Q}$  and  $\mathbf{t}_j$  a target feature vector in  $\mathcal{T}$ , similarity score  $\mathbf{S}(i, j)$  between  $\mathbf{q}_i$  and  $\mathbf{t}_j$  is calculated using a whiten cosine similarity measure [71]:

$$\mathbf{S}(i, j) = \frac{-\mathbf{q}_i^t \mathbf{\Lambda}^{-1} \mathbf{t}_j}{\|\mathbf{\Lambda}^{-1/2} \mathbf{q}_i\| \|\mathbf{\Lambda}^{-1/2} \mathbf{t}_j\|} \quad (3.22)$$

where  $\mathbf{\Lambda}$  is defined in Equation 3.15, and  $\|\cdot\|$  denotes the norm operator. Finally, the similarity matrix  $\mathbf{S}$  is analyzed by BEE to determine the recognition performance.

In the FRGC ver1.0, face recognition performance is derived using a *gallery* set and a *probe* set, which are drawn from the target and the query sets, respectively. Due to the fact that only one image is available for each subject in the gallery set, the recognition

**Table 3.1** Color Configurations that Outperform Intensity Images

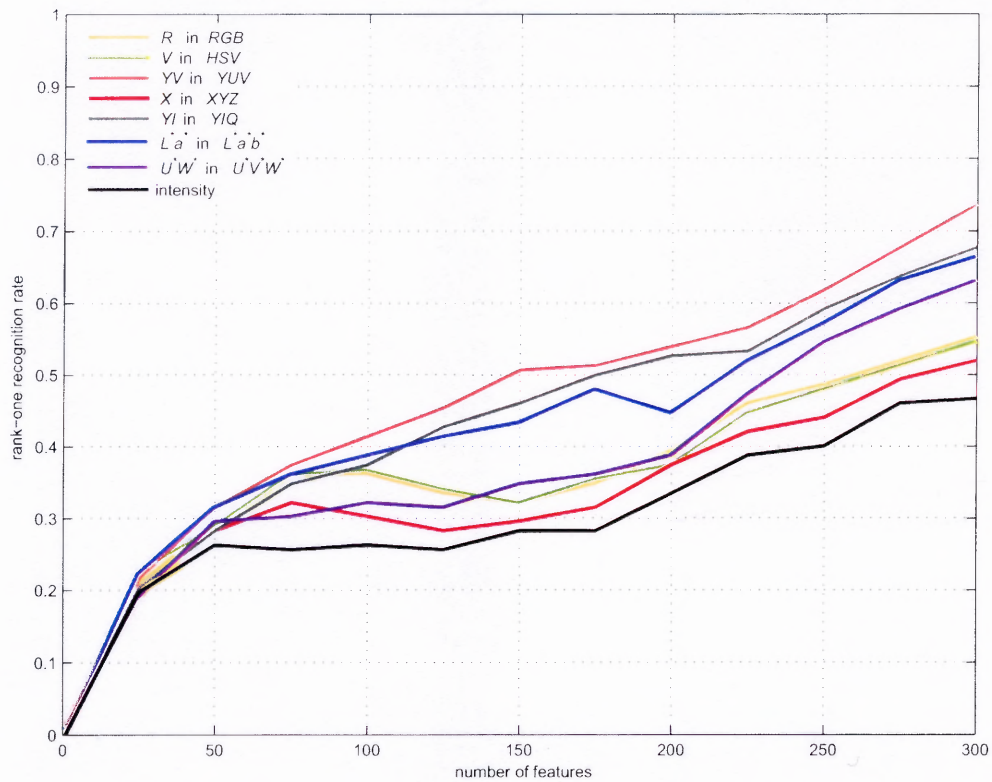
color space	color configurations
<i>RGB</i>	<i>R, RG</i>
<i>HSV</i>	<i>V</i>
<i>YUV</i>	<i>Y, YV</i>
<i>YC<sub>b</sub>C<sub>r</sub></i>	<i>Y, YC<sub>r</sub></i>
<i>XYZ</i>	<i>X, Y, XY</i>
<i>YIQ</i>	<i>Y, YI</i>
<i>L<sup>*</sup>a<sup>*</sup>b<sup>*</sup></i>	<i>L<sup>*</sup>, L<sup>*</sup>a<sup>*</sup>, L<sup>*</sup>b<sup>*</sup>, L<sup>*</sup>a<sup>*</sup>b<sup>*</sup></i>
<i>U<sup>*</sup>V<sup>*</sup>W<sup>*</sup></i>	<i>W<sup>*</sup>, U<sup>*</sup>W<sup>*</sup></i>
<i>L<sup>*</sup>u<sup>*</sup>v<sup>*</sup></i>	<i>L<sup>*</sup>, L<sup>*</sup>u<sup>*</sup></i>

performance is characterized by the rank order statistics and graphically displayed as cumulative match characteristic (CMC) curves, where rank  $r$  indicates the probability that the gallery image from the correct individual is among the top  $r$  matches to a probe. Note that a probe,  $\mathbf{p}_i$ , is compared against every gallery and is classified as belonging to the class of the closest match by a nearest neighbor classifier [71]:

$$\delta(\mathbf{p}_i, \mathbf{g}_k) = \max_j \delta(\mathbf{p}_i, \mathbf{g}_j) \longrightarrow \mathbf{p}_i \in \omega_k \quad (3.23)$$

where  $\mathbf{g}_k$ ,  $k = \{1, 2, \dots, L\}$  is the gallery feature vector for class  $\omega_k$ , and  $\delta(\mathbf{p}_i, \mathbf{g}_k)$  is a whiten cosine similarity measure.

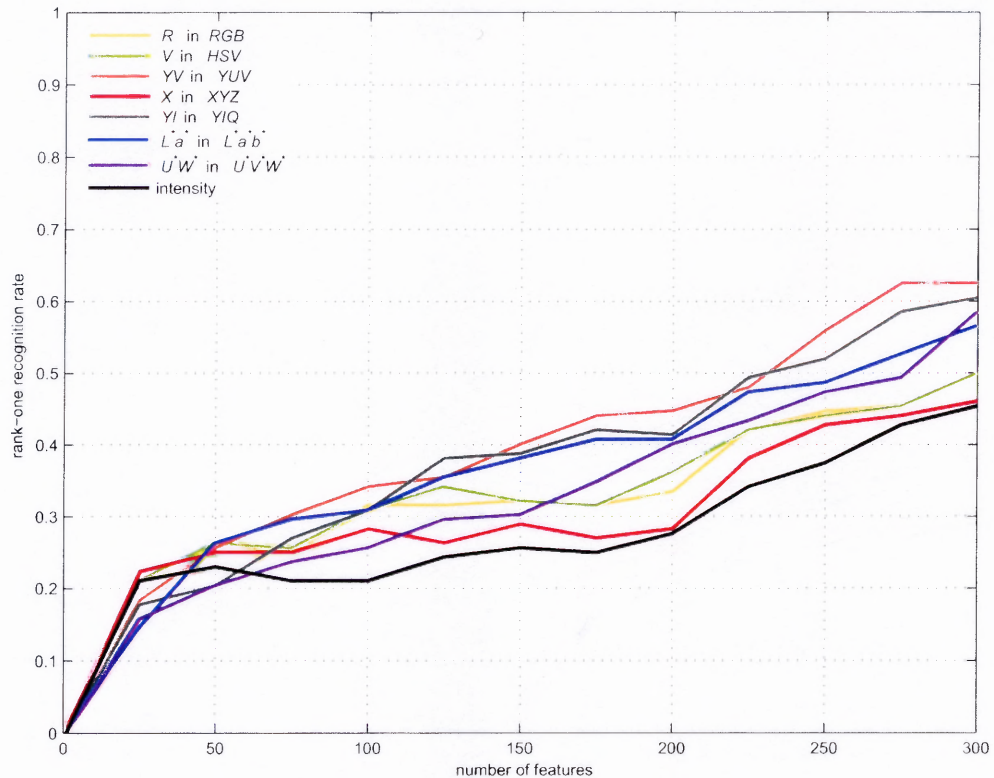
In the FRGC ver2.0, the target and the query sets are directly used to derive recognition performance. In particular, BEE analyzes the similarity matrix,  $\mathbf{S}$ , and generates three receiver operating characteristic (ROC) curves using three different subsets [71]: The first subset, corresponding to the ROC I, contains the target and the query images that were taken in the same year, and the query was taken seven or more days after the target. The



**Figure 3.14** Comparative face recognition performance of seven color configurations using 456 FRGC color images.

second subset, corresponding to the ROC II, consists of the target and the query images where the query was taken 7 or more days after the target, regardless of year. The third subset, corresponding to the ROC III, includes the target and the query images where the query was taken in a later year than the target.

The BEE baseline performance, either in the FRGC ver1.0 or the FRGC ver2.0, is derived by applying the intensity images defined by averaging the red, green, and blue component images in the *RGB* color space. The BEE baseline algorithm is a PCA implementation with a specific configuration. In particular, the BEE baseline algorithm applies 146 PCA features to derive the FRGC ver1.0 baseline performance, and 1,024 PCA features to derive the FRGC ver2.0 baseline performance.



**Figure 3.15** Comparative face recognition performance of seven color configurations using 456 FRGC color images with eye location variations occurred in  $11 \times 11$  pixel region.

### 3.4.2 Experiments

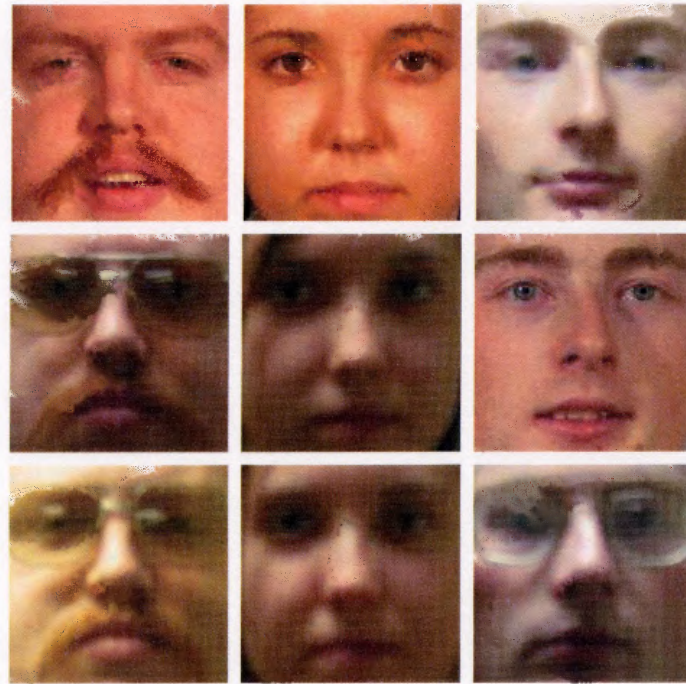
The FRGC is designed to challenge researchers and advance the face recognition technologies by presenting a six-experiment challenge problem and data corpus of 50,000 images [71]. Among these six experiments, the BEE baseline algorithm reveals that Experiment 4 designed to measure face recognition performance using uncontrolled frontal still images is the most challenging one [71]. Experiment 4 is thus chosen to evaluate the feasibility of the color feature extraction method. Specifically, 1,126 images (consisting of 366 *training* images, 152 controlled *gallery* images, and 608 uncontrolled *probe* images) from the FRGC ver1.0, and 30,702 images (consisting of 6,660 *training* images, 16,028 controlled *target* images, and 8,014 uncontrolled *query* images) from the FRGC ver2.0 are used. Note that the 6,660 training images are a subset of the *large still training set* [71],

from which 30 images for each subject are randomly chosen. The example target, query, and training images are shown in Figure 3.16. The top image in either column one or column two is a target image, while the remaining two are query images. Images in the third column are training images. Figure 3.16 shows that the controlled target images have better image quality than the uncontrolled query images in terms of illumination, resolution, blurring, etc.

The preprocessing procedure of this part of experiments normalizes the face images to a spatial resolution of  $64 \times 64$  to extract the facial regions that contain only the faces, so that the performance of face recognition is not affected by the factors not related to face, such as hair style. Figure 3.16(a) shows some example target, query, and training images of the size of  $64 \times 64$ . Note that in either column one or column two, the top image is a controlled target image, while the remaining two are uncontrolled query images. The third column shows example training images. Figure 3.16 shows that the controlled target images have better image quality than the uncontrolled query images in terms of illumination, resolution, blurring, etc. Note that this preprocessing procedure simplifies the one performed by the BEE baseline algorithm in two aspects: (i) BEE preprocessing normalizes images to a larger resolution of  $130 \times 150$ , and (ii) BEE preprocessing applies an oval mask to extract the facial region. Figure 3.16(b) shows a BEE intensity image normalized to  $130 \times 150$  with an oval mask.

The first set of the experiments assesses the face recognition performance of the BEE baseline algorithm using an FRGC ver1.0 dataset that contains 366 training images, 152 gallery images, and 608 probe images. The training images are taken from subjects who do not appear in either the gallery or the probe set. The gallery images are taken from 152 subjects in controlled environment with good illumination conditions. The probe images are taken from the same 152 subjects in uncontrolled environment with challenging image quality. Note that the number of probe images for each subject ranges from two to seven.



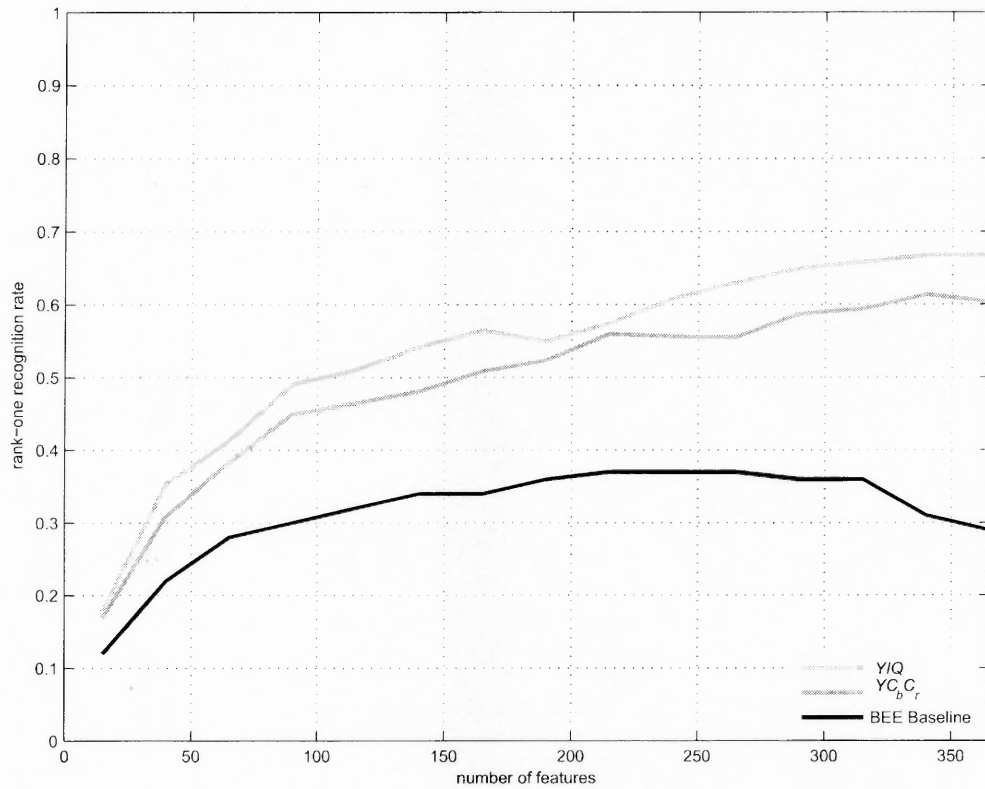


(a)



(b)

**Figure 3.16** (a) Example target, query, and training images normalized to  $64 \times 64$ . The top image in either column one or column two is a target image, while the remaining two are query images. Images in the third column are training images. (b) A BEE intensity image of the size  $150 \times 130$  with an oval mask applied to extract the face region.

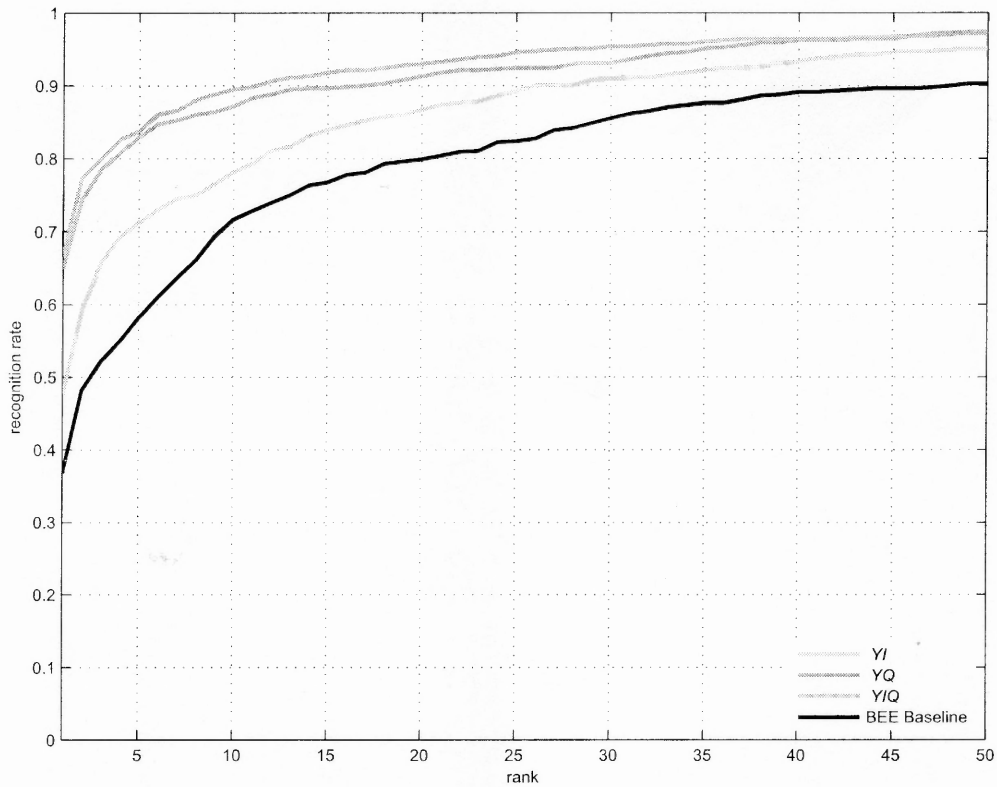


**Figure 3.17** Comparative face recognition performance using the  $YIQ$  color configuration, the  $YCbCr$  color configuration, and the BEE intensity images. The horizontal axis indicates the number of features used, and the vertical axis represents the rank-one face recognition rate, which is the accuracy rate for the top response being correct.

Among seven color configurations, the experimental results presented in Subsection 3.3.2 show that the pure achromatic components, such as the  $Y$  component, or the pure chromatic components, such as the  $I$  component, the  $Q$  component, or the  $IQ$  component, do not improve face recognition performance. The experiments conducted here thus only focus on color configurations that comprise both achromatic and chromatic component images. Specifically, the face recognition performance of the  $YI$ , the  $YQ$ , and the  $YIQ$  color configurations in the  $YIQ$  color space, and the  $YC_b$ , the  $YC_r$ , and the  $YC_bCr$  color configurations in the  $YCbCr$  color space is assessed.

Figure 3.17 shows comparative face recognition performance of the BEE baseline algorithm using the  $YIQ$  color configuration, the  $YC_bC_r$  color configuration, and the BEE intensity images (see Equation 3.2). The horizontal axis indicates the number of features used, and the vertical axis represents the rank-one face recognition rate, which is the accuracy rate for the top response being correct. The curves corresponding to the  $YIQ$  and the  $YC_bC_r$  color configurations indicate that using more features generally leads to a higher recognition accuracy. Note that the number of eigenvectors derived by the BEE baseline algorithm using 366 training images is 365; hence the total number of features derived by projecting a color configuration onto these eigenvectors is 365. The curve associated to the BEE baseline, however, shows different tendency where the face recognition performance peaks when a certain number of features (about 40% of the total features) are used. Note that the face recognition performance of the  $YI$ , the  $YQ$ , the  $YC_b$ , and the  $YC_r$  color configurations is also assessed using different number of features, and the experimental results all indicate that using more features results in higher recognition accuracy. The results thus suggest that one should use 365 features for the color configurations and 146 features (accounting for 40% of total features) for the BEE intensity images in order to conduct a fair comparison.

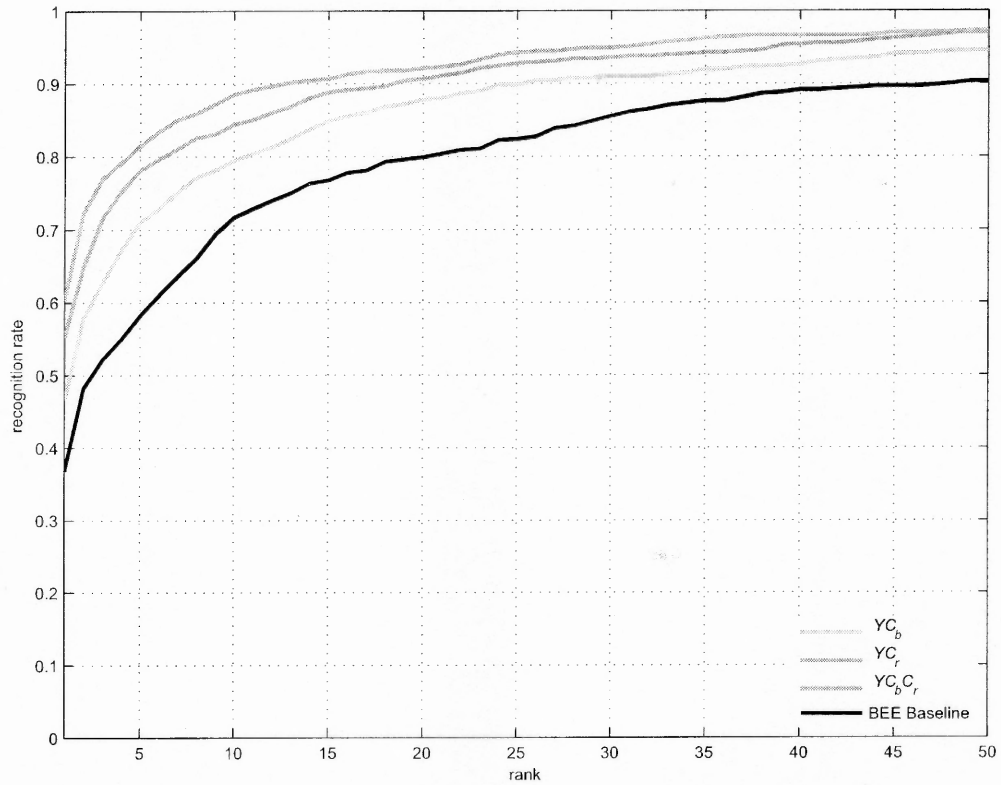
Figure 3.18 and Figure 3.19 show the cumulative match characteristic (CMC) curves derived using the color configurations defined in the  $YIQ$  and the  $YC_bC_r$  color spaces, respectively. The BEE baseline performance is also included for comparison. The horizontal axis denotes the rank, where rank  $k$  indicates the probability that the gallery image from the correct individual is among the top  $k$  matches to a probe, and the vertical axis represents the correct recognition rate. In particular, Figure 3.18 shows that in the  $YIQ$  color space, the  $YIQ$  color configuration performs the best, followed in order by the  $YQ$  color configuration, and the  $YI$  color configuration; Figure 3.19 shows that in the  $YC_bC_r$  color space, the  $YC_bC_r$  color configuration performs the best, followed in order by the  $YC_r$  color configuration, and the  $YC_b$  color configuration. Note that the CMC curves corresponding



**Figure 3.18** The CMC curves derived using the color configurations defined in the  $YIQ$  color space. The BEE baseline performance is also included for comparison. Note that the curves corresponding to the color configurations and the BEE baseline performance are derived using 365 and 146 features, respectively.

to the color configurations and the BEE baseline performance are derived using 365 and 146 features, respectively.

