

Stan Z. Li  
Anil K. Jain *Editors*

# Handbook of Face Recognition

*Second Edition*

 Springer

# Handbook of Face Recognition



Stan Z. Li • Anil K. Jain  
Editors

# Handbook of Face Recognition

Second Edition



Springer

*Editors*

Stan Z. Li

Institute of Automation, Center Biometrics  
Research & Security  
Chinese Academy of Science  
Room 1227, No. 95 Zhongguancun East Rd  
Beijing 100190  
People's Republic of China  
[szli@cbsr.ia.ac.cn](mailto:szli@cbsr.ia.ac.cn)

Anil K. Jain

Dept. Computer Science & Engineering  
Michigan State University  
East Lansing, MI 48824-1226  
USA

[jain@cse.msu.edu](mailto:jain@cse.msu.edu)

ISBN 978-0-85729-931-4

e-ISBN 978-0-85729-932-1

DOI 10.1007/978-0-85729-932-1

Springer London Dordrecht Heidelberg New York

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

Library of Congress Control Number: 2011936022

© Springer-Verlag London Limited 2011

Apart from any fair dealing for the purposes of research or private study, or criticism or review, as permitted under the Copyright, Designs and Patents Act 1988, this publication may only be reproduced, stored or transmitted, in any form or by any means, with the prior permission in writing of the publishers, or in the case of reprographic reproduction in accordance with the terms of licenses issued by the Copyright Licensing Agency. Enquiries concerning reproduction outside those terms should be sent to the publishers.

The use of registered names, trademarks, etc., in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant laws and regulations and therefore free for general use.

The publisher makes no representation, express or implied, with regard to the accuracy of the information contained in this book and cannot accept any legal responsibility or liability for any errors or omissions that may be made.

*Cover design:* VTeX UAB, Lithuania

Printed on acid-free paper

Springer is part of Springer Science+Business Media ([www.springer.com](http://www.springer.com))

# Preface

Face recognition is one of the most important abilities that we use in our daily lives. There are several reasons for the growing interest in automated face recognition, including rising concerns for public security, the need for identity verification for physical and logical access, and the need for face analysis and modeling techniques in multimedia data management and digital entertainment. Research in automatic face recognition started in the 1960s. Recent years have seen significant progress in this area and a number of face recognition and modeling systems have been developed and deployed. However, accurate and robust face recognition still offers a number of challenges to computer vision and pattern recognition researchers, especially under unconstrained environments.

This book is written with two primary motivations. The first is to compile major approaches, algorithms, and technologies available for automated face recognition. The second is to provide a reference book to students, researchers, and practitioners.

The book is intended for anyone who plans to work in face recognition or who wants to become familiar with the state-of-the-art in face recognition. It also provides references for scientists and engineers working in image processing, computer vision, biometrics and security, computer graphics, animation, and the computer game industry. The material contained in the book fits the following categories: advanced tutorial, state-of-the-art survey, and a guide to current technology.

This second edition consists of twenty seven chapters, with additions and updates from the sixteen chapters in the first edition. It covers all the subareas and major components necessary for developing face recognition algorithms, designing operational systems, and addressing related issues in large scale applications. Each chapter focuses on a specific topic or system component, introduces background information, reviews up-to-date techniques, presents results, and points out challenges and future directions.

The twenty seven chapters are divided into four parts according to the main problems addressed. Part I, *Face Image Modeling and Representation*, consists of ten chapters, presenting theories in face image modeling and facial feature representation. Part II, *Face Recognition Techniques*, also consists of ten chapters, presenting techniques for face detection, landmark detection, and face recognition in static face

images, in video, in non-visible spectrum images, and in 3D. Part III, *Performance Evaluation: Machines and Humans*, consists of three chapters, presenting methods and programs for face recognition evaluation and also studies and comparisons with human performance. Part IV, *Face Recognition Applications*, consists of four chapters, presenting various applications of face recognition and related issues.

A project like this requires the efforts and support of many individuals and organizations. First of all, we would like to thank all the authors for their outstanding contributions which made this edition possible. We also thank Wayne Wheeler and Simon Rees, the Springer editors for their support and patience during the course of this project. Thanks are also due to a number of individuals who have assisted us during the editing phase of this project, including Shikun Feng, Shengcai Liao, Xiangsheng Huang, Brendan Klare, Unsang Park, Abhishek Nagar, and not the least Kim Thompson for her careful proofreading of the manuscript. Stan Z. Li would like to acknowledge the support of the Chinese National Natural Science Foundation Project #61070146, the National Science and Technology Support Program Project #2009BAK43B26, the AuthenMetric R&D Funds, and the TABULA RASA project (<http://www.tabularasa-euproject.org>) under the Seventh Framework Programme for research and technological development (FP7) of the European Union (EU), grant agreement #257289. Anil Jain's research was partially supported by the WCU (World Class University) program funded by the Ministry of Education, Science and Technology through the National Research Foundation of Korea (R31-10008) to the Brain & Cognitive Engineering Department, Korea University where he is an Adjunct Professor.

Beijing, People's Republic of China  
East Lansing, USA

Stan Z. Li  
Anil K. Jain

# Contents

<b>1</b>	<b>Introduction</b>	1
	Stan Z. Li and Anil K. Jain	
<b>Part I Face Image Modeling and Representation</b>		
<b>2</b>	<b>Face Recognition in Subspaces</b>	19
	Gregory Shakhnarovich and Baback Moghaddam	
<b>3</b>	<b>Face Subspace Learning</b>	51
	Wei Bian and Dacheng Tao	
<b>4</b>	<b>Local Representation of Facial Features</b>	79
	Joni-Kristian Kämäräinen, Abdenour Hadid, and Matti Pietikäinen	
<b>5</b>	<b>Face Alignment Models</b>	109
	Phil Tresadern, Tim Cootes, Chris Taylor, and Vladimir Petrović	
<b>6</b>	<b>Morphable Models of Faces</b>	137
	Reinhard Knothe, Brian Amberg, Sami Romdhani, Volker Blanz, and Thomas Vetter	
<b>7</b>	<b>Illumination Modeling for Face Recognition</b>	169
	Ronen Basri and David Jacobs	
<b>8</b>	<b>Face Recognition Across Pose and Illumination</b>	197
	Ralph Gross, Simon Baker, Iain Matthews, and Takeo Kanade	
<b>9</b>	<b>Skin Color in Face Analysis</b>	223
	J. Birgitta Martinkauppi, Abdenour Hadid, and Matti Pietikäinen	
<b>10</b>	<b>Face Aging Modeling</b>	251
	Unsang Park and Anil K. Jain	

**Part II Face Recognition Techniques**

<b>11</b>	<b>Face Detection</b>	277
	Stan Z. Li and Jianxin Wu	
<b>12</b>	<b>Facial Landmark Localization</b>	305
	Xiaoqing Ding and Liting Wang	
<b>13</b>	<b>Face Tracking and Recognition in Video</b>	323
	Rama Chellappa, Ming Du, Pavan Turaga, and Shaohua Kevin Zhou	
<b>14</b>	<b>Face Recognition at a Distance</b>	353
	Frederick W. Wheeler, Xiaoming Liu, and Peter H. Tu	
<b>15</b>	<b>Face Recognition Using Near Infrared Images</b>	383
	Stan Z. Li and Dong Yi	
<b>16</b>	<b>Multispectral Face Imaging and Analysis</b>	401
	Andreas Koschan, Yi Yao, Hong Chang, and Mongi Abidi	
<b>17</b>	<b>Face Recognition Using 3D Images</b>	429
	I.A. Kakadiaris, G. Passalis, G. Toderici, E. Efrat, P. Perakis, D. Chu, S. Shah, and T. Theoharis	
<b>18</b>	<b>Facial Action Tracking</b>	461
	Jörgen Ahlberg and Igor S. Pandzic	
<b>19</b>	<b>Facial Expression Recognition</b>	487
	Yingli Tian, Takeo Kanade, and Jeffrey F. Cohn	
<b>20</b>	<b>Face Synthesis</b>	521
	Yang Wang, Zicheng Liu, and Baining Guo	

**Part III Performance Evaluation: Machines and Humans**

<b>21</b>	<b>Evaluation Methods in Face Recognition</b>	551
	P. Jonathon Phillips, Patrick Grother, and Ross Micheals	
<b>22</b>	<b>Dynamic Aspects of Face Processing in Humans</b>	575
	Heinrich H. Bülthoff, Douglas W. Cunningham, and Christian Wallraven	
<b>23</b>	<b>Face Recognition by Humans and Machines</b>	597
	Alice J. O'Toole	

**Part IV Face Recognition Applications**

<b>24</b>	<b>Face Recognition Applications</b>	617
	Thomas Huang, Ziyou Xiong, and Zhenqiu Zhang	
<b>25</b>	<b>Large Scale Database Search</b>	639
	Michael Brauckmann and Christoph Busch	

Contents	ix
<b>26 Face Recognition in Forensic Science . . . . .</b>	655
Nicole A. Spaun	
<b>27 Privacy Protection and Face Recognition . . . . .</b>	671
Andrew W. Senior and Sharathchandra Pankanti	
<b>Index . . . . .</b>	693



# Contributors

**Mongi Abidi** Imaging, Robotics, and Intelligent Systems Lab, University of Tennessee, Knoxville, TN 37996, USA, [abidi@utk.edu](mailto:abidi@utk.edu)

**Jörgen Ahlberg** Division of Information Systems, Swedish Defence Research Agency (FOI), P.O. Box 1165, 583 34 Linköping, Sweden, [jorahl@foi.se](mailto:jorahl@foi.se)

**Brian Amberg** Department of Mathematics and Computer Science, University of Basel, Bernoullistrasse 16, 4056 Basel, Switzerland, [brian.amberg@unibas.ch](mailto:brian.amberg@unibas.ch)

**Heinrich H. Bülthoff** Max Planck Institute for Biological Cybernetics, Speemannstrasse 38, 72076 Tübingen, Germany, [heinrich.buelthoff@tuebingen.mpg.de](mailto:heinrich.buelthoff@tuebingen.mpg.de); Department of Brain and Cognitive Engineering, Korea University, Seoul, Korea

**Simon Baker** Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA, [simonb@cs.cmu.edu](mailto:simonb@cs.cmu.edu)

**Ronen Basri** The Weizmann Institute of Science, Rehovot 76100, Israel, [ronen.basri@weizmann.ac.il](mailto:ronen.basri@weizmann.ac.il)

**Wei Bian** Centre for Quantum Computation & Intelligence Systems, FEIT, University of Technology, Sydney, NSW 2007, Australia, [wei.bian@student.uts.edu.au](mailto:wei.bian@student.uts.edu.au)

**Volker Blanz** Universität Siegen, Hölderlinstrasse 3, 57068 Siegen, Germany, [blanz@mpi-sb.mpg.de](mailto:blanz@mpi-sb.mpg.de)

**Michael Brauckmann** L-1 Identity Solutions AG, Bochum, Germany, [MBrauckmann@l1id.com](mailto:MBrauckmann@l1id.com)

**Christoph Busch** Hochschule Darmstadt/Fraunhofer IGD, Darmstadt, Germany, [christoph.busch@igd.fraunhofer.de](mailto:christoph.busch@igd.fraunhofer.de)

**Hong Chang** Imaging, Robotics, and Intelligent Systems Lab, University of Tennessee, Knoxville, TN 37996, USA, [hchang2@utk.edu](mailto:hchang2@utk.edu)

**Rama Chellappa** Department of Electrical and Computer Engineering, Center for Automation Research, University of Maryland, College Park, MD 20742, USA, [rama@umiacs.umd.edu](mailto:rama@umiacs.umd.edu)

**D. Chu** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA

**Jeffrey F. Cohn** Department of Psychology, University of Pittsburgh, Pittsburgh, PA 15260, USA, [jeffcohn@pitt.edu](mailto:jeffcohn@pitt.edu)

**Tim Cootes** Imaging Science and Biomedical Engineering, University of Manchester, Manchester, UK, [t.cootes@man.ac.uk](mailto:t.cootes@man.ac.uk)

**Douglas W. Cunningham** Max Planck Institute for Biological Cybernetics, Speemannstrasse 38, 72076 Tübingen, Germany, [douglas.cunningham@tu-cottbus.de](mailto:douglas.cunningham@tu-cottbus.de); Brandenburg Technical University, 03046 Cottbus, Germany

**Xiaoqing Ding** State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China, [dingxq@tsinghua.edu.cn](mailto:dingxq@tsinghua.edu.cn)

**Ming Du** Department of Electrical and Computer Engineering, Center for Automation Research, University of Maryland, College Park, MD 20742, USA, [mingdu@umiacs.umd.edu](mailto:mingdu@umiacs.umd.edu)

**E. Efraty** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA

**Ralph Gross** Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA, [rgross@cs.cmu.edu](mailto:rgross@cs.cmu.edu)

**Patrick Grother** National Institute of Standards and Technology, Gaithersburg, MD 20899, USA, [pgrother@nist.gov](mailto:pgrother@nist.gov)

**Baining Guo** Microsoft Research Asia, Beijing 100080, China, [bainguo@microsoft.com](mailto:bainguo@microsoft.com)

**Abdenour Hadid** Machine Vision Group, Department of Electrical and Information Engineering, University of Oulu, P.O. Box 4500, 90014 Oulu, Finland, [hadid@ee.oulu.fi](mailto:hadid@ee.oulu.fi)

**Thomas Huang** University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, [huang@ifp.uiuc.edu](mailto:huang@ifp.uiuc.edu)

**David Jacobs** University of Maryland, College Park, MD 20742, USA, [djacobs@umiacs.umd.edu](mailto:djacobs@umiacs.umd.edu)

**Anil K. Jain** Michigan State University, East Lansing, MI 48824, USA, [jain@cse.msu.edu](mailto:jain@cse.msu.edu)

**I.A. Kakadiaris** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA, [ioannisk@grip.cis.upenn.edu](mailto:ioannisk@grip.cis.upenn.edu)

**Takeo Kanade** Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA, [tk@cs.cmu.edu](mailto:tk@cs.cmu.edu)

**Reinhard Knothe** Department of Mathematics and Computer Science, University of Basel, Bernoullistrasse 16, 4056 Basel, Switzerland, [reinhard.knothe@unibas.ch](mailto:reinhard.knothe@unibas.ch)

**Andreas Koschan** Imaging, Robotics, and Intelligent Systems Lab, University of Tennessee, Knoxville, TN 37996, USA, [akoschan@utk.edu](mailto:akoschan@utk.edu)

**Joni-Kristian Kämäriäinen** Machine Vision and Pattern Recognition Laboratory, Lappeenranta University of Technology, Lappeenranta, Finland,  
[Joni.Kamarainen@lut.fi](mailto:Joni.Kamarainen@lut.fi)

**Stan Z. Li** Center for Biometrics and Security Research & National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, [szli@cbsr.ia.ac.cn](mailto:szli@cbsr.ia.ac.cn)

**Xiaoming Liu** Visualization and Computer Vision Lab, GE Global Research, Niskayuna, NY 12309, USA, [liux@ge.com](mailto:liux@ge.com)

**Zicheng Liu** Microsoft Research, Redmond, WA 98052, USA,  
[zliu@microsoft.com](mailto:zliu@microsoft.com)

**J. Birgitta Martinkauppi** Department of Electrical Engineering and Automation, University of Vaasa, Wolffintie 34, 65101 Vaasa, Finland, [birmar@uwasa.fi](mailto:birmar@uwasa.fi)

**Iain Matthews** Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA, [iainm@cs.cmu.edu](mailto:iainm@cs.cmu.edu)

**Ross Micheals** National Institute of Standards and Technology, Gaithersburg, MD 20899, USA, [rossm@nist.gov](mailto:rossm@nist.gov)

**Baback Moghaddam** Mitsubishi Electric Research Labs, Cambridge, MA 02139, USA, [baback@merl.com](mailto:baback@merl.com)

**Alice J. O'Toole** School of Behavioral and Brain Sciences, The University of Texas at Dallas, 800 W. Campbell Rd., Richardson, TX 75083-0688, USA, [otoole@utdallas.edu](mailto:otoole@utdallas.edu)

**Igor S. Pandzic** Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, 10000 Zagreb, Croatia, [igor.pandzic@fer.hr](mailto:igor.pandzic@fer.hr)

**Sharathchandra Pankanti** IBM Research, Yorktown Heights, NY 10598, USA, [sharat@us.ibm.com](mailto:sharat@us.ibm.com)

**Unsang Park** Michigan State University, East Lansing, MI 48824, USA, [parkunsa@cse.msu.edu](mailto:parkunsa@cse.msu.edu)

**G. Passalis** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA; Computer Graphics Laboratory, Department of Informatics and Telecommunications, University of Athens, Ilisia 15784, Greece

**P. Perakis** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA; Computer Graphics Labora-

tory, Department of Informatics and Telecommunications, University of Athens, Ilisia 15784, Greece

**Vladimir Petrović** Imaging Science and Biomedical Engineering, University of Manchester, Manchester, UK

**P. Jonathon Phillips** National Institute of Standards and Technology, Gaithersburg, MD 20899, USA, [jonathon@nist.gov](mailto:jonathon@nist.gov)

**Matti Pietikäinen** Machine Vision Group, Department of Electrical and Information Engineering, University of Oulu, P.O. Box 4500, 90014 Oulu, Finland, [mkp@ee.oulu.fi](mailto:mkp@ee.oulu.fi)

**Sami Romdhani** Department of Mathematics and Computer Science, University of Basel, Bernoullistrasse 16, 4056 Basel, Switzerland, [sami.romdhani@unibas.ch](mailto:sami.romdhani@unibas.ch)

**Andrew W. Senior** Google Research, New York, NY 10011, USA, [andrewsenior@google.com](mailto:andrewsenior@google.com)

**S. Shah** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA

**Gregory Shakhnarovich** Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, MA 02139, USA, [gregory@ai.mit.edu](mailto:gregory@ai.mit.edu)

**Nicole A. Spaun** Forensic Audio, Video and Image Analysis Unit, Federal Bureau of Investigation, Quantico, VA, USA, [Nicole.Spaun@us.army.mil](mailto:Nicole.Spaun@us.army.mil); United States Army Europe Headquarters, Heidelberg, Germany; USAREUR, CMR 420, Box 2872, APO AE 09036, USA

**Dacheng Tao** Centre for Quantum Computation & Intelligence Systems, FEIT, University of Technology, Sydney, NSW 2007, Australia, [dacheng.tao@uts.edu.au](mailto:dacheng.tao@uts.edu.au)

**Chris Taylor** Imaging Science and Biomedical Engineering, University of Manchester, Manchester, UK

**T. Theoharis** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA; Computer Graphics Laboratory, Department of Informatics and Telecommunications, University of Athens, Ilisia 15784, Greece

**Yingli Tian** Department of Electrical Engineering, The City College of New York, New York, NY 10031, USA, [ytian@ccny.cuny.edu](mailto:ytian@ccny.cuny.edu)

**G. Toderici** Computational Biomedicine Lab, Department of Computer Science, University of Houston, Houston, TX 77204, USA

**Phil Tresadern** Imaging Science and Biomedical Engineering, University of Manchester, Manchester, UK

**Peter H. Tu** Visualization and Computer Vision Lab, GE Global Research, Niskayuna, NY 12309, USA, [tu@ge.com](mailto:tu@ge.com)

**Pavan Turaga** Department of Electrical and Computer Engineering, Center for Automation Research, University of Maryland, College Park, MD 20742, USA, [pturaga@umiacs.umd.edu](mailto:pturaga@umiacs.umd.edu)

**Thomas Vetter** Department of Mathematics and Computer Science, University of Basel, Bernoullistrasse 16, 4056 Basel, Switzerland, [thomas.vetter@unibas.ch](mailto:thomas.vetter@unibas.ch)

**Christian Wallraven** Max Planck Institute for Biological Cybernetics, Speemannstrasse 38, 72076 Tübingen, Germany, [wallraven@korea.ac.kr](mailto:wallraven@korea.ac.kr); Department of Brain and Cognitive Engineering, Korea University, Seoul, Korea

**Liting Wang** State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China, [wangltmail@tsinghua.edu.cn](mailto:wangltmail@tsinghua.edu.cn)

**Yang Wang** Carnegie Mellon University, Pittsburgh, PA 15213, USA, [wangy@cs.cmu.edu](mailto:wangy@cs.cmu.edu)

**Frederick W. Wheeler** Visualization and Computer Vision Lab, GE Global Research, Niskayuna, NY 12309, USA, [wheeler@ge.com](mailto:wheeler@ge.com)

**Jianxin Wu** School of Computer Engineering, Nanyang Technological University, Singapore, Singapore, [jxwu@ntu.edu.sg](mailto:jxwu@ntu.edu.sg)

**Ziyou Xiong** United Technologies Research Center, East Hartford, CT 06108, USA, [xiongz@utrc.utc.com](mailto:xiongz@utrc.utc.com)

**Yi Yao** Visualization and Computer Vision Lab, GE Global Research, Niskayuna, NY 12309, USA, [yi.yao@ge.com](mailto:yi.yao@ge.com)

**Dong Yi** Center for Biometrics and Security Research & National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, [dyi@cbsr.ia.ac.cn](mailto:dyi@cbsr.ia.ac.cn)

**Zhenqiu Zhang** University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, [zhang6@uiuc.edu](mailto:zhang6@uiuc.edu)

**Shaohua Kevin Zhou** Siemens Corporate Research, 755 College Road East, Princeton, NJ 08540, USA, [kzhou@scr.siemens.com](mailto:kzhou@scr.siemens.com)



# Chapter 1

## Introduction

**Stan Z. Li and Anil K. Jain**

### 1.1 Face Recognition

Face recognition is a task that humans perform routinely and effortlessly in our daily lives. Wide availability of powerful and low-cost desktop and embedded computing systems has created an enormous interest in automatic processing of digital images in a variety of applications, including biometric authentication, surveillance, human-computer interaction, and multimedia management. Research and development in automatic face recognition follows naturally.

Face recognition has several advantages over other biometric modalities such as fingerprint and iris: besides being natural and nonintrusive, the most important advantage of face is that it can be captured at a distance and in a covert manner. Among the six biometric attributes considered by Hietmeyer [16], facial features scored the highest compatibility in a Machine Readable Travel Documents (MRTD) [27] system based on a number of evaluation factors, such as enrollment, renewal, machine requirements, and public perception, shown in Fig. 1.1. Face recognition, as one of the major biometric technologies, has become increasingly important owing to rapid advances in image capture devices (surveillance cameras, camera in mobile phones), availability of huge amounts of face images on the Web, and increased demands for higher security.

The first automated face recognition system was developed by Takeo Kanade in his Ph.D. thesis work [18] in 1973. There was a dormant period in automatic face recognition until the work by Sirovich and Kirby [19, 38] on a low dimen-

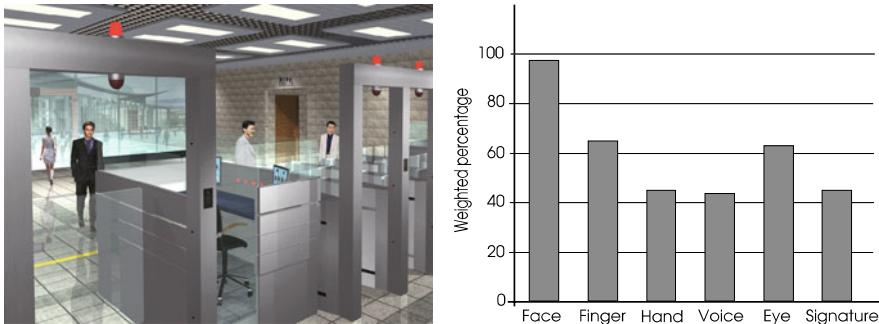
---

S.Z. Li (✉)

Center for Biometrics and Security Research & National Laboratory of Pattern Recognition,  
Institute of Automation, Chinese Academy of Sciences, Beijing, China  
e-mail: [szli@cbsr.ia.ac.cn](mailto:szli@cbsr.ia.ac.cn)

A.K. Jain

Michigan State University, East Lansing, MI 48824, USA  
e-mail: [jain@cse.msu.edu](mailto:jain@cse.msu.edu)



**Fig. 1.1** A scenario of using biometric MRTD systems for passport control (*left*), and a comparison of various biometric traits based on MRTD compatibility (*right*, from Hietmeyer [16] with permission)

sional face representation, derived using the Karhunen–Loeve transform or Principal Component Analysis (PCA). It is the pioneering work of Turk and Pentland on Eigenface [42] that reinvigorated face recognition research. Other major milestones in face recognition include: the Fisherface method [3, 12], which applied Linear Discriminant Analysis (LDA) after a PCA step to achieve higher accuracy; the use of local filters such as Gabor jets [21, 45] to provide more effective facial features; and the design of the AdaBoost learning based cascade classifier architecture for real time face detection [44].

Face recognition technology is now significantly advanced since the time when the Eigenface method was proposed. In the constrained situations, for example where lighting, pose, stand-off, facial wear, and facial expression can be controlled, automated face recognition can surpass human recognition performance, especially when the database (gallery) contains a large number of faces.<sup>1</sup> However, automatic face recognition still faces many challenges when face images are acquired under unconstrained environments. In the following sections, we give a brief overview of the face recognition process, analyze technical challenges, propose possible solutions, and describe state-of-the-art performance.

This chapter provides an introduction to face recognition research. Main steps of face recognition processing are described. Face detection and recognition problems are explained from a face subspace viewpoint. Technology challenges are identified and possible strategies for solving some of the problems are suggested.

## 1.2 Categorization

As a biometric system, a face recognition system operates in either or both of two modes: (1) face verification (or authentication), and (2) face identification (or recognition). Face verification involves a one-to-one match that compares a query face

<sup>1</sup>Most individuals can identify only a few thousand people in real life.

image against an enrollment face image whose identity is being claimed. Person verification for self-serviced immigration clearance using E-passport is one typical application.

Face identification involves one-to-many matching that compares a query face against multiple faces in the enrollment database to associate the identity of the query face to one of those in the database. In some identification applications, one just needs to find the most similar face. In a watchlist check or face identification in surveillance video, the requirement is more than finding most similar faces; a confidence level threshold is specified and all those faces whose similarity score is above the threshold are reported.

The performance of a face recognition system largely depends on a variety of factors such as illumination, facial pose, expression, age span, hair, facial wear, and motion. Based on these factors, face recognition applications may be divided into two broad categories in terms of a user's cooperation: (1) *cooperative* user scenarios and (2) *noncooperative* user scenarios.

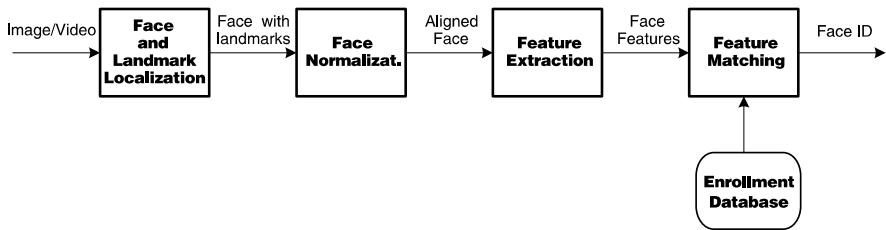
The cooperative case is encountered in applications such as computer login, physical access control, and e-passport, where the user is willing to be cooperative by presenting his/her face in a proper way (for example, in a frontal pose with neutral expression and eyes open) in order to be granted the access or privilege.

In the noncooperative case, which is typical in surveillance applications, the user is unaware of being identified. In terms of distance between the face and the camera, near field face recognition (less than 1 m) for cooperative applications (e.g., access control) is the least difficult problem, whereas far field noncooperative applications (e.g., watchlist identification) in surveillance video is the most challenging.

Applications in-between the above two categories can also be foreseen. For example, in face-based access control at a distance, the user is willing to be cooperative but he is unable to present the face in a favorable condition with respect to the camera. This may present challenges to the system even though such cases are still easier than identifying the identity of the face of a subject who is not cooperative. However, in almost all of the cases, ambient illumination is the foremost challenge for most face recognition applications.

### 1.3 Processing Workflow

Face recognition is a visual pattern recognition problem, where the face, represented as a three-dimensional object that is subject to varying illumination, pose, expression, and other factors, needs to be identified based on acquired images. While two-dimensional face images are commonly used in most applications, certain applications requiring higher levels of security demand the use of three-dimensional (depth or range) images or optical images beyond the visual spectrum. A face recognition system generally consists of four modules as depicted in Fig. 1.2: face localization, normalization, feature extraction, and matching. These modules are explained below.



**Fig. 1.2** Depiction of face recognition processing flow

*Face detection* segments the face area from the background. In the case of video, the detected faces may need to be tracked across multiple frames using a face tracking component. While face detection provides a coarse estimate of the location and scale of the face, *face landmarking* localizes facial landmarks (e.g., eyes, nose, mouth, and facial outline). This may be accomplished by a landmarking module or face alignment module.

*Face normalization* is performed to normalize the face geometrically and photometrically. This is necessary because state-of-the-art recognition methods are expected to recognize face images with varying pose and illumination. The geometrical normalization process transforms the face into a standard frame by face cropping. Warping or morphing may be used for more elaborate geometric normalization. The photometric normalization process normalizes the face based on properties such as illumination and gray scale.

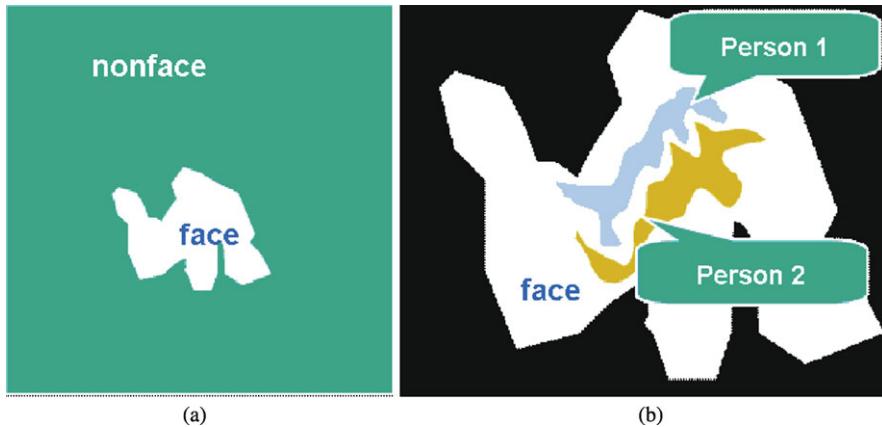
*Face feature extraction* is performed on the normalized face to extract salient information that is useful for distinguishing faces of different persons and is robust with respect to the geometric and photometric variations. The extracted face features are used for face matching.

In *face matching* the extracted features from the input face are matched against one or many of the enrolled faces in the database. The matcher outputs ‘yes’ or ‘no’ for 1:1 verification; for 1:N identification, the output is the identity of the input face when the top match is found with sufficient confidence or unknown when the top match score is below a threshold. The main challenge in this stage of face recognition is to find a suitable similarity metric for comparing facial features.

The accuracy of face recognition systems highly depends on the features that are extracted to represent the face which, in turn, depend on correct face localization and normalization. While face recognition still remains a challenging pattern recognition problem, it may be analyzed from the viewpoint of face subspaces or manifolds, as follows.

## 1.4 Face Subspace

Although face recognition technology has significantly improved and can now be successfully performed in “real-time” for images and videos captured under favorable (constrained) situations, face recognition is still a difficult endeavor, especially



**Fig. 1.3** Face subspace or manifolds. **a** Face versus nonface manifolds. **b** Face manifolds of different individuals

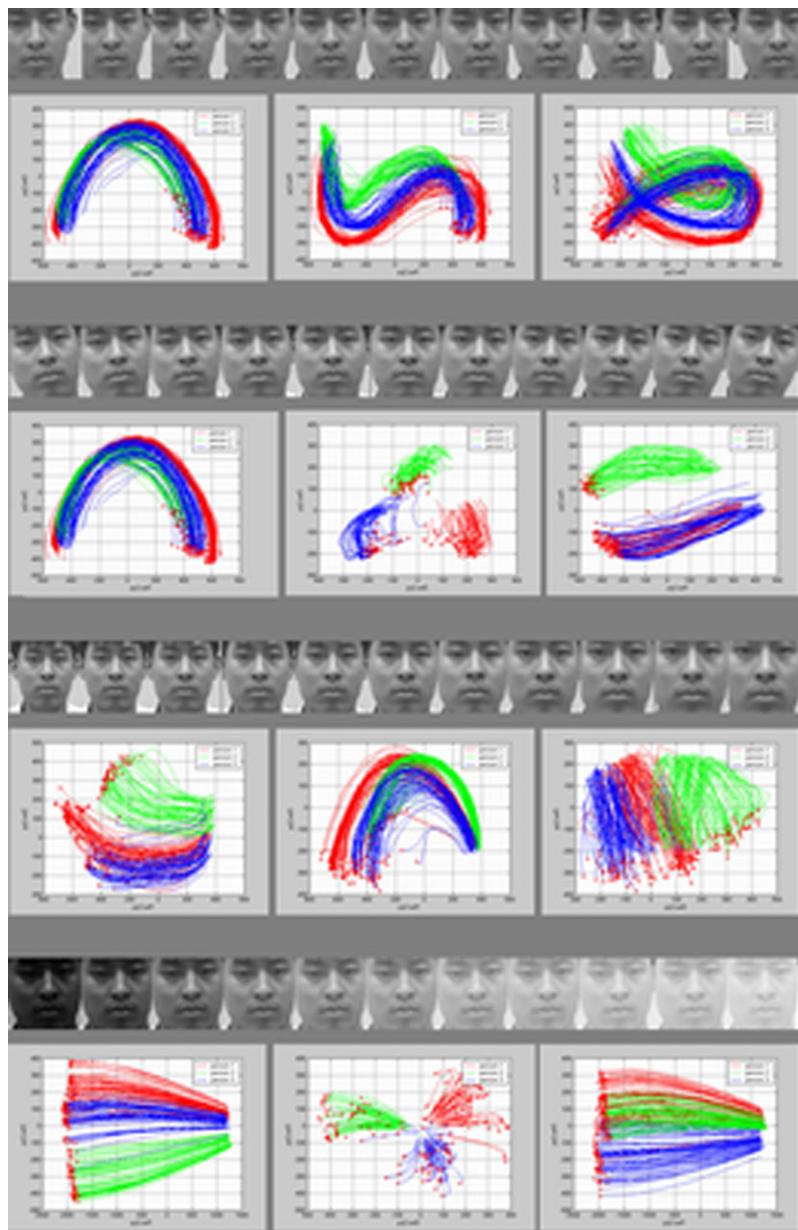
for unconstrained tasks where viewpoint, illumination, expression, occlusion, and facial accessories can vary considerably. This can be illustrated from face subspace or manifold viewpoint.

Subspace analysis techniques for face recognition are based on the fact that a class of patterns of interest, such as the face, resides in a subspace of the input image space. For example, a  $64 \times 64$  8-bit image with 4096 pixels can express a large number of pattern classes, such as trees, houses, and faces. However, among the  $256^{4096} > 10^{9864}$  possible “configurations,” only a tiny fraction correspond to faces. Therefore, the pixel-based image representation is highly redundant, and the dimensionality of this representation could be greatly reduced when only the face patterns are of interest.

The eigenface or PCA method [19, 42] derives a small number (typically 40 or lower) of principal components or eigenfaces from a set of training face images. Given the eigenfaces as basis for a face subspace, a face image is compactly represented by a low dimensional feature vector and a face can be reconstructed as a linear combination of the eigenfaces. The use of subspace modeling techniques has significantly advanced the face recognition technology.

The manifold or distribution of all the faces accounts for variations in facial appearance whereas the nonface manifold accounts for all objects other than the faces. If we examine these manifolds in the image space, we find them highly nonlinear and nonconvex [5, 41]. Figure 1.3(a) illustrates face versus nonface manifolds and Fig. 1.3(b) illustrates the manifolds of two individuals in the entire face manifold. Face detection can be considered as a task of distinguishing between the face and nonface manifolds in the image (subwindow) space and face recognition can be considered as a task of distinguishing between faces of different individuals in the face manifold.

Figure 1.4 further demonstrates the nonlinearity and nonconvexity of face manifolds in a PCA subspace spanned by the first three principal components, where the



**Fig. 1.4** Nonlinearity and nonconvexity of face manifolds under (from top to bottom) translation, rotation, scaling, and Gamma transformations

plots are drawn from real face image data. Each plot depicts the manifolds of three individuals (in three colors). The data consists of 64 frontal face images for each

individual. A transform (horizontal transform, in-plane rotation, size scaling, and gamma transform for the 4 groups, respectively) is performed on each face image with 11 gradually varying parameters, producing 11 transformed face images; each transformed image is cropped to contain only the face region; the 11 cropped face images form a sequence. A curve in this figure represents such a sequence in the PCA space, and so there are 64 curves for each individual. The three-dimensional (3D) PCA space is projected on three different 2D spaces (planes). We can observe the nonlinearity of the trajectories.

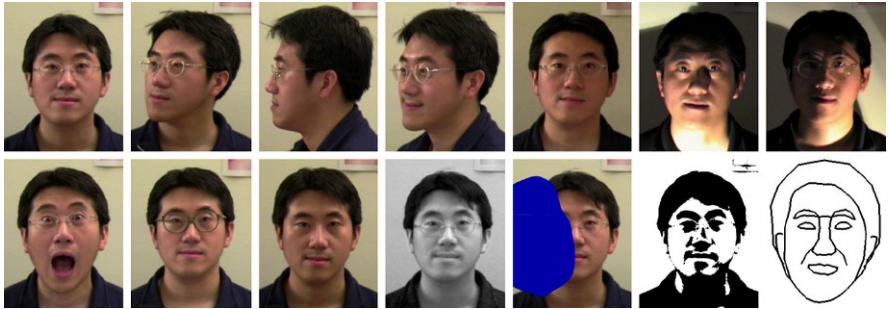
The following observations can be drawn based on Fig. 1.4. First, while this example is demonstrated in the PCA space, more complex (nonlinear and nonconvex) trajectories are expected in the original image space. Second, although these face images have been subjected to geometric transformations in the 2D plane and pointwise lighting (gamma) changes, more significant complexity of trajectories is expected for geometric transformations in 3D space (for example, out-of-plane head rotations) and ambient lights.

## 1.5 Technology Challenges

As shown in Fig. 1.3, the problem of face detection is highly nonlinear and nonconvex, even more so for face matching. Face recognition evaluation reports, for example Face Recognition Technology (FERET) [34], Face Recognition Vendor Test (FRVT) [31] and other independent studies, indicate that the performance of many state-of-the-art face recognition methods deteriorates with changes in lighting, pose, and other factors [8, 43, 50]. The key technical challenges in automatic face recognition are summarized below.

**Large Variability in Facial Appearance** Whereas shape and reflectance are intrinsic properties of a face, the appearance (i.e., the texture) of a face is also influenced by several other factors, including the facial pose (or, equivalently, camera viewpoint), illumination, and facial expression. Figure 1.5 shows an example of large intra-subject variations caused by these factors. Aging is also an important factor that leads to an increase in the intra-subject variations especially in applications requiring duplication of government issued photo ID documents (e.g., driver licenses and passports). In addition to these, various imaging parameters, such as aperture, exposure time, lens aberrations, and sensor spectral response also increase intra-subject variations. Face-based person identification is further complicated by possible small inter-subject variations (Fig. 1.6). All these factors are confounded in the image data, so “the variations between the images of the same face due to illumination and viewing direction are almost always larger than the image variation due to change in face identity” [30]. This variability makes it difficult to extract the intrinsic information about the face identity from a facial image.

**Complex Nonlinear Manifolds** As illustrated above, the entire face manifold is highly nonconvex, and so is the face manifold of any individual under various



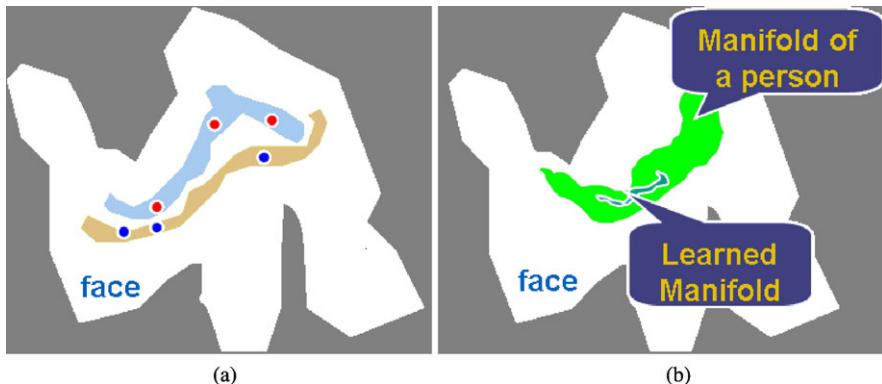
**Fig. 1.5** Intra-subject variations in pose, illumination, expression, occlusion, accessories (e.g., glasses), color, and brightness. (Courtesy of Rein-Lien Hsu [17])



**Fig. 1.6** Similarity of frontal faces between **a** twins (downloaded from [www.marykateandashley.com](http://www.marykateandashley.com)); and **b** a father and his son (downloaded from BBC news, [news.bbc.co.uk](http://news.bbc.co.uk))

changes. Linear methods such as PCA [19, 42], independent component analysis (ICA) [2], and linear discriminant analysis (LDA) [3]) project the data linearly from a high-dimensional space (for example, the image space) to a low-dimensional subspace. As such, they are unable to preserve the nonconvex variations of face manifolds necessary to differentiate among individuals. In a linear subspace, Euclidean distance and, more generally, the Mahalanobis distance do not perform well for discriminating between face and nonface manifolds and between manifolds of different individuals (Fig. 1.7(a)). This limits the power of the linear methods to achieve highly accurate face detection and recognition in many practical scenarios.

**High Dimensionality and Small Sample Size** Another challenge in face recognition is the generalization ability, which is illustrated in Fig. 1.7(b). The figure depicts a canonical face image of size  $112 \times 92$  which resides in a 10,304-dimensional feature space. The number of example face images per person (typically fewer than 10, and sometimes just one) available for learning the manifold is usually much smaller



**Fig. 1.7** Challenges in face recognition from subspace viewpoint. **a** Euclidean distance is unable to differentiate between individuals. When using Euclidean distance, an inter-person distance can be smaller than an intra-person distance. **b** The learned manifold or classifier is unable to characterize (i.e., generalize) unseen images of the same face

than the dimensionality of the image space; a system trained on a small number of examples may not generalize well to unseen instances of the face.

## 1.6 Solution Strategies

There are two strategies for tackling the challenges outlined in Sect. 1.5: (i) extract invariant and discriminative face features, and (ii) construct a robust face classifier. A set of features, constituting a feature space, is deemed to be good if the face manifolds are simple (i.e., less nonlinear and nonconvex). This requires two stages of processing: (1) normalizing face images geometrically and photometrically (for example, using geometric warping into a standard frame and photometric illumination correction) and (2) extracting features in the normalized images, such as using Gabor wavelets and LBP (local binary pattern), that are stable with respect to possible geometric and photometric variations.

A powerful classification engine is still necessary to deal with difficult nonlinear classification and regression problems in the constructed feature space. This is because the normalization and feature extraction cannot solve the problems of nonlinearity and nonconvexity. Learning methods are useful tools to find good features and build powerful robust classifiers based on these features. The two stages of processing may be designed jointly using learning methods.

In the early development of face recognition [6, 13, 18, 36], geometric facial features such as eyes, nose, mouth, and chin were explicitly used. Properties of the features and relations (e.g., areas, distances, angles) between the features were used as descriptors for face recognition. Advantages of this approach include economy and efficiency when achieving data reduction and insensitivity to variations in illumination and viewpoint. However, facial feature detection and measurement

techniques developed to date are not sufficiently reliable for the geometric feature-based recognition [9]. Further, geometric properties alone are inadequate for face recognition because rich information contained in the facial texture or appearance is not utilized. These are the main reasons why early feature-based techniques were not effective.

Statistical learning methods are the mainstream approach that has been used in building current face recognition systems. Effective features and classifiers are learned from training data (appearance images or features extracted therefrom). During the learning, both prior knowledge about face(s) and variations encountered in the training data are taken into consideration. The appearance-based approach, such as PCA [42] and LDA [3] based methods, has significantly advanced face recognition technology. Such an approach generally operates directly on an image-based representation (i.e., array of pixel intensities). It extracts features in a subspace derived from training images. Using PCA, an “optimal” face subspace is constructed to represent only the face object; using LDA, a discriminant subspace is constructed to distinguish faces of different persons. It is now well known that LDA-based methods generally yields better results than PCA-based methods [3].

These linear, holistic appearance-based methods encode prior knowledge contained in the training data and avoid instability of manual selection and tuning needed in the early geometric feature-based methods. However, they are not effective in describing local variations in the face appearance and are unable to capture subtleties of face subspaces: protrusions of nonconvex manifolds may be smoothed out and concavities may be filled in, thereby loosing useful information. Note that the appearance-based methods require that the face images be properly aligned, typically based on the eye locations.

Nonlinear subspace methods use nonlinear transforms to convert a face image into a feature vector in a discriminative feature space. Kernel PCA [37] and kernel LDA [29] use kernel tricks to map the original data into a high-dimension space to make the data separable. Manifold learning, which assumes that face images occupy a low-dimensional manifold in the original space, attempts to model such manifolds. These include ISOMAP [39], LLE [35], and LPP [15]. Although these methods achieve good performance on the training data, they tend to overfit and hence do not generalize well to unseen data.

The most successful approach to date for handling the nonconvex face distribution works with local appearance-based features extracted using appropriate image filters. This is advantageous in that distributions of face images in local feature space are less affected by the changes in facial appearance. Early work in this direction included local features analysis (LFA) [33] and Gabor wavelet-based features [21, 45]. Current methods are based on local binary pattern (LBP) [1] and many variants (for example ordinal feature [23], Scale-Invariant Feature Transform (SIFT) [26], and Histogram of Oriented Gradients (HOG) [10]). While these features are general-purpose and can be extracted from arbitrary images, face-specific local filters may be learned from images [7, 20].

A large number of local features can be generated by varying parameters associated with the position, scale, and orientation of the filters. For example, more than

400 000 local appearance features can be generated when an image of size  $100 \times 100$  is filtered with Gabor filters with five different scales and eight different orientation for all pixel positions. While some of these features are useful for face recognition, others may be less useful or may even degrade the recognition performance. Boosting based methods have been implemented to select good local features [46, 48, 49]. A discriminant analysis step can be applied to further transform the space of the selected local features to discriminative subspace of a lower dimensionality to achieve better face classification [22, 24, 25]. This leads to a framework for learning both effective features and powerful classifiers.

There have been only a few studies reported on face recognition at a distance. These approaches can be essentially categorized into two groups: (i) generating a super resolution face image from the given low resolution image [11, 32] and (ii) acquiring high resolution face image using a special camera system (e.g., a high resolution camera or a PTZ camera) [4, 14, 28, 40, 47].

The availability of high resolution face images (i.e., tens of megapixels per image) provides new opportunities in facial feature representation and matching. In the 2006 Face Recognition Vendor Test (FRVT) [31], the best face matching accuracies were obtained from the high resolution 2D images or 3D images. This underlines the importance of developing advanced sensors as well as robust feature extraction and matching algorithms in achieving high face recognition accuracy. The increasing popularity of infrared cameras also supports the importance of sensing techniques.

## 1.7 Current Status

For cooperative scenarios, frontal face detection and tracking in normal lighting environment is a reasonably well-solved problem. Assuming the face is captured with sufficient image resolution, 1:1 face verification also works satisfactorily well for cooperative frontal faces. Figure 1.8 illustrates an application of face verification at the 2008 Beijing Olympic Games. This system verifies the identity of a ticket holder (spectator) at entrances to the National Stadium (Bird's Nest). Each ticket is associated with a unique ID number, and the ticket holder is required to submit his registration form with a two-inch ID/passport photo attached. The face photo is scanned into the system. At the entrance, the ticket is read in by an RFID reader, and the face image is captured using a video camera, which is compared with the enrollment photo scan, and the verification result is produced.

A novel solution to deal with uncontrolled illumination is to use active near infrared (NIR) face imaging to control the illumination direction and the strength. This enables the system to achieve high face recognition accuracy. The NIR face recognition technology has been in use at China–Hong Kong border<sup>2</sup> for self-service immigration clearance since 2005 (see Fig. 1.8).

---

<sup>2</sup>The Shenzhen (China)–Hong Kong border is the world's largest border crossing point, with more than 400 000 crossings every day.



**Fig. 1.8** 1:1 Face verification used at the 2008 Beijing Olympic Games, and 1:NIR face verification used at the China–Hong Kong border control since 2005



**Fig. 1.9** An embedded NIR face recognition system for access control in 1:N identification mode and watch-list face surveillance and identification at subways

One-to-many face identification using the conventional, visible band face images has not yet met the accuracy requirements of practical applications even for cooperative scenarios. The main problem is the uncontrolled ambient illumination. The NIR face recognition provides a good solution, even for 1:N identification. Embedded NIR face recognition based access control products (Fig. 1.9) have been on the market since 2008.

Face recognition in noncooperative scenarios, such as watch-list identification, remains a challenging task. Major problems include pose, illumination, and motion blur. Because of growing emphasis on security, there have been several watch-list identification application trials. On the right of Fig. 1.9, it shows a snapshot of 1:N watch-list face surveillance and identification at a Beijing Municipal Subways station, aimed at identifying suspects in the crowd. CCTV cameras are mounted at the subway entrances and exits, in such a way that images of frontal faces are more likely to be captured. The best system could achieve a recognition rate of up to 60% at a FAR = 0.1%.

## 1.8 Summary

Face recognition technology has made impressive gains, but it is still not able to meet the accuracy requirements of many applications. A sustained and collaborative effort is needed to address many of the open problems in face recognition.

## References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face recognition with local binary patterns. In: Proceedings of the European Conference on Computer Vision, pp. 469–481. Prague, Czech Republic (2004)
2. Bartlett, M.S., Lades, H.M., Sejnowski, T.J.: Independent component representations for face recognition. In: Proceedings of the SPIE, Conference on Human Vision and Electronic Imaging III, vol. 3299, pp. 528–539 (1998)
3. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 711–720 (1997)
4. Bernardino, K., v. d. Camp, F., Stiefelhagen, R.: Automatic person detection and tracking using fuzzy controlled active cameras. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
5. Bichsel, M., Pentland, A.P.: Human face recognition and the face image set's topology. *CVGIP, Image Underst.* **59**, 254–261 (1994)
6. Brunelli, R., Poggio, T.: Face recognition: Features versus templates. *IEEE Trans. Pattern Anal. Mach. Intell.* **15**(10), 1042–1052 (1993)
7. Cao, Z., Yin, Q., Tang, X., Sun, J.: Face recognition with learning-based descriptor. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2010)
8. Chellappa, R., Wilson, C., Sirohey, S.: Human and machine recognition of faces: A survey. *Proc. IEEE* **83**, 705–740 (1995)
9. Cox, I.J., Ghosn, J., Yianilos, P.: Feature-based face recognition using mixture-distance. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 209–216 (1996)
10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 886–893 (2005)
11. Dedeoglu, G., Kanade, T., August, J.: High-zoom video hallucination by exploiting spatio-temporal regularities. In: Proceedings of IEEE International Conference on Computer Vision, pp. 151–158 (2004)
12. Etemad, K., Chellappa, R.: Face recognition using discriminant eigenvectors. In: Proceedings of the International Conference on Acoustic, Speech and Signal Processing (1996)
13. Goldstein, A.J., Harmon, L.D., Lesk, A.B.: Identification of human faces. *Proc. IEEE* **59**(5), 748–760 (1971)
14. Hampapur, A., Pankanti, S., Senior, A., Tian, Y.-L., Brown, L., Bolle, R.: Face cataloger: multi-scale imaging for relating identity to location. In: Proc. IEEE Conference Advanced Video and Signal Based Surveillance, pp. 13–20 (2003)
15. He, X., Yan, S., Hu, Y., Niyogi, P., Zhang, H.: Face recognition using laplacianfaces. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(3), 328–340 (2005)
16. Hietmeyer, R.: Biometric identification promises fast and secure processing of airline passengers. *ICAO J.* **55**(9), 10–11 (2000)
17. Hsu, R.-L.: Face detection and modeling for recognition. PhD thesis, Michigan State University (2002)

18. Kanade, T.: Picture processing system by computer complex and recognition of human faces. PhD thesis, Kyoto University (1973)
19. Kirby, M., Sirovich, L.: Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(1), 103–108 (1990)
20. Kumar, R., Banerjee, A., Vemuri, B.: Volterrafaces: discriminant analysis using Volterra kernels. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 150–155 (2009)
21. Lades, M., Vorbruggen, J., Buhmann, J., Lange, J., von der Malsburg, C., Wurtz, R.P., Konnen, W.: Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. Comput.* **42**, 300–311 (1993)
22. Lei, Z., Liao, S., Pietikäinen, M., Li, S.Z.: Face recognition by exploring information jointly in space, scale and orientation. *IEEE Trans. Image Process.* **20**(1), 247–256 (2011)
23. Liao, S., Lei, Z., Zhu, X., Sun, Z., Li, S.Z., Tan, T.: Face recognition using ordinal features. In: Proceedings of IAPR International Conference on Biometrics, pp. 40–46 (2006)
24. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block local binary patterns for face recognition. In: Proceedings of IAPR International Conference on Biometrics, pp. 828–837 (2007)
25. Liu, C.: Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(5), 725–737 (2006)
26. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of IEEE International Conference on Computer Vision, p. 1150, Los Alamitos, CA (1999)
27. Machine Readable Travel Documents (MRTD). <http://www.icao.int/mrtd/overview/overview.cfm>
28. Marchesotti, L., Piva, S., Turolla, A., Minetti, D., Regazzoni, C.: Cooperative multisensor system for real-time face detection and tracking in uncontrolled conditions. In: Proceedings of SPIE Int'l Conf. Image and Video Communications and Processing (2005)
29. Mika, S., Ratsch, G., Weston, J., Schölkopf, B., Müller, K.-R.: Fisher discriminant analysis with kernels. In: Neural Networks for Signal Processing IX, pp. 41–48 (1999)
30. Moses, Y., Adini, Y., Ullman, S.: Face recognition: The problem of compensating for changes in illumination direction. In: Proceedings of the European Conference on Computer Vision, vol. A, pp. 286–296 (1994)
31. NIST: Face Recognition Vendor Tests (FRVT) (2006). <http://www.frvt.org>
32. Park, J., Lee, S.: Stepwise reconstruction of high-resolution facial image based on interpolated morphable face model. In: Proc. Int'l Conf. Audio-and Video-based Biometric Person Authentication, pp. 102–111 (2005)
33. Penev, P., Atick, J.: Local feature analysis: A general statistical theory for object representation. *Neural Syst.* **7**(3), 477–500 (1996)
34. Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.J.: The FERET evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(10), 1090–1104 (2000)
35. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**(22), 2323–2326 (2000)
36. Samal, A., Iyengar, P.A.: Automatic recognition and analysis of human faces and facial expressions: A survey. *Pattern Recognit.* **25**, 65–77 (1992)
37. Schölkopf, B., Smola, A., Müller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.* **10**, 1299–1319 (1999)
38. Sirovich, L., Kirby, M.: Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A* **4**(3), 519–524 (1987)
39. Tenenbaum, J., Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(22), 2319–2323 (2000)
40. Tistarelli, M., Li, S., Chellappa, R. (eds.): *Handbook of Remote Biometrics for Surveillance and Security*. Springer, Berlin (2009)
41. Turk, M.: A random walk through eigenspace. *IEICE Trans. Inf. Syst.* **E84-D**(12), 1586–1695 (2001)
42. Turk, M.A., Pentland, A.P.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–86 (1991)

43. Valentin, D., Abdi, H., O'Toole, A.J., Cottrell, G.W.: Connectionist models of face processing: A survey. *Pattern Recognit.* **27**(9), 1209–1230 (1994)
44. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p. 511 (2001)
45. Wiskott, L., Fellous, J., Kruger, N., v. d. Malsburg, C.: Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 775–779 (1997)
46. Yang, P., Shan, S., Gao, W., Li, S.Z., Zhang, D.: Face recognition using Ada-boosted Gabor features. In: *Proceedings of International Conference on Automatic Face and Gesture Recognition*, pp. 356–361 (2004)
47. Yao, Y., Abidi, B., Kalka, N., Schmid, N., Abidi, M.: Improving long range and high magnification face recognition: Database acquisition, evaluation, and enhancement. *Comput. Vis. Image Underst.* **111**(2), 111–125 (2008)
48. Zhang, L., Li, S.Z., Qu, Z., Huang, X.: Boosting local feature based classifiers for face recognition. In: *Proceedings of First IEEE Workshop on Face Processing in Video*, Washington, DC (2004)
49. Zhang, G., Huang, X., Li, S.Z., Wang, Y., Wu, X.: Boosting local binary pattern (LBP)-based face recognition. In: Li, S.Z., Lai, J., Tan, T., Feng, G., Wang, Y. (eds.) *Advances in Biometric Personal Authentication*, vol. 3338, pp. 180–187. Springer, Berlin (2005)
50. Zhao, W., Chellappa, R., Phillips, P., Rosenfeld, A.: Face recognition: A literature survey. *ACM Comput. Surv.* 399–458 (2003)



## **Part I**

# **Face Image Modeling and Representation**



# Chapter 2

## Face Recognition in Subspaces

Gregory Shakhnarovich and Baback Moghaddam

### 2.1 Introduction

Images of faces, represented as high-dimensional pixel arrays, often belong to a manifold of intrinsically low dimension. Face recognition, and computer vision research in general, has witnessed a growing interest in techniques that capitalize on this observation and apply algebraic and statistical tools for extraction and analysis of the underlying manifold. In this chapter, we describe in roughly chronologic order techniques that identify, parameterize, and analyze linear and nonlinear subspaces, from the original Eigenfaces technique to the recently introduced Bayesian method for probabilistic similarity analysis. We also discuss comparative experimental evaluation of some of these techniques as well as practical issues related to the application of subspace methods for varying pose, illumination, and expression.

### 2.2 Face Space and Its Dimensionality

Computer analysis of face images deals with a visual signal (light reflected off the surface of a face) that is registered by a digital sensor as an array of pixel values. The pixels may encode color or only intensity. In this chapter, we assume the latter case (i.e., gray-level imagery). After proper normalization and resizing to a fixed  $m$ -by- $n$  size, the pixel array can be represented as a point (i.e., vector) in an  $mn$ -dimensional *image space* by simply writing its pixel values in a fixed (typically raster) order. A critical issue in the analysis of such multidimensional data is the

---

G. Shakhnarovich (✉)

Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, MA 02139, USA  
e-mail: [gregory@ai.mit.edu](mailto:gregory@ai.mit.edu)

B. Moghaddam

Mitsubishi Electric Research Labs, Cambridge, MA 02139, USA  
e-mail: [baback@merl.com](mailto:baback@merl.com)

*dimensionality*, the number of coordinates necessary to specify a data point. Below we discuss the factors affecting this number in the case of face images.

### 2.2.1 Image Space Versus Face Space

To specify an arbitrary image in the image space, one needs to specify every pixel value. Thus, the “nominal” dimensionality of the space, dictated by the pixel representation, is  $mn$ , a high number even for images of modest size. Recognition methods that operate on this representation suffer from a number of potential disadvantages, most of them rooted in the so-called curse of dimensionality.

- Handling high-dimensional examples, especially in the context of similarity- and matching-based recognition, is computationally expensive.
- For parametric methods, the number of parameters one needs to estimate typically grows exponentially with the dimensionality. Often this number is much higher than the number of images available for training, making the estimation task in the image space ill-posed.
- Similarly, for nonparametric methods, the sample complexity—the number of examples needed to represent the underlying distribution of the data efficiently—is prohibitively high.

However, much of the surface of a face is smooth and has regular texture. Therefore, per-pixel sampling is in fact unnecessarily dense: The value of a pixel is typically highly correlated with the values of the surrounding pixels. Moreover, the appearance of faces is highly constrained; for example, any frontal view of a face is roughly symmetrical, has eyes on the sides, nose in the middle, and so on. A vast proportion of the points in the image space does not represent physically possible faces. Thus, the natural constraints dictate that the face images are in fact confined to a subspace referred to as the *face subspace*.

### 2.2.2 Principal Manifold and Basis Functions

It is common to model the face subspace as a (possibly disconnected) *principal manifold* embedded in the high-dimensional image space. Its *intrinsic* dimensionality is determined by the number of degrees of freedom within the face subspace; the goal of subspace analysis is to determine this number and to extract the *principal modes* of the manifold. The principal modes are computed as functions of the pixel values and referred to as *basis functions* of the principal manifold.

To make these concepts concrete, consider a straight line in  $\mathbb{R}^3$ , passing through the origin and parallel to the vector  $\mathbf{a} = [a_1, a_2, a_3]^T$ . Any point on the line can be described by three coordinates; nevertheless, the subspace that consists of all points on the line has a single degree of freedom, with the principal mode corresponding

to translation along the direction of  $\mathbf{a}$ . Consequently, representing the points in this subspace requires a single basis function:  $\phi(x_1, x_2, x_3) = \sum_{j=1}^3 a_j x_j$ . The analogy here is between the line and the face subspace and between  $\mathbb{R}^3$  and the image space.

Note that, in theory, according to the described model any face image should fall in the face subspace. In practice, owing to sensor noise, the signal usually has a nonzero component outside the face subspace. This introduces uncertainty into the model and requires algebraic and statistical techniques capable of extracting the basis functions of the principal manifold in the presence of noise. In Sect. 2.2.3, we briefly describe principal component analysis, which plays an important role in many of such techniques. For a more detailed discussion, see Gerbrands [12] and Jolliffe [17].

### 2.2.3 Principal Component Analysis

Principal component analysis (PCA) [17] is a dimensionality reduction technique based on extracting the desired number of *principal components* of the multidimensional data. The first principal component is the linear combination of the original dimensions that has the maximum variance; the  $n$ th principal component is the linear combination with the highest variance, subject to being orthogonal to the  $n - 1$  first principal components.

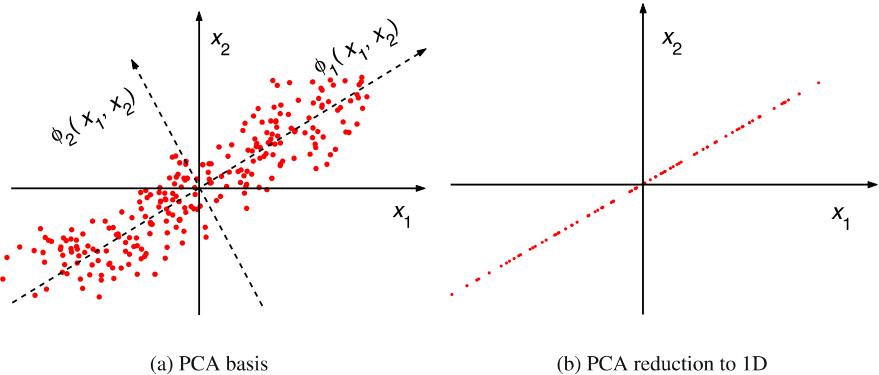
The idea of PCA is illustrated in Fig. 2.1a; the axis labeled  $\phi_1$  corresponds to the direction of maximum variance and is chosen as the first principal component. In a two-dimensional case, the second principal component is then determined uniquely by the orthogonality constraints; in a higher-dimensional space the selection process would continue, guided by the variances of the projections.

PCA is closely related to the Karhunen–Loëve Transform (KLT) [21], which was derived in the signal processing context as the orthogonal transform with the basis  $\Phi = [\phi_1, \dots, \phi_N]^T$  that for any  $k \leq N$  minimizes the average  $L_2$  reconstruction error for data points  $\mathbf{x}$

$$\varepsilon(\mathbf{x}) = \left\| \mathbf{x} - \sum_{i=1}^k (\phi_i^T \mathbf{x}) \phi_i \right\|. \quad (2.1)$$

One can show [12] that, under the assumption that the data are zero-mean, the formulations of PCA and KLT are identical. Without loss of generality, we hereafter assume that the data are indeed zero-mean; that is, the mean face  $\bar{\mathbf{x}}$  is always subtracted from the data.

The basis vectors in KLT can be calculated in the following way. Let  $\mathbf{X}$  be the  $N \times M$  data matrix whose columns  $\mathbf{x}_1, \dots, \mathbf{x}_M$  are *observations* of a signal embedded in  $\mathbb{R}^N$ ; in the context of face recognition,  $M$  is the number of available face images, and  $N = mn$  is the number of pixels in an image. The KLT basis  $\Phi$  is obtained by solving the eigenvalue problem  $\mathbf{A} = \Phi^T \Sigma \Phi$ , where  $\Sigma$  is the covariance



**Fig. 2.1** The concept of PCA/KLT. **a** Solid lines, the original basis; dashed lines, the KLT basis. The dots are selected at regularly spaced locations on a straight line rotated at  $30^\circ$  and then perturbed by isotropic 2D Gaussian noise. **b** The projection (1D reconstruction) of the data using only the first principal component

matrix of the data

$$\boldsymbol{\Sigma} = \frac{1}{M} \sum_{i=1}^M \mathbf{x}_i \mathbf{x}_i^T \quad (2.2)$$

$\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_m]^T$  is the eigenvector matrix of  $\boldsymbol{\Sigma}$ , and  $\boldsymbol{\Lambda}$  is the diagonal matrix with eigenvalues  $\lambda_1 \geq \dots \geq \lambda_N$  of  $\boldsymbol{\Sigma}$  on its main diagonal, so  $\boldsymbol{\phi}_j$  is the eigenvector corresponding to the  $j$ th largest eigenvalue. Then it can be shown that the eigenvalue  $\lambda_i$  is the variance of the data projected on  $\boldsymbol{\phi}_i$ .

Thus, to perform PCA and extract  $k$  principal components of the data, one must project the data onto  $\boldsymbol{\Phi}_k$ , the first  $k$  columns of the KLT basis  $\boldsymbol{\Phi}$ , which correspond to the  $k$  highest eigenvalues of  $\boldsymbol{\Sigma}$ . This can be seen as a linear projection  $\mathbb{R}^N \rightarrow \mathbb{R}^k$ , which retains the maximum energy (i.e., variance) of the signal. Another important property of PCA is that it *decorrelates* the data: the covariance matrix of  $\boldsymbol{\Phi}_k^T \mathbf{X}$  is always diagonal.

The main properties of PCA are summarized by the following

$$\mathbf{x} \approx \boldsymbol{\Phi}_k \mathbf{y}, \quad \boldsymbol{\Phi}_k^T \boldsymbol{\Phi}_k = \mathbf{I}, \quad E\{y_i y_j\}_{i \neq j} = 0 \quad (2.3)$$

namely, approximate reconstruction, orthonormality of the basis  $\boldsymbol{\Phi}_k$ , and decorrelated principal components  $y_i = \boldsymbol{\phi}_i^T \mathbf{x}$ , respectively. These properties are illustrated in Fig. 2.1, where PCA is successful in finding the principal manifold, and in Fig. 2.8a (see later), where it is less successful, owing to clear nonlinearity of the principal manifold.

PCA may be implemented via singular value decomposition (SVD). The SVD of an  $M \times N$  matrix  $\mathbf{X}$  ( $M \geq N$ ) is given by

$$\mathbf{X} = \mathbf{U} \mathbf{D} \mathbf{V}^T \quad (2.4)$$

where the  $M \times N$  matrix  $\mathbf{U}$  and the  $N \times N$  matrix  $\mathbf{V}$  have orthonormal columns, and the  $N \times N$  matrix  $\mathbf{D}$  has the singular values<sup>1</sup> of  $X$  on its main diagonal and zero elsewhere.

It can be shown that  $\mathbf{U} = \Phi$ , so SVD allows efficient and robust computation of PCA without the need to estimate the data covariance matrix  $\Sigma$  (2.2). When the number of examples  $M$  is much smaller than the dimension  $N$ , this is a crucial advantage.

### 2.2.4 Eigenspectrum and Dimensionality

An important largely unsolved problem in dimensionality reduction is the choice of  $k$ , the intrinsic dimensionality of the principal manifold. No analytical derivation of this number for a complex natural visual signal is available to date. To simplify this problem, it is common to assume that in the noisy embedding of the signal of interest (in our case, a point sampled from the face subspace) in a high-dimensional space, the *signal-to-noise ratio* is high. Statistically, that means that the variance of the data along the principal modes of the manifold is high compared to the variance within the complementary space.

This assumption relates to the *eigenspectrum*, the set of eigenvalues of the data covariance matrix  $\Sigma$ . Recall that the  $i$ th eigenvalue is equal to the variance along the  $i$ th principal component; thus, a reasonable algorithm for detecting  $k$  is to search for the location along the decreasing eigenspectrum where the value of  $\lambda_i$  drops significantly. A typical eigenspectrum for a face recognition problem, and the natural choice of  $k$  for such a spectrum, is shown in Fig. 2.3b (see later).

In practice, the choice of  $k$  is also guided by computational constraints, related to the cost of matching within the extracted principal manifold and the number of available face images. See Penev and Sirovich [29] as well as Sects. 2.3.2 and 2.3.4 for more discussion on this issue.

## 2.3 Linear Subspaces

Perhaps the simplest case of principal manifold analysis arises under the assumption that the principal manifold is linear. After the origin has been translated to the *mean face* (the average image in the database) by subtracting it from every image, the face subspace is a linear subspace of the image space. In this section, we describe methods that operate under this assumption and its generalization, a multilinear manifold.

---

<sup>1</sup>A singular value of a matrix  $X$  is the square root of an eigenvalue of  $XX^T$ .



**Fig. 2.2** Eigenfaces: the average face on the left, followed by seven top eigenfaces. From Turk and Pentland [36], with permission

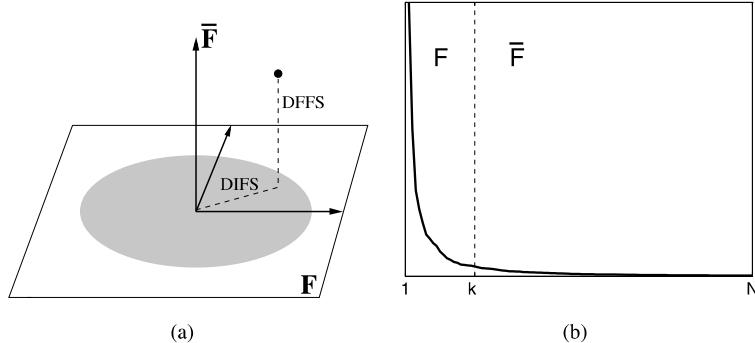
### 2.3.1 Eigenfaces and Related Techniques

In their ground-breaking work in 1990, Kirby and Sirovich [19] proposed the use of PCA for face analysis and representation. Their paper was followed by the “eigenfaces” technique by Turk and Pentland [35], the first application of PCA to face recognition. Because the basis vectors constructed by PCA had the same dimension as the input face images, they were named “eigenfaces.” Figure 2.2 shows an example of the mean face and a few of the top eigenfaces. Each face image was projected (after subtracting the mean face) into the principal subspace; the coefficients of the PCA expansion were averaged for each subject, resulting in a single  $k$ -dimensional representation of that subject. When a test image was projected into the subspace, Euclidean distances between its coefficient vector and those representing each subject were computed. Depending on the distance to the subject for which this distance would be minimized, and the PCA reconstruction error (2.1), the image was classified as belonging to one of the familiar subjects, as a new face, or as a nonface. The latter demonstrates the dual use of subspace techniques for *detection*: When the appearance of an object class (e.g., faces) is modeled by a subspace, the distance from this subspace can serve to classify an object as a member or a nonmember of the class.

### 2.3.2 Probabilistic Eigenspaces

The role of PCA in the original Eigenfaces was largely confined to dimensionality reduction. The similarity between images  $\mathbf{I}_1$  and  $\mathbf{I}_2$  was measured in terms of the Euclidean norm of the difference  $\Delta = \mathbf{I}_1 - \mathbf{I}_2$  projected to the subspace, essentially ignoring the variation modes within the subspace and outside it. This was improved in the extension of eigenfaces proposed by Moghaddam and Pentland [24, 25], which uses a *probabilistic* similarity measure based on a parametric estimate of the probability density  $p(\Delta | \Omega)$ .

A major difficulty with such estimation is that normally there are not nearly enough data to estimate the parameters of the density in a high dimensional space. Moghaddam and Pentland overcame this problem by using PCA to divide the vector space  $\mathbb{R}^N$  into two subspaces, as shown in Fig. 2.3: the principal subspace  $F$ , obtained by  $\Phi_k$  (the first  $k$  columns of  $\Phi$ ) and its orthogonal complement  $\bar{F}$  spanned by the remaining columns of  $\Phi$ . The operating assumption here is that the data have



**Fig. 2.3** **a** Decomposition of  $\mathbb{R}^N$  into the principal subspace  $F$  and its orthogonal complement  $\bar{F}$  for a Gaussian density. **b** Typical eigenvalue spectrum and its division into the two orthogonal subspaces

intrinsic dimensionality  $k$  (at most) and thus reside in  $F$ , with the exception of additive white Gaussian noise within  $\bar{F}$ . Every image can be decomposed into two orthogonal components by projection into these two spaces. Figure 2.3a shows the decomposition of  $\Delta$  into distance *within* face subspace (DIFS) and the distance *from* face subspace (DFFS). Moreover, the probability density can be decomposed into two orthogonal components.

$$P(\Delta | \Omega) = P_F(\Delta | \Omega) \cdot P_{\bar{F}}(\Delta | \Omega). \quad (2.5)$$

In the simplest case,  $P(\Delta | \Omega)$  is a Gaussian density. As derived by Moghaddam and Pentland [24], the complete likelihood estimate in this case can be written as the product of two independent marginal Gaussian densities

$$\begin{aligned} \hat{P}(\Delta | \Omega) &= \left[ \frac{\exp(-\frac{1}{2} \sum_{i=1}^k \frac{y_i^2}{\lambda_i})}{(2\pi)^{k/2} \prod_{i=1}^k \lambda_i^{1/2}} \right] \cdot \left[ \frac{\exp(-\frac{\epsilon^2(\Delta)}{2\rho})}{(2\pi\rho)^{(N-k)/2}} \right] \\ &= P_F(\Delta | \Omega) \hat{P}_{\bar{F}}(\Delta | \Omega; \rho) \end{aligned} \quad (2.6)$$

where  $P_F(\Delta | \Omega)$  is the true marginal density in  $F$ ;  $\hat{P}_{\bar{F}}(\Delta | \Omega; \rho)$  is the estimated marginal density in  $\bar{F}$ ;  $y_i = \phi_i^T \Delta$  are the principal components of  $\Delta$ ; and  $\epsilon(\Delta)$  is the PCA reconstruction error (2.1). The information-theoretical optimal value for the noise density parameter  $\rho$  is derived by minimizing the Kullback–Leibler (KL) divergence [8] and can be shown to be simply the average of the  $N - k$  smallest eigenvalues

$$\rho = \frac{1}{N - k} \sum_{i=k+1}^N \lambda_i. \quad (2.7)$$

This is a special case of the recent, more general factor analysis model called probabilistic PCA (PPCA) proposed by Tipping and Bishop [34]. In their formulation,

the above expression for  $\rho$  is the maximum-likelihood solution of a latent variable model in contrast to the minimal-divergence solution derived by Moghaddam and Pentland [24].

In practice, most of the eigenvalues in  $\bar{F}$  cannot be computed owing to insufficient data, but they can be estimated, for example, by fitting a nonlinear function to the available portion of the eigenvalue spectrum and estimating the average of the eigenvalues beyond the principal subspace. Fractal power law spectra of the form  $f^{-n}$  are thought to be typical of “natural” phenomenon and are often a good fit to the decaying nature of the eigenspectrum, as illustrated by Fig. 2.3b.

In this probabilistic framework, the recognition of a test image  $x$  is carried out in terms of computing for every database example  $x_i$  the difference  $\Delta = x - x_i$  and its decomposition into the  $F$  and  $\bar{F}$  components and then ranking the examples according to the value in (2.6).

### 2.3.3 Linear Discriminants: Fisherfaces

When substantial changes in illumination and expression are present, much of the variation in the data is due to these changes. The PCA techniques essentially select a subspace that retains most of that variation, and consequently the similarity in the face subspace is not necessarily determined by the identity.