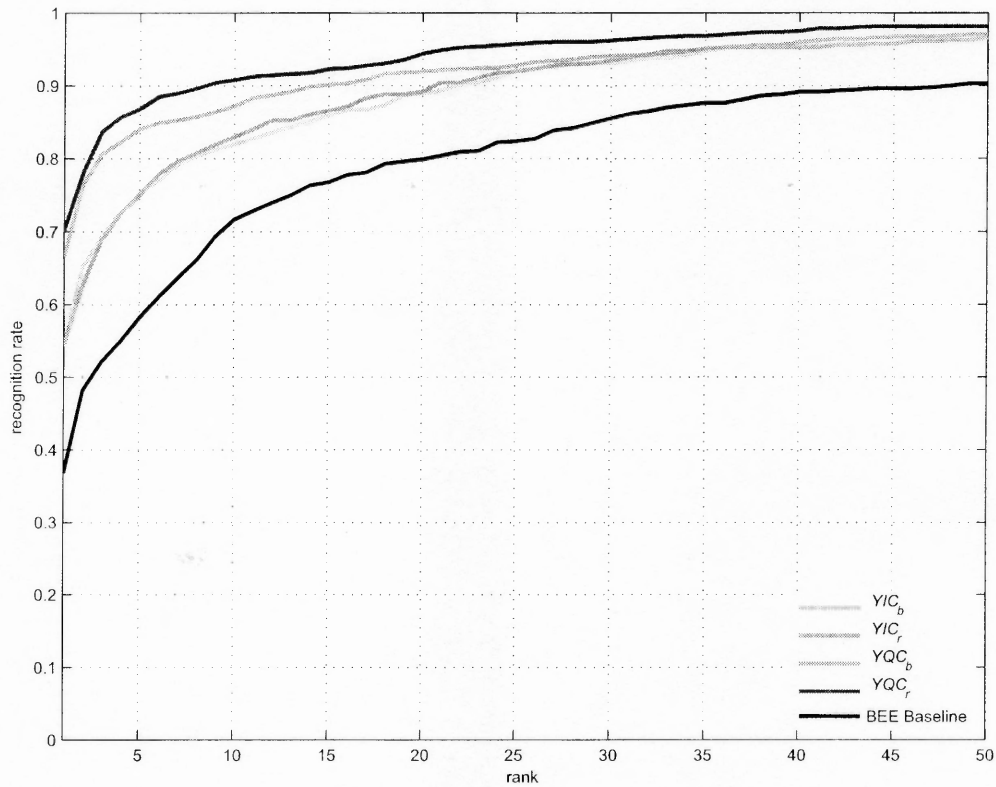
The second set of experiments explores the possibility of defining the color configurations using component images across different color spaces. This experiment applies the FRGC ver1.0 dataset, and focus on the color configurations comprising both achromatic and chromatic components. Specifically, the face recognition performance of the  $YIC_r$ , the  $YIC_b$ , the  $YQC_r$ , and the  $YQC_b$  color configurations is assessed. Figure 3.20 shows the CMC curves derived using these four color configurations. In particular, the top curve in Figure 3.20 corresponding to the  $YQC_r$  color configuration displays the best face



**Figure 3.19** The CMC curves derived using the color configurations defined in the  $YC_b C_r$  color space. The BEE baseline performance is also included for comparison. Note that the curves corresponding to the color configurations and the BEE baseline performance are derived using 365 and 146 features, respectively.

recognition performance followed in order by the  $YQC_b$ , the  $YIC_r$ , and the  $YIC_b$  color configurations. The rank-one face recognition rate of the best color configuration is 70% comparing to the BEE baseline rank-one rate of 37%.

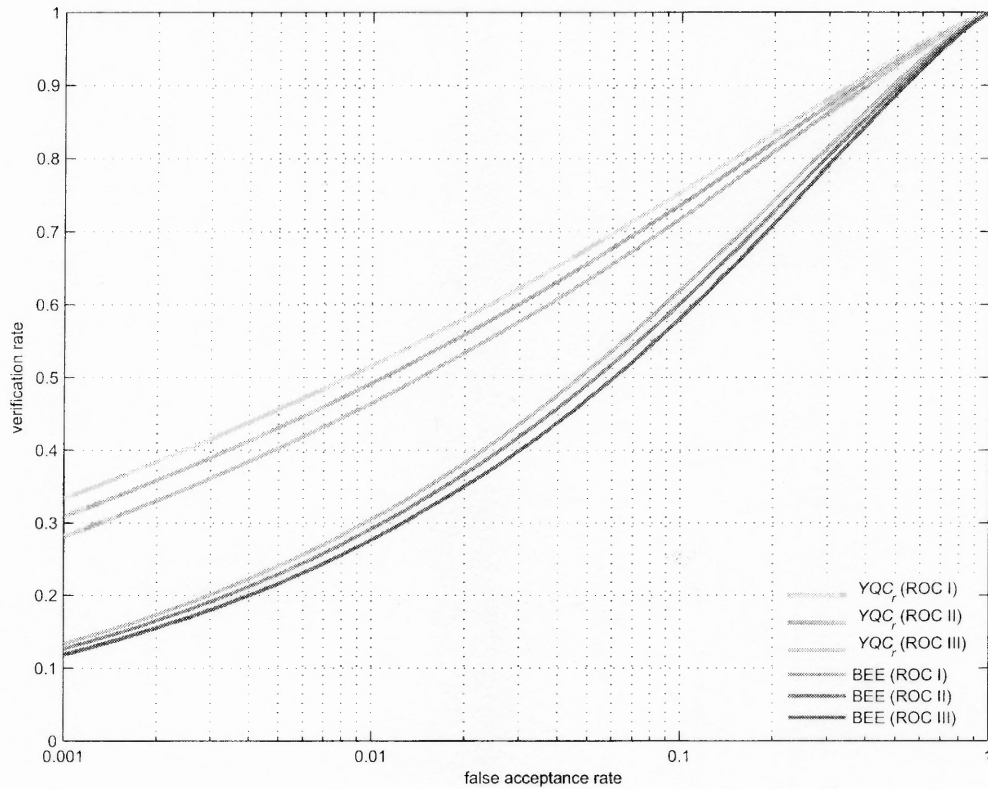
The last set of experiments applies the best performing color configuration,  $YQC_r$ , onto an FRGC ver2.0 dataset consisting of 12,776 training images, 16,028 controlled target images, and 8,014 uncontrolled query images. Note that 12,776 training images are not all used for training, but rather, 30 images for each subject are randomly chosen to form a smaller training set. As there are 222 subjects available in the original training set, this smaller training set thus contains 6,660 images. Again, the BEE baseline algorithm derives



**Figure 3.20** The CMC curves derived using the color configurations defined across different color spaces. The BEE baseline performance is also included for comparison. Note that the curves corresponding to the color configurations and the BEE baseline performance are derived using 365 and 146 features, respectively.

at most  $(6660 - 1)$  eigenvectors, and the maximum dimensionality of the feature vectors is 6659.

In the FRGC ver2.0, as detailed in Subsection 3.4.1, BEE analyzes the similarity matrix and generates three receiver operating characteristic (ROC) curves, namely the ROC I, the ROC II, and the ROC III, corresponding to three different subsets. The ROC curves show the false acceptance rate (FAR) on the horizontal axis and the probability of correct verification (computed as  $1 - \text{false rejection rate (FRR)}$ ) on the vertical axis. The face recognition performance using the  $YQC_r$  color configuration and the BEE intensity images is shown in Figure 3.21. In particular, Figure 3.21 shows that the  $YQC_r$  color configuration



**Figure 3.21** The ROC curves derived using the  $YQC_r$  color configuration and the BEE intensity images.

improves ROC I face verification rate (at 0.1% false acceptance rate) from 13% to 33%; ROC II face verification rate from 13% to 31%; and ROC III face verification rate from 12% to 28%. Table 3.2 summarizes the face recognition performance of the  $YQC_r$  color configuration and the BEE baseline performance using the FRGC databases.

### 3.5 Conclusion

This chapter presents extensive assessments of face recognition performance in different color spaces, and defines a color feature extraction method to help improve the baseline performance of the FRGC problems. The comparative assessments of face recognition performance are implemented in twelve color spaces by evaluating seven color configurations

**Table 3.2** Comparative Face Recognition Performance of the  $YQC_r$  Color Configuration and the BEE Intensity Images

	BEE Baseline	$YQC_r$
FRGC v1	37%	70%
FRGC v2 (ROC I)	13%	33%
FRGC v2 (ROC II)	13%	31%
FRGC v2 (ROC III)	12%	28%

for every color space. The experimental results show that some color configurations, such as  $YV$  in the  $YUV$  color space and  $YI$  in the  $YIQ$  color space, help improve face recognition performance. In addition, a color feature extraction method, which uses the color configurations defined by the color component images across different color spaces, is implemented to improve the FRGC baseline performance. Experimental results using both the FRGC ver1.0 and ver2.0 datasets demonstrate the proposed feature extraction method helps improve face recognition performance. In particular, the best performing color configuration,  $YQC_r$ , achieves 70% rank-one face recognition rate comparing to the FRGC ver1.0 baseline performance of 37%, and achieves 33% face verification rate (at 0.1% FAR) comparing to the FRGC ver2.0 baseline performance of 13%.



## CHAPTER 4

### FACE RECOGNITION USING EVOLUTIONARY COLOR FISHER FEATURES

In the previous chapter, extensive assessments of face recognition performance in different color spaces have been performed, and a color feature extraction method has been designed to improve the baseline performance of the Face Recognition Grand Challenge (FRGC) problems. The best performing color configuration,  $YQC_r$ , defined in the  $YIQ$  and the  $YC_bC_r$  color spaces, achieves 33% verification rate (at 0.1% false acceptance rate) comparing to the BEE baseline performance of 13% at the same false acceptance rate. Although this research discovers a configuration that significantly improves upon the BEE baseline performance, however, it limits the optimization within well-defined color spaces, such as  $RGB$ ,  $YC_bC_r$ ,  $YIQ$ , etc. To further elevate the FRGC performance, this chapter presents a novel feature extraction method for face recognition implementing genetic algorithms (GAs) and Fisher linear discriminant (FLD) to derive the optimal discriminating features that lead to an effective image representation for face recognition.

The method works as follows. First, a new color space,  $LC_1C_2$ , consisting of one luminance ( $L$ ) channel and two chrominance channels ( $C_1, C_2$ ) is introduced as a linear transformation of the input  $RGB$  color space. The specific transformation from the  $RGB$  color space to the  $LC_1C_2$  color space is optimized by a GA where a fitness function guides the evolution toward higher recognition accuracy. After the color feature extraction, i.e., transforming  $RGB$  color images to  $LC_1C_2$  color images, the FLD method is applied to further extract discriminating features, termed color Fisher features (CFFs). Finally, CFFs are applied to enhance FRGC baseline performance.

The feasibility of the proposed method is demonstrated using the FRGC databases. Specifically, using an FRGC ver1.0 dataset containing 366 training images, 152 controlled gallery images, and 608 uncontrolled probe images, the  $LC_1C_2$  color features improve

the rank-one face recognition rate of the Biometric Experimentation Environment (BEE) baseline performance from 37% to 75%, and CFFs achieve 73% rank-one rate using the same dataset. When using an FRGC ver2.0 dataset consisting of 6,660 training images, 16,028 controlled target images, and 8,014 uncontrolled query images, the  $LC_1C_2$  color features improve the face verification rate at 0.1% false acceptance rate (FAR) of the BEE baseline performance from 12% to 32%, and CFFs achieve 69% verification rate at the same fixed FAR of 0.1%.

The remainder of this chapter is organized as follows. Section 4.1 briefly reviews prior pattern recognition research implementing evolutionary computations. Section 4.2 defines the  $LC_1C_2$  color space in Subsection 4.2.1, details the evolutionary procedure that optimizes the  $RGB$  to  $LC_1C_2$  transformation in Subsection 4.2.2, and presents CFF derivation in Subsection 4.2.3. Section 4.3 elaborates the experimental setup and reports the experimental results. Finally, Section 4.4 summarizes this face recognition approach.

## 4.1 Background

Although color information has been widely applied to face detection [85], [40], [114], more research is needed for applying color to face recognition. Torres et al. [100] showed that color images do not significantly improve the face recognition performance in comparison with the intensity images. Yip and Sinha [115], on the other hand, suggested that color information improve face recognition accuracy when low-resolution images are applied. Recently, Shih and Liu assessed the performance of content-based face image retrieval using twelve color spaces. Their experimental results showed that some color configurations help improve the face retrieval performance [86].

Genetic algorithms [39], [28], based on the Darwinian evolutionary theory, have ubiquitous applications in solving optimization problems. Lee et al. [50] applied GA in a handwritten numeral recognition system to avoid local minima in training a multilayer cluster neural network. Sakano [79] utilized GA as a translator to translate high dimensional

feature vectors into character images in optical character readers (OCRs). Bebis et al. [2] presented a model-based approach that applied GAs for recognizing two-dimensional (2D) or three-dimensional (3D) objects from 2D intensity images. Cho and Chi [15] recently developed a structural pattern recognition method, which applies the genetic evolution to tune the learning parameters and improve the classification accuracy.

In the context of face recognition [53], [54] or detection [85], [52], GAs are often applied as feature extraction / selection techniques. Vafaie and De Jong [103] presented a GA-based representation transformation to create transformed feature spaces. The application of this transformation showed an improvement of classification performance. Jeon et al. [44] proposed an evolutionary module by implementing parallel GAs to evolve filter blocks. This evolutionary architecture is capable of recognizing faces in an environment with illumination variations. Xu et al. [110] developed a face recognition system, which adopted GAs to select optimal features derived from an independent component analysis (ICA) implementation.

## 4.2 Methodology

The proposed method integrates GAs and FLD to derive optimal discriminating features for face recognition. In the following subsections, Subsection 4.2.1 defines the  $LC_1C_2$  color space, Subsection 4.2.2 details the evolutionary procedure that optimizes the  $RGB$  to  $LC_1C_2$  transformation, and Subsection 4.2.3 presents CFF derivation.

### 4.2.1 The $LC_1C_2$ Color Space

To improve face recognition performance using unconventional color spaces, a new color space,  $LC_1C_2$ , is introduced. The  $LC_1C_2$  color space is defined by one luminance ( $L$ ) channel and two chrominance channels ( $C_1, C_2$ ). The luminance channel,  $L$ , is designed to capture intensity or luminance characteristics of an image, and the chrominance channels,  $C_1$  and  $C_2$ , are to extract color properties, such as hue and saturation. The transformation

from the  $RGB$  color space to the  $LC_1C_2$  color space is as follows [87]:

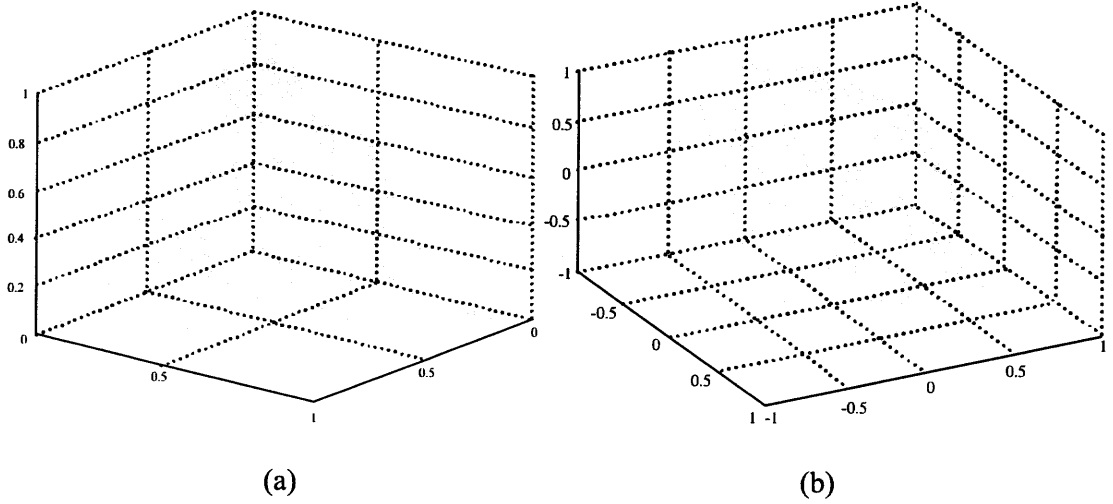
$$\begin{bmatrix} L \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (4.1)$$

where the coefficients,  $a_1, a_2, \dots, a_9$ , are not explicitly specified; but rather, they will be optimized by GA where a fitness function guides the evolution toward higher recognition accuracy.

The definition of the  $LC_1C_2$  color space is motivated by the research results presented in Chapter 3 that assesses the face image retrieval performance in twelve conventional color spaces [86]. The experimental results show that some color configurations in the linear color spaces, such as  $YIQ$  and  $YUV$ , help improve the face retrieval performance. To incorporate the characteristics of these two color spaces, five constraints are imposed on the coefficients  $a_1, a_2, \dots, a_9$  [87]:

$$\left\{ \begin{array}{l} 0 < a_i < 1, \quad \text{if } i = 1 \text{ or } 2 \text{ or } 3 \\ -1 < a_i < 1, \quad \text{otherwise} \\ a_1 + a_2 + a_3 = 1 \\ a_4 + a_5 + a_6 = 0 \\ a_7 + a_8 + a_9 = 0 \end{array} \right. \quad (4.2)$$

Note that the constraints specified in Equation 4.2 not only preserve the properties of the  $YUV$  and the  $YIQ$  color spaces but also significantly reduce the search space of the coefficients. In particular, Figure 4.1(a) shows that the search space of  $a_1, a_2, a_3$  has been reduced from a three-dimensional space to the shaded triangle area, and Figure 4.1(b) shows that the search space of  $a_4, a_5, a_6$  (or  $a_7, a_8, a_9$ ) has been reduced from a three-dimensional space to the shaded hexagon area. Under these constraints, an instance of the

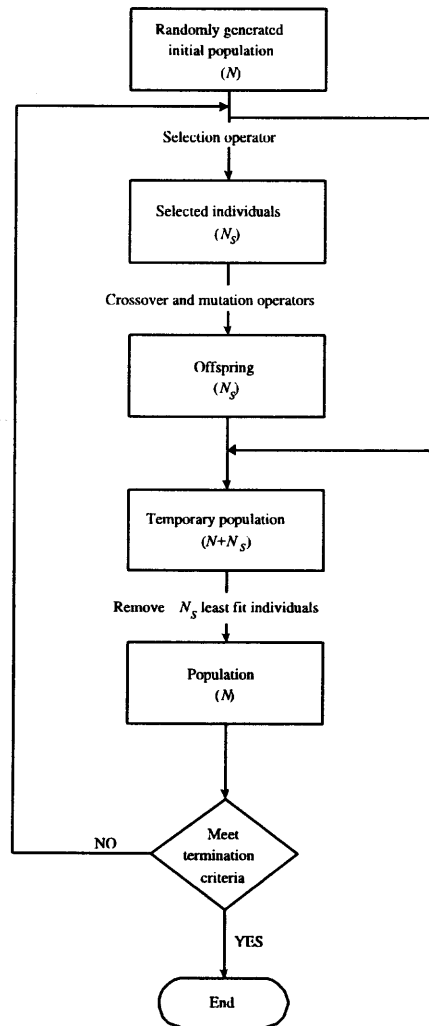


**Figure 4.1** (a) The search space of  $a_1, a_2, a_3$ , and (b) the search space of  $a_4, a_5, a_6$  (or  $a_7, a_8, a_9$ ).

$LC_1C_2$  color space thus corresponds to one point in the shaded triangle area and two points in the shaded hexagon area rather than three arbitrary points in a three-dimensional space when no constraint is applied.

#### 4.2.2 Evolutionary Computation for the Optimal $LC_1C_2$

The goal of Genetic Algorithms (GAs) is to seek the optimal color space transformation that leads to an effective image representation for face recognition. In particular, a Genitor [107] reproduction scheme is adopted for the evolutionary task. Figure 4.2 illustrates an overview of the GA procedure. For each generation, a fixed number of individuals ( $N_s$ ) are selected by the selection operator from a population of size  $N$  to reproduce offspring through the crossover and the mutation operators. The newly generated offspring are then placed back to the current population to form a temporary population of size  $N + N_s$ , from which the  $N_s$  least fit individuals are removed to return the population to its original size. The following subsections detail the chromosome representation, the fitness score derivation, and the genetic operators.



**Figure 4.2** An overview of the GA procedure. For each generation, a fixed number of individuals ( $N_s$ ) are selected by the selection operator from a population of size  $N$  to reproduce offspring through the crossover and the mutation operators. The newly generated offspring are then placed back to the current population to form a temporary population of size  $N + N_s$ , from which the  $N_s$  least fit individuals are removed to return the population to its original size.

**Chromosome Representation** The  $LC_1C_2$  color space, as discussed in Subsection 4.2.1, is determined by the transformation matrix in Equation 4.1. Due to the constraints specified in Equation 4.2, it is appropriate to use six real numbers to represent a chromosome. Let  $\mathbf{c} \in \mathbb{R}^6$  be a chromosome of length 6, the transformation from the  $RGB$  color space to the  $LC_1C_2$  color space can be expressed as [87]:

$$\begin{bmatrix} L \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} \mathbf{c}(1) & \mathbf{c}(2) & 1 - \mathbf{c}(1) - \mathbf{c}(2) \\ \mathbf{c}(3) & \mathbf{c}(4) & 0 - \mathbf{c}(3) - \mathbf{c}(4) \\ \mathbf{c}(5) & \mathbf{c}(6) & 0 - \mathbf{c}(5) - \mathbf{c}(6) \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (4.3)$$

where  $\mathbf{c}(i)$ ,  $i = \{1, 2, \dots, 6\}$ , are known as *alleles* in the genetic terminology, and the values of  $\mathbf{c}(i)$  are subject to the following constraints [87]:

$$\begin{cases} 0 < \mathbf{c}(1), & \mathbf{c}(2), & 1 - \mathbf{c}(1) - \mathbf{c}(2) < 1 \\ -1 < \mathbf{c}(3), & \mathbf{c}(4), & 0 - \mathbf{c}(3) - \mathbf{c}(4) < 1 \\ -1 < \mathbf{c}(5), & \mathbf{c}(6), & 0 - \mathbf{c}(5) - \mathbf{c}(6) < 1 \end{cases} \quad (4.4)$$

**Fitness Scores** The derivation of fitness scores consists of four steps. First, given a chromosome, the input  $RGB$  color images are transformed into the  $LC_1C_2$  color images as defined by Equation 4.3. Second, the  $L$ , the  $C_1$ , and the  $C_2$  component images are histogram equalized, normalized to zero mean and unit variance, and concatenated by rows (or by columns) to form a color feature vector. The normalization procedure is to prevent the intensity values of one component image from unduly dominating the intensity values of other component images in the concatenated vector. Third, the color feature vectors apply the PCA method to derive rank-one face recognition rate as a *raw score*. Finally, a sharing function modulates the raw score by an amount equal to the number of similar individuals in the population to define the *fitness score*. The idea of sharing was first introduced by Holland [39] and later improved by Goldberg [29]. The essence of the sharing technique is to reduce the effect of genetic drift resulting from the selection operator. It

maintains population diversity and prevents GA from being trapped in local optima of the search space [82]. Let  $g_i$  be a raw score of an individual  $i$ , the fitness score  $f_i$  is defined as follow [28]:

$$f_i = \frac{g_i}{\sum_{j=1}^N Sh(d_{ij})} \quad (4.5)$$

where  $N$  is the population size,  $d_{ij}$  denotes a distance function representing the distance between two individuals  $i$  and  $j$ , and  $Sh(\cdot)$  is a sharing function. In particular, an  $L_1$  distance measure is adopted, and the sharing function is defined as [28]:

$$Sh(d_{ij}) = \begin{cases} 1 - \left(\frac{d_{ij}}{\sigma}\right)^\alpha, & \text{if } d_{ij} < \sigma \\ 0, & \text{otherwise} \end{cases} \quad (4.6)$$

where  $\sigma$  is a constant equal to the maximum distance between  $i$  and  $j$ , and  $\alpha$  is set to be 1.

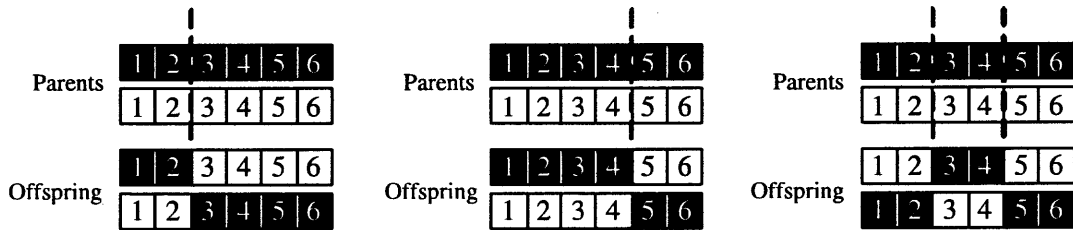
**Genetic Operators** The reproduction of GA is governed by three genetic operators: selection, crossover, and mutation. Specifically, during the course of reproducing two offspring, two parents are first drawn from the population by the selection operator. Then a crossover operator recombines the parents to produce two temporary offspring. Finally, a mutation operator is applied onto each temporary offspring to generate two final offspring. In particular, a stochastic remainder sampling (SRS) selector, a component-wise crossover operator, and a random number mutation operator are deployed.

#### **Genetic Operator-I: Stochastic Remainder Sampling**

The procedure of the SRS selector starts with calculating the probability of selection,  $P_i$ , for each individual  $i$  [28]:

$$P_i = \frac{f_i}{\sum_{j=1}^N f_j} \quad (4.7)$$





**Figure 4.3** The component-wise crossover operator. Two chromosomes of length six are capable of producing three different kinds of offspring. The dashed lines indicate possible crossover points.

where  $f_i$  is the fitness score as defined in Equation 4.5, and  $N$  is the population size. Then the expected individual count  $e_i$  is calculated for each individual  $i$  [28]:

$$e_i = P_i \cdot N \quad (4.8)$$

The value  $e_i$  indicates the number of copies that the individual  $i$  is expected in the next generation. The SRS selector thus allocates the number of individuals according to the integral part of  $e_i$  and then uses the fractional part of  $e_i$  to calculate weights in a roulette wheel selection to fill the remaining population slots.

### Genetic Operator-II: Component-Wise Crossover

The component-wise crossover operator is defined such that the crossover points are always incident at the position of multiples of two, i.e., two alleles associated to the same luminance / chrominance component are never broken up. The reason of such arrangement is twofold: First, the constraints listed in Equation 4.4 will never be violated by the crossover operator. Second, a good color component is preserved. Because it takes two alleles to define one component, breaking up two alleles defining a particular component is essentially deleting that component from the population. Figure 4.3 illustrates the idea of

```

    if  $\mathbf{c}(i)$  is to be mutated
    [
        if  $i = 1$ 
             $\mathbf{c}(i) \leftarrow \text{rand}(0, 1 - \mathbf{c}(2))$ 
        end
        if  $i = 2$ 
             $\mathbf{c}(i) \leftarrow \text{rand}(0, 1 - \mathbf{c}(1))$ 
        end
        if  $i = 3$  or  $i = 5$ 
            if  $\mathbf{c}(i + 1) > 0$ 
                 $\mathbf{c}(i) \leftarrow \text{rand}(-1.0, 1.0 - \mathbf{c}(i + 1))$ 
            else
                 $\mathbf{c}(i) \leftarrow \text{rand}(-1.0 - \mathbf{c}(i + 1), 1.0)$ 
            end
        end
        if  $i = 4$  or  $i = 6$ 
            if  $\mathbf{c}(i - 1) > 0$ 
                 $\mathbf{c}(i) \leftarrow \text{rand}(-1.0, 1.0 - \mathbf{c}(i - 1))$ 
            else
                 $\mathbf{c}(i) \leftarrow \text{rand}(-1.0 - \mathbf{c}(i - 1), 1.0)$ 
            end
        end
    ]
end

```

**Figure 4.4** The mutation operator that guarantees the satisfaction of constraints in Equation 4.4.

component-wise crossover where the dashed lines indicate possible crossover points. Note that two chromosomes of length six are capable of producing three different kinds of offspring.

### Genetic Operator-III: Random Number Mutation

The mutation operator introduces new alleles into the population. As real number chromosomes are implemented in the experiments, this operator basically changes an allele from one real number to another. However, an arbitrary change of an allele might violate the constraints specified in Equation 4.4. To prevent this from happening, the mutation operator considers together the associated alleles. Figure 4.4 shows the definition of the mutation operator, which guarantees the satisfaction of constraints in Equation 4.4. In particular,  $c(i)$  represents the  $i^{th}$  allele on a chromosome  $c$ , and  $rand(p, q)$  is a random number generator that generates a real number in the range of  $p$  and  $q$  (exclusive).

#### 4.2.3 Color Fisher Features

After the transformation from  $RGB$  to  $LC_1C_2$ , the Fisher linear discriminant (FLD) [96], [22] method is applied to further extracts discriminating features, termed color Fisher features (CFFs). FLD is a popular discriminant criterion for face recognition [3], [55]. Let  $\mathbf{x}$  be a random vector representing a face from the training set,  $\omega_1, \omega_2, \dots, \omega_L$  be class labels, and  $N_1, N_2, \dots, N_L$  represent the number of images within each class. Let  $\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_L$  and  $\mathbf{m}$  be the means of the classes and the grand mean. The within- and between- class scatter matrices  $\Sigma_w$  and  $\Sigma_b$  are defined as follows [25]:

$$\Sigma_w = \sum_{i=1}^L P(\omega_i) \mathcal{E} \{ (\mathbf{x} - \mathbf{m}_i) (\mathbf{x} - \mathbf{m}_i)^t | \omega_i \} \quad (4.9)$$

$$\Sigma_b = \sum_{i=1}^L P(\omega_i) (\mathbf{m}_i - \mathbf{m}) (\mathbf{m}_i - \mathbf{m})^t \quad (4.10)$$

where  $P(\cdot)$  is *a priori* probability and  $\mathcal{E}(\cdot)$  the expectation operator. The FLD procedure maximizes the ratio,  $|\Psi^t \Sigma_b \Psi| / |\Psi^t \Sigma_w \Psi|$ , where the optimization matrix,  $\Psi$ , consisting of the eigenvectors of the matrix  $\Sigma_w^{-1} \Sigma_b$  can be derived by [25]:

$$\Sigma_w^{-1} \Sigma_b \Psi = \Psi \Delta \quad (4.11)$$

where  $\Psi$ ,  $\Delta$  are the eigenvector and eigenvalue matrices of  $\Sigma_w^{-1} \Sigma_b$ .

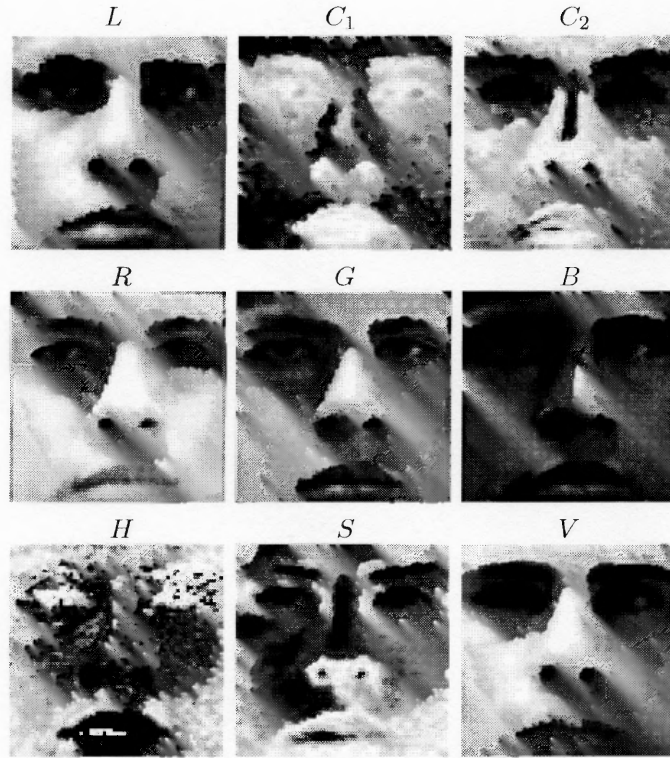
After the training, the target and the query sets (or the gallery and the probe sets in ver1.0) are applied to assess the face recognition performance using a nearest neighbor classifier. Let  $\mathcal{Q}$  be a query feature set of size  $|\mathcal{Q}|$  obtained by projecting query pattern vectors onto the eigenvectors derived from the training process (i.e.,  $\Psi$  in Equation 4.11). The target feature set  $\mathcal{T}$  of size  $|\mathcal{T}|$  is produced in a similar manner. Then a similarity matrix  $\mathbf{S}$  of size  $|\mathcal{Q}| \times |\mathcal{T}|$  is derived by computing a similarity score between every query-target pair. Let  $\mathbf{q}_i$  represents a query feature vector in  $\mathcal{Q}$  and  $\mathbf{t}_j$  a target feature vector in  $\mathcal{T}$ , when the PCA method is used, the similarity score  $\mathbf{S}(i, j)$  between  $\mathbf{q}_i$  and  $\mathbf{t}_j$  is derived using a cosine similarity measure [71]:

$$\mathbf{S}(i, j) = \frac{-\mathbf{q}_i^t \mathbf{t}_j}{\|\mathbf{q}_i\| \|\mathbf{t}_j\|} \quad (4.12)$$

Finally, the similarity matrix  $\mathbf{S}$  is analyzed by BEE to determine the recognition performance as detailed in Section 3.4.1.

### 4.3 Experiments

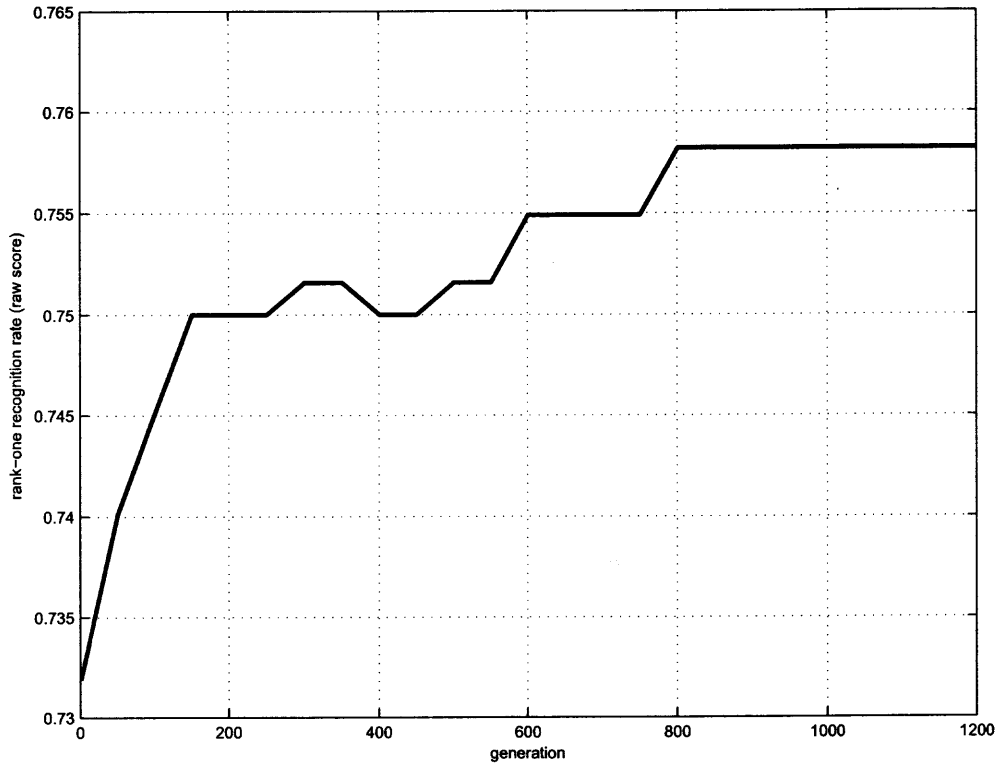
The experiments are organized into three major sets. The first set of experiments applies the 1,126 FRGC ver1.0 images to evolve the optimal color space transformation from the *RGB* color space to the *LC<sub>1</sub>C<sub>2</sub>* color space. The CMC performance of the FRGC ver1.0 Experiment 4 of both PCA and FLD is also reported in this set of experiments. After determining the transformation matrix, the second set of experiments applies the 30,702



**Figure 4.5** Three component images in the  $LC_1C_2$ , the  $HSV$ , and the  $RGB$  color spaces, respectively. The component images have a spatial resolution  $64 \times 64$ .

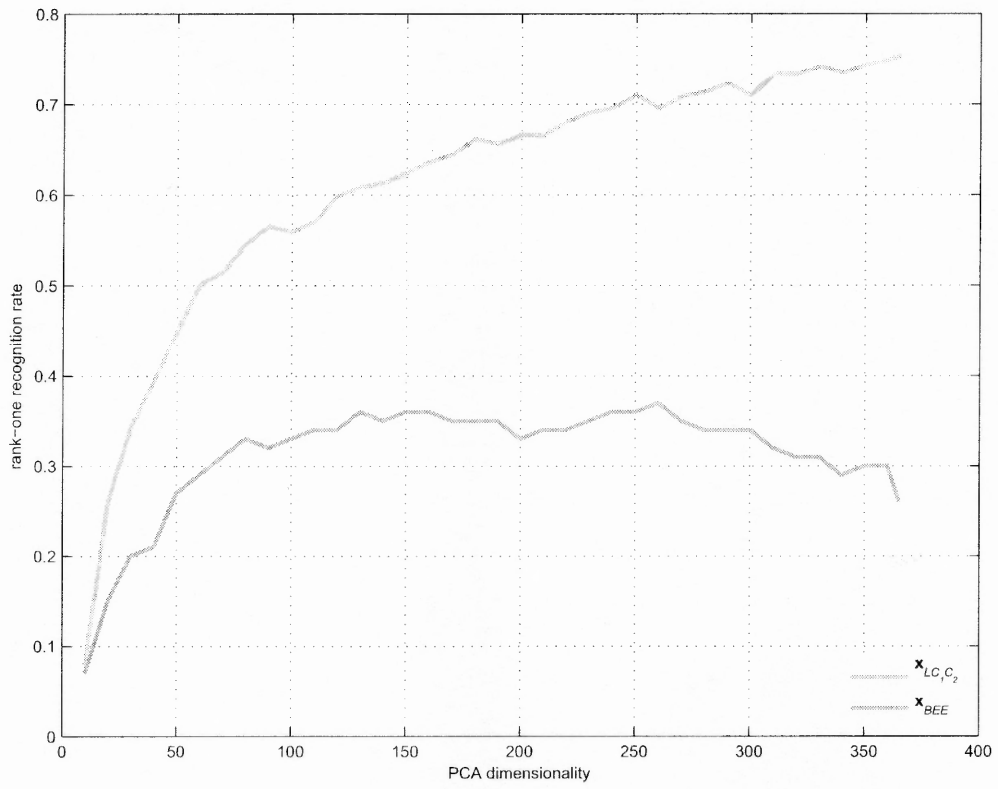
FRGC ver2.0 images in the  $LC_1C_2$  color space using the PCA method. Note that the face verification performance in two conventional color spaces,  $RGB$  and  $HSV$ , are also included for comparison. Finally, the last set of experiments applies the same 30,702 FRGC ver2.0 images to the FLD method in the  $LC_1C_2$ , the  $RGB$ , and the  $HSV$  color spaces. The second and the third sets of experiments report the ROC performance of the FRGC ver2.0 Experiment 4.

The data preparation, as detailed in Section 3.4.2, normalizes the face images to a spatial resolution of  $64 \times 64$  to extract the facial region that contains only face. Figure 4.5 shows example color component images in the  $LC_1C_2$ , the  $RGB$ , and the  $HSV$  color spaces. The displayed images have a smaller spatial resolution than the BEE intensity images (see Figure 3.16(b))



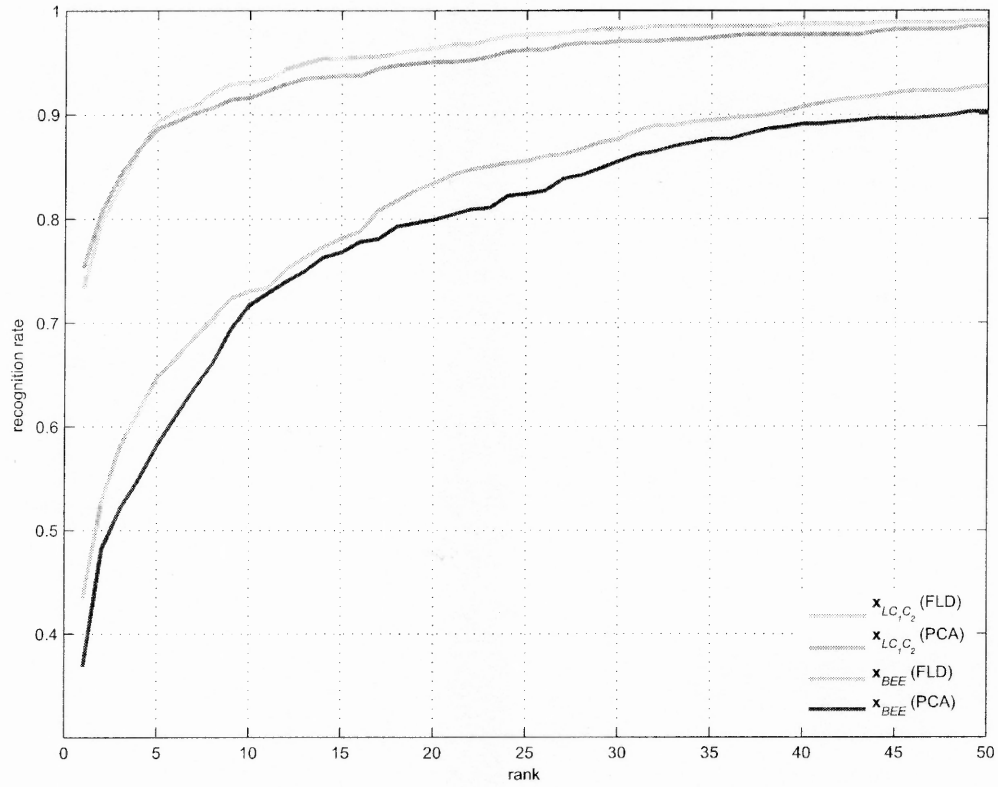
**Figure 4.6** The course of evolution using the FRGC ver1.0 dataset. The horizontal axis represents the generation, and the vertical axis denotes the rank-one face recognition accuracy derived by the best performing chromosome in each generation.

After the preprocessing, the pattern vector is defined by concatenating the three component images in a color space. Specifically, each component image is first histogram equalized to improve the image quality by increasing its dynamic range [30]. Then it is normalized to zero mean and unit variance to reduce image variations, which in turn prevents one component image from unduly dominating the others during the concatenation. Finally, the three normalized component images are concatenated to form a pattern vector. In the successive discussion,  $\mathbf{x}_{BEE} \in \mathbb{R}^{19500}$  represents the pattern vector defined by the BEE intensity image, and  $\mathbf{x}_{LC_1C_2}$ ,  $\mathbf{x}_{RGB}$ ,  $\mathbf{x}_{HSV} \in \mathbb{R}^{12288}$  represent the pattern vectors in the  $LC_1C_2$ ,  $RGB$ ,  $HSV$  color spaces, respectively.



**Figure 4.7** The rank-one face recognition rates of the PCA method using the  $x_{LC_1C_2}$  and the  $x_{BEE}$  representations.

Following the evolutionary procedure described in Subsection 4.2.2, the first set of experiments evolves the transformation matrix from  $RGB$  to  $LC_1C_2$  using the 1,126 images from FRGC ver1.0. Figure 4.6 shows the course of evolution. The horizontal axis represents generation, and the vertical axis denotes the rank-one face recognition accuracy derived by the best performing chromosome in each generation. The termination criterion has been set to be 1,200 generations. Note that during the evolution, the best performing chromosome corresponds to the one with the highest fitness score, but not necessarily to the one with the best recognition accuracy. As the rank-one recognition rates serve as raw scores in GA whose values are modulated by the sharing function (see Equation 4.6), the curve shown in Figure 4.6 does not monotonically increase. Note that when the GA con-



**Figure 4.8** The FRGC ver1.0 CMC performance of the  $x_{LC_1C_2}$  and the  $x_{BEE}$  representations. The PCA performance using the  $x_{BEE}$  representation is the BEE baseline performance.

verges, the best performing chromosome corresponds to both the highest fitness score and the best recognition accuracy.

Figure 4.7 compares the rank-one face recognition rates of the PCA method using the  $x_{LC_1C_2}$  and the  $x_{BEE}$  representations. As the number of eigenvectors derived by PCA using 366 training images is 365 (i.e.,  $366 - 1$ ), the total number of features derived by projecting a pattern vector onto these eigenvectors is 365. Figure 4.7 shows that the face recognition performance peaks when 365 and 143 PCA features are used for the  $x_{LC_1C_2}$  and the  $x_{BEE}$ , respectively. These results suggest that different PCA dimensionality should be used in order to conduct a fair comparison between different representations.