Belhumeur et al. [2] propose to solve this problem with “Fisherfaces”, an application of Fisher’s linear discriminant (FLD). FLD selects the linear subspace  $\Phi$ , which maximizes the ratio

$$\frac{|\Phi^T S_b \Phi|}{|\Phi^T S_w \Phi|} \quad (2.8)$$

where

$$S_b = \sum_{i=1}^m N_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T$$

is the *between-class* scatter matrix, and

$$S_w = \sum_{i=1}^m \sum_{x \in X_i} (x - \bar{x}_i)(x - \bar{x}_i)^T$$

is the *within-class* scatter matrix;  $m$  is the number of subjects (classes) in the database. Intuitively, FLD finds the projection of the data in which the classes are most linearly separable. It can be shown that the dimension of  $\Phi$  is at most  $m - 1$ .<sup>2</sup>

Because in practice  $S_w$  is usually singular, the Fisherfaces algorithm first reduces the dimensionality of the data with PCA so (2.8) can be computed and then

---

<sup>2</sup>For comparison, note that the objective of PCA can be seen as maximizing the total scatter across all the images in the database.

applies FLD to further reduce the dimensionality to  $m - 1$ . The recognition is then accomplished by a NN classifier in this final subspace. The experiments reported by Belhumeur et al. [2] were performed on data sets containing frontal face images of 5 people with drastic lighting variations and another set with faces of 16 people with varying expressions and again drastic illumination changes. In all the reported experiments Fisherfaces achieve a lower error rate than eigenfaces.

### 2.3.4 Bayesian Methods

Consider now a feature space of  $\Delta$  vectors, the differences between two images ( $\Delta = \mathbf{I}_j - \mathbf{I}_k$ ). One can define two classes of facial image variations: *intrapersonal* variations  $\Omega_I$  (corresponding, for example, to different facial expressions and illuminations of the *same* individual) and *extrapersonal* variations  $\Omega_E$  (corresponding to variations between *different* individuals). The similarity measure  $S(\Delta)$  can then be expressed in terms of the intrapersonal *a posteriori* probability of  $\Delta$  belonging to  $\Omega_I$  given by the Bayes rule.

$$S(\Delta) = P(\Omega_I | \Delta) = \frac{P(\Delta | \Omega_I)P(\Omega_I)}{P(\Delta | \Omega_I)P(\Omega_I) + P(\Delta | \Omega_E)P(\Omega_E)}. \quad (2.9)$$

Note that this particular Bayesian formulation, proposed by Moghaddam et al. [27], casts the standard face recognition task (essentially an  $m$ -ary classification problem for  $m$  individuals) into a *binary* pattern classification problem with  $\Omega_I$  and  $\Omega_E$ .

The densities of both classes are modeled as high-dimensional Gaussians, using an efficient PCA-based method described in Sect. 2.3.2.

$$\begin{aligned} P(\Delta | \Omega_E) &= \frac{e^{-\frac{1}{2}\Delta^T \Sigma_E^{-1} \Delta}}{(2\pi)^{D/2} |\Sigma_E|^{1/2}}, \\ P(\Delta | \Omega_I) &= \frac{e^{-\frac{1}{2}\Delta^T \Sigma_I^{-1} \Delta}}{(2\pi)^{D/2} |\Sigma_I|^{1/2}}. \end{aligned} \quad (2.10)$$

These densities are zero-mean, because for each  $\Delta = \mathbf{I}_j - \mathbf{I}_i$  there exists a  $\mathbf{I}_i - \mathbf{I}_j$ .

By PCA, the Gaussians are known to occupy only a subspace of image space (face subspace); thus, only the top few eigenvectors of the Gaussian densities are relevant for modeling. These densities are used to evaluate the similarity in (2.9). Computing the similarity involves first subtracting a candidate image  $\mathbf{I}$  from a database example  $\mathbf{I}_j$ . The resulting  $\Delta$  image is then projected onto the eigenvectors of the extrapersonal Gaussian and also the eigenvectors of the intrapersonal Gaussian. The exponentials are computed, normalized, and then combined as in (2.9). This operation is iterated over all examples in the database, and the example that achieves the maximum score is considered the match. For large databases, such evaluations are expensive and it is desirable to simplify them by off-line transformations.

To compute the likelihoods  $P(\Delta | \Omega_I)$  and  $P(\Delta | \Omega_E)$ , the database images  $\mathbf{I}_j$  are preprocessed with *whitening* transformations [11]. Each image is converted and stored as a set of two whitened subspace coefficients:  $\mathbf{y}_{\Phi_I}$  for intrapersonal space and  $\mathbf{y}_{\Phi_E}$  for extrapersonal space

$$\mathbf{y}_{\Phi_I}^j = \mathbf{\Lambda}_I^{-\frac{1}{2}} \mathbf{V}_I \mathbf{I}_j, \quad \mathbf{y}_{\Phi_E}^j = \mathbf{\Lambda}_E^{-\frac{1}{2}} \mathbf{V}_E \mathbf{I}_j \quad (2.11)$$

where  $\mathbf{\Lambda}_X$  and  $\mathbf{V}_X$  are matrices of the largest eigenvalues and eigenvectors, respectively, of  $\Sigma_X$  ( $X$  being a substituting symbol for  $I$  or  $E$ ).

After this preprocessing, evaluating the Gaussians can be reduced to simple Euclidean distances as in (2.12). Denominators are of course precomputed. These likelihoods are evaluated and used to compute the *maximum a posteriori* (MAP) similarity  $S(\Delta)$  in (2.9). Euclidean distances are computed between the  $k_I$ -dimensional  $\mathbf{y}_{\Phi_I}$  vectors as well as the  $k_E$ -dimensional  $\mathbf{y}_{\Phi_E}$  vectors. Thus, roughly  $2 \times (k_E + k_I)$  arithmetic operations are required for each similarity computation, avoiding repeated image differencing and projections

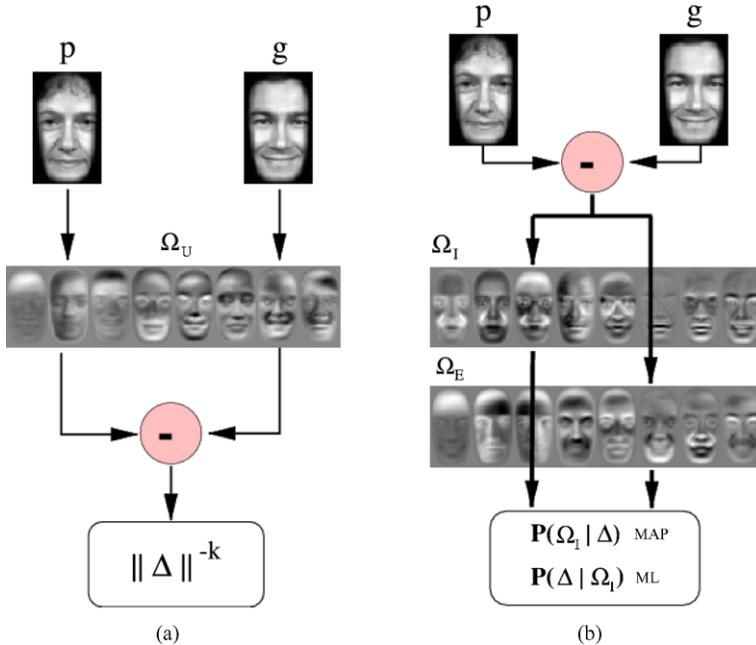
$$\begin{aligned} P(\Delta | \Omega_I) &= P(\mathbf{I} - \mathbf{I}_j | \Omega_I) = \frac{e^{-\|\mathbf{y}_{\Phi_I} - \mathbf{y}_{\Phi_I}^j\|^2/2}}{(2\pi)^{k_I/2} |\Sigma_I|^{1/2}}, \\ P(\Delta | \Omega_E) &= P(\mathbf{I} - \mathbf{I}_j | \Omega_E) = \frac{e^{-\|\mathbf{y}_{\Phi_E} - \mathbf{y}_{\Phi_E}^j\|^2/2}}{(2\pi)^{k_E/2} |\Sigma_E|^{1/2}}. \end{aligned} \quad (2.12)$$

The *maximum likelihood* (ML) similarity matching is even simpler, as only the intrapersonal class is evaluated, leading to the following modified form for the similarity measure.

$$S'(\Delta) = P(\Delta | \Omega_I) = \frac{e^{-\|\mathbf{y}_{\Phi_I} - \mathbf{y}_{\Phi_I}^j\|^2/2}}{(2\pi)^{k_I/2} |\Sigma_I|^{1/2}}. \quad (2.13)$$

The approach described above requires two projections of the difference vector  $\Delta$ , from which likelihoods can be estimated for the Bayesian similarity measure. The computation flow is illustrated in Fig. 2.4b. The projection steps are linear while the posterior computation is nonlinear. Because of the double PCA projections required, this approach has been called a “dual eigenspace” technique. Note the projection of the difference vector  $\Delta$  onto the “dual eigenfaces” ( $\Omega_I$  and  $\Omega_E$ ) for computation of the posterior in (2.9).

It is instructive to compare and contrast LDA (Fisherfaces) and the dual subspace technique by noting the similar roles of the between-class/within-class and extrapersonal/intrapersonal subspaces. One such analysis was presented by Wang and Tang [39] where PCA, LDA, and Bayesian methods were “unified” under a three-parameter subspace method. Ultimately, the optimal probabilistic justification of LDA is for the case of two Gaussian distributions of equal covariance (although LDA tends to perform rather well even when this condition is not strictly true). In contrast, the dual formulation is entirely general and probabilistic by definition, and it makes no appeals to geometry, Gaussianity, or symmetry of the underlying data



**Fig. 2.4** Signal flow diagrams for computing the similarity  $g$  between two images. **a** Original eigenfaces. **b** Bayesian similarity. The difference image is projected through both sets of (intra/extrapersonal) eigenfaces to obtain the two likelihoods

or, in fact, the two “meta classes” (intra-, and extrapersonal). These two probability distributions can take on any form (e.g., arbitrary mixture models), not just single Gaussians, although the latter case does make for easy visualization by diagonalizing the dual covariances as two sets of “eigenfaces”.

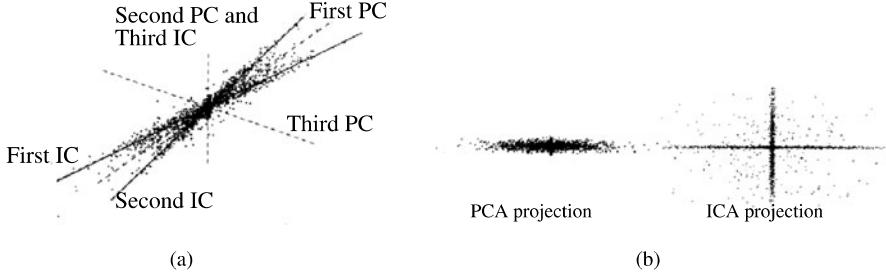
### 2.3.5 Independent Component Analysis and Source Separation

While PCA minimizes the sample covariance (second-order dependence) of the data, independent component analysis (ICA) [6, 18] minimizes higher-order dependencies as well, and the components found by ICA are designed to be non-Gaussian. Like PCA, ICA yields a linear projection  $\mathbb{R}^N \rightarrow \mathbb{R}^M$  but with different properties

$$\mathbf{x} \approx \mathbf{A}\mathbf{y}, \quad \mathbf{A}^T \mathbf{A} \neq \mathbf{I}, \quad P(\mathbf{y}) \approx \prod p(y_i) \quad (2.14)$$

that is, approximate reconstruction, *nonorthogonality* of the basis  $\mathbf{A}$ , and the near-factorization of the joint distribution  $P(\mathbf{y})$  into marginal distributions of the (non-Gaussian) ICs.

An example of ICA basis is shown in Fig. 2.5, where it is computed from a set of 3D points. The 2D subspace recovered by ICA appears to reflect the distribution



**Fig. 2.5** ICA vs. PCA decomposition of a 3D data set. **a** The bases of PCA (orthogonal) and ICA (nonorthogonal). **b** Left: the projection of the data onto the top two principal components (PCA). Right: the projection onto the top two independent components (ICA). (From Bartlett et al. [1], with permission)

of the data much better than the subspace obtained with PCA. Another example of an ICA basis is shown in Fig. 2.8b where we see two unordered nonorthogonal IC vectors, one of which is roughly aligned with the first principal component vector in Fig. 2.8a (see later), (i.e., the direction of maximum variance). Note that the actual non-Gaussianity and statistical independence achieved in this toy example are minimal at best, and so is the success of ICA in recovering the principal modes of the data.

ICA is intimately related to the *blind source separation* problem: decomposition of the input signal (image)  $\mathbf{x}$  into a linear combination (mixture) of independent source signals. Formally, the assumption is that  $\mathbf{x}^T = \mathbf{A}\mathbf{s}^T$ , with  $\mathbf{A}$  the unknown mixing matrix. ICA algorithms<sup>3</sup> try to find  $\mathbf{A}$  or the *separating matrix*  $\mathbf{W}$  such that  $\mathbf{u}^T = \mathbf{W}\mathbf{x}^T = \mathbf{W}\mathbf{A}\mathbf{s}^T$ . When the data consist of  $M$  observations with  $N$  variables, the input to ICA is arranged in an  $N \times M$  matrix  $\mathbf{X}$ .

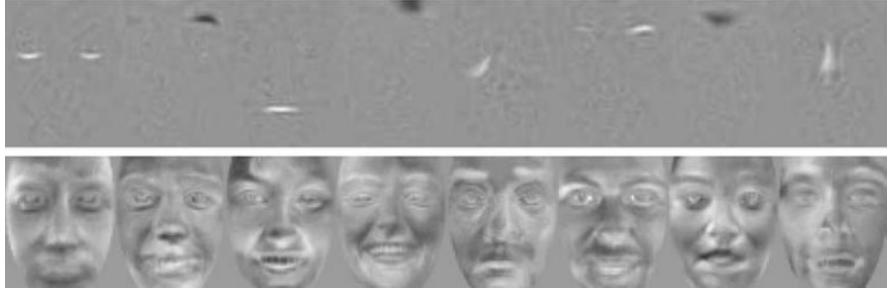
Bartlett et al. [1, 10] investigated the use of ICA framework for face recognition in two fundamentally different architectures:

**Architecture I** Rows of  $\mathbf{S}$  are *independent basis images*, which combined by  $\mathbf{A}$  yield the input images  $\mathbf{X}$ . Learning  $\mathbf{W}$  allows us to estimate the basis images in the rows of  $\mathbf{U}$ . In practice, for reasons of computational tractability, PCA is first performed on the input data  $\mathbf{X}$  to find the top  $K$  eigenfaces; these are arranged in the columns of a matrix  $\mathbf{E}$ .<sup>4</sup> Then ICA is performed on  $\mathbf{E}^T$ ; that is, the images are variables, and the pixel values are observations. Let  $\mathbf{C}$  be the PCA coefficient matrix, that is,  $\mathbf{X} = \mathbf{C}\mathbf{E}^T$ . Then the  $k$  independent ICA basis images (Fig. 2.6, top) are estimated by the rows of  $\mathbf{U} = \mathbf{W}\mathbf{E}^T$ , and the coefficients for the data are computed from  $\mathbf{X} = \mathbf{E}\mathbf{W}^{-1}\mathbf{U}$ .

**Architecture II** This architecture assumes that the sources in  $\mathbf{S}$  are independent coefficients, and the columns of the mixing matrix  $\mathbf{A}$  are the basis images; that is, the

<sup>3</sup>A number of algorithms exist; most notable are Jade [5], InfoMax, and FastICA [16].

<sup>4</sup>These eigenfaces are linear combination of the original images, which under the assumptions of ICA should not affect the resulting decomposition.



**Fig. 2.6** Basis images obtained with ICA: Architecture I (*top*) and II (*bottom*). (From Draper et al. [10], with permission)

variables in the source separation problem are the pixels. Similar to Architecture I, ICA is preceded by PCA; however, in this case the input to ICA is the coefficient matrix  $\mathbf{C}$ . The resulting ICA basis consists of the columns of  $\mathbf{E}\mathbf{A}$  (Fig. 2.6, bottom), and the coefficients are found in the rows of  $\mathbf{U} = \mathbf{W}\mathbf{C}^T$ . These coefficients give the *factorial representation* of the data.

Generally, the bases obtained with Architecture I reflect more local properties of the faces, whereas the bases in Architecture II have global properties and much more resemble faces (Fig. 2.6).

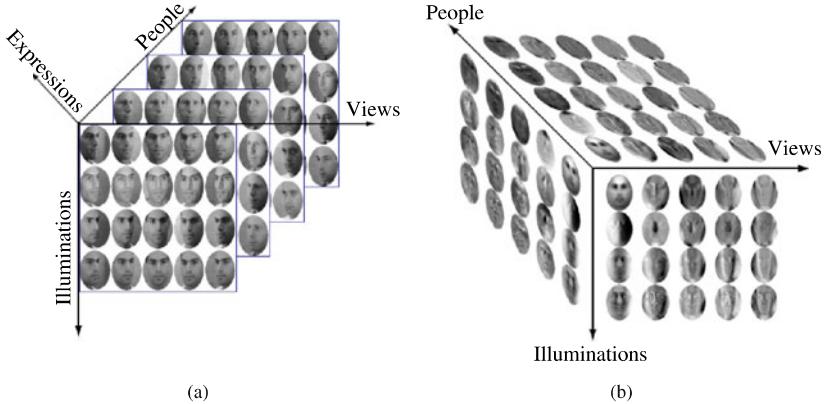
### 2.3.6 Multilinear SVD: “Tensorfaces”

The linear analysis methods discussed above have been shown to be suitable when pose, illumination, or expression are fixed across the face database. When any of these parameters is allowed to vary, the linear subspace representation does not capture this variation well (see Sect. 2.6.1). In Sect. 2.4, we discuss recognition with nonlinear subspaces. An alternative, *multilinear* approach, called “tensorfaces,” has been proposed by Vasilescu and Terzopoulos in [37, 38].

Tensor is a multidimensional generalization of a matrix: a *n-order tensor*  $\mathcal{A}$  is an object with  $n$  indices, with elements denoted by  $a_{i_1, \dots, i_n} \in \mathbb{R}$ . Note that there are  $n$  ways to *flatten* this tensor (i.e., to rearrange the elements in a matrix): The  $i$ th row of  $\mathcal{A}_{(s)}$  is obtained by concatenating all the elements of  $\mathcal{A}$  of the form  $a_{i_1, \dots, i_{s-1}, i, i_{s+1}, \dots, i_n}$ .

A generalization of matrix multiplication for tensors is the *l-mode* product  $\mathcal{A} \times_l \mathbf{M}$  of a tensor  $\mathcal{A}$  and an  $m \times k$  matrix  $\mathbf{M}$ , where  $k$  is the  $l$ th dimension of  $\mathcal{A}$ .

$$(\mathcal{A} \times_l \mathbf{M})_{i_1, \dots, i_{l-1}, j, i_{l+1}, \dots, i_n} = \sum_{i=1}^k a_{i_1, \dots, i_{l-1}, i, i_{l+1}, \dots, i_n} m_{ji}. \quad (2.15)$$



**Fig. 2.7** Tensorfaces. **a** Data tensor; the four dimensions visualized are identity, illumination, pose, and the pixel vector. The fifth dimension corresponds to expression (only the subtensor for neutral expression is shown). **b** Tensorfaces decomposition. (From Vasilescu and Terzopoulos [37], with permission)

Under this definition, Vasilescu and Terzopoulos proposed [38] an algorithm they called *n-mode SVD*, which decomposes an *n*-dimensional tensor  $\mathcal{A}$  into

$$\mathcal{A} = \Sigma \times_1 U_1 \times_2 U_2 \cdots \times_n U_n. \quad (2.16)$$

The role of the *core tensor*  $\Sigma$  in this decomposition is similar to the role of the singular value matrix  $\Sigma$  in SVD (2.4): It governs the interactions between the *mode matrices*  $U_1, \dots, U_n$ , which contain the orthonormal bases for the spaces spanned by the corresponding dimensions of the data tensor. The mode matrices can be obtained by flattening the tensor across the corresponding dimension and performing PCA on the columns of the resulting matrix; then the core tensor is computed as

$$\Sigma = \mathcal{A} \times_1 U_1^T \times_2 U_2^T \cdots \times_N U_n^T.$$

The notion of tensor can be applied to a face image ensemble in the following way [38]: Consider a set of  $N$ -pixel images of  $N_p$  people's faces, each photographed in  $N_v$  viewpoints, with  $N_i$  illuminations and  $N_e$  expressions. The entire set may be arranged in an  $N_p \times N_v \times N_i \times N_e \times N$  tensor of order 5. Figure 2.7a illustrates this concept: Only four dimensions are shown; to visualize the fifth one (expression), imagine that the four-dimensional tensors for different expressions are "stacked."

In this context, the face image tensor can be decomposed into

$$\mathcal{A} = \Sigma \times_1 U_p \times_2 U_v \times_3 U_i \times_4 U_e \times_5 U_{\text{pixels}}. \quad (2.17)$$

Each mode matrix represents a parameter of the object appearance. For example, the columns of the  $N_e \times N_e$  matrix  $U_e$  span the space of expression parameters. The columns of  $U_{\text{pixels}}$  span the image space; these are exactly the eigenfaces that would be obtained by direct PCA on the entire data set.

Each person in the database can be represented by a single  $N_p$  vector, which contains coefficients with respect to the bases comprising the tensor

$$\mathcal{B} = \mathcal{Z} \times_2 \mathbf{U}_v \times_3 \mathbf{U}_i \times_4 \mathbf{U}_e \times_5 \mathbf{U}_{\text{pixels}}.$$

For a given viewpoint  $v$ , illumination  $i$ , and expression  $e$ , an  $N_p \times N$  matrix  $\mathbf{B}_{v,i,e}$  can be obtained by indexing into  $\mathcal{B}$  for  $v, i, e$  and flattening the resulting  $N_p \times 1 \times 1 \times 1 \times N$  subtensor along the identity (people) mode. Now a training image  $\mathbf{x}_{p,v,e,i}$  of a person  $j$  under the given conditions can be written as

$$\mathbf{x}_{p,v,e,i} = \mathbf{B}_{v,i,e}^T \mathbf{c}_p \quad (2.18)$$

where  $\mathbf{c}_j$  is the  $j$ th row vector of  $\mathbf{U}_p$ .

Given an input image  $\mathbf{x}$ , a candidate coefficient vector  $\mathbf{c}_{v,i,e}$  is computed for all combinations of viewpoint, expression, and illumination, solving (2.18). The recognition is carried out by finding the value of  $j$  that yields the minimum Euclidean distance between  $\mathbf{c}$  and the vectors  $\mathbf{c}_j$  across all illuminations, expressions, and viewpoints.<sup>5</sup>

Vasilescu and Terzopoulos [38] reported experiments involving the data tensor consisting of images of  $N_p = 28$  subjects photographed in  $N_i = 3$  illumination conditions from  $N_v = 5$  viewpoints, with  $N_e = 3$  different expressions; the images were resized and cropped so they contain  $N = 7493$  pixels. The performance of tensorfaces is reported to be significantly better than that of standard eigenfaces described in Sect. 2.3.1.

## 2.4 Nonlinear Subspaces

In this section, we describe a number of techniques that do not assume that the principal manifold is linear.

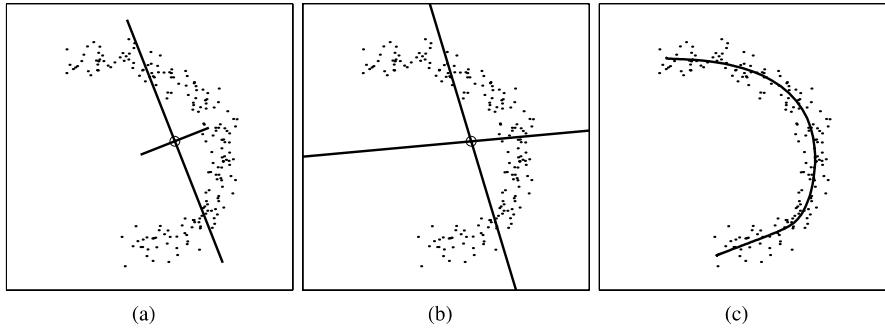
### 2.4.1 Principal Curves and Nonlinear PCA

The defining property of nonlinear principal manifolds is that the *inverse image* of the manifold in the original space  $\mathbb{R}^N$  is a nonlinear (curved) lower-dimensional surface that “passes through the middle of the data” while minimizing the sum total distance between the data points and their projections on that surface. Often referred to as *principal curves* [14], this formulation is essentially a nonlinear regression on the data. An example of a principal curve is shown in Fig. 2.8c.

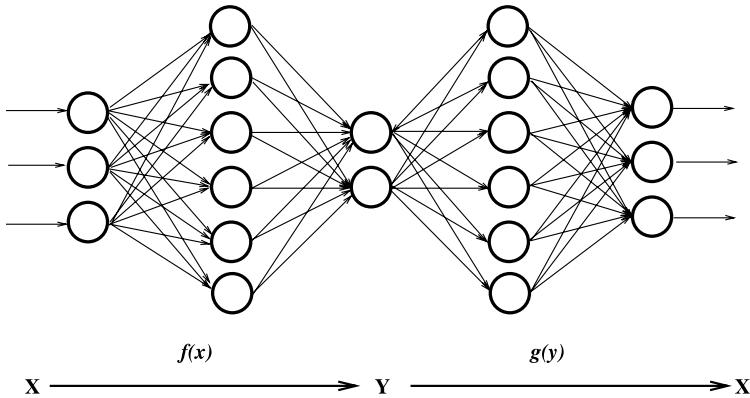
One of the simplest methods for computing nonlinear principal manifolds is the nonlinear PCA (NLPCA) autoencoder multilayer neural network [9, 20] shown in

---

<sup>5</sup>This also provides an estimate of the parameters (e.g., illumination) for the input image.



**Fig. 2.8** **a** PCA basis (linear, ordered, and orthogonal). **b** ICA basis (linear, unordered, and nonorthogonal). **c** Principal curve (parameterized nonlinear manifold). The circle shows the data mean



**Fig. 2.9** Autoassociative (“bottleneck”) neural network for computing principal manifolds  $y \in \mathbb{R}^k$  in the input space  $x \in \mathbb{R}^N$

Fig. 2.9. The “bottleneck” layer forms a lower-dimensional manifold representation by means of a nonlinear *projection* function  $f(x)$ , implemented as a weighted sum-of-sigmoids. The resulting principal components  $y$  have an inverse mapping with a similar nonlinear *reconstruction* function  $g(y)$ , which reproduces the input data as accurately as possible. The NLPCA computed by such a multilayer sigmoidal neural network is equivalent (with certain exceptions<sup>6</sup>) to a *principal surface* under the more general definition [13, 14]. To summarize, the main properties of NLPCA are

$$y = f(x), \quad x \approx g(y), \quad P(y) = ? \quad (2.19)$$

---

<sup>6</sup>The class of functions attainable by this neural network restricts the projection function  $f()$  to be smooth and differentiable, and hence suboptimal in some cases [22].

corresponding to nonlinear projection, approximate reconstruction, and typically no prior knowledge regarding the joint distribution of the components, respectively (however, see Zemel and Hinton [43] for an example of devising suitable priors in such cases). The principal curve in Fig. 2.8c was generated with a 2-4-1-4-2 layer neural network of the type shown in Fig. 2.9. Note how the principal curve yields a compact, relatively accurate representation of the data, in contrast to the linear models (PCA and ICA).

### 2.4.2 Kernel-PCA and Kernel-Fisher Methods

Recently nonlinear principal component analysis has been revived with the “kernel eigenvalue” method of Schölkopf et al. [32]. The basic methodology of KPCA is to apply a nonlinear mapping to the input  $\Psi(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathbb{R}^L$  and then solve for a linear PCA in the resulting feature space  $\mathbb{R}^L$ , where  $L$  is larger than  $N$  and possibly infinite. Because of this increase in dimensionality, the mapping  $\Psi(\mathbf{x})$  is made implicit (and economical) by the use of kernel functions satisfying Mercer’s theorem [7]

$$k(\mathbf{x}_i, \mathbf{x}_j) = [\Psi(\mathbf{x}_i) \cdot \Psi(\mathbf{x}_j)] \quad (2.20)$$

where kernel evaluations  $k(\mathbf{x}_i, \mathbf{x}_j)$  in the input space correspond to dot-products in the higher dimensional feature space. Because computing covariance is based on dot-products, performing a PCA in the feature space can be formulated with kernels in the input space without the explicit (and possibly prohibitively expensive) direct computation of  $\Psi(\mathbf{x})$ . Specifically, assuming that the projection of the data in feature space is zero-mean (“centered”), the covariance is given by

$$\Sigma_K = \langle \Psi(\mathbf{x}_i), \Psi(\mathbf{x}_i)^T \rangle \quad (2.21)$$

with the resulting eigenvector equation  $\lambda V = \Sigma_K V$ . Since the eigenvectors (columns of  $V$ ) must lie in the span of the training data  $\Psi(\mathbf{x}_i)$ , it must be true that for each training point

$$\lambda (\Psi(\mathbf{x}_i) \cdot V) = (\Psi(\mathbf{x}_i) \cdot \Sigma_K V) \quad \text{for } i = 1, \dots, T \quad (2.22)$$

and that there must exist coefficients  $\{w_i\}$  such that

$$V = \sum_{i=1}^T w_i \Psi(\mathbf{x}_i). \quad (2.23)$$

Using the definition of  $\Sigma_K$ , substituting the above equation into (2.22) and defining the resulting  $T$ -by- $T$  matrix  $K$  by  $K_{ij} = [\Psi(\mathbf{x}_i) \cdot \Psi(\mathbf{x}_j)]$  leads to the equivalent eigenvalue problem formulated in terms of kernels in the input space

$$T \lambda w = Kw \quad (2.24)$$

where  $\mathbf{w} = (w_1, \dots, w_T)^T$  is the vector of expansion coefficients of a given eigenvector  $V$  as defined in (2.23). The kernel matrix  $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$  is then diagonalized with a standard PCA.<sup>7</sup> Orthonormality of the eigenvectors,  $(V^n \cdot V^n) = 1$ , leads to the equivalent normalization of their respective expansion coefficients,  $\lambda_n(\mathbf{w}^n \cdot \mathbf{w}^n) = 1$ .

Subsequently, the KPCA principal components of any input vector can be efficiently computed with simple kernel evaluations against the dataset. The  $n$ th principal component  $y_n$  of  $\mathbf{x}$  is given by

$$y_n = (V_n \cdot \Psi(\mathbf{x})) = \sum_{i=1}^T w_i^n k(\mathbf{x}, \mathbf{x}_i) \quad (2.25)$$

where  $V_n$  is the  $n$ th eigenvector of the feature space defined by  $\Psi$ . As with PCA, the eigenvectors  $V_n$  can be ranked by decreasing order of their eigenvalues  $\lambda_n$  and a  $d$ -dimensional manifold projection of  $\mathbf{x}$  is  $\mathbf{y} = (y_1, \dots, y_d)^T$ , with individual components defined by (2.25).

A significant advantage of KPCA over neural network and principal curves is that KPCA does not require nonlinear optimization, is not subject to overfitting, and does not require prior knowledge of network architecture or the number of dimensions. Furthermore, unlike traditional PCA, one can use more eigenvector projections than the input dimensionality of the data (because KPCA is based on the matrix  $K$ , the number of eigenvectors or features available is  $T$ ). On the other hand, the selection of the optimal kernel (and its associated parameters) remains an “engineering problem.” Typical kernels include Gaussians  $\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma^2)$ , polynomials  $(\mathbf{x}_i \cdot \mathbf{x}_j)^d$  and sigmoids  $\tanh(a(\mathbf{x}_i \cdot \mathbf{x}_j) + b)$ , all of which satisfy Mercer’s theorem [7].

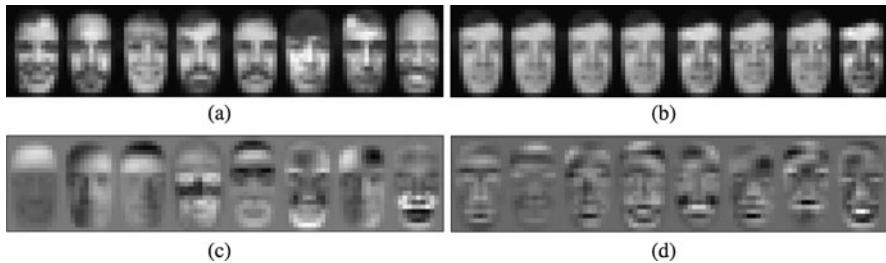
Similar to the derivation of KPCA, one may extend the Fisherfaces method (see Sect. 2.3.3) by applying the FLD in the feature space. Yang [42] derived the kernel Fisherfaces algorithm, which maximizes the between-scatter to within-scatter ratio in the feature space through the use of the kernel matrix  $K$ . In experiments on two data sets that contained images from 40 and 11 subjects, respectively, with varying pose, scale, and illumination, this algorithm showed performance clearly superior to that of ICA, PCA, and KPCA and somewhat better than that of the standard Fisherfaces.

## 2.5 Empirical Comparison of Subspace Methods

Moghaddam [23] reported on an extensive evaluation of many of the subspace methods described above on a large subset of the FERET data set [31] (see also Chap. 13).

---

<sup>7</sup>However, computing  $\Sigma_K$  in (2.21) requires “centering” the data by computing the mean of  $\Psi(\mathbf{x}_i)$ . Because there is no explicit computation of  $\Psi(\mathbf{x}_i)$ , the equivalent must be carried out when computing the kernel matrix  $K$ . For details on “centering”  $K$ , see Schölkopf et al. [32].



**Fig. 2.10** Experiments on FERET data. **a** Several faces from the gallery. **b** Multiple probes for one individual, with different facial expressions, eyeglasses, variable ambient lighting, and image contrast. **c** Eigenfaces. **d** ICA basis images

The experimental data consisted of a training “gallery” of 706 individual FERET faces and 1123 “probe” images containing one or more views of every person in the gallery. All these images were aligned and normalized as described by Moghaddam and Pentland [25]. The multiple probe images reflected various expressions, lighting, glasses on/off, and so on. The study compared the Bayesian approach described in Sect. 2.3.4 to a number of other techniques and tested the limits of the recognition algorithms with respect to image resolution or equivalently the amount of visible facial detail. Because the Bayesian algorithm was independently evaluated in DARPA’s 1996 FERET face recognition competition [31] with medium resolution images ( $84 \times 44$  pixels)—achieving an accuracy of  $\approx 95\%$  on  $O(10^3)$  individuals—it was decided to lower the resolution (the number of pixels) by a factor 16. Therefore, the aligned faces in the data set were downsampled to  $21 \times 12$  pixels, yielding input vectors in a  $\mathbb{R}^{N=252}$  space. Several examples are shown in Fig. 2.10a, b.

The reported results were obtained with a fivefold Cross-Validation (CV) analysis. The total data set of 1829 faces (706 unique individuals and their collective 1123 probes) was randomly partitioned into five subsets with unique (nonoverlapping) individuals and their associated probes. Each subset contained both gallery and probe images of  $\approx 140$  unique individuals. For each of the five subsets, the recognition task was correctly matching the multiple probes to the  $\approx 140$  gallery faces using the other four subsets as training data. Note that with  $N = 252$  and using 80% of the entire dataset for training, there are nearly three times as many training samples than the data dimensionality; thus, parameter estimations (for PCA, ICA, KPCA, and the Bayesian method) were properly overconstrained.

The resulting five experimental trials were pooled to compute the mean and standard deviation of the recognition rates for each method. The fact that the training and testing sets had no overlap in terms of individual identities led to an evaluation of the algorithms’ *generalization* performance—the ability to recognize new individuals who were not part of the manifold computation or density modeling with the training set.

The baseline recognition experiments used a default manifold dimensionality of  $k = 20$ . This choice of  $k$  was made for two reasons: It led to a reasonable PCA reconstruction error of  $MSE = 0.0012$  (or 0.12% per pixel with a normalized intensity

range of  $[0, 1]$ ) and a baseline PCA recognition rate of  $\approx 80\%$  (on a different 50/50 partition of the dataset), thereby leaving a sizable margin for improvement. Note that because the recognition experiments were essentially a 140-way classification task, chance performance was approximately 0.7%.

### 2.5.1 PCA-Based Recognition

The baseline algorithm for these face recognition experiments was standard PCA (eigenface) matching. The first eight principal eigenvectors computed from a single partition are shown in Fig. 2.10c. Projection of the test set probes onto the 20-dimensional linear manifold (computed with PCA on the training set only) followed by nearest-neighbor matching to the  $\approx 140$  gallery images using a Euclidean metric yielded a mean recognition rate of 77.31%, with the highest rate achieved being 79.62% (Table 2.1). The full image-vector nearest-neighbor (template matching) (i.e., on  $x \in \mathbb{R}^{252}$ ) yielded a recognition rate of 86.46% (see dashed line in Fig. 2.11). Clearly, performance is degraded by the  $252 \rightarrow 20$  dimensionality reduction, as expected.

### 2.5.2 ICA-Based Recognition

For ICA-based recognition (Architecture II, see Sect. 2.3.5) two algorithms based on fourth-order cumulants were tried: the “JADE” algorithm of Cardoso [5] and the fixed-point algorithm of Hyvärinen and Oja [15]. In both algorithms a PCA whitening step (“sphering”) preceded the core ICA decomposition. The corresponding *nonorthogonal* JADE-derived ICA basis is shown in Fig. 2.10d. Similar basis faces were obtained with the method of Hyvärinen and Oja. These basis faces are the columns of the matrix  $A$  in (2.14), and their linear combination (specified by the ICs) reconstructs the training data. The ICA manifold projection of the test set was obtained using  $y = A^{-1}x$ . Nearest-neighbor matching with ICA using the Euclidean  $L_2$  norm resulted in a mean recognition rate of 77.30% with the highest rate being 82.90% (Table 2.1). We found little difference between the two ICA algorithms and noted that ICA resulted in the largest performance variation in the five trials (7.66% SD). Based on the mean recognition rates it is unclear whether ICA provides a systematic advantage over PCA or whether “more non-Gaussian” and/or “more independent” components result in a better manifold for *recognition* purposes with this dataset.

Note that the experimental results of Bartlett et al. [1] with FERET faces did favor ICA over PCA. This seeming disagreement can be reconciled if one considers the differences in the experimental setup and in the choice of the similarity measure. First, the advantage of ICA was seen primarily with more difficult time-separated images. In addition, compared to the results of Bartlett et al. [1] the faces in this

**Table 2.1** Recognition accuracies with  $k = 20$  subspace projections using fivefold cross validation. Results are in percents

Partition	PCA	ICA	KPCA	Bayes
1	78.00	82.90	83.26	95.46
2	79.62	77.29	92.37	97.87
3	78.59	79.19	88.52	94.49
4	76.39	82.84	85.96	92.90
5	73.96	64.29	86.57	93.45
Mean	77.31	77.30	87.34	94.83
SD	2.21	7.66	3.39	1.96

experiment were cropped much tighter, leaving no information regarding hair and face shape, and they were much lower in resolution, factors that when combined make the recognition task much more difficult.

The second factor is the choice of the distance function used to measure similarity in the subspace. This matter was further investigated by Draper et al. [10]. They found that the best results for ICA are obtained using the cosine distance, whereas for eigenfaces the  $L_1$  metric appears to be optimal; with  $L_2$  metric, which was also used in the experiments of Moghaddam [23], the performance of ICA (Architecture II) was similar to that of eigenfaces.

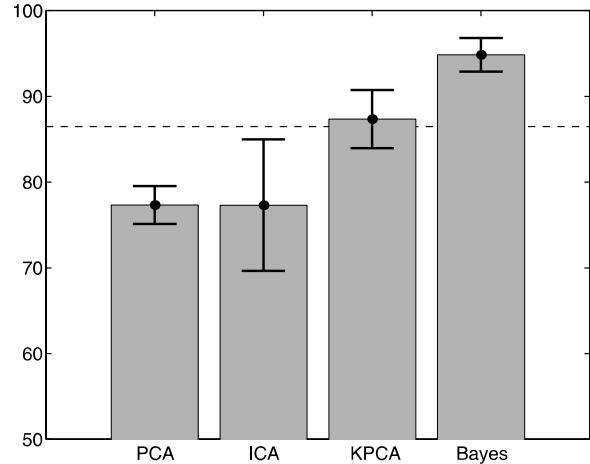
### 2.5.3 KPCA-Based Recognition

For KPCA, the parameters of Gaussian, polynomial, and sigmoidal kernels were first fine-tuned for best performance with a different 50/50 partition validation set, and Gaussian kernels were found to be the best for this data set. For each trial, the kernel matrix was computed from the corresponding training data. Both the test set gallery and probes were projected onto the kernel eigenvector basis (2.25) to obtain the nonlinear principal components which were then used in nearest-neighbor matching of test set probes against the test set gallery images. The mean recognition rate was found to be 87.34%, with the highest rate being 92.37% (Table 2.1). The standard deviation of the KPCA trials was slightly higher (3.39) than that of PCA (2.21), but Fig. 2.11 indicates that KPCA does in fact do better than both PCA and ICA, hence justifying the use of nonlinear feature extraction.

### 2.5.4 MAP-Based Recognition

For Bayesian similarity matching, appropriate training  $\Delta$ s for the two classes  $\Omega_I$  (Fig. 2.10b) and  $\Omega_E$  (Fig. 2.10a) were used for the dual PCA-based density estimates  $P(\Delta | \Omega_I)$  and  $P(\Delta | \Omega_E)$ , which were both modeled as single Gaussians

**Fig. 2.11** Recognition performance of PCA, ICA, and KPCA manifolds versus Bayesian (MAP) similarity matching with a  $k = 20$  dimensional subspace. Dashed line indicates the performance of nearest-neighbor matching with the full-dimensional image vectors



with subspace dimensions of  $k_I$  and  $k_E$ , respectively. The total subspace dimensionality  $k$  was divided evenly between the two densities by setting  $k_I = k_E = k/2$  for modeling.<sup>8</sup>

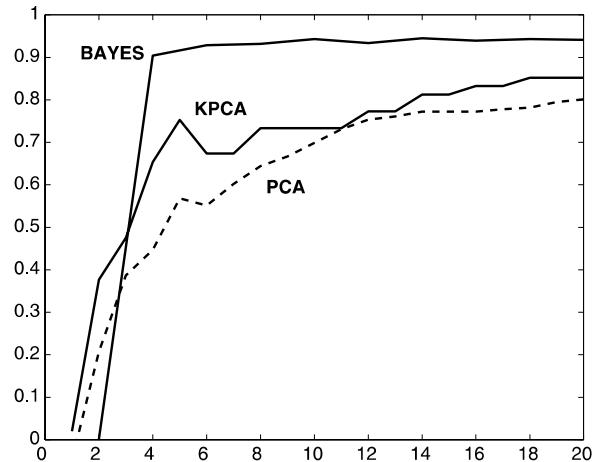
With  $k = 20$ , Gaussian subspace dimensions of  $k_I = 10$  and  $k_E = 10$  were used for  $P(\Delta | \Omega_I)$  and  $P(\Delta | \Omega_E)$ , respectively. Note that  $k_I + k_E = 20$ , thus matching the total number of projections used with the three principal manifold techniques. Using the maximum *a posteriori* (MAP) similarity in (2.9), the Bayesian matching technique yielded a mean recognition rate of 94.83%, with the highest rate achieved being 97.87% (Table 2.1). The standard deviation of the five partitions for this algorithm was also the lowest (1.96) (Fig 2.11).

### 2.5.5 Compactness of Manifolds

The performance of various methods with different size manifolds can be compared by plotting their recognition rates  $R(k)$  as a function of the first  $k$  principal components. For the manifold matching techniques, this simply means using a subspace dimension of  $k$  (the first  $k$  components of PCA/ICA/KPCA), whereas for the Bayesian matching technique this means that the subspace Gaussian dimensions should satisfy  $k_I + k_E = k$ . Thus all methods used the same number of subspace projections. This test was the premise for one of the key points investigated by Moghaddam [23]: Given the *same* number of subspace projections, which of these techniques is better at data modeling and subsequent recognition? The presumption is that the one achieving the highest recognition rate with the smallest dimension is preferred.

<sup>8</sup>In practice,  $k_I > k_E$  often works just as well. In fact, as  $k_E \rightarrow 0$ , one obtains a maximum-likelihood similarity  $S = P(\Delta | \Omega_I)$  with  $k_I = k$ , which for this data set is only a few percent less accurate than MAP [26].

**Fig. 2.12** Recognition accuracy  $R(k)$  of PCA, KPCA, and Bayesian similarity with increasing dimensionality  $k$  of the principal subspace. ICA results, not shown, are similar to those of PCA



For this particular dimensionality test, the total data set of 1829 images was partitioned (split) in half: a training set of 353 gallery images (randomly selected) along with their corresponding 594 probes and a testing set containing the remaining 353 gallery images and their corresponding 529 probes. The training and test sets had no overlap in terms of individuals' identities. As in the previous experiments, the test set probes were matched to the test set gallery images based on the projections (or densities) computed with the training set. The results of this experiment are shown in Fig. 2.12, which plots the recognition rates as a function of the dimensionality of the subspace  $k$ . This is a more revealing comparison of the relative performance of the methods, as *compactness* of the manifolds—defined by the lowest acceptable value of  $k$ —is an important consideration in regard to both generalization error (overfitting) and computational requirements.

### 2.5.6 Discussion

The relative performance of the principal manifold techniques and Bayesian matching is summarized in Table 2.1 and Fig. 2.11. The advantage of probabilistic matching over metric matching on both linear and nonlinear manifolds is quite evident ( $\approx 18\%$  increase over PCA and  $\approx 8\%$  over KPCA). Note that the dimensionality test results in Fig. 2.12 indicate that KPCA outperforms PCA by a  $\approx 10\%$  margin, and even more so with only few principal components (a similar effect was reported by Schölkopf et al. [32] where KPCA outperforms PCA in low-dimensional manifolds). However, Bayesian matching achieves  $\approx 90\%$  with only four projections—two for each  $P(\Delta | \Omega)$ —and dominates both PCA and KPCA throughout the entire range of subspace dimensions in Fig. 2.12.

A comparison of the subspace techniques with respect to multiple criteria is shown in Table 2.2. Note that PCA, KPCA, and the dual subspace density estimation are uniquely defined for a given training set (making experimental comparisons

**Table 2.2** Comparison of the subspace techniques across multiple attributes ( $k = 20$ )

	PCA	ICA	KPCA	Bayes
Accuracy	77%	77%	87%	95%
Computation	$10^8$	$10^9$	$10^9$	$10^8$
Uniqueness	Yes	No	Yes	Yes
Projections	Linear	Linear	Nonlinear	Linear

repeatable), whereas ICA is not unique owing to the variety of techniques used to compute the basis and the iterative (stochastic) optimizations involved. Considering the relative computation (of training), KPCA required  $\approx 7 \times 10^9$  floating-point operations compared to PCA’s  $\approx 2 \times 10^8$  operations. On the average, ICA computation was one order of magnitude larger than that of PCA. Because the Bayesian similarity method’s learning stage involves two separate PCAs, its computation is merely twice that of PCA (the same order of magnitude).

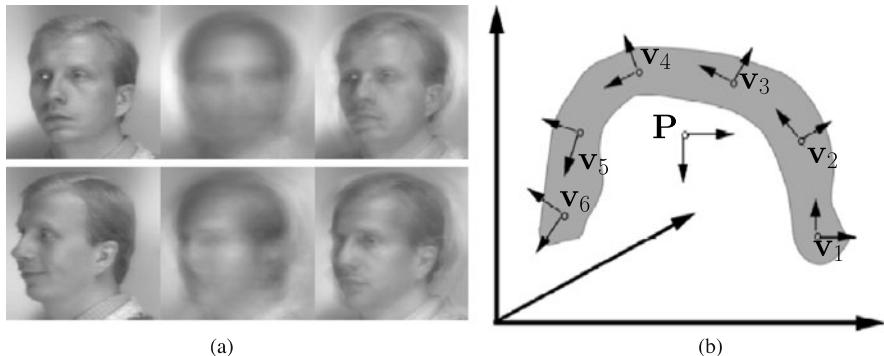
Considering its significant performance advantage (at low subspace dimensionality) and its relative simplicity, the dual-eigenface Bayesian matching method is a highly effective subspace modeling technique for face recognition. In independent FERET tests conducted by the U.S. Army Laboratory [31], the Bayesian similarity technique outperformed PCA and other subspace techniques, such as Fisher’s linear discriminant (by a margin of at least 10%). Experimental results described above show that a similar recognition accuracy can be achieved using mere “thumbnails” with 16 times fewer pixels than in the images used in the FERET test. These results demonstrate the Bayesian matching technique’s robustness with respect to image resolution, revealing the surprisingly small amount of facial detail required for high accuracy performance with this learning technique.

## 2.6 Methodology and Usage

In this section, we discuss issues that require special care from the practitioner, in particular, the approaches designed to handle database with varying imaging conditions. We also present a number of extensions and modifications of the subspace methods.

### 2.6.1 Multiple View-Based Approach for Pose

The problem of face recognition under general viewing conditions (change in pose) can also be approached using an eigenspace formulation. There are essentially two ways to approach this problem using an eigenspace framework. Given  $M$  individuals under  $C$  different views, one can do recognition and pose estimation in a universal eigenspace computed from the combination of  $MC$  images. In this way, a single



**Fig. 2.13** Parametric versus view-based eigenspace methods. **a** Reconstructions of the input image (left) with parametric (middle) and view-based (right) eigenspaces. *Top*: training image; *bottom*: novel (test) image. **b** Difference in the way the two approaches span the manifold

parametric eigenspace encodes identity as well as pose. Such an approach, for example, has been used by Murase and Nayar [28] for general 3D object recognition.

Alternatively, given  $M$  individuals under  $C$  different views, we can build a view-based set of  $C$  distinct eigenspaces, each capturing the variation of the  $M$  individuals in a common view. The view-based eigenspace is essentially an extension of the eigenface technique to multiple sets of eigenvectors, one for each combination of scale and orientation. One can view this architecture as a set of parallel observers, each trying to explain the image data with their set of eigenvectors. In this view-based, multiple-observer approach, the first step is to determine the location and orientation of the target object by selecting the eigenspace that best describes the input image. This can be accomplished by calculating the likelihood estimate using each viewspace's eigenvectors and then selecting the maximum.

The key difference between the view-based and parametric representations can be understood by considering the geometry of face subspace, illustrated in Fig. 2.13b. In the high-dimensional vector space of an input image, multiple-orientation training images are represented by a set of  $C$  distinct regions, each defined by the scatter of  $M$  individuals. Multiple views of a face form nonconvex (yet connected) regions in image space [3]. Therefore, the resulting ensemble is a highly complex and nonseparable manifold.

The parametric eigenspace attempts to describe this ensemble with a projection onto a single low-dimensional linear subspace (corresponding to the first  $k$  eigenvectors of the  $MC$  training images). In contrast, the view-based approach corresponds to  $C$  independent subspaces, each describing a particular region of the face subspace (corresponding to a particular view of a face). The principal manifold  $v_c$  of each region  $c$  is extracted separately. The relevant analogy here is that of modeling a complex distribution by a single cluster model or by the union of several component clusters. Naturally, the latter (view-based) representation can yield a more accurate representation of the underlying geometry.

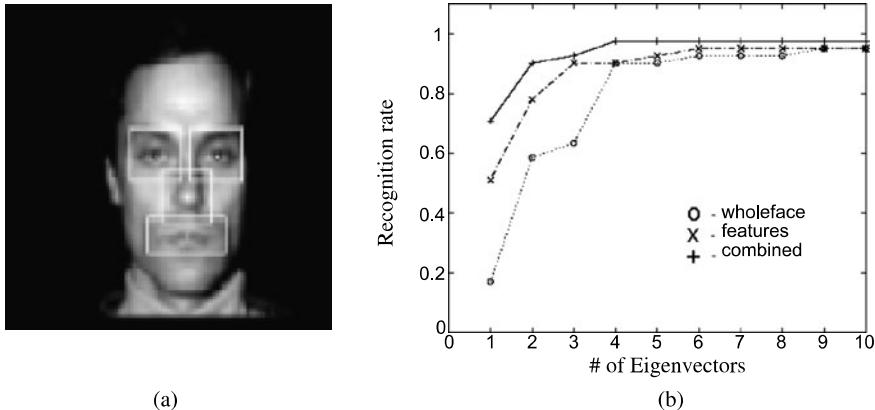
This difference in representation becomes evident when considering the quality of reconstructed images using the two methods. Figure 2.13 compares reconstruc-



**Fig. 2.14** Multiview face image data used in the experiments described in Sect. 2.6.1. (From Moghaddam and Pentland [25], with permission)

tions obtained with the two methods when trained on images of faces at multiple orientations. In the top row of Fig. 2.13a, we see first an image in the training set, followed by reconstructions of this image using first the parametric eigenspace and then the view-based eigenspace. Note that in the parametric reconstruction, neither the pose nor the identity of the individual is adequately captured. The view-based reconstruction, on the other hand, provides a much better characterization of the object. Similarly, in the bottom row of Fig. 2.13a, we see a novel view ( $+68^\circ$ ) with respect to the training set ( $-90^\circ$  to  $+45^\circ$ ). Here, both reconstructions correspond to the nearest view in the training set ( $+45^\circ$ ), but the view-based reconstruction is seen to be more representative of the individual's identity. Although the quality of the reconstruction is not a direct indicator of the recognition power, from an information-theoretical point-of-view, the multiple eigenspace representation is a more accurate representation of the signal content.

The view-based approach was evaluated [25] on data similar to that shown in Fig. 2.14 which consisted of 189 images: nine views of 21 people. The viewpoints were evenly spaced from  $-90^\circ$  to  $+90^\circ$  along the horizontal plane. In the first series of experiments, the interpolation performance was tested by training on a subset of the available views ( $\pm 90^\circ$ ,  $\pm 45^\circ$ ,  $0^\circ$ ) and testing on the intermediate views ( $\pm 68^\circ$ ,  $\pm 23^\circ$ ). A 90% average recognition rate was obtained. A second series of experiments tested the extrapolation performance by training on a range of views (e.g.,  $-90^\circ$  to  $+45^\circ$ ) and testing on novel views outside the training range (e.g.,  $+68^\circ$  and  $+90^\circ$ ). For testing views separated by  $\pm 23^\circ$  from the training range, the average recognition rate was 83%. For  $\pm 45^\circ$  testing views, the average recognition rate was 50%.



**Fig. 2.15** Modular eigenspaces. **a** Rectangular patches whose appearance is modeled with eigenfeatures. **b** Performance of eigenfaces, eigenfeatures, and the layered combination of both as a function of subspace dimension. (From Pentland et al. [30], with permission)

### 2.6.2 Modular Recognition

The eigenface recognition method is easily extended to facial features [30], as shown in Fig. 2.15a. This leads to an improvement in recognition performance by incorporating an additional layer of description in terms of facial features. This can be viewed as either a modular or layered representation of a face, where a coarse (low-resolution) description of the whole head is augmented by additional (higher resolution) details in terms of salient facial features. Pentland et al. [30] called the latter component *eigenfeatures*. The utility of this layered representation (eigenface plus eigenfeatures) was tested on a small subset of a large face database: a representative sample of 45 individuals with two views per person, corresponding to different facial expressions (neutral vs. smiling). This set of images was partitioned into a training set (neutral) and a testing set (smiling). Because the difference between these particular facial expressions is primarily articulated in the mouth, this feature was discarded for recognition purposes.

Figure 2.15b shows the recognition rates as a function of the number of eigenvectors for eigenface-only, eigenfeature only, and the combined representation. What is surprising is that (for this small dataset at least) the eigenfeatures alone were sufficient to achieve an (asymptotic) recognition rate of 95% (equal to that of the eigenfaces).

More surprising, perhaps, is the observation that in the lower dimensions of eigenspace eigenfeatures outperformed the eigenface recognition. Finally, by using the combined representation, one gains a slight improvement in the asymptotic recognition rate (98%). A similar effect was reported by Brunelli and Poggio [4], where the cumulative normalized correlation scores of templates for the face, eyes, nose, and mouth showed improved performance over the face-only templates.

A potential advantage of the eigenfeature layer is the ability to overcome the shortcomings of the standard eigenface method. A pure eigenface recognition sys-

tem can be fooled by gross variations in the input image (e.g., hats, beards). However, the feature-based representation may still find the correct match by focusing on the characteristic nonoccluded features (e.g., the eyes and nose).

### 2.6.3 Recognition with Sets

An interesting recognition paradigm involves the scenario in which the input consists not of a single image but of a *set* of images of an unknown person. The set may consist of a contiguous *sequence* of frames from a video or a noncontiguous, perhaps unordered, set of photographs extracted from a video or obtained from individual snapshots. The former case is discussed in Chap. 13 (recognition from video). In the latter case, which we consider here, no temporal information is available. A possible approach, and in fact the one often taken until recently, has been to apply standard recognition methods to every image in the input set and then combine the results, typically by means of voting.

However, a large set of images contains more information than every individual image in it: It provides clues not only on the possible appearance on one's face but also on the typical patterns of variation. Technically, just as a set of images known to contain an individual's face allows one to represent that individual by an estimated intrinsic subspace, so the unlabeled input set leads to a subspace estimate that represents the unknown subject. The recognition task can then be formulated in terms of matching the subspaces.

One of the first approaches to this task has been the mutual subspace method (MSM) [41], which extracts the principal linear subspace of fixed dimension (via PCA) and measures the distance between subspaces by means of *principal angles* (the minimal angle between any two vectors in the subspaces). MSM has the desirable feature that it builds a compact model of the distribution of observations. However, it ignores important statistical characteristics of the data, as the eigenvalues corresponding to the principal components, as well as the means of the samples, are disregarded in the comparison. Thus its decisions may be statistically suboptimal.

A probabilistic approach to measuring subspace similarity has been proposed [33]. The underlying statistical model assumes that images of the  $j$ th person's face have probability density  $p_j$ ; the density of the unknown subject's face is denoted by  $p_0$ . The task of the recognition system is then to find the class label  $j^*$ , satisfying

$$j^* = \operatorname{argmax}_j \Pr(p_0 = p_j). \quad (2.26)$$

Therefore, given a set of images distributed by  $p_0$ , solving (2.26) amounts to choosing optimally between  $M$  hypotheses of the form in statistics is sometimes referred to as the two-sample hypothesis: that two sets of examples come from the same distribution. A principled way to solve this task is to choose the hypothesis  $j$  for which the *Kullback-Leibler divergence* between  $p_0$  and  $p_j$  is minimized.

In reality, the distributions  $p_j$  are unknown and must be estimated from data, as well as  $p_0$ . Shakhnarovich et al. [33] modeled these distributions as Gaussians (one per subject), which are estimated according to the method described in Sect. 2.3.2. The KL divergence is then computed in closed form. In the experiments reported by these authors [33], this method significantly outperformed the MSM.

Modeling the distributions by a single Gaussian is somewhat limiting; Wolf and Shashua [40] extended this approach and proposed a nonparametric discriminative method: *kernel principal angles*. They devised a positive definite kernel that operates on pairs of data matrices by projecting the data (columns) into a feature space of arbitrary dimension, in which principal angles can be calculated by computing inner products between the examples (i.e., application of the kernel). Note that this approach corresponds to nonlinear subspace analysis in the original space; for instance, one can use polynomial kernels of arbitrary degree. In experiments that included a face recognition task on a set of nine subjects, this method significantly outperformed both MSM and the Gaussian-based KL-divergence model of Shakhnarovich et al. [33].

## 2.7 Conclusions

Subspace methods have been shown to be highly successful in face recognition, as they have in many other vision tasks. The exposition in this chapter roughly follows the chronologic order in which these methods have evolved. Two most notable directions in this evolution can be discerned: (1) the transition from linear to general, possibly nonlinear, and disconnected manifolds; and (2) the introduction of probabilistic and specifically Bayesian methods for dealing with the uncertainty and with similarity. All of these methods share the same core assumption: that ostensibly complex visual phenomena such as images of human faces, represented in a high-dimensional measurement space, are often intrinsically low-dimensional. Exploiting this low dimensionality allows a face recognition system to simplify computations and to focus the attention on the features of the data relevant for the identity of a person.

**Acknowledgements** We thank M.S. Bartlett and M.A.O. Vasilescu for kind permission to use figures from their published work and for their comments. We also acknowledge all who contributed to the research described in this chapter.

## References

1. Bartlett, M., Lades, H., Sejnowski, T.: Independent component representations for face recognition. In: Proceedings of the SPIE: Conference on Human Vision and Electronic Imaging III, vol. 3299, pp. 528–539 (1998)
2. Belhumeur, V., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 711–720 (1997)

3. Bichsel, M., Pentland, A.: Human face recognition and the face image set's topology. *CVGIP, Image Underst.* **59**(2), 254–261 (1994)
4. Brunelli, R., Poggio, T.: Face recognition: Features vs. templates. *IEEE Trans. Pattern Anal. Mach. Intell.* **15**(10), 1042–1052 (1993)
5. Cardoso, J.-F.: High-order contrasts for independent component analysis. *Neural Comput.* **11**(1), 157–192 (1999)
6. Comon, P.: Independent component analysis—a new concept? *Signal Process.* **36**, 287–314 (1994)
7. Courant, R., Hilbert, D.: *Methods of Mathematical Physics*, vol. 1. Interscience, New York (1953)
8. Cover, M., Thomas, J.: *Elements of Information Theory*. Wiley, New York (1994)
9. DeMers, D., Cottrell, G.: Nonlinear dimensionality reduction. In: *Advances in Neural Information Processing Systems*, pp. 580–587. Morgan Kaufmann, San Francisco (1993)
10. Draper, B.A., Baek, K., Bartlett, M.S., Beveridge, J.R.: Recognizing faces with PCA and ICA. *Comput. Vis. Image Underst.* **91**(1–2), 115–137 (2003)
11. Fukunaga, K.: *Introduction to Statistical Pattern Recognition*, 2nd edn. Academic Press, San Diego (1990)
12. Gerbrands, J.J.: On the relationships between SVD, KLT and PCA. *Pattern Recognit.* **14**, 375–381 (1981)
13. Hastie, T.: Principal curves and surfaces. PhD thesis, Stanford University (1984)
14. Hastie, T., Stuetzle, W.: Principal curves. *J. Am. Stat. Assoc.* **84**(406), 502–516 (1989)
15. Hyvärinen, A., Oja, E.: A family of fixed-point algorithms for independent component analysis. Technical Report A40, Helsinki University of Technology (1996)
16. Hyvärinen, A., Oja, E.: Independent component analysis: algorithms and applications. *Neural Netw.* **13**(4–5), 411–430 (2000)
17. Jolliffe, I.T.: *Principal Component Analysis*. Springer, New York (1986)
18. Jutten, C., Herault, J.: Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Process.* **24**, 1–10 (1991)
19. Kirby, M., Sirovich, L.: Application of the Karhunen–Loéve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(1), 103–108 (1990)
20. Kramer, M.A.: Nonlinear principal components analysis using autoassociative neural networks. *AIChE J.* **32**(2), 233–243 (1991)
21. Loève, M.M.: *Probability Theory*. Van Nostrand, Princeton (1955)
22. Malthouse, E.C.: Some theoretical results on nonlinear principal component analysis. Technical report, Northwestern University (1998)
23. Moghaddam, B.: Principal manifolds and Bayesian subspaces for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(6), 780–788 (2002)
24. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object detection. In: *Proceedings of IEEE International Conference on Computer Vision*, pp. 786–793, Cambridge, MA, June 1995
25. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 696–710 (1997)
26. Moghaddam, B., Jebara, T., Pentland, A.: Efficient MAP/ML similarity matching for face recognition. In: *Proceedings of International Conference on Pattern Recognition*, pp. 876–881, Brisbane, Australia, August 1998
27. Moghaddam, B., Jebara, T., Pentland, A.: Bayesian face recognition. *Pattern Recognit.* **33**(11), 1771–1782 (2000)
28. Murase, H., Nayar, S.K.: Visual learning and recognition of 3D objects from appearance. *Int. J. Comput. Vis.* **14**(1), 5–24 (1995)
29. Penet, P., Sirovich, L.: The global dimensionality of face space. In: *Proc. of IEEE Internation Conf. on Face and Gesture Recognition*, pp. 264–270. Grenoble, France (2000)
30. Pentland, A., Moghaddam, B., Starner, T.: View-based and modular eigenspaces for face recognition. In: *Proceedings of IEEE Computer Vision and Pattern Recognition*, pp. 84–91, Seattle, WA, June 1994. IEEE Computer Society Press, Los Alamitos (1994)

31. Phillips, P.J., Moon, H., Rauss, P., Rizvi, S.: The FERET evaluation methodology for face-recognition algorithms. In: Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 137–143, June 1997
32. Schölkopf, B., Smola, A., Müller, K.-R.: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.* **10**(5), 1299–1319 (1998)
33. Shakhnarovich, G., Fisher, J.W., Darrell, T.: Face recognition from long-term observations. In: Proceedings of European Conference on Computer Vision, pp. 851–865, Copenhagen, Denmark, May 2002
34. Tipping, M., Bishop, C.: Probabilistic principal component analysis. Technical Report NCRG/97/010, Aston University, September 1997
35. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–86 (1991)
36. Turk, M., Pentland, A.: Face recognition using eigenfaces. In: Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 586–590, Maui, Hawaii, December 1991
37. Vasilescu, M., Terzopoulos, D.: Multilinear Subspace Analysis of Image Ensembles. In: Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 93–99, Madison, WI, June 2003
38. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear analysis of image ensembles: TensorFaces. In: Proceedings of European Conference on Computer Vision, pp. 447–460, Copenhagen, Denmark, May 2002
39. Wang, X., Tang, X.: Unified subspace analysis for face recognition. In: Proceedings of IEEE International Conference on Computer Vision, pp. 318–323, Nice, France, October 2003
40. Wolf, L., Shashua, A.: Learning over Sets using Kernel Principal Angles. *J. Mach. Learn. Res.* **4**, 913–931 (2003)
41. Yamaguchi, O., Fukui, K., Maeda, K.-I.: Face recognition using temporal image sequence. In: Proc. of IEEE Internation Conf. on Face and Gesture Recognition, pp. 318–323, Nara, Japan, April 1998
42. Yang, M.-H.: Kernel eigenfaces vs. kernel fisherfaces: Face recognition using kernel methods. In: Proc. of IEEE Internation Conf. on Face and Gesture Recognition, pp. 215–220, Washington, DC, May 2002
43. Zemel, R.S., Hinton, G.E.: Developing population codes by minimizing description length. In: Cowan, J.D., Tesauro, G., Alspector, J. (eds.) *Advances in Neural Information Processing Systems*, vol. 6, pp. 11–18. Morgan Kaufmann, San Francisco (1994)



# Chapter 3

## Face Subspace Learning

Wei Bian and Dacheng Tao

### 3.1 Introduction

The last few decades have witnessed a great success of subspace learning for face recognition. From principal component analysis (PCA) [43] and Fisher's linear discriminant analysis [1], a dozen of dimension reduction algorithms have been developed to select effective subspaces for the representation and discrimination of face images [17, 21, 45, 46, 51]. It has demonstrated that human faces, although usually represented by thousands of pixels encoded in high-dimensional arrays, they are intrinsically embedded in a very low dimensional subspace [37]. The using of subspace for face representation helps to reduce "the curse of dimensionality" in subsequent classification, and suppress variations of lighting conditions and facial expressions. In this chapter, we first briefly review conventional dimension reduction algorithms and then present the trend of recent dimension reduction algorithms for face recognition.

The earliest subspace method for face recognition is Eigenface [43], which uses PCA [23] to select the most representative subspace for representing a set of face images. It extracts the principal eigenspace associated with a set of training face images. Mathematically, PCA maximizes the variance in the projected subspace for a given dimensionality, decorrelates the training face images in the projected subspace, and maximizes the mutual information between appearance (training face images) and identity (the corresponding labels) by assuming that face images are Gaussian distributed. Thus, it has been successfully applied for face recognition. By projecting face images onto the subspace spanned by Eigenface, classifiers can be used in the subspace for recognition. One main limitation of Eigenface is that the

---

W. Bian (✉) · D. Tao

Centre for Quantum Computation & Intelligence Systems, FEIT, University of Technology,  
Sydney, NSW 2007, Australia

e-mail: [wei.bian@student.uts.edu.au](mailto:wei.bian@student.uts.edu.au)

D. Tao

e-mail: [dacheng.tao@uts.edu.au](mailto:dacheng.tao@uts.edu.au)

class labels of face images cannot be explored in the process of learning the projection matrix for dimension reduction. Another representative subspace method for face recognition is Fisherface [1]. In contrast to Eigenface, Fisherface finds class specific linear subspace. The dimension reduction algorithm used in Fisherface is Fisher's linear discriminant analysis (FLDA), which simultaneously maximizes the between-class scatter and minimizes the within-class scatter of the face data. FLDA finds in the feature space a low dimensional subspace where the different classes of samples remain well separated after projection to this subspace. If classes are sampled from Gaussian distributions, all with identical covariance matrices, then FLDA maximizes the mean value of the KL divergences between different classes. In general, Fisherface outperforms Eigenface due to the utilized discriminative information.

Although FLDA shows promising performance on face recognition, it has the following major limitations. FLDA discards the discriminative information preserved in covariance matrices of different classes. FLDA models each class by a single Gaussian distribution, so it cannot find a proper projection for subsequent classification when samples are sampled from complex distributions, for example, mixtures of Gaussians. In face recognition, face images are generally captured with different expressions or poses, under different lighting conditions and at different resolution, so it is more proper to assume face images from one person are mixtures of Gaussians. FLDA tends to merge classes which are close together in the original feature space. Furthermore, when the size of the training set is smaller than the dimension of the feature space, FLDA has the undersampled problem.

To solve the aforementioned problems in FLDA, a dozen of variants have been developed in recent years. Especially, the well-known undersample problem of FLDA has received intensive attention. Representative algorithms include the optimization criterion for generalized discriminant analysis [44], the unified subspace selection framework [44] and the two stage approach via QR decomposition [52]. Another important issue is that FLDA meets the class separation problem [39]. That is because FLDA puts equal weights on all class pairs, although intuitively close class pairs should contribute more to the recognition error [39]. To reduce this problem, Lotlikar and Kothari [30] developed the fractional-step FLDA (FS-FLDA) by introducing a weighting function. Loog et al. [28] developed another weighting method for FLDA, namely the approximate pairwise accuracy criterion (aPAC). The advantage of aPAC is that the projection matrix can be obtained by the eigenvalue decomposition. Both methods use weighting schemes to select a subspace that better separates close class pairs. Recently, the general mean [39] (including geometric mean [39] and harmonic mean [3]) base subspace selection and the max-min distance analysis (MMDA) [5] have been proposed to adaptively choose the weights.

Manifold learning is a new technique for reducing the dimensionality in face recognition and has received considerable attentions in recent years. That is because face images lie in a low-dimensional manifold. A large number of algorithms have been proposed to approximate the intrinsic manifold structure of a set of face images, such as locally linear embedding (LLE) [34], ISOMAP [40], Laplacian eigenmaps (LE) [2], Hessian eigenmaps (HLL) [11], Generative Topographic Mapping

(GTM) [6] and local tangent space alignment (LTSA) [53]. LLE uses linear coefficients, which reconstruct a given measurement by its neighbors, to represent the local geometry, and then seeks a low-dimensional embedding, in which these coefficients are still suitable for reconstruction. ISOMAP preserves global geodesic distances of all pairs of measurements. LE preserves proximity relationships by manipulations on an undirected weighted graph, which indicates neighbor relations of pairwise measurements. LTSA exploits the local tangent information as a representation of the local geometry and this local tangent information is then aligned to provide a global coordinate. Hessian Eigenmaps (HLLE) obtains the final low-dimensional representations by applying eigen-analysis to a matrix which is built by estimating the Hessian over neighborhood. All these algorithms have the out of sample problem and thus a dozen of linearizations have been proposed, for example, locality preserving projections (LPP) [20] and discriminative locality alignment (DLA) [55]. Recently, we provide a systematic framework, that is, patch alignment [55], for understanding the common properties and intrinsic difference in different algorithms including their linearizations. In particular, this framework reveals that: i) algorithms are intrinsically different in the patch optimization stage; and ii) all algorithms share an almost-identical whole alignment stage. Another unified view of popular manifold learning algorithms is the graph embedding framework [48]. It is shown that manifold learning algorithms are more effective than conventional dimension reduction algorithms, for example, PCA and FLDA, in exploiting local geometry information.

In contrast to conventional dimension reduction algorithms that obtain a low dimensional subspace with each basis being a linear combination of all the original high dimensional features, sparse dimension reduction algorithms [9, 24, 59] select bases composed by only a small number of features of the high dimensional space. The sparse subspace is more interpretable both psychologically and physiologically. One popular sparse dimension reduction algorithm is sparse PCA, which generalizes the standard PCA by imposing sparsity constraint on the basis of the low dimensional subspace. The Manifold elastic net (MEN) [56] proposed recently is another sparse dimension reduction algorithm. It obtains a sparse projection matrix by imposing the elastic net penalty (i.e., the combination of the lasso penalty and the  $L_2$ -norm penalty) over the loss (i.e., the criterion) of a discriminative manifold learning, and formulates the problem as lasso which can be efficiently solved. In sum, sparse learning has many advantages, because (1) sparsity can make the data more succinct and simpler, so the calculation of the low dimensional representation and the subsequent recognition becomes more efficient. Parsimony is especially important for large scale face recognition systems; (2) sparsity can control the weights of original variables and decrease the variance brought by possible over-fitting with the least increment of the bias. Therefore, the learn model can generalize better and obtain high recognition rate for distorted face images; and (3) sparsity provides a good interpretation of a model, thus reveals an explicit relationship between the objective of the model and the given variables. This is important for understanding face recognition.

One fundamental assumption in face recognition, including dimension reduction, is that the training and test samples are independent and identically distributed

(i.i.d.) [22, 31, 38]. It is, however, very possible that this assumption does not hold, for example, the training and test face images are captured under different expressions, postures or lighting conditions, letting alone test subjects do not even appear in the training set [38]. Transfer learning has emerged as a new learning scheme to deal with such problem. By properly utilizing the knowledge obtained from the auxiliary domain task (training samples), it is possible to boost the performance on the target domain task (test samples). The idea of cross domain knowledge transfer was also introduced to subspace learning [31, 38]. It has shown that by using transfer subspace learning, the recognition performance on the cases where the face images in training and test sets are not identically distributed can be significantly improved compared with comparison against conventional subspace learning algorithms.

The rest of this chapter presents three groups of dimension reduction algorithms for face recognition. Specifically, Sect. 3.2 presents the general mean criterion and the max-min distance analysis (MMDA). Section 3.3 is dedicated to manifold learning algorithms, including the discriminative locality alignment (DLA) and manifold elastic net (MEN). The transfer subspace learning framework is presented in Sect. 3.4. In all of these sections, we first present principles of algorithms and then show thorough empirical studies.

## 3.2 Subspace Learning—A Global Perspective

Fisher's linear discriminant analysis (FLDA) is one of the most well-known methods for linear subspace selection, and has shown great value in subspace based face recognition. Being developed by Fisher [14] for binary-class classification and then generalized by Rao [33] for multiple-class tasks, FLDA utilizes the ratio of the between-class to within-class scatter as a definition of discrimination. It can be verified that under the homoscedastic Gaussian assumption, FLDA is Bayes optimal [18] in selecting a  $c - 1$  dimensional subspace, wherein  $c$  is the class number. Suppose there are  $c$  classes, represented by homoscedastic Gaussians  $N(\mu_i, \Sigma | \omega_i)$  with the prior probability  $p_i$ ,  $1 \leq i \leq c$ , where  $\mu_i$  is the mean of class  $\omega_i$  and  $\Sigma$  is the common covariance. The Fisher's criterion is given by [15]

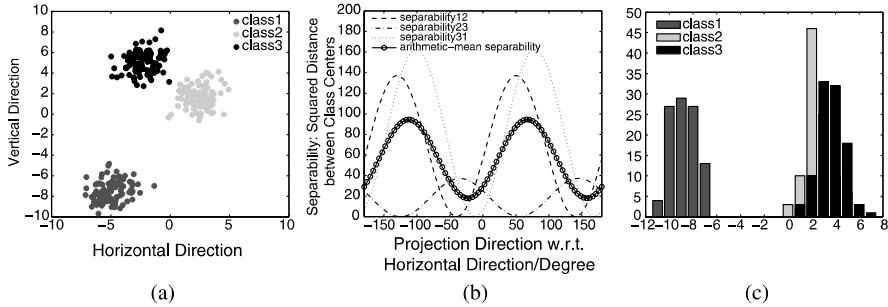
$$\max_W \text{tr}((W^T \Sigma W)^{-1} W^T S_b W) \quad (3.1)$$

where

$$S_b = \sum_{i=1}^c p_i (\mu_i - \mu)(\mu_i - \mu)^T, \quad \text{with } \mu = \sum_{i=1}^c p_i \mu_i. \quad (3.2)$$

It has been pointed out that the Fisher's criterion implies the maximization of the arithmetic mean of the pairwise distances between classes in the subspace. To see this, let us first define the distance between classes  $\omega_i$  and  $\omega_j$  in the subspace  $W$  as

$$\Delta(\omega_i, \omega_j | W) = \text{tr}((W^T \Sigma W)^{-1} W^T D_{ij} W), \quad \text{with } D_{ij} = (\mu_i - \mu_j)(\mu_i - \mu_j)^T. \quad (3.3)$$



**Fig. 3.1** An illustrative example on the class separation problem of FLDA. **a** 2-dimensional scatter plot of three classes, **b** plots of pairwise separabilities and the arithmetic mean (FLDA) separability verse projection directions, from  $-180$  degree to  $180$  degree with respect to horizontal direction in **(a)**, and **c** shows the histogram of three classes projected onto the FLDA direction, which is around  $66$  degree

Then, simple algebra shows that (3.1) is equivalent to the arithmetic mean criterion below

$$\max_W A(W) = \sum_{\substack{1 \leq i < j \leq c}} p_i p_j \Delta(\omega_i, \omega_j | W). \quad (3.4)$$

We call it arithmetic mean based subspace selection (AMSS). Since the arithmetic mean of all pairwise distance is used as the criterion, one apparent disadvantage of (3.4) is that it ignores the major contributions of close class pairs to classification error and may cause the merge of those class pairs in the selected subspace. Such phenomenon of FLDA or AMSS is called the class separation problem [39].