**Table 4.1** The Rank-One Face Recognition Rates of the FRGC ver1.0 CMC Performance

	PCA	FLD
$\mathbf{x}_{LC_1C_2}$	75.33%	73.36%
$\mathbf{x}_{BEE}$	36.84% <sup>†</sup>	43.42%

<sup>†</sup> the BEE baseline performance

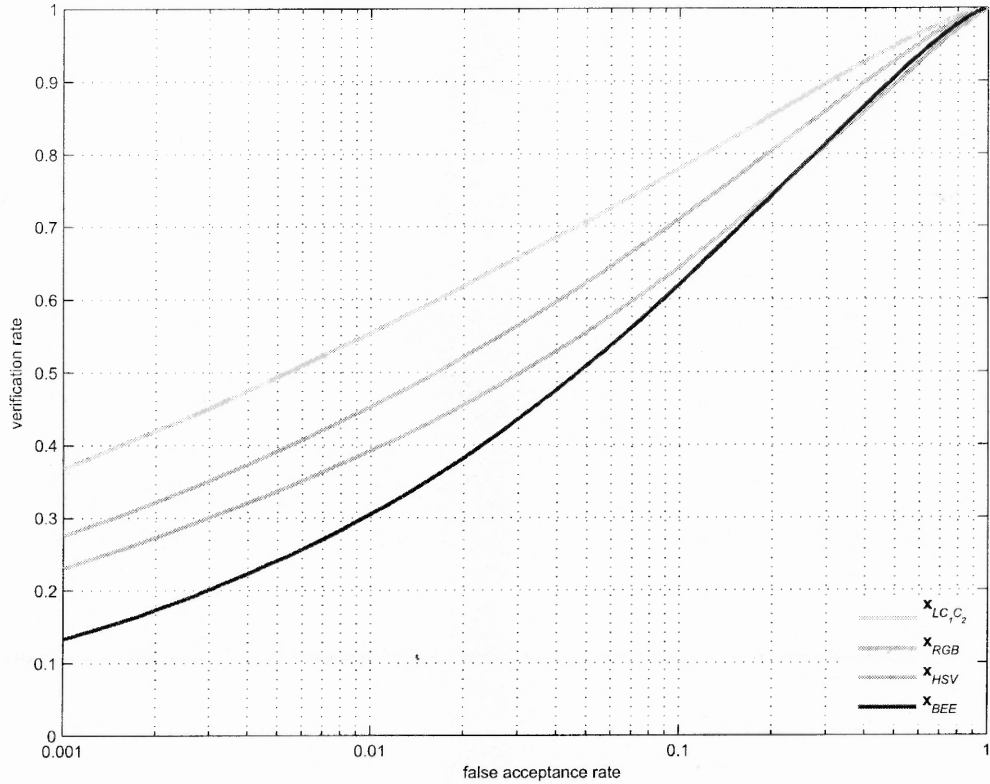
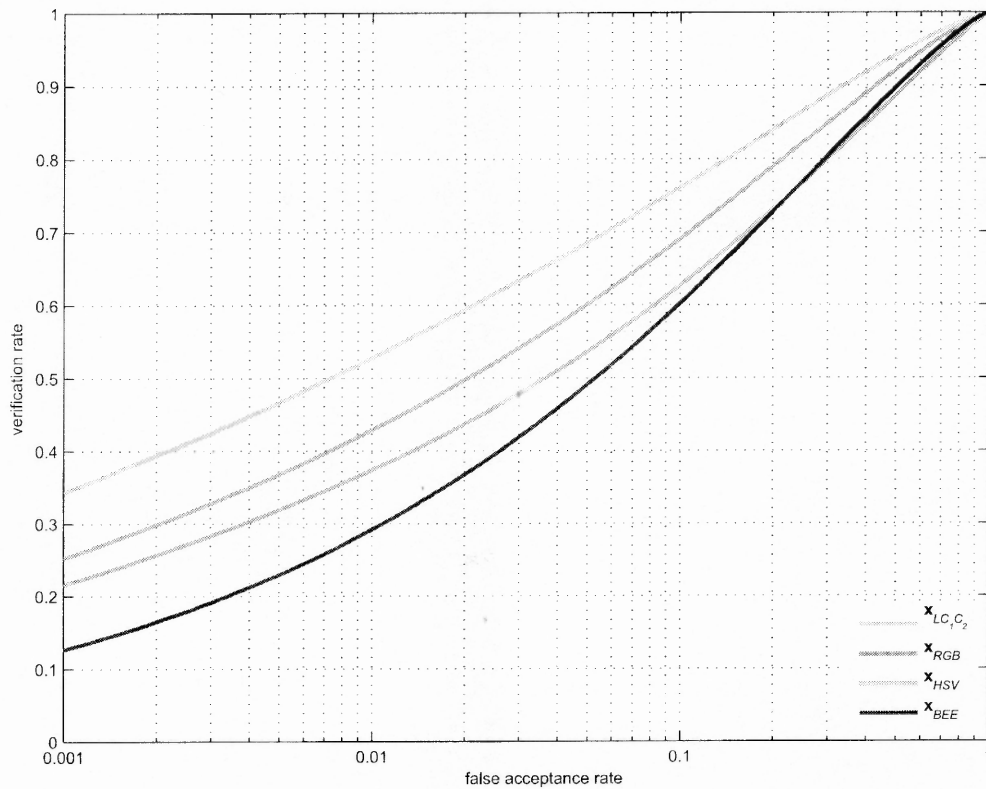
**Figure 4.9** The ROC I performance of FRGC ver2.0 Experiment 4 using the PCA method. The curve corresponding to the  $\mathbf{x}_{BEE}$  representation is the BEE baseline performance.

Figure 4.8 shows the FRGC ver1.0 CMC performance of the  $\mathbf{x}_{LC_1C_2}$  and the  $\mathbf{x}_{BEE}$  representations. The PCA performance using the  $\mathbf{x}_{BEE}$  representation is the BEE baseline performance. The horizontal axis denotes the rank  $r$ , and the vertical axis indicates the probability that the correct match is included in the top  $r$  matches. Figure 4.8 shows that the  $LC_1C_2$  color features improve the BEE baseline performance by large margins. In particular, when

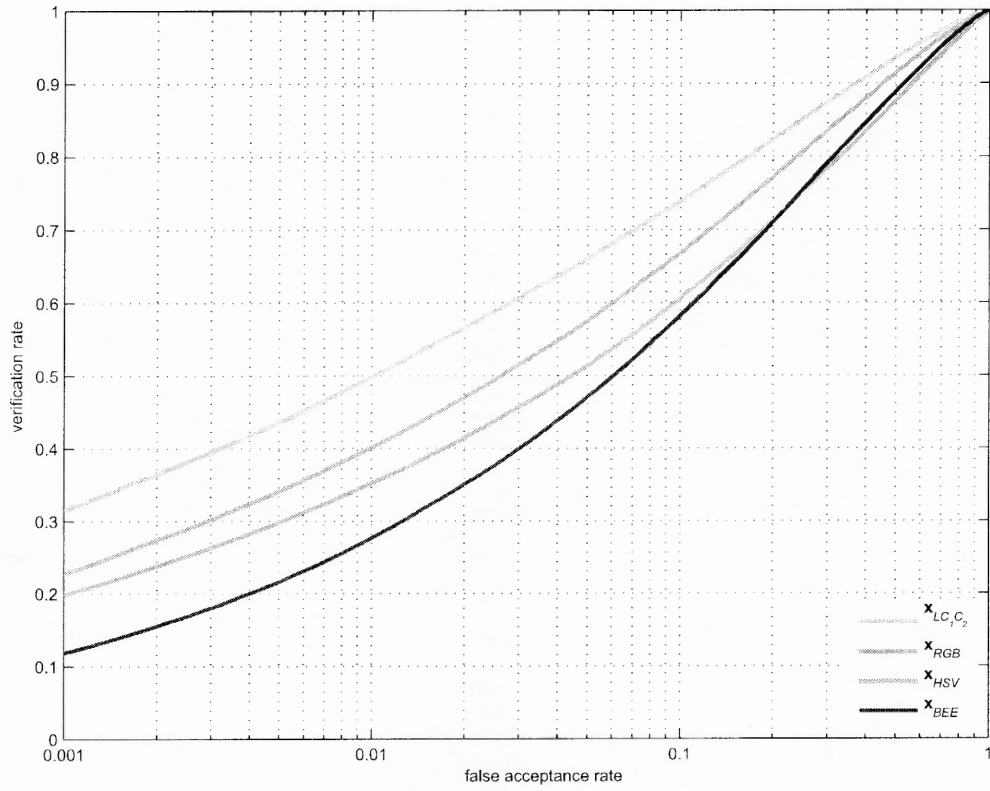


**Figure 4.10** The ROC II performance of FRGC ver2.0 Experiment 4 using the PCA method. The curve corresponding to the  $x_{BEE}$  representation is the BEE baseline performance.

**Table 4.2** Face Verification Rates (at 0.1% FAR) of FRGC ver2.0 Experiment 4 Using PCA

	ROCI	ROCII	ROC III
$x_{LC,C_2}$	36.79%	34.31%	31.49%
$x_{HSV}$	27.53%	25.21%	22.76%
$x_{RGB}$	23.08%	21.55%	19.81%
$x_{BEE}$	13.36% <sup>†</sup>	12.67% <sup>†</sup>	11.87% <sup>†</sup>

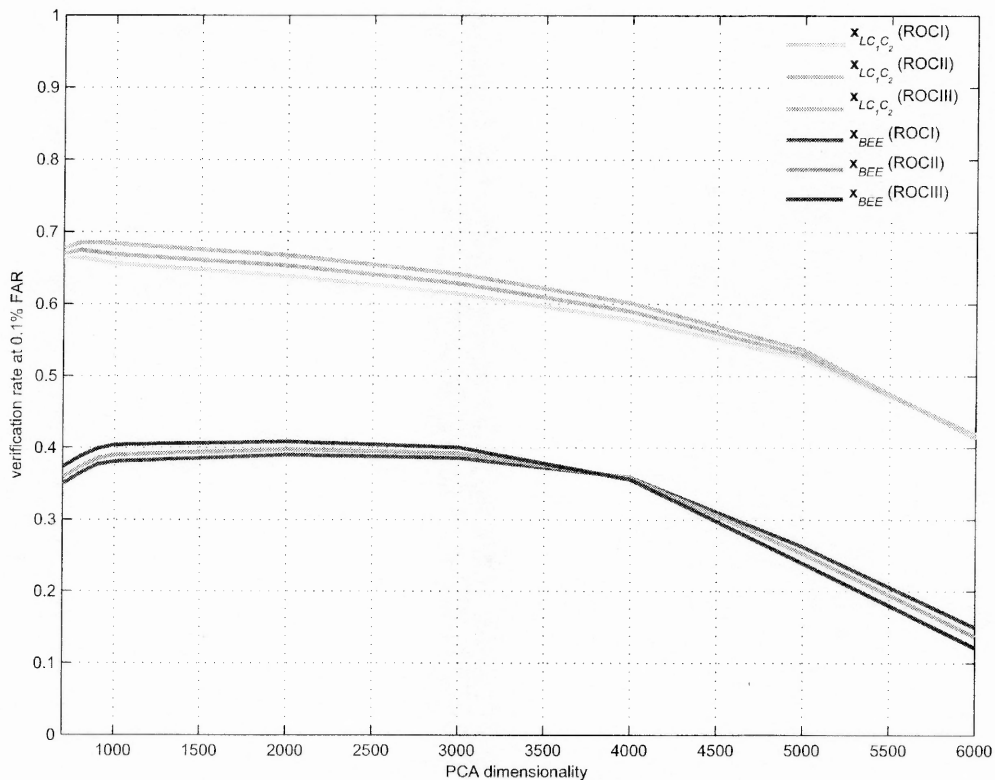
<sup>†</sup> the BEE baseline performance



**Figure 4.11** The ROC III performance of FRGC ver2.0 Experiment 4 using the PCA method. The curve corresponding to the  $\mathbf{x}_{BEE}$  representation is the BEE baseline performance.

the PCA method is used, the rank-one recognition rate is improved by 38% (from 37% to 75%), and when the FLD method is applied, the rank-one rate is improved by 30% (from 43% to 73%). Note that in the FRGC ver1.0, the number of training images per subject is limited. As a result, FLD does not improve much upon the PCA method. The rank-one face recognition rates of the FRGC ver1.0 CMC performance are summarized in Table 4.1.

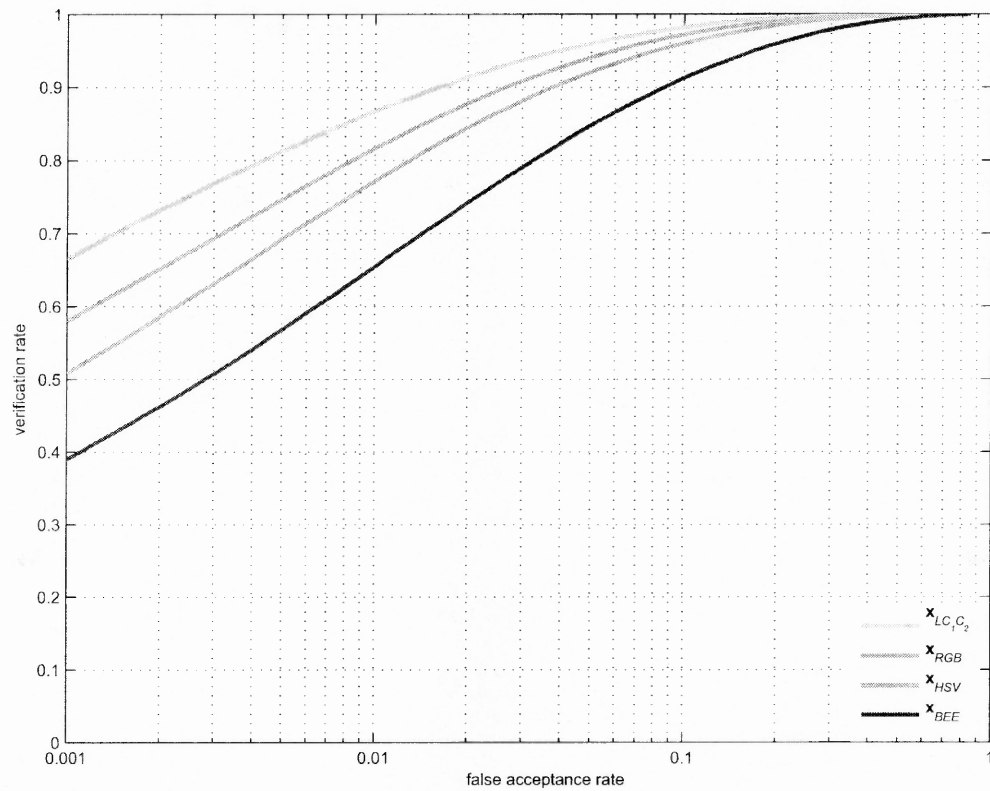
The second set of experiments assesses the face verification performance of the PCA method using the 30,702 images from the FRGC ver2.0. As there are 6,660 training images available in the training set (see Section 3.4.1), PCA derives at most 6,659 (i.e.,  $6660 - 1$ ) features, and the maximum dimensionality of the PCA space is thus 6,659. Note that in this set of experiments, 1,000 PCA features are empirically chosen for face verification



**Figure 4.12** The FLD performance using the  $x_{LC_1C_2}$  and the  $x_{BEE}$  representations in different PCA spaces.

(i.e., using 1,000 PCA features to derive the similarity matrix in Equation 3.21). Figures 4.9, 4.10, and 4.11 show the ROC I, the ROC II, and the ROC III performance of the FRGC ver2.0 Experiment 4. The curves corresponding to the  $x_{BEE}$  representation are the BEE baseline performance. The horizontal axis indicates the false acceptance rate (FAR), and the vertical axis is the verification rate. These three figures indicate that the  $x_{LC_1C_2}$  representation achieves the best verification performance, followed in order by  $x_{HSV}$ ,  $x_{RGB}$ , and  $x_{BEE}$ . The face verification rates at 0.1% FAR of these four representations are summarized in Table 4.2.

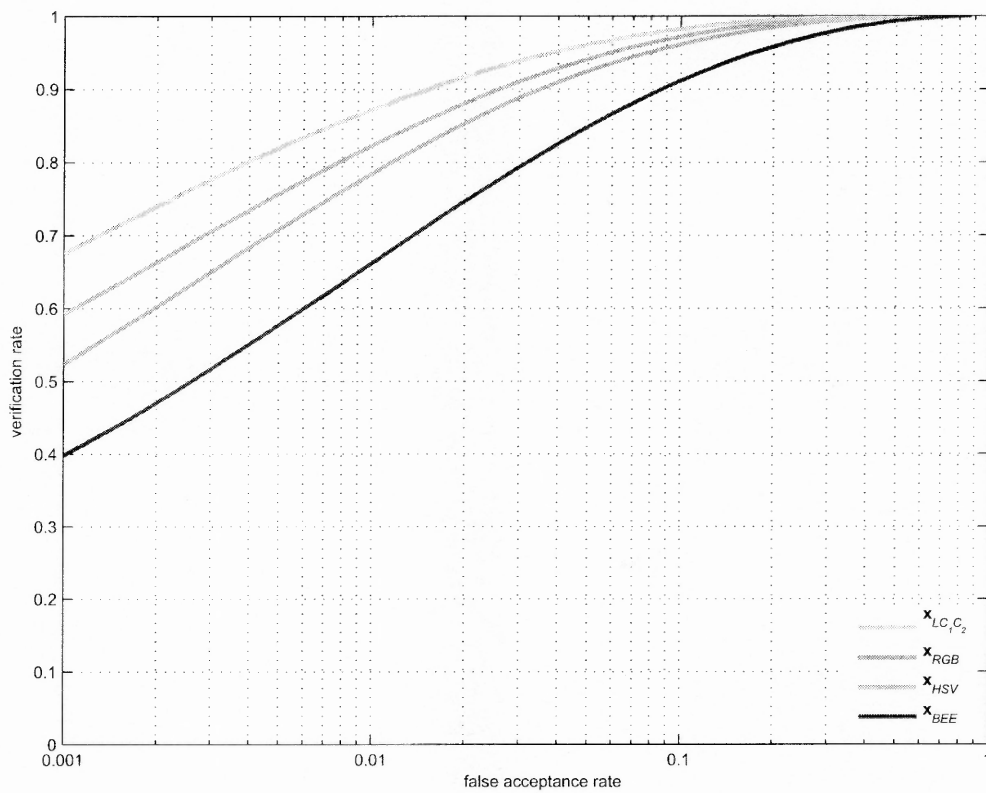
The next set of experiments applies the  $x_{LC_1C_2}$  and the  $x_{BEE}$  representations to the FLD method. Note that the dimensionality of  $x_{LC_1C_2}$  is 12,288 while there are only 6,660 training images available. To avoid the within-class scatter matrix  $\Sigma_w$  (see Equation 4.9) from being



**Figure 4.13** The ROC I performance of FRGC ver2.0 Experiment 4 using the FLD method.

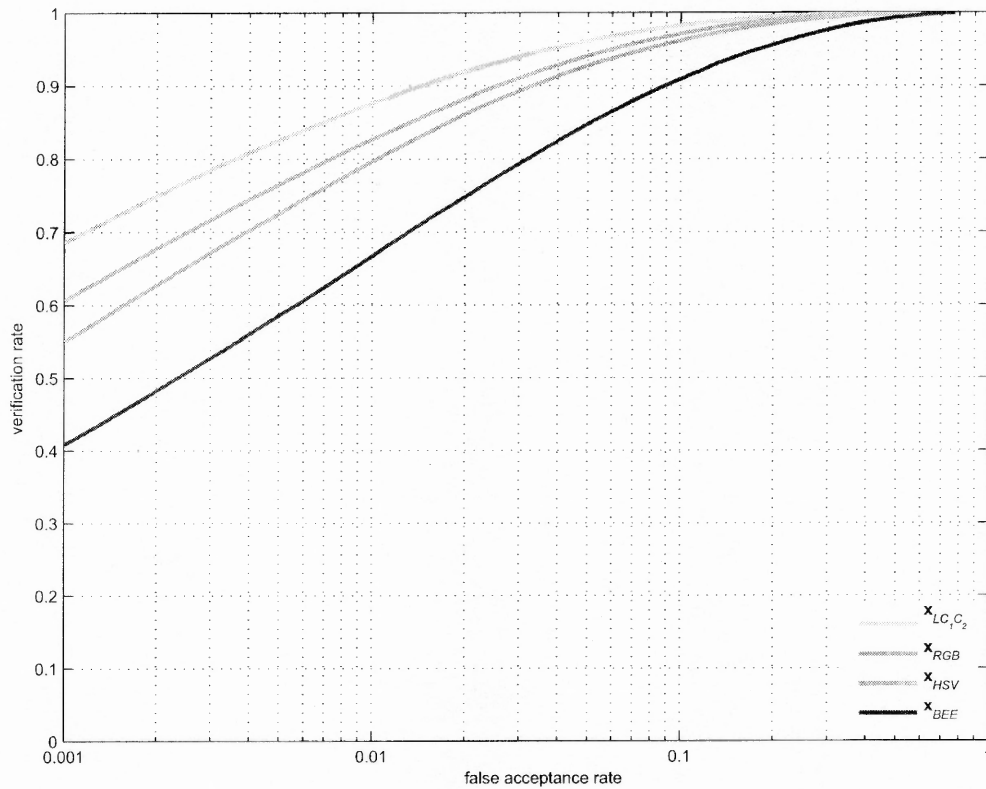
**Table 4.3** Face Verification Rates (at 0.1% FAR) of FRGC ver2.0 Experiment 4 Using FLD

	ROCI	ROCII	ROC III	PCA Space	FLD Space
$x_{LC_1C_2}$	66.45%	67.50%	68.53%	800	221
$x_{HSV}$	58.01%	59.21%	60.69%	800	221
$x_{RGB}$	50.82%	52.34%	55.01%	800	221
$x_{BEE}$	39.06%	39.79%	40.89%	2,000	221



**Figure 4.14** The ROC II performance of FRGC ver2.0 Experiment 4 using the FLD method.

singular, a PCA procedure is applied prior to the FLD method to reduce the dimensionality of the input space [3]. Note that the trailing eigenvalues from the PCA procedure tend to capture noise, and cause the whitening step of FLD to fit for misleading variations [55]. This in turn deteriorates the generalization performance of the FLD method. Therefore, the FLD method is implemented using different number of PCA features to empirically determine the optimal PCA dimensionality. Figure 4.12 shows the FLD performance using the  $\mathbf{x}_{LC_1C_2}$  and the  $\mathbf{x}_{BEE}$  representations in different PCA spaces. The horizontal axis represents the PCA dimensionality in which FLD performs, and the vertical axis indicates the verification rate at 0.1% FAR. Note that since there are 222 subjects in the FRGC ver2.0 training set, the FLD method derives at most 221 (i.e.,  $222 - 1$ ) FLD features, and the maximum



**Figure 4.15** The ROC III performance of FRGC ver2.0 Experiment 4 using the FLD method.

dimensionality of the FLD space is 221. The curves displayed in Figure 4.12 are derived using 221 FLD features.