Figure 3.1 illustrates the class separation problem of FLDA [5]. In the toy example, three class are represented by homoscedastic Gaussian distributions on the two dimensional space. And we want to find a one dimensional subspace (or projection direction) such that the three classes can be well separated. Varying the one dimensional subspace, that is, changing the angle of projection direction with respect to the horizontal direction, the three pairwise distances change. FLDA finds the subspace that maximizes the average of the three pairwise distances. However, as illustrated, the obtained one dimensional subspace by FLDA severely merges the blue and green classes.

### 3.2.1 General Mean Criteria

To improve the separation between close class pairs, the general mean criteria has been proposed by Tao et al., of which two examples are the geometric mean based subspace selection (GMSS) [39] and the harmonic mean based subspace selection

(HMSS) [3]

$$\max_W G(W) = \prod_{\substack{1 \leq i < j \leq c}} \Delta(\omega_i, \omega_j | W)^{(p_i p_j)} \quad (\text{GMSS}) \quad (3.5)$$

and

$$\max_W H(W) = \left[ \sum_{1 \leq i < j \leq c} \frac{p_i p_j}{\Delta(\omega_i, \omega_j | W)} \right]^{-1} \quad (\text{HMSS}). \quad (3.6)$$

We give an mathematical analysis to interpret how criteria (3.5) and (3.6) work in dealing with the class separation problem, and why criterion (3.6) is even better than criterion (3.5). Consider a general criterion below

$$\max_W J(W) = f(\Delta(\omega_1, \omega_2 | W), \Delta(\omega_1, \omega_3 | W), \dots, \Delta(\omega_{c-1}, \omega_c | W)). \quad (3.7)$$

In order to reduce the class separation problem, the objective  $J(W)$  must has the ability to balance all the pairwise distances. We claim that this ability relies on the partial derivative of  $J(W)$  with respect to the pairwise distances. Apparently, an increment of any  $\Delta(\omega_i, \omega_j | W)$  will enlarge  $J(W)$ , and for this an small one should have bigger inference, because from the classification point of view when the distance between two classes is small then any increment of the distance will significantly improve the classification accuracy, but when the distance is large enough then the improvement of accuracy will be ignorable (it is well known that for Gaussian distribution the probability out the range of  $\pm 3\sigma$  is less than 0.01%). Besides, the partial derivatives must vary as the varying of the pairwise distances so as to take account of the current values of the pairwise distances in the procedure of subspace selection, but not only the initial distances in the original high dimensional space. According to the discussion above, the partial derivatives must be monotone decreasing functions of  $\Delta(\omega_i, \omega_j | W)$ . In the cases of criteria (3.4) and (3.5), we set  $J(W) = \log G(W)$  and  $J(W) = -H^{-1}(W)$ , and then the derivatives are calculated as below

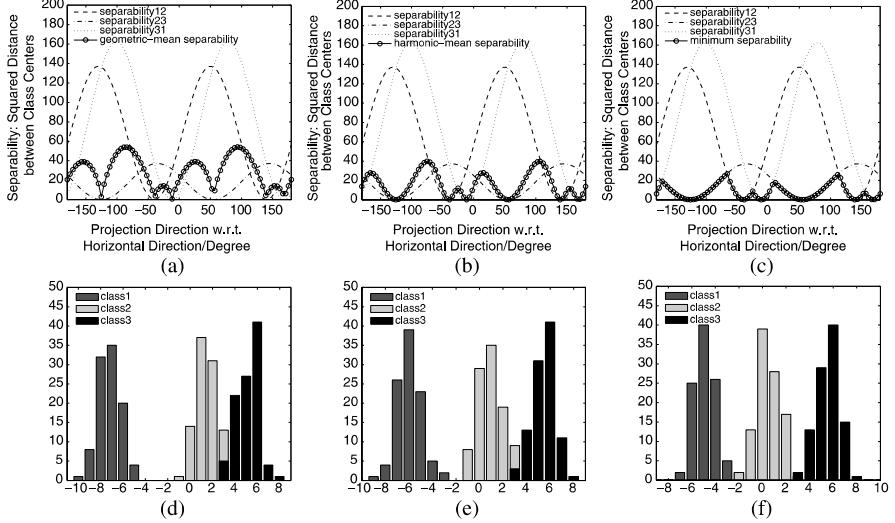
$$\frac{\partial \log G(W)}{\partial \Delta(\omega_i, \omega_j | W)} = \frac{q_i q_j}{(\Delta(\omega_i, \omega_j | W))^{-1}} \quad (3.8)$$

and

$$\frac{\partial -H^{-1}(W)}{\partial \Delta(\omega_i, \omega_j | W)} = \frac{q_i q_j}{(\Delta(\omega_i, \omega_j | W))^{-2}}. \quad (3.9)$$

We can see that in both cases the partial derivative monotonically decreases with respect to the pairwise distance and thus provides the ability to reduce the class separation problem. However, note that the order of decreasing for HMSS is higher than that for GMSS (-2 vs -1), which implies that HMSS is more powerful than GMSS in reducing the class separation problem. Besides, as  $\Delta(\omega_i, \omega_j | W)$  increases, we have

$$\log(\Delta(\omega_i, \omega_j | W)) \rightarrow \infty \quad (3.10)$$



**Fig. 3.2** GMSS, HMSS and MMDA for the same three-class problem in Fig. 3.1: first column, GMSS; second column, HMSS; third column, MMDA. Top row shows plots of pairwise separations and the separations by different criteria, i.e., GMSS, HMSS and MMDA. Bottom row shows the histograms of the three classes projected onto the GMSS, HMSS and MMDA directions, which are around 93 degree, 104 degree and 115 degree, respectively

but

$$-\Delta(\omega_i, \omega_j | W)^{-1} \rightarrow 0. \quad (3.11)$$

The logarithm value (3.10) is unbounded, and thus in GMSS a large pairwise distance still possibly affects small ones. In contrast, the bounded result (3.11) makes HMSS is more favorable. To solve the maximization problems of (3.5) and (3.6), [39] provides a gradient descent algorithm with a projection onto the orthogonal constraint set. Further, [3] suggests exploiting the structure of orthogonal constraint and optimizing the subspace on the Grassmann manifold [12]. For details of these optimization algorithms, please refer to [39] and [3]. The corresponding results of GMMS and HMSS on the illustrative example in Fig. 3.1 are shown in Fig. 3.2. One can see that the merged class pair in the FLDA subspace is better separated by using the more sophisticated methods.

### 3.2.2 Max–Min Distance Analysis

Previous discussions show that GMSS and HMSS are able to reduce the class separation problem of FLDA. Such merits come from the inherence of geometric or harmonic means in adaptively emphasizing small pairwise distance between classes. A further question is: can we select a subspace that mostly considers small pairwise distance? Namely, we may intend to find an optimal subspace which gives the

maximized minimum pairwise distance. Generally, such aim cannot be achieved by GMSS or HMSS, neither other subspace selection methods. To this end, [5] proposed the max-min distance analysis (MMDA) criterion,

$$\max_W \min_{1 \leq i < j \leq c} \Delta(\omega_i, \omega_j | W) \quad (3.12)$$

where the inner minimization chooses the minimum pairwise distance of all class pairs in the selected subspace, and the outer maximization maximizes this minimum distance. Let the optimal value and solution of (3.12) be  $\Delta_{\text{opt}}$  and  $W_{\text{opt}}$ , and then we have

$$\Delta(\omega_i, \omega_j | W_{\text{opt}}) \geq \Delta_{\text{opt}}, \quad \text{for all } i \neq j, \quad (3.13)$$

which ensures the separation (as best as possible) of any class pairs in the selected low dimensional subspace. Furthermore, by taking the prior probability of each class into account, the MMDA criterion is given by

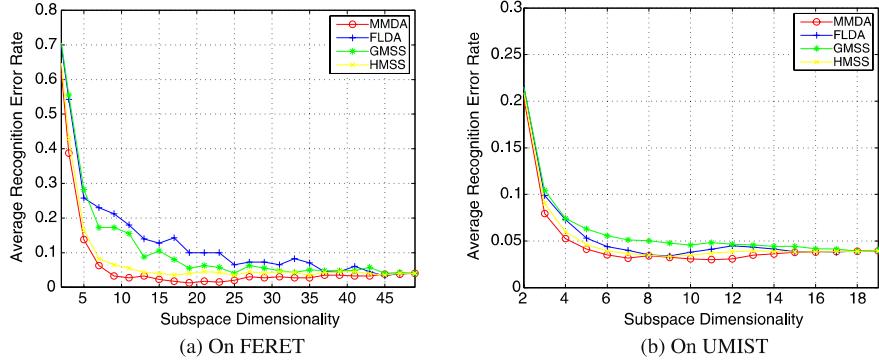
$$\max_W \min_{1 \leq i < j \leq c} \{(p_i p_j)^{-1} \Delta(\omega_i, \omega_j | W)\}. \quad (3.14)$$

Note that, the use of  $(p_i p_j)^{-1}$  as weighting factor is an intuitive choice. In order to obtain a relatively high accuracy, it has to put more weight on classes with high prior probabilities; however, because the minimization in the max-min operation has a negative effect, we need to put a smaller factor, for example, the inverse factor  $(p_i p_j)^{-1}$ , on the pairwise distance between high-prior probability classes so that it has a greater chance to be maximized.

The solving of MMDA criteria (3.12) and (3.14) can be difficult. The inner minimizations there are over discrete variables  $i$  and  $j$ , and thus it makes the objective function for the outer maximization nonsmooth. To deal with this nonsmooth max-min problem [5] introduced the convex relaxation technique. Specifically, the authors proposed a sequential semidefinite programming (SDP) relaxation algorithm, with which an approximate solution of (3.12) or (3.14) can be obtained in polynomial time. Refer to [5] for details of the algorithm. The MMDA result on the illustrative example in Fig. 3.1 is shown in Fig. 3.2, from which one can see that MMDA gives the best separation between blue and green classes among the four criteria.

### 3.2.3 Empirical Evaluation

The evaluation of general mean criteria, including GMSS and HMSS, and the MMDA are conducted on two benchmark face image datasets, UMIST [1] and FERET [32]. The UMIST database consists of 564 face images from 20 individuals. The individuals are a mix of race, sex and appearance and are photographed in a range of poses from profile to frontal views. The FERET database contains 13 539 face images from 1565 subjects, with varying pose, facial expression and



**Fig. 3.3** Face recognition by subspace selection and nearest neighbor classification in the selected low dimensional subspace

age. 50 subjects with 7 images for each are used in the evaluation. Images from both databases are cropped with reference to the eyes, and normalized to 40 by 40 pixel arrays with 256 gray levels per pixel. On UMIST, 7 images for each subject are used for training and the rest images are used for test, while on FERET, a 6 to 1 split is used for training/test setup. The average recognition performances over ten random trials are shown in Fig. 3.3. One can see that, on FERET, the general mean criterion (GMSS and HMSS) and MMDA show significant improvements on recognition rate compared with FLDA, while on UMIST, though GMSS gives slight inferior performance to FLDA, HMSS and MMDA still improve the performance in certain extent.

### 3.2.4 Related Works

In addition to the general mean criteria and max-min distance analysis, there are also some methods proposed in recent years to deal with the class separation problem of FLDA. Among these methods, approximate pairwise accuracy criterion (aPAC) [28] and fractional step LDA (FS-LDA) [30] are the most representative ones, and both of them use weighting schemes to emphasize close class pairs during subspace selection. Besides, the Bayes optimality of FLDA is further studied when the dimensionality of subspace is less than class number minus 1. In particular, it is shown that the one dimensional Bayes optimal subspace can be obtained by convex optimization given the information of the order of class centers projected onto the subspace [18]. Such result generalizes the early result of Bayes optimal one dimensional Bayes optimal subspace on a special case of three Gaussian distributions [36]. Further, the authors of [18] suggested selecting a general subspace by greedy one dimensional subspace selection and orthogonal projection. The homoscedastic Gaussian assumption is another limitation of FLDA. Various methods have been developed to extend FLDA to heteroscedastic Gaussian cases, e.g., the using of information theoretic divergences such as Kullback–Leibler divergence [10,

[39], and Chernoff [29] or Bhattacharyya distance [35] to measures the discrimination among heteroscedastic Gaussian distributions. Besides, nonparametric and semiparametric method provide alternative ways for extensions of FLDA, by which classic work includes Fukunaga’s nonparametric discriminant analysis (NDA) [16], its latest extension to multiclass case [27] and subclass discriminant analysis [57]. In addition, recent studies show that FLDA can be converted to a least square problem via a proper coding of class labels [49, 50]. The advantages of such least square formulation are that the computational speed can be significantly improved and also regularizations on the subspace are more readily imposed.

### 3.3 Subspace Learning—A Local Perspective

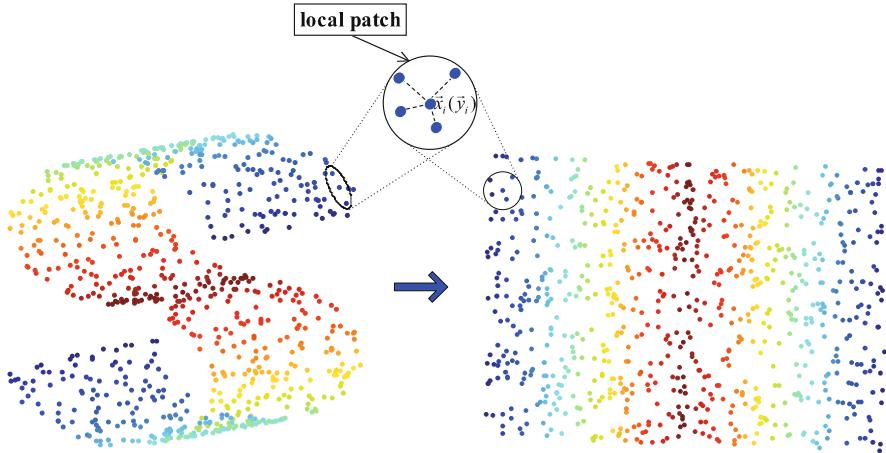
It has shown that the global linearity of PCA and FLDA prohibit their effectiveness for non-Gaussian distributed data, such as face images. By considering the local geometry information, a dozen of manifold learning algorithms have been developed, such as locally linear embedding (LLE) [34], ISOMAP [40], Laplacian eigenmaps (LE) [2], Hessian eigenmaps (HLE) [11], and local tangent space alignment (LTSA) [53]. All of these algorithms have been developed intuitively and pragmatically, that is, on the base of the experience and knowledge of experts for their own purposes. Therefore, it will be more informative to provide some a systematic framework for understanding the common properties and intrinsic differences in the algorithms. In this section, we introduce such a framework, that is, “patch alignment”, which consists of two stages: part optimization and whole alignment. The framework reveals (i) that algorithms are intrinsically different in the patch optimization stage and (ii) that all algorithms share an almost identical whole alignment stage.

#### 3.3.1 Patch Alignment Framework

The patch alignment framework [55] is composed of two ingredients, first, part optimization and then whole alignment. For part optimization, different algorithms have different optimization criteria over patches, each of which is built by one measurement associated with its related ones. For whole alignment, all part optimizations are integrated into together to form the final global coordinate for all independent patches based on the alignment trick. Figure 3.4 illustrates the patch alignment framework.

Given an instance  $x_i$  and its  $k$  nearest neighbors  $[x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(k)}]$ , the part optimization at  $x_i$  is defined by

$$\arg \min_{Y_i} \text{tr}(Y_i L_i Y_i^T) \quad (3.15)$$



**Fig. 3.4** Patch alignment framework

where  $Y_i = [y_i, y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(k)}]$  is projection of the local patch  $X_i = [x_i, x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(k)}]$  onto the low dimensional subspace, and  $L_i$  encodes the local geometry information at instance  $x_i$  and is chosen algorithm-specifically. By summarizing part optimizations over all instances, we get

$$\arg \min_{Y_1, Y_2, \dots, Y_n} \sum_{i=1}^n \text{tr}(Y_i L_i Y_i^T). \quad (3.16)$$

Let  $Y = [y_1, y_2, \dots, y_n]$  be the projection of all instances  $X = [x_1, x_2, \dots, x_n]$ . As for each local patch  $Y_i$  should be a subset of the whole alignment  $Y$ , the relationship between them can be expressed by

$$Y_i = Y S_i \quad (3.17)$$

where  $S_i$  is a proper 0-1 matrix called the selection matrix. Thus,

$$\begin{aligned} & \arg \min_Y \sum_{i=1}^N \text{tr}(Y_i L_i Y_i^T) \\ &= \arg \min_Y \sum_{i=1}^N \text{tr}(Y S_i L_i S_i^T Y^T) \\ &= \arg \min_Y \text{tr}(Y L Y^T) \end{aligned} \quad (3.18)$$

with

$$L = \left( \sum_{i=1}^N S_i L_i S_i^T \right) \quad (3.19)$$

**Table 3.1** Manifold learning algorithms filled in the patch alignment framework

Algorithm	Patch $X_i$	Representation of part optimization $L_i$	Objective function
LLE	Given instance and its neighbors	$\begin{bmatrix} 1 & -c_i^T \\ -c_i & c_i c_i^T \end{bmatrix}$	Nonlinear
NPE			Linear
ONPP			Orthogonal linear
ISOMAP	Given instance and the rest ones	$(1/N) \cdot \tau(D_G^i)$	Nonlinear
LE	Given instance and its connected ones in the undirected graph	$\begin{bmatrix} \sum_{j=1}^l (w_i)_j & -w_i^T \\ -w_i & \text{diag}(w_i) \end{bmatrix}$	Nonlinear
LPP			Linear
LTSA	Given instance and its neighbors	$R_{k+1} - V_i V_i^T$ , where $V_i$ denotes $d$ largest right singular vectors of $X_i R_{k+1}$	Nonlinear
LLTSA			Linear
	Given instance and its neighbors	$H_i H_i^T$	Nonlinear

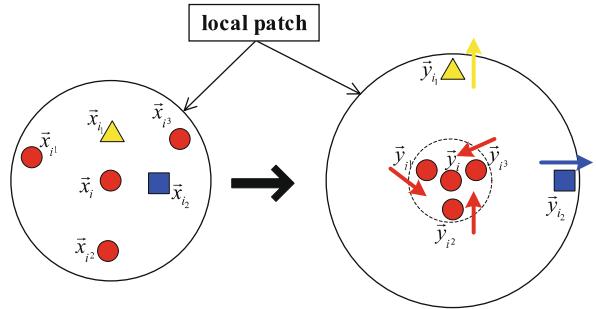
called the alignment matrix. Further by letting  $Y = U^T X$ , that is, a linear projection, (3.18) is rewritten as

$$\arg \min_U \text{tr}(U^T X L X^T U). \quad (3.20)$$

Further, we can impose the orthogonal constraint  $U^T U = I$  on the projection matrix  $U$ , or the constraint  $Y^T Y = I$  on the  $Y$ , which leads to  $U^T X X^T U = I$ . In both cases, (3.20) is solved by eigen- or generalized eigen-decomposition.

Among all the manifold learning algorithms, the most representatives are locally linear embedding (LLE) [34], ISOMAP [40], Laplacian eigenmaps (LE) [2]. LLE uses linear coefficients to represent local geometry information, and find a low-dimensional embedding such that these coefficients are still suitable for reconstruction. ISOMAP preserves geodesic distances between all instance pairs. And LE preserves proximity relationships by manipulations on an undirected weighted graph, which indicates neighbor relations of pairwise instances. It has been shown that all these algorithms can be filled into the patch alignment framework, where the difference among algorithms lies in the part optimization stage while the whole alignment stage is almost the same. There are also other manifold learning algorithms, for example, Hessian eigenmaps (HLLE) [11], Generative Topographic Mapping (GTM) [6] and local tangent space alignment (LTSA) [53]. We can use the patch alignment framework to explain them in a unified way. Table 3.1 summarizes these algorithms in the patch alignment framework.

**Fig. 3.5** The motivation of DLA. The measurements with the same shape and color come from the same class



### 3.3.2 Discriminative Locality Alignment

One representative subspace selection method based on the patch alignment framework is the discriminative locality alignment (DLA) [54]. In DLA, the discriminative information, encoded in labels of samples, is imposed on the part optimization stage and then the whole alignment stage constructs the global coordinate in the projected low-dimensional subspace.

Given instance  $x_i$  and its  $k$  nearest neighbors  $[x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(k)}]$ , we divide the  $k$  neighbors into two groups according to the label information, that is, belonging to the same class with  $x_i$  or not. Without losing generality, we can assume the first  $k_1$  neighbors  $[x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(k_1)}]$  having the same class label with  $x_i$  and the rest  $k - k_1$  neighbors  $[x_i^{(k_1+1)}, x_i^{(k_1+2)}, \dots, x_i^{(k)}]$  having different class labels (otherwise, we just have to resort the indexes properly). And their low dimensional representations are  $y_i$ ,  $[y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(k_1)}]$  and  $[y_i^{(k_1+1)}, y_i^{(k_1+2)}, \dots, y_i^{(k)}]$ , respectively. The key idea of DLA is enforcing  $y_i$  close to  $[y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(k_1)}]$  while pushing it apart from  $[y_i^{(k_1+1)}, y_i^{(k_1+2)}, \dots, y_i^{(k)}]$ . Figure 3.5 illustrates such motivation.

For instance,  $x_i$  and its same class neighbors, we expect the summation of squared distance in the low dimensional subspace to be as small as possible, that is,

$$\arg \min_{y_i} \sum_{p=1}^{k_1} \|y_i - y_i^{(p)}\|^2 \quad (3.21)$$

However, for  $x_i$  and its different class neighbors, we want the corresponding result to be large, that is,

$$\arg \max_{y_i} \sum_{p=k_1+1}^k \|y_i - y_i^{(p)}\|^2 \quad (3.22)$$

A convenient tradeoff between (3.21) and (3.22) is

$$\arg \min_{Y_i} \left( \sum_{p=1}^{k_1} \|y_i - y_i^{(p)}\|^2 - \gamma \sum_{p=k_1+1}^k \|y_i - y_i^{(p)}\|^2 \right) \quad (3.23)$$

where  $\gamma$  is a scaling factor between 0 and 1 to balance the importance between measures of the within-class distance and the between-class distance. Let

$$\omega_i = \left[ \overbrace{1, \dots, 1}^{k_1}, \overbrace{-\gamma, \dots, -\gamma}^{k-k_1} \right]^T, \quad (3.24)$$

then (3.23) is readily rewritten as

$$\arg \min_{Y_i} \text{tr}(Y_i L_i Y_i^T), \quad (3.25)$$

where

$$L_i = \begin{bmatrix} \sum_{j=1}^k \omega_j & -\omega_i^T \\ -\omega_i & \text{diag}(\omega_i) \end{bmatrix}. \quad (3.26)$$

To obtain the projection mapping  $y = U^T x$ , we just substitute (3.26) into the whole alignment formula (3.18), and solve the eigen-decomposition problem with constraint  $U^T U = I$ . It is worth emphasizing some merits of DLA here: (1) it exploits local geometry information of data distribution; (2) it is ready to deal with the case of nonlinear boundaries for class separation; (3) it avoids the matrix singularity problem.

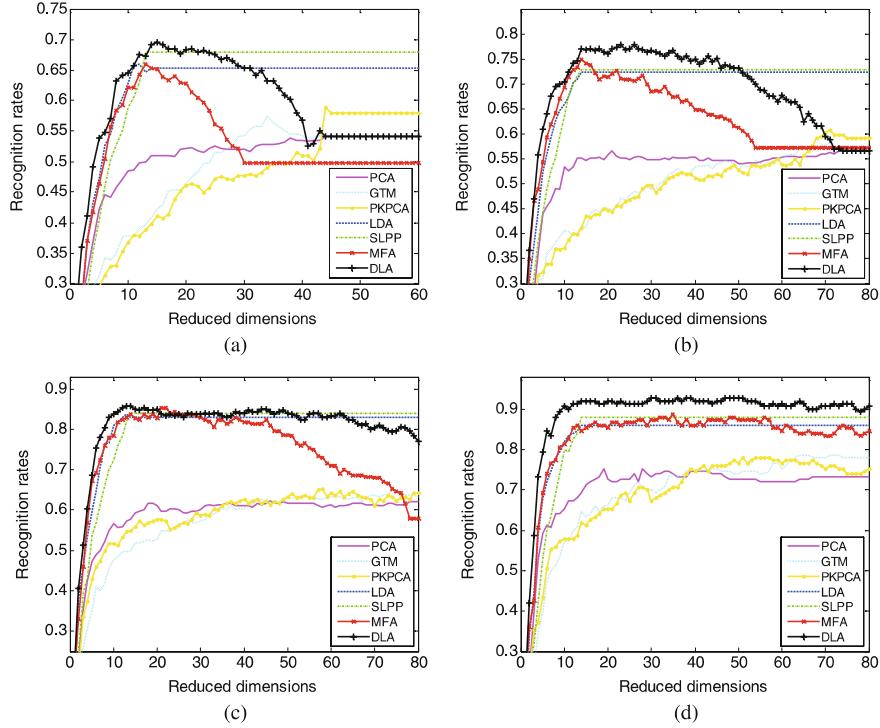
Now we evaluate the performance of the proposed DLA in comparison with six representative algorithms, that is, PCA [23], Generative Topographic Mapping (GTM) [6], Probabilistic Kernel Principal Components Analysis (PKPCA) [42], LDA [14], SLPP [7] and MFA [48], on Yale face image dataset [1]. For training, we randomly selected different numbers (3, 5, 7, 9) of images per individual, used 1/2 of the rest images for validation, and 1/2 of the rest images for testing. Such trial was independently performed ten times, and then the average recognition results were calculated. Figure 3.6 shows the average recognition rates versus subspace dimensions on the validation sets, which help to select the best subspace dimension. It can be seen that DLA outperforms the other algorithms.

### 3.3.3 Manifold Elastic Net

Manifold elastic net (MEN) [56] is a subspace learning method built upon the patch alignment framework. However, the key feature of MEN is that it is able to achieve sparse basis (projection matrix) by imposing the popular elastic net penalty (i.e., the combination of the lasso penalty and the L2 norm penalty). As sparse basis are more interpretable both psychologically and physiologically, MEN is expected to give more meaningful results on face recognition, which will be shown in experiments later.

First, MEN uses the same part optimization and whole alignment as in DLA, that is, the following minimization is considered

$$\arg \min_Y \text{tr}(Y L Y^T). \quad (3.27)$$



**Fig. 3.6** Recognition rate vs. subspace dimension on Yale dataset. **a** 3 images per subject for training; **b** 5 images per subject for training; **c** 7 images per subject for training; **d** 9 images per subject for training

However, rather than substituting  $Y = U^T X$  directly, (3.27) is reformed equivalently as below

$$\arg \min_{Y, U} \text{tr}(YLY^T) + \beta \|Y - U^T X\|^2. \quad (3.28)$$

Note that (3.28) indeed will lead to  $Y = U^T X$ . Given the equivalence between the two formulations, the latter is more convenient to incorporate the minimization of classification error. Specifically, letting stores the response or prediction result, which are proper encodings of the class label information, we expect  $U^T X$  to be close to  $T$ , that is,

$$\arg \min_U \|T - U^T X\|^2. \quad (3.29)$$

By combining (3.28) and (3.29), we get the main objective of MEN

$$\arg \min_{Y, U} \|T - U^T X\|^2 + \alpha \text{tr}(Y^T LY) + \beta \|Y - U^T X\|^2 \quad (3.30)$$

where  $\alpha$  and  $\beta$  are trade-off parameters to control the impacts of different terms.

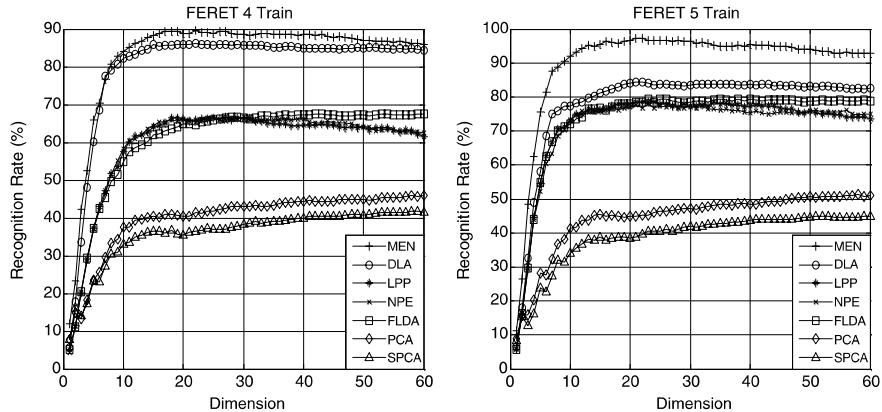
To obtain a sparse projection matrix  $U$ , an ideal approach is to restrict the number of nonzero entries in it, that is, using the  $L_0$  norm as a penalty over (3.30). However, the  $L_0$  norm penalized (3.30) is an NP-hard problem and thus intractable practically. One attractive way of approximating the  $L_0$  norm is the  $L_1$  norm, i.e., the Lasso penalty [41], which is convex and actually the closest convex relaxation of the  $L_0$  norm. Various efficient algorithms exist for solving Lasso penalized least square regression problem, including the LARS [13]. However, the lasso penalty has the following two disadvantages: (1) the number of variables to be selected is limited by the number of observations and (2) the lasso penalized model can only selects one variable from a group of correlated ones and does not care which one should be selected. These limitations of Lasso are well addressed by the so-called elastic net penalty, which combines the  $L_2$  and  $L_1$  norm together. MEN adopts the elastic net penalty [58]. In detail, the  $L_2$  of the projection matrix is helpful to increase the dimension (and the rank) of the combination of the data matrix and the response. In addition, the combination of the  $L_1$  and  $L_2$  of the projection matrix is convex with respect to the projection matrix and thus the obtained projection matrix has the grouping effect property. The final form of MEN is given by

$$\begin{aligned} \arg \min_{Y, U} & \|T - U^T X\|^2 + \alpha \operatorname{tr}(Y^T LY) + \beta \|Y - U^T X\|^2 \\ & + \lambda_1 \|U\|_1 + \lambda_2 \|U\|^2. \end{aligned} \quad (3.31)$$

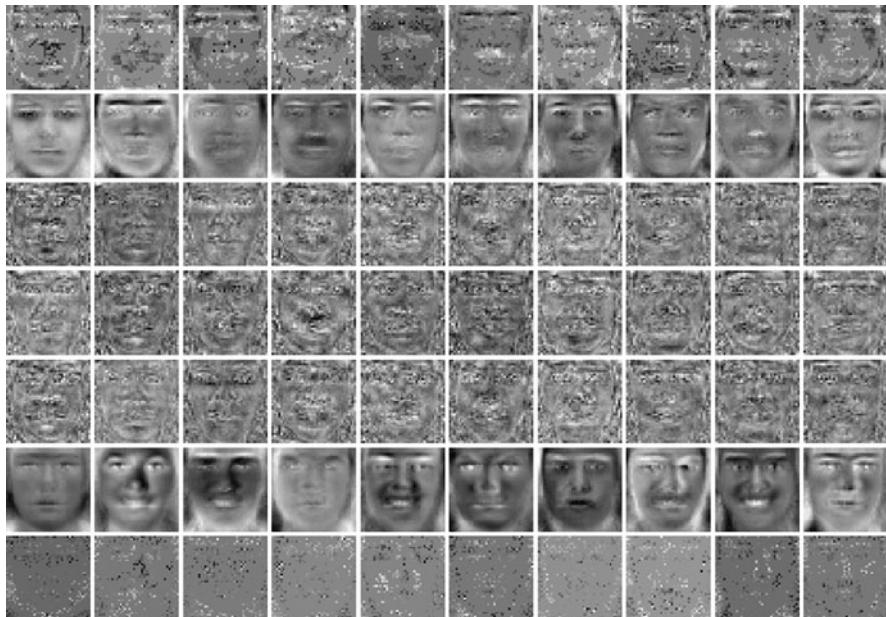
We report an empirical evaluation of MEN on the FERET dataset. From in total 13 539 face images of 1565 individuals, 100 individuals with 7 images per subject are randomly selected in the experiment. 4 or 5 images per individual are selected as training set, and the remaining is used for test. All experiments are repeated five times, and the average recognition rates are calculated. Six representative dimension reduction algorithms, that is, principal component analysis (PCA) [23], Fisher's linear discriminant analysis (FLDA) [14], discriminative locality alignment (DLA) [54], supervised locality preserving projection (SLPP) [7], neighborhood preserving embedding (NPE) [19], and sparse principal component analysis (SPCA) [9], are also performed for performance comparison.

The performance of recognition is summarized in Fig. 3.7. Apparently, the seven algorithms are divided into 3 groups according to their performance. The baseline level methods are PCA and SPCA, which is because they are both unsupervised methods and thus may not give satisfying performance due to the missing of label information. LPP, NPE and LDA only show moderate performance. In contrast, DLA and MEN give rise to significant improvements. Further, the sparsity of MEN makes it outperform DLA. The best performance of MEN is actually not surprising, since it considers the most aspects on data representation and distribution, including the sparse property, the local geometry information and classification error minimization.

Figure 3.8 shows the first ten bases selected by different subspace selection methods. One can see that the bases selected by LPP, NPE and FLDA are contaminated by considerable noises, which explains why they only give moderate recognition

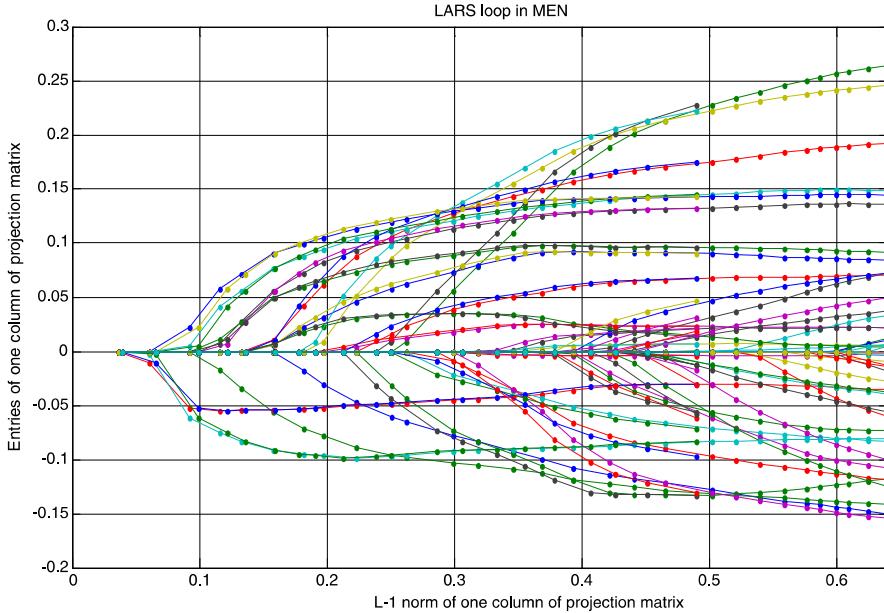


**Fig. 3.7** Performance evaluation on the FERET dataset



**Fig. 3.8** Plots of first 10 bases obtained from 7 dimensionality reduction algorithms on FERET for each column, from top to bottom: MEN, DLA, LPP, NPE, FLDA, PCA, and SPCA

performance. The bases from PCA, that is, Eigenfaces, are smooth but present relatively few discriminative information. In terms of sparsity, SPCA gives the desired bases; however, the problem is that the patterns presented in these bases are not grouped so that cannot provide meaningful interpretation. The bases from MEN, which we call “MEN’s faces”, have a low level of noise and are also reasonably sparse. And more importantly, thanks to the elastic net penalty, the sparse patterns

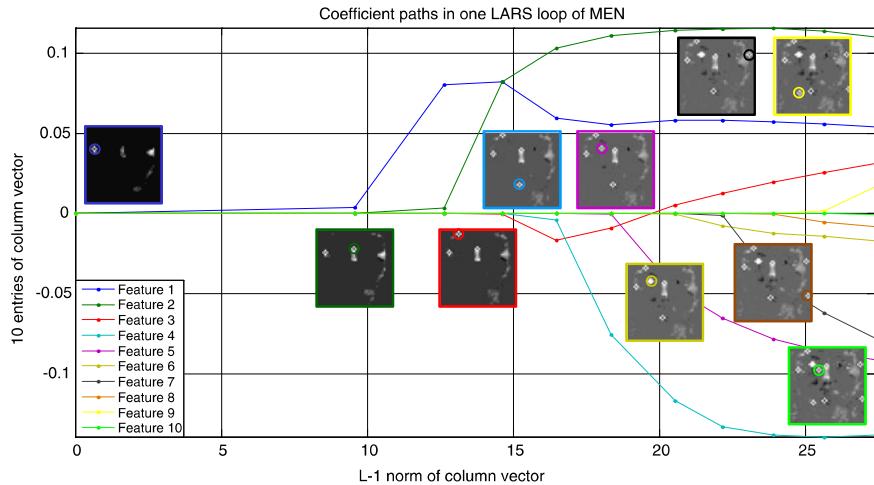


**Fig. 3.9** Entries of one column of projection matrix vs. its  $L_1$  norm in one LARS loop of MEN

of MEN’s bases are satisfying grouped, which gives meaningful interpretations, for example, most discriminative facial features are obtained, including eyebrows, eyes, nose, mouth, ears and facial contours.

The optimization algorithm of MEN is built upon LARS. In each LARS loop of the MEN algorithm, all entries of one column in the projection matrix are zeros initially. They are sequentially added into the active set according to their importance. The values of active ones are increased with equal altering correlation. In this process, the  $L_1$  norm of the column vector is augmented gradually. Figure 3.9 shows the altering tracks of some entries of the column vector in one LARS loop. These tracks are called “coefficient paths” in LARS. As shown by these plots, one can observe that every coefficient path starts from zero when the corresponding variable becomes active, and then changes its direction when another variable is added into the active set. All the paths keep in the directions which make the correlations of their corresponding variables equally altering. The  $L_1$  norm is increasing along the greedy augment of entries. The coefficient paths proceed along the gradient decent direction of objective function on the subspace, which is spanned by the active variables.

In addition, Fig. 3.10 shows 10 of the 1600 coefficient paths from LAPS loop. It can be seen that MEN selects ten important features sequentially. For each feature, its corresponding coefficient path and the “MEN face” when the feature is added into active set are assigned the same color which is different with the other 9 features. In each “MEN face”, the new added active feature is marked by a small circle, and all the active features are marked by white crosses. The features selected by



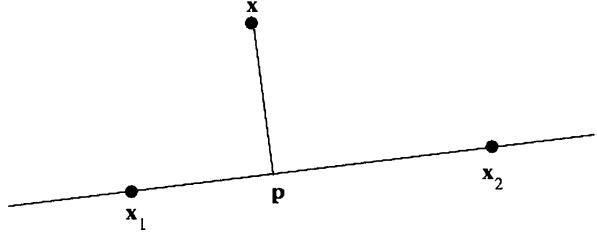
**Fig. 3.10** Coefficient paths of 10 features in one column vector

MEN can produce explicit interpretation of the relationship between facial features and face recognition: feature 1 is the left ear, feature 2 is the top of nose, feature 3 is on the head contour, feature 4 is the mouth, feature 5 and feature 6 are on the left eye, feature 7 is the right ear, and feature 8 is the left corner of mouth. These features are already verified of great importance in face recognition by many other famous face recognition methods. Moreover, Fig. 3.10 also shows MEN can group correlated features, for example, feature 5 and feature 6 are selected sequentially because they are both on the left eye. In addition, features which are not very important, such as feature 9 and feature 10 in Fig. 3.10, are selected after the selection of the other more significant features and assigned smaller value than those more important ones. Therefore, MEN is a powerful algorithm in feature selection.

### 3.3.4 Related Works

Applying the idea of manifold learning, that is, exploring local geometry information of data distribution, into semisupervised or transductive subspace selection leads to a new framework of dimension reduction by manifold regularization. One example is the recently proposed manifold regularized sliced inverse regression (MRSIR) [4]. Sliced inverse regression (SIR) was proposed for sufficient dimension reduction. In a regression setting, with the predictors  $X$  and the response  $Y$ , the sufficient dimension reduction (SDR) subspace  $B$  is defined by the conditional independency  $Y \perp X | B^T X$ . Under the assumption that the distribution of  $X$  is elliptic symmetric, it has been proved that the SDR subspace  $B$  is related to the inverse regression curve  $E(X | Y)$ . It can be estimated at least partially by a generalized eigendecomposition between the covariance matrix of the predictors  $\text{Cov}(X)$  and

**Fig. 3.11** Point  $\mathbf{p}$  is the projection of query  $\mathbf{x}$  onto the feature line  $\overline{\mathbf{x}_1 \mathbf{x}_2}$



the covariance matrix of the inverse regression curve  $\text{Cov}(E(X | Y))$ . If  $Y$  is discrete, this is straightforward. While  $Y$  is continuous, it is discretized by slicing its range into several slices so as to estimate  $E(X | Y)$  at each slice.

Suppose  $\Gamma$  and  $\Sigma$  are respectively the empirical estimates of  $\text{Cov}(E(X | Y))$  and  $\text{Cov}(X)$  based on a training data set. Then, the SDR subspace  $B$  is given by

$$\max_B \text{trace}((B^T \Sigma B)^{-1} B^T \Gamma B). \quad (3.32)$$

To construct the manifold regularization, [4] uses the graph Laplacian  $L$  of the training data  $X = [x_1, x_2, \dots, x_n]$ . Letting  $Q = \frac{1}{n(n-1)} X L X^T$  and  $S = \frac{1}{n(n-1)} X D X^T$ , with  $D$  being the degree matrix, then MRSIR is defined by

$$\max_B \text{trace}((B^T \Sigma B)^{-1} B^T \Gamma B) - \eta \text{trace}((B^T S B)^{-1} B^T Q B), \quad (3.33)$$

where  $\eta$  is a positive weighting factor. The use of manifold regularization extends SIR in many ways, that is, it utilizes the local geometry that is ignored originally and enables SIR to deal with the transductive/semisupervised subspace selection problems.

So far we have introduced subspace selection methods that exploit local geometry information of data distribution. Based on these methods, classification can be performed in the low dimensional embedding. However, as the final goal is classification, an alternative approach is to do classification directly using the local geometry information. This generally leads to nonparametric classifiers, for example, nearest neighbor (NN) classifier. The problem is that simple NN classifier cannot provide satisfying recognition performance when data are of very high dimensions as in face recognition. To this end, Li and Liu proposed the nearest feature line (NFL) for face recognition [25, 26]. In NFL, a query is projected onto a line segment between any two instances within each class, and the nearest distance between the query and the projected point is used to determine its class label. Figure 3.11 shows an example of projecting a query  $\mathbf{x}$  onto the feature line spanned by instances  $\mathbf{x}_1$  and  $\mathbf{x}_2$ , where the projected point  $\mathbf{p}$  is given by

$$\mathbf{p} = \mathbf{x}_1 + \mu * (\mathbf{x}_2 - \mathbf{x}_1), \quad (3.34)$$

with

$$\mu = \frac{(\mathbf{x} - \mathbf{x}_1)^T (\mathbf{x}_2 - \mathbf{x}_1)}{\|\mathbf{x}_2 - \mathbf{x}_1\|^2}. \quad (3.35)$$

One extension of NFL is the nearest linear combination (NLC) [25]. There a query is projected onto a linear subspace spanned by a set of basis vectors, where the basis vectors can be any form from a subspace analysis or a set of local features, and the distance between the query and the projection point is used as the metric for classification. Empirical studies shown that NFL and NLC produces significantly better performance than the simple nearest neighborhood (NN) when the number of prototype templates (basis vectors representing the class) is small.

Another method related to NLC and NS approach is the sparse representation classifier (SRC) [47], which treats the face recognition problem as searching for an optimal sparse combination of gallery images to represent the probe one. SRC differs from the standard NLC in the norm used to define the projection distance. Instead of using the 2-norm as in NLC [25], SRC uses the 1- or 0-norm, such that the sparsity emerges.

## 3.4 Transfer Subspace Learning

Conventional algorithms including subspace selection methods are built under the assumption that training and test samples are independent and identically distributed (i.i.d.). For practical applications, however, this assumption cannot be hold always. Particularly, in face recognition, the difference of expressions, postures, aging problem and lighting conditions makes the distributions of training and test face different. To this end, a transfer subspace learning (TSL) framework is proposed [38]. TSL extends conventional subspace learning methods by using a Bregman divergence based regularization, which encourages the difference between the training and test samples in the selected subspace to be minimized. Thus, we can approximately assume the samples of training and test are almost i.i.d. in the learnt subspace.

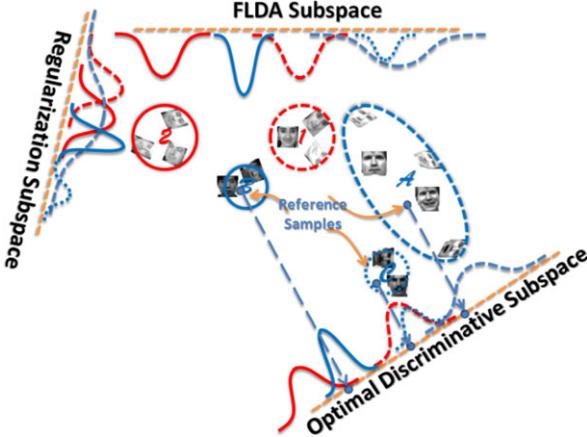
### 3.4.1 TSL Framework

The TSL framework [38] is presented by the following unified form

$$\arg \min_U F(U) + \lambda D_U(P_l || P_u) \quad (3.36)$$

where  $F(U)$  is the objective function of a subspace selection method, for example, FLDA or PCA et al., and  $D_U(P_l || P_u)$  is the Bregman divergence between the training data distribution  $P_l$  and the test data distribution  $P_u$  in the low dimension subspace  $U$ , and parameter  $\lambda$  controls the balance between the objective function and the regularization. Note that generally the objective function  $F(U)$  only depends on the training data.

For example, when  $F(U)$  is chosen to be FLDA's objective, (32) will give a subspace in which the training and test data distributions are close to each other and



**Fig. 3.12** Two classes of training samples are marked as 1 and 2, while three classes of test samples are marked as A, B and C. *Blue circles* A and C are merged together in the FLDA subspace, where discrimination of the training samples can be well preserved. *Blue circles* A and B are mixed in the regularization subspace, where there exists the smallest divergence between training domain (1, 2) and test domain (A, B and C). *Blue circles* A, B and C can be well separated in the discriminative subspace, which is obtained by optimizing the combination of the proposed regularization (the divergence between training sets 1, 2 and test sets A, B, C) and FLDA

the discriminative information in the training data is partially preserved. In particular, suppose we have two classes of training samples, represented by two red circles (1 and 2, e.g., face images in the FERET dataset), and three classes of test samples, represented by three blue circles (A, B and C, e.g., face images in the YALE dataset), as shown in Fig. 3.12. On one hand, FLDA finds a subspace that fails to separate the test circle A from the test circle C, but the subspace is helpful to distinct different subjects in the training set. On the other hand, the minimization of the Bregman divergence between training and test distributions would give a subspace that makes the training data and test data almost i.i.d., but give little discriminative power. Apparently, neither of them individually can find a best discriminative subspace for test. However, as shown in the figure, a combination of FLDA and the Bregman regularization does find the optimal subspace for discrimination, wherein A, B and C can be well separated and samples in them can be correctly classified with given references. It is worth emphasizing that the combination works well because the training and test samples are coming from different domains but both domains share some common properties.

The authors suggest solving (3.36) by gradient descent method [38],

$$U \leftarrow U - \tau \left( \frac{\partial F(U)}{\partial U} + \lambda \frac{\partial D_U(P_l || P_u)}{\partial U} \right) \quad (3.37)$$

where  $\tau$  is the learning rate, that is, step size for updating. As  $F(U)$  is usually known, so is its derivative. The problem remaining is how to estimate  $D_U(P_l || P_u)$  and its derivatives.

**Definition 1** (Bregman divergence regularization) Let  $f : S \rightarrow R$  be a convex function defined on a closed convex set  $S \in R^+$ . We denote the first order derivative of  $f$  as  $f'$ , whose inverse function as  $\xi = (f')^{-1}$ . The probability density for the training and test samples in the projected subspace  $U$  are  $p_l(y)$  and  $p_u(y)$  respectively, wherein  $y = U^T x$  is the low-dimensional representation of the sample  $x$ . The difference at  $\xi(p_l(y))$  between the function  $f$  and the tangent line to  $f$  at  $(\xi(p_l(y)), f(\xi(p_l(y))))$  is given by:

$$\begin{aligned} & d(\xi(p_l(y)), \xi(p_u(y))) \\ &= \{f(\xi(p_u(y))) - f(\xi(p_l(y)))\} - p_l(y)\{\xi(p_u(y)) - \xi(p_l(y))\}. \end{aligned} \quad (3.38)$$

Based on (3.38), the Bregman divergence regularization, which measures the distance between  $p_l(y)$  and  $p_u(y)$ , is a convex function given by

$$D_U(P_l || P_u) = \int d(\xi(p_l(y)), \xi(p_u(y))) d\mu \quad (3.39)$$

where  $d\mu$  is the Lebesgue measure.

By taking a special form  $f(y) = y^2$ ,  $D_U(P_l || P_u)$  can be expressed as [38]

$$\begin{aligned} & D_W(P_l || P_u) \\ &= \int (p_l(y) - p_u(y))^2 dy \\ &= \int (p_l(y)^2 - 2p_l(y)p_u(y) + p_u(y)^2) dy. \end{aligned} \quad (3.40)$$

Further, the kernel density estimation (KDE) technique is used to estimate  $p_l(y)$  and  $p_u(y)$ . Suppose there are  $n_l$  training instances  $\{x_1, x_2, \dots, x_{n_l}\}$  and  $n_u$  test instances  $\{x_1, x_2, \dots, x_{n_l}\}$ , then through projection  $y_i = U^T x_i$ , we have the estimates [38]

$$p_l(y) = (1/n_l) \sum_{i=1}^{n_l} G_{\Sigma_1}(y - y_i) \quad (3.41)$$

and

$$p_u(y) = (1/n_u) \sum_{i=n_l+1}^{n_l+n_u} G_{\Sigma_2}(y - y_i) \quad (3.42)$$

where  $G_{\Sigma_1}(y)$  is a Gaussian kernel with covariance  $\Sigma_1$ , so is  $G_{\Sigma_2}(y)$ . With these estimates, the quadratic divergence (3.40) is rewritten as [38]

$$\begin{aligned}
D_W(P_l || P_u) = & \frac{1}{n_l^2} \sum_{s=1}^{n_l} \sum_{t=1}^{n_l} G_{\Sigma_{11}}(y_t - y_s) + \frac{1}{n_u^2} \sum_{s=n_l+1}^{n_l+n_u} \sum_{t=n_l+1}^{n_l+n_u} G_{\Sigma_{22}}(y_t - y_s) \\
& - \frac{2}{n_l n_u} \sum_{s=1}^{n_l} \sum_{t=n_l+1}^{n_l+n_u} G_{\Sigma_{12}}(y_t - y_s)
\end{aligned} \tag{3.43}$$

where  $\Sigma_{11} = \Sigma_1 + \Sigma_1$ ,  $\Sigma_{12} = \Sigma_1 + \Sigma_2$  and  $\Sigma_{22} = \Sigma_2 + \Sigma_2$ . Further, by basis matrix calculus, we have

$$\begin{aligned}
\frac{\partial D_U(P_l || P_u)}{\partial U} = & \frac{2}{n_l^2} \sum_{i=1}^{n_l} \sum_{t=1}^{n_l} G_{\Sigma_{11}}(y_i - y_t) (\Sigma_{11})^{-1}(y_t - y_i) x_i^T \\
& - \frac{2}{n_l n_u} \sum_{i=1}^{n_l} \sum_{t=n_l+1}^{n_l+n_u} G_{\Sigma_{12}}(y_i - y_t) (\Sigma_{12})^{-1}(y_t - y_i) x_i^T \\
& + \frac{2}{n_u^2} \sum_{i=n_l+1}^{n_l+n_u} \sum_{t=n_l+1}^{n_l+n_u} G_{\Sigma_{22}}(y_i - y_t) (\Sigma_{22})^{-1}(y_t - y_i) x_i^T \\
& - \frac{2}{n_l n_u} \sum_{i=l+1}^{n_l+n_u} \sum_{t=1}^{n_l} G_{\Sigma_{12}}(y_i - y_t) (\Sigma_{12})^{-1}(y_t - y_i) x_i^T. \quad (3.44)
\end{aligned}$$

### 3.4.2 Cross Domain Face Recognition

Based on the YALE, UMIST and a subset of FERET datasets, cross-domain face recognition is performed by applying the TSL framework. In detail, we have (1) Y2F: the training set is on YALE and the test set is on FERET; (2) F2Y: the training set is on FERET and the test set is on YALE; and (3) YU2F: the training set is on the combination of YALE and UMIST and the test set is on FERET. In the training stage, the labeling information of test images is blind to all subspace learning algorithms. However, one reference image for each test class is preserved so that the classification can be done in the test stage. The nearest neighbor classifier is adopted for classification, i.e., we calculate the distance between a test image and every reference image and predict the label of the test image as that of the nearest reference image.

We compare TSL algorithms, for example, TPCA, TFLDA, TLPP, TMFA, and TDLA, with conventional subspace learning algorithms, for example, PCA [23], FLDA [14], LPP [20], MFA [48], DLA [54] and the semi-supervised discriminant analysis (SDA) [8]. Table 3.2 shows the recognition rate of each algorithm with the corresponding optimal subspace dimension. In detail, conventional subspace learning algorithms, for example, FLDA, LPP and MFA, perform poorly because they assume training and test samples are i.i.d. variables and this assumption is unsuitable for cross-domain tasks. Although SDA learns a subspace by taking test samples into account, it assumes samples in a same class are drawn from an identical

**Table 3.2** Recognition rates of different algorithms under three experimental settings.

The number in the parenthesis is the corresponding subspace dimensionality

	Y2F	F2Y	YU2F
LDA	39.71(70)	36.36(30)	29.57(30)
LPP	44.57(65)	44.24(15)	45.00(35)
MFA	40.57(65)	34.54(60)	27.85(70)
DLA	50.43(80)	50.73(15)	50.86(65)
SDA	44.42(65)	41.81(40)	32.00(35)
MMDR	45.60(60)	42.00(75)	49.75(80)
TLDA	57.28(15)	50.51(20)	55.57(45)
TLPP	58.28(30)	53.93(25)	58.42(30)
TMFA	63.14(70)	56.96(35)	65.42(70)
TDLA	63.12(60)	61.82(30)	65.57(70)

underlying manifold. Therefore, SDA is not designed for the cross-domain tasks. Although MMDR considers the distribution bias between the training and the test samples, it ignores the discriminative information contained in the training samples. We have given an example in the synthetic data test to show that the training discriminative information is helpful to separate test classes. Example TSL algorithms perform consistently and significantly better than others, because the training discriminative information can be properly transferred to test samples by minimizing the distribution distance between the training and the test samples. In particular, TDLA performs best among all TSL examples because it inherits the merits of DLA in preserving both the discriminative information of different classes and the local geometry of samples in an identical class.

**Acknowledgements** The authors thank Prof. Stan Z. Li for insightful discussions on nearest feature line.

## References

1. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 711–720 (1997)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* **15**(6), 1373–1396 (2003)
3. Bian, W., Tao, D.: Harmonic mean for subspace selection. In: 19th International Conference on Pattern Recognition, pp. 1–4 (2008)
4. Bian, W., Tao, D.: Manifold regularization for sir with rate root-n convergence (2010)
5. Bian, W., Tao, D.: Max-min distance analysis by using sequential sdp relaxation for dimension reduction. *IEEE Trans. Pattern Anal. Mach. Intell.* **99**(PrePrints) (2010)
6. Bishop, C.M., Svensén, M., Williams, C.K.I.: GTM: The generative topographic mapping. Technical Report NCRG/96/015, Neural Computing Research Group, Dept of Computer Science & Applied Mathematics, Aston University, Birmingham B4 7ET, United Kingdom, April 1997

7. Cai, D., He, X., Han, J.: Using graph model for face analysis. Technical report, Computer Science Department, UIUC, UIUCDCS-R-2005-2636, September 2005
8. Cai, D., He, X., Han, J.: Srd: An efficient algorithm for large-scale discriminant analysis. *IEEE Trans. Knowl. Data Eng.* **20**(1), 1–12 (2008)
9. D’aspremont, A., Ghaoui, L.E., Jordan, M.I., Lanckriet, G.R.G.: A direct formulation for sparse PCA using semidefinite programming. *SIAM Rev.* **49**(3), 434–448 (2007)
10. Decell, H., Mayekar, S.: Feature combinations and the divergence criterion. *Comput. Math. Appl.* **3**(4), 71–76 (1977)
11. Donoho, D.L., Grimes, C.: Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proc. Natl. Acad. Sci. USA* **100**(10), 5591–5596 (2003)
12. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.* **20**, 303–353 (1998)
13. Efron, B., Hastie, T., Johnstone, I., Tibshirani, R.: Least angle regression. *Ann. Stat.* **32**, 407–499 (2004)
14. Fisher, R.A.: The use of multiple measurements in taxonomic problems. *Ann. Eugen.* **7**, 179–188 (1936)
15. Fukunaga, K.: Introduction to Statistical Pattern Recognition, 2nd edn. Academic Press, San Diego (1990)
16. Fukunaga, K., Mantock, J.: Nonparametric discriminant analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **5**, 671–678 (1983)
17. Graham, D.B., Allinson, N.M.: Characterizing virtual eigensignatures for general purpose face recognition. In: Wechsler, H., Phillips, P.J., Bruce, V., Fogelman-Soulie, F., Huang, T.S. (eds.) *Face Recognition: From Theory to Applications*. NATO ASI Series F, Computer and Systems Sciences, vol. 163, pp. 446–456 (1998)
18. Hamsici, O.C., Martinez, A.M.: Bayes optimality in linear discriminant analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(4), 647–657 (2008)
19. He, X., Cai, D., Yan, S., Zhang, H.-J.: Neighborhood preserving embedding. In: Proc. Int. Conf. Computer Vision (ICCV’05) (2005)
20. He, X., Niyogi, P.: Locality preserving projections. In: Thrun, S., Saul, L., Scholkopf, B. (eds.) *Advances in Neural Information Processing Systems*, vol. 16. MIT Press, Cambridge (2004)
21. He, X., Yan, S., Hu, Y., Niyogi, P.: Face recognition using Laplacianfaces. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(3), 328–340 (2005)
22. Huang, J., Smola, A.J., Gretton, A., Borgwardt, K.M., Schölkopf, B.: Correcting sample selection bias by unlabeled data. In: NIPS, pp. 601–608 (2006)
23. Jolliffe, I.: Principal Component Analysis, 2nd edn. Springer Series in Statistics, Springer, New York (2002)
24. Li, L.: Sparse sufficient dimension reduction. *Biometrika* **94**(3), 603–613 (2007)
25. Li, S.Z.: Face recognition based on nearest linear combinations. In: CVPR, pp. 839–844 (1998)
26. Li, S.Z., Lu, J.: Face recognition using the nearest feature line method. *IEEE Trans. Neural Netw.* **10**(2), 439–443 (1999)
27. Li, Z., Lin, D., Tang, X.: Nonparametric discriminant analysis for face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(4), 755–761 (2009)
28. Loog, M., Duin, R., Haeb-Umbach, R.: Multiclass linear dimension reduction by weighted pairwise Fisher criteria. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(7), 762–766 (2001)
29. Loog, M., Duin, R.P.W.: Linear dimensionality reduction via a heteroscedastic extension of lda: The Chernoff criterion. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**, 732–739 (2004)
30. Lotlikar, R., Kothari, R.: Fractional-step dimensionality reduction. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(6), 623–627 (2000)
31. Pan, S.J., Kwok, J.T., Yang, Q.: Transfer learning via dimensionality reduction. In: Proc. of the Twenty-Third AAAI Conference on Artificial Intelligence (2008)
32. Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.J.: The Feret evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 1090–1104 (2000)
33. Rao, C.R.: The utilization of multiple measurements in problems of biological classification. *J. R. Stat. Soc., Ser. B, Methodol.* **10**(2), 159–203 (1948)

34. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**, 2323–2326 (2000)
35. Saon, G., Padmanabhan, M.: Minimum Bayes error feature selection for continuous speech recognition. In: *Advances in Neural Information Processing Systems*, vol. 13, pp. 800–806. MIT Press, Cambridge (2001)
36. Schervish, M.: Linear discrimination for three known normal populations. *J. Stat. Plan. Inference* **10**, 167–175 (1984)
37. Shakhnarovich, G., Moghaddam, B.: Face recognition in subspaces. In: *Handbook of Face Recognition*, pp. 141–168 (2004)
38. Si, S., Tao, D., Geng, B.: Bregman divergence-based regularization for transfer subspace learning. *IEEE Trans. Knowl. Data Eng.* **22**(7), 929–942 (2010)
39. Tao, D., Li, X., Wu, X., Maybank, S.J.: Geometric mean for subspace selection. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(2), 260–274 (2009)
40. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**(5500), 2319–2323 (2000)
41. Tibshirani, R.: Regression shrinkage and selection via the lasso. *J. R. Stat. Soc., Ser. B, Stat. Methodol.* **58**, 267–288 (1996)
42. Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. *J. R. Stat. Soc., Ser. B, Stat. Methodol.* **61**(3), 611–622 (1999)
43. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**, 71–86 (1991)
44. Wang, X., Tang, X.: A unified framework for subspace face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**, 1222–1228 (2004)
45. Wang, X., Tang, X.: Subspace analysis using random mixture models. In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 574–580 (2005)
46. Wang, X., Tang, X.: Random sampling for subspace face recognition. *Int. J. Comput. Vis.* **70**, 91–104 (2006)
47. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**, 210–227 (2009)
48. Yan, S., Xu, D., Zhang, B., Zhang, H.-J., Yang, Q., Lin, S.: Graph embedding and extensions: A general framework for dimensionality reduction. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(1), 40–51 (2007)
49. Ye, J.: Least squares linear discriminant analysis. In: *Proceedings of the 24th International Conference on Machine Learning, ICML '07*, pp. 1087–1093 (2007)
50. Ye, J., Ji, S.: Discriminant analysis for dimensionality reduction: An overview of recent developments. In: Boulgouris, N., Plataniotis, K.N., Micheli-Tzanakou, E. (eds.) *Biometrics: Theory, Methods, and Applications*. Wiley-IEEE Press, New York (2010). Chap. 1
51. Ye, J., Li, Q.: A two-stage linear discriminant analysis via qr-decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(6), 929–941 (2005)
52. Ye, J., Li, Q.: A two-stage linear discriminant analysis via qr-decomposition. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**, 929–941 (2005)
53. Zhang, Z., Zha, H.: Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM J. Sci. Comput.* **26**, 313–338 (2005)
54. Zhang, T., Tao, D., Yang, J.: Discriminative locality alignment. In: *Proceedings of the 10th European Conference on Computer Vision*, pp. 725–738. Berlin, Heidelberg, 2008
55. Zhang, T., Tao, D., Li, X., Yang, J.: Patch alignment for dimensionality reduction. *IEEE Trans. Knowl. Data Eng.* **21**, 1299–1313 (2009)
56. Zhou, T., Tao, D., Wu, X.: Manifold elastic net: A unified framework for sparse dimension reduction. *Data Min. Knowl. Discov.* (2010)
57. Zhu, M., Martinez, A.M.: Subclass discriminant analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**, 1274–1286 (2006)
58. Zou, H., Hastie, T.: Regularization and variable selection via the Elastic Net. *J. R. Stat. Soc. B* **67**, 301–320 (2005)
59. Zou, H., Hastie, T., Tibshirani, R.: Sparse principal component analysis. *J. Comput. Graph. Stat.* **15** (2004)



# Chapter 4

## Local Representation of Facial Features

Joni-Kristian Kämäriinen, Abdenour Hadid, and Matti Pietikäinen

The aim of this chapter is to give a comprehensive overview of different facial representations and in particular describe local facial features.

### 4.1 Introduction

Developing face recognition systems involves two crucial issues: facial representation and classifier design [47, 101]. The aim of facial representation is to derive a set of features from the raw face images which minimizes the intra-class variations (i.e., within face instances of a same individual) and maximizes the extra-class variations (i.e., between face images of different individuals). Obviously, if inadequate facial representations are adopted, even the most sophisticated classifiers fail to accomplish the face recognition task. Therefore, it is important to carefully decide on what facial representation to adopt when designing face recognition systems. Ideally, the facial feature representation should: (i) discriminate different individuals well while tolerating within-class variations; (ii) be easily extracted from the raw face images in order to allow fast processing; and (iii) lie in a low dimensional space (short vector length) in order to avoid a computationally expensive classifier. Naturally, it is

---

J.-K. Kämäriinen (✉)

Machine Vision and Pattern Recognition Laboratory, Lappeenranta University of Technology,  
Lappeenranta, Finland

e-mail: [Joni.Kamarainen@lut.fi](mailto:Joni.Kamarainen@lut.fi)

A. Hadid · M. Pietikäinen

Machine Vision Group, Dept. of Electrical and Information Engineering, University of Oulu,  
Oulu, Finland

A. Hadid

e-mail: [hadid@ee.oulu.fi](mailto:hadid@ee.oulu.fi)

M. Pietikäinen

e-mail: [mkp@ee.oulu.fi](mailto:mkp@ee.oulu.fi)

not easy to find features which meet all these criteria because of the large variability in facial appearances due to different imaging factors such as scale, orientation, pose, facial expressions, lighting conditions, aging, presence of glasses, etc. These considerations are important for the other subtasks in face biometrics: detection, localization and registration, and verification, and thus, a key issue in face recognition is finding efficient facial feature representations.

Numerous methods have been proposed in literature for representing facial images for recognition purposes. The earliest attempts, such as Kanade's work in early 70s [41], are based on representing faces in terms of geometrical relationships, such as distances and angles, between the facial landmarks (eyes, mouth etc.). Later, appearance based techniques have been proposed. These methods generally consider a face as a 2D array of pixels and aim at deriving descriptors for face appearance without explicit use of face geometry. Following these lines, different holistic methods such as Principal Component Analysis (PCA) [82], Linear Discriminant Analysis (LDA) [21] and the more recent 2D PCA [92] have been widely studied. Lately local descriptors have gained an increasing attention due to their robustness to challenges such as pose and illumination changes. Among these descriptors are Gabor filters and Local Binary Patterns [2] which are shown to be very successful in encoding facial appearance.

### ***4.1.1 Structure and Scope of the Chapter***

The aim of this chapter is to give a comprehensive overview of different facial representations and in particular describe local facial features. Section 4.2 discusses the major methods which have been proposed in literature. Then, more detailed descriptions of two widely used approaches, namely local binary patterns and Gabor filters, are presented in Sects. 4.3 and 4.4, respectively. Section 4.5 discusses related issues and promising directions. Finally, concluding remarks are drawn in Sect. 4.6.

The methods discussed in this chapter can be applied to detection and recognition of faces or face parts (landmarks). Face parts are also referred to as facial features, but we use the terms feature and facial feature interchangeably for any features extracted from the face area. We specifically discuss local binary patterns in the context of face recognition and Gabor features in the context of face part detection, but they can be used in the both tasks. Furthermore, the feature extraction methods are discussed from the face image processing point of view and other face description methods are available for the modeling purposes, such as the active shape models and morphable model described in the following chapters. These novel modeling methods can also be applied to face recognition without explicit feature extraction and classification as discussed in this chapter.

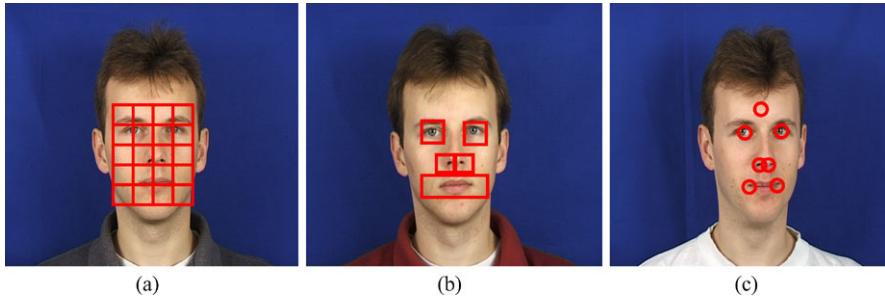
## 4.2 Review of Facial Feature Representations

We first justify and restrict the scope of this chapter to generic features which do not require optimization or learning stages and then proceed to the actual review.

Zhao et al. [101] divide face recognition algorithms into (i) *appearance-based* (holistic), (ii) *feature-based*, and (iii) *hybrid* approaches. This taxonomy is widely accepted and also applies to face detection, localization and verification algorithms [33]. This chapter specifically focuses on the feature-based and hybrid methods which utilize representations of local face parts. Zhao et al. further divide the feature-based and hybrid approaches into: (1) *generic methods based on generic image processing features*, such as edges, lines, curves, etc.; (2) *feature-template-based methods* that are used to detect specific facial features such as eyes, nostrils, etc.; and (3) *structural matching methods* that take into consideration geometrical constraints on the features. From the feature extraction point of view, the holistic approach and the feature-template-based methods are equivalent. They both learn a scanning window template or templates to represent and detect faces or facial parts. The most popular solutions are Viola–Jones detector [85] and PCA or LDA computed subspace-templates (Eigenfaces or Fisherfaces) [9] and their seminal works. These methods can be effective, but we do not include the Haar-cascades produced by the Viola–Jones method or subspace templates produced by the PCA and LDA to this chapter since they are not generic features. They should be considered as learned statistical or algorithmic detectors themselves. Subspace methods are discussed in Chap. 3 and Viola–Jones type boosted detectors in Chap. 11. The Haar-like features used by the Viola–Jones detector, however, are generic features for facial feature representation. The structural matching methods are not in the scope either since they too involve the learning stage for a “constellation model” which captures information about spatial relationships between local features. Typical examples are active shape models, discussed in Chap. 4, and the Elastic Bunch Graph Matching (EBGM) [89]. The generic low level features used by these methods, however, belong to this chapter.

The selection of features for a proper facial feature representation is actually similar to the feature selection and extraction task occurring in the most computer vision and image analysis applications. But what features are the most suitable for face biometrics? The best results have been achieved by concatenating and learning person specific features computed from several local areas, for example, from fixed area (Fig. 4.1(a)) or varying area regions (Fig. 4.1(b)) which can be regular or feature-driven, or simply at specific locations with no strictly defined spatial extent (Fig. 4.1(c)). As already mentioned, implementations based on the subspace approach [11] and the boosted Haar-like features [103] for face detection and recognition exist, but they are not included here due to their need of task-specific learning.

Computer vision and image processing literature contains numerous features and feature extraction methods. In face biometrics, however, certain features retain their popularity and continuously succeed to producing state-of-the-art results for various benchmarks. Widely adopted are features constructed from responses of Gabor filters on various orientations and scales. More recent, and particularly successful,



**Fig. 4.1** Facial feature computation from **a** a regular grid of fixed size regions, **b** irregular variable size regions (feature-driven) and **c** around central feature locations

are local binary pattern (LBP) features. In order to verify their status and to spot new trends, we reviewed the recently published feature-intense articles in the top tier forums of computer vision and face biometrics. A short summary of the review is presented in Table 4.1. We draw the following conclusions: (1) Gabor filters and other similar “local oriented frequency approaches” are still a popular choice and produce state-of-the-art results in face detection and recognition; (2) a new feature appears in the literature: the SIFT descriptor which is popular in visual object categorization and baseline matching; (3) gray-level patch remains as a popular choice as well despite of its extreme simplicity; and finally (4) success of LBP in biometrics promotes other similar algorithmically constructed features. An interesting work is the method by Xu et al. [90], which uses several different kind of features on different processing levels in their hierarchical system.

The most popular region features, modular PCA, LBP and Gabor magnitudes, were compared for face recognition in [103]. The LBP and Gabor features produced good results and were generally recommended. In Table 4.1, we classify many features, such as complex and smooth wavelets, steerable filters and difference of Gaussians, to Gabor-based methods, because there is no fundamental difference between them and properly utilized they should lead to equally good results. Similarly, SIFT, LBP and Daugman’s phase descriptor have similar characteristics. The flexibility of LBP features, however, makes them more suitable and preferable for face biometrics. The flexibility, appearing as various intuitive parameterizations and extensions to the standard LBP are further discussed in Sect. 4.3. The Haar-like features seem to succeed for the boosting approaches, but as a generic method for face biometrics there is no clear evidence for their success. Their accuracy to locate different facial landmarks have been studied in [11] and recently, other kind of features, such as anisotropic Gaussian [60] or constructed features [87], have succeeded in the boosting scheme.

It is clear from all previously published surveys and from the recent state-of-the-art results that the three mentioned features pop up as very popular and successful: features based on Gabor filter responses, local binary patterns (LBPs) and Haar-like features. Since the Haar-like features are covered in Chap. 11, this chapter

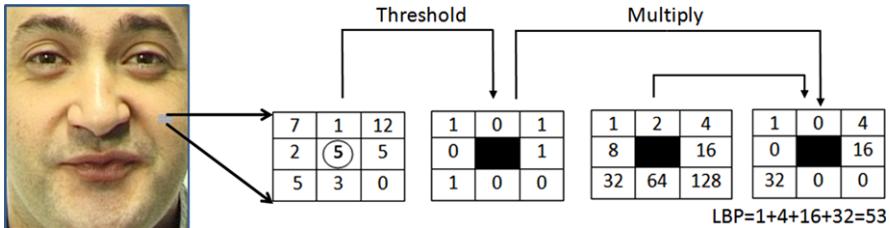
**Table 4.1** Feature-based methods for face detection and/or recognition. Papers utilizing LBP are numerous and therefore not included here but in Sect. 4.3

#	Ref.	Feature(s)	Comment
1	Zhang et al. [98]	“Local derivative pattern”	Similar to LBP
2	Kozakaya et al. [42]	Histogram of gradients (HOG)	Similar to SIFT
3	Zhang and Wang [94]	SIFT	
4	Su et al. [77]	Gabor	Reg. grid, magn. only
5	Pinto et al. [68]	Gabor, Patch	Magn. only, post-processing
6	Hua and Akbarzadeh [34]	Gradient descriptor in [88]	
7	Lee et al. [46]	Modular PCA	
8	Liu and Dai [53]	Wavelet	Similar to Gabor
9	McCool and Marcel [56]	DCT coeffs.	Similar to Gabor magn. histogram
10	Ashraf et al. [7]	Patch	
11	Ding and Martinez [19]	Patch and geometric	
12	Liang et al. [50]	Patch	
13	Meyers and Wolf [59]	Gabor	V1 type post-processing
14	Mian et al. [61]	3D descriptor and SIFT	
15	Xu et al. [90]	Patch, gradient (AAM) and geometric	Fusion over layers of processing
16	Yan et al. [91]	Haar based pattern (LAB)	Similar to LBP
17	Gökberk et al. [27]	Gabor	Magn. only, centroids
18	Shastri and Levine [75]	Non-negative sparse codebook	Similar to Gabor magn.
19	Zhang et al. [97]	Gabor	Daugman’s phase code [18] (similar to SIFT)
20	Arca et al. [6]	Gabor	Magn. only, centroids
21	Bicego et al. [10]	SIFT	
22	Ekenel and Stiefelhagen [20]	DCT coeffs.	Similar to Gabor magn. histogram
23	Zhang and Jia [93]	Steerable filters	Similar to Gabor
24	Dalal and Triggs [14]	Histogram of gradients (HOG)	Similar to SIFT

introduces the remaining two and presents results from face recognition and facial feature localization experiments.