The results in Figure 4.12 show that the dimensionality of the PCA space noticeably affects the verification performance. Specifically, the  $x_{LC_1C_2}$  and the  $x_{BEE}$  representations achieve the best performance in an 800 and a 2,000 PCA spaces, respectively. Note that the verification performance using different number of FLD features is also investigated. The experimental results, however, show that using more FLD features generally leads to a higher verification accuracy. Therefore, when the  $x_{LC_1C_2}$  representation is used, FLD is performed in an 800 PCA space, and 221 FLD features are used for verification (i.e., using 221 FLD features to derive the similarity matrix in Equation 4.12).

Figures 4.13, 4.14, and 4.15 show the ROC I, the ROC II, and the ROC III performance of FRGC ver2.0 Experiment 4 using the the FLD method, i.e., applying the CFF features. The curves corresponding to the  $\mathbf{x}_{LC_1C_2}$ , the  $\mathbf{x}_{HSV}$ , and the  $\mathbf{x}_{RGB}$  representations are derived in an 800 PCA space using 221 FLD features. The curves corresponding to the  $\mathbf{x}_{BEE}$  representation are obtained in a 2,000 PCA space with 221 FLD features. Again, the  $\mathbf{x}_{LC_1C_2}$  representation achieves the best verification performance, followed in order by  $\mathbf{x}_{HSV}$ ,  $\mathbf{x}_{RGB}$ , and  $\mathbf{x}_{BEE}$ . The face verification rates at 0.1% FAR using these four representations are summarized in Table 4.3.

#### 4.4 Conclusion

This chapter presents a novel color feature extraction method using evolutionary computation. Experimental results using 1,126 images from the FRGC ver1.0 and 30,702 images from the FRGC ver2.0 show that the new color feature extraction method improves the face recognition performance for both the PCA and the FLD methods. In particular, for FRGC ver1.0 Experiment 4, the evolved color features improve the rank-one face recognition rate from 37% to 75% using the PCA method and achieve 73% using the FLD method. For FRGC ver2.0 Experiment 4, the evolved color features improve the face verification rate at 0.1% FAR from 12% to 32% using the PCA method and achieve 69% using the FLD method.



## CHAPTER 5

### 2D AND 3D FACE RECOGNITION USING CONVOLUTION FILTERS

In Chapters 3 and 4, numerous feature extraction methods utilizing color information are studied. Face recognition accuracy in general or the Face Recognition Grand Challenge (FRGC) Experiment 4 baseline performance in particular has been noticeably improved, e.g., from 12% to 69% using the  $LC_1C_2$  color configuration and FLD. These research, however, focuses mainly on two-dimensional (2D) imaging modality and color representations while leaving three-dimensional (3D) imaging modality and other feature extraction techniques undiscovered. To explore these new research areas, this chapter presents an approach to apply convolution filters to extract discriminating features from both 2D and 3D modalities for face recognition.

The major challenges faced by 2D face recognition come from the illumination, pose and expression variations that introduce statistical differences to facial images. To overcome these difficulties, recent developments [36], [61] have been focused on exploring new facial modalities, such as 3D range data, that eliminate these factors not related to subjects' identities. Although 3D face recognition has its own drawbacks, such as high cost and decreased ease-of-use of data acquisition equipments, integrating 2D and 3D face recognition systems can have the potential to complement each other and lead to a better biometric authentication system.

The 2D and 3D imaging modalities adopted by this research are both 2D images, which significantly reduce the high computational complexity generally required by 3D-based face recognition systems. Although both modalities are represented by 2D images, they display very distinct properties. In particular, 2D modality displays intensity texture, while 3D modality shows curvature shape, of which pixels in the neighborhood have much smaller variation than intensity texture. Because of this fundamental difference, specific

feature extraction methods may be required to derive discriminating features for 2D and 3D modalities, respectively.

To investigate the possibility of using convolution filters for feature extraction, four filters are defined based on wavelet functions: Gaussian derivative [18], [58], [35], Morlet [18], complex Morlet [97], and complex frequency B-spline [97]. The filters are then applied to convolve with 2D or 3D modality to output convolution features. Finally, the convolution features are used to compute two separate similarity measures for 2D and 3D modalities, which are then linearly fused to calculate a final similarity measure.

The feasibility of the proposed method is demonstrated using the Face Recognition Grand Challenge (FRGC) ver2.0 Experiment 3 containing 4,950 2D color images (943 controlled and 4,007 uncontrolled) and 4,950 3D recordings. The experimental results show that the Gaussian derivative convolution filter extracts the most discriminating features from 3D modality among the four filters, and the complex frequency B-spline convolution filter outperforms other filters when 2D modality is applied. Furthermore, the fusion of 2D and 3D similarity measures improves FRGC ver2.0 Experiment 3 baseline performance from 67% to 82% (ROC I), from 61% to 78% (ROC II), and from 54% to 72% (ROC III).

The remainder of this chapter is organized as follows. Section 5.1 reviews prior research and current study on 3D face recognition. Section 5.2 reviews four well-defined wavelet functions and defines four convolution filters. Section 5.3 detailed the derivation of convolution features and the fusion of similarity measures. Section 5.4 presents the experimental results on FRGC ver2.0 Experiment 3, and finally, Section 5.5 summarizes the research on 2D and 3D face recognition.

## 5.1 Background

Research in 3D face recognition can be categorized into two major groups: 2D+3D and 3D [13]. The 2D+3D face recognition generally refers to the use of multiple imaging modalities, such as 2D images and 3D range data. Beumier et al. proposed a face recognition system [4] that evaluated “3D central and lateral profiles” and “gray level central and lateral profiles” separately, and then fused them with the Fishers linear discriminant method. Wang et al. [106] used the point signature [16] and the stacked Gabor filter responses [108] to identify 3D and 2D features. Bronstein et al. [7] developed a 3D face recognition algorithm based on geometric invariants [21] that allowed mapping 2D facial texture images onto 3D special geometry. Recently, Tsalakanidou et al. [101] introduced a face authentication system integrating 2D intensity and 3D range data based on a low-cost real-time structured light sensor. This method employed a hidden markov model (HMM) for face authentication.

The 3D face recognition systems, on the other hand, are concerned with approaches that only apply 3D modality. Gordon et al. [31] proposed a curvature-based method for face recognition from 3D data stored in a cylindrical coordinate system. Lao et al. [49] proposed a 3D facial model consisting of a sparse depth map constructed from stereo images using isoluminance lines for the stereo matching. By searching for arcs whose radius are of certain ranges, the candidate irises can be located efficiently. Lee et al. [51] designed a feature extraction method calculating mean and variance of depth from windows around the nose and applying a nearest neighbor classifier for face recognition. Moreno et al. [67] extracted a number of features from 3D data, and found that curvature and line features perform better than area features. Heshner et al. [36] developed an ICA based face recognition algorithm utilizing 3D range data. Medioni et al. [62] applied a normalized cross-correlation algorithm to compute a distance map, whose statistics are then used for similarity measurement.

## 5.2 Convolution Filters

This section first reviews the four wavelet functions: Gaussian derivative [18], [58], [35], Morlet [18], complex Morlet [97], and complex frequency B-spline [97], and then derives the convolution filters based on these wavelet functions.

### 5.2.1 Gaussian Derivative Filter

The Gaussian derivative is defined by taking the  $p^{th}$  derivative of a Gaussian function  $e^{-x^2/2}$ , where  $p$  is a positive integer. The second derivative of Gaussian is thus defined by [18], [58], [35]:

$$\Psi(x) = \left(2/\sqrt{3}\right) \pi^{-1/4} (1 - x^2) e^{-x^2/2} \quad (5.1)$$

where  $-(2/\sqrt{3}) \pi^{-1/4}$  is a normalization constant [18]. Note that the standard deviation of Gaussian in Equation 5.1 is assumed to be one. To perform a general derivation with the standard deviation being  $\sigma$ , Equation 5.1 can be rewritten as:

$$\Psi(x) = -C_1 \frac{d^2}{dx^2} \left( e^{-x^2/(2\sigma^2)} \right) = C_2 \left[ 1 - \left( \frac{x}{\sigma} \right)^2 \right] e^{-x^2/(2\sigma^2)} \quad (5.2)$$

where  $C_1$  and  $C_2$  are positive normalization constants. Based on this formulation, the Gaussian derivative filter  $\Psi(x, y)$  is defined by:

$$\Psi(\mathbf{v}) = -C_g \nabla^2 e^{-\frac{1}{2} \mathbf{v}^t \boldsymbol{\Sigma}^{-1} \mathbf{v}} \quad (5.3)$$

where  $\mathbf{v} = (x \ y)^t$ , and  $C_g$  is a positive constant,  $\boldsymbol{\Sigma}$  the covariance matrix, and  $\nabla^2$  the Laplacian operator. To simplify this formulation, the covariance matrix  $\boldsymbol{\Sigma}$  is assumed to be diagonal, i.e.,  $x$  and  $y$  are uncorrelated, so the inverse of  $\boldsymbol{\Sigma}$  is:

$$\boldsymbol{\Sigma}^{-1} = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}^{-1} = \begin{bmatrix} \frac{1}{\sigma_x^2} & 0 \\ 0 & \frac{1}{\sigma_y^2} \end{bmatrix} \quad (5.4)$$

By substituting Equation 5.4 to Equation 5.3, the convolution filter  $\Psi$  becomes [30], [60]:

$$\begin{aligned}\Psi(x, y) &= -C_g \nabla^2 e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} \\ &= -C_g \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} \\ &= -C_g \frac{\partial^2}{\partial x^2} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} - C_g \frac{\partial^2}{\partial y^2} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}}\end{aligned}\quad (5.5)$$

The derivatives with respect to  $x$  and  $y$  can be further expressed as [30], [60]:

$$\frac{\partial^2}{\partial x^2} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} = -\frac{1}{\sigma_x^2} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} + \frac{x^2}{\sigma_x^4} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} \quad (5.6)$$

$$\frac{\partial^2}{\partial y^2} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} = -\frac{1}{\sigma_y^2} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} + \frac{y^2}{\sigma_y^4} e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} \quad (5.7)$$

Following Equations 5.6 and 5.7, Equation 5.5 is rewritten as:

$$\Psi(x, y) = C_g \left[ \frac{1}{\sigma_x^2} + \frac{1}{\sigma_y^2} - \left( \frac{x^2}{\sigma_x^4} + \frac{y^2}{\sigma_y^4} \right) \right] e^{-\frac{1}{2} \frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}} \quad (5.8)$$

Note that the filter defined by Equation 5.8 is anisotropic if  $\sigma_x \neq \sigma_y$ . To translate and rotate this filter in a two-dimensional space, one can introduce a translation vector  $\mathbf{b}$  and a rotation matrix  $\mathbf{R}$  to Equation 5.8, such that:

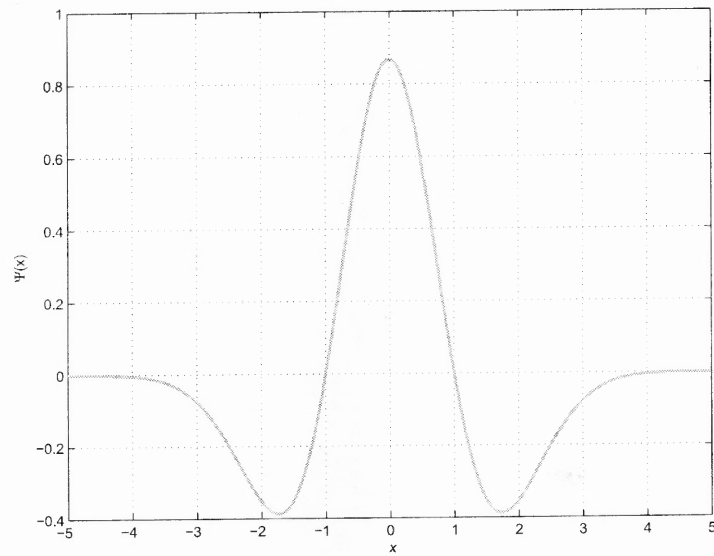
$$\Psi(\mathbf{v}) = C_g \left( \frac{1}{\sigma_x^2} + \frac{1}{\sigma_y^2} - (\mathbf{R}[\mathbf{v} - \mathbf{b}])^t \mathbf{A}^2 (\mathbf{R}[\mathbf{v} - \mathbf{b}]) \right) e^{-\frac{1}{2} ((\mathbf{R}[\mathbf{v} - \mathbf{b}])^t \mathbf{A} (\mathbf{R}[\mathbf{v} - \mathbf{b}]))} \quad (5.9)$$

where  $\mathbf{b}$ ,  $\mathbf{R}$ , and  $\mathbf{A}$  are defined by:

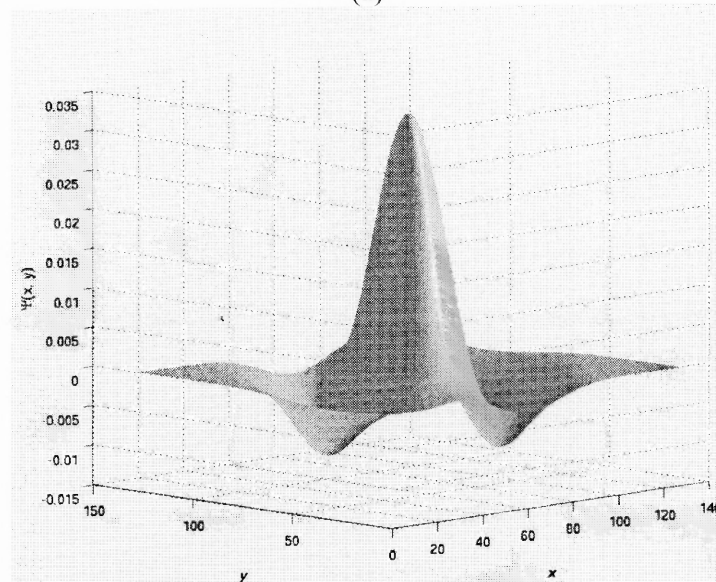
$$\mathbf{b} = \begin{pmatrix} b_x \\ b_y \end{pmatrix}; \quad \mathbf{R} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}; \quad \mathbf{A} = \mathbf{\Sigma}^{-1} = \begin{bmatrix} \frac{1}{\sigma_x^2} & 0 \\ 0 & \frac{1}{\sigma_y^2} \end{bmatrix} \quad (5.10)$$

Finally, let  $\mathbf{u} = (u_1 \ u_2)^t = \mathbf{A}^{1/2} (\mathbf{R}[\mathbf{v} - \mathbf{b}])$ , and Equation 5.9 can be simplified as follows:

$$\Psi(\mathbf{u}) = C_g \left( \frac{1}{\sigma_x^2} + \frac{1}{\sigma_y^2} - \mathbf{u}^t \mathbf{A} \mathbf{u} \right) e^{-\frac{1}{2} (\mathbf{u}^t \mathbf{u})} \quad (5.11)$$



(a)



(b)

**Figure 5.1** (a) An example 1D second derivative of Gaussian wavelet. (b) An example 2D Gaussian derivative convolution filter applying the parameters:  $\sigma_x = 16$ ,  $\sigma_y = 32$ ,  $\mathbf{b} = (0 \ 0)^t$  and  $\theta = 30^\circ$ .

Figure 5.1(a) shows an example 1D second derivative of Gaussian wavelet. Figure 5.1(b) shows an example 2D Gaussian derivative convolution filter applying the parameters:  $\sigma_x = 16$ ,  $\sigma_y = 32$ ,  $\mathbf{b} = (0 \ 0)^t$  and  $\theta = 30^\circ$ .

### 5.2.2 Morlet Filter

Morlet wavelet is defined by a cosine function modulated by a Gaussian envelop. The 1D Morlet wavelet is defined as follows [18]:

$$\Psi(x) = C_m e^{-x^2/2} \cos(5x) \quad (5.12)$$

where  $C_m$  is a constant. To extend this definition to 2D, one has to map both the linear term,  $x$ , and quadratic term,  $x^2$ , to two-dimensional. In particular, the quadratic term can be mapped to  $\mathbf{u}^t \mathbf{u}$  such that:

$$e^{-x^2/2} \rightarrow e^{-\mathbf{u}^t \mathbf{u}/2} \quad (5.13)$$

where  $\mathbf{u} = (u_1 \ u_2)^t = \mathbf{A}^{1/2} (\mathbf{R} [\mathbf{v} - \mathbf{b}])$ , and  $\mathbf{v} = (x \ y)^t$ . Note that the definitions of  $\mathbf{A}$ ,  $\mathbf{R}$ , and  $\mathbf{b}$  are identical to the ones defined by Equation 5.10.

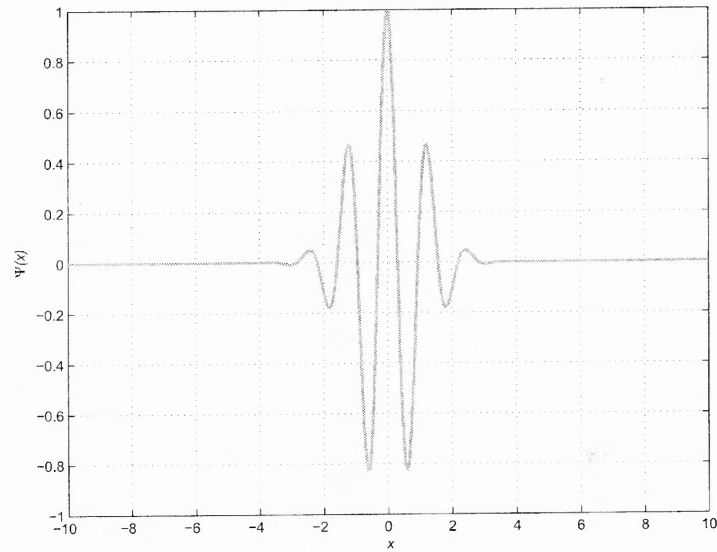
The mapping of the linear term follows the derivation proposed by Akansu et al. [1] that projects  $\mathbf{u}$  onto a parameter vector  $\mathbf{k} = (k \ 0)^t$  [1], [45], [68]:

$$\cos(5x) \rightarrow \cos(5\mathbf{u} \cdot \mathbf{k}) \quad (5.14)$$

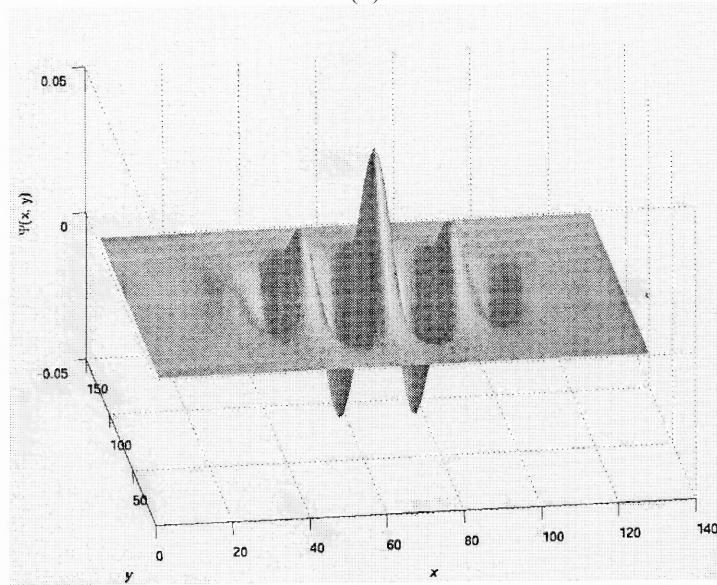
where  $(\cdot)$  is the inner product operator. Therefore, following Equations 5.13 and 5.14, 2D Morlet convolution filter is defined by:

$$\Psi(\mathbf{u}) = C_n e^{-\mathbf{u}^t \mathbf{u}/2} \cos(5\mathbf{u} \cdot \mathbf{k}) \quad (5.15)$$

where  $C_n$  is a constant. Note that unlike the 2D Gaussian derivative filter whose anisotropic characteristic relies on  $\mathbf{A}$  (see Equation 5.10), the 2D Morlet filter is anisotropic regardless of  $\mathbf{A}$  (see Equation 5.10). Therefore, to simplify Equation 5.15,  $\sigma_x = \sigma_y = \sigma$  can be



(a)



(b)

**Figure 5.2** (a) An example 1D Morlet wavelet. (b) An example 2D Morlet convolution filter applying the parameters:  $\sigma = 16$ ,  $\theta = 0$ , and  $k = 1$ .



applied to  $\mathbf{A}$ , and  $\mathbf{b}$  can be assumed to be  $(0 \ 0)^t$ . The definition of  $\mathbf{u}$  is thus:

$$\begin{aligned} \mathbf{u} = \mathbf{A}^{1/2} \mathbf{R} [\mathbf{v} - \mathbf{b}] &= \begin{bmatrix} \frac{1}{\sigma} & 0 \\ 0 & \frac{1}{\sigma} \end{bmatrix} \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ &= \begin{pmatrix} \frac{x \cos \theta - y \sin \theta}{\sigma} \\ \frac{x \sin \theta + y \cos \theta}{\sigma} \end{pmatrix} \end{aligned} \quad (5.16)$$

Finally, following Equations 5.15 and 5.16, the Morlet convolution filter is defined by:

$$\Psi(x, y) = C_n e^{-\mathbf{u}^t \mathbf{u} / 2} \cos\left(\frac{5k(x \cos \theta - y \sin \theta)}{\sigma}\right) \quad (5.17)$$

Figure 5.2(a) shows an example 1D Morlet wavelet. Figure 5.2(b) shows an example 2D Morlet convolution filter applying the parameters:  $\sigma = 16$ ,  $\theta = 0$ , and  $k = 1$ .