### 4.3 Local Binary Patterns

The use of local binary patterns in face analysis started in 2004 when a novel facial representation for face recognition was proposed [1, 2]. In this approach, the face



**Fig. 4.2** The basic LBP operator

image is divided into several regions from which the LBP features are extracted and concatenated into an enhanced feature histogram which is used as a face descriptor. The approach has evolved to be a growing success and has been adopted and further developed by a large number of research groups and companies around the world. The LBP operator and its variants have been used not only in face recognition but also in various other face-related problems such as face detection, facial expression recognition, gender classification, age estimation and visual speech recognition. The success of LBP in face description is due to the discriminative power and computational simplicity of the operator, and its robustness to monotonic gray scale changes caused by, for example, illumination variations. The use of histograms as features also makes the LBP approach robust to face misalignment and pose variations. The Matlab code of the LBP operators can be found and freely downloaded from <http://www.ee.oulu.fi/mvg/page/downloads>.

### 4.3.1 Local Binary Patterns

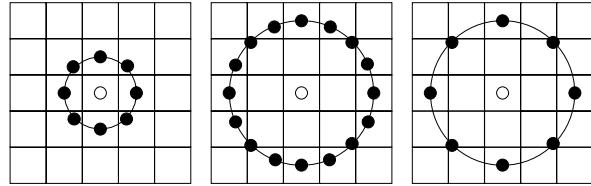
#### 4.3.1.1 LBP in the Spatial Domain

The LBP texture analysis operator, introduced by Ojala et al. [63, 64], is defined as a gray-scale invariant texture measure, derived from a general definition of texture in a local neighborhood. It is a powerful texture descriptor and among its properties in real-world applications are its discriminative power, computational simplicity and tolerance against monotonic gray-scale changes.

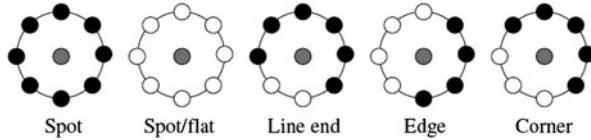
The original LBP operator forms labels for the image pixels by thresholding the  $3 \times 3$  neighborhood with the center value and considering the result as a binary number. The histogram of these  $2^8 = 256$  different labels can then be used as an image descriptor. See Fig. 4.2 for an illustration of the basic LBP operator. The operator has been extended to use neighborhoods of different sizes [64]. Using a circular neighborhood and bilinear interpolation at noninteger pixel coordinates allow any radius and number of sampling points. In the following, the notation  $(P, R)$  will be used for pixel neighborhoods which means  $P$  sampling points on a circle of radius  $R$ . See Fig. 4.3 for an example of circular neighborhoods.

Another extension to the original operator is the definition of so called *uniform patterns* [64]. This extension was inspired by the fact that some binary patterns

**Fig. 4.3** Neighborhood set for different  $(P, R)$ . The pixel values are bilinearly interpolated whenever the sampling point is not in the center of a pixel



**Fig. 4.4** Examples of texture primitives detected by LBP (white circles represent ones and black zeros)



occur more frequently than others in texture images. A local binary pattern is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is traversed circularly. For example, the patterns 00000000 (0 transitions), 01110000 (2 transitions) and 11001111 (2 transitions) are uniform whereas the patterns 11001001 (4 transitions) and 01010011 (6 transitions) are not. In the computation of the LBP labels, uniform patterns are used so that there is a separate label for each uniform pattern and all the non-uniform patterns are labeled with a single label. For example, when using  $(8, R)$  neighborhood, there are a total of 256 patterns of which 58 are uniform thus yielding to the total of 59 different labels.

Ojala et al. noticed in their experiments with texture images that uniform patterns account for almost 90% of all patterns when using the  $(8, 1)$  neighborhood and around 70% for the  $(16, 2)$  neighborhood. We have found that 90.6% of the patterns in the  $(8, 1)$  neighborhood and 85.2% of the patterns in the  $(8, 2)$  neighborhood are uniform in the case of preprocessed FERET face images [67]. Each LBP code can be regarded as a micro-texton. Local primitives which are codified by these bins include different types of curved edges, spots, flat areas etc. as illustrated in Fig. 4.4.

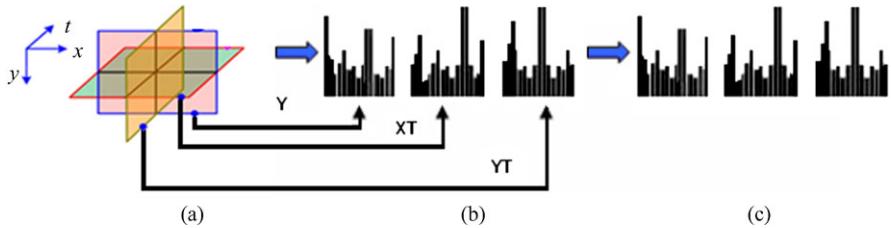
We use the following notation for the LBP operator:  $\text{LBP}_{P,R}^{u2}$ . The subscript denotes the operator in a  $(P, R)$  neighborhood. Superscript  $u2$  stands for uniform patterns of maximum of 2 transitions and labeling all remaining patterns with a single label.

After the LBP labeled image  $f_l(x, y)$  has been obtained, the LBP histogram can be defined as

$$H_i = \sum_{x,y} I\{f_l(x, y) = i\}, \quad i = 0, \dots, n - 1, \quad (4.1)$$

in which  $n$  is the number of different labels produced by the LBP operator and

$$I\{A\} = \begin{cases} 1, & \text{if } A \text{ is true,} \\ 0, & \text{if } A \text{ is false.} \end{cases}$$



**Fig. 4.5** **a** Three planes of dynamic texture; **b** LBP histograms of each plane; **c** Concatenated feature

When the image patches whose histograms are to be compared have different sizes, the histograms must be normalized to get a coherent description:

$$N_i = \frac{H_i}{\sum_{j=0}^{n-1} H_j}. \quad (4.2)$$

#### 4.3.1.2 Spatiotemporal LBP

The original LBP operator was defined to only deal with the spatial information, but recently it has been extended to a spatiotemporal representation for dynamic texture (DT) analysis. This has yielded to so called Volume Local Binary Pattern operator (VLBP) [99]. The idea behind VLBP consists of looking at dynamic texture as a set of volumes in the \$(X,Y,T)\$-space where \$X\$ and \$Y\$ denote the spatial coordinates and \$T\$ the frame index (time). The neighborhood of each pixel is thus defined in a three dimensional space. Then, similarly to LBP, volume textons can be defined and extracted into histograms. Therefore, VLBP combines motion and appearance into a dynamic texture description.

To make the VLBP computationally simple and easy to extend, the cooccurrences of the LBP on the three orthogonal planes (LBP-TOP) was introduced [99]. LBP-TOP consists of the three orthogonal planes: \$XY\$, \$XT\$ and \$YT\$, and concatenating local binary pattern co-occurrence statistics in these three directions. The circular neighborhoods are generalized to elliptical sampling to fit to the space-time statistics. The LBP codes are extracted from the \$XY\$, \$XT\$ and \$YT\$ planes, denoted as \$XY\$-LBP, \$XT\$-LBP and \$YT\$-LBP, for all pixels, and statistics of the three different planes are concatenated into a single histogram. The procedure is shown in Fig. 4.5. In this representation, dynamic texture (DT) is encoded by \$XY\$-LBP, \$XT\$-LBP and \$YT\$-LBP.

Using equal radii for the time and spatial axes is not reasonable for dynamic textures [99] and therefore, in the \$XT\$ and \$YT\$ planes, different radii can be assigned to sample neighboring points in space and time. More generally, the radii in axes \$X\$, \$Y\$ and \$T\$, and the number of neighboring points in the \$XY\$, \$XT\$ and \$YT\$ planes can also be different denoted by \$R\_X\$, \$R\_Y\$ and \$R\_T\$, \$P\_{XY}\$, \$P\_{XT}\$ and \$P\_{YT}\$. The corresponding feature is denoted as \$\text{LBP-TOP}\_{P\_{XY}, P\_{XT}, P\_{YT}, R\_X, R\_Y, R\_T}\$.

Let us assume we are given an  $X \times Y \times T$  dynamic texture ( $x_c \in \{0, \dots, X - 1\}$ ,  $y_c \in \{0, \dots, Y - 1\}$ ,  $t_c \in \{0, \dots, T - 1\}$ ). A histogram of the DT can be defined as

$$H_{i,j} = \sum_{x,y,t} I\{f_j(x, y, t) = i\}, \quad i = 0, \dots, n_j - 1; \quad j = 0, 1, 2, \quad (4.3)$$

in which  $n_j$  is the number of different labels produced by the LBP operator in the  $j$ th plane ( $j = 0 : XY$ ,  $1 : XT$  and  $2 : YT$ ) and  $f_i(x, y, t)$  expresses the LBP code of central pixel  $(x, y, t)$  in the  $j$ th plane. Similarly to the original LBP, the histograms must be normalized to get a coherent description for comparing the DTs:

$$N_{i,j} = \frac{H_{i,j}}{\sum_{k=0}^{n_j-1} H_{k,j}}. \quad (4.4)$$

#### 4.3.1.3 Multi-Scale LBP

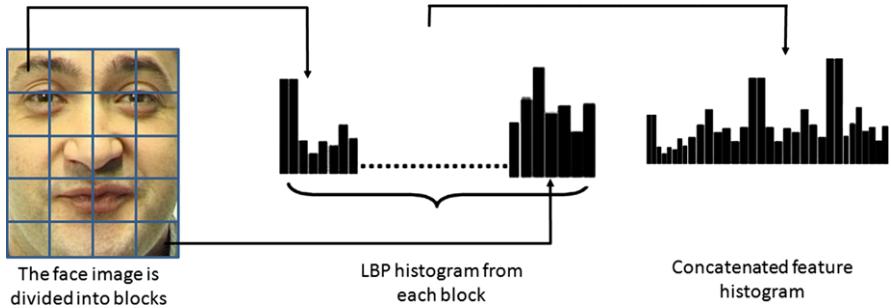
Noticing that LBP features calculated in a local  $3 \times 3$  neighborhood cannot capture large-scale structures, multi-scale LBP has been proposed to overcome this limitation. A straightforward way of enlarging the spatial support area is to combine the information provided by  $N$  LBP operators with varying  $P$  and  $R$  values. This way, each pixel in an image gets  $N$  different LBP codes. The most accurate information would be obtained by using the joint distribution of these codes. However, such a distribution would be overwhelmingly sparse with any reasonable image size. Therefore, only the marginal distributions of the different operators are considered. Even though the LBP codes at different radii are not statistically independent in the typical case, using multi-resolution analysis often enhances the discriminative power of the resulting features. With most applications, this straightforward way of building a multi-scale LBP operator has resulted in very good accuracy.

An extension of multi-scale LBP operator is the multiscale block local binary pattern (MB-LBP) [51] which has gained popularity especially in facial image analysis. The key idea of MB-LBP is to compare average pixel values within small blocks instead of comparing pixel values. The operator always considers 8 neighbors, producing labels from 0 to 255. For instance, if the block size is  $3 \times 3$  pixels, the corresponding MB-LBP operator compares the average gray value of the center block to the average values of the 8 neighboring blocks of the same size and the effective area of the operator is  $9 \times 9$  pixels.

#### 4.3.2 Face Description Using LBP

##### 4.3.2.1 Description of Static Face Images

In the LBP approach for texture classification [64], the occurrences of the LBP codes in an image are collected into a histogram. The classification is then performed by



**Fig. 4.6** Example of an LBP based facial representation

computing simple histogram similarities. However, considering a similar approach for facial image representation results in a loss of spatial information and therefore one should codify the texture information with their locations. One way to achieve this goal is to use the LBP texture descriptors to build several local descriptions of the face and combine them into a global description. Such local descriptions have gained interest lately which is understandable given the limitations of the holistic representations. These local feature based methods seem to be more robust against variations in pose or illumination than holistic methods.

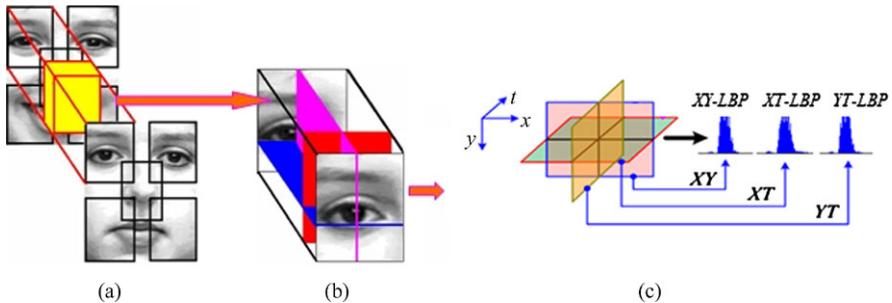
The basic methodology for LBP based face description is as follows: The facial image is divided into local regions and LBP texture descriptors are extracted from the each region independently. The descriptors are then concatenated to a global face description, as shown in Fig. 4.6.

The basic histogram that is used to gather information about LBP codes in an image can be extended into a *spatially enhanced histogram* which encodes both the appearance and the spatial relations of facial regions. As the facial regions  $R_0, R_1, \dots, R_{m-1}$  have been determined, the spatially enhanced histogram is defined as

$$H_{i,j} = \sum_{x,y} I\{f_l(x, y) = i\} I\{(x, y) \in R_j\}, \quad i = 0, \dots, n-1, \quad j = 0, \dots, m-1.$$

This histogram effectively has a description of the face on three different levels of locality: the LBP labels for the histogram contain information about the patterns on a pixel-level, the labels are summed over a small region to produce information on a regional level and the regional histograms are concatenated to build a global description of the face. It should be noted that when using the histogram based methods the regions  $R_0, R_1, \dots, R_{m-1}$  do not need to be rectangular. Neither do they need to be of the same size or shape, and they do not necessarily have to cover the whole image. It is also possible to have partially overlapping regions.

This outlines the original LBP based facial representation [1, 2] that has been later adopted to various facial image analysis tasks [31, 45]. Figure 4.6 shows an example of an LBP based facial representation.



**Fig. 4.7** Features in each block volume. **a** Block volumes; **b** LBP features from three orthogonal planes; **c** Concatenated features for one block volume with the appearance and motion

#### 4.3.2.2 Description of Face Sequences

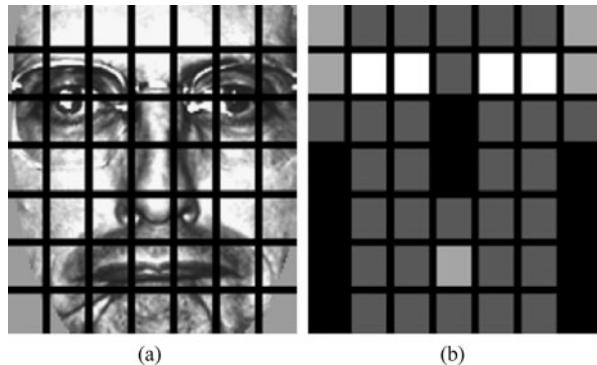
How can moving faces be efficiently represented? Psychophysical findings state that facial movements can provide valuable information to face analysis. Therefore, efficient facial representations should encode both appearance and motion. We thus describe an LBP based spatiotemporal representation for face analysis in videos using region-concatenated descriptors. Like in [2], an LBP description computed over a whole face sequence encodes only the occurrences of the micro-patterns without any indication about their locations. To overcome this effect, a representation in which the face image is divided into several overlapping blocks is used. The LBP-TOP histograms in each block are computed and concatenated into a single histogram, as illustrated in Fig. 4.7. All features extracted from the each volume are connected to represent the appearance and motion of the face in the sequence. The basic VLBP features could also be considered and extracted on the basis of region motion in the same way as the LBP-TOP features.

The LBP-TOP histograms in each block volume are computed and concatenated into a single histogram. All features extracted from each block volume are connected to represent the appearance and motion of the face. In this way, we effectively have a description of the face on three different levels of locality. The labels (bins) in the histogram contain information from three orthogonal planes, describing appearance and temporal information at the pixel level. The labels are summed over a small block to produce information on a regional level expressing the characteristics for the appearance and motion in specific locations, and all information from the regional level is concatenated to build a global description of the face sequence.

#### 4.3.3 Face Recognition Using LBP Descriptors

This section describes the application of the LBP based face description to face recognition. Typically a nearest neighbor classification rule is used in the face recognition task. This is due to the fact that the number of training (gallery) images per

**Fig. 4.8** **a** An example of a facial image divided into  $7 \times 7$  windows. **b** The weights set for weighted  $\chi^2$  dissimilarity measure. The *black squares* indicate weight 0.0, *dark gray* 1.0, *light gray* 2.0 and *white* 4.0



subject is low, often only one. However, the idea of a spatially enhanced histogram can be exploited further when defining the distance measure for the classifier. An indigenous property of the proposed face description method is that each element in the enhanced histogram corresponds to a certain small area of the face. Based on the psychophysical findings, which indicate that some facial features (such as eyes) play a more important role in human face recognition than other features [101], it can be expected that some of the facial regions contribute more than others in terms of extra-personal variance. Utilizing this assumption the regions can be weighted based on the importance of the information they contain. Figure 4.8 shows an example of weighting different facial regions. The weighted Chi square distance can be defined as

$$\chi_w^2(\mathbf{x}, \xi) = \sum_{j,i} w_j \frac{(x_{i,j} - \xi_{i,j})^2}{x_{i,j} + \xi_{i,j}}, \quad (4.5)$$

in which  $\mathbf{x}$  and  $\xi$  are the normalized enhanced histograms to be compared, indices  $i$  and  $j$  refer to  $i$ th bin corresponding to the  $j$ th local region and  $w_j$  is the weight for the region  $j$ .

In [1, 2, 4], Ahonen et al. performed a set of experiments on the FERET face images [67]. The results showed that the LBP approach yields higher face recognition rates than the control algorithms (PCA [82], Bayesian Intra/Extra-personal Classifier (BIC) [62] and Elastic Bunch Graph Matching EBGM [89]). To gain better understanding on whether the obtained recognition results are due to general idea of computing texture features from local facial regions or due to the discriminatory power of the local binary pattern operator, we also compared LBP to three other texture descriptors, namely the gray-level difference histogram, homogeneous texture descriptor [55] and an improved version of the texton histogram [83]. The details of these experiments can be found in [4]. The results confirmed the validity of the LBP approach and showed that the performance of LBP in face description exceeds that of other texture operators as shown in Table 4.2. We believe that the main explanation for the better performance over other texture descriptors is the tolerance to monotonic gray-scale changes. Additional advantages are the computational efficiency and avoidance of gray-scale normalization prior to the LBP operator.

**Table 4.2** The recognition rates obtained using different texture descriptors for local facial regions. The first four columns show the recognition rates for the FERET test sets and the last three columns contain the mean recognition rate of the permutation test with a 95% confidence interval

Method	fb	fc	dup I	dup II	lower	mean	upper
Difference histogram	0.87	0.12	0.39	0.25	0.58	0.63	0.68
Homogeneous texture	0.86	0.04	0.37	0.21	0.58	0.62	0.68
Texton Histogram	0.97	0.28	0.59	0.42	0.71	0.76	0.80
LBP (nonweighted)	0.93	0.51	0.61	0.50	0.71	0.76	0.81



**Fig. 4.9** Example of Gallery and probe images from the FRGC database, and their corresponding filtered images with Tan and Triggs' preprocessing chain [80]

Recently, Tan and Triggs developed a very effective preprocessing chain for face images and obtained excellent results using LBP-based face recognition for the FRGC database [80]. Since then, many others have adopted their preprocessing chain for applications dealing with severe illumination variations. Figure 4.9 shows an example of gallery and probe images from the FRGC database and the corresponding filtered images with the preprocessing method.

Chan et al. [12] considered multi-scale LBPs and derived new face descriptor from Linear Discriminant Analysis (LDA) of multi-scale local binary pattern histograms. The face image is first partitioned into several non-overlapping regions. In each region, multi-scale uniform LBP histograms are extracted and concatenated into a regional feature. The features are then projected on the LDA space to be used as a discriminative facial descriptor. The method was tested in face identification on the standard FERET database and in face verification on the XM2VTS database with very promising results.

Zhang et al. [95] considered the LBP methodology for face recognition and used AdaBoost learning algorithm for selecting an optimal set of local regions and their weights. This yielded to a smaller feature vector than that used in the original LBP approach [1]. However, no significant performance enhancement was obtained. Later, Huang et al. [36] proposed a variant of AdaBoost called JSBoost for selecting the optimal set of LBP features for face recognition.

In order to deal with strong illumination variations, Li et al. developed a very successful system combining near infrared (NIR) imaging with local binary pattern features and AdaBoost learning [49]. The invariance of LBP with respect to mono-

tonic gray level changes makes the features extracted from NIR images illumination invariant.

In [70], Rodriguez and Marcel proposed an approach based on adapted, client-specific LBP histograms for the face verification task. The method considers local histograms as probability distributions and computes a log-likelihood ratio instead of  $\chi^2$  similarity. A generic face model is considered as a collection of LBP histograms. Then, a client-specific model is obtained by an adaptation technique from the generic model under a probabilistic framework. The reported experimental results show that the proposed method yields good performance on two benchmark databases (XM2VTS and BANCA). Later, Ahonen and Pietikäinen [3] have further enhanced the face verification performance on the BANCA database by developing a novel method for estimating the local distributions of LBP labels. The method is based on kernel density estimation in  $xy$ -space, and it provides much better spatial accuracy than the block-based method of Rodriguez and Marcel [70].

#### **4.3.4 LBP in Other Face-Related Problems**

The LBP approach has also been adopted to several other face analysis tasks such as facial expression recognition [23, 74], gender recognition [78], age classification [86], face detection [30, 71, 91], iris recognition [79], head pose estimation [54] and 3D face recognition [48]. For instance, LBP is used in [35] with Active Shape Model (ASM) for localizing and representing facial key points since an accurate localization of such points of the face is crucial to many face analysis and synthesis problems. The local appearance of the key points in the facial images are modeled with an Extended version of Local Binary Patterns (ELBP). ELBP was proposed in order to encode not only the first derivation information of facial images but also the velocity of local variations. The experimental analysis showed that the combination ASM-ELBP enhances the face alignment accuracy compared to the original ASM method.

In [30], the authors devised another LBP based representation which is suitable for low-resolution images and has a short feature vector needed for fast processing. A specific aspect of this representation is the use of overlapping regions and a 4-neighborhood LBP operator ( $LBP_{4,1}$ ) to avoid statistical unreliability due to long histograms computed over small regions. Additionally, the holistic description of a face was enhanced by including the global LBP histogram computed over the whole face image. The proposed representation performed well in the face detection problem.

Spatiotemporal LBP descriptors, especially LBP-TOP, have been successfully utilized in many video-based applications, for example, dynamic facial expression recognition [100], visual speech recognition [102] and gender recognition from videos [29]. They can effectively describe appearance, horizontal motion and vertical motion from the video sequence. LBP-TOP based approach was also extended to include multiresolution features which are computed from different sized blocks,



**Fig. 4.10** Selected 15 slices for different facial expression pairs

different neighboring samplings and different sampling scales, and utilize AdaBoost to select the slice features for all the expression classes or every class pair, to improve the performance with short feature vectors. After that, on the basis of selected slices, the location and feature types of most discriminative features for every class pair are considered. Figure 4.10 shows the selected features for two expression pairs. They are different and specific depending on the expressions.

## 4.4 Gabor Features

### 4.4.1 Introduction

Methods using Gabor features have been particularly successful in biometrics. For example, Daugman's iris code [18] is The Method for iris recognition, Gabor features were used in the two best methods in the ICPR 2004 face recognition contest [57] and they are among the top performers in fingerprint matching [38], and so on. It is interesting, why feature extraction based on the Gabor's principle of simultaneous localization in the frequency and spatial domains [25], is so successful in many applications of computer vision and image processing. The same principle was independently found as an intuitive requirement for a “general picture processing operator” by Granlund [28], and later rigorously defined in 2D by Daugman [16].

As the well-known result in face recognition, Lades et al. developed a Gabor based system using dynamic link architecture (DLA) framework which recognizes faces by extracting a set of features (“Gabor jet”) at each node of a rectangular grid over the face image [44]. Later, Wiskott et al. extended the approach and developed the well-known Gabor wavelet-based elastic bunch graph matching (EBGM) method to label and recognize faces [89]. In the EBGM algorithm, faces are represented as graphs with nodes positioned at fiducial points (such as the eyes and the tip of the nose) and edges labeled with distance vectors. Each node contains a set of Gabor wavelet coefficients, known as a jet. Thus, the geometry of the face is encoded by the edges while the local appearance is encoded by the jets. The identification of a face consists of determining among the constructed graphs the one which maximizes the graph similarity function.

In this section, we first explain the main properties of Gabor filters, then describe how image features can be constructed from filter responses, and finally, demonstrate how these features can accurately and efficiently represent and detect facial features. Note that, similarly to LBP, Gabor filters can be used to either detect face parts or whole face for recognition. In the previous sections, we explained the use of LBP for face appearance description. For completeness, we focus below on the use of Gabor filters for representing and detecting facial landmarks.

#### 4.4.2 Gabor Filter

Gabor filter is Gabor function changed into the linear filter form, that is, a signal or an image can be convolved with the filter to produce a “response image”. This process is similar to edge detection. Gabor features are formed by combining responses of several filters from a single or multiple spatial locations. Gabor function provides the minimal joint-uncertainty  $\Delta t \times \Delta f$  simultaneously in the time (spatial) and frequency domains. In 1946, Dennis Gabor proved that: “*The signal which occupies the minimum area  $\Delta t \Delta f = \frac{1}{2}$  is the modulation product of a harmonic oscillation<sup>(\*)</sup> of any frequency with pulse of the form of a probability function<sup>(\*\*)</sup>*” [25]

$$\psi(t) = \underbrace{e^{-\alpha^2(t-t_0)^2}}_{(**)} \underbrace{e^{j2\pi f_0 t + \phi}}_{(*)}. \quad (4.6)$$

In (4.6),  $\alpha$  is the sharpness (time duration and bandwidth) of the Gaussian,  $t_0$  is the time shift defining the time location of the Gaussian,  $f_0$  is the frequency of the harmonic oscillations (frequency location), and  $\phi$  denotes the phase shift of the oscillation. The Gabor elementary function in (4.6) has a Fourier spectrum of analytical form

$$\Psi(f) = \sqrt{\frac{\pi}{\alpha^2}} e^{-(\frac{\pi}{\alpha})^2(f-f_0)^2} e^{-j2\pi t_0(f-f_0) + \phi}. \quad (4.7)$$

Two important findings can be seen in (4.6) and (4.7): Gabor function, or more precisely its magnitude, has the Gaussian form in the time domain and frequency domain; The Gaussian is located at  $t_0$  in time and  $f_0$  in frequency; If you increase the bandwidth  $\alpha$ , the function will shrink in time (more accurate), but stretch in frequency (more inaccurate). These are the properties which help understand Gabor filter as a linear operator operating in time and frequency simultaneously. For the linear filter form, the function is typically simplified by centering it to origin ( $t_0 = 0$ ) and removing the phase shift ( $\phi = 0$ ).

Gabor's original idea was to synthesize signals using a set of these elementary functions. That research direction has lead to the theory of Gabor expansion (Gabor transform) [8] and more generally to the Gabor frame theory [22]. Feature extraction, however, is signal analysis. The development of the 2D Gabor elementary

functions began from Granlund in 1978, when he defined some fundamental properties and proposed the form of a general picture processing operator. The general picture processing operator had a form of the Gabor elementary function in two dimensions and it was derived directly from the needs of the image processing without a connection to Gabor's work [28]. It is noteworthy that Granlund addressed many properties, such as the octave spacing of the frequencies, that were reinvented later for the Gabor filters. Despite the original contribution of Granlund the most referred works are those conducted by Daugman [16, 17]. Daugman was the first who exclusively derived the uncertainty principle in two dimensions and showed the similarity between a structure based on the 2D Gabor functions and the organization and the characteristics of the mammalian visual system. Again, several simplifications are justifiable [39] and 2D Gabor function can be defined as

$$\begin{aligned}\psi(x, y) &= e^{-(\alpha^2 x'^2 + \beta^2 y'^2)} e^{j2\pi f_0 x'}, \\ x' &= x \cos \theta + y \sin \theta, \\ y' &= -x \sin \theta + y \cos \theta,\end{aligned}\tag{4.8}$$

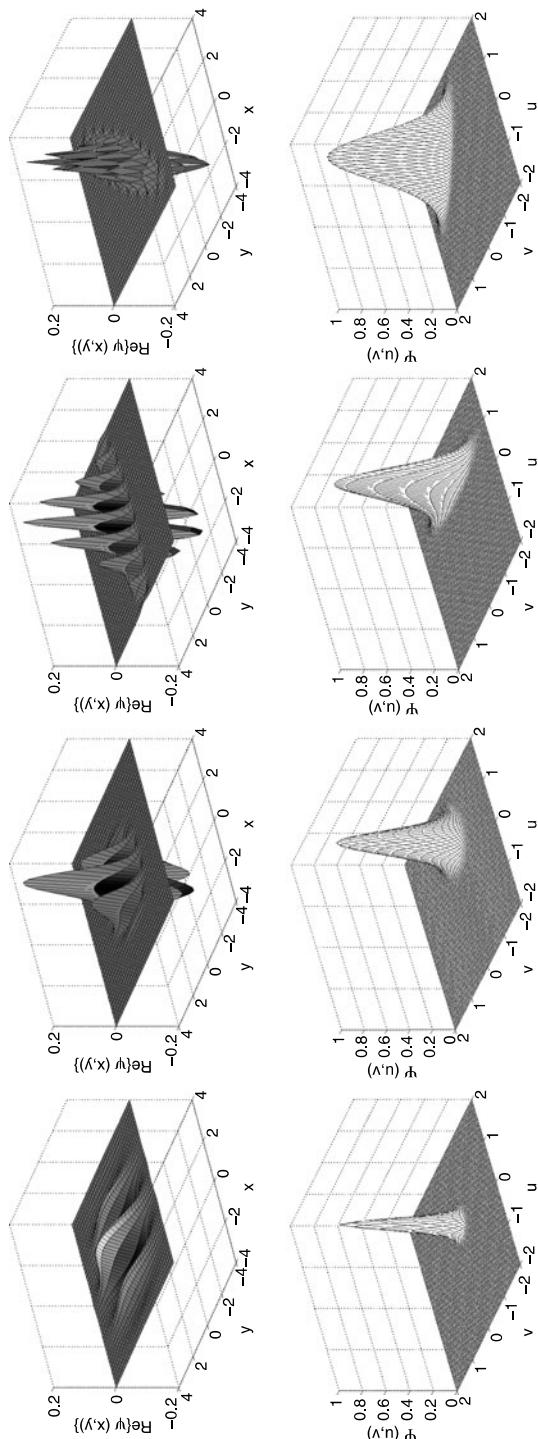
where the new parameters are  $\beta$  for sharpness of the second Gaussian axis and  $\theta$  for its orientation. In practice, the sharpness is connected to the frequency in order to make filters self-similar (Gabor wavelets) [39]. This is achieved by setting  $\alpha = |f_0|/\gamma$  and  $\beta = |f_0|/\eta$  and by normalizing the filter. Finally, the 2D Gabor filter in the spatial domain is

$$\begin{aligned}\psi(x, y) &= \frac{f^2}{\pi \gamma \eta} e^{-(\frac{f^2}{\gamma^2} x'^2 + \frac{f^2}{\eta^2} y'^2)} e^{j2\pi f x'}, \\ x' &= x \cos \theta + y \sin \theta, \\ y' &= -x \sin \theta + y \cos \theta,\end{aligned}\tag{4.9}$$

where  $f$  is the central frequency of the filter,  $\theta$  the rotation angle of the Gaussian major axis and the plane wave,  $\gamma$  the sharpness along the major axis, and  $\eta$  the sharpness along the minor axis (perpendicular to the wave). In the given form, the aspect ratio of the Gaussian is  $\eta/\gamma$ . The normalized 2D Gabor filter function has an analytical form in the frequency domain

$$\begin{aligned}\Psi(u, v) &= e^{-\frac{\pi^2}{f^2} (\gamma^2 (u' - f)^2 + \eta^2 v'^2)}, \\ u' &= u \cos \theta + v \sin \theta, \\ v' &= -u \sin \theta + v \cos \theta.\end{aligned}\tag{4.10}$$

The effects of the Gabor filter parameters, interpretable via the Fourier similarity theorem, are demonstrated in Fig. 4.11.



**Fig. 4.11** 2D Gabor filter functions with different values of the parameters  $f$ ,  $\theta$ ,  $\gamma$ , and  $\eta$  in the space (top) and frequency domains (bottom). *Left:* ( $f = 0.5$ ,  $\theta = 0^\circ$ ,  $\gamma = 1.0$ ,  $\eta = 1.0$ ); *middle-left:* ( $f = 1.0$ ,  $\theta = 0^\circ$ ,  $\gamma = 2.0$ ,  $\eta = 0.5$ ); *right:* ( $f = 1.0$ ,  $\theta = 0^\circ$ ,  $\gamma = 2.0$ ,  $\eta = 0.5$ )

### 4.4.3 Constructing Gabor Features

Gabor features are constructed by convolution of an input image  $\xi(x, y)$  with the filter in (4.9)

$$\begin{aligned} r_\xi(x, y; f, \theta) &= \psi(x, y; f, \theta) * \xi(x, y) \\ &= \int \int_{-\infty}^{\infty} \psi(x - x_\tau, y - y_\tau; f, \theta) \xi(x_\tau, y_\tau) dx_\tau dy_\tau. \end{aligned} \quad (4.11)$$

The convolution produces a response image  $r_\xi$  of the same size. Only a single filter rarely succeeds but the response images are computed for a “bank” of filters tuned on various frequencies and orientations. The frequencies are typically drawn from the logarithmic scale similar to wavelets [15]:

$$f_k = c^{-k} f_{\max}, \quad \text{for } k = 0, \dots, m - 1 \quad (4.12)$$

where  $f_{\max}$  is the maximum frequency (the smallest scale) and  $c$  is the frequency scaling factor. Some useful values for  $c$  include  $c = 2$  for octave spacing and  $c = \sqrt{2}$  for half-octave spacing. The filter orientations are spaced uniformly

$$\theta_k = \frac{k\pi}{n}, \quad k = \{0, \dots, n - 1\}. \quad (4.13)$$

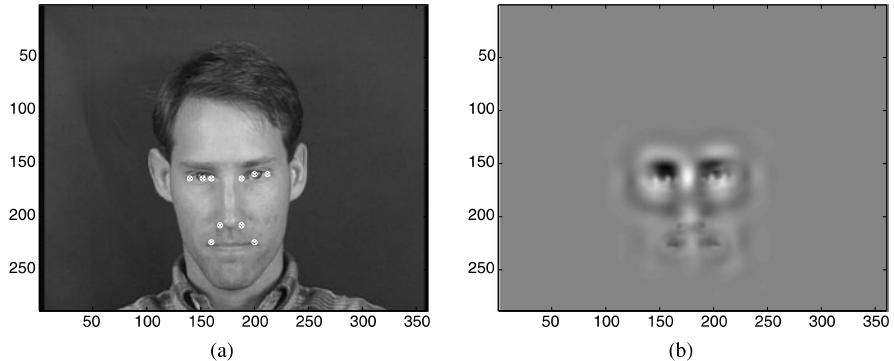
For real signals the responses on  $[\pi, 2\pi[$  are complex conjugates of responses on  $[0, \pi[$  and therefore only the responses for the half plane are needed:

$$\theta_k = \frac{k\pi}{n}, \quad k = \{0, \dots, n - 1\}. \quad (4.14)$$

For a bank of Gabor filters, the responses computed at a single location  $(x_0, y_0)$  with the parameters drawn from (4.12) and (4.14) a feature matrix  $\mathbf{G}$  can be constructed as

$$\mathbf{G} = \begin{pmatrix} r(x_0, y_0; f_0, \theta_0) & r(x_0, y_0; f_0, \theta_1) & \dots & r(x_0, y_0; f_0, \theta_{n-1}) \\ r(x_0, y_0; f_1, \theta_0) & r(x_0, y_0; f_1, \theta_1) & \dots & r(x_0, y_0; f_1, \theta_{n-1}) \\ \vdots & \vdots & \ddots & \vdots \\ r(x_0, y_0; f_{m-1}, \theta_0) & r(x_0, y_0; f_{m-1}, \theta_1) & \dots & r(x_0, y_0; f_{m-1}, \theta_{n-1}) \end{pmatrix}. \quad (4.15)$$

In (4.15) the columns denote responses over different orientations and rows over different frequencies (scales). This structure is called as “simple Gabor feature space” formally defined in [43], later revised in [39] and utilized in face detection in [32]. A significant simplification made in the proposed feature space is the use of only one spatial location  $(x', y')$  to represent an object. The assumption is justified if the objects are simple or if they are distinguishable from each other in the feature space. This is not the case with, for example, the human face, but seems to hold between salient sub-parts, such as nostrils, eyes, mouth corners, etc. The filters in one location tuned to various frequencies and orientations span a sub-space whose accuracy



**Fig. 4.12** Reconstruction from responses at 10 different locations (four orientations and five frequencies): **a** original; **b** reconstruction

decreases from the filter origin. This is demonstrated in Fig. 4.12 where an original face is reconstructed using filter responses from 10 locations.

Operations for rotation and scale invariant searches of objects can be defined as a column-wise circular shift of the response matrix corresponding to the rotation of the object around the location  $(x_0, y_0)$  and a row-wise shift corresponding to the scaling of an object by a factor  $c$  [43]. An illumination invariance can be achieved by normalizing the feature matrix [43].

#### 4.4.4 Learning Facial Features

In principle, Gabor features can be used similarly to LBPs or any other local features. The filter responses are computed for various frequencies and orientations, and a descriptor formed from the responses inside one or multiple fixed-size windows as illustrated in Fig. 4.6. For example, Zou et al. [103] proposed a face recognition method using such region descriptor and reported state-of-the-art results for the FERET database: fb: 99.5%, fc: 99.5%, dup I: 85.0% and dup II: 79.5%. Gabor face descriptor is easy to implement, but for completeness, in this section we concentrate on local facial features and utilize the simple feature matrix to represent and learn them.

We assume an annotated training set of face images. The annotations are, for example, the centroids of selected facial landmarks (see Fig. 4.12(a)). Any classifier or pattern recognition method can be used to learn the facial representations from extracted Gabor features. A completely statistical approach, however, possess superior properties as compared to other methods [37]: the decision making has an interpretable basis from which the most probable option can be chosen and a within-class comparison can be performed using statistical hypothesis testing [66]. In the statistical approaches, a class is typically represented in terms of a class conditional probability density function (pdf) over feature space. It should be noted, that finding

a proper pdf estimate has a crucial impact on the success of the facial feature detection. Typically, the form of the pdf's is somehow restricted and the estimation is reduced to a problem of fitting the restricted model to the observed features. Often simple models such as a single Gaussian distribution (normal distributed random variable) can efficiently represent features but a more general model, such as a finite mixture model, must be used to approximate more complex pdf's. We adopt the method in [37] where Gaussian mixture models represent facial feature conditional pdf's given the Gabor feature matrix.

The multiresolution Gabor feature in a single location can be converted from the matrix in (4.15) to a feature vector

$$\mathbf{g} = [r(x_0, y_0; f_0, \theta_0) \ r(x_0, y_0; f_0, \theta_1) \ \dots \ r(x_0, y_0; f_{m-1}, \theta_{n-1})]. \quad (4.16)$$

Since the feature vector is complex valued the complex Gaussian distribution function needs to be used,

$$\mathcal{N}^C(\mathbf{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{\pi^D |\Sigma|} \exp[-(\mathbf{x} - \boldsymbol{\mu})^* \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu})], \quad (4.17)$$

where  $\Sigma$  denotes the covariance matrix. It should be noted that the pure complex form of the Gaussian in (4.17) provides computational stability in the parameter estimation as compared to a concatenation of real and imaginary parts to two real numbers as the dimensionality of the problem doubles in the latter case [66]. Now, a Gaussian mixture model (GMM) probability density function can be defined as a weighted sum of Gaussians

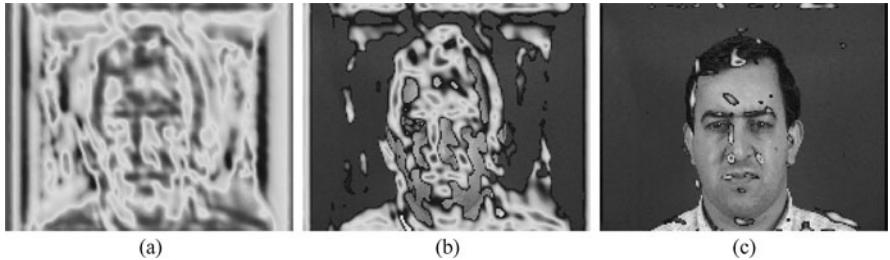
$$p(\mathbf{x}; \boldsymbol{\theta}) = \sum_{c=1}^C \alpha_c \mathcal{N}^C(\mathbf{x}; \boldsymbol{\mu}_c, \Sigma_c), \quad (4.18)$$

where  $\alpha_c$  is the weight of the  $c$ th component. The weight can be interpreted as *a priori* probability that a value of the random variable is generated by the  $c$ th source, and thus,  $0 \leq \alpha_c \leq 1$  and  $\sum_{c=1}^C \alpha_c = 1$ . The Gaussian mixture model probability density function can be completely defined by the parameter list

$$\boldsymbol{\theta} = \{\alpha_1, \boldsymbol{\mu}_1, \Sigma_1, \dots, \alpha_C, \boldsymbol{\mu}_C, \Sigma_C\}. \quad (4.19)$$

The main question remains how the parameters in (4.19) can be estimated from the given training data. The most popular estimation method is the expectation maximization (EM) algorithm, but the EM algorithm requires knowledge of the number of Gaussians,  $C$ , as an input parameter. The number is often unknown and this is a strong motivation to apply unsupervised methods, such as that of Figueiredo–Jain (FJ) [24] or the greedy EM algorithm [84]. Of the two unsupervised methods, the Figueiredo–Jain method provides more accurate results and its complex extension can be directly applied to pdf's of the complex feature vectors in (4.16) [66].

The probability distribution values, likelihoods, can be directly used to find the best or rank facial feature candidates [66]. It is even possible to reduce the search



**Fig. 4.13** Example of using density quantile of pdf values: **a** Pdf surface for the *left nostril* class; **b** Pdf values belonging to 0.5 density quantile; **c** Pdf values belonging to 0.05 density quantile [37]

---

**Algorithm 4.1:** Train facial feature classifier

---

```

1: for all Training images do
2:   Align and normalize image to represent an object in a standard pose
3:   Extract multiresolution Gabor features at given locations
4:   Normalize the features
5:   Store the features to the sample matrix  $P$  and their corresponding class
   labels to the target vector  $T$ 
6: end for
7: With samples in  $P$  estimate class conditional pdf's for each class using
   Gaussian mixture models and FJ algorithm

```

---

space considerably by discarding image features beyond a requested score level, that is, density quantile [66]. In Fig. 4.13, the use of density quantile for reducing the search space is demonstrated; it is clear that the spatial area corresponding to the 0.05 (0.95 confidence) density quantile contains the correct image feature.

#### 4.4.5 Detecting Facial Features

A supervised learning algorithm to extract simple Gabor features (multiresolution Gabor features) and to estimate the class conditional pdf's for the facial features is presented in Algorithm 4.1. Matlab functionality for efficient computation of the multiresolution Gabor features [76] and for the Gaussian mixture models and the FJ algorithm are publicly available [26]. In Algorithm 4.2, the main steps to extract the features from an image are shown.

**Experiments Using the XM2VTS Face Database** XM2VTS facial image database is a publicly available database for benchmarking face detection and recognition methods [58]. The frontal part of the database contains 600 training images and 560 test images of size  $720 \times 576$  (width  $\times$  height) pixels. For facial images ten

**Algorithm 4.2:** Extract  $K$  best face features of each class from an image  $I$ 


---

```

1: Compute multiresolution Gabor features  $G(x, y; f_m, \theta_n)$  for the whole image
    $I(x, y)$ 
2: for all Scale shifts do
3:   for all Rotation shifts do
4:     Shift Gabor features
5:     Normalize Gabor features
6:     Calculate confidence scores (pdf values) for all classes and for all  $(x, y)$ 
7:     Update feature class confidence at each location
8:   end for
9: end for
10: Sort the features by their score for each class
11: Return the  $K$  best features of each facial feature class

```

---

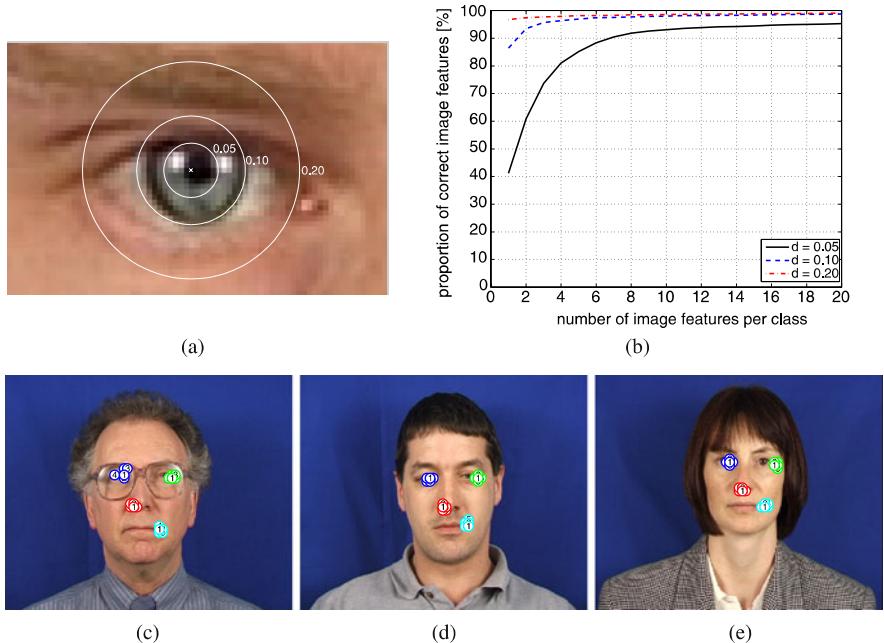
specific regions (see Fig. 4.12(a)) have been shown to have favorable properties to act as keypoints [32]. A normalized distance between the eyes, 1.0, will be used as measure of image feature detection accuracy. The distance measure is demonstrated in Fig. 4.14(a).

Gabor parameters were experimentally selected by using a cross-validation procedure over the training and evaluation sets in the database:  $n = 4$ ,  $m = 6$ ,  $k = \sqrt{3}$  and  $f_{\text{high}} = 1/40$ . Image features were extracted in a ranked order and a keypoint was considered to be correctly extracted if it was within a pre-set pixel distance limit from the correct location. Results with XM2VTS are presented in Fig. 4.14(b). The distances are scale normalized, so that the distance between centers of the eyes is 1.0 (see Fig. 4.14(a) for a demonstration). On average, 4 correct image features were included in the first 10 image features within distance limit 0.05, but as the number of features was increased to 100: over 9 for 0.05 and almost all features found for 0.10 and 0.20. It should be noted that accuracies of 0.10 and 0.20 are still very good for face registration and recognition. Increasing the number of image features over 100 (10 per class) did not improve the results anymore, but relaxing the distance limit to 0.10 almost perfect result were reached with only 10 first image features from each class. Typical detection results are demonstrated in Figs. 4.14(c)–(e).

Methods for accurate face and facial feature detection and localization based on the described Gabor representations have been proposed and reported to produce state-of-the-art detection accuracy for more difficult and realistic data sets (XM2VTS/non-frontal, BANCA and BioID) [32, 40].

## 4.5 Discussions on Local Features

A drawback of the LBP method, as well as of all local descriptors that apply vector quantization, is that they are not robust in the sense that a small change in the input



**Fig. 4.14** **a** Demonstration of accuracy distance measure; **b** Performance for facial feature detection in XM2VTS test images; **c, d, e** Examples of extracted features (left eye center: *blue*, right eye outer corner: *green*, left nostril: *red*, right mouth corner: *cyan*, 5 best feature for each landmark numbered from 1 to 5) [37]

image would always cause a small change in the output. LBP may not work properly for noisy images or on flat image areas of constant gray level. Many variants of LBP have been proposed to improve its robustness. For instance, Tan and Triggs proposed a three-level operator called local ternary patterns for example, to deal with problems on flat image areas [80]. Liao et al. [52] introduced dominant local binary patterns which make use of the most frequently occurred patterns of LBP to improve the recognition accuracy compared to the original uniform patterns. Raja and Gong proposed sparse multiscale local binary patterns to better exploit the discriminative capacity of multiscale features available [69]. Inspired by LBP, higher order local derivative patterns (LDP) were proposed by Zhang et al., with applications in face recognition [98].

LBP has also inspired the development of new effective local face descriptors, such as the Weber Law Descriptor (WLD) containing differential excitation and orientation components [13] and the blur-invariant Local Phase Quantization (LPQ) descriptor [65]. The LPQ descriptor has received wide interest in blur-invariant face recognition [5]. LPQ is based on quantizing the Fourier transform phase in local neighborhoods. Similarly to the widely used LBP based face description, histograms of LPQ labels computed within local regions are also adopted as a face descriptor. The experiments showed that such LPQ descriptors are highly discriminative

and produce very promising face recognition results, outperforming LBP both with blurred and sharp images on CMU PIE and FRGC 1.0.4 datasets.

A current trend in the development of new effective local face image descriptors is to combine the strengths of complementary descriptors. From the beginning, the LBP operator was designed as a complementary measure of local image contrast. Applying LBP to Gabor-filtered face images, or using LBP and Gabor methods jointly, have provided excellent results in face recognition [81, 96]. For instance, Zhang et al. [96] proposed the extraction of LBP features from images obtained by filtering a facial image with 40 Gabor filters of different scales and orientations. Excellent results have been obtained on the all FERET sets. A downside of the method lies in the high dimensionality of the feature vector (LBP histogram) which is calculated from 40 Gabor images derived from each single original image. To overcome this problem of large feature dimensions, Shan et al. [73] presented a new extension using Fisher Discriminant Analysis (FDA), instead of the  $\chi^2$  (Chi-square), and histogram intersection, which have been previously used in [96]. The authors constructed an ensemble of piecewise FDA classifiers, each of which is built based one segment of the high-dimensional LBP histograms. Impressive results were reported on the FERET database. Other works have also successfully exploited the complementary of Gabor filters and LBP features by fusing the two set of features e.g. for age classification [86]. Combining ideas from Haar and LBP features have also given excellent results in accurate and illumination invariant face detection [71, 91].

Features based on Gabor filters are very versatile. By post-processing they can be transformed, for example, to binary descriptors of texture similar to LBPs. For example, in the Daugman's iris code the response phase is quantized to two bits (four quadratures in the complex plane) [18]. The Daugman's descriptor is very discriminative and its histograms were used in face recognition in [97]. Utilization of the phase information is important for discrimination, but many other efficient post-processing methods exist in the literature and they are used in human visual system oriented recognition methods [72]. Another important property of Gabor filters is that the original signal can be reconstructed. This property was employed in this chapter where we introduced the efficient facial feature descriptor based on Gabor features at a single location. Recently, the importance of phase information have been noticed and very good recognition results reported for features based on Gabor phase [96]. It is important to notice that the complex-valued response, including both magnitude and phase, is the most natural representation, and should be used in methods based on Gabor filters.

## 4.6 Conclusions

Finding efficient facial or facial feature representations is a key issue in developing robust face recognition systems. Many methods have been proposed for this purpose. Local feature based methods seem to be more robust against variations in pose or illumination than holistic methods. Especially methods based on Gabor filter

responses and local binary patterns have been particularly successful in face image processing.

**Acknowledgements** Abdennour Hadid and Matti Pietikäinen thank the Academy of Finland for the financial support.

## References

1. Ahonen, T., Hadid, A., Pietikäinen, M.: Face recognition with local binary patterns. In: Proc. of the ECCV (2004)
2. Ahonen, T., Hadid, A., Pietikäinen, M.: Face description with local binary patterns: Application to face recognition. *IEEE Transactions on PAMI* **28**(12) (2006)
3. Ahonen, T., Pietikäinen, M.: Pixelwise local binary pattern models of faces using kernel density estimation. In: Proc. of the International Conference on Biometrics (2009)
4. Ahonen, T., Pietikäinen, M., Hadid, A., Mäenpää, T.: Face recognition based on the appearance of local regions. In: Proc. of the ICPR (2004)
5. Ahonen, T., Rahtu, E., Ojansivu, V., Heikkilä, J.: Recognition of blurred faces using local phase quantization. In: Proc. of the ICPR (2008)
6. Arca, S., Campadelli, P., Lanzarotti, R.: A face recognition system based on automatically terminated fiducial points. *Pattern Recognit.* **39**, 432–443 (2006)
7. Ashraf, A., Lucey, S., Chen, T.: Learning patch correspondences for improved viewpoint invariant face recognition. In: Proc. of the CVPR (2008)
8. Bastiaans, M.J.: Gabor’s signal expansion and the Zak transform. *Appl. Opt.* **33**(23), 5241–5255 (1994)
9. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Trans. PAMI* **19**(7) (1997)
10. Biciego, M., Lagorio, A., Grossi, E., Tistarelli, M.: On the use of SIFT features for face authentication. In: Proc. of the CVPR (2006)
11. Castrillón, M., Déniz, O., Hernández, D., Lorenzo, J.: A comparison of face and facial feature detectors based on the Viola–Jones general object detection framework. *Mach. Vis. Appl.* **22**, 481–494 (2011). doi:[10.1007/s00138-010-0250-7](https://doi.org/10.1007/s00138-010-0250-7)
12. Chan, C.-H., Kittler, J., Messer, K.: Multi-scale local binary pattern histograms for face recognition. In: Proc. of the International Conference on Biometrics (2007)
13. Chen, J., Shan, S., He, C., Zhao, G., Pietikäinen, M., Chen, X., Gao, W.: WLD: A robust local image descriptor. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(9), 1705–1720 (2010)
14. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. of the CVPR (2005)
15. Daubechies, I.: The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Inf. Theory* **36**(5) (1990)
16. Daugman, J.G.: Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *J. Opt. Soc. Am. A* **2**(7), 1160–1169 (1985)
17. Daugman, J.G.: Complete discrete 2-D Gabor transform by neural networks for image analysis and compression. *IEEE Trans. Acoust. Speech Signal Process.* **36**(7) (1988)
18. Daugman, J.: High confidence visual recognition of persons by a test of statistical independence. *IEEE Trans. PAMI* **25**(9) (1993)
19. Ding, L., Martinez, A.: Precise detailed detection of faces and facial features. In: Proc. of the CVPR (2008)
20. Ekenel, H., Stiefelhagen, R.: Analysis of local appearance-based face recognition: Effects on feature selection and feature normalization. In: Proc. of the CVPR (2006)
21. Etemad, K., Chellappa, R.: Discriminant analysis for recognition of human face images. *J. Opt. Soc. Am. A* **14**, 1724–1733 (1997)

22. Feichtinger, H., Strohmer, T. (eds.): *Gabor Analysis and Algorithms*. Birkhäuser, Basel (1998)
23. Feng, X., Pietikäinen, M., Hadid, A.: Facial expression recognition with local binary patterns and linear programming. *Pattern Recognit. Image Anal.* **15**(2), 546–548 (2005)
24. Figueiredo, M., Jain, A.: Unsupervised learning of finite mixture models. *IEEE Trans. PAMI* **24**(3) (2002)
25. Gabor, D.: Theory of communication. *J. Inst. Electr. Eng.* **93**, 429–457 (1946)
26. GMMBayes Toolbox for Matlab. <http://www2.it.lut.fi/project/gmmbayes>
27. Gökberk, B., Irfanoglu, M., Akarun, L., Alpaydin, E.: Learning the best subset of local features for face recognition. *Pattern Recognit.* **40**, 1520–1532 (2007)
28. Granlund, G.H.: In search of a general picture processing operator. *Comput. Graph. Image Process.* **8**, 155–173 (1978)
29. Hadid, A., Pietikäinen, M.: Combining appearance and motion for face and gender recognition from videos. *Pattern Recognit.* **42**(11), 2818–2827 (2009)
30. Hadid, A., Pietikäinen, M., Ahonen, T.: A discriminative feature space for detecting and recognizing faces. In: Proc. of the CVPR (2004)
31. Hadid, A., Zhao, G., Ahonen, T., Pietikäinen, M.: Face analysis using local binary patterns. In: Mirmehdi, M., Xie, X., Suri, J. (eds.) *Handbook of Texture Analysis*, pp. 347–373. Imperial College Press, London (2008)
32. Hamouz, M., Kittler, J., Kamarainen, J.-K., Paalanen, P., Kalviainen, H., Matas, J.: Feature-based affine-invariant localization of faces. *IEEE Trans. PAMI* **27**(9) (2005)
33. Hjelmas, E., Low, B.K.: Face detection: A survey. *Comput. Vis. Image Underst.* **83**(3), 236–274 (2001)
34. Hua, G., Akbarzadeh, A.: A robust elastic and partial matching metric for face recognition. In: Proc. of the ICCV (2009)
35. Huang, X., Li, S.Z., Wang, Y.: Shape localization based on statistical method using extended local binary pattern. In: Proc. of the International Conference on Image and Graphics (2004)
36. Huang, X., Li, S., Wang, Y.: Jensen–Shannon boosting learning for object recognition. In: Proc. of the CVPR (2005)
37. Ilonen, J., Kamarainen, J.-K., Paalanen, P., Hamouz, M., Kittler, J., Kälviäinen, H.: Image feature localization by multiple hypothesis testing of Gabor features. *IEEE Trans. Image Process.* **17**(3) (2008)
38. Jain, A., Chen, Y., Demirkus, M.: Pores and ridges: Fingerprint matching using level 3 features. *IEEE Trans. PAMI* **29**(1) (2007)
39. Kamarainen, J.-K., Kyrki, V., Kälviäinen, H.: Invariance properties of Gabor filter based features—overview and applications. *IEEE Trans. Image Process.* **15**(5), 1088–1099 (2006)
40. Kamarainen, J.-K., Hamouz, M., Kittler, J., Paalanen, P., Ilonen, J., Drobchenko, A.: Object localisation using generative probability model for spatial constellation and local image features. In: Proc. of the ICCV Workshop on Non-Rigid Registration and Tracking Through Learning (2007)
41. Kanade, T.: Picture processing system by computer complex and recognition of human faces. Doctoral dissertation, Kyoto University (1973)
42. Kozakaya, T., Shibata, T., Yuasa, M., Yamaguchi, O.: Facial feature localization using weighted vector concentration approach. *Image Vis. Comput.* **28**, 772–780 (2010)
43. Kyrki, V., Kamarainen, J.-K., Kälviäinen, H.: Simple Gabor feature space for invariant object recognition. *Pattern Recognit. Lett.* **25**(3), 311–318 (2003)
44. Lades, M., Vorbrüggen, J.C., Buhmann, J., Lange, J., von der Malsburg, C., Würtz, R.P., Konen, W.: Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. Comput.* **42**, 300–311 (1993)
45. LBP bibliography. [http://www.ee.oulu.fi/mvg/page/lbp\\_bibliography](http://www.ee.oulu.fi/mvg/page/lbp_bibliography)
46. Lee, P.-H., Hsu, G.-S., Hung, Y.-P.: Face verification and identification using facial trait code. In: Proc. of the CVPR (2009)
47. Li, S.Z., Jain, A.K. (eds.): *Handbook of Face Recognition*. Springer, New York (2005)

48. Li, S., Zhao, C., Zhu, X., Lei, Z.: Learning to fuse 3D + 2D based face recognition at both feature and decision levels. In: Proc. of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures (2005)
49. Li, S.Z., Chu, R., Liao, S., Zhang, L.: Illumination invariant face recognition using near-infrared images. *IEEE Trans. PAMI* **29**(4) (2007)
50. Liang, L., Xiao, R., Wen, F., Sun, J.: Face alignment via component-based discriminative search. In: Proc. of the ECCV (2008)
51. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block local binary patterns for face recognition. In: Proc. of the International Conference on Biometrics (2007)
52. Liao, S., Law, M., Chung, A.: Dominant local binary patterns for texture classification. *IEEE Trans. Image Process.* **18**(5), 1107–1118 (2009)
53. Liu, C.-C., Dai, D.-Q.: Face recognition using dual-tree complex wavelet features. *IEEE Trans. Image Process.* 2593–2599 (2009)
54. Ma, B., Zhang, W., Shan, S., Chen, X., Gao, W.: Robust head pose estimation using LGBP. In: Proc. of the ICPR (2006)
55. Manjunath, B., Ohm, J.R., Vinod, V.V., Yamada, A.: Color and texture descriptors. *IEEE Trans. Circuits Syst. Video Technol.* **11**(6) (2001). Special Issue on MPEG-7
56. McCool, C., Marcel, S.: Parts-based face verification using local frequency bands. In: Proc. of the International Conference on Biometrics (2009)
57. Messer, K. et al.: Face authentication test on the BANCA database. In: Proc. of the ICPR (2004)
58. Messer, K., Matas, J., Kittler, J., Luettin, J., Maitre, G.: XM2VTSDB: The extended M2VTS database. In: Proc. of the International Conference on Audio and Video-based Biometric Person Authentication (1999)
59. Meyers, E., Wolf, L.: Using biologically inspired features for face processing. *Int. J. Comput. Vis.* **76**, 93–104 (2008)
60. Meynet, J., Popovici, V., Thiran, J.-P.: Face detection with boosted Gaussian features. *Pattern Recognit.* **40**, 2283–2291 (2007)
61. Mian, A., Bennamoun, M., Owens, R.: Keypoint detection and local feature matching for textured 3d face recognition. *Int. J. Comput. Vis.* **79**, 1–12 (2008)
62. Moghaddam, B., Nastar, C., Pentland, A.: A Bayesian similarity measure for direct image matching. In: Proc. of the ICPR (1996)
63. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on feature distributions. *Pattern Recognit.* **29**, 51–59 (1996)
64. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. PAMI* **24** (2002)
65. Ojansivu, V., Heikkilä, J.: Blur insensitive texture classification using local phase quantization. In: Proc. of the International Conference on Image and Signal Processing (2008)
66. Paalanen, P., Kamarainen, J.-K., Ilonen, J., Kälviäinen, H.: Feature representation and discrimination based on Gaussian mixture model probability densities—practices and algorithms. *Pattern Recognit.* **39**(7), 1346–1358 (2006)
67. Phillips, P., Moon, H., Rizvi, S.A., Rauss, P.J.: The FERET evaluation methodology for face-recognition algorithms. *IEEE Trans. PAMI* **22** (2000)
68. Pinto, N., DiCarlo, J., Cox, D.: How far can you get with a modern face recognition test set using only simple features? In: Proc. of the CVPR (2009)
69. Raja, Y., Gong, S.: Sparse multiscale local binary patterns. In: Proc. of the British Machine Vision Conference (2006)
70. Rodriguez, Y., Marcel, S.: Face authentication using adapted local binary pattern histograms. In: Proc. of the ECCV (2006)
71. Roy, A., Marcel, S.: Haar local binary pattern feature for fast illumination invariant face detection. In: Proc. of the British Machine Vision Conference (2009)
72. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Object recognition with cortex-like mechanisms. *IEEE Trans. PAMI* **29**(3) (2007)
73. Shan, S., Zhang, W., Su, Y., Chen, X., Gao, W.: Ensemble of piecewise FDA based on spatial histograms of local (Gabor) binary patterns for face recognition. In: Proc. of the ICPR (2006)

74. Shan, C., Gong, S., McOwan, P.: Facial expression recognition based on local binary patterns:a comprehensive study. *Image Vis. Comput.* **27**(6), 803–816 (2009)
75. Shastri, B., Levine, M.: Face recognition using localized features based on non-negative sparse coding. *Mach. Vis. Appl.* **18**, 107–122 (2007)
76. SimpleGabor Toolbox for Matlab. <http://www2.it.lut.fi/project/simplegabor>
77. Su, Y., Shan, S., Chen, X., Gao, W.: Hierarchical ensemble of global and local classifiers for face recognition. *IEEE Trans. Image Process.* **18**(8) (2009)
78. Sun, N., Zheng, W., Sun, C., Zou, C., Zhao, L.: Gender classification based on boosting local binary pattern. In: Proc. of the International Symposium on Neural Networks (2006)
79. Sun, Z., Tan, T., Qiu, X.: Graph matching iris image blocks with local binary pattern. In: Proc. of the International Conference on Biometrics (2006)
80. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. In: ICCV Workshop on Analysis and Modeling of Faces and Gestures (2007)
81. Tan, X., Triggs, B.: Fusing Gabor and LBP feature sets for kernel-based face recognition. In: Proc. of the ICCV Workshop on Analysis and Modeling of Faces and Gestures (2007)
82. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**, 71–86 (1991)
83. Varma, M., Zisserman, A.: Texture classification: Are filter banks necessary? In: Proc. of the CVPR (2003)
84. Verbeek, J.J., Vlassis, N., Kröse, B.: Efficient greedy learning of Gaussian mixture models. *Neural Comput.* **5**(2), 469–485 (2003)
85. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition, pp. 511–518 (2001)
86. Wang, J., Yau, W., Wang, H.: Age categorization via ECOC with fused Gabor and LBP features. In: Proc. of the IEEE Workshop on Applications of Computer Vision (2009)
87. Wang, P., Ji, Q.: Multi-view face and eye detection using discriminant features. *Comput. Vis. Image Underst.* **105**, 99–111 (2007)
88. Winder, S., Brown, M.: Learning local image descriptors. In: Proc. of the CVPR (2007)
89. Wiskott, L., Fellous, J.-M., Krüger, N., von der Malsburg, C.: Face recognition by elastic bunch graph matching. *IEEE Trans. PAMI* **19** (1997)
90. Xu, Z., Chen, H., Zhu, S.-C., Luo, J.: A hierarchical compositional model for face representation and sketching. *IEEE Trans. PAMI* **30**(6) (2008)
91. Yan, S., Shan, S., Chen, X., Gao, W.: Locally assembled binary (LAB) feature with feature-centric cascade for fast and accurate face detection. In: Proc. of the CVPR (2008)
92. Yang, J., Zhang, D., Frangi, A.F., Yu Yang, J.: Two-dimensional PCA: A new approach to appearance-based face representation and recognition. *IEEE Trans. PAMI* **26** (2004)
93. Zhang, X., Jia, Y.: Face recognition with local steerable phase feature. *Pattern Recognit. Lett.* **27**, 1927–1933 (2006)
94. Zhang, G., Wang, Y.: Faceprint: Fusion of local features for 3d face recognition. In: Proc. of the International Conference on Biometrics (2009)
95. Zhang, G., Huang, X., Li, S.Z., Wang, Y., Wu, X.: Boosting local binary pattern LBP-based face recognition. In: Proc. of the Chinese Conference on Biometric Recognition (2004)
96. Zhang, W., Shan, S., Gao, W., Chen, X., Zhang, H.: Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition. In: Proc. of the ICCV (2005)
97. Zhang, B., Shan, S., Chen, X., Gao, W.: Histogram of Gabor phase patterns (HGPP): A novel object representation approach for face recognition. *IEEE Trans. Image Process.* **16**(1) (2007)
98. Zhang, B., Gao, Y., Zhao, S., Liu, J.: Local derivate pattern versus local binary pattern: Face recognition with high-order local pattern descriptor. *Image Vis. Comput.* **28**, 772–780 (2010)
99. Zhao, G., Pietikäinen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Trans. PAMI* **29**(6) (2007)
100. Zhao, G., Pietikäinen, M.: Boosted multi-resolution spatiotemporal descriptors for facial expression recognition. *Pattern Recognit. Lett.* **30**, 1117–1127 (2009)

101. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: Face recognition: A literature survey. *ACM Comput. Surv.* **34**(4), 399–458 (2003)
102. Zhao, G., Barnard, M., Pietikäinen, M.: Lipreading with local spatiotemporal descriptors. *IEEE Trans. Multimed.* **11**(7) (2009)
103. Zou, J., Ji, Q., Nagy, G.: A comparative study of local matching approach for face recognition. *IEEE Trans. Image Process.* **16**(10) (2007)

# Chapter 5

## Face Alignment Models

Phil Tresadern, Tim Cootes, Chris Taylor, and Vladimir Petrović

### 5.1 Introduction

In building models of facial appearance, we adopt a statistical approach that learns the ways in which the shape and texture of the face vary across a range of images. We rely on obtaining a suitably large and representative training set of images of faces, each of which is annotated with a set of feature points that define correspondences across the set. The positions of the feature points also define the shape of the face, and are analysed to learn the ways in which the shape can vary. The patterns of intensities are analysed in a similar way to learn how the texture can vary. The result is a model which is capable of synthesising any of the training images and generalising from them, but is specific enough that only face-like images are generated.

To build a statistical appearance model, we require a set of training images that covers the types of variation we want the model to represent. For instance, if we are only interested in faces with neutral expressions, we need only include neutral expressions in the model. If, however, we want to synthesise and recognise a range of expressions, the training set should include images of people smiling, frowning, winking and so on. Ideally, the faces in the training set should be of at least as high a resolution as those in the images we wish to synthesise or interpret.

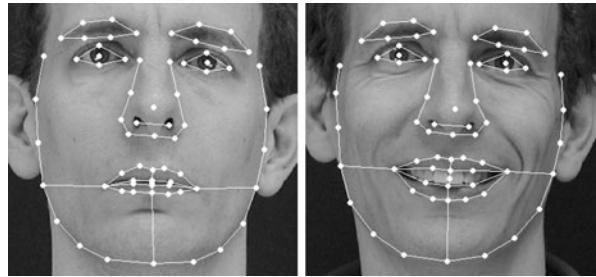
#### 5.1.1 Statistical Models of Shape

To define a shape model, we first annotate each face with a fixed number of points that define the key facial features (and their correspondences across the training set) and represent the shape of the face in the image. Typically, we place points around

---

P. Tresadern · T. Cootes (✉) · C. Taylor · V. Petrović  
Imaging Science and Biomedical Engineering, University of Manchester, Manchester, UK  
e-mail: [t.cootes@man.ac.uk](mailto:t.cootes@man.ac.uk)

**Fig. 5.1** Examples of 68 points defining facial features on two frontal images



the main facial features (eyes, nose, mouth and eye-brows) together with points that define the boundary of the face (Fig. 5.1). The more points we use, the more subtle the variations in shape that we can represent.

If we annotate the face with  $n$  feature points,  $\{(x_i, y_i)\}$ , then we can represent the geometry of the face with a  $2n$  element vector,

$$\mathbf{x} = (x_1, \dots, x_n, y_1, \dots, y_n)^T, \quad (5.1)$$

such that  $N$  training images provide  $N$  training vectors  $\mathbf{x}_i$ . The *shape* of the face can then be defined as that property of the configuration of points which is invariant under (that is, not explained by) some global transformation. In other words, if  $S_t(\mathbf{x})$  applies a transformation defined by parameters  $t$  to the points  $\mathbf{x}$ , the configurations of points defined by  $\mathbf{x}$  and  $S_t(\mathbf{x})$  are considered to have the same shape. Typically, we use either the similarity transformation or the affine transformation where a 2D similarity has four parameters ( $x$ - and  $y$ -translation, rotation and scaling) and a 2D affine transformations has six ( $x$ - and  $y$ -translation, rotation, scaling, aspect ratio and skew).

Given a set of shapes as training data, we can then apply formal statistical techniques [23] to analyse their variation and synthesise new shapes that are similar. Before we perform statistical analysis on these training vectors, however, it is important that we first remove any differences that are attributable to the global transformation,  $S_t(\mathbf{x})$ , leaving only genuine differences in shape (that is, we must align the shapes into a common coordinate frame).

### 5.1.1.1 Aligning Sets of Shapes

Of the various methods of aligning shapes into a common coordinate frame, the most popular is *Procrustes Analysis* [27] that finds the parameters,  $t$ , that transform each shape in the set,  $\mathbf{x}_i$ , so that it is aligned with a mean shape,  $\bar{\mathbf{x}}$ , in the sense that minimises their sum of squared distances,  $D = \sum_i |S(\mathbf{x}_i) - \bar{\mathbf{x}}|^2$ . Though we can solve this analytically for a *set* of shapes, in practice we use a simple and effective iterative approach (Algorithm 5.1) to converge on a solution. At each iteration, we ensure that this measure is well defined by aligning the mean shape to the coordinate frame such that it is centred at the origin, has unit scale and some fixed but arbitrary orientation.

---

**Algorithm 5.1:** Aligning a set of shapes

---

- 1: Translate each example so that its centre of gravity is at the origin.
  - 2: Choose one example as an initial estimate of the mean shape,  $\bar{\mathbf{x}}$ , and scale so that  $|\bar{\mathbf{x}}| = 1$ .
  - 3: Record this first estimate as  $\bar{\mathbf{x}}_0$  to define the default reference frame.
  - 4: **repeat**
  - 5:   Align each shape with the current estimate of the mean shape.
  - 6:   Re-estimate mean from aligned shapes.
  - 7:   Apply constraints on the current estimate of the mean by aligning it with  $\bar{\mathbf{x}}_0$  and scaling so that  $|\bar{\mathbf{x}}| = 1$ .
  - 8: **until** converged (that is, the change in the mean since the previous iteration is sufficiently small).
- 

### 5.1.1.2 Linear Models of Shape Variation

We now have  $N$  training vectors,  $\mathbf{x}_i$ , each with  $n$  points in  $d$  dimensions (usually  $d = 2$  or  $d = 3$ ) that are aligned to a common coordinate frame. By treating these vectors as points in  $nd$ -dimensional space and modelling their distribution, we can generate new examples that are similar to those in the original training set (that is, sample from the distribution) and examine new shapes to decide whether they are plausible examples (that is, evaluate a sample's probability).

Not every  $nd$ -dimensional vector forms a face, however, and the set of valid face shapes typically lies on a manifold that can be described by  $k < nd$  underlying model parameters,  $\mathbf{b}$ . By computing a parameterised model of the form  $\mathbf{x} = M(\mathbf{b})$ , we can approximate the distribution over shape,  $p(\mathbf{x})$ , with the distribution over model parameters,  $p(\mathbf{b})$ , and therefore generate new shapes by sampling from  $p(\mathbf{b})$  and applying the model, or evaluate a shape's probability by evaluating the probability of its corresponding parameters,  $p(M^{-1}(\mathbf{x}))$ . The simplest approximation we can make to the manifold is a linear subspace that passes through the mean shape (equivalent to assuming a Gaussian distribution over both  $\mathbf{x}$  and  $\mathbf{b}$ ). An effective approach to estimating the subspace parameters is to apply Principal Component Analysis (PCA) to the training vectors,  $\mathbf{x}_i$  (Algorithm 5.2). This computes the directions of greatest variance in  $nd$ -dimensional space, of which we keep only the  $k$  most ‘significant’. Each training example can then be described by its  $k < nd$  projections onto these directions, reducing the effective dimensionality of the data.

We can then approximate any example shape from the training set,  $\mathbf{x}$  using

$$\mathbf{x} \approx \bar{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s \quad (5.4)$$

where  $\mathbf{P}_s = (\phi_1 | \phi_2 | \dots | \phi_k)$  contains the eigenvectors corresponding to the  $k$  largest eigenvalues and  $\mathbf{b}_s$  is a  $k$ -dimensional vector given by

$$\mathbf{b}_s = (\mathbf{P}_s^T \mathbf{P}_s)^{-1} \mathbf{P}_s^T (\mathbf{x} - \bar{\mathbf{x}}) = \mathbf{P}_s^T (\mathbf{x} - \bar{\mathbf{x}}) \quad (5.5)$$

---

**Algorithm 5.2:** Principal component analysis
 

---

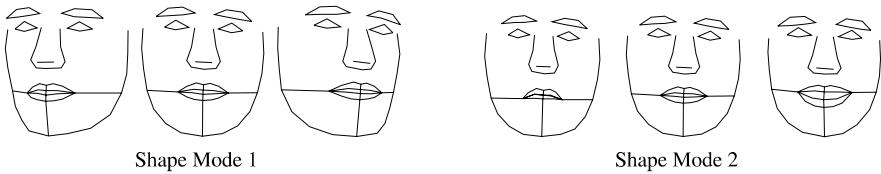
- 1: Compute the mean of the data,

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i. \quad (5.2)$$

- 2: Compute the covariance of the data,

$$\mathbf{S} = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T. \quad (5.3)$$

- 3: Compute the orthonormal eigenvectors,  $\phi_j$ , and their corresponding eigenvalues,  $\lambda_j$ , of  $\mathbf{S}$  (sorted such that  $\lambda_j \geq \lambda_{j+1}$ ). Efficient methods of computing the eigenvectors and eigenvalues exist for the case in which there are fewer samples than dimensions in the vectors [11].
- 



**Fig. 5.2** Two modes of a face shape model (varied by  $\pm 2$  s.d. from the mean). The first shape mode corresponds mostly to 3D head rotation whereas the second captures facial expression

where  $\mathbf{P}_s^T \mathbf{P}_s = \mathbf{I}$  because the eigenvectors are orthonormal. The vector  $\mathbf{b}_s$  therefore defines a set of parameters of a deformable shape model and by varying the elements of  $\mathbf{b}_s$  we can vary the generated shape,  $\mathbf{x}$ , via (5.4).

Given a shape in the model frame,  $\mathbf{x} = M(\mathbf{b})$ , we can then generate the corresponding shape in the image frame,  $\mathbf{X}$ , by applying a suitable transformation such that  $\mathbf{X} = S_t(\mathbf{x})$ . Typically  $S_t$  will be a similarity transformation described by a scaling,  $s$ , an in-plane rotation,  $\theta$ , and a translation,  $(t_x, t_y)$ . By representing the scaling and rotation jointly as  $(s_x, s_y)$ , where  $s_x = (s \cos \theta - 1)$  and  $s_y = s \sin \theta$ , we ensure that the pose parameter vector,  $\mathbf{t} = (s_x, s_y, t_x, t_y)^T$ , is zero for the identity transformation and that  $S_t(\mathbf{x})$  is linear in the pose parameters.

Due to the ordering of the eigenvalues,  $\lambda_j$ , the corresponding modes of shape variation are sorted in descending order of ‘importance’. For example, a model built from examples of a single individual with different viewpoints and expressions will often select 3D rotation of the head (that causes a large change in the projected shape) as the most significant mode, followed by expression (Fig. 5.2).

### 5.1.1.3 Choosing the Number of Shape Modes

The number of modes,  $k$ , that we keep determines how much meaningful shape variation (and how much meaningless noise) is represented by the model, and should therefore be chosen with care. A simple approach is to choose the smallest  $k$  that explains a given proportion (e.g., 98%) of the total variance exhibited in the training set,  $V_T = \sum \lambda_j$ , where each eigenvalue  $\lambda_j$  gives the variance of the data about the mean in the direction of the  $j$ th eigenvector. More specifically, it is normal to choose the smallest  $k$  that satisfies

$$\sum_{j=1}^k \lambda_j \geq f_v \cdot V_T \quad (5.6)$$

where  $f_v$  defines the proportion of  $V_T$  that we want to explain (e.g.,  $f_v = 0.98$ ). Alternatively, we could build a sequence of models with increasing  $k$  and choose the smallest model that approximates all training examples to within a given accuracy (e.g., at most one pixel of error for any feature point). Performing this test in a miss-one-out manner—testing each example against models built from all other examples—gives us further confidence in the chosen  $k$ .

### 5.1.1.4 Fitting the Model to New Points

Suppose now we wish to find the best pose and shape parameters to align a model instance  $\mathbf{x}$  to a new set of image points,  $\mathbf{X}'$ . Minimising the sum of square distances between corresponding model and image points is equivalent to minimising the expression

$$E_{\text{pts}} = |\mathbf{X}' - S_t(\bar{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s)|^2 \quad (5.7)$$

or, more generally,

$$E_{\text{pts}} = (\mathbf{X}' - S_t(\bar{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s))^T \cdot \mathbf{W}_{\text{pts}} \cdot (\mathbf{X}' - S_t(\bar{\mathbf{x}} + \mathbf{P}_s \mathbf{b}_s)) \quad (5.8)$$

where  $\mathbf{W}_{\text{pts}}$  is a diagonal matrix that applies a different weight for each point. If the allowed global transformation  $S_t(\cdot)$  is more complex than a simple translation then this is a non-linear equation with no analytic solution. A good approximation can be found rapidly, however, by using a two-stage iterative approach (Algorithm 5.3) where each step solves a linear equation for common choices of transformation (e.g., similarity or affine) [30].

If the weights in  $\mathbf{W}_{\text{pts}}$  relate to the uncertainty in the estimates of positions of target points,  $\mathbf{X}'$ , then  $E_{\text{pts}}$  can be related to the log likelihood. Adding a term representing the prior distribution on the shape parameters,  $p(\mathbf{b}_s)$ , then gives the log posterior,

$$E'_{\text{pts}} = E_{\text{pts}} - 2 \log p(\mathbf{b}_s). \quad (5.9)$$

---

**Algorithm 5.3:** Shape model fitting
 

---

- 1: **repeat**
  - 2:   Solve for the pose parameters,  $\mathbf{t}$ , assuming a fixed shape,  $\mathbf{b}_s$ .
  - 3:   Solve for the shape parameters,  $\mathbf{b}_s$ , assuming a fixed pose,  $\mathbf{t}$ .
  - 4: **until** convergence.
- 

If the training shapes,  $\{\mathbf{x}_i\}$ , have a multivariate Gaussian distributed then the parameters,  $\mathbf{b}$ , will have an axis-aligned Gaussian distribution,  $p(\mathbf{b}) = N(\mathbf{0}, \Lambda)$ , where  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_k)$  such that

$$\log p(\mathbf{b}_s) \propto \sum_{j=1}^k b_j^2 / \lambda_j + \text{const.} \quad (5.10)$$

This prior, however, biases shape parameters toward zero which may be unjustified. Therefore, a common approach is to apply a uniform elliptical prior over shape parameters such that

$$\log p(\mathbf{b}_s) = \begin{cases} \text{const} & \text{if } \mathbf{b}^T \Lambda^{-1} \mathbf{b} < \tau, \\ \infty & \text{otherwise} \end{cases} \quad (5.11)$$

where a suitable limit can be estimated from the data. In this case, maximising the posterior amounts to a linear projection (giving the maximum likelihood solution), followed by truncation of the corresponding shape parameters to lie within the elliptical bounds.

### 5.1.1.5 Further Reading

Our experiments on 2D images suggest that a Gaussian distribution over shape parameters (implying a linear subspace of face shapes) is a good approximation as long as the training set contains only modest viewpoint variation. Nonlinear changes in shape such as introduced by large viewpoint variation [15], result in a linear subspace model that captures all of the required variation but in doing so also permits invalid shapes that lie off the nonlinear manifold [43].

To account explicitly for 3D head rotation (or viewpoint change), several studies have computed a 3D linear shape model and fitted it to new image data under perspective [5, 61] or orthogonal [43] projection. These approaches separate all rigid movement of the head from nonrigid shape deformation while maintaining some linearity for efficient computation (see Chap. 6 for more details on explicit 3D models).

Other studies avoid modelling nonlinearities explicitly, instead opting to use methods that estimate the parameters of the nonlinear manifold directly. Early examples outside of the face domain include polynomial regression [54] and mixture

models [8] whereas later studies applied nonlinear PCA [24], Kernel PCA [46], the Gaussian Process Latent Variable Model [32] and tensor-based models [36] to the face alignment problem.

### 5.1.2 Statistical Models of Texture

Though the *shape* of a face may give a weak indication of identity, the *texture* of the face provides a far stronger cue for recognition. We therefore apply similar techniques to those used to build a shape model in order to build a model of texture, given a set of training images. Fitting the texture model to new image data then summarising the properties of the underlying face (including its identity) through the texture model parameters.

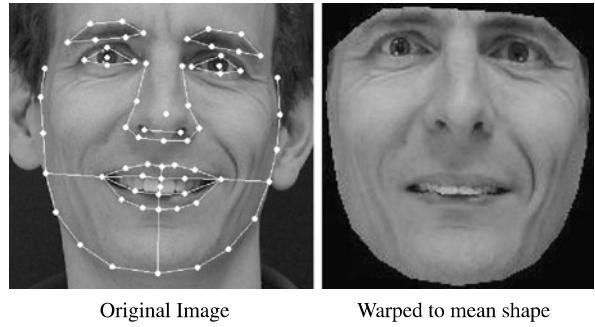
#### 5.1.2.1 Aligning Sets of Textures

Given a set of training images of faces that are coarsely-aligned (e.g., with respect to similarity transformations only), it has been shown that a linear subspace-based face model [34] provides a useful representation for recognition [58]. However, this coarse alignment does not compensate for nonrigid variation in shape due to identity, pose or expression. As a result, corresponding pixels over the set of training images actually originate from different points on the face (or possibly even the background) and spurious texture variation creeps into the model, reducing recognition performance [19].

To address this problem, we use the correspondences between facial *features* (e.g., eyes, nose and mouth) over the set of labelled training images to define an approximate correspondence between the *pixels* in the underlying image [4, 18]. In particular, we apply a continuous deformation—such as an interpolating spline or a piece-wise affine warp using a triangulation of the region—to warp each training image so that its feature points match a reference shape (typically the mean shape). The intensity information is then sampled from the *shape-normalised* image over the region covered by the mean shape (Fig. 5.3) to form a texture vector,  $\mathbf{g}_{im}$ . Since  $\mathbf{g}_{im}$  is defined in the normalised shape frame, it has a fixed number of pixels,  $n_{\text{pixels}}$ , that is independent of the size of the object in the target image.

This nonlinear sampling (Algorithm 5.4) applies a *geometric* alignment of the textures, ensuring that corresponding elements over the set of texture vectors represent corresponding points on the face so that computed image statistics are meaningful. As in the case of the shape model, however, we want our texture model to represent only those changes that cannot be explained by a global transformation (e.g., due to changes in brightness and contrast). We therefore apply a *photometric* alignment of the texture samples before computing the image statistics that will define our texture model.

**Fig. 5.3** Example of face warped to the mean shape. Although the main shape variations due to smiling have been removed, there is considerable texture difference from a purely neutral face




---

#### Algorithm 5.4: Texture sampling

---

- 1: Precompute the positions of the sample pixels (typically all pixels in the region of interest) in the model reference frame,  $(x_{s,i}, y_{s,i})$ .
  - 2: Construct a warping function,  $W_{\mathbf{X}}(x, y)$  which maps the points of the mean shape onto the target points,  $\mathbf{X}$ .
  - 3: For each element  $i$  in  $\mathbf{g}_{im}$  sample, the target image at  $W_{\mathbf{X}}(x_{s,i}, y_{s,i})$  using interpolation if appropriate.
- 

More specifically, we express a texture in the image frame as a 1D affine transformation,  $T_{\mathbf{u}}(\cdot)$ , of the corresponding model texture,  $\mathbf{g}$ , such that

$$\mathbf{g}_{im} = T_{\mathbf{u}}(\mathbf{g}) = (1 + u_1) \cdot \mathbf{g} + u_2 \cdot \mathbf{1} \quad (5.12)$$

where  $\mathbf{u} = (u_1, u_2)^T$  is a vector of parameters corresponding to contrast and brightness, and  $\mathbf{u} = \mathbf{0}$  gives the identity transformation. Unlike shape normalisation, this transformation is linear and we can find a closed-form solution for parameters to give the sampled vector,  $\mathbf{g}_s$ , zero sum and unit variance is the number of elements in the vectors. The normalised texture vector in the model frame is then given by the inverse transformation,::

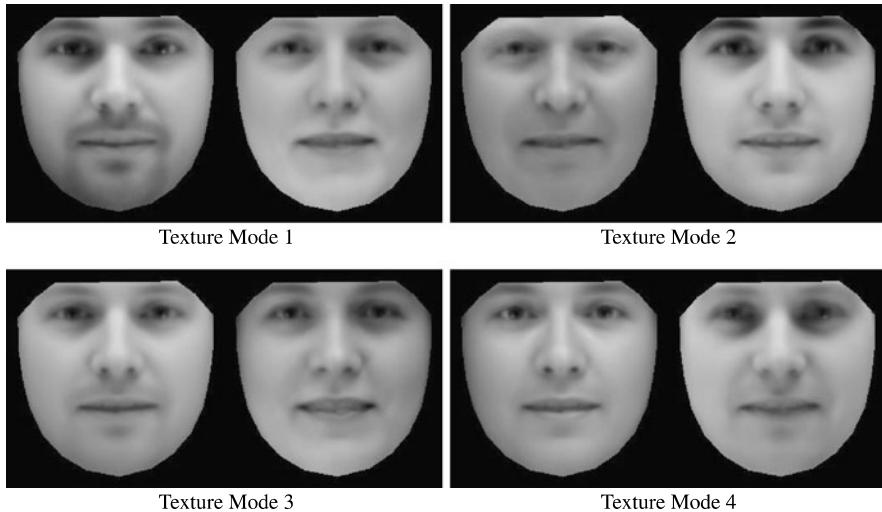
$$\mathbf{u}_2 = (\mathbf{g}_{im} \cdot \mathbf{1}) / n_{\text{pixels}}, \quad (5.13)$$

$$1 + u_1 = |\mathbf{g}_{im}|^2 / n_{\text{pixels}} - u_2^2 \quad (5.14)$$

where  $n_{\text{pixels}}$  is the number of elements in the vectors. The normalised texture vector in the model frame is then given by the inverse transformation,  $\mathbf{g}_s = T_{\mathbf{u}}^{-1}(\mathbf{g}_{im})$ :

$$\mathbf{g}_s = (\mathbf{g}_{im} - u_2 \cdot \mathbf{1}) / (1 + u_1). \quad (5.15)$$

For colour images, each plane can be normalised separately though we have found that grey-scale models are able to generalise to unseen images more effectively than colour models.



**Fig. 5.4** Four modes of a face texture model built from 400 images (including neutral, smiling, frowning and surprised expressions) of 100 different individuals with around 20 000 pixels per example. Texture parameters have been varied by  $\pm 2$  standard deviations from the mean

### 5.1.2.2 Linear Models of Texture Variation

Once we have compensated for the effects of brightness and contrast, we apply PCA to the set of normalised texture vectors to obtain a linear subspace model of texture,

$$\mathbf{g} \approx \bar{\mathbf{g}} + \mathbf{P}_g \mathbf{b}_g, \quad (5.16)$$

where  $\bar{\mathbf{g}}$  is the mean texture over the training set,  $\mathbf{P}_g$  is a set of orthogonal *modes of texture variation* and  $\mathbf{b}_g$  is a vector of texture parameters. We can then generate a variety of plausible, shape-normalised face textures (Fig. 5.4) by varying  $\mathbf{b}_g$  within limits learnt from the training set. Brightness and contrast variation can then be added by varying  $\mathbf{u}$  and applying (5.12):

$$\mathbf{g}_{im} = (1 + u_1) \cdot (\bar{\mathbf{g}} + \mathbf{P}_g \mathbf{b}_g) + u_2 \cdot \mathbf{1}. \quad (5.17)$$

### 5.1.2.3 Choosing the Number of Texture Modes

As with the shape model, the simplest means of choosing the number of texture modes is to keep the smallest number of modes needed to capture a fixed proportion (e.g., 98%) of the total texture variation in the training set. Since the number of elements in the texture vector is typically much higher than in a shape vector, the texture model usually needs many more modes than the shape model to capture the same proportion of variance—278 modes were needed to capture 98% of the variance in our example (Fig. 5.4).

---

**Algorithm 5.5:** Fitting a texture model to new data
 

---

- 1: Compute the global transformation parameters,  $\mathbf{u}$ , using (5.13) and (5.14).
  - 2: Compute the texture model parameters,  $\mathbf{b}_g = \mathbf{P}_g^T(T_{\mathbf{u}}^{-1}(\mathbf{g}_{im}) - \bar{\mathbf{g}})$ .
- 

### 5.1.2.4 Fitting the Model to New Textures

Like the shape model, fitting the texture model to new data proceeds in a two-step algorithm (Algorithm 5.5) and the model fitting to  $\mathbf{g}_{im}$  is then given by (5.17). Unlike when fitting a shape model, however, no iteration is required for the texture model.

### 5.1.2.5 Further Reading

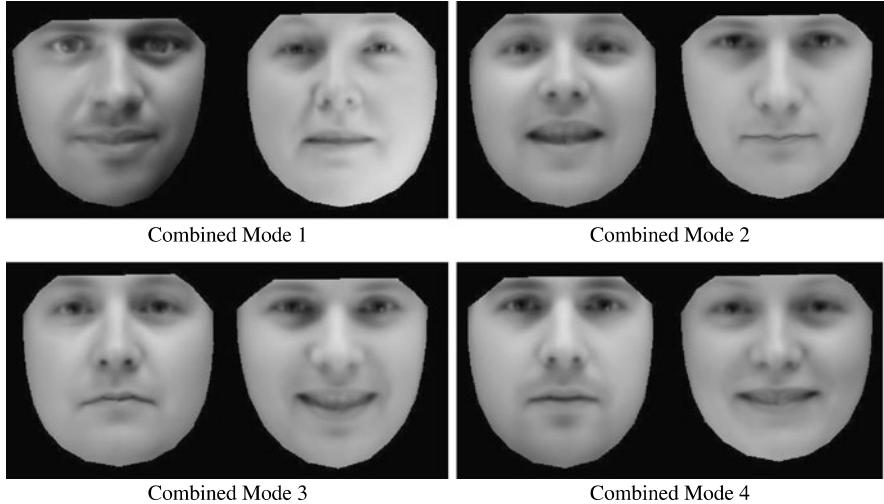
Though raw image intensities (or colour values) are adequate for most applications, modelling local image gradients may offer improved performance since gradients yield more information, are less sensitive to lighting and seem to favour edges over flat regions. If we compute local image gradients  $(g_x, g_y)$  via a straightforward linear transformation of the intensities, however, the subsequent Principal Component Analysis effectively reverses this transformation such that the basis images are almost identical to those obtained from raw intensities (apart from some boundary effects).

Instead, robust matching was demonstrated using a *non-linearly* normalised gradient at each pixel [10]:  $(g'_x, g'_y) = (g_x, g_y)/(g + g_0)$  where  $g$  is the magnitude of the gradient, and  $g_0$  is the mean gradient magnitude over a region. Other approaches have demonstrated improved performance by combining multiple feature bands such as intensity, hue and edge information [56], by including features derived from measures of ‘cornerness’ [53] and by learning filters that give smooth error surfaces [35].

Like shapes, textures also lie on a low-dimensional, nonlinear manifold embedded in the high-dimensional texture space [40]. As a result, linear methods such as PCA often cannot capture sufficient variance in the training set without also permitting invalid textures. If the training set is such that this becomes a problem (e.g., when significant viewpoint variation is present), multilinear [60] and nonlinear methods such as Locally Linear Embedding [47], IsoMap [57] and Laplacian Eigenmaps [3] may be useful (though probabilistic interpretation of such methods is nontrivial).

## 5.1.3 Combined Models of Appearance

The shape and texture of any example in a normalised frame can thus be summarised by the parameter vectors,  $\mathbf{b}_s$  and  $\mathbf{b}_g$ , and though shape and texture may be considered independently [33], this can miss informative correlations between shape and



**Fig. 5.5** Four modes of combined shape and texture model built from the same 400 face images as the texture-only model (Fig. 5.4). Combined parameters were varied by  $\pm 2$  standard deviations from the mean

texture variations (e.g., square jaws correlating with facial hair). We therefore model these correlations by concatenating shape and texture parameter vectors into a single vector,

$$\mathbf{b} = \begin{pmatrix} \mathbf{W}_s \mathbf{b}_s \\ \mathbf{b}_g \end{pmatrix} = \begin{pmatrix} \mathbf{W}_s \mathbf{P}_s^T (\mathbf{x} - \bar{\mathbf{x}}) \\ \mathbf{P}_g^T (\mathbf{g} - \bar{\mathbf{g}}) \end{pmatrix}, \quad (5.18)$$

where  $\mathbf{W}_s$  is a diagonal matrix of weights for each shape parameter, that accounts for the difference in units between the shape and texture models (see Sect. 5.1.3.1). We then apply a PCA on these combined vectors to give a model

$$\mathbf{b} \approx \mathbf{P}_c \mathbf{c} = \begin{pmatrix} \mathbf{P}_{cs} \\ \mathbf{P}_{cg} \end{pmatrix} \mathbf{c} \quad (5.19)$$

where  $\mathbf{P}_c$  are the eigenvectors and  $\mathbf{c}$  is a vector of *appearance* parameters (with zero-mean by construction) that jointly controls both the shape and texture of the model. Note that the linear nature of the model allows us to express the shape and grey-levels directly as functions of  $\mathbf{c}$

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{Q}_s \mathbf{c} \quad \text{where } \mathbf{Q}_s = \mathbf{P}_s \mathbf{W}_s^{-1} \mathbf{P}_{cs}, \quad (5.20)$$

$$\mathbf{g} = \bar{\mathbf{g}} + \mathbf{Q}_g \mathbf{c} \quad \text{where } \mathbf{Q}_g = \mathbf{P}_g \mathbf{P}_{cg}. \quad (5.21)$$

An example image can then be synthesised for a given  $\mathbf{c}$  by generating the shape-free, grey-level image from the vector  $\mathbf{g}$  and warping it using the control points described by  $\mathbf{x}$  to give images that combine variations due to identity, lighting, viewpoint and expression (Fig. 5.5).

### 5.1.3.1 Choosing Shape Parameter Weights

In the combined model, the elements of  $\mathbf{b}_s$  have units of distance whereas those of  $\mathbf{b}_g$  have units of intensity. As a result, applying unweighted PCA to the concatenated parameter vectors may incorrectly place greater emphasis on capturing variation in one more than the other. To address this problem, we first scale the shape parameters via the weighting matrix,  $\mathbf{W}_s$ , so that the units of  $\mathbf{b}_s$  and  $\mathbf{b}_g$  are comparable.

A simple approach to choosing  $\mathbf{W}_s$  is to set  $\mathbf{W}_s = r\mathbf{I}$  where  $r^2$  is the ratio of the total intensity variation to the total shape variation in the normalised frames. A more systematic approach is to measure the effect of varying  $\mathbf{b}_s$  on the sample  $\mathbf{g}$  by displacing each element of  $\mathbf{b}_s$  from its optimum value for each training example and sampling the image given the displaced shape; the RMS change in  $\mathbf{g}$  per unit change in shape parameter  $b_s$  gives the weight  $w_s$  to be applied to that parameter in (5.18). In practice, however, we have found that synthesis and search algorithms are relatively insensitive to the choice of  $\mathbf{W}_s$ .

### 5.1.3.2 Separating Sources of Variability

In many applications, some sources of appearance variation are more useful than others. In face recognition, for example, variations due to identity are essential whereas variations due to other sources (e.g., expression) are a nuisance whose effects we want to minimise. Since the combined appearance model mixes these two sources, each element of the parameter vector encodes both between- and within-identity variation. The sources can, however, be separated by splitting the subspace defined by  $\mathbf{P}_c$  into two orthogonal subspaces,

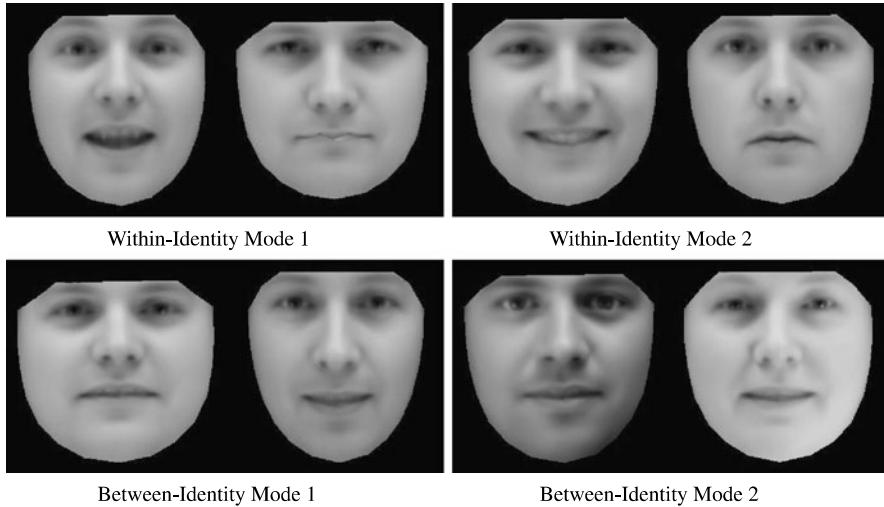
$$\mathbf{c} = \mathbf{P}_b \mathbf{c}_b + \mathbf{P}_w \mathbf{c}_w, \quad (5.22)$$

where  $\mathbf{P}_b$  and  $\mathbf{c}_b$  encode between-identity variation, and  $\mathbf{P}_w$  and  $\mathbf{c}_w$  encode within-identity variation [24].

Computing the within-identity subspace is straightforward if we know the identity of the person in every training image—the columns of  $\mathbf{P}_w$  are the eigenvectors of the covariance matrix computed using the deviation of each  $\mathbf{c}$  from the mean appearance vector *for the same identity*. Varying  $\mathbf{c}_w$  then indirectly changes  $\mathbf{c}$  and thus the appearance of the face but only in ways that a specific individual's face can change, such as expression (Fig. 5.6, top).

The orthogonal subspace,  $\mathbf{P}_b$ , that represents between-identity variation can then be computed by subtracting the within-identity variation,  $\mathbf{P}_w \mathbf{P}_w^T \mathbf{c}$ , and doing PCA over the resulting appearance parameter vectors. Alternatively, the between-class covariance matrix can be computed using the set of identity-specific means, though the mean is not guaranteed to be free of corruption by some non-neutral expression or head pose (Fig. 5.6, bottom). Iterative methods have also shown success in separating different sources of variability [16].

In contrast, tensor-based methods keep sources of variability separate at all times by building a multilinear model of texture variation [36, 60]. These methods, however, have strict requirements in terms of training data—namely, every combination



**Fig. 5.6** (*Top*) Two within-identity modes of individual face variation; (*bottom*) two between-identity modes of variation between individuals. Some residual variation in expression is present due to not every mean face being completely neutral

of variation must be present in the training set (that is, every expression at every pose for every identity).

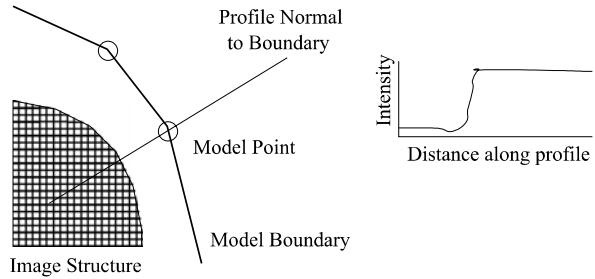
## 5.2 Active Shape Models (ASMs)

Once we have built a statistical shape model from labelled training images, we need a method of matching the model to an unseen image of the face so that we can interpret the underlying properties of the image. One method, known as the *Active Shape Model* (ASM) [11], does this by alternating between locally searching for features to maximise a ‘goodness of fit’ measure and regularising the located shape to filter out spurious local matches caused by noisy data.

### 5.2.1 Goodness of Fit

Given a set of shape parameter values,  $\mathbf{b}_s$ , and pose parameters,  $\mathbf{t}$ , we can define the shape of the object in the image frame. If we also define a measure of how well given parameters explain the observed image data, we can find ‘better’ parameter values by searching in a local region around each feature point to find alternative feature locations that match the model more closely. In general, we can model appearance with a 2D patch centred at the feature location and search a 2D region of interest around the current estimate for better matches (see Sect. 5.2.5).

**Fig. 5.7** At each model point, we sample along a profile normal to the boundary



In the specific case of the Active Shape Model [11], however, we reduce computational demands by looking along 1D linear profiles that pass through each model point and are normal to the model boundary (Fig. 5.7). If we assume that the model boundary corresponds to an edge, the strongest edge along the profile suggests a new location for the model point. Model points, however, are not always found on the strongest edge in the locality—they may instead be associated with a weaker secondary edge or some other image structure—and so instead we learn from the training set what to look for in the target image.

One popular method is to build a statistical model of the grey-level structure along the profile, normal to the boundary in the training set. Suppose for a given point we sample along a profile  $k$  pixels either side of the model point in the  $i$ th training image. We then have  $2k + 1$  samples which can be put in a vector  $\mathbf{g}_i$ . To avoid the effects of a constant offset in the intensities (that is, differences in brightness), we sample the derivative along the profile rather than the absolute grey-level values. We similarly compensate for changes in contrast by dividing through by the sum of absolute element values such that

$$\mathbf{g}_i \rightarrow \frac{1}{\sum_j |g_{ij}|} \mathbf{g}_i. \quad (5.23)$$

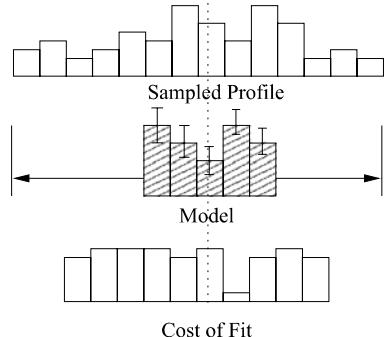
We repeat this for every training image to get a set of normalised samples,  $\{\mathbf{g}_i\}$ , whose distribution we can then model. If we assume that these profile samples have a multivariate Gaussian distribution, for example, we can build a statistical model of the grey-level profiles by computing their mean,  $\bar{\mathbf{g}}$ , and covariance,  $\mathbf{S}_g$ . The quality of fit of a new sample,  $\mathbf{g}_s$ , to the model is then given by the Mahalanobis distance of the sample from the model mean,

$$f(\mathbf{g}_s) = (\mathbf{g}_s - \bar{\mathbf{g}})^T \mathbf{S}_g^{-1} (\mathbf{g}_s - \bar{\mathbf{g}}), \quad (5.24)$$

and is related to the negative log of the probability that  $\mathbf{g}_s$  is drawn from the learned distribution such that minimising  $f(\mathbf{g}_s)$  is equivalent to finding the maximum likelihood solution.

In practice, when performing a local search for a given feature point we first sample a profile of  $m > k$  pixels either side of the current estimate. We then test the quality of fit of the corresponding grey-level model to each of the  $2(m - k) + 1$  possible positions along the sample and choose the one which gives the best match

**Fig. 5.8** Search along sampled profile to find best fit of grey-level model




---

#### Algorithm 5.6: Active Shape Model (ASM) fitting

---

```

1: repeat
2:   Examine a region of the image around each point,  $\mathbf{X}_i$ , to find the best
      nearby match,  $\mathbf{X}'_i$  (see Sect. 5.2.1).
3:   Update the parameters  $(\mathbf{t}, \mathbf{b}_s)$  to fit the shape model to the newly found
      local matches  $\mathbf{X}'$  (see Sect. 5.1.1.4).
4: until converged (that is, change in regularised estimate is sufficiently
   small).

```

---

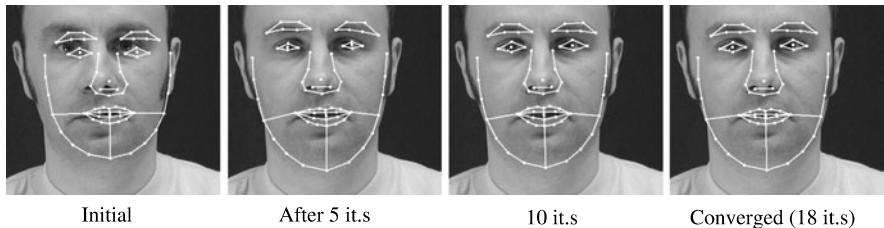
(as shown in Fig. 5.8) that is, the lowest value of  $f(\mathbf{g}_s)$ . Repeating this for each feature point gives a new estimate for the shape of the face.

### 5.2.2 Iterative Model Refinement

Local searches on their own, however, are prone to spurious matches due to noisy data and unmodelled image properties. To ensure that the estimated shape agrees with the statistical model learned from training data (see Sect. 5.1.1), we regularise our solution by fitting the shape model to the local matches. Hopefully, this regularised estimate is closer to the true solution such that repeating the search-regularise cycle gives progressively better estimates (Algorithm 5.6). Since each point also has a quality of match score, given by (5.24), these scores can be used to weight points differently during model fitting, as in (5.8), according to our belief in their reliability [30].

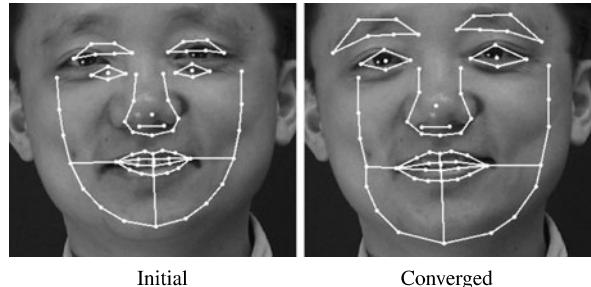
### 5.2.3 Multi-Resolution Active Shape Models

To avoid local minima when searching the image, it is useful to smooth the error function in early stages and reduce the level of smoothing gradually with each



**Fig. 5.9** Successful search for a face using the Active Shape Model

**Fig. 5.10** Failure of the Active Shape Model to localise a face where the search profiles are not long enough to locate the edges of face



iteration. In practice, we apply this smoothing by implementing the ASM in a multi-resolution framework using a Gaussian image pyramid. This involves first searching for the object in a coarse image, then refining the shape in a series of progressively finer resolution images. Not only is this more robust to local minima but also more efficient, since less complex models can be used at the coarse levels of the pyramid.

#### 5.2.4 Examples of ASM Search

In one example of an ASM search to locate the features of a face (Fig. 5.9), we place the model instance near the centre of the image and perform a coarse to fine search, starting on the 3rd level of a Gaussian pyramid ( $1/8$  the resolution in  $x$  and  $y$  compared to the original image). In the first few iterations, large improvements are made that get the position and scale roughly correct. As the search progresses, however, more subtle adjustments to the shape are made using the finer resolution images. After 18 iterations (with at most 10 iterations per pyramid level), the process has converged and gives a good match to the target image.

In another example, the ASM fails to localise the face (Fig. 5.10). This is most likely due to the initialisation being too far from the true solution such that the correct feature positions are beyond the scope of the local search and the process falls into a local minimum.

### 5.2.5 Further Reading

The Active Shape Model can be viewed as a specific example of a ‘constrained local model’ (CLM)—a class of algorithms that perform a local search for each feature (based on an independent set of learned texture models) then fit a learned shape model to the set of local matches. Addressing susceptibility to local minima, however, has been a driving force for various modifications to the match metric and search algorithm.

Although profile gradients have proven to be effective for local search, discriminative models of profile intensity can distinguish between correct and incorrect matches and improve performance further [59]. Better still, using 2D patches instead of 1D profiles makes the model even more discriminative [44], based on measures such as normalised correlation [21], boosted classification [20] or mixtures of linear experts [50] to define a match score. Where to look for potential matches is usually defined by hand (e.g., a rectangular or elliptical grid) but may also be learned from training data [38].

Once a response surface (that is, the set of match scores for all candidate locations) has been computed for each point, the ASM naïvely picks the best match for each point before projecting the set of matches back onto the subspace of permitted shapes. Effectively, this approximates each response surface by a Gaussian likelihood function with diagonal covariance; by including off-diagonal terms, we can model directional uncertainty and can further improve performance [41, 45]. If the response surface is not approximated by a parametric function, or is approximated by a complex function such as a mixture of Gaussians [28] or nonparametric kernel density estimate [51], the match function may be optimised using iterative methods such as gradient-free optimisers such as the Nelder–Mead Simplex method [21] or mean-shift [51].

When using a PCA model of shape, each feature imposes constraints on every other feature such that computational limitations force us to select the match for each point independently of all other points. By assuming conditional independence between features, however, we can reduce the complexity of the graph and use Markov Random Field methods at little or no cost in efficiency [37]. Simplifying the graph in this way allows us to consider multiple candidates for each feature point and therefore increase robustness by avoiding local minima due to spurious matches that do not agree with the possible matches for other feature points. When choosing which dependencies to eliminate, trees [25] and  $k$ -fans [17] are popular due to their simplicity though more effective graph structures may be learned from training data [29].

## 5.3 Active Appearance Models (AAMs)

One criticism of the approaches related to the ASM is that they use only sparse local information around the points of interest. In addition, they often treat the information at each point as independent which is rarely the case. These criticisms

---

**Algorithm 5.7:** Image residual computation
 

---

- 1: Use (5.21) to compute the shape,  $\mathbf{x}$ , and texture,  $\mathbf{g}_m$ , in the normalised reference frame.
  - 2: Compute the shape in the image frame by applying  $\mathbf{X} = S_t(\mathbf{x})$ .
  - 3: Sample the target image in the region defined by  $\mathbf{X}$  to get  $\mathbf{g}_{im}$  (see Sect. 5.1.2.1).
  - 4: Normalise the sampled texture with respect to brightness and contrast using  $\mathbf{g}_s = T_u^{-1}(\mathbf{g}_{im})$ .
  - 5: Compute the residual,  $\mathbf{r}(\mathbf{p}) = \mathbf{g}_s - \mathbf{g}_m$ .
- 

are largely addressed by the following approach—dubbed the *Active Appearance Model* (AAM) [14]—that uses a combined model of appearance (Sect. 5.1.3) for image interpretation via the *interpretation through synthesis* paradigm: if we can find appearance model parameters which synthesise a face very similar to that in the target image, those parameters summarise the shape and texture of the face and can therefore be used directly for interpretation. In contrast to the Active Shape Model, the Active Appearance Model directly *predicts* incremental updates to appearance parameters from image residuals rather than performing a local search, making the method very efficient.

### 5.3.1 Goodness of Fit

Given combined appearance model parameters,  $\mathbf{c}$ , a set of pose parameters,  $\mathbf{t}$ , and a set of texture normalisation parameters,  $\mathbf{u}$ , we can concatenate the parameters into a single vector,  $\mathbf{p} = (\mathbf{c}^T | \mathbf{t}^T | \mathbf{u}^T)^T$ , synthesise a new face image and compute the residual,  $\mathbf{r}(\mathbf{p}) = \mathbf{g}_s - \mathbf{g}_m$ , with respect to the observed data. We then assess the quality of the synthesis (Algorithm 5.7) by some function of  $\mathbf{r}(\mathbf{p})$ , such as the sum of squared error,

$$E_{\text{sse}}(\mathbf{p}) = |\mathbf{r}(\mathbf{p})|^2 = \mathbf{r}(\mathbf{p})^T \mathbf{r}(\mathbf{p}), \quad (5.25)$$

as used in our examples. Like the ASM, we also can make assumptions about the distributions of residuals to estimate  $p(\mathbf{r} | \mathbf{p})$  and place the matching in a Bayesian framework [9].

### 5.3.2 Updating Model Parameters

Given one estimate of the parameters,  $\mathbf{p} = \mathbf{p}^* + \delta\mathbf{p}$  (where  $\delta\mathbf{p}$  is our displacement from the true solution,  $\mathbf{p}^*$ ), and the corresponding residual,  $\mathbf{r}(\mathbf{p})$ , we then want to modify the parameters by  $\delta\mathbf{p}$  to minimise  $|\mathbf{r}(\mathbf{p} - \delta\mathbf{p})|^2$ . Though we could do this

via gradient descent [33], the AAM instead assumes that  $\delta\mathbf{p}$  can be predicted linearly from the residual vector such that  $\delta\mathbf{p} = \mathbf{R}\mathbf{r}(\mathbf{p})$ . In this section, we present two approaches to learning the matrix  $\mathbf{R}$  from training data that consists of random parameter displacements,  $\{\delta\mathbf{p}\}$  (stored in the columns of a matrix,  $\mathbf{C}$ ), and the corresponding residuals,  $\{\mathbf{r}(\mathbf{p}^* + \delta\mathbf{p})\}$  (stored in the columns of a matrix,  $\mathbf{V}$ ).

Since we want the model to be independent of the background in the training images, perturbed texture samples that include pixels from the background must be accounted for when building the model. One approach is to remove background pixels from the update model though we use the simpler alternative of setting background pixels to some random value.

### 5.3.2.1 Estimating $\mathbf{R}$ via Linear Regression

Given parameter displacements,  $\mathbf{C}$ , and the corresponding image residuals,  $\mathbf{V}$ , a linear update relationship gives

$$\mathbf{C} = \mathbf{RV} \Rightarrow \mathbf{R} = \mathbf{CV}^\dagger \quad (5.26)$$

where  $\mathbf{V}^\dagger$  is the pseudo-inverse of  $\mathbf{V}$  [12].

Unless, however, there are more displacements than pixels modelled (a rare occurrence) the model will overfit to the training data. To address this problem, applying PCA to reduce the dimensionality of the residuals (and effectively increase sampling density) before performing the regression has been shown to reduce overfitting and improve performance [31]. Alternatively, rather than projecting onto a lower dimensional subspace that maximises the variance of the projected inputs (that is, image residuals), Canonical Component Analysis (CCA) improves performance further [22] by computing subspaces for both inputs and outputs (that is, parameter displacements) that maximises the correlation between their respective projections.

### 5.3.2.2 Estimating $\mathbf{R}$ via Gauss–Newton Approximation

An alternative way to avoid overfitting is suggested by the first order Taylor expansion,

$$\mathbf{r}(\mathbf{p} - \delta\mathbf{p}) = \mathbf{r}(\mathbf{p}) - \frac{\partial \mathbf{r}}{\partial \mathbf{p}} \delta\mathbf{p}, \quad (5.27)$$

where the  $ij$ th element of the matrix  $\frac{\partial \mathbf{r}}{\partial \mathbf{p}}$  is  $\frac{\partial r_i}{\partial p_j}$  such that  $|\mathbf{r}(\mathbf{p} - \delta\mathbf{p})|^2$  is minimised with respect to  $\delta\mathbf{p}$  by the RMS solution,

$$\delta\mathbf{p} = \mathbf{R}\mathbf{r}(\mathbf{p}) \quad \text{where } \mathbf{R} = \left( \frac{\partial \mathbf{r}^T}{\partial \mathbf{p}} \frac{\partial \mathbf{r}}{\partial \mathbf{p}} \right)^{-1} \frac{\partial \mathbf{r}^T}{\partial \mathbf{p}}. \quad (5.28)$$

In a standard optimisation scheme, we would recalculate  $\frac{\partial \mathbf{r}}{\partial \mathbf{p}}$  at every step—a computationally expensive operation. Since it is being computed in a normalised reference frame, however, we assume that it is approximately fixed and can be pre-computed from the training set [14]. In practice, we express (5.27) in terms of the training data,  $\mathbf{C}$  and  $\mathbf{V}$ , to give

$$\frac{\partial \mathbf{r}}{\partial \mathbf{p}} = \arg \min_{\mathbf{J}} \|\mathbf{V} - \mathbf{JC}\|_F^2 \quad \Rightarrow \quad \frac{\partial \mathbf{r}}{\partial \mathbf{p}} = \mathbf{VC}^\dagger \quad (5.29)$$

where  $\|\cdot\|_F$  denotes the Frobenius norm. This Gauss–Newton approximation is popular because computing the pseudoinverse  $\mathbf{C}^\dagger$  is usually quicker and more robust than computing  $\mathbf{V}^\dagger$  due to their relative sizes. We then precompute  $\mathbf{R}$  via (5.28) and use it in all subsequent image searches. To ensure a reliable estimate, we measure residuals at displacements of differing magnitudes (typically up to 0.5 standard deviations of each parameter) and combine them by smoothing with a Gaussian kernel. Qualitatively, computing the update via a Gauss–Newton approximation should be more stable, has a clearer mathematical interpretation and allows extra constraints to be incorporated easily [9]. Quantitatively, however, tests comparing the different approaches [7] have shown that using linear regression gives better localisation performance.

### 5.3.3 Iterative Model Refinement

Given an initial estimation of the model parameters,  $\mathbf{c}$ , the pose,  $\mathbf{t}$ , and the texture transformation,  $\mathbf{u}$ , we repeatedly apply (5.28) to update model parameters based on the measured residual,  $\mathbf{r}$ , giving estimates that get progressively closer to the true solution (Algorithm 5.8).

When we update the parameter vector  $\mathbf{p} = (\mathbf{c}^T | \mathbf{t}^T | \mathbf{u}^T)^T$ , the simplest approach is to subtract a the predicted displacement  $\delta \mathbf{p} = (\delta \mathbf{c}^T | \delta \mathbf{t}^T | \delta \mathbf{u}^T)^T$  such that  $\mathbf{p} \rightarrow \mathbf{p} - \delta \mathbf{p}$ . The update step, however, estimates corrections in the *model* frame which must then be projected into the image frame using the current pose and texture transformations. Strictly speaking, therefore, we should update the parameters controlling the pose,  $\mathbf{t}$ , and texture transformation,  $\mathbf{u}$ , by *composing* the resulting transformations (during both training and image search). In other words, we should compute pose parameters  $\mathbf{t}'$  such that  $S_{\mathbf{t}'}(\mathbf{x}) = S_{\mathbf{t}}(S_{\delta \mathbf{t}}(\mathbf{x}))$  and new texture transformation parameters  $\mathbf{u}'$  such that  $T_{\mathbf{u}'}(\mathbf{g}) = T_{\mathbf{u}}(T_{\delta \mathbf{u}}(\mathbf{g}))$  where updates are applied in the model frame before transforming to the image frame.

### 5.3.4 Multi-Resolution Active Appearance Models

As in the Active Shape Model, we estimate the appearance models and update matrices at a range of image resolutions using a Gaussian image pyramid. We can then

---

**Algorithm 5.8:** Active Appearance Model (AAM) fitting

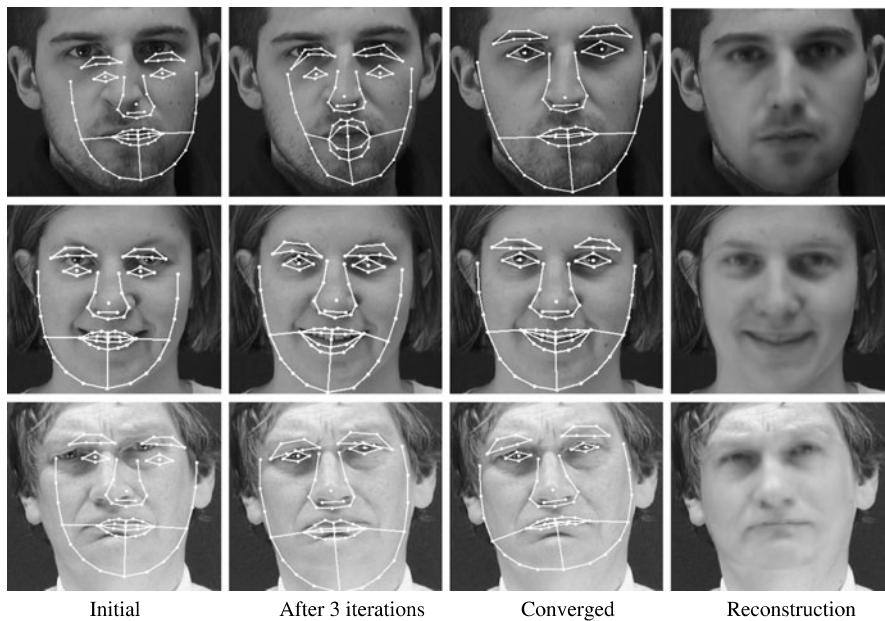
---

- 1: Calculate the image points,  $\mathbf{X}$ , and model frame texture,  $\mathbf{g}_m$ .
  - 2: Sample the image to get  $\mathbf{g}_{im}$
  - 3: Normalise with respect to brightness and contrast using  $\mathbf{g}_s = T_{\mathbf{u}}^{-1}(\mathbf{g}_{im})$ .
  - 4: Compute the residual,  $\mathbf{r} = \mathbf{g}_s - \mathbf{g}_m$ , and corresponding error,  $E_{sse} = |\mathbf{r}|^2$ .
  - 5: **repeat**
  - 6:   Predict the displacement from the true model parameters,  $\delta\mathbf{p} = \mathbf{R}\mathbf{r}(\mathbf{p})$ .
  - 7:   Set  $k = 1$ .
  - 8:   **repeat**
  - 9:     Compute the updated model parameters,  $\mathbf{p}' = \mathbf{p} - k \cdot \delta\mathbf{p}$ . STATE Calculate the new points,  $\mathbf{X}'$ , and model frame texture,  $\mathbf{g}'_m$ .
  - 10:    Sample the image at the new points to get  $\mathbf{g}'_{im}$
  - 11:    Normalise with respect to brightness and contrast using  $\mathbf{g}'_s = T_{\mathbf{u}'}^{-1}(\mathbf{g}'_{im})$ .
  - 12:    Calculate a new residual vector,  $\mathbf{r}' = \mathbf{g}'_s - \mathbf{g}'_m$ , and corresponding error,  $E'_{sse} = |\mathbf{r}'|^2$ .
  - 13:    Set  $k = k/2$
  - 14:   **until**  $E'_{sse} < E_{sse}$
  - 15:   Set  $\mathbf{p} = \mathbf{p}'$ ,  $\mathbf{r} = \mathbf{r}'$  and  $E_{sse} = E'_{sse}$ .
  - 16: **until** converged ( $E_{sse} <$  threshold) or maximum number of iterations exceeded
- 

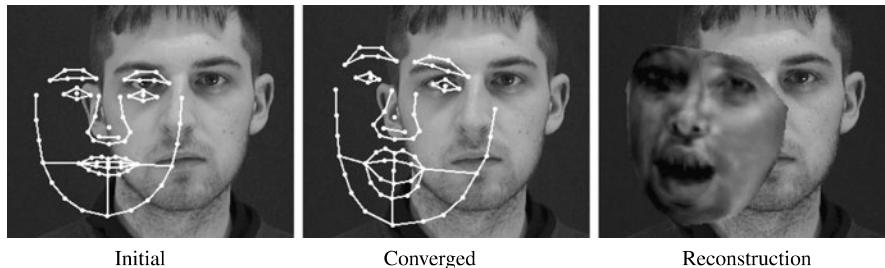
use a multi-resolution search algorithm in which we start at a coarse resolution and iterate to convergence at each level before projecting the current solution to the next level of the model [33]. This is more efficient and can converge to the correct solution from further away than search at a single resolution. Computationally, the complexity of the AAM at a given level is  $O(n_{modes} \cdot n_{pixels})$  since each iteration samples  $n_{pixels}$  points from the image then multiplies by a  $n_{modes} \times n_{pixel}$  matrix.

### 5.3.5 Examples of AAM Search

When using an AAM to localise a face in a previously unseen image, the algorithm typically requires fewer than 20 iterations to converge to a faithful reproduction of the face (Fig. 5.11). Like the ASM, however, the AAM is prone to local minima if started too far from the true solution (Fig. 5.12).



**Fig. 5.11** Search using the Active Appearance Model on faces not in the training set, showing evolution of the shape and the final image reconstruction. Initial iterations are performed using a low resolution model and resolution increases as the search progresses



**Fig. 5.12** Example of AAM search failure where the initialisation was too far from true position. The model has matched the eye and eyebrow to the wrong side of the face, and attempted to explain the dark background by shading one side of the reconstructed face

### 5.3.6 Alternative Strategies

Following the Active Appearance Model, a variety of related approaches to matching models of shape and texture have been suggested. Here, we summarise some of the key contributions.

### 5.3.6.1 Shape AAM

Though combining shape and appearance parameters has its uses in capturing correlations, treating the parameters separately can have computational benefits. Consider the case where we use the residuals to update only the pose,  $\mathbf{t}$ , and shape model parameters,  $\mathbf{b}_s$ , such that

$$\delta\mathbf{t} = \mathbf{R}_t \mathbf{r} \quad \text{and} \quad \delta\mathbf{b}_s = \mathbf{R}_s \mathbf{r}, \quad (5.30)$$

where the model texture,  $\mathbf{g}_m$ , is now simply the projection of the normalised sample,  $\mathbf{g}_s$ , onto the texture subspace (since shape and texture are treated independently). In this case,

$$\delta\mathbf{b}_s = \mathbf{R}_s(\mathbf{g}_s - (\bar{\mathbf{g}} + \mathbf{P}_g \mathbf{P}_g^T(\mathbf{g}_s - \bar{\mathbf{g}}))) \quad (5.31)$$

$$= \mathbf{R}_s((\mathbf{I} - \mathbf{P}_g \mathbf{P}_g^T)(\mathbf{g}_s - \bar{\mathbf{g}})) \quad (5.32)$$

$$= \mathbf{R}'_s \mathbf{g}_s - \mathbf{g}_0 \quad (5.33)$$

where  $\mathbf{R}'_s = \mathbf{R}_s(\mathbf{I} - \mathbf{P}_g \mathbf{P}_g^T)$  and  $\mathbf{g}_0 = \mathbf{R}_s(\mathbf{I} - \mathbf{P}_g \mathbf{P}_g^T)\bar{\mathbf{g}}$  can be precomputed such that the texture model is required only to compute the texture error for the purposes of detecting convergence. Using a fixed number of iterations or changes in the shape parameters, however, dispenses with the texture model altogether and results in a much faster (though less accurate) algorithm. If required, a combined model of shape and texture can be used to apply *post hoc* constraints to the relative shape and texture parameter vectors by projecting them into the combined appearance space. This approach, known as the ‘Shape AAM’ [13], is closely related to the ‘Active Blob’ method [52] that uses an elastic deformation model rather than a statistical model of shape.

### 5.3.6.2 Compositional Approach

As noted earlier (Sect. 5.3.3), pose and texture transformation parameters should be updated via composition (rather than addition) and it can be shown that there are benefits from updating shape parameters in the same way [42]. If we consider (5.4) as a parameterised transformation of the mean shape,  $\mathbf{x} = U_{\mathbf{b}}(\bar{\mathbf{x}})$ , then we need to find parameters,  $\mathbf{b}'$ , such that  $U_{\mathbf{b}'}(\bar{\mathbf{x}}) = U_{\mathbf{b}}(U_{\delta\mathbf{b}}(\bar{\mathbf{x}}))$ , for example by approximating the transformation with a thin-plate spline (Algorithm 5.9). Using the *inverse compositional* image alignment algorithm [1] improves efficiency further by specifying Jacobians and Hessians as functions of template images (rather than sampled images) such that they can be precomputed, thus saving computation at run-time. Also decoupling shape from texture for efficiency, the resulting inverse compositional AAM [42] has demonstrated model fitting at speeds of up to 200 frames per second.

---

**Algorithm 5.9:** Compositional AAM fitting with a Thin Plate Spline [6]

- 
- 1: Compute the thin plate spline,  $T_{\text{tps}}(\cdot)$ , that maps the points  $\bar{\mathbf{x}}$  to  $\mathbf{x}$
  - 2: Compute the modified mean points,  $\mathbf{x}_\delta = \bar{\mathbf{x}} + \mathbf{P}_s \delta$
  - 3: Apply the transformation,  $\mathbf{x}' = T_{\text{tps}}(\mathbf{x}_\delta)$
  - 4: Find the shape parameters which best match,  $\mathbf{b}'_s = \mathbf{P}_s^T (\mathbf{x}' - \bar{\mathbf{x}})$
- 

### 5.3.7 Further Reading

Since their introduction, Active Appearance Models have spawned many variants [26] and also demonstrated considerable success in medical image analysis (for which, software is publicly available [55]). In addition to the two variants already described (Sect. 5.3.6), other modifications include methods for expressing the update matrix,  $\mathbf{R}$ , as a function of the current residual for improved convergence [2] and sequential implementations that tune the training data to match the expected error distribution [49].

Predicting parameter updates via nonlinear regression has also been proposed, where boosting a number of weak regressors is currently popular [48, 63]. Using gradient descent-based algorithms to minimise an error metric learned from training data has also shown promise [39], as has selecting updates via a pairwise comparison of two potential candidates [62].

## 5.4 Conclusions

In this chapter, we have described powerful statistical models of the shape and texture of faces that are capable of synthesising a wide range of convincing face images. Algorithms such as the Active Shape Model (ASM) and Active Appearance Model (AAM) rapidly fit these appearance models to unseen image data such that the parameters capture the underlying properties of the face, isolating those sources of variation that are essential to face recognition (that is, identity) from those that are not (e.g., expression).

One weakness of both the ASM and AAM (and their variations) is that they are local optimisation techniques and tend to fall into local minima if initialisation is poor. Where independent estimates of feature point positions are available (e.g., from an eye tracker) these can be incorporated into the matching schemes and lead to more reliable matching [9].

These approaches also rely on an annotated corpus of training data and therefore can only deal effectively with certain types of variation in appearance. For example, person-specific variation that cannot be corresponded (e.g., wrinkles on the forehead or the appearance of moles) tends to get blurred out by the averaging process inherent in the modelling. This suggests that these methods may be improved by adding further layers of information to the model in order to represent individual differences which are poorly represented as a result of pooling in the current models.

Open questions (some of which are currently under investigation) include:

- How do we obtain accurate correspondences across the training set?
- What is the optimal choice of model size and number of model modes?
- How should image structure be represented?
- What is the best method of matching the model to the image?
- How do we avoid local minima in the error surface?

**Acknowledgements** The authors would like to thank their numerous colleagues who have contributed to the research summarised in this chapter, including C. Beeston, F. Bettinger, D. Cooper, D. Cristinacce, G. Edwards, A. Hill, J. Graham, H. Kang, P. Kittipanya-ngam and M. Roberts.

## References

1. Baker, S., Matthews, I.: Lucas–Kanade 20 years on: A unifying framework. Part I: The quantity approximated, the warp update rule and the gradient descent approximation. *Int. J. Comput. Vis.* (2004)
2. Batur, A.U., Hayes, M.H.: Adaptive active appearance models. *IEEE Trans. Med. Imaging* **14**(11), 1707–1721 (2005)
3. Belkin, M., Nigoyi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.* **15**, 1373–1396 (2003)
4. Benson, P.J., Perrett, D.I.: Synthesizing continuous-tone caricatures. *Image Vis. Comput.* **9**, 123–129 (1991)
5. Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. *IEEE Trans. Pattern Anal. Mach. Intell.* (2003)
6. Bookstein, F.L.: Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.* **11**(6), 567–585 (1989)
7. Cootes, T.F., Kittipanya-ngam, P.: Comparing variations on the active appearance model algorithm. In: 13th British Machine Vision Conf., vol. 2, pp. 837–846, September 2002
8. Cootes, T., Taylor, C.J.: A mixture model for representing shape variation. *Image Vis. Comput.* **17**(8), 567–574 (1999)
9. Cootes, T.F., Taylor, C.J.: Constrained active appearance models. In: 8th Int'l Conf. on Comp. Vis., vol. 1, pp. 748–754, July 2001. IEEE Computer Society Press, Los Alamitos (2001)
10. Cootes, T.F., Taylor, C.J.: On representing edge structure for model matching. *Comput. Vis. Pattern Recognit.* **1**, 1114–1119 (2001)
11. Cootes, T.F., Taylor, C.J., Cooper, D., Graham, J.: Active shape models—their training and application. *Comput. Vis. Image Underst.* **61**(1), 38–59 (1995)
12. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. In: Burkhardt, H., Neumann, B. (eds.) 5th European Conf. on Comp. Vis., vol. 2, pp. 484–498. Springer, Berlin (1998)
13. Cootes, T.F., Edwards, G.J., Taylor, C.J.: A comparative evaluation of active appearance model algorithms. In: British Machine Vision Conf., vol. 2, pp. 680–689, September 1998
14. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(6), 681–685 (2001)
15. Cootes, T.F., Wheeler, G.V., Walker, K.N., Taylor, C.J.: View-based active appearance models. *Image Vis. Comput.* **20**, 657–664 (2002)
16. Costen, N., Cootes, T.F., Taylor, C.J.: Compensating for ensemble-specificity effects when building facial models. *Image Vis. Comput.* **20**, 673–682 (2002)
17. Crandall, D., Felzenszwalb, P., Huttenlocher, D.: Spatial priors for part-based recognition using statistical models. In: Proc. IEEE Conf. on Comp. Vis. and Patt. Recog., vol. 1 (2005)

18. Craw, I., Cameron, P.: Parameterising images for recognition and reconstruction. In: 2nd British Machine Vision Conf., pp. 367–370. Springer, London (1991)
19. Craw, I., Cameron, P.: Face recognition by computer. In: Hogg, D., Boyle, R. (eds.) 3rd British Machine Vision Conf., pp. 489–507. Springer, London (1992)
20. Cristinacce, D., Cootes, T.: Facial feature detection using AdaBoost with shape constraints. In: Proc. British Machine Vision Conf. (2003)
21. Cristinacce, D., Cootes, T.F.: Automatic feature localisation with constrained local models. *Pattern Recognit.* **41**, 3054–3067 (2008)
22. Donner, R., Reitner, M., Langs, G., Peloschek, P., Bischof, H.: Fast active appearance model search using canonical correlation analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(10), 1690–1694 (2006)
23. Dryden, I., Mardia, K.V.: *The Statistical Analysis of Shape*. Wiley, London (1998)
24. Edwards, G.J., Lanitis, A., Taylor, C.J., Cootes, T.F.: Statistical models of face images—improving specificity. *Image Vis. Comput.* **16**(3), 203–211 (1998)
25. Felzenszwalb, P., Huttenlocher, D.: Pictorial structures for object recognition. *Int. J. Comput. Vis.* **61**(1), 55–79 (2005)
26. Gao, X., Su, Y., Li, X., Tao, D.: A review of active appearance models. *IEEE Trans. Syst. Man Cybern., Part C, Appl. Rev.* **40**(2), 145–158 (2010)
27. Goodall, C.: Procrustes methods in the statistical analysis of shape. *J. R. Stat. Soc. B* **53**(2), 285–339 (1991)
28. Gu, L., Kanade, T.: A generative shape regularization model for robust face alignment. In: Proc. European Conf. on Computer Vision (2008)
29. Gu, L., Xing, E.P., Kanade, T.: Learning GMRF structures for spatial priors. In: Proc. IEEE Conf. on Comp. Vis. and Patt. Recog. (2007)
30. Hill, A., Cootes, T.F., Taylor, C.J.: Active shape models and the shape approximation problem. *Image Vis. Comput.* **14**, 601–607 (1996)
31. Hou, X., Li, S., Zhang, H., Cheng, Q.: Direct appearance models. In: Computer Vision and Pattern Recognition Conf. 2001, vol. 1, pp. 828–833 (2001)
32. Huang, Y., Liu, Q., Metaxas, D.N.: A component based deformable model for generalized face alignment. In: Proc. IEEE Int'l Conf. on Comp. Vis., pp. 1–8 (2007)
33. Jones, M.J., Poggio, T.: Multidimensional morphable models: A framework for representing and matching object classes. *Int. J. Comput. Vis.* **2**(29), 107–131 (1998)
34. Kirby, M., Sirovich, L.: Application of the Karhunen–Loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(1), 103–108 (1990)
35. la Torre, F.D., Collet, A., Quero, M., Cohn, J.F., Kanade, T.: Filtered component analysis to increase robustness to local minima in appearance models. In: Proc. IEEE Conf. on Comp. Vis. and Patt. Recog. (2007)
36. Lee, H.-S., Kim, D.: Tensor-based AAM with continuous variation estimation: Application to variation-robust face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(6), 1102–1116 (2009)
37. Liang, L., Wen, F., Xu, Y.-Q., Tang, X., Shum, H.-Y.: Accurate face alignment using shape constrained Markov network. In: Proc. IEEE Conf. on Comp. Vis. and Patt. Recog. (2006)
38. Liang, L., Xiao, R., Wen, F., Sun, J.: Face alignment via component-based discriminative search. In: Proc. European Conf. on Computer Vision (2008)
39. Liu, X.: Discriminative face alignment. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(11), 1941–1954 (2009)
40. Lu, H.-M., Fainman, Y., Hecht-Nelson, R.: Image manifolds. In: Proc. SPIE Symposium on Electronic Imaging: Science and Technology (1998)
41. Lucey, S., Wang, Y., Saragih, J., Cohn, J.F.: Non-rigid face tracking with enforced convexity and local appearance consistency constraint. *Image Vis. Comput.* **28**(5), 781–789 (2010)
42. Matthews, I., Baker, S.: Active appearance models revisited. *Int. J. Comput. Vis.* **26**(10), 135–164 (2004)
43. Matthews, I., Xiao, J., Baker, S.: 2D vs. 3D deformable face models: Representational power, construction, and real-time fitting. *Int. J. Comput. Vis.* **75**(1), 93–113 (2007)

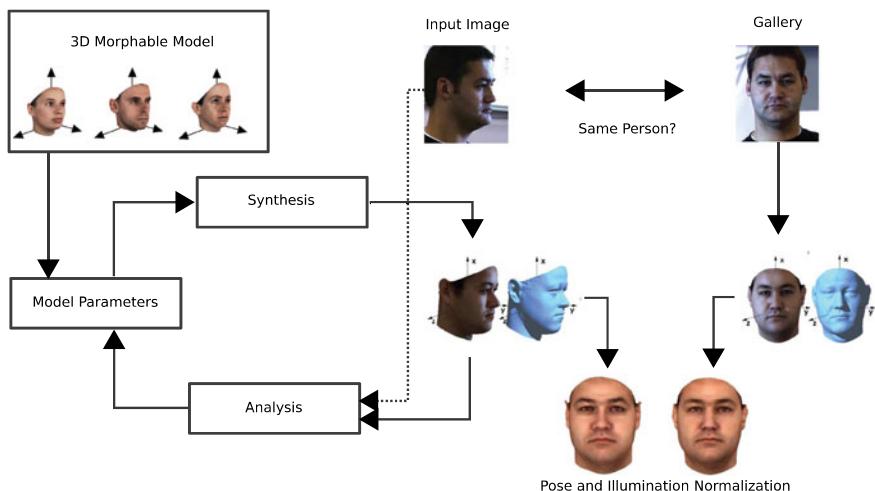
44. Milborrow, S., Nicolls, F.: Locating facial features with an extended active shape model. In: Proc. European Conf. on Computer Vision (2008)
45. Paquet, U.: Convexity and Bayesian constrained local models. In: Proc. IEEE Conf. on Comp. Vis. and Patt. Recog. (2009)
46. Romdhani, S., Gong, S., Psarrou, A.: A multi-view non-linear active shape model using kernel PCA. In: 10th British Machine Vision Conf., vol. 2, pp. 483–492, September 1999
47. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* (2000)
48. Saragih, J., Goecke, R.: A nonlinear discriminative approach to AAM fitting. In: Proc. IEEE Int'l Conf. on Comp. Vis. (2007)
49. Saragih, J., Goecke, R.: Learning AAM fitting through simulation. *Pattern Recognit.* **42**(11), 2628–2636 (2009)
50. Saragih, J.M., Lucey, S., Cohn, J.F.: Deformable model fitting with a mixture of local experts. In: Proc. IEEE Int'l Conf. on Comp. Vis. (2009)
51. Saragih, J.M., Lucey, S., Cohn, J.F.: Face alignment through subspace constrained mean-shifts. In: Proc. IEEE Int'l Conf. on Comp. Vis. (2009)
52. Sclaroff, S., Isidoro, J.: Active blobs. In: 6th Int'l Conf. on Comp. Vis., pp. 1146–1153 (1998)
53. Scott, I.M., Cootes, T.F., Taylor, C.J.: Improving appearance model matching using local image structure. In: Information Processing in Medical Imaging, pp. 258–269. Springer, Berlin (2003)
54. Sozou, P.D., Cootes, T.F., Taylor, C.J., Mauro, E.C.D.: Non-linear generalization of point distribution models using polynomial regression. *Image Vis. Comput.* **13**(5), 451–457 (1995)
55. Stegmann, M.B., Ersbøll, B.K., Larsen, R.: FAME—a flexible appearance modelling environment. *IEEE Trans. Med. Imaging* **22**(10), 1319–1331 (2003)
56. Stegmann, M.B., Larsen, R.: Multi-band modelling of appearance. *Image Vis. Comput.* **21**(1), 66–67 (2003)
57. Tenenbaum, J.B., Silva, V.D., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* **290**, 2319–2323 (2000)
58. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–86 (1991)
59. van Ginneken, B., Frangi, A.F., Stall, J.J., ter Haar Romeny, B.M.: Active shape model segmentation with optimal features. *IEEE Trans. Med. Imaging* **21**, 924–933 (2002)
60. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear analysis of image ensembles: TensorFaces. In: Proc. European Conf. on Computer Vision (2002)
61. Vetter, T.: Learning novel views to a single face image. In: 2nd Int'l Conf. on Automatic Face and Gesture Recognition 1996, pp. 22–27, October 1996
62. Wu, H., Liu, X., Doretto, G.: Face alignment via boosted ranking model. In: Proc. IEEE Conf. on Comp. Vis. and Patt. Recog. (2008)
63. Zhou, S.K., Comaniciu, D.: Shape regression machine. In: Proc. Int'l Conf. on Information Processing in Medical Imaging (2007)



# Chapter 6

## Morphable Models of Faces

Reinhard Knothe, Brian Amberg, Sami Romdhani, Volker Blanz,  
and Thomas Vetter



---

R. Knothe (✉) · B. Amberg · S. Romdhani · T. Vetter

Department of Mathematics and Computer Science, University of Basel, Bernoullistrasse 16,  
4056 Basel, Switzerland

e-mail: [reinhard.knothe@unibas.ch](mailto:reinhard.knothe@unibas.ch)

B. Amberg

e-mail: [brian.amberg@unibas.ch](mailto:brian.amberg@unibas.ch)

S. Romdhani

e-mail: [sami.romdhani@unibas.ch](mailto:sami.romdhani@unibas.ch)

T. Vetter

e-mail: [thomas.vetter@unibas.ch](mailto:thomas.vetter@unibas.ch)

V. Blanz

Universität Siegen, Hölderlinstrasse 3, 57068 Siegen, Germany

e-mail: [blanz@mpi-sb.mpg.de](mailto:blanz@mpi-sb.mpg.de)

## 6.1 Introduction

Our approach is based on an *analysis by synthesis* framework. In this framework, an input image is analyzed by searching for the parameters of a generative model such that the generated image is as similar as possible to the input image. The parameters are then used for high-level tasks such as identification.

To be applicable to all input face images, a good model must be able to generate all possible face images. Face images vary widely with respect to the imaging conditions (illumination and the position of the camera relative to the face, called pose) and with respect to the identity and the expression of the face. A generative model must not only allow for these variations but must also separate the sources of variation such that e.g. the identity can be determined regardless of pose, illumination or expression.

In this chapter, we present the Morphable Model, a three-dimensional (3D) representation that enables the accurate modeling of any illumination and pose as well as the separation of these variations from the rest (identity and expression). The Morphable Model is a generative model consisting of a linear 3D shape and appearance model plus an imaging model, which maps the 3D surface onto an image. The 3D shape and appearance are modeled by taking linear combinations of a training set of example faces. We show that linear combinations yield a realistic face only if the set of example faces is in correspondence. A good generative model should accurately distinguish faces from nonfaces. This is encoded in the probability distribution over the model parameters, which assigns a high probability to faces and a low probability to nonfaces. The distribution is learned together with the shape and appearance space from the training data.

Based on these principles, we detail the construction of a 3D Morphable Face Model in Sect. 6.2. The main step of model construction is to build the correspondences of a set of 3D face scans. Such models have become a well-established technology which is able to perform various tasks, not only face recognition, but also face image analysis [6] (e.g., estimating the 3D shape from a single photograph), expression transfer from one photograph to another [10, 46], animation of faces [10], training of feature detectors [22, 24], and stimuli generation for psychological experiments [29] to name a few. The power of these models comes at the cost of an expensive and tedious construction process, which has led the scientific community often to focus on more easily constructed but less powerful models. Recently, a complete 3D Morphable Face Model built from 3D face scans, the *Basel Face Model* (BFM), was made available to the public ([faces.cs.unibas.ch](http://faces.cs.unibas.ch)) [34]. An alternative approach to construct a 3D Morphable Model is to generate the model directly from a video sequence [12] using nonrigid structure from motion. While this requires far less manual intervention, it also results in a less detailed and inaccurate model.

With a good generative face model, we are half the way to a face recognition system. The remaining part of the system is the face analysis algorithm (the *fitting algorithm*). The fitting algorithm finds the parameters of the model that generate an image which is as close as possible to the input image. In this chapter, we focus on fitting the model to a single image. We detail two fitting algorithms in Sect. 6.4.

Based on these two fitting algorithms, identification results are presented in Sect. 6.5 for face images varying in illumination and pose, as well as for 3D face scans. The fitting methods presented here are energy minimization methods, a different class of fitting methods are regression based, and try to learn a correspondence between appearance and model coefficients. For 2D Models, [20] proposed to learn a linear regression mapping the residual between a current estimate and the final fit, and [27] proposed to use a support vector regression on Haar features of the image to directly predict 6 coefficients of a 2D mouth model from 12 Haar features of mouth images.

### ***6.1.1 Three-Dimensional Representation***

Each individual face can generate a variety of images when seen from different viewpoints, under different illumination and with different expressions. This huge diversity of face images makes their analysis difficult. In addition to the general differences between individual faces, the appearance variations in images of a single faces can be separated into the following four sources.

- Pose changes can result in dramatic changes in images. Due to self-occlusions different parts of the object become visible or invisible. Additionally, the parts seen in two views change their spatial configuration relative to each other.
- Illumination changes influence the appearance of a face even if the pose of the face is fixed. The distribution of light sources around a face changes the brightness distribution in the image, the locations of attached shadows, and specular reflections. Additionally, cast shadows can generate prominent contours in facial images.
- Facial expressions are another source of variations in images. Only a few facial landmarks that are directly coupled with the bony structure of the skull, such as the corners of the eye or the position of the earlobes, are constant in a face. Most other features can change their spatial configuration or position via articulation of the jaw or muscle action (e.g., moving eyebrows, lips, or cheeks).
- On a longer timescale faces change because of aging, change of hairstyle, and the use of makeup or accessories.

The isolation and explicit description of these sources of variations must be the ultimate goal of a face analysis system. For example, it is desirable that the parameters that code the identity of a person are not perturbed by a modification of pose. In an analysis by synthesis framework, this implies that the face model must account for each of these variations independently by explicit parameters.

We need a generative model which is a concise description of the observed phenomena. The image of a face is generated according to the laws of physics which describe the interaction of light with the face surface and the camera. The parameters for pose and illumination can therefore be described most concisely when modeling the face as a 3D surface. A concise description of the variability of human faces on the other hand can not be derived from physics. We therefore describe the variations in 3D shape and albedo of human faces with parameters learned from examples.

### 6.1.2 Correspondence-Based Representation

Early face recognition techniques used a purely *appearance-based representation* for face analysis. In an appearance based representation such as *eigenfaces* [40, 43] and their generalized version from [33], it is assumed that face images behave like a vector space, that is, the result of linearly combining face images yields a new face image. These techniques have been demonstrated to be successful when applied to images of a set of very different objects, or when the viewpoint and lighting conditions are close to constant and the image resolution is relatively low. These limitations come from the wrong assumption that *face images* form a vector space. We illustrate this in Fig. 6.1 by showing that already the mean of two face images has double edges and is no longer a face. Even though *face images* do not form a vector space, *faces* do form a vector space when using the correct representation. The artifacts visible in the averaged face in Fig. 6.1 come from the fact that pixels from different positions on the two faces, have been combined to generate the new pixel value. In an image, face shape and appearance are mixed. If one separates the face shape and the face appearance by establishing *correspondence* between the input images, then the faces can be linearly combined, see Fig. 6.1 lower right.

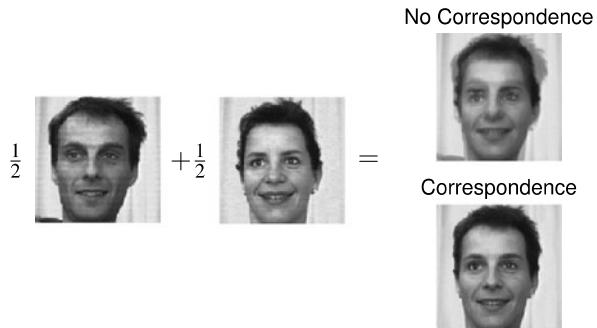
Separating face shape and face appearance means that one chooses an underlying parametrization of a face (the face domain), independently from the images. The shape of a face in an image is then expressed as the correspondence between the face domain and the image, and the appearance of a face by the image mapped back into the face domain. Within the face domain, the color and shape of different faces can be linearly combined to yield new faces. In addition, one also separates the shape into a local deformation and a camera model, which positions the face inside the target image. When discretizing the face domain, we can express face shape and appearance by a vectors of shape displacements and color. The combination of the shape and appearance vector spaces is called face subspace.