### 5.2.3 Complex Morlet Filter

Complex Morlet is defined by a complex exponential modulated by a Gaussian envelop. The 1D complex Morlet is defined as follows [97]:

$$\Psi(x) = \frac{1}{\sqrt{\pi f_b}} e^{2i\pi f_c x} e^{-x^2/f_b} \quad (5.18)$$

where  $f_b$  is a bandwidth parameter, and  $f_c$  is a wavelet center frequency. To extend this definition to two-dimensional, one can apply Equation 5.14 and Equation 5.13 to map the linear ( $x$ ) and the quadratic ( $x^2$ ) terms, respectively:

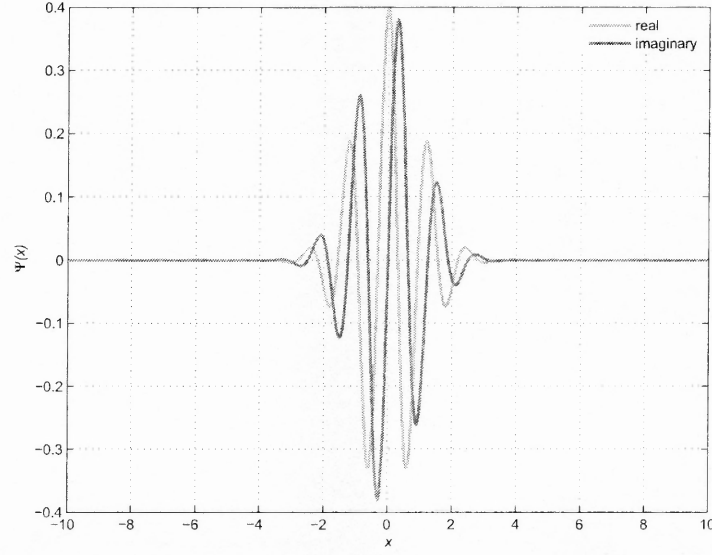
$$e^{-x^2/f_b} \rightarrow e^{-\mathbf{u}^t \mathbf{u} / f_b} \quad (5.19)$$

and

$$e^{2i\pi f_c x} \rightarrow e^{2i\pi f_c (\mathbf{u} \cdot \mathbf{k})} \quad (5.20)$$

The 2D complex Morlet convolution filter is thus defined by [1]:

$$\Psi(\mathbf{u}) = \frac{1}{\sqrt{\pi f_b}} e^{2i\pi f_c (\mathbf{u} \cdot \mathbf{k})} e^{-\mathbf{u}^t \mathbf{u} / f_b} + C_x \quad (5.21)$$



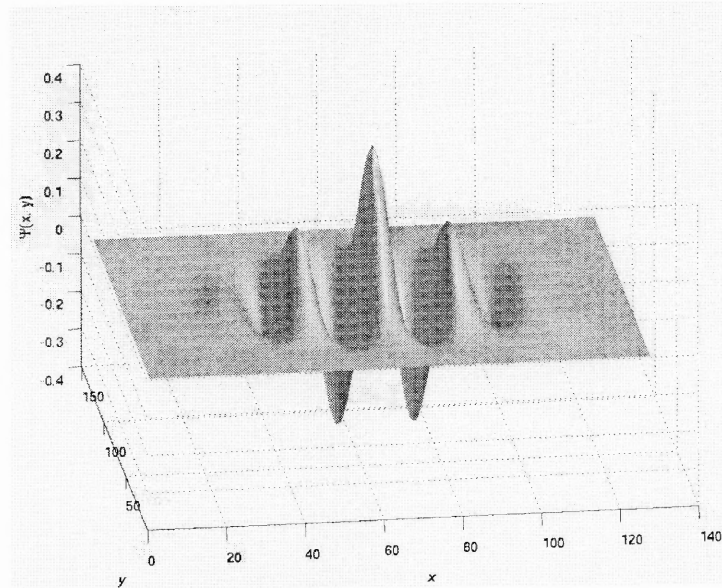
**Figure 5.3** An example 1D complex Morlet wavelet applying the parameters:  $f_b = 2$  and  $f_c = 0.8$ . The real and the imaginary parts are represented by two separate curves.

where  $C_x$  is a constant that enforces the admissibility condition of the wavelet [1]. Note that 2D complex Morlet convolution filter is also anisotropic regardless of  $\mathbf{A}$ . Therefore, Equation 5.16 can be applied to simplify Equation 5.21 such that:

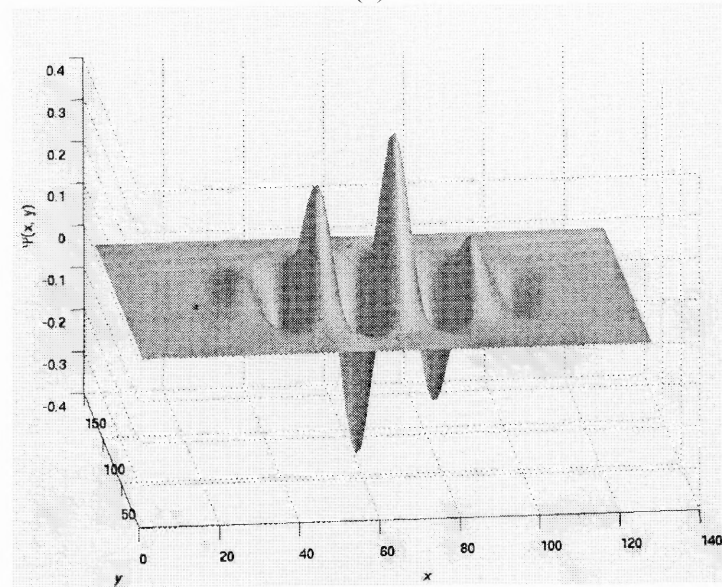
$$\Psi(x, y) = \frac{1}{\sqrt{\pi f_b}} e^{\frac{2i\pi f_c k}{\sigma}(x \cos \theta - y \sin \theta)} e^{-\frac{1}{f_b} \mathbf{u}^t \mathbf{u}} + C_x \quad (5.22)$$

where  $C_x$  is negligible and can be dropped in implementation when  $2\pi f_c k / \sigma \geq 5.6$  [1], [20].

Figure 5.3 shows an example 1D complex Morlet wavelet applying the parameters:  $f_b = 2$  and  $f_c = 0.8$ . Note that the real and the imaginary parts are represented by two separate curves. Figure 5.4 shows an example 2D complex Morlet convolution filter applying the parameters:  $f_b = 2$ ,  $f_c = 0.8$ ,  $\sigma = 16$ ,  $\theta = 0$ , and  $k = 1$ . The real and the imaginary parts are displayed in Figure 5.4 (a) and Figure 5.4 (b), respectively.

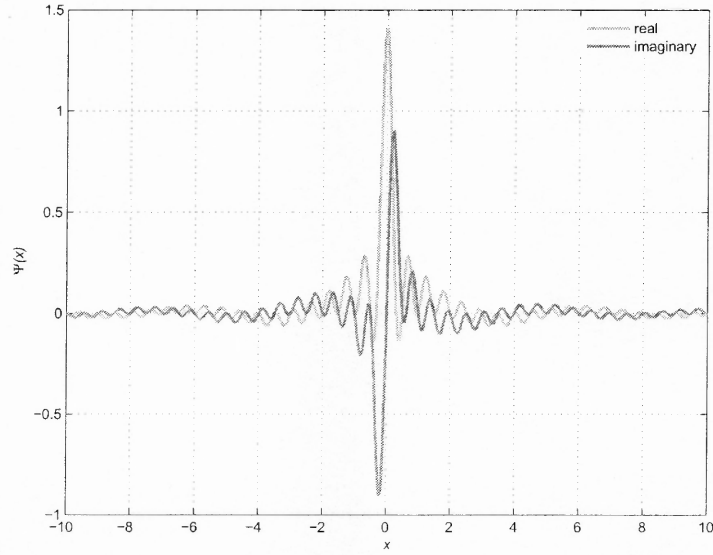


(a)



(b)

**Figure 5.4** An example 2D complex Morlet convolution filter applying the parameters:  $f_b = 2$ ,  $f_c = 0.8$ ,  $\sigma = 16$ ,  $\theta = 0$ , and  $k = 1$ . The real and the imaginary parts are displayed in (a) and (b), respectively.



**Figure 5.5** an example 1D complex frequency B-spline wavelet applying the parameters:  $f_b = 2$ ,  $f_c = 0.8$ , and  $m = 1$ . The real and the imaginary parts are represented by two separate curves.

#### 5.2.4 Complex Frequency B-Spline Filter

The 1D complex frequency B-spline wavelet is defined as follows [97]:

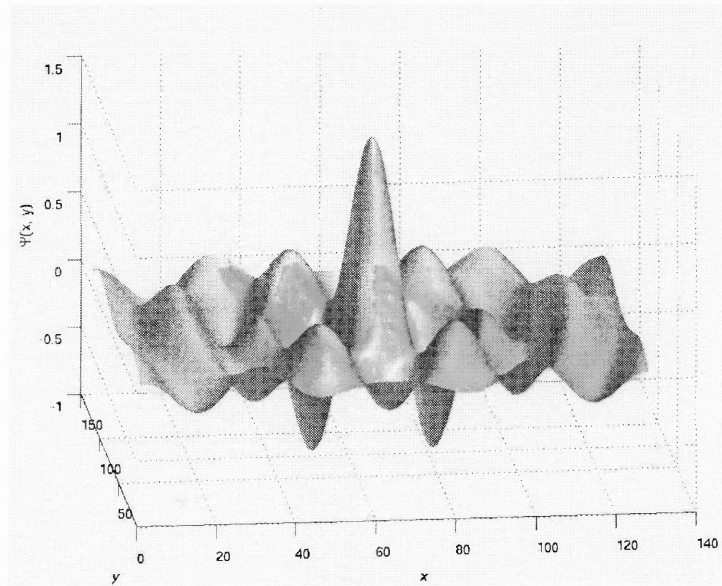
$$\Psi(x) = \sqrt{f_b} \left( \text{sinc} \left( \frac{f_b x}{m} \right) \right)^m e^{2i\pi f_c x} \quad (5.23)$$

where  $f_b$  is a bandwidth parameter,  $f_c$  is a wavelet center frequency, and  $m$  is an integer order parameter ( $m \geq 0$ ). To extend this definition to 2D, one can follow Equation 5.14 such that:

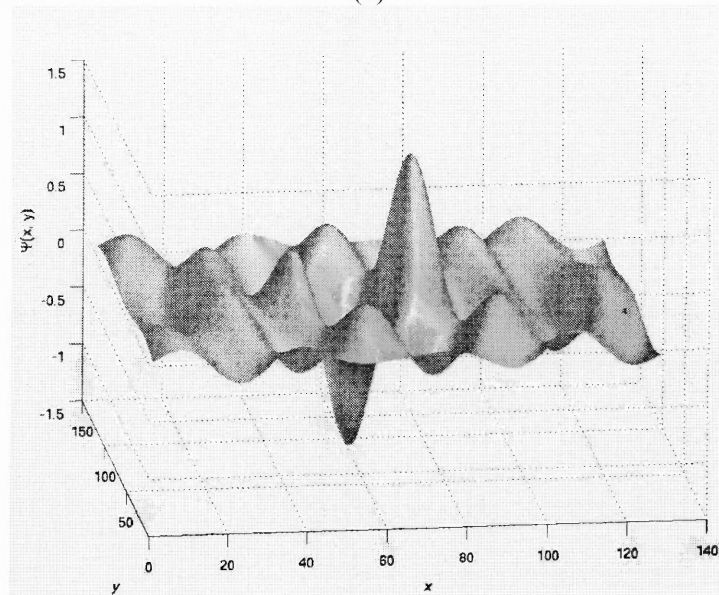
$$\Psi(\mathbf{u}) = \sqrt{f_b} \left( \text{sinc} \left( \frac{f_b (\mathbf{u} \cdot \mathbf{k})}{m} \right) \right)^m e^{2i\pi f_c (\mathbf{u} \cdot \mathbf{k})} \quad (5.24)$$

Then  $\mathbf{u}$  is substituted by Equation 5.16, and the parameter vector  $\mathbf{k}$  is also assumed to be  $(k \ 0)^t$ . The 2D complex frequency B-spline convolution filter thus can be defined by:

$$\Psi(x, y) = \sqrt{f_b} \left( \text{sinc} \left( \frac{f_b k}{m\sigma} (x \cos \theta - y \sin \theta) \right) \right)^m e^{\frac{2i\pi f_c k}{\sigma} (x \cos \theta - y \sin \theta)} \quad (5.25)$$



(a)



(b)

**Figure 5.6** An example 2D complex frequency B-spline convolution filter applying the parameters:  $f_b = 2$ ,  $f_c = 0.6$ ,  $k = 1$ ,  $m = 1$ ,  $\sigma = 20$ , and  $\theta = 0$ . The real and the imaginary parts are displayed in (a) and (b), respectively.

However, with this parameterization, the filter only shows variances in one direction mainly due to the sinc function. To overcome this problem, this research adopts a derivation proposed by Foufoula et al. [24] that replaces  $\mathbf{u} \cdot \mathbf{k}$  with  $\sqrt{\mathbf{u}^t \mathbf{u}}$  in the sinc function:

$$\Psi(\mathbf{u}) = \sqrt{f_b} \left( \text{sinc} \left( \frac{f_b}{m} \sqrt{u_1^2 + u_2^2} \right) \right)^m e^{\frac{2i\pi f_c k}{\sigma} (x \cos \theta - y \sin \theta)} \quad (5.26)$$

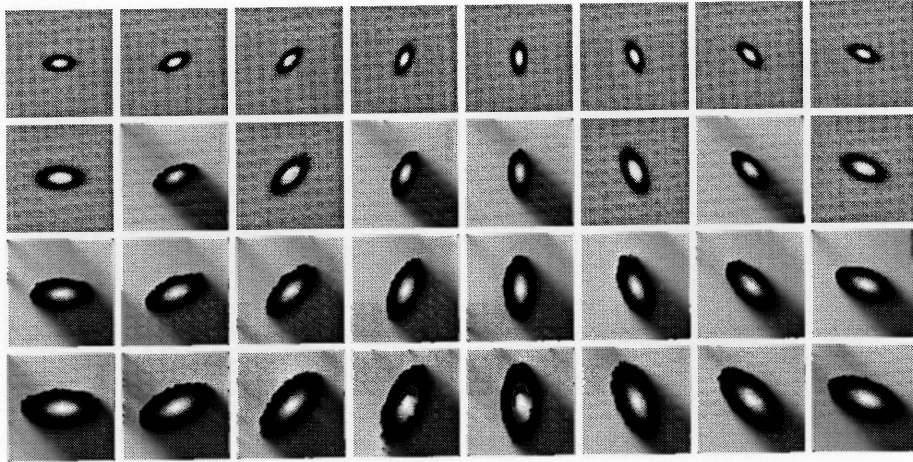
where  $\mathbf{u} = (u_1 \ u_2)^t$ . Figure 5.5 shows an example 1D complex frequency B-spline wavelet applying the parameters:  $f_b = 2$ ,  $f_c = 0.8$ , and  $m = 1$ . The 1D complex frequency B-spline wavelet with  $m = 1$  is also referred as “Shannon wavelet.” Note that the real and the imaginary parts are represented by two separate curves in Figure 5.5. Figure 5.6 shows an example 2D complex frequency B-spline convolution filter applying the parameters:  $f_b = 2$ ,  $f_c = 0.6$ ,  $k = 1$ ,  $m = 1$ ,  $\sigma = 20$ , and  $\theta = 0$ . The real and the imaginary parts are displayed in Figure 5.6 (a) and Figure 5.6 (b), respectively.

### 5.3 Convolution Features and Decision Fusion

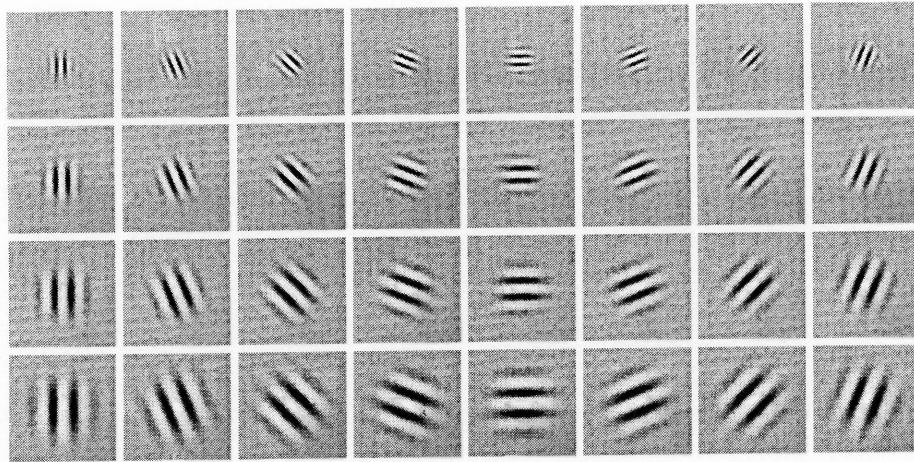
Convolution features are defined by the convolution outputs of an image and a 2D convolution filters derived in Section 5.2. Let  $\mathbf{I}$  be a gray-scale 2D image representing either a 2D or a 3D imaging modality, the convolution features of the image  $\mathbf{I}$  is defined as follows:

$$\mathbf{V}_{\theta,s} = \mathbf{I} \otimes \Psi_{\theta,s} \quad (5.27)$$

where  $\otimes$  denotes the convolution operator, and  $(\theta, s)$  defines (orientation, scale) of a convolution filter. In particular, for each convolution filter, eight orientations  $(\frac{\pi}{8}, \frac{2\pi}{8}, \frac{3\pi}{8}, \dots, \frac{8\pi}{8})$  and four scales are applied to derive convolution features, i.e., for each image  $\mathbf{I}$ , 32 convolutions are computed and 32 output images  $\mathbf{V}_{\theta,s}$  are derived. Note that for the complex filters, such as complex Morlet or complex frequency B-spline, the convolution results contain both real and imaginary parts. In such cases, the magnitudes of the results are used to represent  $\mathbf{V}_{\theta,s}$ . Figures 5.7, 5.8, 5.9 and 5.10 display Gaussian derivative, Morlet, complex



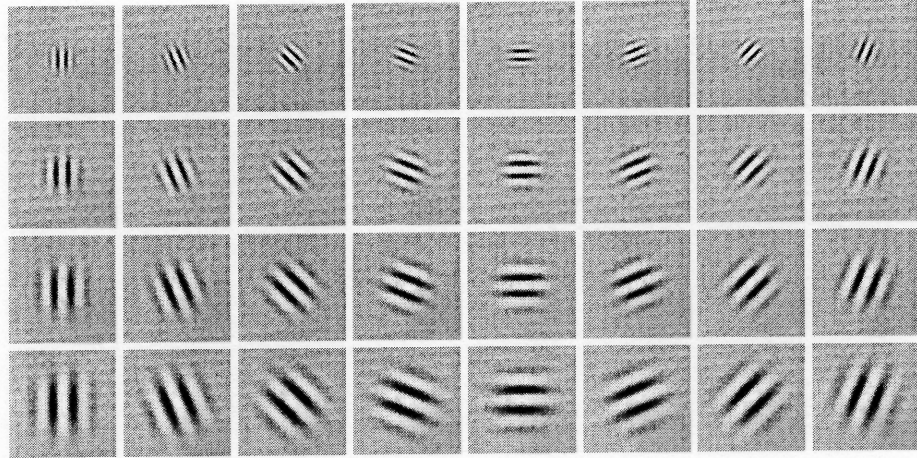
**Figure 5.7** The Gaussian derivative convolution filters in eight orientations and four scales.



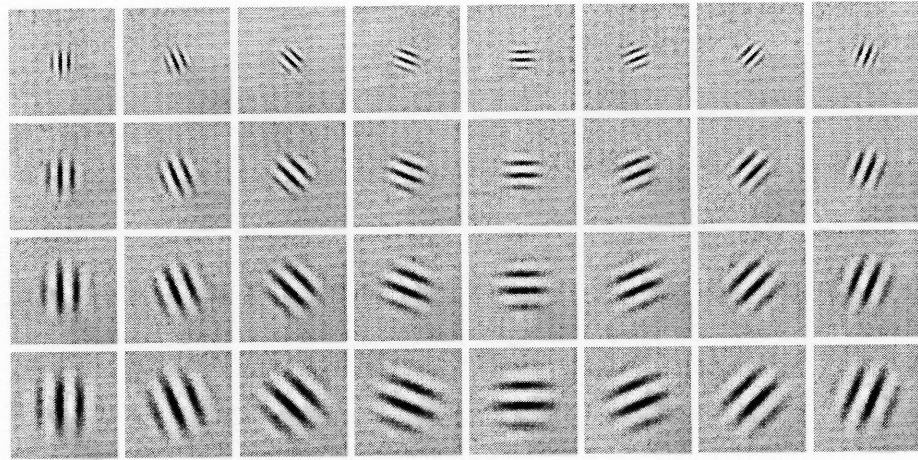
**Figure 5.8** The Morlet convolution filters in eight orientations and four scales.

Morlet, and complex frequency B-spline convolution filters, respectively. In each figure, 32 filters are displayed corresponding to eight orientations and four scales. Note that for complex convolution filters, the real parts are shown in (a) and the imaginary parts are in (b).

After the convolution, all the output images are individually normalized to zero mean and unit variance, and then the normalized images are concatenated to form a convolution feature vector representing the original image  $\mathbf{I}$ . The feature vectors of 2D and 3D modali-



(a)



(b)

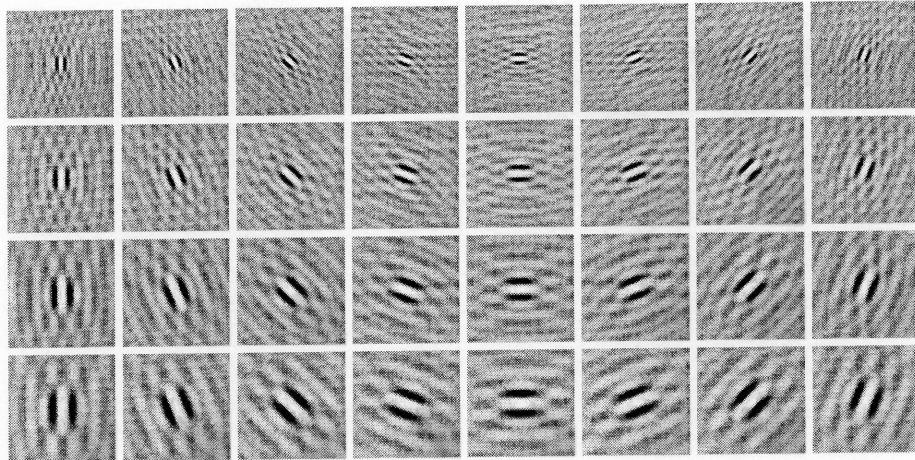
**Figure 5.9** The complex Morlet convolution filters in eight orientations and four scales. (a) The real parts. (b) The imaginary parts.

ties are then applied to BEE baseline algorithm (see Section 3.4.1) to derive two similarity matrices,  $\mathbf{S}_{2D}$  and  $\mathbf{S}_{3D}$ . Finally, the similarity matrix,  $\mathbf{S}$ , is computed by a linear combination of  $\mathbf{S}_{2D}$  and  $\mathbf{S}_{3D}$ :

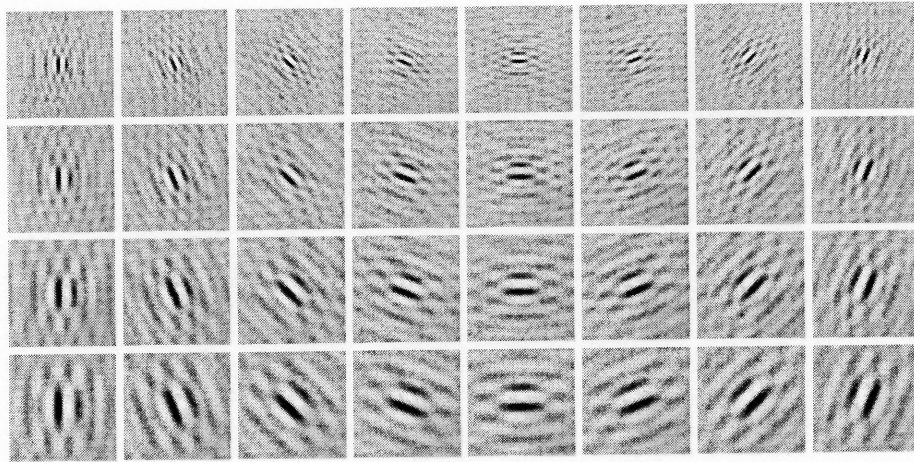
$$\mathbf{S} = \alpha \mathbf{S}_{2D} + (1 - \alpha) \mathbf{S}_{3D} \quad (5.28)$$

where  $\alpha$  is a positive fractional constant.