A separation of shape and appearance—also called an *object center representation*—has been proposed by multiple authors [5, 15, 23, 28, 45]; for a review see Beymer and Poggio [4]. It should be noted that some of these approaches do not use the correspondence between points which are actually corresponding on the underlying faces: The methods using 2D face models, for example, Lanitis et al. [28], often put into correspondence the 2D occluding contour, which corresponds to different positions on the face depending on the pose. In contrast, our approach uses the 3D shape of faces as the face domain and establishes the true correspondences between the underlying faces.

### 6.1.3 Face Statistics

In the previous section, we explained that correspondences enable the generation of new faces as a linear combination of training faces. However, the coefficients of the linear combination do not have a uniform distribution. This distribution is learned

**Fig. 6.1** Computing the average of two face images using different image representations. No correspondence information is used (*top right*) and using correspondence (*bottom right*)



from example faces using the currently widely accepted assumption that the face subspace is Gaussian. Under this assumption, PCA is used to learn a probability model of faces that is used as prior probability at the analysis step (see Sect. 6.4.1). More details about the face statistics of our model are given in Sects. 6.2.3 and 6.2.4.

## 6.2 3D Morphable Model Construction

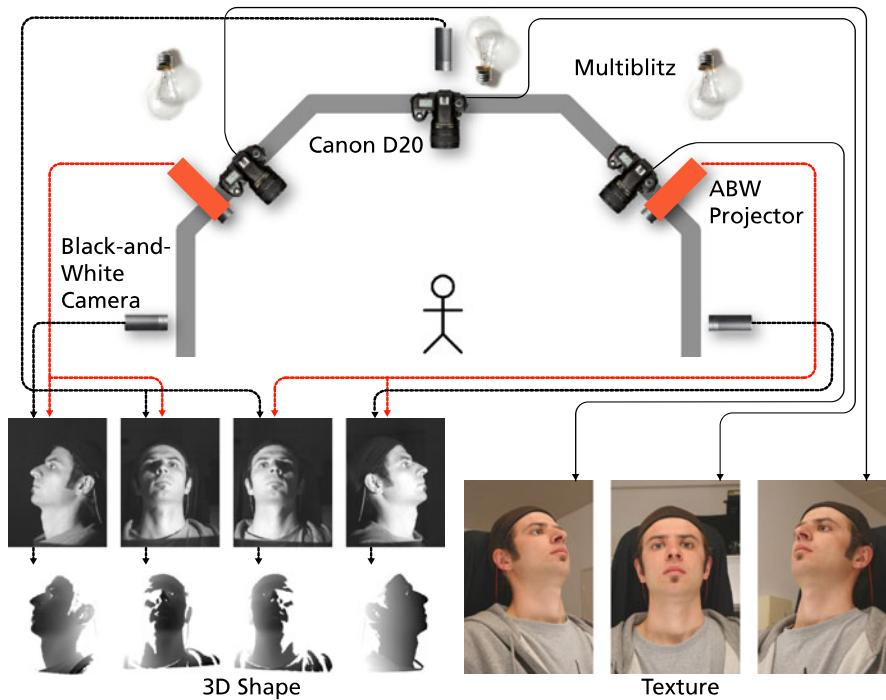
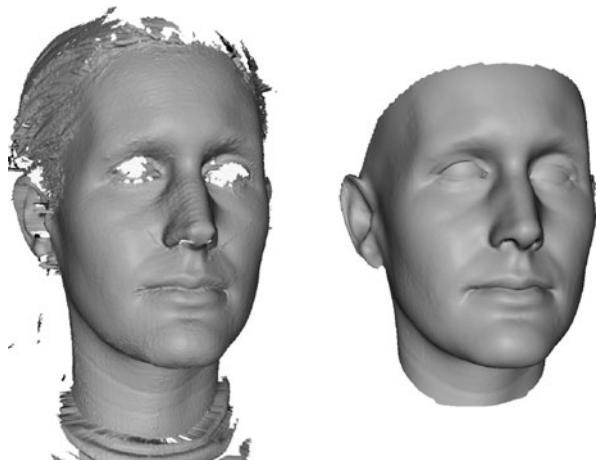
The construction of a 3D Morphable Model requires a set of example 3D face scans with a large variety. The results presented in this section were obtained with a Morphable Model constructed with 200 scans (100 female and 100 male), most of them Europeans. This Morphable Model has been made publicly available (Basel Face Model: [faces.cs.unibas.ch](http://faces.cs.unibas.ch) [34]) and can be freely used for noncommercial purposes. The constructions are performed in three steps: First, the scans are preprocessed. This semiautomatic step aims to remove the scanning artifacts and to select the part of the head that is to be modeled (from one ear to the other and from the neck to the forehead). In the second step, the correspondences are computed between each of the scans and a reference face mesh (see Fig. 6.2 for an example). The registered faces scans are then aligned using Generalized Procrustes Analysis such that they do not contain a global rigid transformation. Then a principal component analysis is performed to estimate the statistics of the 3D shape and color of the faces.

### 6.2.1 3D Face Scanning

For 3D face scanning it is important to have not only high accuracy but also a short acquisition time, such that the scans are not disrupted by involuntary motions, and the scanning of facial expressions is possible. We decided to use a coded light system (Fig. 6.3) because of its high accuracy and short acquisition time ( $\sim 1$  s) compared to laser scanners ( $\sim 15$  s as used in [6]).

The system captures the face shape from ear to ear (Fig. 6.2, left) and takes three color photographs. The 3D shape of the eyes and hair cannot be captured with our system, due to their reflection properties.

**Fig. 6.2** The registration of the original scan (*left*) establishes a common parametrization and fills in missing data (*right*)



**Fig. 6.3** 3D face scanning device developed by ABW-3D. The system consists of two structured light projectors, three gray level cameras for the shape, three 8 mega pixel SLR cameras and three studio flash lights

### 6.2.2 Registration

To establish correspondence, there exist two different approaches: mesh-based algorithms and algorithms modeling the continuous surface using variational techniques. Variational methods are mostly used in medical image analysis (e.g., [17, 25, 30]) where the input is typically already a voxelized volume. For 3D face surfaces, mesh-based algorithms are mostly used (e.g., [1, 3]). Here, we use a nonrigid ICP method similar to [1], that is applied in the 3D domain on triangulated meshes. It progressively deforms a template towards the measured surface. The correlated correspondence algorithm [3] is a very different approach to range scan registrations that is applicable to both faces and bodies.

The nonrigid ICP algorithm works as follows: First, for each vertex of the template a corresponding target point in the scanned surface is determined. This is done by searching for the point in the scan which is closest to the vertex of the deformed template, has a compatible normal and does not lie on the border of the scan. The template is then deformed such that the distance between the deformed template and the correspondence points is minimized subject to a regularization which prohibits strong deformations, bringing the template closer to the surface. We use a regularization which minimizes the second derivative of the deformation measured along the template surface. By starting with a strong regularization, we first recover the global deformations. The regularization is then lowered, allowing progressively more local deformations.

The steps of the algorithm are as follows:

---

**Algorithm 6.1:** Nonrigid ICP registration

---

**for**  $\theta \leftarrow \theta_1 > \dots > \theta_N$  **do**

**repeat**

- 1     Find candidate correspondences by searching for the closest point with a compatible normal for each model vertex.
- 2     Weight the correspondences by their distance using a robust estimator.
- 3     Find a deformation regularized by  $\theta$ , which minimizes the distance to the correspondence points.

**until** *median change in vertex positions < threshold*.

---

The template shape is created in a bootstrapping process, starting with a manually created head model with an optimized mesh using discrete conformal mappings [26]. It is first registered to the target scans, then the average over all registrations is used as a new template. This process is iterated a few times involving further manual corrections to achieve a good template shape. The template defines the parametrization of the model and it is used to fill in holes in the measurement. Whenever no correspondences are found, the deformation is extended smoothly along the

template surface by the regularization. This fills in unknown regions with the deformed template shape.

The regularization uses a discrete approximation of the second derivative of the deformation field, which is calculated by finite differencing between groups of three neighboring vertices. The cost function in [1] uses first order finite differences, and can be adapted in a straightforward way to second order finite differences.

The scanner produces four partially overlapping measurements (shells) of the target surface (Fig. 6.3), which have to be blended. This blending is done during registration by determining the closest point as a weighted average of the closest points of all shells. The weighting is performed with a reliability value computed from the distance between the scan border and the angle between surface normal and camera direction.

The registration method so far is purely shape based, but we have included additional cues from our scanning system. Some face features like the outline of the lips and the eyebrows are purely texture based, and do not have corresponding shape variations. To align these features in the model, we semi-automatically label the outlines of the lips, eyes and eyebrows in the texture photographs, and constrain the corresponding vertices of the deformed template to lie in the extrusion surfaces defined by the back-projection of these lines. Additionally, as the shape of the ears is not correctly measured by the scanner, we mark the outline of the ears in the images to get at least the overall shape of the ears right.

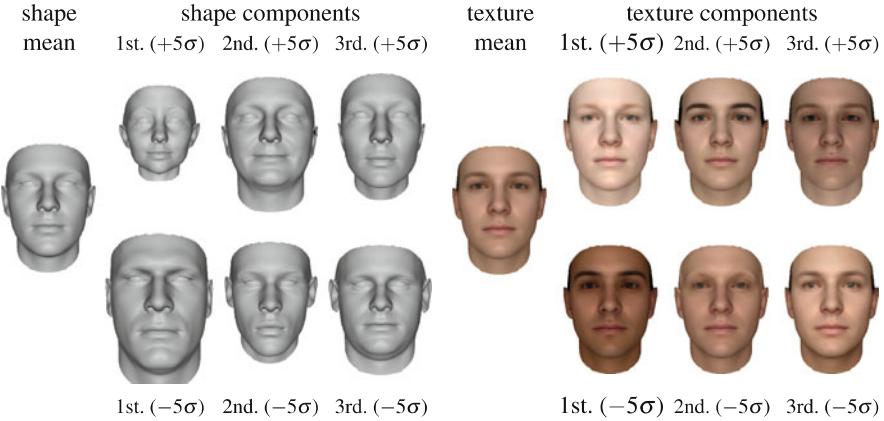
To initialize the registration, some landmarks are used. The weighting of the landmark term is reduced to zero during the optimization, as these points cannot be marked as accurately as the line landmarks. In a preprocessing step, the scanned data is smoothed using mean curvature flow [16] on the depth images.

Since the shape of the eyeballs is not correctly measured by the scanner, we replace them by spheres fit to the vertices of the eyes. This is done after the registration in a post-processing step.

The projection of each pixel of the high resolution texture photos onto the geometry is calculated, resulting in three overlapping texture maps. These are blended based on the distance from the visible boundaries and the orientation of the normal relative to the viewing direction. Hair is manually removed from the resulting texture, and the missing data is filled in by a diffusion process.

### 6.2.3 PCA Subspace

Section 6.1.2 introduced the idea of a face subspace, wherein all faces are constructable by a generative model lie. We now detail the construction of the face subspace for a 3D Morphable Model. The face subspace is constructed by putting a set of  $M$  example 3D face scans into correspondence with a reference face. This introduces a consistent labeling of all  $N_v$  3D vertices across all the scans: each registered face is represented by a triangular mesh with  $N_v = 53\,490$  vertices. Each vertex  $j$  consists of a 3D point  $(x_j, y_j, z_j)^T \in \mathbb{R}^3$  with an associated per vertex



**Fig. 6.4** The mean together with the first three principle components of the shape (*left*) and texture (*right*) PCA model. Shown is the mean shape resp. texture plus/minus five standard deviations  $\sigma$

color  $(r_j, g_j, b_j)^T \in \mathbb{R}^3$ . Due to the correspondence, the mesh topology is the same for each face. These 3D meshes are a discretization of the continuous underlying face domain. Each shape or texture can be represented as a  $3 \times N_v$  matrix

$$\mathbf{S} = \begin{pmatrix} x_1 & x_2 & \cdots & x_{N_v} \\ y_1 & y_2 & \cdots & y_{N_v} \\ z_1 & z_2 & \cdots & z_{N_v} \end{pmatrix}, \quad \mathbf{T} = \begin{pmatrix} r_1 & r_2 & \cdots & r_{N_v} \\ g_1 & g_2 & \cdots & g_{N_v} \\ b_1 & b_2 & \cdots & b_{N_v} \end{pmatrix}. \quad (6.1)$$

We can now take linear combinations of the  $M$  example faces to produce new faces corresponding to new individuals.

$$\mathbf{S} = \sum_{i=1}^M \alpha_i \cdot \mathbf{S}_i, \quad \mathbf{T} = \sum_{i=1}^M \beta_i \cdot \mathbf{T}_i. \quad (6.2)$$

While these linear combinations do contain all new faces, they also contain non faces. All convex combinations of faces ( $\sum \alpha_i = 1, \alpha_i \in [0, 1]$ ) are again faces, and also vectors close to the convex area spanned by the examples are faces, but points far away from the convex area correspond to shapes which are very unlikely faces. When for example setting all but one coefficient to 100, we get a face which is 100 times as big as the example face. It is very unlikely that we will encounter such a face in the real world. This argument shows, that each coefficient vector needs an assigned probability of describing a face. We model this probability by a Gaussian distribution with a block diagonal matrix, which assumes that shape and texture are decorrelated. Assuming a Gaussian allows us to approximate the face subspace with a smaller set of orthogonal basis vectors, which are computed with Principal Component Analysis (PCA) from the training examples.

Principal component analysis (PCA) is a statistical tool that transforms the space such that the covariance matrix is diagonal (i.e., it decorrelates the data). We de-

scribe the application of PCA to shapes; its application to textures is straightforward. The resulting model is shown in Fig. 6.4. After subtracting their average,  $\bar{\mathbf{S}}$ , the exemplars are arranged in a data matrix  $\mathbf{A}$  and the eigenvectors of its covariance matrix  $\mathbf{C}$  are computed using the singular value decomposition [37] of  $\mathbf{A}$ .

$$\begin{aligned}\bar{\mathbf{S}} &= \frac{1}{M} \sum_{i=1}^M \mathbf{S}_i, \quad \mathbf{a}_i = \text{vec}(\mathbf{S}_i - \bar{\mathbf{S}}), \quad \mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_M) = \mathbf{U} \mathbf{W}^T, \\ \mathbf{C} &= \frac{1}{M} \mathbf{A} \mathbf{A}^T = \frac{1}{M} \mathbf{U} \mathbf{W}^2 \mathbf{U}^T.\end{aligned}\quad (6.3)$$

The component  $\text{vec}(\mathbf{S})$  vectorizes  $\mathbf{S}$  by stacking its columns. The  $M$  columns of the orthogonal matrix  $\mathbf{U}$  are the eigenvectors of the covariance matrix  $\mathbf{C}$ , and  $\sigma_i^2 = \frac{\lambda_i^2}{M}$  are its eigenvalues, where the  $\lambda_i$  are the elements of the diagonal matrix  $\mathbf{W}$ , arranged in decreasing order. Let us denote  $\mathbf{U}_{\cdot,i}$ , the column  $i$  of  $\mathbf{U}$ , and the principal component  $i$ , reshaped into a  $3 \times N_v$  matrix, by  $\mathbf{S}^i = \mathbf{U}_{\cdot,i}^{(3)}$ . The notation  $\mathbf{a}_{m \times 1}^{(n)}$  [31] folds the  $m \times 1$  vector  $\mathbf{a}$  into an  $n \times (m/n)$  matrix.

Now, instead of describing a novel shape and texture as a linear combination of examples, as in (6.2), we express them as a linear combination of  $N_S$  shape and  $N_T$  texture principal components.

$$\mathbf{S} = \bar{\mathbf{S}} + \sum_{i=1}^{N_S} \alpha_i \cdot \mathbf{S}^i, \quad \mathbf{T} = \bar{\mathbf{T}} + \sum_{i=1}^{N_T} \beta_i \cdot \mathbf{T}^i. \quad (6.4)$$

The advantage of this formulation is that the probabilities associated with a shape and texture are readily available.

$$p(\mathbf{S}) \propto e^{-\frac{1}{2} \sum_i \frac{\alpha_i^2}{\sigma_{S,i}^2}}, \quad p(\mathbf{T}) \propto e^{-\frac{1}{2} \sum_i \frac{\beta_i^2}{\sigma_{T,i}^2}}. \quad (6.5)$$

### 6.2.4 Regularized Morphable Model

The correspondence estimation, detailed in Sect. 6.2.2, may, for some scans, be wrong in some regions. In this section, we present a scheme aiming to improve the correspondence by regularizing it using statistics derived from scans that do not present correspondence errors. This is achieved by modifying the model construction: probabilistic PCA [42] is used instead of PCA, which regularizes the model by allowing the exemplars to be noisy.

#### 6.2.4.1 Probabilistic PCA

Instead of assuming a linear model for the shape, as in the previous section, we assume a linear Gaussian model

$$\text{vec} \mathbf{S} = \text{vec} \bar{\mathbf{S}} + \mathbf{C}_S \cdot \boldsymbol{\alpha} + \boldsymbol{\varepsilon} \quad (6.6)$$

where  $\mathbf{C}_S$ , whose columns are the regularized shape principal components, has dimensions  $3N_v \times N_S$ , and the shape coefficients  $\boldsymbol{\alpha}$  and the noise  $\boldsymbol{\varepsilon}$  have a Gaussian distribution with zero mean and covariance  $\mathbf{I}$  and  $\sigma^2\mathbf{I}$ , respectively.

Tipping and Bishop [42] use the EM algorithm [18] to iteratively estimate  $\mathbf{C}_S$  and the projection of the example vectors to the model,  $\mathbf{K} = [\boldsymbol{\alpha}_1 \ \boldsymbol{\alpha}_2 \ \dots \ \boldsymbol{\alpha}_M]$ . The algorithm starts with  $\mathbf{C}_S = \mathbf{A}$ ; and then at each iteration it computes a new estimate of the shape coefficients  $\mathbf{K}$  (expectation step, or *e-step*) and of the regularized principal components  $\mathbf{C}_S$  (maximization step, or *m-step*). The coefficients of the example shapes, the unobserved variables, are estimated at the *e-step*.

$$\mathbf{K} = \mathbf{B}^{-1} \mathbf{C}_S^T \mathbf{A} \quad \text{with } \mathbf{B} = \mathbf{C}_S^T \mathbf{C}_S + \sigma^2 \mathbf{I}. \quad (6.7)$$

This is the maximum a posteriori estimator of  $\mathbf{K}$ ; that is, the expected value of  $\mathbf{K}$  given the posterior distribution  $p(\boldsymbol{\alpha} | \mathbf{C}_S)$ . At the *m-step*, the model is estimated by computing the  $\mathbf{C}_S$ , which maximizes the likelihood of the data, given the current estimate of  $\mathbf{K}$  and  $\mathbf{B}$ .

$$\mathbf{C}_S = \mathbf{A} \cdot \mathbf{K}^T \cdot (\sigma^2 \cdot M \cdot \mathbf{B}^{-1} + \mathbf{K} \cdot \mathbf{K}^T)^{-1}. \quad (6.8)$$

These two steps are iterated in sequence until the algorithm is judged to have converged. In the original algorithm, the value of  $\sigma^2$  is also estimated at the m-step as

$$\sigma^2 = \frac{1}{3N_v \cdot M} \text{tr}(\mathbf{A}\mathbf{A}^T - \mathbf{C}_S \mathbf{K} \mathbf{A}^T) \quad (6.9)$$

but in our case, with  $M \ll 3N_v$ , this would yield an estimated value of zero. Therefore, we prefer to estimate  $\sigma^2$  by replacing  $\mathbf{A}$  and  $\mathbf{K}$  in (6.9) with a data matrix of test vectors (vectors not used in estimating  $\mathbf{C}_S$ ) and its corresponding coefficients matrix obtained via (6.7). If a test set is not available, we can still get an estimate of  $\sigma^2$  by cross validation.

### 6.2.5 Segmented Morphable Model

Our Morphable Model is derived from statistics computed on 200 example faces. As a result, the dimensions of the shape and texture spaces,  $N_S$  and  $N_T$ , are limited to 199. This might not be enough to account for the rich variations of individualities present in humankind. Naturally, one way to augment the dimension of the face subspace would be to use 3D scans of more persons but they are not available. Hence we resort to another scheme: We segment the face into four regions (nose, eyes, mouth, and the rest) and use a separate set of shape and texture coefficients to code them [6]. This method multiplies by four the dimensionality of the Morphable Model and results in an increased flexibility. However, this process must be taken with care, since a segmented model loses the correlation between the segments. More formally, segmenting is the same as assuming zero covariance between the

segments. The fitting results in Sect. 6.4 and the identification results in Sect. 6.5 are based on a segmented Morphable Model with  $N_S = N_T = 100$  for all segments. In the rest of the chapter, we denote the shape and texture parameters by  $\alpha$  and  $\beta$  when they can be used interchangeably for the global and the segmented parts of the model. When we want to distinguish them, we use, for the shape parameters,  $\alpha^g$  for the global model (full face) and  $\alpha^{s_1}$  to  $\alpha^{s_4}$  for the segmented parts (the same notation is used for the texture parameters).

### 6.2.6 Identity/Expression Separated 3D Morphable Model

The 3D Morphable Model separates shape and albedo parameters from pose and lighting, which makes pose and lighting-invariant recognition possible. The same idea can be used for expression-invariant face recognition from 3D shape [2] (see Sect. 6.5.5).

For these experiments, an identity/expression separated 3D Morphable Model [10] built from 270 subjects was used. It was built from one neutral expression face scan per identity and 135 expression scans of a subset of the subjects. The identity model was built from the 270 neutral expression scans as in Sect. 6.2.3.

Additionally, for each of the 135 expression scans, we calculated an expression vector as the difference between the expression scan and the corresponding neutral scan of that subject. This data is already mode-centered, if we regard the neutral expression as the natural mode of expression data. On these offset vectors again PCA was applied to get  $N_E$  expression components  $\mathbf{S}_E^i$  and expression coefficients  $\alpha'_i$ , such that the complete expression model is

$$\mathbf{S} = \bar{\mathbf{S}} + \sum_{i=1}^{N_S} \alpha_i \cdot \mathbf{S}^i + \sum_{i=1}^{N_E} \alpha'_i \cdot \mathbf{S}_E^i. \quad (6.10)$$

This model assumes, that it is possible to transfer the expression deformation from one face to another. Even if this should not be strictly true, which is what authors using for example, tensor based models [46] assume, it is a good enough assumption to make the method *invariant* to expressions. And a big advantage of this independence assumption is that we can train on far less data than in a tensor framework, because we do not need the full Cartesian product of expressions and identities. In fact we have not even had any expression scan available for most of the training subjects, but were able to learn useful statistics from the available scans.

We perform identification by fitting the model from (6.10), to new scans, taking the Gaussian prior over the identity and expression coefficients into account. The maximum likelihood coefficients given a new observation are unique, even if the shape and expression basis are not linearly independent.

We use the registered scans and a mirrored version of each registered scan to increase the variability of the model. This allows us to calculate a model with more than 175 neutral coefficients.

### 6.3 Morphable Model to Synthesize Images

One part of the analysis by synthesis loop is the synthesis (i.e., the generation of accurate face images viewed from any pose and illuminated under any condition). This process is explained in this section.

#### 6.3.1 Shape Projection

To render the image of a face, the 3D shape must be projected to the 2D image frame. This is performed in two steps. First, a 3D rotation and translation (i.e., a rigid transformation) maps the object-centered coordinates,  $\mathbf{S}$ , to a position relative to the camera.

$$\mathbf{W} = \mathbf{R}_\gamma \mathbf{R}_\theta \mathbf{R}_\phi \mathbf{S} + \mathbf{t}_w \mathbf{1}_{1 \times N_v}. \quad (6.11)$$

The angles  $\phi$  and  $\theta$  control in-depth rotations around the vertical and horizontal axis, and  $\gamma$  defines a rotation around the camera axis;  $\mathbf{t}_w$  is a 3D translation. A projection then maps a vertex  $k$  to the image plane in  $(x_j, y_j)$ . We typically use one of two projections, either the perspective or the weak perspective projection.

$$\text{Perspective: } \begin{cases} x_j = t_x + f \frac{\mathbf{W}_{1,j}}{\mathbf{W}_{3,j}}, \\ y_j = t_y + f \frac{\mathbf{W}_{2,j}}{\mathbf{W}_{3,j}}, \end{cases} \quad \text{Weak perspective: } \begin{cases} x_j = t_x + f \mathbf{W}_{1,j}, \\ y_j = t_y + f \mathbf{W}_{2,j}, \end{cases} \quad (6.12)$$

where  $f$  is the focal length of the camera, which is located in the origin; and  $(t_x, t_y)$  defines the image-plane position of the optical axis.

#### 6.3.2 Illumination and Color Transformation

##### 6.3.2.1 Ambient and Directed Light

We simulate the illumination of a face using an ambient light and a directed light. The effects of the illumination are obtained using the standard Phong model, which approximately describes the diffuse and specular reflection on a surface [21]; see [7] for further details. The parameters of this model are the intensity of the ambient light ( $L_{r,amb}$ ,  $L_{g,amb}$ ,  $L_{b,amb}$ ), the intensity of the directed light ( $L_{r,dir}$ ,  $L_{g,dir}$ ,  $L_{b,dir}$ ), its direction ( $\theta_l$  and  $\phi_l$ ), the specular reflectance of human skin ( $k_s$ ), and the angular distribution of the specular reflections of human skin ( $v$ ).

##### 6.3.2.2 Color Transformation

Input images may vary a lot with respect to the overall tone of color. To be able to handle a variety of color images as well as gray level images and even paintings, we

apply gains  $g_r$ ,  $g_g$ ,  $g_b$ , offsets  $o_r$ ,  $o_g$ ,  $o_b$ , and a color contrast  $c$  to each channel [6]. This is a linear transformation that multiplies the RGB color of a vertex (after it has been illuminated) by the matrix  $\mathbf{M}$  and adds the vector  $\mathbf{o} = [o_r, o_g, o_b]^T$ , where

$$\mathbf{M}(c, g_r, g_g, g_b) = \begin{pmatrix} g_r & 0 & 0 \\ 0 & g_g & 0 \\ 0 & 0 & g_b \end{pmatrix} \cdot \left[ \mathbf{I} + (1 - c) \begin{pmatrix} 0.3 & 0.59 & 0.11 \\ 0.3 & 0.59 & 0.11 \\ 0.3 & 0.59 & 0.11 \end{pmatrix} \right]. \quad (6.13)$$

For brevity, the illumination and color transformation parameters are regrouped in the vector  $\boldsymbol{\iota}$ . Hence, the illuminated texture depends on the coefficients of the linear combination regrouped in  $\boldsymbol{\beta}$ , on the light parameters  $\boldsymbol{\iota}$ , and on  $\boldsymbol{\alpha}$  and  $\boldsymbol{\rho}$  used to compute the normals and the viewing direction of the vertices required for the Phong illumination model. Similarly to the shape, we denote the color of a vertex  $i$  by the vector  $\mathbf{t}_i(\theta)$ , where  $\theta$  denotes the ensemble of model parameters.

Using the Morphable Model framework, the image of the face of any individual seen from any angle and illuminated from any direction can be obtained from the shape parameters  $\alpha_i$ , the texture parameters  $\beta_i$ , the shape projection parameters, and the illumination parameters by

$$\mathbf{I}(x_i(\theta), y_i(\theta)) = \mathbf{t}_i(\theta). \quad (6.14)$$

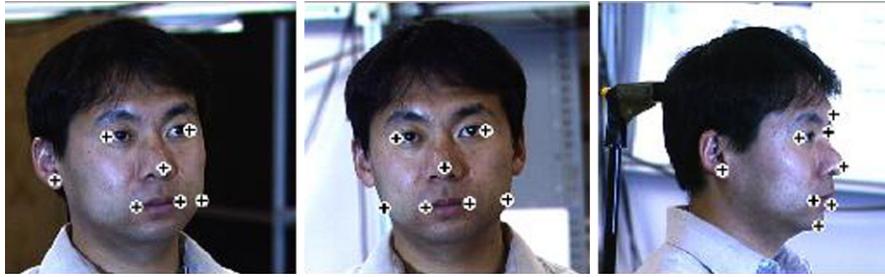
where  $x_i$  and  $y_i$  are computed by (6.12).

In a nutshell, the prior models accounting for the variations of the face image are devised as follows: Gaussian probability models for the registered 3D shape and albedo, a Phong reflectance model, a single directed light source for the illumination model, and rigid pose variations.

## 6.4 Image Analysis with a 3D Morphable Model

In the analysis by synthesis framework, an algorithm seeks the parameters of the model that render a face as close to the input image as possible. These parameters explain the image and can be used for high-level tasks such as identification. This algorithm is called a *fitting algorithm*. It is characterized by the following four features.

- **Efficient:** The computational load allowed for the fitting algorithm is clearly dependent on the applications. Security applications, for instance, require fast algorithms (i.e., near real time).
- **Robust** (against non-Gaussian noise): The assumption of normality of the difference between the image synthesized by the model and the input image is generally violated owing to the presence of accessories or artifacts (glasses, hair, specular highlight).
- **Accurate:** The accuracy of the reconstruction must be sufficient to allow the subsequent use of the reconstructed parameters.



**Fig. 6.5** Initialization: Seven landmarks for front and side views and eight for the profile view are manually labeled for each input image

- **Automatic:** The fitting should require as little human intervention as possible, optimally with no initialization. For frontal images, the process has been completely automated in [13].

An algorithm capable of any of the four aforementioned features is difficult to set up. An algorithm capable of *all* four features is the holy grail of model-based computer vision. In this chapter, we present two fitting algorithms. The first one, called *stochastic newton optimization* (SNO) is accurate but computationally expensive: a fitting takes 4.5 minutes on a 2 GHz Pentium IV. SNO is detailed elsewhere [7].

Romdhani and Vetter [39] improved the fitting by using a cost function that, as well as the pixel intensity, uses various image features such as the edges or the location of the specular highlights. The overall cost function obtained is smoother and, hence, a stochastic optimization algorithm is not needed to avoid the local minima problem. This leads to the Multi-Features Fitting (MMF) algorithm that has a wider radius of convergence and a higher level of precision.

In Sect. 6.5, we compare identification results for both algorithms. Both, the SNO and the MMF fitter uses the MPI Morphable Model detailed in [6]. Additionally, we show results obtained with the MMF fitter and the BFM (see Sect. 6.2).

As initialization, the algorithms require the correspondences between some of the model vertices (typically eight) and the input image. In the experiments shown here, these correspondences are set manually, while we expect that these points could also be found manually using methods such as [22, 24]. The landmark points are required to obtain a good initial condition for the iterative algorithm. The 2D positions in the image of these  $N_l$  points are set in the matrix  $\mathbf{L}_{2 \times N_l}$ . They are in correspondence with the vertex indices set in the vector  $\mathbf{v}_{N_l \times 1}$ . The positions of these landmarks for three views are shown in Fig. 6.5.

#### 6.4.1 Maximum a Posteriori Estimation of the Parameters

Both algorithms presented aim to find the model parameters  $\alpha, \rho, \beta, \iota$  that explain an input image. To increase the robustness of the algorithms, these parameters are estimated by a *maximum a posteriori* (MAP) estimator, which maximizes

$p(\boldsymbol{\alpha}, \boldsymbol{\rho}, \boldsymbol{\beta}, \boldsymbol{\iota} | \mathbf{I}_{\text{input}}, \mathbf{L})$  [6]. Applying the Bayes rule and neglecting the dependence between parameters yield

$$p(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\rho}, \boldsymbol{\iota} | \mathbf{I}_{\text{input}}, \mathbf{L}) \propto p(\mathbf{I}_{\text{input}} | \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\rho}, \boldsymbol{\iota}) \cdot p(\mathbf{L} | \boldsymbol{\alpha}, \boldsymbol{\rho}) \cdot p(\boldsymbol{\alpha}) \cdot p(\boldsymbol{\beta}) \cdot p(\boldsymbol{\rho}) \cdot p(\boldsymbol{\iota}). \quad (6.15)$$

The expression of the priors  $p(\boldsymbol{\alpha})$  and  $p(\boldsymbol{\beta})$ , is given by (6.5). For each shape projection and illumination parameter, we assume a Gaussian probability distribution with mean  $\bar{\rho}_i$  and  $\bar{\iota}_i$  and variance  $\sigma_{\rho,i}^2$  and  $\sigma_{\iota,i}^2$ . These values are set manually.

Assuming that the  $x$  and  $y$  coordinates of the landmark points are independent and that they have the same Gaussian distribution with variance  $\sigma_L^2$ , we arrive at the energy

$$E_L = -2 \log p(\mathbf{L} | \boldsymbol{\alpha}, \boldsymbol{\rho}) = \frac{1}{\sigma_L^2} \sum_j^{N_l} \left\| \mathbf{L}_{\cdot,j} - \begin{pmatrix} x_{\mathbf{v}_j} \\ y_{\mathbf{v}_j} \end{pmatrix} \right\|^2. \quad (6.16)$$

#### 6.4.2 Stochastic Newton Optimization

The likelihood of the input image given the model parameters is expressed in the image frame. Assuming that all the pixels are independent and that they have the same Gaussian distribution with variance  $\sigma_I^2$ , gives:

$$E_I = -2 \log p(\mathbf{I}_{\text{input}} | \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\rho}, \boldsymbol{\iota}) = \frac{1}{\sigma_I^2} \sum_{x,y} \left\| \mathbf{I}_{\text{input}}(x, y) - \mathbf{I}(x, y; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\rho}, \boldsymbol{\iota}) \right\|^2. \quad (6.17)$$

The sum is carried out over the pixels that are projected from the vertices in  $\mathcal{Q}(\boldsymbol{\alpha}, \boldsymbol{\rho})$ . For each pixel location, the norm is computed over the three color channels. The overall energy to be minimized is then:

$$E = \frac{1}{\sigma_I^2} E_I + \frac{1}{\sigma_L^2} E_L + \sum_i \frac{\alpha_i^2}{\sigma_{S,i}^2} + \sum_i \frac{\beta_i^2}{\sigma_{T,i}^2} + \sum_i \frac{(\rho_i - \bar{\rho}_i)^2}{\sigma_{\rho,i}^2} + \sum_i \frac{(\iota_i - \bar{\iota}_i)^2}{\sigma_{\iota,i}^2}. \quad (6.18)$$

This log-likelihood is iteratively minimized by performing a Taylor expansion up to the second order (i.e., approximating the log-likelihood by a quadratic function) and computing the update that minimizes the quadratic approximation. The update is added to the current parameter to obtain the new parameters.

We use a stochastic minimization to decrease the odds of getting trapped in a local minima and to decrease the computational time: Instead of computing  $E_I$  and its derivatives on all pixels of  $\Psi(\boldsymbol{\alpha}, \boldsymbol{\rho})$ , it is computed only on a subset of 40 pixels thereof. These pixels are randomly chosen at each iteration. The first derivatives are computed analytically using the chain rule. The Hessian is approximated by a diagonal matrix computed by numeric differentiation every 1000 iterations. This algorithm was further detailed by Blanz and Vetter [7]. The SNO algorithm is extremely accurate (see the experiments in Sect. 6.5).



**Fig. 6.6** Stochastic Newton optimization fitting results: Three-dimensional reconstruction from CMU-PIE images using the SNO fitting algorithm using the MPI model. *Top*: originals. *Middle*: reconstructions rendered into original. *Bottom*: novel views. The pictures shown here are difficult to fit due to harsh illumination, profile views, or eyeglasses. Illumination in the third image is not fully recovered, so part of the reflections are attributed to texture

#### 6.4.2.1 Fitting Results

Several fitting results and reconstructions are shown in Fig. 6.6. They were obtained with the SNO algorithm and the MPI model on some of the PIE images (see Sect. 13.5.4 of Chap. 13). These images are illuminated with ambient light and one directed light source. The algorithm was initialized with seven or eight landmark points (depending on the pose of the input image) (Fig. 6.5). In the third column, the separation between the albedo of the face and the illumination is not optimal: part of the specular reflections were attributed to the texture by the algorithm. This may be due to shortcomings of the Phong illumination model for reflections at grazing angles or to a prior probability inappropriate for this illumination condition. (The prior probabilities of the illumination and rigid parameters,  $\sigma_{\rho,i}^2$  and  $\sigma_{t,i}^2$ , are kept constant for fitting the 4488 PIE images.)

#### 6.4.3 Multiple Feature Fitting

In [39], an improvement over the SNO algorithm is proposed. It is based on the assumption that the prior by the texture and shape PCA models are not strong enough

to obtain an accurate estimate of the 3D shape when only a few manually set anchor points are used as input. This is because the cost function to be minimized is highly nonconvex and exhibits many local minima. In fact, the shape model requires the correspondence between the input image and the reference frame to be found for every visible vertices. Using only facial color information to recover, the correspondence is not optimal and may be trapped in regions that present similar intensity variations (eyes/eyebrows, for instance). This is why, we use not only the pixel intensities but also other features of the input image to obtain a more accurate estimate of the correspondence and, as a result, of the 3-D shape. One example of such a feature is the edges. Other features that improve the shape and texture estimate are the specular highlights and the texture constraints. The specular highlight feature uses the specular highlight location, detected on the input image, to refine the normals and, thereby, the 3-D shape of the vertices affected. The texture constraint enforces that the estimated texture lies within a specific range (typically [0, 255]), which improves the illumination estimate. The overall resulting cost function is smoother and easier to minimize, making the system more robust and reliable. A question raised by this problem is how to fuse the different image cues to form the optimal parameter estimate. We chose a Bayesian framework and maximize the posterior probability of the parameters given the image and its features.

This analysis algorithm, called the Multiple Feature Fitting algorithm, is briefly outlined here; a more detailed explanation is provided in [38, 39]. It is demonstrated that, if the features (pixel intensities, edges, and specular highlights) are independent and extracted from the input image by a deterministic algorithm, then the overall cost function is a linear combination of the cost function of each feature taken separately

$$\min_{\theta} \tau^I E_I + \tau^E E_E + \tau^S E_S + \tau^P E_P + \tau^T E_T \quad (6.19)$$

where  $E_I$  denotes the pixel intensity feature (6.17),  $E_P = \sum_{i=1}^{N_S} (\alpha_i^2 / \sigma_{S,i}^2) + \sum_{i=1}^{N_T} (\beta_i^2 / \sigma_{T,i})^2$  denotes the prior feature and  $E_E$ ,  $E_S$  and  $E_T$  denote, respectively, the edge, specular highlights, and texture constraints cost functions. The  $\tau$ 's are weighting parameters. A detailed explanation of these cost functions is provided in [39]. The overall cost function is minimized using a Levenberg–Marquardt optimization algorithm.

The image edges provide information about the 2-D shape independent of the texture and of the illumination. Hence, the cost function used to fit the edge features provides a more direct constraint on the correspondences and on the shape and pose parameters. The edge feature is useful to recover the correspondences of specific facial characteristics (eyes, eyebrows, mouth, nose). On the other hand, it does not carry much depth information. So it is beneficial to use the edge and intensity features in combination. The specular highlights are easy to detect: the pixels with a specular highlight saturate. Additionally, they give a direct relationship between the 3-D geometry of the surface at these points, the camera direction, and the light direction: a point on a specular highlight has a normal that has the direction of the bisector of the angle formed by the light source direction and the camera direction.

Hence, the specular highlight cost function is used to refine the shape estimate for the vertices that are projected onto specular highlights of the input image.

In order to accurately estimate the 3-D shape, it is necessary to recover the texture, the light direction, and its intensity. To separate the contribution of the texture from light in a pixel intensity value, a Gaussian texture prior model is used (see (6.5)). However, it appears that this prior model is not restrictive enough and is able to instantiate invalid textures (negative and overflowing color values). To constrain the texture model and to improve the separation of light source strength from albedo, we introduce a feature that constrains the range of valid albedo values.

## 6.5 Experimental Evaluation

We evaluated the 3D Morphable Model and the fitting algorithms on two applications: identification and verification. In the identification task, an image of an unknown person is provided to our system. The unknown face image is then compared to a database of known people, called the gallery set. The ensemble of unknown images is called the probe set. In the identification task, it is assumed that the individual in the unknown image is in the gallery. In a verification task, the individual in the unknown image claims an identity. The system must then accept or reject the claimed identity. Verification performance is characterized by two statistics: The verification rate is the rate at which legitimate users are granted access. The false alarm rate is the rate at which impostors are granted access. See Sect. 14.1 of Chap. 14 for more detailed explanations of these two tasks.

We evaluated our approach on three data sets. **Set 1:** a portion of the FERET data set containing images with various poses. In the FERET nomenclature, these images correspond to the series ba through bk. We omitted the images bj as the subjects present an expression that is not accounted for by our 3D Morphable Model. This data set includes 194 individuals across nine poses at constant lighting condition except for the series bk, which used a frontal view at another illumination condition than the rest of the images. **Set 2:** a portion of the CMU–PIE data set including images of 68 individuals at a neutral expression viewed from 13 different angles at ambient light. **Set 2:** a portion of the CMU–PIE data set containing images of the same 68 individuals at three poses (frontal, side, and profile) and illuminated by 21 different directions and by ambient light only. Among the 68 individuals in Set 2, a total of 28 wear glasses, which are not modeled and could decrease the accuracy of the fitting. None of the individuals present in these three sets was used to construct the 3D Morphable Model. These sets cover a large ethnic variety, not present in the set of 3D scans used to build the model. Refer to Chap. 13 for a formal description of the FERET and PIE set of images.

Identification and verification are performed by fitting an input face image to the 3D Morphable Model, thereby extracting its identity parameters,  $\alpha$  and  $\beta$ . Then recognition tasks are achieved by comparing the identity parameters of the input image with those of the gallery images. We defined the identity parameters of a face image, denoted by the vector  $c$ , by stacking the shape and texture parameters of the

global and segmented models (see Sect. 6.2.5) and rescaling them by their standard deviations.

$$\mathbf{c} = \left[ \frac{\alpha_1^g}{\sigma_{S,1}}, \dots, \frac{\alpha_{99}^g}{\sigma_{S,99}}, \frac{\beta_1^g}{\sigma_{T,1}}, \dots, \frac{\beta_{99}^g}{\sigma_{T,99}}, \frac{\alpha_1^{s_1}}{\sigma_{S,1}}, \dots, \frac{\alpha_{99}^{s_1}}{\sigma_{S,99}}, \dots, \frac{\beta_{99}^{s_4}}{\sigma_{T,99}} \right]^T. \quad (6.20)$$

We defined two distance measures to compare two identity parameters  $\mathbf{c}_1$  and  $\mathbf{c}_2$ . The first measure,  $d_A$ , is based on the angle between the two vectors (it can also be seen as a normalized correlation), and is insensitive to the norm of both vectors. This is favorable for recognition tasks, as increasing the norm of  $\mathbf{c}$  produces a caricature (see Sect. 6.2.4) which does not modify the perceived identity. The second distance [7],  $d_W$ , is based on discriminant analysis [19] and favors directions where identity variations occur. Denoting by  $\mathbf{C}_W$  the pooled within-class covariance matrix, these two distances are defined by:

$$d_A = \frac{\mathbf{c}_1^T \cdot \mathbf{c}_2}{\sqrt{(\mathbf{c}_1^T \cdot \mathbf{c}_1)(\mathbf{c}_2^T \cdot \mathbf{c}_2)}} \quad \text{and} \quad d_W = \frac{\mathbf{c}_1^T \cdot \mathbf{C}_W \cdot \mathbf{c}_2}{\sqrt{(\mathbf{c}_1^T \cdot \mathbf{C}_W \cdot \mathbf{c}_1)(\mathbf{c}_2^T \cdot \mathbf{C}_W \cdot \mathbf{c}_2)}} \quad (6.21)$$

Results on Sets 1 and 3 use the distance  $d_W$  with, for Set 1, a within-class covariance matrix learned on Set 2, and vice versa.

### 6.5.1 Pose Variation

In this section, we present identification and verification results for images of faces that vary in pose. Table 6.1 compares percentages of correct rank 1 identification obtained with the SNO and MFF fitting algorithm on Set 1 (FERET). Table 6.2 shows more details for the SNO fitting. The 10 poses were used to constitute gallery sets. The results are detailed for each probe pose. The results for the front view gallery (here in bold) were first reported in [7]. The first plot of Fig. 6.7 shows the ROC for a verification task for the front view gallery and the nine other poses in the probe set. The verification rate for a false alarm rate of 1% is 87.9%.

### 6.5.2 Pose and Illumination Variations

In this section, we investigate the performance of our method in the presence of combined pose and illumination variations. The SNO and the MMF algorithm was applied to the images of Set 2, CMU-PIE images of 68 individuals varying with respect to three poses, 21 directed light and ambient light conditions. Table 6.3 presents the rank 1 identification performance averaged over all lighting conditions for front, side, and profile view galleries. Illumination 13 was selected for the galleries. The second plot of Fig. 6.7 shows the ROC for a verification using as gallery

**Table 6.1** Rank 1 identification results obtained on Set 1 (subset of the FERET database) for the SNO or MFF, resp. with the MPI or BFM, resp

Probe view	Pose $\phi$	Identification rate		
		SNO, MPI [7]	MFF, MPI [38]	MFF, BFM [34]
bb	38.9°	96.4%	92.7%	97.4%
bc	27.4°	99.0%	99.5%	99.5%
bd	18.9°	99.5%	99.5%	100.0%
be	11.2°	Gallery		
ba	1.1°	100.0%	96.9%	99.0%
bf	-7.1°	97.4%	99.5%	99.5%
bg	-16.3°	96.4%	95.8%	97.9%
bh	-26.5°	95.9%	89.6%	94.8%
bi	-37.9°	91.2%	77.1%	83.0%
bk	0.1°	94.3%	80.7%	90.7%
Mean		96.7%	92.4%	95.8%

**Table 6.2** SNO identification performances on Set 1 (subset of the FERET database)

Parameter	Performance (%) by probe view										
	bi	bh	bg	bf	ba	be	bd	bc	bb	bk	Mean
$\phi$	-37.9°	-26.5°	-16.3°	-7.1°	1.1°	11.2°	18.9°	27.4°	38.9°	0.1°	

Gallery view											
bi	-	98.5	94.8	87.6	85.6	87.1	87.1	84.0	77.3	76.8	86.5
bh	99.5	-	97.4	95.9	91.8	95.9	94.8	92.3	83.0	86.1	93.0
bg	97.9	99.0	-	99.0	95.4	96.9	96.9	91.2	81.4	89.2	94.1
bf	95.9	99.5	99.5	-	97.9	96.9	99.0	94.8	88.1	95.4	96.3
ba	<b>90.7</b>	<b>95.4</b>	<b>96.4</b>	<b>97.4</b>	-	<b>99.5</b>	<b>96.9</b>	<b>95.4</b>	<b>94.8</b>	<b>96.9</b>	<b>95.9</b>
be	91.2	95.9	96.4	97.4	100.0	-	99.5	99.0	96.4	94.3	96.7
bd	88.7	97.9	96.9	99.0	97.9	99.5	-	99.5	98.5	92.3	96.7
bc	87.1	90.7	91.2	94.3	96.4	99.0	99.5	-	99.0	87.6	93.9
bb	78.9	80.4	77.8	80.9	87.6	94.3	94.8	99.0	-	74.7	85.4
bk	83.0	88.1	92.3	95.4	96.9	94.3	93.8	88.7	79.4	-	90.2

The overall mean of the table is 92.9%.  $\phi$  is the average estimated azimuth pose angle of the face. Ground truth for  $\phi$  is not available. Condition bk has different illumination than the others. The row in bold is the front view gallery (condition ba).

a side view illuminated by light 13 and using all other images of the set as probes. The verification rate for a 1% false alarm rate was 77.5%. These results were first reported by Blanz and Vetter [7].

**Table 6.3** Mean percentage of correct identification obtained after a SNO or MMF fitting, resp. on Set 2, averaged over all lighting conditions for front, side, and profile view galleries SNO [7]

Gallery view	Performance (%) by probe view			Mean
	Front	Side	Profile	
Front	99.8% (97.1–100)	97.8% (82.4–100)	79.5% (39.7–94.1)	92.3%
Side	99.5% (94.1–100)	99.9% (98.5–100)	85.7% (42.6–98.5)	95.0%
Profile	83.0% (72.1–94.1)	86.2% (61.8–95.6)	98.3% (83.8–100)	89.0%
Mean				92.1%

MMF (with MPI model) [38]

Gallery view	Performance (%) by probe view			Mean
	Front	Side	Profile	
Front	99.9% (98.5–100.0)	98.4% (91.0–100.0)%	75.6% (38.8–94.0)	91.3%
Side	96.4% (89.6–100.0)	99.3% (97.0–100.0)	83.7% (52.2–100.0)	93.1%
Profile	76.3% (64.2–91.0)	86.0% (67.2–98.5)	89.4% (64.2–98.5)	83.9%

MFF (with BFM) [34]

Gallery view	Performance (%) by probe view			Mean
	Front	Side	Profile	
Front	98.9%	96.1%	75.7%	90.2%
Side	96.9%	99.9%	87.8%	94.9%
Profile	79.0%	89.0%	98.3%	88.8%
Mean	91.6%	95.0%	87.3%	91.3%

Numbers in parenthesis are percentages for the worst and best illumination within each probe set

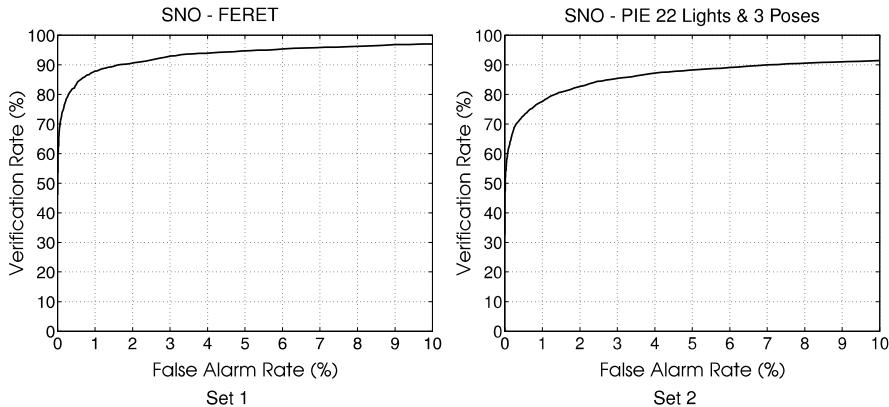
### 6.5.3 Identification Confidence

In this section, we present an automated technique for assessing the quality of the fitting in terms of a fitting score (FS). We show that the fitting score is correlated with identification performance and hence, may be used as an identification confidence measure. This method was first presented by Blanz et al. [9].

A fitting score can be derived from the image error and from the model coefficients of each fitted segment from the average.

$$FS = f\left(\frac{E_I}{N_{vv}}, \boldsymbol{\alpha}_g, \boldsymbol{\beta}_g, \boldsymbol{\alpha}_{s_1}, \boldsymbol{\beta}_{s_1}, \dots, \boldsymbol{\beta}_{s_4}\right). \quad (6.22)$$

Although the FS can be derived by a Bayesian method, we learned it using a support vector machine (SVM) (see Vapnik [44] for a general description of SVM and Blanz et al. [9] for details about FS learning).



**Fig. 6.7** Receiver operator characteristic for a verification task obtained with the SNO algorithms on different sets of images

**Fig. 6.8** Identification results as a function of the fitting score

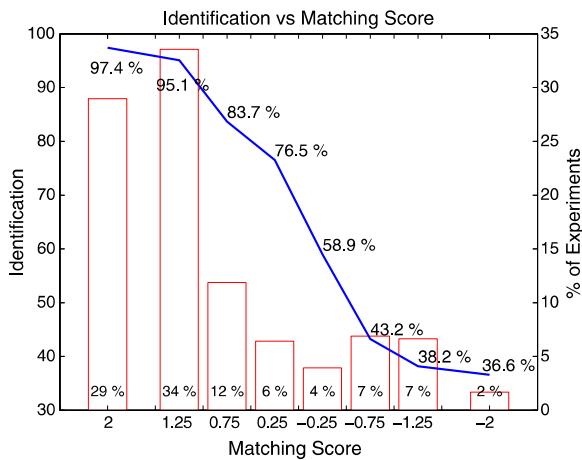


Figure 6.8 shows the identification results for the PIE images varying in illumination across three poses, with respect to the FS for a gallery of side views.  $FS > 0$  denotes good fittings and  $FS < 0$  poor ones. We divided the probe images into eight bins of different FS and computed the percentage of correct rank 1 identification for each of these bins. There is a strong correlation between the FS and identification performance, indicating that the FS is a good measure of identification confidence.

### 6.5.4 Virtual Views as an Aid to Standard Face Recognition Algorithms

The face recognition vendor test (FRVT) 2002 [35] was an independently administered assessment, conducted by the U.S. government of the performance of commercially available automatic face recognition systems. The test is described in Sect. 14.2 of Chap. 14. It was realized that identification of face images significantly drops if the probe image is nonfrontal. This is a common scenario, because the gallery images are typically taken in a controlled situation, while the probe image might only be a snapshot by a surveillance camera. As there are far fewer probe images than gallery images, it is feasible to invest preprocessing time into the probe image, while a preprocessing of the huge gallery set would be too expensive. Hence, one of the questions addressed by FRVT02 is this: Do identification performances of nonfrontal face images improve if the pose of the probe is normalized by our 3D Morphable Model? To answer this question, we normalized the pose of a series of images [8] which were then used with the top performing face recognition systems. Normalizing the pose means to fit an input image where the face is nonfrontal, thereby estimating its 3D structure, and to synthesize an image with a frontal view of the estimated face. Examples of pose-normalized images are shown in Fig. 6.9. As neither the hair nor the shoulders are modeled, the synthetic images are rendered into a standard frontal face image of one person. This normalization is performed by the following steps.

1. Manually define up to 11 landmark points on the input image to ensure optimal quality of the fitting.
2. Run the SNO fitting algorithm described in Sect. 6.4.2 yielding a 3D estimation of the face in the input image.
3. Render the 3D face in front of the standard image using the rigid parameters (position, orientation, and size) and illumination parameters of the standard image. These parameters were estimated by fitting the standard face image.
4. Draw the hair of the standard face in front of the forehead of the synthetic image. This makes the transition between the standard image and the synthetic image smoother.

The normalization was applied to images of 87 individuals at five poses (frontal, two side views, one up view, and a down view). Identifications were performed by the 10 participants to FRVT02 (see pages 31 and 32 of Phillips et al. [35]) using the frontal view images as gallery and nine probe sets: four probe sets with images of nonfrontal views, four probe sets with the normalized images of the nonfrontal views and one probe set with our preprocessing normalization applied to the front images. The comparison of performances between the normalized images (called morph images) and the raw images is presented on Fig. 6.10 for a verification experiment (the hit rate is plotted for a false alarm rate of 1%).

The frontal morph probe set provides a baseline for how the normalization affects an identification system. In the frontal morph probe set, the normalization is applied to the gallery images. The results on this probe set are shown on the first column



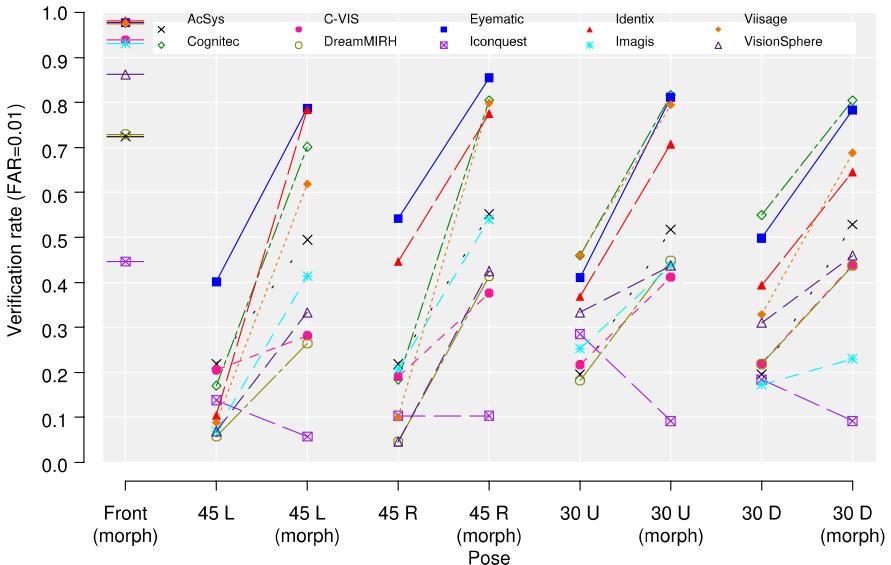
**Fig. 6.9** From the original images (*top row*), we recover the 3D shape (*middle row*), by SNO fitting. Mapping the texture of visible face regions on the surface and rendering it into a standard background, which is a face image we selected, produces virtual front views (*bottom row*). Note that the frontal-to-frontal mapping, which served as a baseline test, involves hairstyle replacement (*bottom row, center*)

of Fig. 6.10. The verification rates would be 1.0, if a system were insensitive to the artifacts introduced by the Morphable Model and did not rely on the person's hairstyle, collar, or other details that are exchanged by the normalization (which are, of course, no reliable features by which to identify one person). The sensitivity to the Morphable Model of the 10 participants ranges from 0.98 down to 0.45. The overall results showed that, with the exception of Iconquest, Morphable Models significantly improved (and usually doubled) performance.

If there is also pose or illumination variation in the gallery, and enough resources are available then an identification directly in the normalized 3D Morphable Model space as proposed in [7].

### 6.5.5 Face Identification on 3D Scans

In this section, we describe an expression-invariant method for face recognition by fitting an identity/expression separated 3D Morphable Model (see Sect. 6.2.6) to



**Fig. 6.10** The effect of the original images versus normalized images using the 3D Morphable Models. The verification rate at a false alarm rate of 1% is plotted. (Courtesy of Jonathon Phillips)

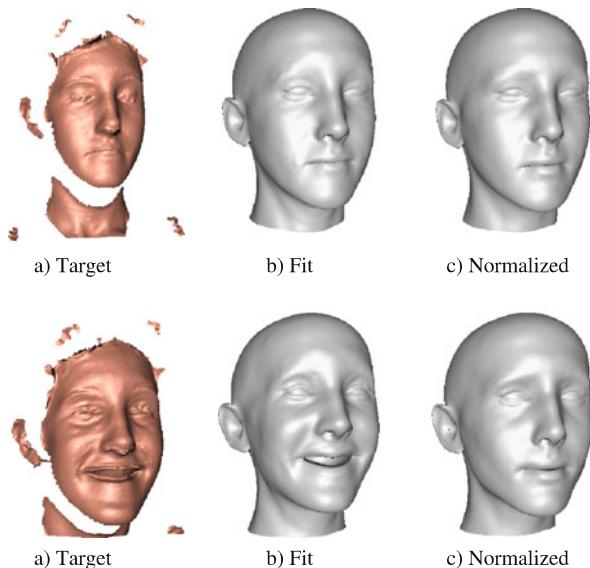
shape data and normalize the resulting face by removing the pose and expression components. The results were first published in [2]. The expression model greatly improves recognition and retrieval rates in the uncooperative setting, while achieving recognition rates on par with the best recognition algorithms in the face recognition great vendor test. The fitting is performed with a robust nonrigid ICP algorithm (a variant of [1]). See Fig. 6.11 for an example of expression normalization. The expression and pose normalized data allow efficient and effective recognition.

A 3D MM has been fitted to range data before and the results were even evaluated on part of the UND database [11]. The approach used here differs from [11] in the fitting method employed, and in the use of an expression model to improve face recognition. Additionally, our method is fully automatic, needing only a single easy to detect directed landmark, while [11] needed manually selected landmarks.

We evaluated the system on two databases with and without the expression model. We used the GavabDB [32] database and the UND [14] database. For both databases, only the shape information was used. The GavabDB database contains 427 scans, with seven scans per ID, three neutral and four expressions. The expressions in this dataset vary considerably, including sticking out the tongue and strong facial distortions. Additionally it has strong artifacts due to facial hair, motion and the bad scanner quality. This dataset is typical for a noncooperative environment. The UND database was used in the face recognition grand challenge [36] and consists of 953 scans, with one to eight scans per ID. It is of better quality and contains only slight expression variations. It represents a cooperative scenario.

The fitting was initialized by detecting the nose, and assuming that the face is upright and looking along the  $z$ -axis. The nose was detected with the method of [41].

**Fig. 6.11** Expression normalization for two scans of the same individual. The robust fitting gives a good estimate (b) of the true face surface given the noisy measurement (a). It fills in holes and removes artifacts using prior knowledge from the face model. The pose and expression normalized faces (c) are used for face recognition



The GavabDB database has the scans already aligned and the tip of the nose is at the origin. We used this information for the GavabDB experiments. The same regularization parameters were used for all experiments, even though the GavabDB data is more noisy than the UND data. The parameters were set manually based on a few scans from the GavabDB database. We used 250 principal identity components and 60 expression components for all experiments.

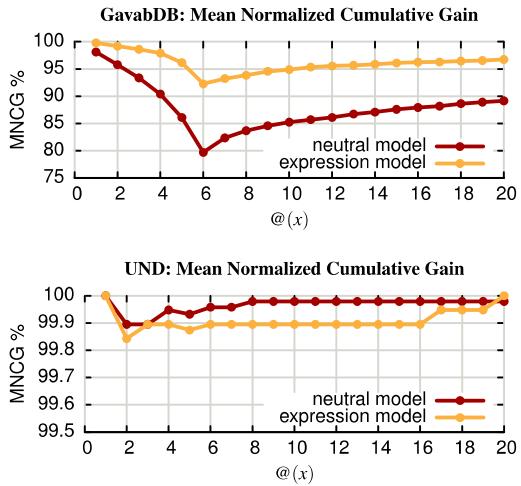
In the experiments, the distances between all scans were calculated, and we measured recognition and retrieval rates by treating every scan once as the probe and all other scans as the gallery. Both databases were used independently.

### 6.5.5.1 Results

As expected, the two datasets behave differently because of the presence of expressions in the examples. We first describe the results for the cooperative and then for the uncooperative setting.

**UND** For the UND database, we have good recognition rates with the neutral model. The mean cumulative normalized gain curve in Fig. 6.12 shows for varying retrieval depth the number of correctly retrieved scans divided by the maximal number of scans that could be retrieved at this level. From this it can be seen that the first match is always the correct match, if there is any match in the database. But for some probes no example is in the gallery. Therefore for face recognition we have to threshold the maximum allowed distance to be able to reject impostors. Varying the distance threshold leads to varying false acceptance rates (FAR) and false rejection rates (FRR), which are shown in Fig. 6.13. Even though we have been tuning the

**Fig. 6.12** For the expression dataset the retrieval rate is improved by including the expression model, while for the neutral expression dataset the performance does not decrease. Plotted is the mean normalized cumulative gain, which is the number of retrieved correct answers divided by the number of possible correct answers. Note also the different scales of the MNCG curves for the two datasets. Our approach has a high accuracy on the neutral (UND) dataset

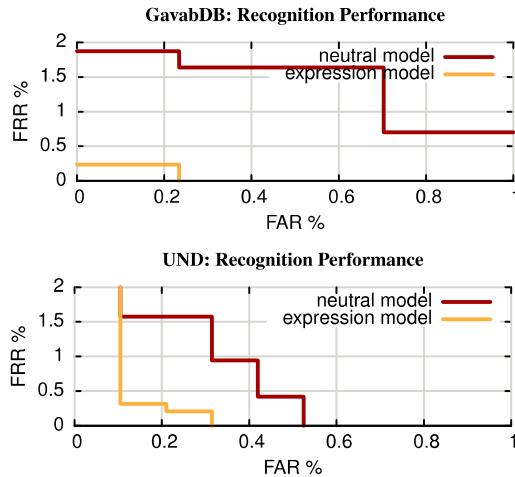


model to the GavabDB dataset and not the UND dataset our recognition rates at any FAR rate are as good or better than the best results from the face recognition vendor test. This shows, that our basic face recognition method without expression modeling gives convincing results. Now we analyze how the expression modeling impacts recognition results on this expression-less database. If face and expression space are not independent, then adding invariance towards expressions should make the recognition rates decrease. In fact, while we find no significant increase in recognition and retrieval rates, the results are also not worse when including expression variance. Let us now turn towards the expression database, where we expect to see an increase in recognition rate due to the expression model.

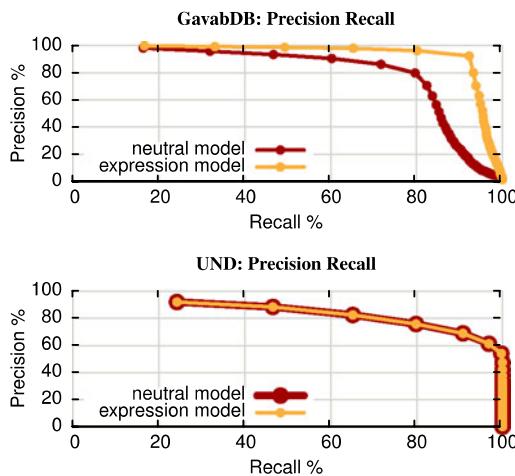
**GavabDB** The recognition rates on the GavabDB without expression model are not quite as good as for the expression-less UND dataset, so here we hope to find some improvement by using expression normalization. And indeed, the closest point recognition rate with only the neutral model is 98.1% which can be improved to 99.7% by adding the expression model. Also the FAR/FRR values decrease considerably. The largest improvement can be seen in retrieval performance, displayed in the precision recall curves in Fig. 6.14 and mean cumulative normalized gain curves in Fig. 6.12. This is because there are multiple examples in the gallery, so finding a single match is relatively easy. But retrieving all examples from the database, even those with strong expressions, is only made possible by the expression model.

## 6.6 Conclusions

We have shown that 3D Morphable Models can be one way to approach challenging real world identification problems. They address in a natural way such difficult problems as combined variations of pose and illumination. Morphable Models can



**Fig. 6.13** Impostor detection is reliable, as the minimum distance to a match is smaller than the minimum distance to a nonmatch. Note the vast increase in recognition performance with the expression model on the expression database, and the fact that the recognition rate is not decreasing on the neutral database, even though we added expression invariance. Already for 0.5% false acceptance rate, we can operate at 0% false rejection rate. false acceptance rate with less than 4% false rejection rate, or less than 0.5% FAR with less than 0.5% FRR



**Fig. 6.14** Use of the expression model improves retrieval performance. Plotted are precision and recall for different retrieval depths. The lower precision of the UND database is due to the fact that some queries have no correct answers. For the UND database, we achieve total recall when querying nine answers, while the maximal number of scans per individual is eight, while for the GavabDB database the expression model gives a strong improvement in recall rate but full recall can not be achieved

be extended, in a straightforward way, to cope with other sources of variation such as facial expression or age.

Our focus was mainly centered on improving the fitting algorithms with respect to accuracy and efficiency. We also investigated several methods for estimating identity from model coefficients. However, a more thorough understanding of the relation between these coefficients and identity might still improve recognition performance. The separation of identity from other attributes could be improved, for instance, by using other features made available by the fitting, such as the texture extracted from the image (after correspondences are recovered by model fitting). Improving this separation might even be more crucial when facial expression or age variation are added to the model.

To model fine and identity-related details such as freckles, birthmarks, and wrinkles, it might be helpful to extend our current framework for representing texture. Indeed, linear combination of textures is a rather simplifying choice. Hence improving the texture model is subject to future research.

Currently our approach is clearly limited by its computational load. However, this disadvantage will evaporate with time as computers increase their clock speed. Adding an automatic landmark detection will enable 3D Morphable Models to compete with state of the art commercial systems such as those that took part in the Face Recognition Vendor Test 2002 [35]. For frontal images, 3D Morphable Model fitting has been completely automated in [13].

## References

- Amberg, B., Romdhani, S., Vetter, T.: Optimal step nonrigid ICP algorithms for surface registration. In: CVPR07, June 2007
- Amberg, B., Knothe, R., Vetter, T.: Expression invariant 3D face recognition with a morphable model. In: Automatic Face and Gesture Recognition (2008)
- Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Pang, H.-C., Davis, J.: The correlated correspondence algorithm for unsupervised surface registration. In: NIPS (2004)
- Beymer, D., Poggio, T.: Image representations for visual learning. *Science* **272**, 1905–1909 (1996)
- Beymer, D., Shashua, A., Poggio, T.: Example based image analysis and synthesis. Technical report, Artificial Intelligence Laboratory, MIT, Cambridge, MA (1993)
- Blanz, V., Vetter, T.: A morphable model for the synthesis of 3D-faces. In: SIGGRAPH 99 (1999)
- Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. *PAMI* (2003)
- Blanz, V., Vetter, T.: Generating frontal views from single, non-frontal images. In: Face Recognition Vendor Test 2002: Technical Appendix O. NISTIR 6965. National Institute of Standards and Technology, Gaithersburg (2003)
- Blanz, V., Romdhani, S., Vetter, T.: Face identification across different poses and illuminations with a 3D morphable model. In: Automatic Face and Gesture Recognition (2002)
- Blanz, V., Basso, C., Poggio, T., Vetter, T.: Reanimating faces in images and video. In: EuroGraphics (2003)
- Blanz, V., Scherbaum, K., Seidel, H.-P.: Fitting a morphable model to 3D scans of faces. In: Proc. of Int. Conf. on Computer Vision ICCV (2007)
- Brand, M.: Morphable 3D models from video. In: CVPR (2001)

13. Breuer, P., Kim, K.I., Kienzle, W., Schölkopf, B., Blanz, V.: Automatic 3D face reconstruction from single images or video. In: Automatic Face and Gesture Recognition (2008)
14. Chang, K.I., Bowyer, K.W., Flynn, P.F.: An evaluation of multimodal 2D+3D face biometrics. In: PAMI (2005)
15. Craw, I., Cameron, P.: Parameterizing images for recognition and reconstruction. In: Proc. BMVC (1991)
16. Deckelnick, K., Dziuk, G., Elliott, C.: Computation of geometric partial differential equations and mean curvature flow. *Acta Numer.* **14**, 139–232 (2005)
17. Dedner, A., Lüthi, M., Albrecht, T., Vetter, T.: Curvature guided level set registration using adaptive finite elements. In: DAGM (2007)
18. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. B* **39**, 1–38 (1977)
19. Duda, R., Hart, P., Stork, D.: Pattern Classification. Wiley, New York (2001)
20. Edwards, G., Taylor, C., Cootes, T.: Interpreting face images using active appearance models. In: Automatic Face and Gesture Recognition (1998)
21. Foley, J., van Dam, A., Feiner, S., Hughes, J.: Computer Graphics: Principles and Practice. Addison-Wesley, Reading (1996)
22. Gu, L., Kanade, T.: 3D alignment of face in a single image. In: CVPR (2006)
23. Hallinan, P.: A deformable model for the recognition of human faces under arbitrary illumination. Ph.D. thesis, Harvard University (1995)
24. Huang, J., Blanz, V., Heisele, B.: Face recognition with support vector machines and 3D head models. In: Pattern Recognition with Support Vector Machines, First International Workshop (2002)
25. Huang, X., Paragios, N., Metaxas, D.N.: Shape registration in implicit spaces using information theory and free form deformations. In: PAMI (2006)
26. Kharevych, L., Springborn, B., Schröder, P.: Discrete conformal mappings via circle patterns. In: SIGGRAPH '05: ACM SIGGRAPH 2005 Courses (2005)
27. Kumar, V., Poggio, T.: Learning-based approach to estimation of morphable model parameters. AI Memo No. 1696 (2000)
28. Lanitis, A., Taylor, C., Cootes, T.: An automatic face identification system using flexible appearance models. In: Proc. British Machine Vision Conference (1994)
29. Leopold, D.A., O'Toole, A.J., Vetter, T., Blanz, V.: Prototype-referenced shape encoding revealed by high-level aftereffects. *Nat. Neurosci.* (2001)
30. Litke, N., Droske, M., Rumpf, M., Schröder, P.: An image processing approach to surface matching. In: Symposium on Geometry Processing (2005)
31. Minka, T.: Old and new matrix algebra useful for statistics (2000). <http://www.stat.cmu.edu/~minka/papers/matrix.html>
32. Moreno, A.B., Sánchez, A.: GavabDB: a 3D face database. In: Workshop on Biometrics on the Internet (2004)
33. Murase, H., Nayar, S.: Visual learning and recognition of 3d objects from appearance. *Int. J. Comput. Vis.* **14**, 5–24 (1995)
34. Paysan, P., Knothe, R., Amberg, B., Romdhani, S., Vetter, T.: A 3D face model for pose and illumination invariant face recognition. In: AVSS (2009)
35. Phillips, P., Grother, P., Michaels, R., Blackburn, D., Tabassi, E., Bone, M.: Face recognition vendor test 2002: evaluation report. In: NISTIR 6965. National Institute of Standards and Technology, Gaithersburg (2003)
36. Phillips, J.P., Scruggs, T.W., O'Toole, A.J., Flynn, P.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: FRVT 2006 and ICE 2006 large-scale results. In: NISTIR 7408 (2007)
37. Press, W., Teukolsky, S., Vetterling, W., Flannery, B.: Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press, Cambridge (1992)
38. Romdhani, S.: Face image analysis using a multiple features fitting strategy. Ph.D. dissertation (2005)
39. Romdhani, S., Vetter, T.: Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In: CVPR (2005)

40. Sirovich, L., Kirby, M.: Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A* **4**, 519–524 (1987)
41. ter Haar, F.B., Veltkamp, R.C.: A 3D face matching framework. In: SMI (2008)
42. Tipping, M., Bishop, C.: Probabilistic principal component analysis. *J. R. Stat. Soc., Ser. B* (1999)
43. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**, 71–86 (1991)
44. Vapnik, V.: The Nature of Statistical Learning. Springer, New York (1995)
45. Vetter, T., Troje, N.: Separation of texture and shape in images of faces for image coding and synthesis. *J. Opt. Soc. Am.* **14**, 2152–2161 (1997)
46. Vlasic, D., Brand, M., Pfister, H., Popović, J.: Face transfer with multilinear models. *ACM Trans. Graph.* (2005)

# Chapter 7

## Illumination Modeling for Face Recognition

Ronen Basri and David Jacobs

### 7.1 Introduction

Changes in lighting can produce large variability in the appearance of faces, as illustrated in Fig. 7.1. Characterizing this variability is fundamental to understanding how to account for the effects of lighting on face recognition. In this chapter, we will discuss solutions to a problem: Given (1) a three-dimensional description of a face, its pose, and its reflectance properties, and (2) a 2D query image, how can we efficiently determine whether lighting conditions exist that can cause this model to produce the query image? We describe methods that solve this problem by producing simple, linear representations of the set of all images a face can produce under all lighting conditions. These results can be directly used in face recognition systems that capture 3D models of all individuals to be recognized. They also have the potential to be used in recognition systems that compare strictly 2D images but that do so using generic knowledge of 3D face shapes.

One way to measure the difficulties presented by lighting, or any variability, is the number of degrees of freedom needed to describe it. For example, the pose of a face relative to the camera has six degrees of freedom—three rotations and three translations. Facial expression has a few tens of degrees of freedom if one considers the number of muscles that may contract to change expression. To describe the light that strikes a face, we must describe the intensity of light hitting each point on the face from each direction. That is, light is a function of position and direction, meaning that light has an infinite number of degrees of freedom. In this chapter, however, we will show that effective systems can account for the effects of lighting

---

Portions of this chapter are reprinted, with permission, from Basri and Jacobs [6], © 2003 IEEE.

R. Basri (✉)

The Weizmann Institute of Science, Rehovot 76100, Israel

e-mail: [ronen.basri@weizmann.ac.il](mailto:ronen.basri@weizmann.ac.il)

D. Jacobs

University of Maryland, College Park, MD 20742, USA

e-mail: [djacobs@umiacs.umd.edu](mailto:djacobs@umiacs.umd.edu)

**Fig. 7.1** Same face under different lighting conditions



using fewer than 10 degrees of freedom. This can have considerable impact on the speed and accuracy of recognition systems.

Support for low-dimensional models is both empirical and theoretical. Principal component analysis (PCA) on images of a face obtained under various lighting conditions shows that this image set is well approximated by a low-dimensional, linear subspace of the space of all images (see, e.g., [19]). Experimentation shows that algorithms that take advantage of this observation can achieve high performance, for example, [17, 21].