(a)

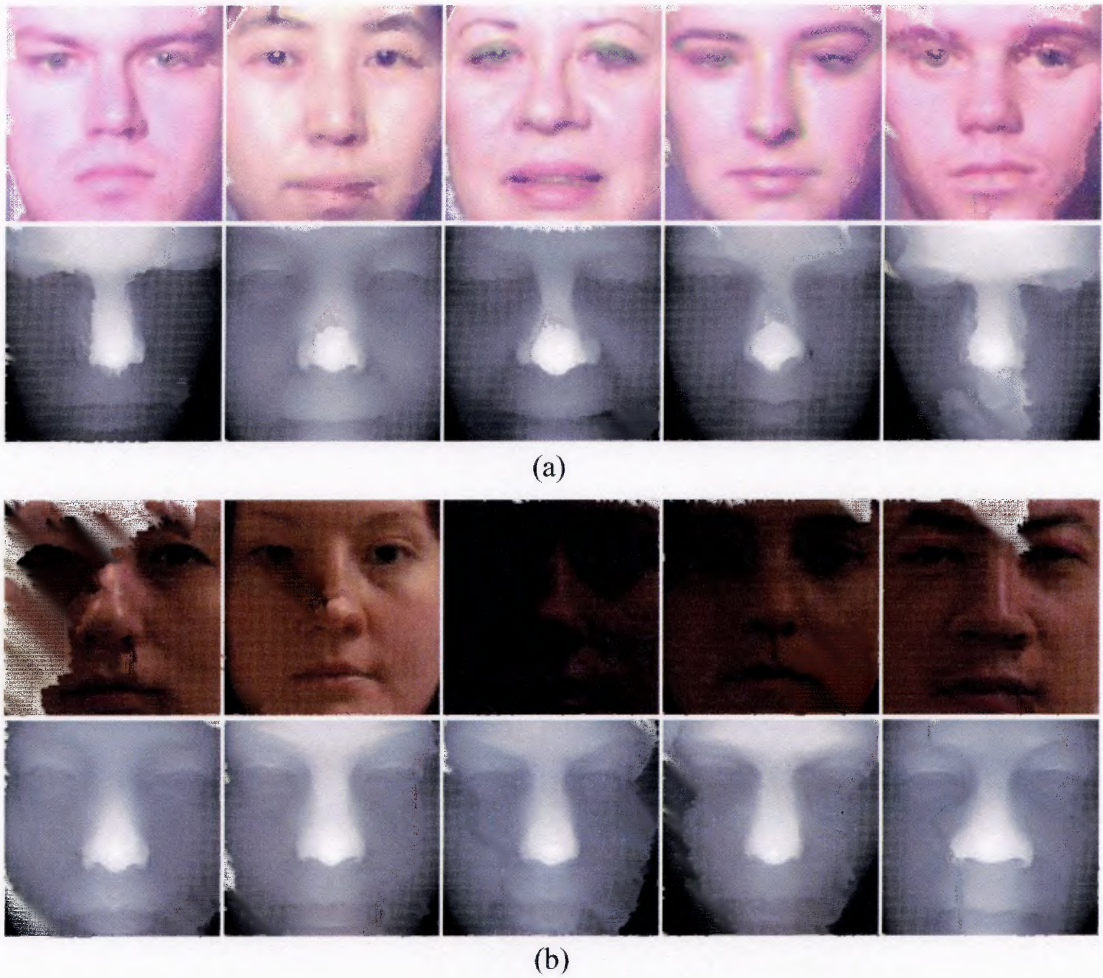


(b)

**Figure 5.10** The complex frequency B-spline convolution filters in eight orientations and four scales. (a) The real parts. (b) The imaginary parts.

## 5.4 Experiments

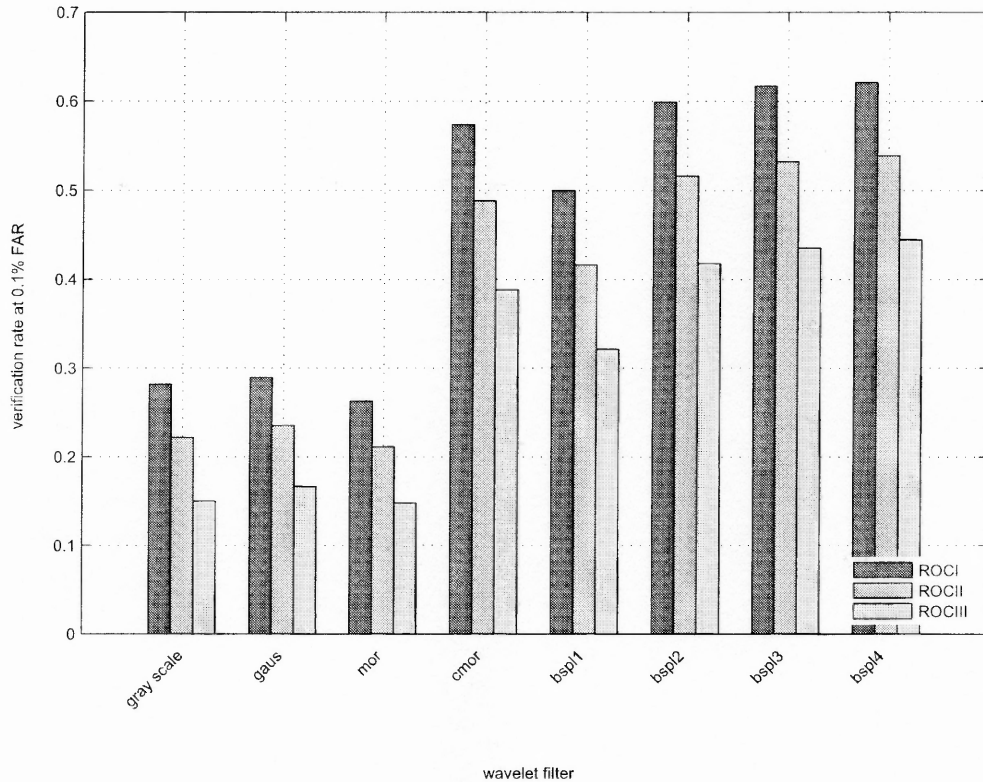
The experiments are organized into three major sets. The first set of experiments applies the four convolution filters to 2D imaging modality to evaluate the discriminating power of the filters. Similarly, the second set of experiments applies the four filters to 3D imaging modality to evaluate the discriminating power of the filters. For the first two sets of experiments, similarity matrices are created, and FRGC ROC performances are generated to assess face recognition performance. Finally, the last set of experiments seeks the optimal



**Figure 5.11** (a) Example controlled 2D training images and their corresponding 3D recordings. (b) Example uncontrolled 2D target images and their corresponding 3D recordings.

fusion parameter,  $\alpha$  (see Equation 5.28), to combine  $\mathbf{S}_{2D}$  and  $\mathbf{S}_{3D}$ , and derives the FRGC ver2.0 Experiment 3 performance.

The experiments presented in this chapter are focused on the FRGC ver2.0 Experiment 3. Unlike the Experiment 4 (see Chapters 3 and 4), where the dataset is divided into three subsets: *training*, *target*, and *query*, the FRGC ver2.0 Experiment 3 is divided into two subsets: the *training* set and the *target / query* set, where the *target* and the *query* sets share the same images. For simplicity, in the following discussion, the term “target set”



**Figure 5.12** Comparative performance of 2D modalities applying the four convolution filters. Face verification rate at 0.1% FAR using gray-scale images, Gaussian derivative convolution features (*gaus*), Morlet convolution features (*mor*), complex Morlet convolution features (*cmor*), and complex frequency B-spline convolution features (*bspl*). For the complex frequency B-spline, four wavelets are defined corresponding to  $m = 1$ ,  $m = 2$ ,  $m = 3$ , and  $m = 4$ , where  $m$  is the order parameter.

is used to refer to this subset. The FRGC ver2.0 Experiment 3 training set contains 943 controlled 2D color images and 943 3D recordings, and the target set consists of 4,007 uncontrolled 2D color images and 4,007 3D recordings. The subjects in the training and the target sets are disjointed.

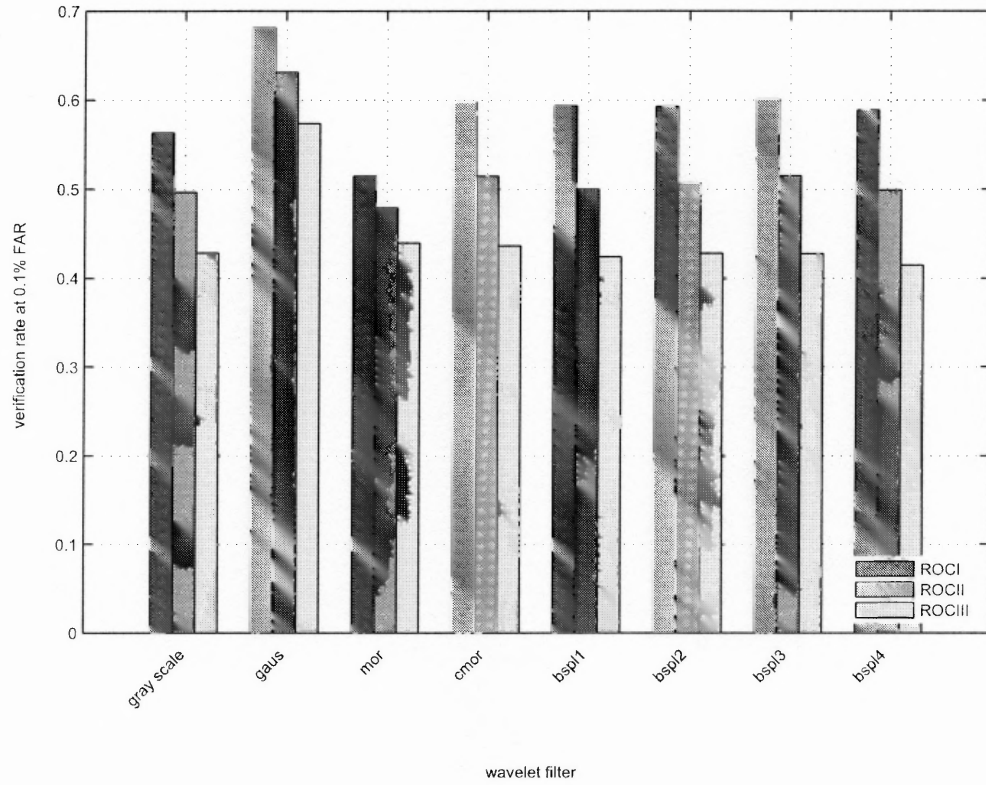
The image modalities used in the experiments are normalized to extract facial regions that contain only faces, so that the face recognition performance is not affected by the factors not related to face, such as hair style. Specifically, for 2D modality, the centers of the eyes are aligned to the predefined locations, and the interocular distances are fix. For

3D modality, the normalization procedure follows the one proposed in [12], [13]. Finally, for both 2D and 3D modalities, subimages of the size  $128 \times 128$  are extracted to locate the facial region. Figures 5.11 (a) and (b), respectively shows example training and target images normalized to the size of  $128 \times 128$ . Note that the controlled 2D color images in the training set have better image quality than the uncontrolled ones in the target set in terms of illumination, resolution, blurring, etc. Also note that due to the difference of image acquisition equipments, the 2D color images in the training set display very distinct color tone than the ones in the target set. Because this variation is not related to subject identities, in the experimental implementation, color images are converted to gray-scale images to reduce the statistical differences caused by image acquisition devices.

The first set of experiments applies the four convolution filters to 2D modalities (i.e., 943 controlled 2D color training images and 4,007 uncontrolled 2D color target images) to extract discriminating convolution features. In particular, the training feature vectors are used to derive the projection basis (see Equation 3.15), and the target feature vectors are used to compute the similarity matrix  $S_{2D}$  (see Equation 3.22). Note that in FRGC ver2.0 Experiment 3, the target and the query are the same, i.e.,  $S_{2D}$  records similarity scores between every two target feature vectors. As a result, the similarity matrix  $S_{2D}$  is symmetric and of size  $4,007 \times 4,007$ . Finally,  $S_{2D}$  is analyzed by BEE to create three ROC performance (see Section 3.4.1 for detailed discussion).

Figure 5.12 shows the face verification rate at 0.1% false acceptance rate (FAR) of ROCI, ROCII, ROCIII performance using gray-scale images, Gaussian derivative convolution features (*gaus*), Morlet convolution features (*mor*), complex Morlet convolution features (*cmor*), and complex frequency B-spline convolution features (*bspl*). Note that for the complex frequency B-spline, four filters are defined corresponding to  $m = 1$ ,  $m = 2$ ,  $m = 3$ , and  $m = 4$  where  $m$  is the order parameter defined by 5.26. The results displayed in Figure 5.12 show that complex convolution features (*cmor*, *bspl1*, *bspl2*, *bspl3*, *bspl4*) generally contribute to better face recognition performance than real-value convolution fea-

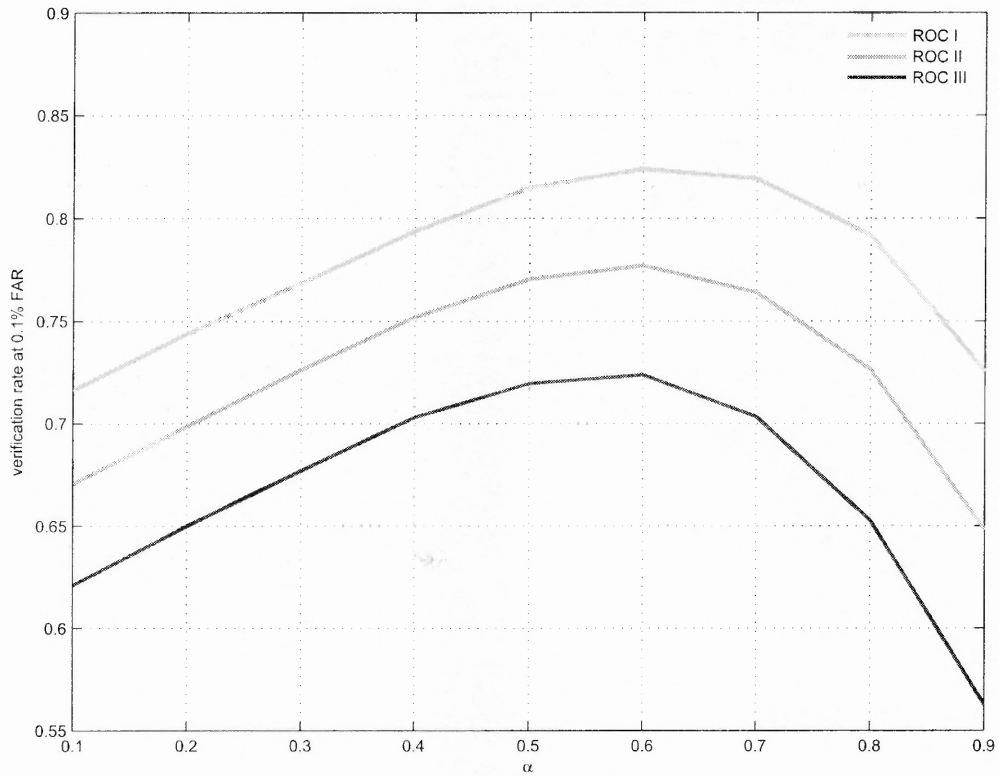




**Figure 5.13** Comparative performance of 3D modalities applying the four convolution filters. Face verification rate at 0.1% FAR using gray-scale images, Gaussian derivative convolution features (*gaus*), Morlet convolution features (*mor*), complex Morlet convolution features (*cmor*), and complex frequency B-spline convolution features (*bspl*). For the complex frequency B-spline, four wavelets are defined corresponding to  $m = 1$ ,  $m = 2$ ,  $m = 3$ , and  $m = 4$ , where  $m$  is the order parameter.

tures (*gaus*, *mor*) or gray-scale images, and suggest that complex convolution filters have the potential to extract discriminating features in 2D modality.

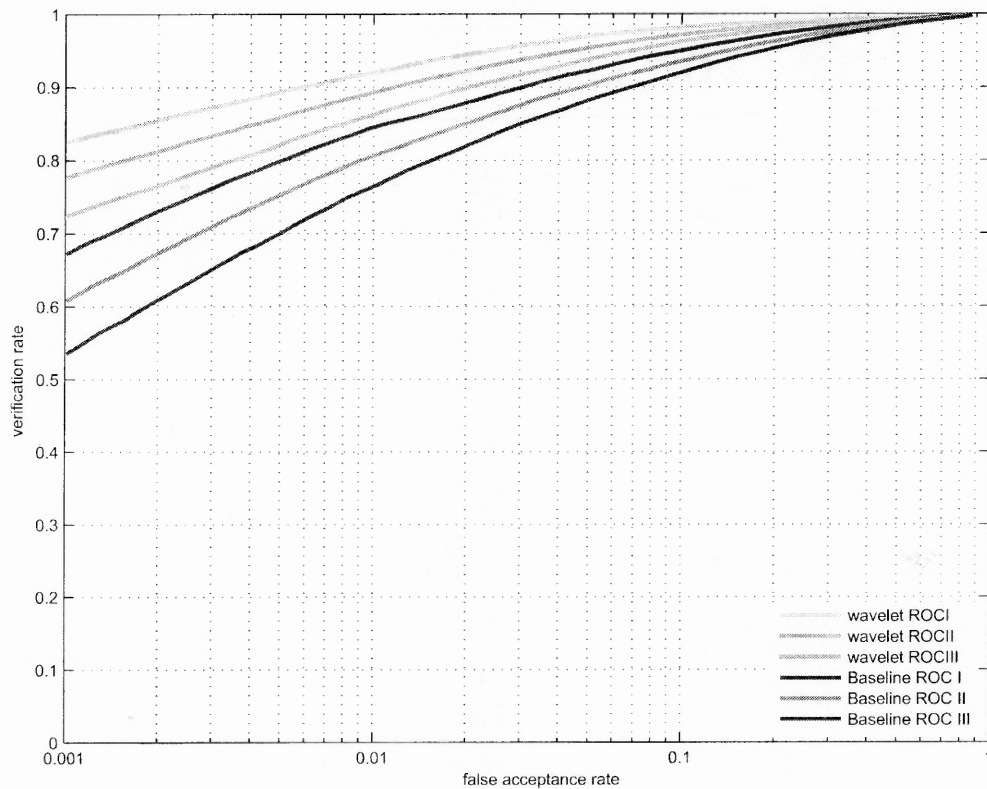
The second set of experiments applies the four convolution filters to 3D modalities (i.e., 943 3D training recordings and 4,007 3D target recordings) to extract discriminating convolution features. This set of experiments shares the same experimental configuration with the first set and generates a square symmetric similarity matrix  $S_{3D}$ . Figure 5.13 shows the face verification rate at 0.1% FAR of ROCI, ROCII, ROCIII performance using gray-scale images, Gaussian derivative convolution features (*gaus*), Morlet convolution features



**Figure 5.14** FRGC ver2.0 Experiment 3 performance using different  $\alpha$ .

(*mor*), complex Morlet convolution features (*cmor*), and complex frequency B-spline convolution features (*bspl*). Again, for the complex frequency B-spline, four filters are defined corresponding to  $m = 1$ ,  $m = 2$ ,  $m = 3$ , and  $m = 4$ . The results displayed in Figure 5.13 show that Gaussian derivative convolution feature achieves the best performance while complex convolution features perform comparably to each other. The results suggest that real-value convolution filters may have the potential to extract discriminating features in 3D modality.

Based on the experimental results reported in the first two sets of experiments, the last set of experiments empirically determines the optimal parameter,  $\alpha$  (see Equation 5.28), for fusing the best performing  $S_{2D}$  (*bspl4*) and the best performing  $S_{3D}$  (*gaus*), and derives FRGC ver2.0 Experiment 3 ROC performance. Figure 5.14 shows the face verification



**Figure 5.15** FRGC ver2.0 Experiment 3 performance using convolution features and similarity fusion. BEE baseline performance is also included for comparison.

rate at 0.1% FAR using different  $\alpha$ . In particular, the performance peaks when  $S_{2D}$  and  $S_{3D}$  are fused with  $\alpha = 0.6$ . Using this parameter, Figure 5.15 displays the FRGC ver2.0 Experiment 3 ROC performance derived by convolution features and similarity fusion. The BEE baseline performance is also included for comparison. Specifically, ROC I has been improved from 67% to 82%, ROC II has been improved from 61% to 78%, and ROC III has been improved from 54% to 72%.

## 5.5 Conclusion

This chapter presents a feature extraction method utilizing convolution filters for 2D and 3D face recognition. Four convolution filters, Gaussian derivatives, Morlet, complex Morlet, and complex frequency B-spline, are defined, and face recognition performance derived using these four convolution filters are investigated. The experimental results applying FRGC ver2.0 Experiment 3 show that complex-valued and real-valued convolution filters may have the potential to extract discriminating features from 2D and 3D imaging modalities, respectively. Furthermore, the fusion of 2D and 3D similarity measures improves FRGC ver2.0 Experiment 3 baseline performance from 67% to 82% (ROC I), from 61% to 78% (ROC II), and from 54% to 72% (ROC III).



## CHAPTER 6

### SUMMARY AND FUTURE RESEARCH

This dissertation focuses on facial analysis and addresses various aspects concerning face detection and recognition. In the following sections, the research achievements and contributions of this dissertation are summarized in Section 6.1, and the future research directions are proposed in Section 6.2.

#### 6.1 Achievements and Contributions

The specific accomplishments of this dissertation is outlined in the following:

##### 1. Face detection

###### (a) Enhance Temporal Differencing for Better FOI Derivation

The temporal differencing is sensitive to dynamic environmental changes, such as illumination variation, and fails to locate a motion FOI when subject stops moving. To address this sensitivity issue, a Gaussian smoothing filter is applied to each frame before calculating the difference image. To detect a motion FOI when subject stops moving, a heuristic procedure is applied to obtain the last-known motion FOI as the current motion FOI if no significant motion is detected. This part of research is detailed in Subsection 2.2.1 on page 8.

###### (b) Employ Skin Color Segmentation for FOI Refinement

The skin color segmentation procedure consists of five stages and is implemented in the  $C_b - C_r$  subspace of the  $YC_bC_r$  color space. It consists of five steps: (i) skin color modeling, (ii) skin-non-skin classification, (iii) region filling and labeling, (iv) rectangle bounding, and (v) skin region rejection. This part of research is detailed in Subsection 2.2.2 on page 10.

###### (c) Design Face and Non-Face Classification Rules Using DBD and SVM

Distribution-based distance (DBD) is defined based on the statistical model of the face class estimated by the first  $M$  principal components, the input image, and the mean face. DBD, together with SVM, are implemented in three classification rules to discriminate between face and non-face patterns. The definition

of DBD is derived in Subsection 2.3.2 on page 15, and the classification rules are detailed in Subsection 2.3.4 on page 20.

## 2. Face Recognition

### (a) Assess Face Recognition Performance in Twelve Color Spaces

To improve existing face recognition algorithms through the utilization of color information, face recognition performance is comparatively assessed in twelve color spaces ( $RGB$ ,  $HSV$ ,  $YUV$ ,  $YC_bC_r$ ,  $XYZ$ ,  $YIQ$ ,  $L^*a^*b^*$ ,  $U^*V^*W^*$ ,  $L^*u^*v^*$ ,  $I_1I_2I_3$ ,  $HSI$ , and  $rgb$ ) by evaluating seven color configurations for every color space. Experimental results using 600 FERET color images corresponding to 200 subjects and 456 FRGC color images of 152 subjects show that some color configurations, such as  $YV$  in the  $YUV$  color space and  $YI$  in the  $YIQ$  color space, help improve face recognition performance. This part of research is detailed in Section 3.3 on page 54.