In addition, we describe theoretical results that, with some simplified assumptions, prove the validity of low-dimensional, linear approximations to the set of images produced by a face. For these results, we assume that light sources are distant from the face, but we do allow arbitrary combinations of point sources (e.g., the Sun) and diffuse sources (e.g., the sky). We also consider only diffuse components of reflectance, modeled as Lambertian reflectance, and we ignore the effects of cast shadows, such as those produced by the nose. We do, however, model the effects of attached shadows, as when one side of a head faces away from a light. Theoretical predictions from these models provide a good fit to empirical observations and produce useful recognition systems. This suggests that the approximations made capture the most significant effects of lighting on facial appearance. Theoretical models are valuable not only because they provide insight into the role of lighting in face recognition, but also because they lead to analytically derived, low-dimensional, linear representations of the effects of lighting on facial appearance, which in turn can lead to more efficient algorithms.

An alternate stream of work attempts to compensate for lighting effects without the use of 3D face models. This work directly matches 2D images using representations of images that are found to be insensitive to lighting variations. These include image gradients [12], Gabor jets [29], the direction of image gradients [13, 24], and projections to subspaces derived from linear discriminants [8]. A large number of these methods are surveyed in [50]. These methods are certainly of interest, especially for applications in which 3D face models are not available. However, methods based on 3D models may be more powerful, as they have the potential to compensate completely for lighting changes, whereas 2D methods cannot achieve such invariance [1, 13, 35]. Another approach of interest, the Morphable Model, is to use general 3D knowledge of faces to improve methods of image comparison.

## 7.2 Background on Reflectance and Lighting

Throughout this chapter, we consider only distant light sources. By a *distant* light source, we mean that it is valid to make the approximation that a light shines on each point in the scene from the same angle and with the same intensity (this also rules out, for example, slide projectors).

We consider two lighting conditions. A *point* source is described by a single direction, represented by the unit vector  $u_l$ , and intensity,  $l$ . These factors can be combined into a vector with three components,  $\bar{l} = l u_l$ . Lighting may also come from multiple sources, including diffuse sources such as the sky. In that case we can describe the intensity of the light as a function of its direction,  $\ell(u_l)$ , which does not depend on the position in the scene. Light, then, can be thought of as a nonnegative function on the surface of a sphere. This allows us to represent scenes in which light comes from multiple sources, such as a room with a few lamps, and also to represent light that comes from extended sources, such as light from the sky, or light reflected off a wall.

Most of the analysis in this chapter accounts for *attached shadows*, which occur when a point in the scene faces away from a light source. That is, if a scene point has a surface normal  $v_r$ , and light comes from the direction  $u_l$ , when  $u_l \cdot v_r < 0$  none of the light strikes the surface. We also discuss methods of handling *cast shadows*, which occur when one part of a face blocks the light from reaching another part of the face. Cast shadows have been treated by methods based on rendering a model to simulate shadows [18], whereas attached shadows can be accounted for with analytically derived linear subspaces.

Building truly accurate models of the way the face reflects light is a complex task. This is in part because skin is not homogeneous; light striking the face may be reflected by oils or water on the skin, by melanin in the epidermis, or by hemoglobin in the dermis, below the epidermis (see, for example, [2, 3, 33], which discuss these effects and build models of skin reflectance; see also Chap. 6). Based on empirical measurements of skin, Marschner et al. [32] state: “The BRDF itself is quite unusual; at small incidence angles it is almost Lambertian, but at higher angles strong forward scattering emerges.” Furthermore, light entering the skin at one point may scatter below the surface of the skin, and exit from another point. This phenomenon, known as subsurface scattering, cannot be modeled by a bidirectional reflectance function (BRDF), which assumes that light leaves a surface from the point that it strikes it. Jensen et al. [25] presented one model of subsurface scattering.

For purposes of realistic computer graphics, this complexity must be confronted in some way. For example, Borshukov and Lewis [11] reported that in *The Matrix Reloaded*, they began by modeling face reflectance using a Lambertian diffuse component and a modified Phong model to account for a Fresnel-like effect. “As production progressed, it became increasingly clear that realistic skin rendering couldn’t be achieved without subsurface scattering simulations.”

However, simpler models may be adequate for face recognition. They also lead to much simpler, more efficient algorithms. This suggests that even if one wishes to model face reflectance more accurately, simple models may provide useful, approximate algorithms that can initialize more complex ones. In this chapter, we discuss

analytically derived representation of the images produced by a convex, Lambertian object illuminated by distant light sources. We restrict ourselves to convex objects so we can ignore the effect of shadows cast by one part of the object on another part of it. We assume that the surface of the object reflects light according to Lambert's law [30], which states that materials absorb light and reflect it uniformly in all directions. The only parameter of this model is the *albedo* at each point on the object, which describes the fraction of the light reflected at that point.

Specifically, according to Lambert's law, if a light ray of intensity  $l$  coming from the direction  $u_l$  reaches a surface point with albedo  $\rho$  and normal direction  $v_r$ , the intensity  $i$  reflected by the point due to this light is given by

$$i = l(u_l)\rho \max(u_l \cdot v_r, 0). \quad (7.1)$$

If we fix the lighting and ignore  $\rho$  for now, the reflected light is a function of the surface normal alone. We write this function as  $r(\theta_r, \phi_r)$ , or  $r(v_r)$ . If light reaches a point from a multitude of directions, the light reflected by the point would be the integral over the contribution for each direction. If we denote  $k(u \cdot v) = \max(u \cdot v, 0)$ , we can write:

$$r(v_r) = \int_{S^2} k(u_l \cdot v_r) \ell(u_l) du_l \quad (7.2)$$

where  $\int_{S^2}$  denotes integration over the surface of the sphere.

### 7.3 PCA Based Linear Lighting Models

We can consider a face image as a point in a high-dimensional space by treating each pixel as a dimension. Then one can use PCA to determine how well one can approximate a set of face images using a low-dimensional, linear subspace. PCA was first applied to images of faces by Sirovitch and Kirby [44], and used for face recognition by Turk and Pentland [45]. Hallinan [19] used PCA to study the set of images that a single face in a fixed pose produces when illuminated by a floodlight placed in various positions. He found that a five- or six-dimensional subspace accurately models this set of images. Epstein et al. [14] and Yuille et al. [47] described experiments on a wide range of objects that indicate that images of Lambertian objects can be approximated by a linear subspace of between three and seven dimensions. Specifically, the set of images of a basketball were approximated to 94.4% by a 3D space and to 99.1% by a 7D space, whereas the images of a face were approximated to 90.2% by a 3D space and to 95.3% by a 7D space. This work suggests that lighting variation has a low-dimensional effect on face images, although it does not make clear the exact reasons for it.

Because of this low-dimensionality, linear representations based on PCA can be used to compensate for lighting variation. Georghiades et al. [18] used a 3D model of a face to render images with attached or with cast shadows. PCA is used to compress these images to a low-dimensional subspace, in which they are compared

to new images (also using nonnegative lighting constraints we discuss in Sect. 7.5). One issue raised by this approach is that the linear subspace produced depends on the face’s pose. Computing this on-line, when pose is determined, is potentially expensive. Georgiades et al. [17] attacked this problem by sampling pose space and generating a linear subspace for each pose. Ishiyama and Sakamoto [21] instead generated a linear subspace in a model-based coordinate system, so this subspace can be transformed in 3D as the pose varies.

## 7.4 Linear Lighting Models without Shadows

The empirical study of the space occupied by the images of various real objects was to some degree motivated by a previous result that showed that Lambertian objects, in the absence of *all* shadows, produce a set of images that form a three-dimensional linear subspace [34, 40]. To see this, consider a Lambertian object illuminated by a point source described by the vector  $\bar{l}$ . Let  $p_i$  denote a point on the object, let  $n_i$  be a unit vector describing the surface normal at  $p_i$ , let  $\rho_i$  denote the albedo at  $p_i$ , and define  $\bar{n}_i = \rho_i n_i$ . In the absence of attached shadows, Lambertian reflectance is described by  $\bar{l}^T \bar{n}_i$ . If we combine all of an object’s surface normals into a single matrix  $N$ , so the  $i$ th column of  $N$  is  $\bar{n}_i$ , the entire image is described by  $I = \bar{l}^T N$ . This implies that any image is a linear combination of the three rows of  $N$ . These are three vectors consisting of the  $x$ ,  $y$ , and  $z$  components of the object’s surface normals, scaled by albedo. Consequently, all images of an object lie in a three-dimensional space spanned by these three vectors. Note that if we have multiple light sources,  $\bar{l}_1 \dots \bar{l}_d$ , we have

$$I = \sum_i (\bar{l}_i N) = \left( \sum_i \bar{l}_i \right) N \quad (7.3)$$

so this image, too, lies in this three-dimensional subspace. Belhumeur et al. [8] reported face recognition experiments using this 3D linear subspace. They found that this approach partially compensates for lighting variation, but not as well as methods that account for shadows.

Hayakawa [20] used factorization to build 3D models using this linear representation. Koenderink and van Doorn [28] augmented this space to account for an additional, perfect diffuse component. When in addition to a point source there is also an ambient light,  $\ell(u_l)$ , which is constant as a function of direction, and we ignore cast shadows, it has the effect of adding the albedo at each point, scaled by a constant to the image. This leads to a set of images that occupy a four-dimensional linear subspace.

## 7.5 Nonlinear Models with Attached Shadows

Belhumeur and Kriegman [9] conducted an analytic study of the images an object produces when shadows are present. First, they pointed out that for arbitrary illumination, scene geometry, and reflectance properties, the set of images produced by an object forms a convex cone in image space. It is a cone because the intensity of lighting can be scaled by any positive value, creating an image scaled by the same positive value. It is convex because two lighting conditions that create two images can always be added together to produce a new lighting condition that creates an image that is the sum of the original two images. They call this set of images the *illumination cone*.

Then they showed that for a convex, Lambertian object in which there are attached shadows but no cast shadows the dimensionality of the illumination cone is  $O(n^2)$  where  $n$  is the number of distinct surface normals visible on the object. For an object such as a sphere, in which every pixel is produced by a different surface normal, the illumination cone has volume in image space. This proves that the images of even a simple object do not lie in a low-dimensional linear subspace. They noted, however, that simulations indicate that the illumination cone is “thin”; that is, it lies near a low-dimensional image space, which is consistent with the experiments described in Sect. 7.3. They further showed how to construct the cone using the representation of Shashua [40]. Given three images obtained with lighting that produces no attached or cast shadows, they constructed a 3D linear representation, clipped all negative intensities at zero, and took convex combinations of the resulting images.

Georghiades and colleagues [17, 18] presented several algorithms that use the illumination cone for face recognition. The cone can be represented by sampling its extremal rays; this corresponds to rendering the face under a large number of point light sources. An image may be compared to a known face by measuring its distance to the illumination cone, which they showed can be computed using nonnegative least-squares algorithms. This is a convex optimization guaranteed to find a global minimum, but it is slow when applied to a high-dimensional image space. Therefore, they suggested running the algorithm after projecting the query image and the extremal rays to a lower-dimensional subspace using PCA.

Also of interest is the approach of Blicher and Roy [10], which buckets nearby surface normals, and renders a model based on the average intensity of image pixels that have been matched to normals within a bucket. This method assumes that similar normals produce similar intensities (after the intensity is divided by the albedo), so it is suitable for handling attached shadows. It is also extremely fast.

## 7.6 Spherical Harmonic Representations

The empirical evidence showing that for many common objects the illumination cone is “thin” even in the presence of attached shadows has remained unexplained

until recently, when Basri and Jacobs [4, 6], and in parallel Ramamoorthi and Hanrahan [38], analyzed the illumination cone in terms of spherical harmonics. This analysis showed that, when we account for attached shadows, the images of a convex Lambertian object can be approximated to high accuracy using nine (or even fewer) basis images. In addition, this analysis provides explicit expressions for the basis images. These expressions can be used to construct efficient recognition algorithms that handle faces under arbitrary lighting. At the same time these expressions can be used to construct new shape reconstruction algorithms that work under unknown combinations of point and extended light sources. We next review this analysis. Our discussion is based primarily on the work of Basri and Jacobs [6].

### 7.6.1 Spherical Harmonics and the Funk–Hecke Theorem

The key to producing linear lighting models that account for attached shadows lies in noting that (7.2), which describes how lighting is transformed to reflectance, is analogous to a convolution on the surface of a sphere. For every surface normal  $v_r$ , reflectance is determined by integrating the light coming from all directions weighted by the kernel  $k(u_l \cdot v_r) = \max(u_l \cdot v_r, 0)$ . For every  $v_r$  this kernel is just a rotated version of the same function, which contains the positive portion of a cosine function. We denote the (unrotated) function  $k(u_l)$  (defined by fixing  $v_r$  at the north pole) and refer to it as the *half-cosine* function. Note that on the sphere convolution is well defined only when the kernel is rotationally symmetrical about the north pole, which indeed is the case for this kernel.

Just as the Fourier basis is convenient for examining the results of convolutions in the plane, similar tools exist for understanding the results of the analog of convolutions on the sphere. We now introduce these tools, and use them to show that when producing reflectance,  $k$  acts as a low-pass filter.

The *surface spherical harmonics* are a set of functions that form an orthonormal basis for the set of all functions on the surface of the sphere. We denote these functions by  $Y_{nm}$ , with  $n = 0, 1, 2, \dots$  and  $-n \leq m \leq n$ :

$$Y_{nm}(\theta, \phi) = \sqrt{\frac{(2n+1)}{4\pi} \frac{(n-|m|)!}{(n+|m|)!}} P_{n|m|}(\cos \theta) e^{im\phi} \quad (7.4)$$

where  $P_{nm}$  represents the *associated Legendre functions*, defined as

$$P_{nm}(z) = \frac{(1-z^2)^{m/2}}{2^n n!} \frac{d^{n+m}}{dz^{n+m}} (z^2 - 1)^n. \quad (7.5)$$

We say that  $Y_{nm}$  is an  $n$ th *order harmonic*.

It is sometimes convenient to parameterize  $Y_{nm}$  as a function of space coordinates  $(x, y, z)$  rather than angles. The spherical harmonics, written  $Y_{nm}(x, y, z)$ , then be-

come polynomials of degree  $n$  in  $(x, y, z)$ . The first nine harmonics then become

$$\begin{aligned} Y_{00} &= \frac{1}{\sqrt{4\pi}}, & Y_{10} &= \sqrt{\frac{3}{4\pi}}z, \\ Y_{11}^e &= \sqrt{\frac{3}{4\pi}}x, & Y_{11}^o &= \sqrt{\frac{3}{4\pi}}y, \\ Y_{20} &= \frac{1}{2}\sqrt{\frac{5}{4\pi}}(3z^2 - 1), & Y_{21}^e &= 3\sqrt{\frac{5}{12\pi}}xz, \\ Y_{21}^o &= 3\sqrt{\frac{5}{12\pi}}yz, & Y_{22}^e &= \frac{3}{2}\sqrt{\frac{5}{12\pi}}(x^2 - y^2), \\ Y_{22}^o &= 3\sqrt{\frac{5}{12\pi}}xy, \end{aligned} \quad (7.6)$$

where the superscripts e and o denote the even and odd components of the harmonics, respectively (so  $Y_{nm} = Y_{n|m|}^e \pm iY_{n|m|}^o$ , according to the sign of  $m$ ; in fact the even and odd versions of the harmonics are more convenient to use in practice because the reflectance function is real).

Because the spherical harmonics form an orthonormal basis, any piecewise continuous function,  $f$ , on the surface of the sphere can be written as a linear combination of an infinite series of harmonics. Specifically, for any  $f$ ,

$$f(u) = \sum_{n=0}^{\infty} \sum_{m=-n}^n f_{nm} Y_{nm}(u) \quad (7.7)$$

where  $f_{nm}$  is a scalar value, computed as

$$f_{nm} = \int_{S^2} f(u) Y_{nm}^*(u) du \quad (7.8)$$

and  $Y_{nm}^*(u)$  denotes the complex conjugate of  $Y_{nm}(u)$ .

Rotating a function  $f$  results in a phase shift. Define for every  $n$  the  $n$ 'th order amplitude of  $f$  as

$$A_n \stackrel{\text{def}}{=} \sqrt{\frac{1}{2n+1} \sum_{m=-n}^n f_{nm}^2}. \quad (7.9)$$

Then rotating  $f$  does not change the amplitude of a particular order. It may shuffle values of the coefficients,  $f_{nm}$ , for a particular order, but it does not shift energy between harmonics of different orders.

Both the lighting function,  $\ell$ , and the Lambertian kernel,  $k$ , can be written as sums of spherical harmonics. Denote by

$$\ell = \sum_{n=0}^{\infty} \sum_{m=-n}^n l_{nm} Y_{nm} \quad (7.10)$$

the harmonic expansion of  $\ell$ , and by

$$k(u) = \sum_{n=0}^{\infty} k_n Y_{n0}. \quad (7.11)$$

Note that, because  $k(u)$  is circularly symmetrical about the north pole, only the zonal harmonics participate in this expansion, and

$$\int_{S^2} k(u) Y_{nm}^*(u) du = 0, \quad m \neq 0. \quad (7.12)$$

Spherical harmonics are useful for understanding the effect of convolution by  $k$  because of the Funk–Hecke theorem, which is analogous to the convolution theorem. Loosely speaking, the theorem states that we can expand  $\ell$  and  $k$  in terms of spherical harmonics, and then convolving them is equivalent to multiplication of the coefficients of this expansion (see Basri and Jacobs [6] for details).

Following the Funk–Hecke theorem, the harmonic expansion of the reflectance function,  $r$ , can be written as:

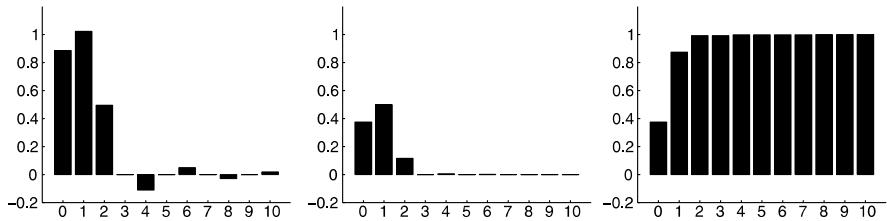
$$r = k * \ell = \sum_{n=0}^{\infty} \sum_{m=-n}^n \left( \sqrt{\frac{4\pi}{2n+1}} k_n l_{nm} \right) Y_{nm}. \quad (7.13)$$

### 7.6.2 Properties of the Convolution Kernel

The Funk–Hecke theorem implies that when producing the reflectance function,  $r$ , the amplitude of the light,  $\ell$ , at every order  $n$  is scaled by a factor that depends only on the convolution kernel,  $k$ . We can use this to infer analytically what frequencies dominate  $r$ . To achieve this, we treat  $\ell$  as a signal and  $k$  as a filter and ask how the amplitudes of  $\ell$  change as it passes through the filter.

The harmonic expansion of the Lambertian kernel (7.11) can be derived [6] yielding

$$k_n = \begin{cases} \frac{\sqrt{\pi}}{2} & n = 0, \\ \sqrt{\frac{\pi}{3}} & n = 1, \\ (-1)^{\frac{n}{2}+1} \frac{\sqrt{(2n+1)\pi}}{2^n(n-1)(n+2)} \binom{n}{\frac{n}{2}} & n \geq 2, \text{ even}, \\ 0 & n \geq 2, \text{ odd}. \end{cases} \quad (7.14)$$



**Fig. 7.2** From left to right: the first 11 coefficients of the Lambertian kernel; the relative energy captured by each of the coefficients; and the cumulative energy

The first few coefficients, for example, are

$$\begin{aligned} k_0 &= \frac{\sqrt{\pi}}{2} \approx 0.8862, & k_1 &= \sqrt{\frac{\pi}{3}} \approx 1.0233, \\ k_2 &= \frac{\sqrt{5\pi}}{8} \approx 0.4954, & k_4 &= -\frac{\sqrt{\pi}}{16} \approx -0.1108, \\ k_6 &= \frac{\sqrt{13\pi}}{128} \approx 0.0499, & k_8 &= \frac{\sqrt{17\pi}}{256} \approx -0.0285 \end{aligned} \quad (7.15)$$

( $k_3 = k_5 = k_7 = 0$ ),  $|k_n|$  approaches zero as  $O(n^{-2})$ . A graphic representation of the coefficients may be seen in Fig. 7.2.

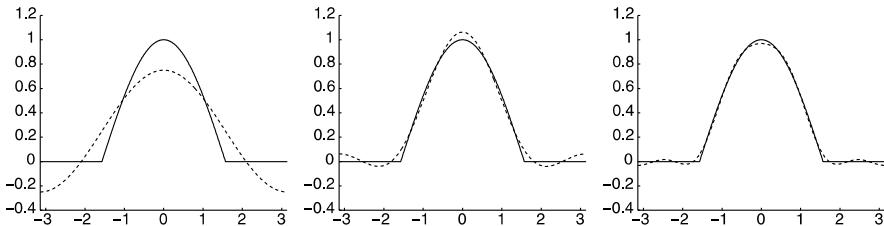
The energy captured by every harmonic term is measured commonly by the square of its respective coefficient divided by the total squared energy of the transformed function. The total squared energy in the half cosine function is given by

$$\int_0^{2\pi} \int_0^\pi k^2(\theta) \sin \theta d\theta d\phi = 2\pi \int_0^{\frac{\pi}{2}} \cos^2 \theta \sin \theta d\theta = \frac{2\pi}{3}. \quad (7.16)$$

(Here, we simplify our computation by integrating over  $\theta$  and  $\phi$  rather than  $u$ . The  $\sin \theta$  factor is needed to account for the varying length of the latitude over the sphere.) Figure 7.2 shows the relative energy captured by each of the first several coefficients. It can be seen that the kernel is dominated by the first three coefficients. Thus, a second-order approximation already accounts for  $(\frac{\pi}{4} + \frac{\pi}{3} + \frac{5\pi}{64})/\frac{2\pi}{3} \approx 99.22\%$  of the energy. With this approximation, the half cosine function can be written as:

$$k(\theta) \approx \frac{3}{32} + \frac{1}{2} \cos \theta + \frac{15}{32} \cos^2 \theta. \quad (7.17)$$

The quality of the approximation improves somewhat with the addition of the fourth order term (99.81%) and deteriorates to 87.5% when a first order approximation is used. Figure 7.3 shows a one-dimensional slice of the Lambertian kernel and its various approximations.



**Fig. 7.3** A slice of the Lambertian kernel (solid line) and its approximations (dashed line) of first (left), second (middle), and fourth (right) order

### 7.6.3 Approximating the Reflectance Function

Because the Lambertian kernel,  $k$ , acts as a low-pass filter, the high frequency components of the lighting have little effect on the reflectance function. This implies that we can approximate the reflectance function that occurs under any lighting conditions using only low-order spherical harmonics. In this section, we show that this leads to an approximation that is always quite accurate.

We achieve a low-dimensional approximation to the reflectance function by truncating the sum in (7.13). That is, we have:

$$r = k * \ell \approx \sum_{n=0}^N \sum_{m=-n}^n \left( \sqrt{\frac{4\pi}{2n+1}} k_n l_{nm} \right) Y_{nm} \quad (7.18)$$

for some choice of order  $N$ . This means considering only the effects of the low order components of the lighting on the reflectance function. Intuitively, we know that because  $k_n$  is small for large  $n$ , this approximation should be good. However, the accuracy of the approximation also depends on  $l_{nm}$ , the harmonic expansion of the lighting.

To evaluate the quality of the approximation, consider first, as an example, lighting,  $\ell = \delta$ , generated by a unit directional (distant point) source at the  $z$  direction ( $\theta = \phi = 0$ ). In this case the lighting is simply a delta function whose peak is at the north pole ( $\theta = \phi = 0$ ). It can be readily shown that

$$r(v) = k * \delta = k(v). \quad (7.19)$$

If the sphere is illuminated by a single directional source in a direction other than the  $z$  direction, the reflectance obtained would be identical to the kernel but shifted in phase. Shifting the phase of a function distributes its energy between the harmonics of the same order  $n$  (varying  $m$ ), but the overall energy in each  $n$  is maintained. The quality of the approximation therefore remains the same, but now for an  $N$ th order approximation we need to use all the harmonics with  $n \leq N$  for all  $m$ . Recall that there are  $2n + 1$  harmonics in every order  $n$ . Consequently, a first-order approximation requires four harmonics. A second-order approximation adds five more harmonics, yielding a 9D space. The third-order harmonics are eliminated by

the kernel, so they do not need to be included. Finally, a fourth order approximation adds nine more harmonics, yielding an 18D space.

We have seen that the energy captured by the first few coefficients  $k_i$  ( $1 \leq i \leq N$ ) directly indicates the accuracy of the approximation of the reflectance function when the light consists of a single point source. Other light configurations may lead to different accuracy. Better approximations are obtained when the light includes enhanced diffuse components of low frequency. Worse approximations are anticipated if the light includes mainly high frequency patterns.

However, even if the light includes mostly high frequency patterns the accuracy of the approximation is still high. This is a consequence of the nonnegativity of light. A lower bound on the accuracy of the approximation for *any* light function is given by

$$\frac{k_0^2}{\frac{2\pi}{3} - \sum_{n=1}^N k_n^2}. \quad (7.20)$$

(Proof appears in Basri and Jacobs [6].)

It can be shown that using a second order approximation (involving nine harmonics) the accuracy of the approximation for any light function exceeds 97.96%. With a fourth order approximation (involving 18 harmonics) the accuracy exceeds 99.48%. Note that the bound computed in (7.20) is not tight, as the case that all the higher order terms are saturated yields a function with negative values. Consequently, the worst case accuracy may even be higher than the bound.

### 7.6.4 Generating Harmonic Reflectances

Constructing a basis to the space that approximates the reflectance functions is straightforward: We can simply use the low order harmonics as a basis (see (7.18)). However, in many cases we want a basis vector for the  $nm$  component of the reflectances to indicate the reflectance produced by a corresponding basis vector describing the lighting,  $Y_{nm}$ . This makes it easy for us to relate reflectances and lighting, which is important when we want to enforce the constraint that the reflectances arise from nonnegative lighting (see Sect. 7.7.1). We call these reflectances *harmonic reflectances* and denote them by  $r_{nm}$ . Using the Funk–Hecke theorem,  $r_{nm}$  is given by

$$r_{nm} = k * Y_{nm} = \left( \sqrt{\frac{4\pi}{2n+1}} k_n \right) Y_{nm}. \quad (7.21)$$

Then, following (7.18),

$$r = k * \ell \approx \sum_{n=0}^N \sum_{m=-n}^n l_{nm} r_{nm}. \quad (7.22)$$

The first few harmonic reflectances are given by

$$\begin{aligned} r_{00} &= \pi Y_{00}, & r_{1m} &= \frac{2\pi}{3} Y_{1m}, & r_{2m} &= \frac{\pi}{4} Y_{2m}, \\ r_{4m} &= \frac{\pi}{24} Y_{4m}, & r_{6m} &= \frac{\pi}{64} Y_{6m}, & r_{8m} &= \frac{\pi}{128} Y_{8m} \end{aligned} \quad (7.23)$$

for  $-n \leq m \leq n$  (and  $r_{3m} = r_{5m} = r_{7m} = 0$ ).

### 7.6.5 From Reflectances to Images

Up to this point, we have analyzed the reflectance functions obtained by illuminating a unit albedo sphere by arbitrary light. Our objective is to use this analysis to represent efficiently the set of images of objects seen under varying illumination. An image of an object under certain illumination conditions can be constructed from the respective reflectance function in a simple way: Each point of the object inherits its intensity from the point on the sphere whose normal is the same. This intensity is further scaled by its albedo.

We can write this explicitly as follows. Let  $p_i$  denote the  $i$ th object point. Let  $n_i$  denote the surface normal at  $p_i$ , and let  $\rho_i$  denote the albedo of  $p_i$ . Let the illumination be expanded with the coefficients  $l_{nm}$  (7.10). Then the image,  $I_i$  of  $p_i$  is

$$I_i = \rho_i r(n_i) \quad (7.24)$$

where

$$r(n_i) = \sum_{n=0}^{\infty} \sum_{m=-n}^n l_{nm} r_{nm}(n_i). \quad (7.25)$$

Then any image is a linear combination of *harmonic images*,  $b_{nm}$ , of the form

$$b_{nm}(p_i) = \rho_i r_{nm}(n_i) \quad (7.26)$$

with

$$I_i = \sum_{n=0}^{\infty} \sum_{m=-n}^n l_{nm} b_{nm}(p_i). \quad (7.27)$$

Figure 7.4 shows the first nine harmonic images derived from a 3D model of a face.

We now discuss how the accuracy of our low dimensional linear approximation to a model's images can be affected by the mapping from the reflectance function to images. The accuracy of our low dimensional linear approximation can vary according to the shape and albedos of the object. Each shape is characterized by a different distribution of surface normals, and this distribution may significantly differ from the distribution of normals on the sphere. Viewing direction also affects



**Fig. 7.4** First nine harmonic images for a model of a face. The *top row* contains the zeroth harmonic (*left*) and the three first order harmonic images (*right*). The *second row* shows the images derived from the second harmonics. Negative values are shown in *black*, positive values in *white*

this distribution, as all normals facing away from the viewer are not visible in the image. Albedo further affects the accuracy of our low dimensional approximation, as it may scale each pixel by a different amount. In the worst case, this can make our approximation arbitrarily poor. For many objects, it is possible to illuminate the object by lighting configurations that produce images for which low order harmonic representations provide a poor approximation.

However, generally, things are not so bad. In general, occlusion renders an arbitrary half of the normals on the unit sphere invisible. Albedo variations and curvature emphasize some normals and deemphasize others. In general, though, the normals whose reflectances are poorly approximated are not emphasized more than any other reflectances, and we can expect our approximation of reflectances on the entire unit sphere to be about as good over those pixels that produce the intensities visible in the image.

The following argument shows that the lower bound on the accuracy of a harmonic approximation to the reflectance function also provides a lower bound on the average accuracy of the harmonic approximation for *any* convex object. (This result was derived by Frolova et al. [15].) We assume that lighting is equally likely from all directions. Given an object, we can construct a matrix  $M$  whose columns contain the images obtained by illuminating the object by a single point source, for all possible source directions. (Of course there are infinitely many such directions, but we can sample them to any desired accuracy.) The average accuracy of a low rank representation of the images of the object then is determined by

$$\min_{M^*} \frac{\|M^* - M\|^2}{\|M\|^2} \quad (7.28)$$

where  $M^*$  is low rank, and  $\|\cdot\|$  denotes the Frobenius Norm of a matrix. Now consider the rows of  $M$ . Each row represents the reflectance of a single surface point under all point sources. Such reflectances are identical to the reflectances of a sphere with uniform albedo under a single point source. (To see this, simply let the surface normal and the lighting directions change roles.) We know that under a point source the reflectance function can be approximated by a combination of the first

nine harmonics to 99.22%. Because by this argument every row of  $M$  can be approximated to the same accuracy, there exists a rank nine matrix  $M^*$  that approximates  $M$  to 99.22%. This argument can be applied to convex objects of any shape. Thus, on average, nine harmonic images approximate the images of an object by at least 99.22%, and likewise four harmonic images approximate the images of an object by at least 87.5%. Note that this approximation can even be improved somewhat by selecting optimal coefficients to better fit the images of the object. Indeed, simulations indicate that optimal selection of the coefficients often increases the accuracy of the second order approximation up to 99.5% and that of the first order approximation to about 95%.

Ramamoorthi [37] further derived expressions to calculate the accuracies obtained with spherical harmonics for orders less than nine. His analysis, in fact, demonstrates that generically the spherical harmonics of the same order are not equally significant. The reason is that the basis images of an object are not generally orthogonal, and in some cases are quite similar. For example, if the  $z$  components of the surface normals of an object do not vary much, some of the harmonic images are quite similar, such as  $b_{00} = \rho$  versus  $b_{10} = \rho z$ . Ramamoorthi's calculations show a good fit (with a slight overshoot) to the empirical results. With his derivations, the accuracy obtained for a 3D representation of a human face is 92% (in contrast to 90.2% in empirical studies) and for 7D 99% (in contrast to 95.3%). The somewhat lower accuracies obtained in empirical studies may be attributed to the presence of specularities, cast shadows, and noisy measurements.

Finally, it is interesting to compare the basis images determined by our spherical harmonic representation with the basis images derived for the case of no shadows. As mentioned in Sect. 7.4, Shashua [40] and Moses [34] pointed out that in the absence of attached shadows every possible image of an object is a linear combination of the  $x$ ,  $y$ , and  $z$  components of the surface normals scaled by the albedo. They therefore proposed using these three components to produce a 3D linear subspace to represent a model's images. Interestingly, these three vectors are identical, up to a scale factor, to the basis images produced by the first-order harmonics in our method.

We can therefore interpret Shashua's method as also making an analytic approximation to a model's images using low-order harmonics. However, our previous analysis tells us that the images of the first harmonic account for only 50% of the energy passed by the half-cosine kernel. Furthermore, in the worst case it is possible for the lighting to contain *no* component in the first harmonic. Most notably, Shashua's method does not make use of the zeroth harmonic (commonly referred to as the DC component). These are the images produced by a perfectly diffuse light source. Nonnegative lighting must always have a significant DC component. We noted in Sect. 7.4 that Koenderink and van Doorn [28] suggested augmenting Shashua's method with this diffuse component. This results in a linear method that uses the four most significant harmonic basis images, although Koenderink and van Doorn proposed it as apparently a heuristic suggestion, without analysis or reference to a harmonic representation of lighting.

## 7.7 Applications

We have developed an analytic description of the linear subspace that lies near the set of images an object can produce. We now show how to use this description in various tasks, including object recognition and shape reconstruction. We begin by describing methods for recognizing faces under different illuminations and poses. Later, we briefly describe reconstruction algorithms for stationary and moving objects.

### 7.7.1 Recognition

In a typical recognition problem, the 3D shape and reflectance properties (including surface normals and albedos) of faces may be available. The task then is, given an image of a face seen under unknown pose and illumination, to recognize the individual. Our spherical harmonic representation enables us to perform this task while accounting for complicated, unknown lighting that includes combinations of point and extended sources. Below, we assume that the pose of the object is already known but that its identity and lighting conditions are not. For example, we may wish to identify a face that is known to be facing the camera; or we may assume that either a human or an automatic system has identified features, such as the eyes and the tip of the nose, that allow us to determine the pose for each face in the database, but that the database is too large to allow a human to select the best match.

Recognition proceeds by comparing a new query image to each model in turn. To compare to a model, we compute the distance between the query image and the nearest image the model can produce. We present two classes of algorithms that vary in their representation of a model's images. The linear subspace can be used directly for recognition, or we can restrict ourselves to a subset of the linear subspace that corresponds to physically realizable lighting conditions.

We stress the advantages we gain by having an *analytic* description of the subspace available, in contrast to previous methods in which PCA could be used to derive a subspace from a sample of an object's images. One advantage of an analytic description is that we know it provides an accurate representation of an object's possible images, not subject to the vagaries of a particular sample of images. A second advantage is efficiency; we can produce a description of this subspace much more rapidly than PCA would allow. The importance of this advantage depends on the type of recognition problem we tackle. In particular, we are interested in recognition problems in which the position of an object is not known in advance but can be computed at run-time using feature correspondences. In this case, the linear subspace must also be computed at run-time, and the cost of doing this is important.

#### 7.7.1.1 Linear Methods

The most straightforward way to use our prior results for recognition is to compare a novel image to the linear subspace of images that correspond to a model, as derived

by our harmonic representation. To do this, we produce the harmonic basis images of each model, as described in Sect. 7.6.5. Given an image  $I$  we seek the distance from  $I$  to the space spanned by the basis images. Let  $B$  denote the basis images. Then we seek a vector  $a$  that minimizes  $\|Ba - I\|$ .  $B$  is  $p \times r$ ,  $p$  is the number of points in the image, and  $r$  is the number of basis images used. As discussed above, nine is a natural value to use for  $r$ , but  $r = 4$  provides greater efficiency and  $r = 18$  offers even better potential accuracy. Every column of  $B$  contains one harmonic image  $b_{nm}$ . These images form a basis for the linear subspace, though not an orthonormal one. Hence we apply a QR decomposition to  $B$  to obtain such a basis. We compute  $Q$ , a  $p \times r$  matrix with orthonormal columns, and  $R$ , an  $r \times r$  matrix so that  $QR = B$  and  $Q^T Q$  is an  $r \times r$  identity matrix. Then  $Q$  is an orthonormal basis for  $B$ , and  $Q^T Q I$  is the projection of  $I$  into the space spanned by  $B$ . We can then compute the distance from the image,  $I$ , and the space spanned by  $B$  as  $\|QQ^T I - I\|$ . The cost of the  $QR$  decomposition is  $O(pr^2)$ , assuming  $p \gg r$ .

The use of an analytically derived basis can have a substantial effect on the speed of the recognition process. In previous work Georgiades et al. [17] performed recognition by rendering the images of an object under many possible lightings and finding an 11D subspace that approximates these images. With our method this expensive rendering step is unnecessary. When  $s$  sampled images are used (typically  $s \gg r$ ), with  $s \ll p$  PCA requires  $O(ps^2)$ . Also, in MATLAB, PCA of a thin, rectangular matrix seems to take exactly twice as long as its QR decomposition. Therefore, in practice, PCA on the matrix constructed by Georgiades et al. would take about 150 times as long as using our method to build a 9D linear approximation to a model's images. (This is for  $s = 100$  and  $r = 9$ . One might expect  $p$  to be about 10 000, but this does not affect the relative costs of the methods.) This may not be significant if pose is known ahead of time and this computation takes place off line. When pose is computed at run time, however, the advantages of our method can become significant.

### 7.7.1.2 Enforcing Nonnegative Light

When we take arbitrary linear combinations of the harmonic basis images, we may obtain images that are not physically realizable. This is because the corresponding linear combination of the harmonics representing lighting may contain negative values. That is, rendering these images may require negative “light,” which of course is physically impossible. In this section, we show how to use the basis images while enforcing the constraint of nonnegative light.

When we use a 9D approximation to an object's images, we can efficiently enforce the nonnegative lighting constraint in a manner similar to that proposed by Belhumeur and Kriegman [9], after projecting everything into the appropriate 9D linear subspace. Specifically, we approximate any arbitrary lighting function as a nonnegative combination of a fixed set of directional light sources. We solve for the best such approximation by fitting to the query image a nonnegative combination of images each produced by a single, directional source.

We can do this efficiently using the 9D subspace that represents an object's images. We project into this subspace a large number of images of the object, in which each image is produced by a single directional light source. Such a light source is represented as a delta function; we can derive the representation of the resulting image in the harmonic basis simply by taking the harmonic transform of the delta function that represents the lighting. Then we can also project a query image into this 9D subspace and find the nonnegative linear combination of directionally lit images that best approximate the query image. Finding the nonnegative combination of vectors that best fit a new vector is a standard, convex optimization problem. We can solve it efficiently because we have projected all the images into a space that is only 9D.

Note that this method is similar to that presented in Georghiades et al. [18]. The primary difference is that we work in a low dimensional space constructed for each model using its harmonic basis images. Georghiades et al. performed a similar computation after projecting all images into a 100-dimensional space constructed using PCA on images rendered from models in a 10-model database. Also, we do not need to explicitly render images using a point source and project them into a low-dimensional space. In our representation, the projection of these images is given in closed form by the spherical harmonics.

A further simplification can be obtained if the set of images of an object is approximated only up to first order. Four harmonics are required in this case. One is the DC component, representing the appearance of the object under uniform ambient light, and three are the basis images also used by Shashua. In this case, we can reduce the resulting optimization problem to one of finding the roots of a sixth degree polynomial in a single variable, which is extremely efficient. Further details of both methods can be found elsewhere [6].

The approach of enforcing nonnegative lighting for 9 harmonics relies on representing lighting as the nonnegative sum of a large number of delta functions. In this way, the nonnegativity of the lighting follows from the nonnegativity of the coefficients of the delta functions. However, in recent work, Shirdhonkar and Jacobs [41] have shown that nonnegativity can be enforced when representing lighting using low frequency spherical harmonics. To do this, one must be able to determine whether a set of low frequency spherical harmonics are consistent with a nonnegative function; that is, could one add higher frequency harmonics to make the complete function nonnegative. By extending Szego's eigenvalue distribution theorem to spherical harmonics, Shirdhonkar and Jacobs show that a matrix constructed using the coefficients of low frequency lighting, represented as spherical harmonics, must be positive semi-definite in order for these harmonics to be consistent with non-negative lighting. This allows them to compute the low frequency lighting that best matches a 3D model to an image by solving a semi-definite programming problem. This leads to solutions that are more accurate and efficient than previous methods that represent lighting using delta functions.

### 7.7.1.3 Specularity

Other work has built on this spherical harmonic representation to account for non-Lambertian reflectance [36]. The method first computes Lambertian reflectance, which constrains the possible location of a dominant compact source of light. Then it extracts highlight candidates as pixels that are brighter than we can predict from Lambertian reflectance. Next, we determine which of these candidates is consistent with a known 3D object. A general model of specular reflectance is used that implies that the surface normals of specular points obtained by thresholding intensity form a disk on the Gaussian sphere. Therefore, the method proceeds by selecting candidate specularities consistent with such a disk. It maps each candidate specularity to the point on the sphere having the same surface normal. Next, a plane is found that separates the specular pixels from the other pixels with a minimal number of misclassifications. The presence of specular reflections that are consistent with the object's known 3D structure then serves as a cue that the model and image match.

This method has succeeded in recognizing shiny objects, such as pottery. However, informal face recognition experiments with this method, using the data set described in the next section, have not shown significant improvements. Our sense is that most of our recognition errors are due to misalignments in pose, and that when a good alignment is found between a 3D model and image a Lambertian model is sufficient to produce good performance on a data set of 42 individuals.

In other work, Georgiades [16] augmented the recognition approach of Georgiades et al. [17] to include specular reflectance. After initialization using a Lambertian model, the position of a single light source and parameters of the Torrance-Sparrow model of specular reflectance are optimized to fit a 3D model of an individual. Face recognition experiments with a data set of 10 individuals show that this produces a reduction in overall errors from 2.96% to 2.47%. It seems probable that experiments with data sets containing large numbers of individuals are needed to truly gauge the value of methods that account for specular reflectance.

### 7.7.1.4 Experiments

We have experimented with these recognition methods using a database of faces collected at NEC in Japan. The database contains models of 42 faces, each including the 3D shape of the face (acquired using a structured light system) and estimates of the albedos in the red, green, and blue color channels. As query images, we use 42 images each of 10 individuals taken across seven poses and six lighting conditions (shown in Fig. 7.5). In our experiment, each of the query images is compared to each of the 42 models, and then the best matching model is selected.

In all methods, we first obtain a 3D alignment between the model and the image using the algorithm of Blicher and Roy [10]. In brief, a dozen or fewer features on the faces were identified by hand, and then a 3D rigid transformation was found to align the 3D features with the corresponding 2D image features.

In all methods, we only pay attention to image pixels that have been matched to some point in the 3D model of the face. We also ignore image pixels that are



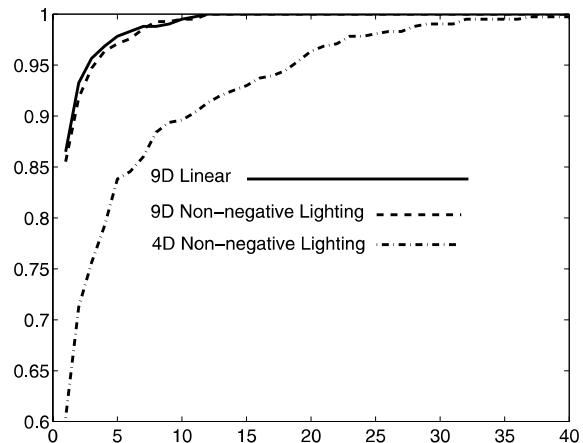
**Fig. 7.5** Test images used in the experiments

of maximum intensity, as they may be saturated and provide misleading values. Finally, we subsample both the model and the image, replacing each  $m \times m$  square with its average values. Preliminary experiments indicate that we can subsample quite a bit without significantly reducing accuracy. In the experiments below, we ran all algorithms subsampling with  $16 \times 16$  squares, while the original images were  $640 \times 480$ .

Our methods produce coefficients that tell us how to combine the harmonic images linearly to produce the rendered image. These coefficients were computed on the sampled image but then applied to harmonic images of the full, unsampled image. This process was repeated separately for each color channel. Then a model was compared to the image by taking the root mean squared error derived from the distance between the rendered face model and all corresponding pixels in the image.

Figure 7.6 shows performance curves for three recognition methods: the 9D linear method and the methods that enforce positive lighting in 9D and 4D. The curves show the fraction of query images for which the correct model is classified among the top  $k$ , as  $k$  varies from 1 to 40. The 4D positive lighting method performs significantly less well than the others, getting the correct answer about 60% of the time. However, it is much faster and seems to be quite effective under simpler pose and lighting conditions. The 9D linear method and 9D positive lighting method each pick the correct model first 86% of the time. With this data set, the difference between these two algorithms is quite small compared to other sources of error. Such errors may include limitations in our model for handling cast shadows and specularities, but they also include errors in the model building and pose determination processes. In fact, on examining our results, we found that one pose (for one person)

**Fig. 7.6** Performance curves for our recognition methods. The vertical axis shows the percentage of times the correct model was found among the  $k$  best matching models; the horizontal axis shows  $k$



was grossly wrong because a human operator selected feature points in the wrong order. We eliminated from our results the six images (under six lighting conditions) that used this pose.

### 7.7.2 Modeling

The recognition methods described in the previous section require detailed 3D models of faces, as well as their albedos. Such models can be acquired in various ways. For example, in the experiments described above we used a laser scanner to recover the 3D shape of a face, and we estimated the albedos from an image taken under ambient lighting (which was approximated by averaging several images of a face). As an alternative, it is possible to recover the shape of a face from images illuminated by structured light or by using stereo reconstruction, although stereo algorithms may give somewhat inaccurate reconstructions for nontextured surfaces. Finally, other studies have developed reconstruction methods that use the harmonic formulation to recover both the shape and the albedo of an object simultaneously. In the remainder of this section, we briefly describe three such methods. We first describe how to recover the shape of an object when the input images are obtained with a stationary object illuminated by variable lighting, a problem commonly referred to as “photometric stereo.” Later, we discuss an approach for shape recovery of a moving object. We conclude with an approach that can recover the shape of faces from single images by exploiting prior knowledge of the generic shape of faces.

#### 7.7.2.1 Photometric Stereo

In photometric stereo, we are given a collection of images of a stationary object under varying illumination. Our objective is to recover the 3D shape of the object

and its reflectance properties, which for a Lambertian object include the albedo at every surface point. Previous approaches to photometric stereo under unknown lighting generally assume that in every image the object is illuminated by a dominant point source for example, [20, 28, 47]. However, by using spherical harmonic representations it is possible to reconstruct the shape and albedo of an object under unknown lighting configurations that include arbitrary collections of point and extended sources. In this section, we summarize this work, which is described in more detail elsewhere [5, 7].

We begin by stacking the input images into a matrix  $M$  of size  $f \times p$ , in which every input image of  $p$  pixels occupies a single row, and  $f$  denotes the number of images in our collection. The low dimensional harmonic approximation then implies that there exist two matrices,  $L$  and  $S$ , of sizes  $f \times r$  and  $r \times p$  respectively, that satisfy

$$M \approx LS \quad (7.29)$$

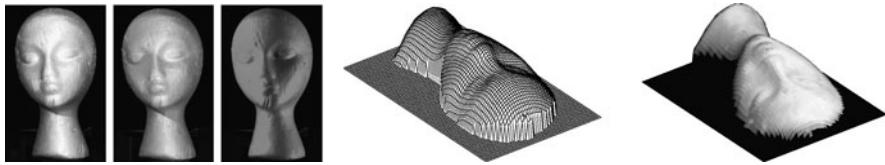
where  $L$  represents the lighting coefficients,  $S$  is the harmonic basis, and  $r$  is the dimension used in the approximation (usually 4 or 9). If indeed we can recover  $L$  and  $S$ , obtaining the surface normals and albedos of the shape is straightforward using (7.23) and (7.26).

We can attempt to recover  $L$  and  $S$  using singular value decomposition (SVD). This produces a factorization of  $M$  into two matrices  $\tilde{L}$  and  $\tilde{S}$ , which are related to the correct lighting and shape matrices by an unknown, arbitrary  $r \times r$  ambiguity matrix  $A$ . We can try to reduce this ambiguity. Consider the case that we use a first-order harmonic approximation ( $r = 4$ ). Omitting unnecessary scale factors, the zero-order harmonic contains the albedo at every point, and the three first-order harmonics contain the surface normal scaled by the albedo. For a given point we can write these four components in a vector:  $p = (\rho, \rho n_x, \rho n_y, \rho n_z)^T$ . Then  $p$  should satisfy  $p^T J p = 0$ , where  $J = \text{diag}\{-1, 1, 1, 1\}$ . Enforcing this constraint reduces the ambiguity matrix from 16 degrees of freedom to just 7. Further resolution of the ambiguity matrix requires additional constraints, which can be obtained by specifying a few surface normals or by enforcing integrability.

A similar technique can be applied in the case of a second order harmonic approximation ( $r = 9$ ). In this case, there are many more constraints on the nine basis vectors, and they can be satisfied by applying an iterative procedure. Using the nine harmonics, the surface normals can be recovered up to a rotation, and further constraints are required to resolve the remaining ambiguity.

An application of these photometric stereo methods is demonstrated in Fig. 7.7. A collection of 32 images of a statue of a face illuminated by two point sources in each image were used to reconstruct the 3D shape of the statue. (The images were simulated by averaging pairs of images obtained with single light sources taken by researchers at Yale.) Saturated pixels were removed from the images and filled in using Wiberg's algorithm [46]; see also [23, 42]. We resolved the remaining ambiguity by matching some points in the scene with hand-chosen surface normals.

Photometric stereo is one way to produce a 3D model for face recognition. An alternative approach is to determine a discrete set of lighting directions that produce a set of images that span the 9D set of harmonic images of an object. In this



**Fig. 7.7** *Left:* three images of a bust illuminated each by two point sources. *Right:* the surface produced by the 4D method (a mesh, and painted with albedo). From Basri, Jacobs, and Kemelmacher [7], © 2007 Springer, with permission

way, the harmonic basis can be constructed directly from images, without building a 3D model. This problem was addressed by Lee et al. [31] and by Sato et al. [39]. Other approaches use harmonic representations to cluster the images of a face under varying illumination [22] or determine the harmonic images of a face from just one image using a statistical model derived from a set of 3D models of other faces [49].

### 7.7.2.2 Objects in Motion

Photometric stereo methods require a still object while the lighting varies. For faces, this requires a cooperative subject and controlled lighting. An alternative approach is to use video of a moving face. Such an approach, presented by Simakov et al. [43], is briefly described below.

We assume that the motion of a face is known, for example, by tracking a few feature points such as the eyes and the tips of the mouth. Thus, we know the epipolar constraints between the images and (in case the cameras are calibrated) also the mapping from 3D to each of the images. To obtain a dense shape reconstruction, we need to find correspondences between points in all images. Unlike stereo, in which we can expect corresponding points to maintain approximately the same intensity, in the case of a moving object we expect points to change their intensity as they turn away from or toward light sources.

We therefore adopt the following strategy. For every point in 3D, we associate a “correspondence measure,” which indicates if its projections in all the images could come from the same surface point. To this end, we collect all the projections and compute the residual of the following set of equations.

$$I_j = \rho l^T R_j Y(n). \quad (7.30)$$

In this equation,  $1 \leq j \leq f$ ,  $f$  is the number of images,  $I_j$  denotes the intensity of the projection of the 3D point in the  $j$ th image,  $\rho$  is the unknown albedo,  $l$  denotes the unknown lighting coefficients,  $R_j$  denotes the rotation of the object in the  $j$ th image, and  $Y(n)$  denotes the spherical harmonics evaluated for the unknown surface normal. Thus, to compute the residual we need to find  $l$  and  $n$  that minimize the difference between the two sides of this equation. (Note that for a single 3D point  $\rho$  and  $l$  can be combined to produce a single vector.)

Once we have computed the correspondence measure for each 3D point, we can incorporate the measure in any stereo algorithm to extract the surface that minimizes the measure, possibly subject to some smoothness constraints.

The algorithm of Simakov et al. [43] described above assumes that the motion between the images is known. Zhang et al. [48] proposed an iterative algorithm that simultaneously recovers the motion assuming infinitesimal motion between images and modeling reflectance using a first order harmonic approximation.

### 7.7.2.3 Reconstruction with Shape Prior

While the previous methods utilize collections of images to achieve 3D reconstruction, it is of interest to explore methods that can recover the shape of faces from just a single image. Recently, Kemelmacher-Shlizerman and Basri [26, 27] proposed such an approach that exploits prior knowledge of the rough shape of faces to make the problem of single view reconstruction well-posed.

The algorithm obtains as input an image of a face to be reconstructed along with a 3D model (shape and albedo) of some different face. Such a model can depict an individual whose 3D shape is available, or an “averaged” model of a collection of faces. The algorithm then attempts to reconstruct the shape of the face in the input image essentially by solving a shape from shading (SFS) problem. However, while SFS is ill-posed and its solution requires knowledge of the lighting conditions, the reflectance properties (albedo) of the object to be reconstructed, and boundary conditions (i.e., depth values at extremal points), this algorithm estimates their values by exploiting the similarity of the input model to the desired shape.

Specifically, Kemelmacher-Shlizerman and Basri seek a solution to the following optimization problem:

$$\min_{l, \rho, z} \int_{\Omega} (I - \rho l^T Y(n))^2 + (\lambda_1 \Delta_z^2 + \lambda_2 \Delta_\rho^2) dx dy. \quad (7.31)$$

In this expression,  $I(x, y)$  is the input image ( $x, y \in \Omega$ ),  $l$  represents the unknown lighting conditions,  $\rho(x, y)$  the unknown albedo,  $z(x, y)$  the unknown depth, and  $Y(n)$  the spherical harmonic basis derived from  $z$ . The first term therefore is a data term fitting the desired reconstruction to the image. For the second term,  $\lambda_1$  and  $\lambda_2$  are preset constants and we define  $\Delta_z(x, y)$  and  $\Delta_\rho(x, y)$  to represent respectively, the (smoothed) difference in shape and albedo between the desired shape and the input model. The role of this regularization term is to keep those differences small. Figure 7.8 shows a reconstruction obtained with this method.

## 7.8 Conclusions

Lighting can be arbitrarily complex, but in many cases its effect is not. When objects are Lambertian, we show that a simple, 9D linear subspace can capture the set of



**Fig. 7.8** Single view reconstruction. The figure shows two triplets of images; each includes an input image, 3D reconstruction (output), and the input image overlaid on the reconstruction. The reference shape used in these runs is shown on the right. Notice that veridical shape is recovered despite change in expression relative to the reference shape. From Kemelmacher-Shlizerman and Basri [27], © 2010 IEEE, with permission

images they produce. This explains prior empirical results. It also gives us a new and effective way to understand the effects of Lambertian reflectance as that of a low-pass filter on lighting.

Moreover, we show that this 9D space can be directly computed from a model, as low-degree polynomial functions of its scaled surface normals. This description allows us to produce efficient recognition algorithms in which we know we are using an accurate approximation of the model's images. In addition, we can use the harmonic formulation to develop reconstruction algorithms to recover the 3D shape and albedos of an object. We evaluate the effectiveness of our recognition algorithms using a database of models and images of real faces.

**Acknowledgements** Major portions of this research were conducted while Ronen Basri and David Jacobs were at the NEC Research Institute, Princeton, NJ. At the Weizmann Institute Ronen Basri is supported in part by European Community grants IST-2000-26001 VIBES and IST-2002-506766 Aim Shape and by the Israel Science Foundation grant 266/02. The vision group at the Weizmann Institute is supported in part by the Moross Foundation. David Jacobs was funded by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), through the Army Research Laboratory (ARL). All statements of fact, opinion or conclusions contained herein are those of the authors and should not be construed as representing the official views or policies of IARPA, the ODNI, or the U.S. Government.

## References

1. Adini, Y., Moses, Y., Ullman, S.: Face recognition: The problem of compensating for changes in illumination direction. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 721–732 (1997)
2. Angelopoulou, E.: Understanding the color of human skin. In: SPIE Conf. on Human Vision and Electronic Imaging VI, vol. 4299, pp. 243–251. SPIE, Bellingham (2001)
3. Angelopoulou, E., Molana, R., Daniilidis, K.: Multispectral skin color modeling. In: IEEE Conf. on Computer Vision and Patt. Recognition, pp. 635–642 (2001)
4. Basri, R., Jacobs, D.: Lambertian reflectances and linear subspaces. In: IEEE Int. Conf. on Computer Vision, vol. II, pp. 383–390 (2001)
5. Basri, R., Jacobs, D.: Photometric stereo with general, unknown lighting. In: IEEE Conf. on Computer Vision and Pattern Recognition, vol. II, pp. 374–381 (2001)
6. Basri, R., Jacobs, D.: Lambertian reflectances and linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(2), 218–233 (2003)
7. Basri, R., Jacobs, D., Kemelmacher, I.: Photometric stereo with general, unknown lighting. *Int. J. Comput. Vis.* **72**(3), 239–257 (2007)

8. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 711–720 (1997)
9. Belhumeur, P., Kriegman, D.: What is the set of images of an object under all possible lighting conditions? *Int. J. Comput. Vis.* **28**(3), 245–260 (1998)
10. Blicher, A., Roy, S.: Fast lighting/rendering solution for matching a 2d image to a database of 3d models: ‘lightsphere’. *IEICE Trans. Inf. Syst.* **E84-D**(12), 1722–1727 (2001)
11. Borshukov, G., Lewis, J.: Realistic human face rendering for ‘the matrix reloaded’. In: SIGGRAPH-2003 Sketches and Applications Program (2003)
12. Brunelli, R., Poggio, T.: Face recognition: Features versus templates. *IEEE Trans. Pattern Anal. Mach. Intell.* **15**(10), 1042–1062 (1993)
13. Chen, H., Belhumeur, P., Jacobs, D.: In search of illumination invariants. In: IEEE Proc. Computer Vision and Pattern Recognition, vol. I, pp. 254–261 (2000)
14. Epstein, R., Hallinan, P., Yuille, A.: *pm2 eigenimages suffice: an empirical investigation of low-dimensional lighting models*. In: IEEE Workshop on Physics-Based Vision, pp. 108–116 (1995)
15. Frolova, D., Simakov, D., Basri, R.: Accuracy of spherical harmonic approximations for images of Lambertian objects under far and near lighting. In: ECCV, pp. 574–587 (2004)
16. Georghiades, A.: Incorporating the Torrance and Sparrow model of reflectance in uncalibrated photometric stereo. In: International Conference on Computer Vision, vol. II, pp. 816–823 (2003)
17. Georghiades, A., Belhumeur, P., Kriegman, D.: From few to many: generative models for recognition under variable pose and illumination. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(6), 643–660 (2001)
18. Georghiades, A., Kriegman, D., Belhumeur, P.: Illumination cones for recognition under variable lighting: faces. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 52–59 (1998)
19. Hallinan, P.: A low-dimensional representation of human faces for arbitrary lighting conditions. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 995–999 (1994)
20. Hayakawa, H.: Photometric stereo under a light source with arbitrary motion. *J. Opt. Soc. Am.* **11**(11), 3079–3089 (1994)
21. Ishiyama, R., Sakamoto, S.: Geodesic illumination basis: compensating for illumination variations in any pose for face recognition. In: IEEE Int. Conf. on Pattern Recognition, vol. 4, pp. 297–301 (2002)
22. Ho, J., Yang, M., Lim, J., Lee, K., Kriegman, D.: Clustering appearances of objects under varying illumination conditions. In: IEEE Conf. on Computer Vision and Pattern Recognition, vol. 1, pp. 11–18 (2003)
23. Jacobs, D.: Linear fitting with missing data for structure-from-motion. *Comput. Vis. Image Underst.* **82**(1), 57–81 (2001)
24. Jacobs, D., Belhumeur, P., Basri, R.: Comparing images under variable illumination. In: IEEE Proc. Computer Vision and Pattern Recognition, pp. 610–617 (1998)
25. Jensen, H., Marschner, S., Levoy, M., Hanrahan, P.: A practical model for subsurface light transport. In: Proc. SIGGRAPH, pp. 511–518 (2001)
26. Kemelmacher, I., Basri, R.: Molding face shapes by example. In: European Conf. on Computer Vision. LNCS, vol. 3951, pp. 277–288 (2006)
27. Kemelmacher-Shlizerman, I., Basri, R.: 3d face reconstruction from a single image using a single reference face shape. *IEEE Trans. Pattern Anal. Mach. Intell.* (forthcoming)
28. Koenderink, J., Doorn, A.V.: The generic bilinear calibration-estimation problem. *Int. J. Comput. Vis.* **23**(3), 217–234 (1997)
29. Lades, M., Vorbrüggen, J., Buhmann, J., Lange, J., von der Malsburg, C., Wurtz, R., Konen, W.: Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. Comput.* **42**(3), 300–311 (1993)
30. Lambert, J.: *Photometria sive de mensura et gradibus luminis, colorum et umbrae*. Eberhard Klett (1760)

31. Lee, K., Ho, J., Kriegman, D.: Nine points of light: acquiring subspaces for face recognition under variable lighting. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 519–526 (2001)
32. Marschner, S., Westin, S., Lafortune, E., Torrance, K., Greenberg, D.: Image-based brdf measurement including human skin. In: 10th Eurographics Workshop on Rendering, pp. 131–144 (1999)
33. Meglinski, I., Matcher, S.: Quantitative assessment of skin layers absorption and skin reflectance spectra simulation in the visible and near-infrared spectral regions. *Physiol. Meas.* **23**, 741–753 (2002)
34. Moses, Y.: Face recognition: generalization to novel images. Ph.D. thesis, Weizmann Institute of Science (1993)
35. Moses, Y., Ullman, S.: Limitations of non model-based recognition schemes. In: Second European Conference on Computer Vision, pp. 820–828 (1992)
36. Osadchy, M., Jacobs, D., Ramamoorthi, R.: Using specularities for recognition. In: International Conference on Computer Vision, vol. II, pp. 1512–1519 (2003)
37. Ramamoorthi, R.: Analytic pca construction for theoretical analysis of lighting variability in a single image of a Lambertian object. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **24**(10) (2002)
38. Ramamoorthi, R., Hanrahan, P.: On the relationship between radiance and irradiance: determining the illumination from images of convex Lambertian object. *J. Opt. Soc. Am.* **18**(10), 2448–2459 (2001)
39. Sato, I., Okabe, T., Sato, Y., Ikeuchi, K.: Appearance sampling for obtaining a set of basis images for variable illumination. In: IEEE Int. Conf. on Computer Vision, vol. II, pp. 800–807 (2003)
40. Shashua, A.: On photometric issues in 3d visual recognition from a single 2d image. *Int. J. Comput. Vis.* **21**(1–2), 99–122 (1997)
41. Shirdhonkar, S., Jacobs, D.: Non-negative lighting and specular object recognition. In: IEEE International Conference on Computer Vision, vol. II, pp. 1323–1330 (2005)
42. Shum, H., Ikeuchi, K., Reddy, R.: Principal component analysis with missing data and its application to polyhedral object modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **17**(9), 854–867 (1995)
43. Simakov, D., Frolova, D., Basri, R.: Dense shape reconstruction of a moving object under arbitrary, unknown lighting. In: IEEE Int. Conf. on Computer Vision, pp. 1202–1209 (2003)
44. Sirovitch, L., Kirby, M.: Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am.* **2**, 586–591 (1987)
45. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–96 (1991)
46. Wiberg, T.: Computation of principal components when data are missing. In: Proc. Second Symp. Computational Statistics, pp. 229–236 (1976)
47. Yuille, A., Snow, D., Epstein, R., Belhumeur, P.: Determining generative models of objects under varying illumination: shape and albedo from multiple images using svd and integrability. *Int. J. Comput. Vis.* **35**(3), 203–222 (1999)
48. Zhang, L., Curless, B., Hertzmann, A., Seitz, S.: Shape and motion under varying illumination: unifying structure from motion, photometric stereo, and multi-view stereo. In: IEEE Int. Conf. on Computer Vision, pp. 618–625 (2003)
49. Zhang, L., Samaras, D.: Face recognition under variable lighting using harmonic image exemplars. In: IEEE Conf. on Computer Vision and Pattern Recognition, vol. I, pp. 19–25 (2003)
50. Zou, X., Kittler, J., Messer, K.: Illumination invariant face recognition: A survey. In: *Biometrics: Theory, Applications, and Systems*, pp. 1–8 (2007)



# Chapter 8

## Face Recognition Across Pose and Illumination

Ralph Gross, Simon Baker, Iain Matthews, and Takeo Kanade

### 8.1 Introduction

The most recent evaluation of commercial face recognition systems shows the level of performance for face verification of the best systems to be on par with finger-print recognizers for frontal, uniformly illuminated faces [38]. Recognizing faces reliably across changes in pose and illumination has proved to be a much more difficult problem [9, 24, 38]. Although most research has so far focused on frontal face recognition, there is a sizable body of work on pose invariant face recognition and illumination invariant face recognition. However, face recognition across pose *and* illumination has received little attention.

#### 8.1.1 Multiview Face Recognition and Face Recognition Across Pose

Approaches addressing pose variation can be classified into two categories depending on the type of gallery images they use. Multiview face recognition is a direct extension of frontal face recognition in which the algorithms require gallery images of every subject at every pose. In face recognition across pose, we are concerned

---

R. Gross (✉) · S. Baker · I. Matthews · T. Kanade  
Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA  
e-mail: [rgross@cs.cmu.edu](mailto:rgross@cs.cmu.edu)

S. Baker  
e-mail: [simonb@cs.cmu.edu](mailto:simonb@cs.cmu.edu)

I. Matthews  
e-mail: [iainm@cs.cmu.edu](mailto:iainm@cs.cmu.edu)

T. Kanade  
e-mail: [tk@cs.cmu.edu](mailto:tk@cs.cmu.edu)

with the problem of building algorithms to recognize a face from a novel viewpoint (i.e., a viewpoint from which it has not previously been seen). In both categories, we furthermore distinguish between model-based and appearance-based algorithms. Model-based algorithms use an explicit two-dimensional (2D) [12] or 3D [10, 15] model of the face, whereas appearance-based methods directly use image pixels or features derived from image pixels [36].

One of the earliest appearance-based multiview algorithms was described by Beymer [6]. After a pose estimation step, the algorithm geometrically aligns the probe images to candidate poses of the gallery subjects using the automatically determined locations of three feature points. This alignment is then refined using optical flow. Recognition is performed by computing normalized correlation scores. Good recognition results are reported on a database of 62 subjects imaged in a number of poses ranging from  $-30^\circ$  to  $+30^\circ$  (yaw) and from  $-20^\circ$  to  $+20^\circ$  (pitch). However, the probe and gallery poses are similar. Pentland et al. [37] extended the popular eigenface approach of Turk and Pentland [47] to handle multiple views. The authors compare the performance of a parametric eigenspace (computed using all views from all subjects) with view-based eigenspaces (separate eigenspaces for each view). In experiments on a database of 21 people recorded in nine evenly spaced views from  $-90^\circ$  to  $+90^\circ$ , view-based eigenspaces outperformed the parametric eigenspace by a small margin.

A number of 2D model-based algorithms have been proposed for face tracking through large pose changes. In one study [13], separate active appearance models were trained for profile, half-profile, and frontal views, with models for opposing views created by simple reflection. Using a heuristic for switching between models, the system was able to track faces through wide angle changes. It has been shown that linear models are able to deal with considerable pose variation so long as all the modeled features remained visible [32]. A different way of dealing with larger pose variations is then to introduce nonlinearities into the model. Romdhani et al. extended active shape models [41] and active appearance models [42] using a kernel PCA to model shape and texture nonlinearities across views. In both cases, models were successfully fit to face images across a full  $180^\circ$  rotation. However, no face recognition experiments were performed.

In many face recognition scenarios, the pose of the probe and gallery images are different. For example, the gallery image might be a frontal “mug shot,” and the probe image might be a three-quarter view captured from a camera in the corner of a room. The number of gallery and probe images can also vary. For example, the gallery might consist of a pair of images for each subject, a frontal mug shot and full profile view (like the images typically captured by police departments). The probe might be a similar pair of images, a single three-quarter view, or even a collection of views from random poses. In these scenarios, multiview face recognition algorithms cannot be used. Early work on face recognition across pose was based on the idea of linear object classes [48]. The underlying assumption is that the 3D shape of an object (and 2D projections of 3D objects) can be represented by a linear combination of prototypical objects. It follows that a rotated view of the object is a linear combination of the rotated views of the prototype objects. Using this idea the

authors were able to synthesize rotated views of face images from a single-example view. This algorithm has been used to create virtual views from a single input image for use in a multiview face recognition system [7]. Lando and Edelman used a comparable example-based technique to generalize to new poses from a single view [31].

A completely different approach to face recognition across pose is based on the work of Murase and Nayar [36]. They showed that different views of a rigid object projected into an eigenspace fall on a 2D manifold. Using a model of the manifold they could recognize objects from arbitrary views. In a similar manner Graham and Allison observed that a densely sampled image sequence of a rotating head forms a characteristic *eigensignature* when projected into an eigenspace [19]. They use radial basis function networks to generate eigensignatures based on a single view input. Recognition is then performed by distance computation between the projection of a probe image into eigenspace and the eigensignatures created from gallery views. Good generalization is observed from half-profile training views. However, recognition rates for tests across wide pose variations (e.g., frontal gallery and profile probe) are weak.

One of the early model-based approaches for face recognition is based on elastic bunch graph matching [49]. Facial landmarks are encoded with sets of complex Gabor wavelet coefficients called jets. A face is then represented with a graph where the various jets form the nodes. Based on a small number of hand-labeled examples, graphs for new images are generated automatically. The similarity between a probe graph and the gallery graphs is determined as average over the similarities between pairs of corresponding jets. Correspondences between nodes in different poses is established manually. Good recognition results are reported on frontal faces in the FERET evaluation [39]. Recognition accuracies decrease drastically, though, for matching half profile images with either frontal or full profile views. For the same framework, a method for transforming jets across pose has been introduced [35]. In limited experiments, the authors show improved recognition rates over the original representation.

### 8.1.2 Illumination Invariant Face Recognition

In addition to face pose, illumination is the next most significant factor affecting the appearance of faces. Ambient lighting changes greatly within and between days and among indoor and outdoor environments. Due to the 3D structure of face, a direct lighting source can cast strong shadows that accentuate or diminish certain facial features. It has been shown experimentally [2] and theoretically for systems based on principal component analysis (PCA) [50] that differences in appearance induced by illumination are larger than differences between individuals. Because dealing with illumination variation is a central topic in computer vision, numerous approaches for illumination invariant face recognition have been proposed.

Early work in illumination invariant face recognition focused on image representations that are mostly insensitive to changes in illumination. In one study [2],

various image representations and distance measures were evaluated on a tightly controlled face database that varied the face pose, illumination, and expression. The image representations include edge maps, 2D Gabor-like filters, first and second derivatives of the gray-level image, and the logarithmic transformations of the intensity image along with these representations. However, none of the image representations was found to be sufficient by itself to overcome variations due to illumination changes. In more recent work, it was shown that the ratio of two images from the same object is simpler than the ratio of images from different objects [27]. In limited experiments, this method outperformed both correlation and PCA but did not perform as well as the illumination cone method described below. A related line of work attempted to extract the object's surface reflectance as an illumination invariant description of the object [25, 30]. We discuss the most recent algorithm in this area in more detail in Sect. 8.4.2. Sashua and Riklin-Raviv [44] proposed a different illumination invariant image representation, the quotient image. Computed from a small set of example images, the quotient image can be used to re-render an object of the same class under a different illumination condition. In limited recognition experiments the method outperforms PCA.

A different approach to the problem is based on the observation that the images of a Lambertian surface, taken from a fixed viewpoint but under varying illumination, lie in a 3D linear subspace of the image space [43]. A number of appearance-based methods exploit this fact to model the variability of faces under changing illumination. Belhumeur et al. [4] extended the eigenface algorithm of Turk and Pentland [47] to fisherfaces by employing a classifier based on Fisher's linear discriminant analysis. In experiments on a face database with strong variations in illumination, fisherfaces outperform eigenfaces by a wide margin. Further work in the area by Belhumeur and Kriegman showed that the set of images of an object in fixed pose but under varying illumination forms a convex cone in the space of images [5]. The illumination cones of human faces can be approximated well by low-dimensional linear subspaces [16]. An algorithm based on this method outperforms both eigenfaces and fisherfaces. More recently, Basri and Jacobs showed that the illumination cone of a convex Lambertian surface can be approximated by a nine-dimensional linear subspace [3]. In limited experiments, good recognition rates across illumination conditions are reported.

Common to all these appearance-based methods is the need for training images of database subjects under a number of different illumination conditions. An algorithm proposed by Sim and Kanade overcomes this restriction [45]. They used a statistical shape-from-shading model to recover the face shape from a single image and synthesize the face under a new illumination. Using this method, they generated images of the gallery subjects under many different illumination conditions to serve as gallery images in a recognizer based on PCA. High recognition rates are reported on the illumination subset of the CMU PIE database [46].

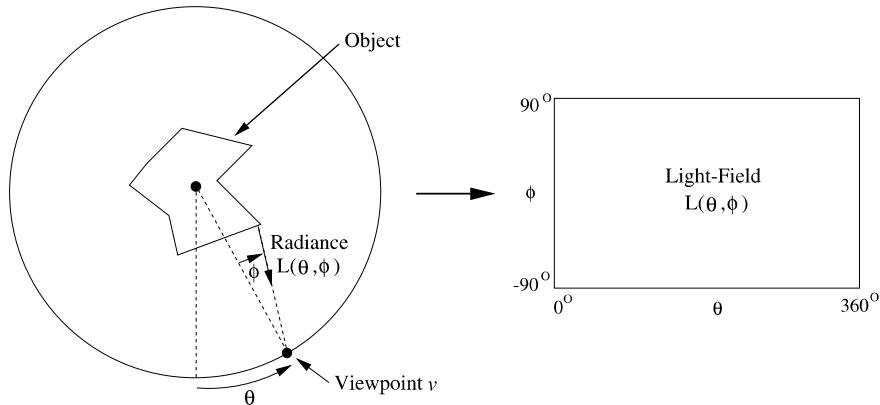
### 8.1.3 Algorithms for Face Recognition Across Pose and Illumination

A number of appearance and model-based algorithms have been proposed to address the problems of face recognition across pose and illumination simultaneously. In one study [17], a variant of photometric stereo was used to recover the shape and albedo of a face based on seven images of the subject seen in a fixed pose. In combination with the illumination cone representation introduced in [5], the authors can synthesize faces in novel pose and illumination conditions. In tests on 4050 images from the Yale Face Database B, the method performed almost without error. In another study [11], a morphable model of 3D faces was introduced. The model was created using a database of Cyberware laser scans of 200 subjects. Following an analysis-by-synthesis paradigm, the algorithm automatically recovers face pose and illumination from a single image. For initialization, the algorithm requires the manual localization of seven facial feature points. After fitting the model to a new image, the extracted model parameters describing the face shape and texture are used for recognition. The authors reported excellent recognition rates on both the FERET [39] and CMU PIE [46] databases. Once fit, the model could also be used to synthesize an image of the subject under new conditions. This method was used in the most recent face recognition vendor test to create frontal view images from rotated views [38]. For 9 of 10 face recognition systems tested, accuracies on the synthesized frontal views were significantly higher than on the original images.

## 8.2 Eigen Light-Fields

We propose an appearance-based algorithm for face recognition across pose. Our algorithm can use any number of gallery images captured at arbitrary poses and any number of probe images also captured with arbitrary poses. A minimum of one gallery and one probe image are needed, but if more images are available the performance of our algorithm generally improves.

Our algorithm operates by estimating (a representation of) the light-field [34] of the subject's head. First, generic training data are used to compute an eigenspace of head light-fields, similar to the construction of eigenfaces [47]. Light-fields are simply used rather than images. Given a collection of gallery or probe images, the projection into the eigenspace is performed by setting up a least-squares problem and solving for the projection coefficients similar to approaches used to deal with occlusions in the eigenspace approach [8, 33]. This simple linear algorithm can be applied to any number of images captured from any poses. Finally, matching is performed by comparing the probe and gallery eigen light-fields.