### (b) Evaluate Face Recognition Performance with Image Alignment Variations

The experimental results show that (i) small variations in eye locations, such as in  $5 \times 5$  pixel region, do not change the face recognition performance much; and (ii) larger variations, such as  $11 \times 11$  pixel region, can significantly deteriorate the face recognition performance. This part of research is detailed in Subsection 3.3.2 on page 65.

### (c) Develop Color Feature Extraction Method Across Different Color Spaces

The color configurations consisting of color component images across different color spaces are investigated. Specifically, the  $YIC_r$ , the  $YIC_b$ , the  $YQC_r$ , and the  $YQC_b$  color configurations defined by the color component images from both the  $YIQ$  and the  $YC_bC_r$  color spaces are applied to improve the baseline performance of the Face Recognition Grand Challenge (FRGC) problems. When using an FRGC ver1.0 dataset containing 1,126 images, the  $YQC_r$  color configuration improves the rank-one face recognition rate of the BEE baseline algorithm from 37% to 70%. When using an FRGC ver2.0 dataset including 30,702 images, the  $YQC_r$  color configuration improves the face verification rate of the BEE baseline algorithm from 13% to 33%. This part of research is detailed in Section 3.4 on page 66.

### (d) Evolve the Optimal Color Space ( $LC_1C_2$ ) for Face Recognition

A new color space,  $LC_1C_2$ , consisting of one luminance ( $L$ ) channel and two chrominance channels ( $C_1, C_2$ ) is introduced as a linear transformation of the input  $RGB$  color space. The specific transformation from the  $RGB$  color space to the  $LC_1C_2$  color space is optimized by genetic algorithms (GAs) where a fitness function guides the evolution toward higher recognition accuracy. After the color feature extraction, i.e., transforming  $RGB$  color images to  $LC_1C_2$

color images, the Fisher linear discriminant (FLD) method is applied to further extract discriminating features, termed color Fisher features (CFFs). The experimental results show that CFFs improve FRGC ver1.0 Experiment 4 rank-one recognition rate from 37% to 73%, and elevate FRGC ver2.0 Experiment 4 face verification rate from 12% to 69%. This part of research is detailed in Section 4.2 on page 86.

(e) Utilize Convolution Filters for 2D and 3D Face Recognition

A feature extraction method utilizing convolution filters for 2D and 3D face recognition is developed. Four convolution filters, Gaussian derivatives, Morlet, complex Morlet, and complex frequency B-spline, are defined, and face recognition performance derived using these four convolution filters are investigated. The experimental results applying FRGC ver2.0 Experiment 3 show that complex-valued and real-valued convolution filters may have the potential to extract discriminating features from 2D and 3D imaging modalities, respectively. Furthermore, the fusion of 2D and 3D similarity measures improves FRGC ver2.0 Experiment 3 baseline performance from 67% to 82% (ROC I), from 61% to 78% (ROC II), and from 54% to 72% (ROC III). The derivation of convolution filters is detailed in Section 5.2 on page 111.

## 6.2 Future Research

Regarding face detection, future research will consider to extend the current system to detect faces in different viewpoints or partially occluded faces. One possibility is to expend the training set to incorporate more facial variations and pose-angled images. The optimal combination of DBD and the SVM output values will also be investigated. One possibility is to apply stochastic search algorithms, such as genetic algorithms, to search for the optimal combination.

Concerning face recognition, the current research has shown that color features and wavelet features are effective for improving face recognition performance and evolutionary computations can help finding the optimal coefficients for feature extraction. Based on these findings, future research may consider to evolve discriminating wavelet features or exploring the feasibility of combining color features and wavelet features to further elevate recognition accuracy.

In addition to independently extending research in both directions, i.e., detection and recognition, the integration of the current face detection system and the feature extraction methods for face recognition will also be studied toward building an automated facial analysis application. As many face recognition applications, such as airport screening and security checkpoint rely on real-time processing, the major challenge of this work will be the processing speed. Although both the DBD-SVM face detection system and color-based feature extraction methods are computational efficient, due to the fact that recognition generally costs considerably more computational resources than detection does, simply combining them together cannot achieve the real-time goal. One way to cope with this difficulty is to employ distributed or parallel computing architecture, where detection and recognition can be handled on two different computers. A good starting point of this approach would be constructing a prototype system that adopts a hand-held computer, e.g., PDA with a camera and Wi-Fi connection, to perform face detection and then submit the detected faces to back-end computers for recognition.

## REFERENCES

- [1] A.N. Akansu and M.J.T. Smith, *Subband and Wavelet Transforms: Design and Applications*, Springer, 1995.
- [2] G. Bebis, E.A. Yfantis, S.J. Louis, and Y.L. Varol, "Genetic object recognition using combinations of views," *IEEE Trans. Evolutionary Computation*, vol. 6, no. 2, pp. 132-146, 2002.
- [3] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, 1997.
- [4] C. Beumier and M. Acheroy, "Face verification from 3D and gray level cues," *Pattern Recognition Letters*, vol. 22, pp. 1321-1329, 2001.
- [5] S. Bleha, C. Slivinsky, and B. Hussien, "Computer-access security systems using keystroke dynamics," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1217-1222, 1990.
- [6] K.W. Bowyer, K. Chang, and P. Flynn, "A survey of approaches to three-dimensional face recognition," in *Proc. the 17th International Conference on Pattern Recognition*, 2004, pp. 358-361.
- [7] A.M. Bronstein, M.M. Bronstein, and R. Kimmel, "Expression-invariant 3D face recognition," in *Proc. Audio and Video-Based Person Authentication*, 2003.
- [8] R. Brunelli and D. Falavigna, "Person identification using multiple cues," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 10, pp. 955-966, 1995.
- [9] ITU-R Recommendation BT.601-2, "Encoding parameters of digital television for studios," in *International Telecommunications Union*, Union, Geneva, Switzerland, 1982.
- [10] A. Can, C.V. Stewart, B. Roysam, and H.L. Tanenbaum, "A feature-based, robust, hierarchical algorithm for registering pairs of images of the curved human retina," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 347-364, 2002.
- [11] G.J. Chamberlin and D.G. Chamberlin, *Colour: Its Measurement, Computation and Application*, Heyden & Son, London, 1980.
- [12] K.I. Chang, K.W. Bowyer, and P.J. Flynn, "Face recognition using 2D and 3D facial data," in *Proc. ACM Workshop Multimodal User Authentication*, December 2003, pp. 25-32.
- [13] K.I. Chang, K.W. Bowyer, and P.J. Flynn, "An evaluation of multimodal 2d+3d face biometrics," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 619-624, 2005.

- [14] R. Chellappa, C.L. Wilson, and S. Sirohey, "Human and machine recognition of faces: A survey," *Proc. IEEE*, vol. 83, no. 5, pp. 705-740, 1995.
- [15] S. Cho and Z. Chi, "Genetic evolution processing of classification," *IEEE Trans. on Knowledge and Data Engineering*, vol. 17, no. 2, pp. 216-231, 2005.
- [16] C. Chua and R. Jarvis, "Point signature: A new representation for 3d object recognition," *International Journal of Computer Vision*, vol. 25, pp. 63-85, 1997.
- [17] S.C. Dass and A.K. Jain, "Markov face models," in *The Eighth IEEE International Conference on Computer Vision*, 2001, pp. 680-687.
- [18] Ingrid Daubechies, *Ten Lectures on Wavelets*, Society for Industrial and Applied Mathematics, 1992.
- [19] J. Daugman, "Face and gesture recognition: Overview," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 675-676, 1997.
- [20] L. Demanet and P. Vandergheynst, "Gabor wavelets on the sphere," in *Proc. Wavelets X Conference*, San Diego, California, 2003.
- [21] A. Elad and R. Kimmel, "Bending invariant representations for surfaces," in *Proc. Computer Vision and Pattern Recognition*, 2001, pp. 168-174.
- [22] K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images," *Journal of the Optical Society of America A*, vol. 14, pp. 1724-1733, 1997.
- [23] G.D. Finlayson, S.S. Chatterjee, and B.V. Funt, "Color angular indexing," in *Proc. European Conference on Computer Vision*, Cambridge, UK, April 14-18, 1996.
- [24] E. Foufoula-Georgiou and P. Kumar, *Wavelets in Geophysics, Volume 4 (Wavelet Analysis and Its Applications)*, Academic Press, 1994.
- [25] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, second edition, 1990.
- [26] C. Garcia and G. Tziritas, "Face detection using quantized skin color regions merging and wavelet packet analysis," *IEEE Trans. Multimedia*, vol. 1, no. 3, pp. 264-277, 1999.
- [27] J.M. Geusebroek, R. van den Boomgaard, A.W.M. Smeulders, and H. Geerts, "Color invariance," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 12, pp. 1338-1350, 2001.
- [28] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.
- [29] D.E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in *Proc. of the 2nd International Conference on Genetic algorithms and their application*, 1987, pp. 41-49.

- [30] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Prentice Hall, 2001.
- [31] G Gordon, "Face recognition based on depth and curvature features," in *SPIE Proc.: Geometric Methods in Computer Vision*, 1991, vol. 1570, pp. 234-247.
- [32] N. Habili and C.C. Lim, "Hand and face segmentation using motion and color cues in digital image sequences," in *Proc. IEEE International Conference on Multimedia and Expo 2001*, Tokyo, Japan, August 22-25, 2001.
- [33] G. Healey and D.A. Slater, "Global color constancy: Recognition of objects by use of illumination invariant properties of color distributions," *Journal of the Optical Society of America A*, vol. 11, no. 11, pp. 3003-3010, 1994.
- [34] B. Heisele, T. Poggio, and M. Pontil, "Face detection in still gray images," A.I. memo AIM-1687, Artificial Intelligence Laboratory, MIT, 2000.
- [35] E. Hernandez and G.L. Weiss, *A First Course on Wavelets*, CRC-Press, 1996.
- [36] C. Heshner, A. Srivastava, and G. Erlebacher, "A novel technique for face recognition using range images," in *Proc. Seventh International Symposium on Signal Processing and Its Applications*, 2003.
- [37] E. Hjelmås and B.K. Low, "Face detection: A survey," *Computer Vision and Image Understanding*, vol. 83, pp. 236-274, 2001.
- [38] P. Ho, "Rotation invariant real-time face detection and recognition system," A.I. memo AIM-2001-010, Artificial Intelligence Laboratory, MIT, 2001.
- [39] J. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975.
- [40] R.L. Hsu, M. Abdel-Mottaleb, and A.K. Jain, "Face detection in color images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696-706, 2002.
- [41] A.K. Jain, R. Bolle, and S. Pankanti, *Biometrics: Personal Identification in Networked Society*, Springer, 1999.
- [42] A.K. Jain, S. Pankanti, S. Prabhakar, L. Hong, and A. Ross, "Biometrics: A grand challenge," in *Proc. the 17th International Conference on Pattern Recognition*, 2004, pp. 935-942.
- [43] A.K. Jain, S. Prabhakar, and H. Lin, "A multichannel approach to fingerprint classification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 348-359, 1999.
- [44] I.J. Jeon, B.M. Choi, and P.K. Rhee, "Evolutionary reconfigurable architecture for robust face recognition," in *Proc. of the 17th International Symposium on Parallel and Distributed Processing*, 2003.

- [45] Y. Ji, K.H. Chang, and C.C. Hung, "Efficient edge detection and object segmentation using Gabor filters," in *Proc. 42nd Annual Southeast Regional Conference*, Alabama, 2004.
- [46] D.B. Judd and G. Wyszecki, *Color in Business, Science and Industry*, John Wiley & Sons, Inc., 1975.
- [47] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 103-108, 1990.
- [48] Z.M. Kovacs-Vajna, "A fingerprint verification system based on triangular matching and dynamic time warping," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1266-1276, 2000.
- [49] S. Lao, Y. Sumi, M. Kawade, and F. Tomita, "3D template matching for pose invariant face recognition using 3D facial model built with isoluminance line based stereo system," in *Proc. International Conference on Pattern Recognition*, 2000, vol. 2, pp. 911-916.
- [50] S. Lee, "Off-line recognition of totally unconstrained handwritten numerals using multi-layer cluster neural network," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 6, pp. 648-652, 1996.
- [51] Y. Lee, K. Park, J. Shim, and T. Yi, "3D face recognition using statistical multiple features for the local depth information," in *Proc. 16th International Conference on Vision Interface*, June 2003.
- [52] C. Liu, "A Bayesian discriminating features method for face detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 725-740, 2003.
- [53] C. Liu, "Enhanced independent component analysis and its application to content based face image retrieval," *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 2, pp. 1117-1127, 2004.
- [54] C. Liu, "Gabor-based kernel PCA with fractional power polynomial models for face recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 572-581, 2004.
- [55] C. Liu and H. Wechsler, "Robust coding schemes for indexing and retrieval from large face databases," *IEEE Trans. on Image Processing*, vol. 9, no. 1, pp. 132-137, 2000.
- [56] C. Liu and H. Wechsler, "A shape and texture based enhanced Fisher classifier for face recognition," *IEEE Trans. on Image Processing*, vol. 10, no. 4, pp. 598-608, 2001.
- [57] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Trans. on Image Processing*, vol. 11, no. 4, pp. 467-476, 2002.
- [58] A.K. Louis, D. Maass, and A. Rieder, *Wavelets : Theory and Applications*, John Wiley & Sons, Inc., 1997.



- [59] L. Lucchese and S.K. Mitra, "Unsupervised color image segmentation," in *Proc. IEEE Workshop on Multimedia Signal Processing*, Los Angeles, CA, December 7-9, 1998.
- [60] R. Manmatha and J.L. Rothfeder, "A scale space approach for automatically segmenting words from historical handwritten documents," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1212-1225, 2005.
- [61] G. Medioni and R. Waupotitsch, "Face recognition and modeling in 3D," in *Proc. IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, October 2003, pp. 232-233.
- [62] G. Medioni and R. Waupotitsch, "Face recognition and modeling in 3D," in *Proc. IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, October 2003.
- [63] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696-710, 1997.
- [64] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 4, pp. 349-361, 2001.
- [65] H. Moon and P.J. Phillips, "Analysis of pca-based face recognition algorithms," in *Empirical Evaluation Techniques in Computer Vision*, K.W. Bowyer and P.J. Phillips, Eds. 1998, Wiley-IEEE Computer Society.
- [66] H. Moon and P.J. Phillips, "Computational and performance aspects of PCA-based face-recognition algorithms," *Perception*, vol. 30, pp. 303-321, 2001.
- [67] A.B. Moreno, A. Sanchez, J.F. Velez, and F.J. Diaz, "Face recognition using 3D surface-extracted descriptors," in *Proc. Irish Machine Vision and Image Processing Conference (IMVIP 2003)*, September 2003.
- [68] P. Moreno, A. Bernardino, and J. Santos-Victor, "Gabor parameter selection for local feature detection," in *Proc. 2nd Iberian Conference on Pattern Recognition and Image Analysis*, Estoril, Portugal, June 7-9 2005.
- [69] Y. Ohta, *Knowledge-Based Interpretation of Outdoor Natural Color Scenes*, Pitman Publishing, London, 1985.
- [70] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," in *Proc. Computer Vision and Pattern Recognition*, 1994, pp. 84-91.
- [71] P.J. Phillips, P.J. Flynn, T. Scruggs, K.W. Bowyer, J. Chang, K. Hoffman, Marques J., J. Min, and W. Worek, "Overview of the face recognition grand challenge," in *Proc. Computer Vision and Pattern Recognition*, San Diego, June 20-25 2005, pp. 947-954.

- [72] P.J. Phillips and E.M. Newton, "Meta-analysis of face recognition algorithms," in *Proc. Fifth International Conference on Automatic Face and Gesture Recognition*, March, 2002, pp. 224-230.
- [73] P.J. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," *Image and Vision Computing*, vol. 16, pp. 295-306, 1998.
- [74] R. Plamondon and S.N. Srihari, "Online and off-line handwriting recognition: a comprehensive survey," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 63-84, 2000.
- [75] R.J. Qian and T.S. Huang, "Object detection using hierarchical MRF and MAP estimation," in *Proc. Computer Vision and Pattern Recognition*, 1997, pp. 186-192.
- [76] H.A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23-38, 1998.
- [77] H.A. Rowley, S. Baluja, and T. Kanade, "Rotation invariant neural network-based face detection," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Santa Barbara, California, USA, June 23-25, 1998, pp. 38-44.
- [78] E. Saber, A.M. Tekalp, R. Eschbach, and K. Knox, "Automatic image annotation using adaptive color classification," *Graphical Models and Image Processing*, vol. 58, no. 2, pp. 115-126, 1996.
- [79] H. Sakano, "Genetic translator: How to apply query learning to practical ocr," in *Proc. of the 16th International Conference on Pattern Recognition*, 2002, pp. 184-187.
- [80] A. Samal and P.A. Iyengar, "Automatic recognition and analysis of human faces and facial expression: A survey," *Pattern Recognition*, vol. 25, no. 1, pp. 65-77, 1992.
- [81] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos, "Biometric identification through hand geometry measurements," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1168-1171, 2000.
- [82] B. Sareni and L. Krähenbuhl, "Fitness sharing and niching methods revisited," *IEEE Trans. Evolutionary Computation*, vol. 2, no. 3, pp. 97-106, 1998.
- [83] H. Schneiderman and T. Kanade, "Probabilistic modeling of local appearance and spatial relationships for object recognition," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Santa Barbara, California, USA, June 23-25, 1998, pp. 45-51.
- [84] H. Schneiderman and T. Kanade, "A statistical method for 3D object detection applied to faces and cars," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2000, pp. 746-751.

- [85] P. Shih and C. Liu, "Face detection using discriminating feature analysis and support vector machine in video," in *Proc. International Conference on Pattern Recognition*, Cambridge, United Kingdom, August 23-26 2004, pp. 407-410.
- [86] P. Shih and C. Liu, "Comparative assessment of content-based face image retrieval in different color spaces," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, no. 7, pp. 873-893, 2005.
- [87] P. Shih and C. Liu, "Evolving effective color features for improving frgc baseline performance," in *Proc. IEEE Computer Vision and Pattern Recognition Workshop on Face Recognition Grand Challenge Experiments*, San Diego, California, June 21 2005.
- [88] P. Shih and C. Liu, "Face detection using discriminating feature analysis and support vector machine," *Pattern Recognition*, vol. 39, no. 2, pp. 260-276, 2006.
- [89] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349-1380, 2000.
- [90] A.R. Smith, "Color gamut transform pairs," *Computer Graphics*, vol. 12, no. 3, pp. 12-19, 1978.
- [91] R. Snelick, U. Uludag, A. Mink, M. Indovina, and A. Jain, "Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 450-455, 2005.
- [92] K. Sobottka and I. Pitas, "Segmentation and tracking of faces in color images," in *Proc. The Second International Conference on Automatic Face and Gesture Recognition*, Killington, Vermont, October 13-16, 1996.
- [93] K.K. Sung, *Learning and Example Selection for Object and Pattern Detection*, Ph.D. thesis, AI Lab, MIT, 1996.
- [94] K.K. Sung and T. Poggio, "Example-based learning for view-based human face detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 39-51, 1998.
- [95] M.J. Swain and D.H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.
- [96] D.L. Swets and J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831-836, 1996.
- [97] A. Teolis, *Computational signal processing with wavelets*, Birkhuser, 1998.
- [98] J.C. Terrillon and S. Akamatsu, "Comparative performance of different chrominance spaces for color segmentation and detection of human faces in complex scene images," in *Proc. the 12th Conf. on Vision Interface*, 1999, pp. 180-187.

- [99] J.C. Terrillon, M.N. Shirazi, H. Fukamachi, and S. Akamatsu, "Comparative performance of different skin chrominance models and chrominance space for the automatic detection of human faces in color images," in *Proc. The Fourth International Conference on Face and Gesture Recognition*, Grenoble, France, March 28-30, 2000.
- [100] L Torres, J.Y. Reutter, and L Lorente, "The importance of color information in face recognition," in *Proc. IEEE International Conference on Image Processing*, Kobe, Japan, October 24-28, 1999.
- [101] F. Tsalakanidou, S. Malassiotis, and M. Strintzis, "Integration of 2D and 3D images for enhanced face authentication," in *Proc. Sixth International Conference on Automated Face and Gesture Recognition*, May 2004, pp. 266-271.
- [102] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 13, no. 1, pp. 71-86, 1991.
- [103] H. Vafaie and K. De Jong, "Feature space transformation using genetic algorithms," *IEEE Intelligent Systems*, vol. 13, no. 2, pp. 57-65, 1998.
- [104] Y.N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, second edition, 1999.
- [105] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Kauai, Hawaii, December 2001, pp. 511-518.
- [106] Y. Wang, C. Chua, and Y. Ho, "Facial feature detection and face recognition from 2d and 3d images," *Pattern Recognition Letters*, vol. 23, pp. 1191-1202, 2002.
- [107] D. Whitley and J Kauth, "Genitor: A different genetic algorithm," in *Proc. the 1988 Rocky Mountain Conference on Artificial Intelligence*, 1988.
- [108] L. Wiskott, J.M. Fellous, N. Kruger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775-779, 1997.
- [109] H. Wu, Q. Chen, and M. Yachida, "Face detection from color images using a fuzzy pattern matching method," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 6, pp. 557-563, 1999.
- [110] Y. Xu, B. Li, and B. Wang, "Face recognition by fast independent component analysis and genetic algorithm," in *Proc. of the 5th International Conference on Computer and Information Technology*, 2004, pp. 194-198.
- [111] M. Yamada, K. Ebihara, and J. Ohya, "New robust real-time method for extracting human silhouettes from color images," in *Proc. The Third International Conference on Automatic Face and Gesture Recognition*, Nara, Japan, April 14-16, 1998.

- [112] J. Yang and A. Waibel, "A real-time face tracker," in *Proc. IEEE Workshop on Applications of Computer Vision*, Sarasota, Florida, December 2-4, 1996.
- [113] M.H. Yang, N. Ahuja, and D. Kriegman, "Face detection using mixtures of linear subspaces," in *Proc. Fifth International Conference on Automatic Face and Gesture Recognition*, Grenoble, France, March, 2000, pp. 70-76.
- [114] M.H. Yang, D. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34-58, 2002.
- [115] A. Yip and P. Sinha, "Role of color in face recognition," AI Memo 2001-035, MIT, December 2001.
- [116] A.L. Yuille, "Deformable templates for face recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 59-70, 1991.
- [117] W. Zhao, R. Chellappa, J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys*, vol. 35, pp. 399-458, 2003.
- [118] W. Zhao, A. Krishnaswamy, R. Chellappa, D.L. Swets, and J. Weng, "Discriminant analysis of principal components for face recognition," in *Face Recognition: From Theory to Applications*, H. Wechsler, P.J. Phillips, V. Bruce, F.F. Soulie, and T.S. Huang, Eds., pp. 73-85. Springer-Verlag, 1998.