**Fig. 8.1** The object is conceptually placed within a *circle*. The angle to the viewpoint  $v$  around the circle is measured by the angle  $\theta$ , and the direction the viewing ray makes with the radius of the circle is denoted  $\phi$ . For each pair of angles  $\theta$  and  $\phi$ , the radiance of light reaching the viewpoint from the object is then denoted by  $L(\theta, \phi)$ , the *light-field*. Although the light-field of a 3D object is actually 4D, we continue to use the 2D notation of this figure in this chapter for ease of explanation

## 8.2.1 Light-Fields Theory

### 8.2.1.1 Object Light-Fields

The *plenoptic function* [1] or *light-field* [34] is a function that specifies the radiance of light in free space. It is a 5D function of position (3D) and orientation (2D). In addition, it is also sometimes modeled as a function of time, wavelength, and polarization, depending on the application in mind. In 2D, the light-field of a 2D object is actually 2D rather than the 3D that might be expected. See Fig. 8.1 for an illustration.

### 8.2.1.2 Eigen Light-Fields

Suppose we are given a collection of light-fields  $L_i(\theta, \phi)$  of objects  $O_i$  (here faces of different subjects) where  $i = 1, \dots, N$ . See Fig. 8.1 for the definition of this notation. If we perform an eigendecomposition of these vectors using PCA, we obtain  $d \leq N$  eigen light-fields  $E_i(\theta, \phi)$  where  $i = 1, \dots, d$ . Then, assuming that the eigenspace of light-fields is a good representation of the set of light-fields under consideration, we can approximate any light-field  $L(\theta, \phi)$  as

$$L(\theta, \phi) \approx \sum_{i=1}^d \lambda_i E_i(\theta, \phi) \quad (8.1)$$

where  $\lambda_i = \langle L(\theta, \phi), E_i(\theta, \phi) \rangle$  is the inner (or dot) product between  $L(\theta, \phi)$  and  $E_i(\theta, \phi)$ . This decomposition is analogous to that used for face and object recogni-

tion [36, 47]. The mean light-field could also be estimated and subtracted from all of the light-fields.

Capturing the complete light-field of an object is a difficult task, primarily because it requires a huge number of images [18, 34]. In most object recognition scenarios, it is unreasonable to expect more than a few images of the object (often just one). However, any image of the object corresponds to a curve (for 3D objects, a surface) in the light-field. One way to look at this curve is as a highly occluded light-field; only a small part of the light-field is visible. Can the eigen coefficients  $\lambda_i$  be estimated from this highly occluded view? Although this may seem hopeless, consider that light-fields are highly redundant, especially for objects with simple reflectance properties such as Lambertian. An algorithm has been presented [33] to solve for the unknown  $\lambda_i$  for eigen *images*. A similar algorithm was implicitly used by Black and Jepson [8]. Rather than using the inner product  $\lambda_i = \langle L(\theta, \phi), E_i(\theta, \phi) \rangle$ , Leonardis and Bischof [33] solved for  $\lambda_i$  as the least-squares solution of

$$L(\theta, \phi) - \sum_{i=1}^d \lambda_i E_i(\theta, \phi) = 0 \quad (8.2)$$

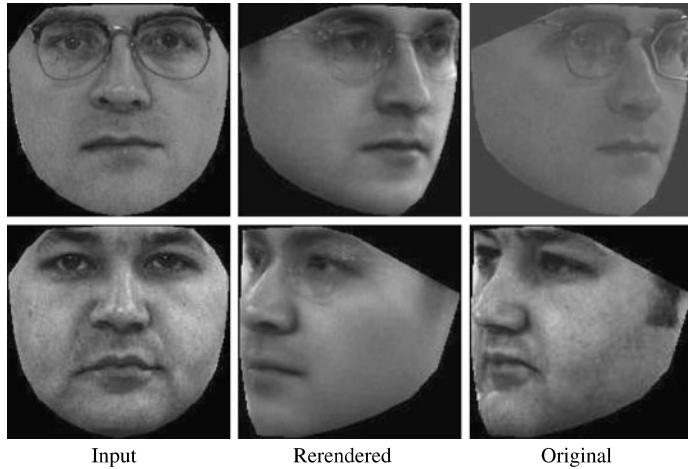
where there is one such equation for each pair of  $\theta$  and  $\phi$  that are unoccluded in  $L(\theta, \phi)$ . Assuming that  $L(\theta, \phi)$  lies *completely within the eigenspace* and that enough pixels are unoccluded, the solution of (8.2) is exactly the same as that obtained using the inner product [21]. Because there are  $d$  unknowns ( $\lambda_1 \dots \lambda_d$ ) in (8.2), at least  $d$  unoccluded light-field pixels are needed to overconstrain the problem, but more may be required owing to linear dependencies between the equations. In practice, two to three times as many equations as unknowns are typically required to get a reasonable solution [33]. Given an image  $I(m, n)$ , the following is then an algorithm for estimating the eigen light-field coefficients  $\lambda_i$ .

1. For each pixel  $(m, n)$  in  $I(m, n)$ , compute the corresponding light-field angles  $\theta_{m,n}$  and  $\phi_{m,n}$ . (This step assumes that the camera intrinsics are known, as well as the relative orientation of the camera to the object.)
2. Find the least-squares solution (for  $\lambda_1 \dots \lambda_d$ ) to the set of equations

$$I(m, n) - \sum_{i=1}^d \lambda_i E_i(\theta_{m,n}, \phi_{m,n}) = 0 \quad (8.3)$$

where  $m$  and  $n$  range over their allowed values. (In general, the eigen light-fields  $E_i$  need to be interpolated to estimate  $E_i(\theta_{m,n}, \phi_{m,n})$ . Also, all of the equations for which the pixel  $I(m, n)$  does not image the object should be excluded from the computation.)

Although we have described this algorithm for a single image  $I(m, n)$ , any number of images can obviously be used (so long as the camera intrinsics and relative orientation to the object are known for each image). The extra pixels from the other images are simply added in as additional constraints on the unknown coefficients  $\lambda_i$  in (8.3). The algorithm can be used to estimate a light-field from a collection of



**Fig. 8.2** Our eigen light-field estimation algorithm for rerendering a face across pose. The algorithm is given the *left-most (frontal) image* as input from which it estimates the eigen light-field and then creates the rotated view shown in the middle. For comparison, the original rotated view is shown in the *right-most column*. In the figure, we show one of the better results (*top*) and one of the worst (*bottom*). Although in both cases the output looks like a face, the identity is altered in the second case

images. Once the light-field has been estimated, it can then be used to render new images of the same object under different poses. (See Vetter and Poggio [48] for a related algorithm.) We have shown [21] that the algorithm correctly rerenders a given object assuming a Lambertian reflectance model. The extent to which these assumptions are valid are illustrated in Fig. 8.2, where we present the results of using our algorithm to rerender faces across pose. In each case, the algorithm received the left-most (frontal) image as input and created the rotated view in the middle. For comparison, the original rotated view is included as the right-most image. The rerendered image for the first subject is similar to the original. Although the image created for the second subject still shows a face in the correct pose, the identity of the subject is not as accurately recreated. We conclude that overall our algorithm works fairly well but that more training data are needed so the eigen light-field of faces can more accurately represent any given face light-field.

### 8.2.2 Application to Face Recognition Across Pose

The eigen light-field estimation algorithm described above is somewhat abstract. To be able to use it for face recognition across pose, we need to do the following things.

*Vectorization:* The input to a face recognition algorithm consists of a collection of images (possibly just one) captured from a variety of poses. The eigen light-field estimation Algorithm operates on light-field vectors (light-fields represented

as vectors). Vectorization consists of converting the input images into a light-field vector (with missing elements, as appropriate.)

*Classification:* Given the eigen coefficients  $a_1 \dots a_d$  for a collection of gallery faces and for a probe face, we need to classify which gallery face is the most likely match.

*Selecting training and testing sets:* To evaluate our algorithm, we have to divide the database used into (disjoint) subsets for training and testing.

We now describe each of these tasks in turn.

### 8.2.2.1 Vectorization by Normalization

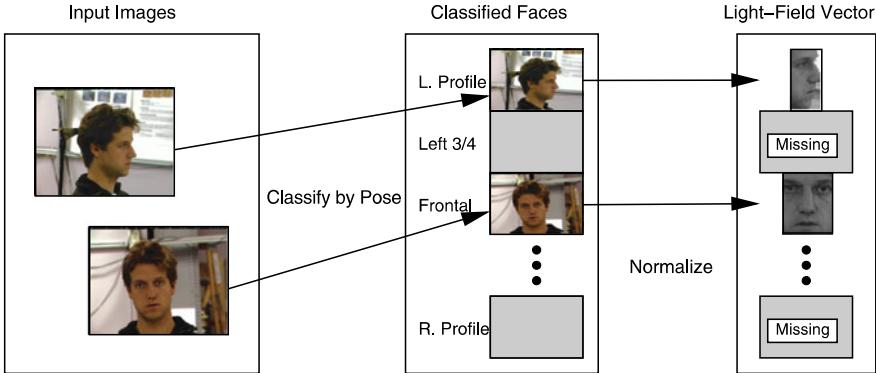
Vectorization is the process of converting a collection of images of a face into a light-field vector. Before we can do this we first have to decide how to discretize the light-field into pixels. Perhaps the most natural way to do this is to uniformly sample the light-field angles ( $\theta$  and  $\phi$  in the 2D case of Fig. 8.1). This is not the only way to discretize the light-field. Any sampling, uniform or nonuniform, could be used. All that is needed is a way to specify what is the allowed set of light-field pixels. For each such pixel, there is a corresponding index in the light-field vector; that is, if the light-field is sampled at  $K$  pixels, the light-field vectors are  $K$  dimensional vectors.

We specify the set of light-field pixels in the following manner. We assume that there are only a finite set of poses  $1, 2, \dots, P$  in which the face can occur. Each face image is first classified into the nearest pose. (Although this assumption is clearly an approximation, its validity is demonstrated by the empirical results in Sect. 8.2.3. In both the FERET [39] and PIE [46] databases, there is considerable variation in the pose of the faces. Although the subjects are asked to place their face in a fixed pose, they rarely do this perfectly. Both databases therefore contain considerable variation away from the finite set of poses. Our algorithm performs well on both databases, so the approximation of classifying faces into a finite set of poses is validated.)

Each pose  $i = 1, \dots, P$  is then allocated a fixed number of pixels  $K_i$ . The total number of pixels in a light-field vector is therefore  $K = \sum_{i=1}^P K_i$ . If we have images from poses 3 and 7, for example, we know  $K_3 + K_7$  of the  $K$  pixels in the light-field vector. The remaining  $K - K_3 - K_7$  are unknown, missing data. This vectorization process is illustrated in Fig. 8.3.

We still need to specify how to sample the  $K_i$  pixels of a face in pose  $i$ . This process is analogous to that needed in appearance-based object recognition and is usually performed by “normalization.” In eigenfaces [47], the standard approach is to find the positions of several canonical points, typically the eyes and the nose, and to warp the input image onto a coordinate frame where these points are in fixed locations. The resulting image is then masked. To generalize eigenface normalization to eigen light-fields, we just need to define such a normalization for each pose.

We report results using two different normalizations. The first is a simple one based on the location of the eyes and the nose. Just as in eigenfaces, we assume that the eye and nose locations are known, warp the face into a coordinate frame in which these canonical points are in a fixed location, and finally crop the image



**Fig. 8.3** Vectorization by normalization. Vectorization is the process of converting a set of images of a face into a light-field vector. Vectorization is performed by first classifying each input image into one of a finite number of poses. For each pose, normalization is then applied to convert the image into a subvector of the light-field vector. If poses are missing, the corresponding part of the light-field vector is missing

with a (pose-dependent) mask to yield the  $K_i$  pixels. For this simple three-point normalization, the resulting masked images vary in size between 7200 and 12 600 pixels, depending on the pose.

The second normalization is more complex and is motivated by the success of active appearance models (AAMs) [12]. This normalization is based on the location of a large number (39–54 depending on the pose) of points on the face. These canonical points are triangulated and the image warped with a piecewise affine warp onto a coordinate frame in which the canonical points are in fixed locations. The resulting masked images for this multipoint normalization vary in size between 20 800 and 36 000 pixels. Although currently the multipoint normalization is performed using hand-marked points, it could be performed by fitting an AAM [12] and then using the implied canonical point locations.

### 8.2.2.2 Classification Using Nearest Neighbor

The eigen light-field estimation algorithm outputs a vector of eigen coefficients  $(a_1, \dots, a_d)$ . Given a set of gallery faces, we obtain a corresponding set of vectors  $(a_1^{\text{id}}, \dots, a_d^{\text{id}})$ , where  $\text{id}$  is an index over the set of gallery faces. Similarly, given a probe face, we obtain a vector  $(a_1, \dots, a_d)$  of eigen coefficients for that face. To complete the face recognition algorithm, we need an algorithm that classifies  $(a_1, \dots, a_d)$  with the index  $\text{id}$ , which is the most likely match. Many classification algorithms could be used for this task. For simplicity, we use the nearest-neighbor algorithm, that classifies the vector  $(a_1, \dots, a_d)$  with the index  $\text{id}$ .

$$\arg \min_{\text{id}} \text{dist}((a_1, \dots, a_d), (a_1^{\text{id}}, \dots, a_d^{\text{id}})) = \arg \min_{\text{id}} \sum_{i=1}^d (a_i - a_i^{\text{id}})^2. \quad (8.4)$$

All of the results reported in this chapter use the Euclidean distance in (8.4). Alternative distance functions, such as the Mahalanobis distance, could be used instead if so desired.

### 8.2.2.3 Selecting the Gallery, Probe, and Generic Training Data

In each of our experiments, we divided the database into three disjoint subsets:

*Generic training data:* Many face recognition algorithms such as eigenfaces, and including our algorithm, require “generic training data” to build a generic face model. In eigenfaces, for example, generic training data are needed to compute the eigenspace. Similarly, in our algorithm, generic data are needed to construct the eigen light-field.

*Gallery:* The gallery is the set of reference images of the people to be recognized (i.e., the images given to the algorithm as examples of each person who might need to be recognized).

*Probe:* The probe set contains the “test” images (i.e., the images to be presented to the system to be classified with the identity of the person in the image).

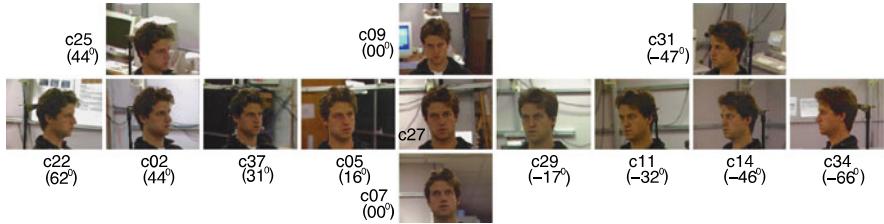
The division into these three subsets is performed as follows. First, we randomly select half of the subjects as the generic training data. The images of the remaining subjects are used for the gallery and probe. There is therefore never any overlap between the generic training data and the gallery and probe.

After the generic training data have been removed, the remainder of the databases are divided into probe and gallery sets based on the pose of the images. For example, we might set the gallery to be the frontal images and the probe set to be the left profiles. In this case, we evaluate how well our algorithm is able to recognize people from their profiles given that the algorithm has seen them only from the front. In the experiments described below we choose the gallery and probe poses in various ways. The gallery and probe are always disjoint unless otherwise noted.

## 8.2.3 Experimental Results

### 8.2.3.1 Databases

We used two databases in our face recognition across pose experiments, the CMU Pose, Illumination, and Expression (PIE) database [46] and the FERET database [39]. Each of these databases contains substantial pose variation. In the pose subset of the CMU PIE database (Fig. 8.4), the 68 subjects are imaged simultaneously under 13 poses totaling 884 images. In the FERET database, the subjects are imaged nonsimultaneously in nine poses. We used 200 subjects from the FERET pose subset, giving 1800 images in total. If not stated otherwise, we used half of the available subjects for training of the generic eigenspace (34 subjects for PIE, 100



**Fig. 8.4** Pose variation in the PIE database. The pose varies from full left profile (c34) to full frontal (c27) and to full right profile (c22). Approximate pose angles are shown below the camera numbers

subjects for FERET) and the remaining subjects for testing. In all experiments (if not stated otherwise), we retain a number of eigenvectors sufficient to explain 95% of the variance in the input data.

### 8.2.3.2 Comparison with Other Algorithms

We compared our algorithm with eigenfaces [47] and FaceIt, the commercial face recognition system from Identix (formerly Visionics).<sup>1</sup>

We first performed a comparison using the PIE database. After randomly selecting the generic training data, we selected the gallery pose as one of the 13 PIE poses and the probe pose as any other of the remaining 12 PIE poses. For each disjoint pair of gallery and probe poses, we computed the average recognition rate over all subjects in the probe and gallery sets. The details of the results are shown in Fig. 8.5 and are summarized in Table 8.1.

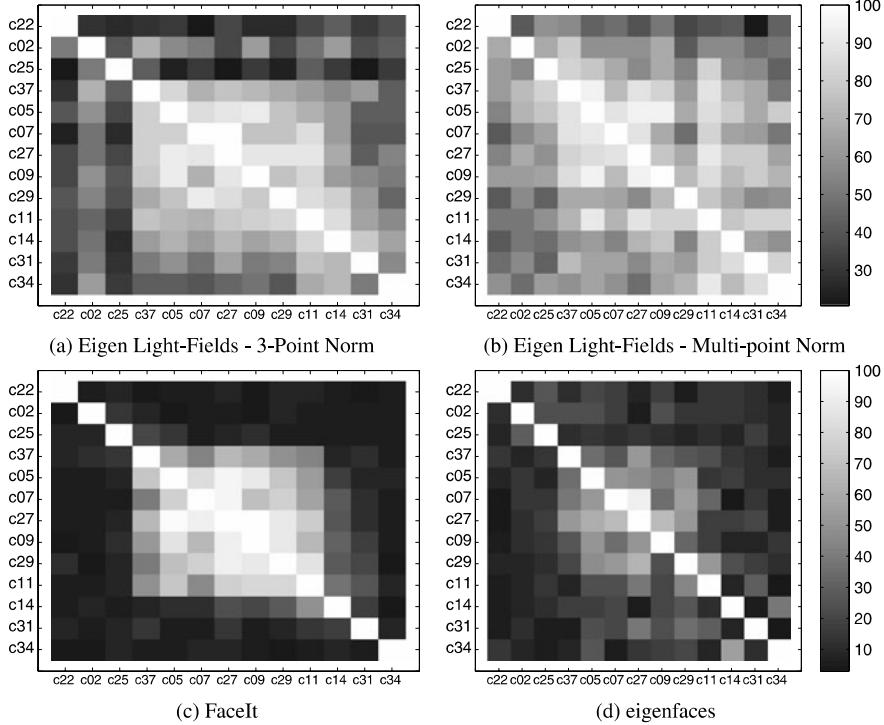
In Fig. 8.5, we plotted  $13 \times 13$  “confusion matrices” of the results. The row denotes the pose of the gallery, the column the pose of the probe, and the displayed intensity the average recognition rate. A lighter color denotes a higher recognition rate. (On the diagonals, the gallery and probe images are the same so all three algorithms obtain a 100% recognition rate.)

Eigen light-fields performed far better than the other algorithms, as witnessed by the lighter color of Fig. 8.5a, b compared to Fig. 8.5c, d. Note how eigen light-fields was far better able to generalize across wide variations in pose, and in particular to and from near-profile views.

Table 8.1 includes the average recognition rate computed over all disjoint gallery-probe poses. As can be seen, eigen light-fields outperformed both the standard eigenfaces algorithm and the commercial FaceIt system.

We next performed a similar comparison using the FERET database [39]. Just as with the PIE database, we selected the gallery pose as one of the nine FERET poses and the probe pose as any other of the remaining eight FERET poses. For each disjoint pair of gallery and probe poses, we computed the average recognition rate over

<sup>1</sup>Version 2.5.0.17 of the FaceIt recognition engine was used in the experiments.



**Fig. 8.5** Comparison with FaceIt and eigenfaces for face recognition across pose on the CMU PIE [46] database. For each pair of gallery and probe poses, we plotted the color-coded average recognition rate. The row denotes the pose of the gallery and the column the pose of the probe. The fact that the images in (a) and (b) are lighter in color than those in (c) and (d) implies that our algorithm performs better

**Table 8.1** Comparison of eigen light-fields with FaceIt and eigenfaces for face recognition across pose on the CMU PIE database. The table contains the average recognition rate computed across all disjoint pairs of gallery and probe poses; it summarizes the average performance in Fig. 8.5

Algorithm	Average recognition accuracy (%)
Eigenfaces	16.6
FaceIt	24.3
Eigen light-fields	
Three-point norm	52.5
Multipoint norm	66.3

all subjects in the probe and gallery sets, and then averaged the results. The results are similar to those for the PIE database and are summarized in Table 8.2. Again, eigen light-fields performed significantly better than either FaceIt or eigenfaces.

**Table 8.2** Comparison of eigen light-fields with FaceIt and eigenfaces for face recognition across pose on the FERET database. The table contains the average recognition rate computed across all disjoint pairs of gallery and probe poses. Again, eigen light-fields outperforms both eigenfaces and FaceIt

Algorithm	Average recognition accuracy (%)
Eigenfaces	39.4
FaceIt	59.3
Eigen light-fields three-point normalization	75.0

Overall, the performance improvement of eigen light-fields over the other two algorithms is more significant on the PIE database than on the FERET database. This is because the PIE database contains more variation in pose than the FERET database. For more evaluation results, see Gross et al. [23].

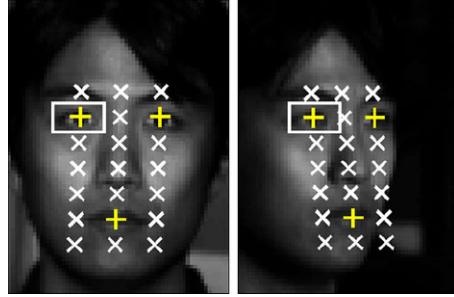
### 8.3 Bayesian Face Subregions

Owing to the complicated 3D nature of the face, differences exist in how the appearance of various face regions change for different face poses. If, for example, a head rotates from a frontal to a right profile position, the appearance of the mostly featureless cheek region only changes little (if we ignore the influence of illumination), while other regions such as the left eye disappear, and the nose looks vastly different. Our algorithm models the appearance changes of the different face regions in a probabilistic framework [28]. Using probability distributions for similarity values of face subregions; we compute the likelihood of probe and gallery images coming from the same subject. For training and testing of our algorithm we use the CMU PIE database [46].

#### 8.3.1 Face Subregions and Feature Representation

Using the hand-marked locations of both eyes and the midpoint of the mouth, we warp the input face images into a common coordinate frame in which the landmark points are in a fixed location and crop the face region to a standard  $128 \times 128$  pixel size. Each image  $I$  in the database is labeled with the identity  $i$  and pose  $\phi$  of the face in the image:  $I = (i, \phi)$ ,  $i \in \{1, \dots, 68\}$ ,  $\phi \in \{1, \dots, 13\}$ . As shown in Fig. 8.6, a  $7 \times 3$  lattice is placed on the normalized faces, and  $9 \times 15$  pixel subregions are extracted around every lattice point. The intensity values in each of the 21 subregions are normalized to have zero mean and unit variance.

As the similarity measure between subregions, we use SSD (sum of squared difference) values  $s_j$  between corresponding regions  $j$  for all image pairs. Because we compute the SSD after image normalization, it effectively contains the same information as normalized correlation.



**Fig. 8.6** Face subregions for two poses of the CMU PIE database. Each face in the database is warped into a normalized coordinate frame using the hand-labeled locations of both eyes and the midpoint of the mouth. A  $7 \times 3$  lattice is placed on the normalized face, and  $9 \times 15$  pixel subregions are extracted around every lattice point, resulting in a total of 21 subregions

### 8.3.2 Modeling Local Appearance Change Across Pose

For probe image  $I_{i,p} = (i, \phi_p)$  with unknown identity  $i$ , we compute the probability that  $I_{i,p}$  is coming from the same subject  $k$  as gallery image  $I_{k,g}$  for each face subregion  $j$ ,  $j \in \{1, \dots, 21\}$ . Using Bayes' rule, we write:

$$P(i = k | s_j, \phi_p, \phi_g) = \frac{P(s_j | i = k, \phi_p, \phi_g) P(i = k)}{P(s_j | i = k, \phi_p, \phi_g) P(i = k) + P(s_j | i \neq k, \phi_p, \phi_g) P(i \neq k)}. \quad (8.5)$$

We assume the conditional probabilities  $P(s_j | i = k, \phi_p, \phi_g)$  and  $P(s_j | i \neq k, \phi_p, \phi_g)$  to be Gaussian distributed and learn the parameters from data. Figure 8.7 shows histograms of similarity values for the right eye region. The examples in Fig. 8.7 show that the discriminative power of the right eye region diminishes as the probe pose changes from almost frontal (Fig. 8.7a) to right profile (Fig. 8.7c).

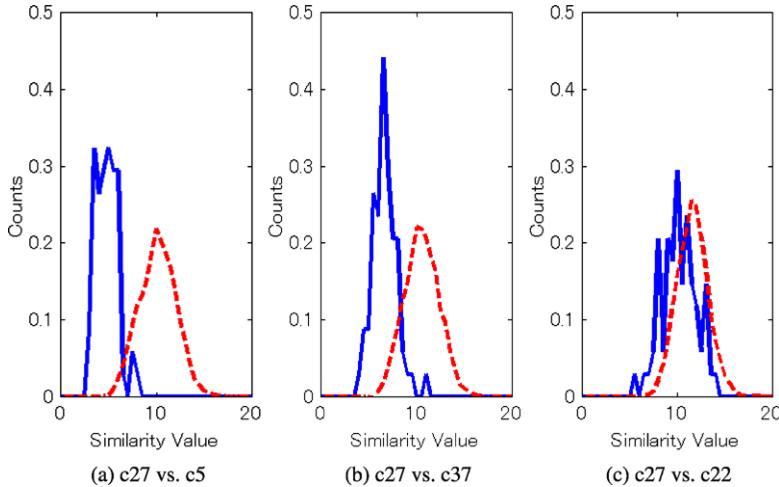
It is reasonable to assume that the pose of each gallery image is known. However, because the pose  $\phi_p$  of the probe images is in general not known, we marginalize over it. We can then compute the conditional densities for similarity value  $s_j$  as

$$P(s_j | i = k, \phi_g) = \sum_p P(\phi_p) P(s_j | i = k, \phi_p, \phi_g)$$

and

$$P(s_j | i \neq k, \phi_g) = \sum_p P(\phi_p) P(s_j | i \neq k, \phi_p, \phi_g).$$

If no other knowledge about the probe pose is given, the pose prior  $P(\phi_p)$  is assumed to be uniformly distributed. Similar to the posterior probability defined in (8.5), we compute the probability of the unknown probe image coming from the



**Fig. 8.7** Histograms of similarity values  $s_j$  for the right eye region across multiple poses. The distribution of similarity values for identical gallery and probe subjects are shown with *solid curves*, the distributions for different gallery and probe subjects are shown with *dashed curves*

same subject (given similarity value  $s_j$  and gallery pose  $\phi_g$ ) as

$$P(i = k \mid s_j, \phi_g) = \frac{P(s_j \mid i = k, \phi_g)P(i = k)}{P(s_j \mid i = k, \phi_g)P(i = k) + P(s_j \mid i \neq k, \phi_g)P(i \neq k)}. \quad (8.6)$$

To decide on the most likely identity of an unknown probe image  $I_{i,p} = (i, \phi_p)$ , we compute match probabilities between  $I_{i,p}$  and all gallery images for all face subregions using (8.5) or (8.6). We currently do not model dependencies between subregions, so we simply combine the different probabilities using the sum rule [29] and choose the identity of the gallery image with the highest score as the recognition result.

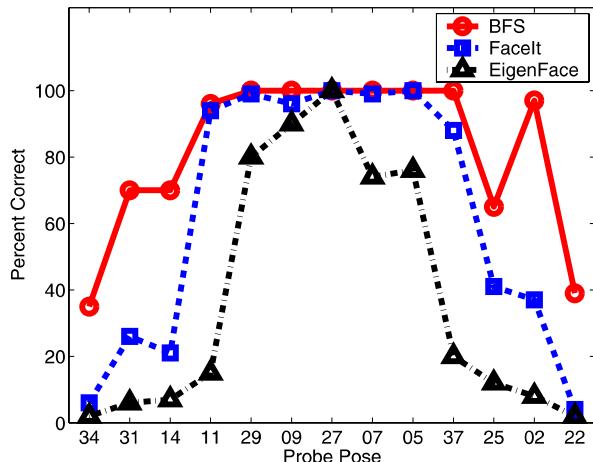
### 8.3.3 Experimental Results

We used half of the 68 subjects in the CMU PIE database for training of the models described in Sect. 8.3.2. The remaining 34 subjects are used for testing. The images of all 68 subjects are used in the gallery. We compare our algorithm to eigenfaces [47] and the commercial FaceIt system.

#### 8.3.3.1 Experiment 1: Unknown Probe Pose

For the first experiment, we assume the pose of the probe images to be unknown. We therefore must use (8.6) to compute the posterior probability that probe and

**Fig. 8.8** Recognition accuracies for our algorithm (labeled BFS), eigenfaces, and FaceIt for frontal gallery images and unknown probe poses. Our algorithm clearly outperforms both eigenfaces and FaceIt



gallery images come from the same subject. We assume  $P(\phi_p)$  to be uniformly distributed, that is,  $P(\phi_p) = \frac{1}{13}$ . Figure 8.8 compares the recognition accuracies of our algorithm with eigenfaces and FaceIt for frontal gallery images. Our system clearly outperforms both eigenfaces and FaceIt. Our algorithm shows good performance up until  $45^\circ$  head rotation between probe and gallery image (poses 02 and 31). The performance of eigenfaces and FaceIt already drops at  $15^\circ$  and  $30^\circ$  rotation, respectively.

### 8.3.3.2 Experiment 2: Known Probe Pose

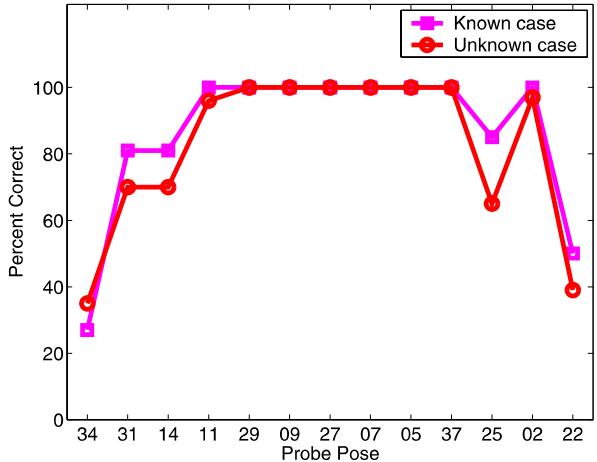
In the case of known probe pose, we can use (8.5) to compute the probability that probe and gallery images come from the same subject. Figure 8.9 compares the recognition accuracies of our algorithm for frontal gallery images for known and unknown probe poses. Only small differences in performances are visible.

Figure 8.10 shows recognition accuracies for all three algorithms for all possible combinations of gallery and probe poses. The area around the diagonal in which good performance is achieved is much wider for our algorithm than for either eigenfaces or FaceIt. We therefore conclude that our algorithm generalizes much better across pose than either eigenfaces or FaceIt.

## 8.4 Face Recognition Across Pose and Illumination

Because appearance-based methods use image intensities directly, they are inherently sensitive to variations in illumination. Drastic changes in illumination such as between indoor and outdoor scenes therefore cause significant problems for appearance-based face recognition algorithms [24, 38]. In this section, we describe two ways to handle illumination variations in facial imagery. The first algorithm

**Fig. 8.9** Comparison of recognition accuracies of our algorithm for frontal gallery images for known and unknown probe poses. Only small differences are visible



extracts illumination invariant subspaces by extending the previously introduced eigen light-fields to Fisher light-fields [22], mirroring the step from eigenfaces [47] to fisherfaces [4]. The second approach combines Bayesian face subregions with an image preprocessing algorithm that removes illumination variation prior to recognition [20]. In both cases, we demonstrate results for face recognition across pose *and* illumination.

#### 8.4.1 Fisher Light-Fields

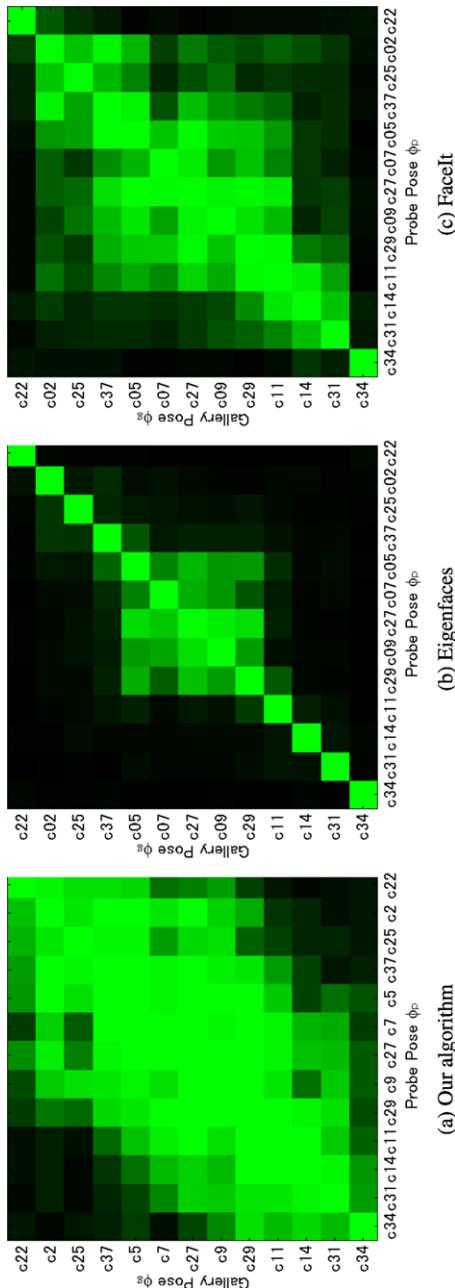
Suppose we are given a set of light-fields  $L_{i,j}(\theta, \phi)$ ,  $i = 1, \dots, N$ ,  $j = 1, \dots, M$  where each of  $N$  objects  $O_i$  is imaged under  $M$  different illumination conditions. We could proceed as described in Sect. 8.2.1.2 and perform PCA on the whole set of  $N \times M$  light-fields. An alternative approach is Fisher's linear discriminant (FLD) [14], also known as linear discriminant analysis (LDA) [51], which uses the available class information to compute a projection better suited for discrimination tasks. Analogous to the algorithm described in Sect. 8.2.1.2, we now find the least-squares solution to the set of equations

$$L(\theta, \phi) - \sum_{i=1}^m \lambda_i W_i(\theta, \phi) = 0 \quad (8.7)$$

where  $W_i$ ,  $i = 1, \dots, m$  are the generalized eigenvectors computed by LDA.

##### 8.4.1.1 Experimental Results

For our face recognition across pose and illumination experiments, we used the pose and illumination subset of the CMU PIE database [46]. In this subset, 68 subjects



**Fig. 8.10** Recognition accuracies for our algorithm, eigenfaces, and Facelt for all possible combinations of gallery and probe poses. Here lighter pixel values correspond to higher recognition accuracies. The area around the diagonal in which good performance is achieved is much wider for our algorithm than for either eigenfaces or Facelt

**Table 8.3** Performance of eigen light-fields and Fisher light-fields with FaceIt on three face recognition across pose and illumination scenarios. In all three cases, eigen light-fields and Fisher light-fields outperformed FaceIt by a large margin

Conditions	Eigen light-fields	Fisher light-fields	FaceIt
Same pose, different illumination	–	81.1%	41.6%
Different pose, same illumination	72.9%	–	25.8%
Different pose, different illumination	–	36.0%	18.1%

are imaged under 13 poses and 21 illumination conditions. Many of the illumination directions introduce fairly subtle variations in appearance, so we selected 12 of the 21 illumination conditions that span the set widely. In total, we used  $68 \times 13 \times 12 = 10\,608$  images in the experiments.

We randomly selected 34 subjects of the PIE database for the generic training data and then removed the data from the experiments (see Sect. 8.2.2.3). There were then a variety of ways to select the gallery and probe images from the remaining data.

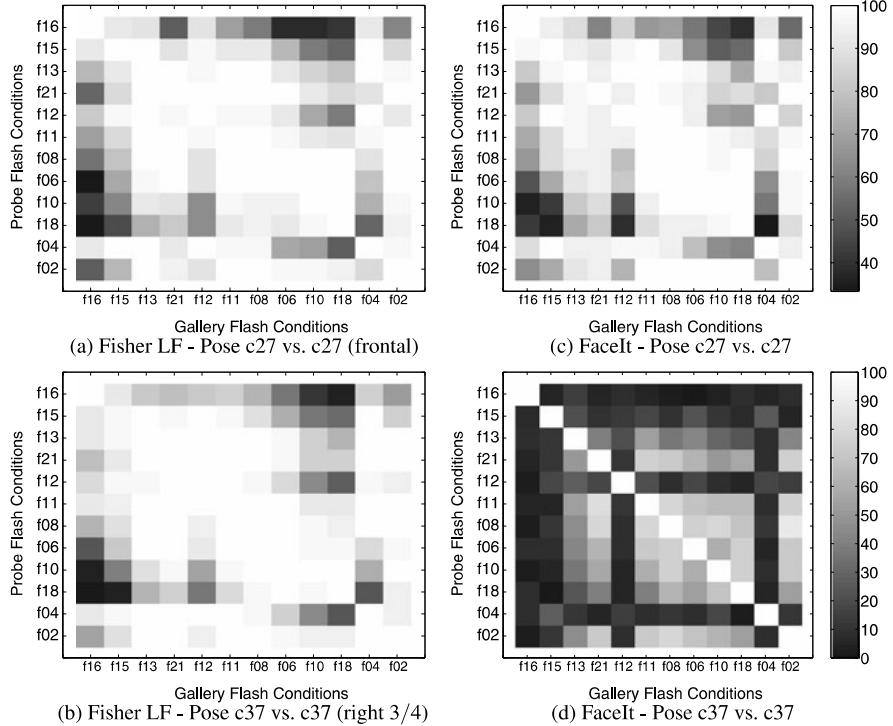
**Same pose, different illumination:** The gallery and probe poses are the same. The gallery and probe illuminations are different. This scenario is like traditional face recognition across illumination but is performed separately for each pose.

**Different pose, same illumination:** The gallery and probe poses are different. The gallery and probe illuminations are the same. This scenario is like traditional face recognition across pose but is performed separately for each possible illumination.

**Different pose, different illumination:** Both the pose and illumination of the probe and gallery are different. This is the most difficult and most general scenario.

We compared our algorithms with FaceIt under these three scenarios. In all cases we generated every possible test scenario and then averaged the results. For “same pose, different illumination,” for example, we consider every possible pose. We generated every pair of disjoint probe and gallery illumination conditions. We then computed the average recognition rate for each such case. We averaged over every pose and every pair of distinct illumination conditions. The results are included in Table 8.3. For “same-pose, different illumination,” the task is essentially face recognition across illumination separately for each pose. In this case, it makes little sense to try eigen light-fields because we know how poorly eigenfaces performs with illumination variation. Fisher light-fields becomes fisherfaces for each pose, which empirically we found outperforms FaceIt. Example illumination “confusion matrices” are included for two poses in Fig. 8.11.

For “different pose, same illumination,” the task reduces to face recognition across pose but for a variety of illumination conditions. In this case there is no intra-class variation, so it makes little sense to apply Fisher light-fields. This experiment is the same as Experiment 1 in Sect. 8.2.3 but the results are averaged over every possible illumination condition. As we found for Experiment 1, eigen light-fields outperforms FaceIt by a large amount.

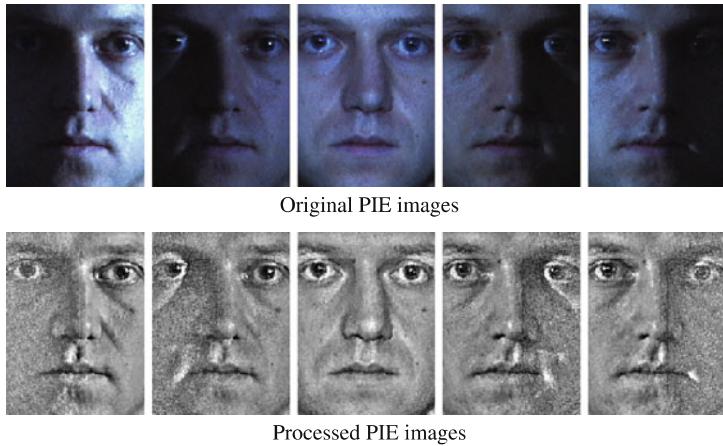


**Fig. 8.11** Example “confusion matrices” for the “same-pose, different illumination” task. For a given pose, and a pair of distinct probe and gallery illumination conditions, we color-code the average recognition rate. The superior performance of Fisher light-fields is witnessed by the lighter color of (a–b) over (c–d)

Finally, in the “different pose, different illumination” task both algorithms perform fairly poorly. However, the task is difficult. If the pose and illumination are both extreme, almost none of the face is visible. Because this case might occur in either the probe or the gallery, the chance that such a difficult case occurs is large. Although more work is needed on this task, note that Fisher light-fields still outperforms FaceIt by a large amount.

#### 8.4.2 Illumination Invariant Bayesian Face Subregions

In general, an image  $I(x, y)$  is regarded as product  $I(x, y) = R(x, y)L(x, y)$ , where  $R(x, y)$  is the reflectance and  $L(x, y)$  is the illuminance at each point  $(x, y)$  [26]. Computing the reflectance and the illuminance fields from real images is, in general, an ill-posed problem. Our approach uses two widely accepted assumptions about human vision to solve the problem: (1) human vision is *mostly* sensitive to scene reflectance and *mostly* insensitive to the illumination conditions; and (2) human



**Fig. 8.12** Result of removing illumination variations with our algorithm for a set of images from the PIE database

vision responds to local changes in contrast rather than to global brightness levels. Our algorithm computes an estimate of  $L(x, y)$  such that when it divides  $I(x, y)$  it produces  $R(x, y)$  in which the local contrast is appropriately enhanced. We find a solution for  $L(x, y)$  by minimizing

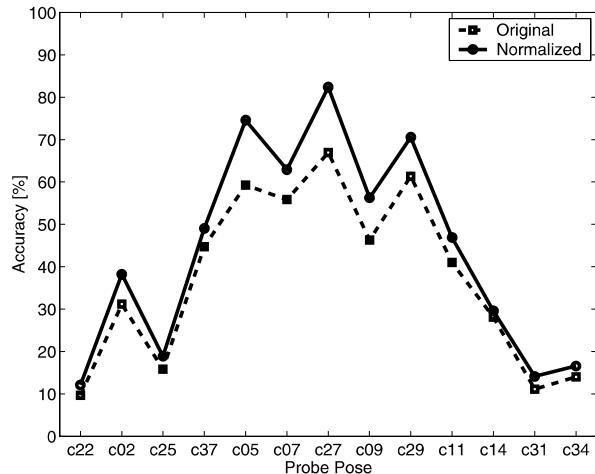
$$J(L) = \iint_{\Omega} \rho(x, y)(L - I)^2 dx dy + \lambda \iint_{\Omega} (L_x^2 + L_y^2) dx dy. \quad (8.8)$$

Here,  $\Omega$  refers to the image. The parameter  $\lambda$  controls the relative importance of the two terms. The space varying permeability weight  $\rho(x, y)$  controls the anisotropic nature of the smoothing constraint. See Gross and Brajovic [20] for details. Figure 8.12 shows examples from the CMU PIE database before and after processing with our algorithm. We used this algorithm to normalize the images of the combined pose and illumination subset of the PIE database. Figure 8.13 compares the recognition accuracies of the Bayesian face subregions algorithm for original and normalized images using gallery images with frontal pose and illumination. The algorithm achieved better performance on normalized images across all probe poses. Overall the average recognition accuracy improved from 37.3% to 44%.

## 8.5 Conclusions

One of the most successful and well studied approaches to object recognition is the *appearance-based* approach. The defining characteristic of appearance-based algorithms is that they directly use the pixel intensity values in an image of the object as the features on which to base the recognition decision. In this chapter, we described an appearance-based method for face recognition across pose based on an

**Fig. 8.13** Recognition accuracies of the Bayesian face subregions algorithm on original and normalized images using gallery images with frontal pose and illumination. For each probe pose, the accuracy is determined by averaging the results for all 21 illumination conditions. The algorithm achieves better performance on normalized images across all probe poses. The probe pose is assumed to be known



algorithm to estimate the eigen light-field from a collection of images. Unlike previous appearance-based methods, our algorithm can use any number of gallery images captured from arbitrary poses and any number of probe images also captured from arbitrary poses. The gallery and probe poses do not need to overlap. We showed that our algorithm can reliably recognize faces across pose and also take advantage of the additional information contained in widely separated views to improve recognition performance if more than one gallery or probe image is available.

In eigen light-fields, all face pixels are treated equally. However, differences exist in how the appearance of various face regions change across face poses. We described a second algorithm, Bayesian face subregions, which derives a model for these differences and successfully employs it for face recognition across pose. Finally, we demonstrated how to extend both algorithms toward face recognition across both pose and illumination. Note, however, that for this task recognition accuracies are significantly lower, suggesting that there still is room for improvement. For example, the model-based approach of Romdhani et al. [40] achieved better results across pose on the PIE database than the appearance-based algorithms described here.

**Acknowledgements** The research described here was supported by U.S. Office of Naval Research contract N00014-00-1-0915 and in part by U.S. Department of Defense contract N41756-03-C-4024. Portions of the research in this paper used the FERET database of facial images collected under the FERET program.

## References

1. Adelson, E., Bergen, J.: The plenoptic function and elements of early vision. In: Landy, M., Movshon, J.A. (eds.) Computational Models of Visual Processing. MIT Press, Cambridge (1991)
2. Adini, Y., Moses, Y., Ullman, S.: Face recognition: The problem of compensating for changes in illumination direction. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 721–732 (1997)

3. Basri, R., Jacobs, D.: Lambertian reflectance and linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(2), 218–233 (2003)
4. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 711–720 (1997)
5. Belhumeur, P., Kriegman, D.: What is the set of images of an object under all possible lighting conditions. *Int. J. Comput. Vis.* **28**(3), 245–260 (1998)
6. Beymer, D.: Face recognition under varying pose. Technical Report 1461, MIT AI Laboratory, Cambridge, MA (1993)
7. Beymer, D., Poggio, T.: Face recognition from one example view. A.I. Memo No. 1536, MIT AI Laboratory, Cambridge, MA (1995)
8. Black, M., Jepson, A.: Eigen-tracking: robust matching and tracking of articulated objects using a view-based representation. *Int. J. Comput. Vis.* **36**(2), 101–130 (1998)
9. Blackburn, D., Bone, M., Phillips, P.: Facial recognition vendor test 2000: evaluation report (2000)
10. Blanz, V., Romdhani, S., Vetter, T.: Face identification across different poses and illumination with a 3D morphable model. In: Proceedings of the Fifth International Conference on Face and Gesture Recognition, pp. 202–207 (2002)
11. Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(9), 1063–1074 (2003)
12. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(6), 681–685 (2001)
13. Cootes, T., Wheeler, G., Walker, K., Taylor, C.: View-based active appearance models. *Image Vis. Comput.* **20**, 657–664 (2002)
14. Fukunaga, K.: Introduction to Statistical Pattern Recognition. Academic Press, San Diego (1990)
15. Georghiades, A., Belhumeur, P., Kriegman, D.: From few to many: Generative models for recognition under variable pose and illumination. In: Proceedings of the Fourth International Conference on Face and Gesture Recognition, pp. 277–284 (2000)
16. Georghiades, A., Kriegman, D., Belhumeur, P.: Illumination cones for recognition under variable lighting: faces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (1998)
17. Georghiades, A., Kriegman, D., Belhumeur, P.: From few to many: generative models for recognition under variable pose and illumination. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(6), 643–660 (2001)
18. Gortler, S., Grzeszczuk, R., Szeliski, R., Cohen, M.: The lumigraph. In: Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH), pp. 43–54 (1996)
19. Graham, D., Allison, N.: Face recognition from unfamiliar views: subspace methods and pose dependency. In: 3rd International Conference on Automatic Face and Gesture Recognition, pp. 348–353 (1998)
20. Gross, R., Brajovic, V.: An image pre-processing algorithm for illumination invariant face recognition. In: 4th International Conference on Audio- and Video Based Biometric Person Authentication (AVBPA), pp. 10–18, June 2003
21. Gross, R., Matthews, I., Baker, S.: Eigen light-fields and face recognition across pose. In: Proceedings of the Fifth International Conference on Face and Gesture Recognition, pp. 1–7 (2002)
22. Gross, R., Matthews, I., Baker, S.: Fisher light-fields for face recognition across pose and illumination. In: Proceedings of the German Symposium on Pattern Recognition (DAGM), pp. 481–489 (2002)
23. Gross, R., Matthews, I., Baker, S.: Appearance-based face recognition and light-fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(4), 449–465 (2004)
24. Gross, R., Shi, J., Cohn, J.: Quo vadis face recognition. In: Third Workshop on Empirical Evaluation Methods in Computer Vision (2001)
25. Horn, B.: Determining lightness from an image. *Comput. Graph. Image Process.* **3**(1), 277–299 (1974)

26. Horn, B.: Robot Vision. MIT Press, Cambridge (1986)
27. Jacobs, D., Belhumeur, P., Basri, R.: Comparing images under variable illumination. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 610–617 (1998)
28. Kanade, T., Yamada, A.: Multi-subregion based probabilistic approach toward pose-invariant face recognition. In: IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA2003), pp. 954–959 (2003)
29. Kittler, J., Hatef, M., Duin, R., Matas, J.: On combining classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(3), 226–239 (1998)
30. Land, E., McCann, J.: Lightness and retinex theory. *J. Opt. Soc. Am.* **61**(1), 1–11 (1971)
31. Lando, M., Edelman, S.: Generalization from a single view in face recognition. In: International Workshop on Automatic Face-and Gesture-Recognition (1995)
32. Lanitis, A., Taylor, C., Cootes, T.: Automatic interpretation and coding of face images using flexible models. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 743–756 (1997)
33. Leonardis, A., Bischof, H.: Robust recognition using eigenimages. *Comput. Vis. Image Underst.* **78**(1), 99–118 (2000)
34. Levoy, M., Hanrahan, M.: Light field rendering. In: Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH), pp. 31–41 (1996)
35. Maurer, T., von der Malsburg, C.: Single-view based recognition of faces rotated in depth. In: International Workshop on Automatic Face and Gesture Recognition, pp. 248–253 (1995)
36. Murase, H., Nayar, S.: Visual learning and recognition of 3-D objects from appearance. *Int. J. Comput. Vis.* **14**, 5–24 (1995)
37. Pentland, A., Moghaddam, B., Starner, T.: View-based and modular eigenspaces for face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 84–91 (1994)
38. Phillips, P.J., Grother, P., Ross, J., Blackburn, D., Tabassi, E., Bone, M.: Face recognition vendor test 2002: evaluation report, March 2003
39. Phillips, P.J., Moon, H., Rizvi, S., Rauss, P.: The FERET evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(10), 1090–1104 (2000)
40. Romdhani, S., Blanz, V., Vetter, T.: Face identification by matching a 3D morphable model using linear shape and texture error functions. In: Proceedings of the European Conference on Computer Vision, pp. 3–19 (2002)
41. Romdhani, S., Gong, S., Psarrou, A.: Multi-view nonlinear active shape model using kernel PCA. In: 10th British Machine Vision Conference, vol. 2, pp. 483–492 (1999)
42. Romdhani, S., Psarrou, A., Gong, S.: On utilising template and feature-based correspondence in multi-view appearance models. In: 6th European Conference on Computer Vision, vol. 1, pp. 799–813 (2000)
43. Shashua, A.: Geometry and photometry in 3D visual recognition. PhD thesis, MIT (1994)
44. Shashua, A., Riklin-Raviv, T.: The Quotient image: class-based re-rendering and recognition with varying illumination conditions. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(2), 129–139 (2001)
45. Sim, T., Kanade, T.: Combining models and exemplars for face recognition: an illuminating example. In: Workshop on Models Versus Exemplars in Computer Vision (2001)
46. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination, and expression database. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(12), 1615–1618 (2003)
47. Turk, M., Pentland, A.: Face recognition using eigenfaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (1991)
48. Vetter, T., Poggio, T.: Linear object classes and image synthesis from a single example image. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 733–741 (1997)
49. Wiskott, L., Fellous, J., Kruger, N., von der Malsburg, C.: Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 775–779 (1997)
50. Zhao, W., Chellappa, R.: Robust face recognition using symmetric shape-from-shading. Technical report, Center for Automation Research, University of Maryland (1999)
51. Zhao, W., Krishnaswamy, A., Chellappa, R., Swets, D., Weng, J.: Discriminant analysis of principal components for face recognition. In: Wechsler, H., Phillips, P.J., Bruce, V., Huang, T. (eds.) *Face Recognition: From Theory to Applications*. Springer, New York (1998)



# Chapter 9

## Skin Color in Face Analysis

J. Birgitta Martinkauppi, Abdenour Hadid, and Matti Pietikäinen

### 9.1 Introduction

Color is a common feature used in machine vision applications. As a cue, it offers several advantages: easy to understand and use. Implementations can be made computationally fast and efficient, thus providing a low level cue. Under stable and uniform illumination, color cue remains robust against geometrical changes. Its ability to separate the targets from background depends on the color dissimilarity between targets and background. In some scenes, the color itself is enough for object detection.

The main difficulty in using color in machine vision applications is that the cameras are not able to distinguish changes of surface colors from color shifts caused by varying illumination spectra. Thus, color is sensitive to changes in illumination which are common under uncontrolled environments. The changes can be due to varying light level, for example, shadowing, varying light color due to changes in spectral power distribution (like daylight and fluorescent light source), or both. Cameras and their settings may produce different appearances which are different from the perception of human vision system.

Several strategies have been employed to reduce the illumination sensitivity. In one strategy, the color information is separated into two components, color inten-

---

J.B. Martinkauppi (✉)

Department of Electrical Engineering and Automation, University of Vaasa, Wolffintie 34,  
65101 Vaasa, Finland

e-mail: [birmar@uwasa.fi](mailto:birmar@uwasa.fi)

A. Hadid · M. Pietikäinen

Machine Vision Group, Department of Electrical and Information Engineering, University  
of Oulu, P.O. Box 4500, 90014 Oulu, Finland

A. Hadid

e-mail: [hadid@ee.oulu.fi](mailto:hadid@ee.oulu.fi)

M. Pietikäinen

e-mail: [mkp@ee.oulu.fi](mailto:mkp@ee.oulu.fi)

sity and color chromaticity. Use of color chromaticity component reduces the effect of varying light levels. To cancel the effect of illumination color and thus different spectral power distributions, numerous color constancy algorithms have been suggested, but their success has been limited [6]. A different strategy to these is to tolerate or adapt the model to the illumination changes. This strategy can produce promising results even under drastic variations in target colors as shown in this chapter for facial recognition.

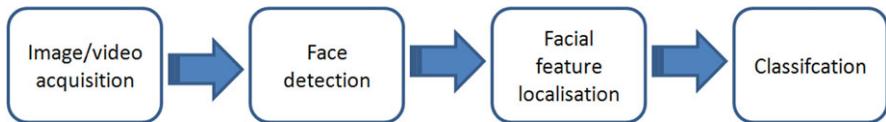
It is often preferable to get rid as much as possible of the dependencies on lighting intensity. The perfect case would be to also cancel-out the effect of the illuminant color (by defining a color representation which is only a function of the surface reflectance) but, thus far this has not been achieved in machine vision. The human visual system is superior in this sense, since human visual perception in which the color is perceived by the eye depends quite significantly on surface reflectance, although the light reaching the eye is a function of surface reflectance, illuminant color and lighting intensity.

For face detection, color has been an intriguing and popular cue. It is often used as a preprocessing step to select regions of interests for further, more computationally demanding processing. For instance, with the appearance-based face detection, an exhaustive scan (at different locations and scales) of the images is conducted when searching for the faces [54]. However, when the color cue is available, one can reduce the search regions by pre-processing the images and selecting the skin-like areas only.

This chapter deals with the role of color in facial image analysis such as face detection and recognition. First, we introduce the use of color information in the field of facial image analysis in particular (Sect. 9.2). Then, in Sect. 9.3, we give an introduction to color formation and discuss the effect of illumination on color appearance, and its consequences. The skin data can come from different sources like real faces, photos or print. Separating the sources of skin data is presented in Sect. 9.4, and skin color modeling is discussed in Sect. 9.5. Section 9.6 reviews the use of color in face detection, while the contribution of color to face recognition is covered in Sect. 9.7. Finally, conclusions are drawn in Sect. 9.8.

## 9.2 Color Cue and Facial Image Analysis

The properties of the face pattern pose a very difficult problem for facial image analysis: a face is a dynamic and nonrigid object which is difficult to handle. Its appearance varies due to changes in pose, expressions, illuminations and other factors such as age and make-up. As a consequence, most of the facial analysis tasks generally involve heavy computations due to the complexity of facial patterns. Therefore, one may need some additional cues, such as color or motion, in order to assist and accelerate the analysis. These additional cues also offer an indication of the reliability of the face analysis results: the more the cues support the analysis, the more one can be confident about the results. For instance, with the appearance-based face



**Fig. 9.1** A general block diagram of face analysis which shows different phases of facial image analysis

detection an exhaustive scan (at different locations and scales) of the images is conducted when searching the faces [54]. However, when the color cue is available, one can reduce the search regions by pre-processing the images and selecting only the skin-like areas. Therefore, it is not surprising that the color of skin has been commonly used to assist face detection. Also, in face recognition, it has been argued that color does play a role under degraded conditions by facilitating low-level facial image analysis such as better estimations of the boundaries, shapes and sizes of facial features [56]. As mentioned above, among the advantages of using color is the computational efficiency and robustness against some geometric changes such as scaling and rotation, when the scene is observed under a uniform illumination field. However, the main limitation with the use of color lies in its sensitivity to illumination changes (especially changes in the chromaticity of the illuminant source which are difficult to cancel-out).

Let us consider the general block diagram of face analysis, shown in Fig. 9.1. The color cue is involved at different stages [36]. In the first stage, the color images (or video sequences) are acquired and preprocessed. The preprocessing may include gamma correction, color space transformation, and so on. It is often preferable to get rid as much as possible of the dependencies on lighting intensity.

Among the different stages shown in Fig. 9.1, the use of color in face detection is probably the most obvious. It is generally used to select the skin-like color regions. Then, simple refining procedures can be launched to discriminate the faces from other skin-like regions such as hands, wood, etc. Thus, much faster face detectors are generally obtained when the color cue is considered.

Using the fact that some facial features, such as eyes, are darker than their surrounding regions, holes should then appear in the face area when labeling the skin pixels. Such observation is commonly exploited when detecting facial features in color images [10, 15, 54].

Does color information contribute to face recognition? The answer to this question is not obvious, although some studies have suggested that color does play a role in face recognition as well, and this contribution becomes evident when the shape cues are degraded [56]. Section 9.7 discusses this issue.

## 9.3 Color Appearance for Color Cameras

### 9.3.1 Color Image Formation and Illumination

Color cameras reproduce the scene with three components which are typically red ( $R$ ), green ( $G$ ) and blue ( $B$ ). The components are named after the spectral range over which the response was integrated. An example of color camera filters is shown in Fig. 9.2. The spectral filters typically operate in the visible wavelength spectrum range, that is, 400 nm–700 nm. Of course, different filter selections affect the obtainable descriptor set and most likely produce different values for the same input.

The descriptors themselves are obtained by filtering the color signal  $C(\lambda)$  with suitable spectral filters and integration over the filtered signal. The color signal is a spectral distribution of electromagnetic radiation, which is the light from an illumination source, light reflected from a surface or a combination of these. This is similar to calculation of human vision responses (see, e.g., [50]).

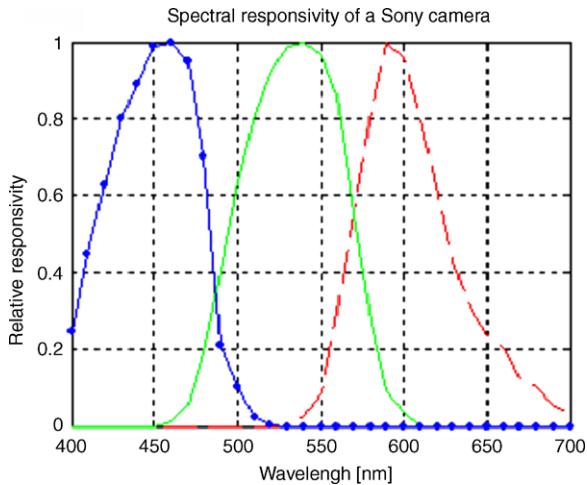
The following simple model represents camera output with white balancing:

$$D = \frac{\int \eta_D(\lambda) I_p(\lambda) S(\lambda) d\lambda}{\int \eta_D(\lambda) I_c(\lambda) d\lambda}, \quad (9.1)$$

where  $D$  is  $R$ ,  $G$  or  $B$  response,  $\lambda$  is the wavelength,  $p$  is prevailing (illumination) and  $c$  is calibration (illumination),  $\eta$  is the spectral responsivity of a spectral filter,  $I$  is the spectral power distribution of the light (SPD), and  $S$  is the spectral reflectance of the surface. The nominator of (9.1) alone describes image formation as a sum of the camera sensitivity, the illumination SPD and the reflectance over the wavelength range. Thus, for each pixel, the output value depends on the illumination, reflectance and camera sensitivity. This is a very simplified presentation of the formation but can be used as a basic theoretical estimation of the camera response to the input light. The denominator models the white balance. White balance means adjusting gains of camera so that the cameras response for white (or very bright gray) is equal in every channels. For example, the response of a white is adjusted to (255, 255, 255).

Equation (9.1) can be used to simulate the effect of illumination. When the prevailing and calibration illumination are the same, then the output image is called as a canonical or calibrated image and colors are canonical colors. This is described in more detail in Sect. 9.3.2. The prevailing and calibration illumination can also be different, and the output image in this case is called non-canonical image. The modeling is, however, more problematic. The problem of normalization can be demonstrated theoretically [32]. Let us assume that the prevailing illumination is originally  $I_{np}$  and its normalization factor is the constant factor  $f_p$ , and the calibration illumination is in the unnormalized format  $I_{nc}$ , which is normalized by the factor constant  $f_c$ . For example, if we insert these variables into (9.1), we can derive the following

**Fig. 9.2** Spectral responsivity curves of a Sony camera, originally obtained from a graph provided by the manufacturer



format:

$$R = \frac{\int \eta_R(\lambda) I_p(\lambda) S(\lambda) d\lambda}{\int \eta_R(\lambda) I_c(\lambda) d\lambda} = \frac{\int \eta_R(\lambda) \frac{I_{np}(\lambda)}{f_p} S(\lambda) d\lambda}{\int \eta_R(\lambda) \frac{I_{nc}(\lambda)}{f_c} d\lambda} = \frac{f_c}{f_p} \frac{\int \eta_R(\lambda) I_{np}(\lambda) S(\lambda) d\lambda}{\int \eta_R(\lambda) I_{nc}(\lambda) d\lambda}. \quad (9.2)$$

The ratio  $f_c/f_p$  is 1 only when the illumination conditions are the same. Different choices for normalization methods may produce different results [32].

### 9.3.2 The Effect of White Balancing

The effect of white balancing on the perceived images and colors is examined in more details in this chapter. White balancing is one of the important factors affecting image quality. The white balancing factor depends on the illumination. Many digital images have been taken under canonical conditions or very near to them to avoid distortions in colors. The color distortions are easily noticed and taken as annoying artifacts. This is especially true for certain colors which humans remember very well; thus, they are referred to as memory colors. One of these memory colors is, quite naturally, skin tone.

Humans are very sensitive to any distortion in skin tones [12, 25], thus, it is not so surprising that these have been investigated a lot. Skin tones refer here to the correct or acceptable colors for skin as perceived by a human. Skin colors refers to all those RGBs which a camera can perceive as skin under different illuminations. Note that human and cameras can perceive skin color differently.

In cameras, white balancing can be done automatically or manually. In manual selection, the user selects the best option for the prevailing illumination, while automatic option provides settings from a program. However, it is not always possible to



**Fig. 9.3** The face is illuminated by the nonuniform illumination field and the white balancing partially fails. The color appearance of the face varies at different parts of the light field

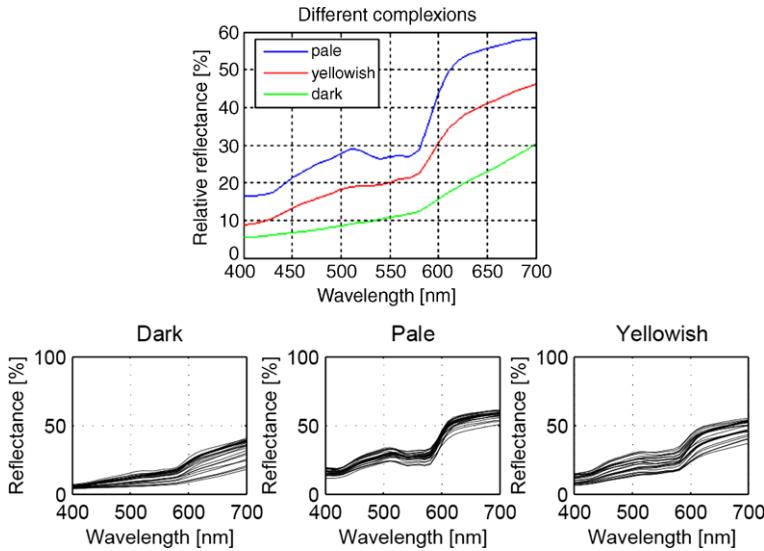
select or compute proper white balancing factors. This is especially true under varying, nonuniform illumination, which can cause more drastic color changes. For example, it is common to have more than one light source on a scene. If these sources with different SPDs shine over an object, it is not possible to conduct the correct white balancing for the whole image. This is demonstrated in Fig. 9.3. The face is imaged under a nonuniform illumination field. The camera was balanced under the light of fluorescent lamps on the ceiling and thus the part of the face under only fluorescent illumination field appears in skin tones. However, the daylight from windows causes a bluish color shift on the right side of the face image. The colors are distorted because the white balancing fails partially. The distortion between these two sides varies to a different degree as a function of illumination field. The nonuniform illumination fields are encountered commonly, but they are rarely considered in face detection or recognition applications.

Of course, one can apply some color correction techniques to improve the quality. For example, Do et al. used sclera region of the eye to estimate illumination color and then apply skin detection [4]. Even a nonuniform illumination field is possible to correct if the light colors are given by the user [16]. However, the failure in white balancing may cause information loss, which is generally very difficult to correct properly.

### 9.3.2.1 Canonical Images and Colors

Even though an image is taken under canonical condition, it does not guarantee that the objects appear in the same colors under different canonical illumination. White, grays and black do appear at least in most of the cases very similarly under different light sources, but of course there are some limitations. It is not possible even in theory to perceive all RGB components for a gray object if the prevailing illumination does not have spectral output in components' spectral range. The ideal camera RGB responses for white in canonical case should have equal RGB values even under different light sources, given that the sources are not very extreme. Cameras do reproduce a white surface quite well over a range of light sources, but of course there is a physical limitation due to gain control, for example.

If a camera has linear response over a certain input signal range, then those grays falling the range will be reproduced in gray colors if the color signal from scene falls into the input range. The grays here refer to those objects whose spectra is constant



**Fig. 9.4** Skin complexions of three skin groups: pale, yellowish, and dark. Their reflectances are smooth and similar, mainly separated by their levels. Measured skin reflectances are available, for example, in the Physics-based face database [31]

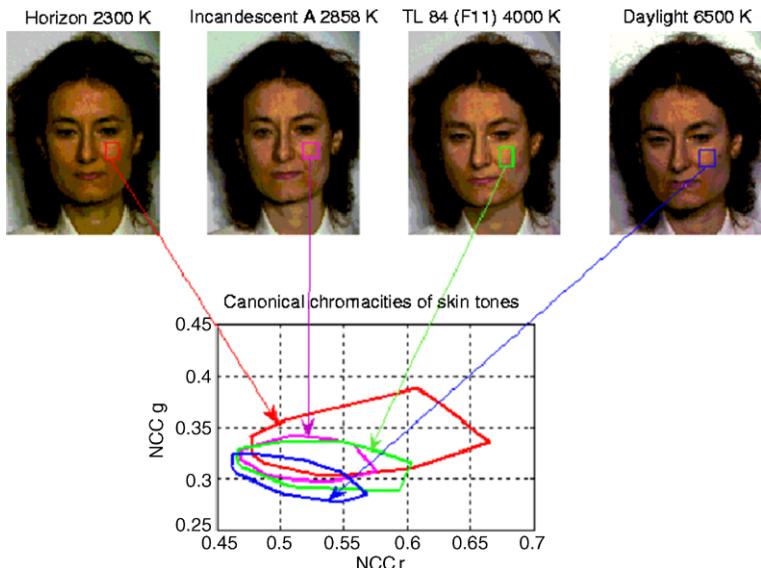
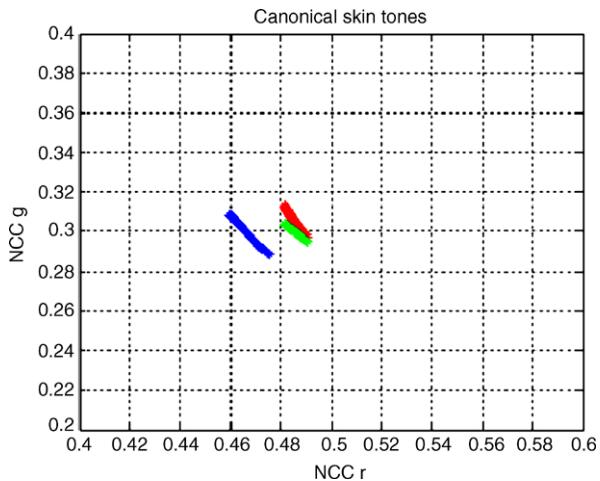
over the wavelength range, but the value of this constant is smaller than the maximum value (“white”). When the spectra is not constant over the range, the effect of illumination cannot be canceled out from the reproduced RGB values. Thus, the object colors will be affected to a different degree between different light sources. Therefore, a camera can reproduce only the achromatic colors similarly under different light sources assuming that the camera is white balanced to the prevailing light sources. This means that skin can have different colors under images taken with different conditions.

The reproduction differences can be demonstrated very easily. First, the objects need to be selected, and, in this case, three skin complexions (pale, yellowish and dark) are used. The spectral reflectances for the complexions are shown in Fig. 9.4. The reflectances are smooth and similar. They are separated mainly by their level, but not their shape [8, 17, 51], which suggests the similar reproduction in color. Due to this, skin spectra can be reconstructed at high quality using only three basis vectors [18, 38]. The similarity is due to the colorants (melanin, carotene, and hemoglobin) determining the reflectance [5]. As a natural object, skin has not uniform coloration.

Using (9.1), the RGB values for skin are calculated using the Sony camera’s responses. The RGB values are then converted into NCC chromaticity. These theoretical skin chromaticities are displayed in Fig. 9.5. The canonical skin values are dissimilar under different illuminations even in an ideal case.

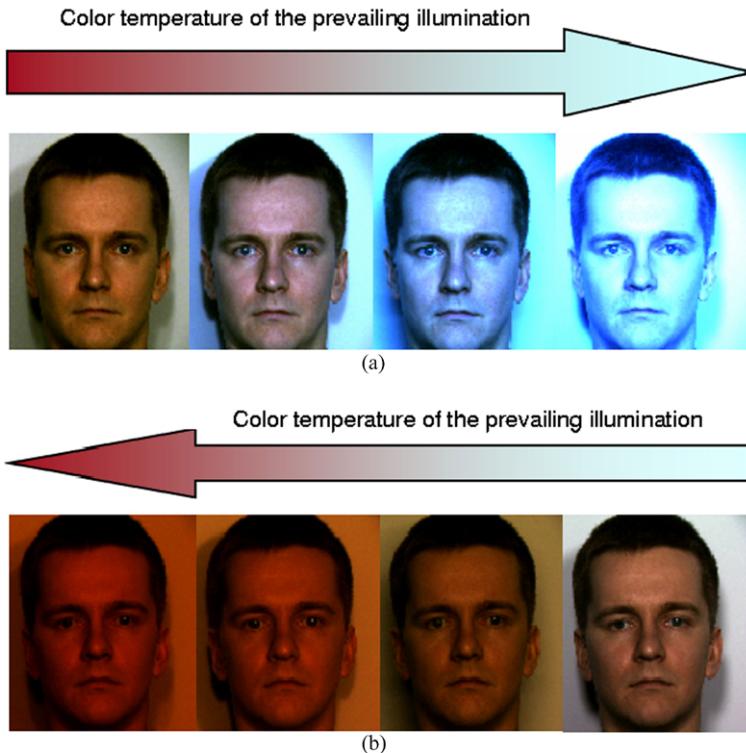
Cameras produce even bigger variations in skin colors: Fig. 9.6 shows skin chromaticities for a Sony camera taken under the same light sources as the ones used in simulation (Horizon 2300 K or light at sunset/sunrise), Incandescent A 2856 K,

**Fig. 9.5** Canonical skin tones were obtained by converting the theoretical skin RGBs to Normalized Color Coordinate (NCC) space



**Fig. 9.6** The skin tone appearance difference can be clearly observed in the four images taken with the Sony camera (see Fig. 9.2). From the selected area marked with a box the RGB values were taken and converted to NCC color space. As shown in the graph below the images, the areas of canonical chromaticities more or less overlap

fluorescent lamp TL84, and daylight D65 6500 K. The overlap between loci is significant. Note that the locus obtained using Horizon light covers a bigger area than for other light sources. This might be due to unsuccessful white balancing.

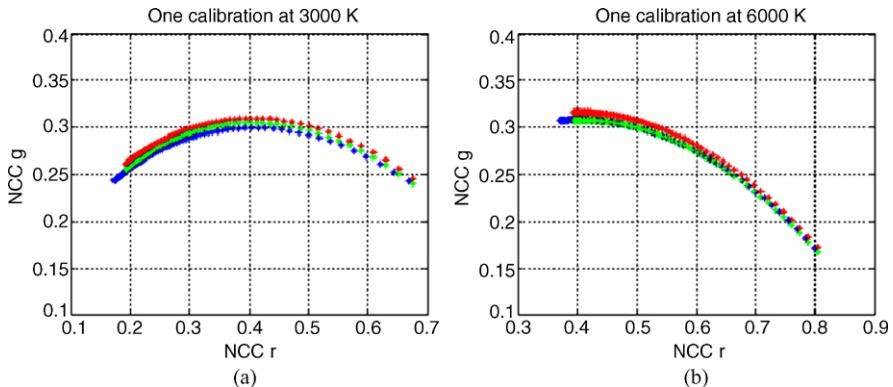


**Fig. 9.7** The color appearance shift is apparent in (a) and (b). The color temperature of the light sources increases from left to right. The arrow indicates the change in the color of the light. The limited dynamic response range causes distortion in color: pixels can saturate to a maximum value (the rightmost image at the upper row) or be under-exposed to zero (the leftmost image at the lower row)

### 9.3.2.2 Non-canonical Images and Colors

If images are not taken under the illumination used in camera calibration, the colors are distorted even more. The distortion will appear as a shift in colors, as can be seen in Fig. 9.7 which displays images taken under four different light sources while the camera is calibrated to one of them. In the upper image series, the camera was calibrated to the light source Horizon (first image on the left) and after light source was changed to incandescent A, TL84 and daylight, respectively. In the lower image series, the camera was calibrated to daylight (first image on the right) and then images were taken under TL84, A and Horizon.

The skin color tends to shift in the direction of illumination color change. More reddish prevailing illumination causes color shift towards red, while more bluish one adds blue components. Of course, a light source with strong spikes in spectra can cause additional distortions for certain colors. Since cameras have limited dynamic response ranges, the colors can be distorted also due to saturation or under-exposure.



**Fig. 9.8** The skin NCC chromaticities were simulated using the data of the Sony camera (see Fig. 9.2) and the skin reflectances from Fig. 9.4. **a** shows the possible skin chromaticities when the camera was calibrated to a Planckian of 3000 K and **b** when the calibration illumination was a Planckian of 6000 K. The chromaticity range depends on the calibration illumination and the possible color temperature range of prevailing illuminations

Manual or automatic brightness control in the camera can alleviate this problem, but manual operation tends to be tedious and automatic control might cause problems by itself.

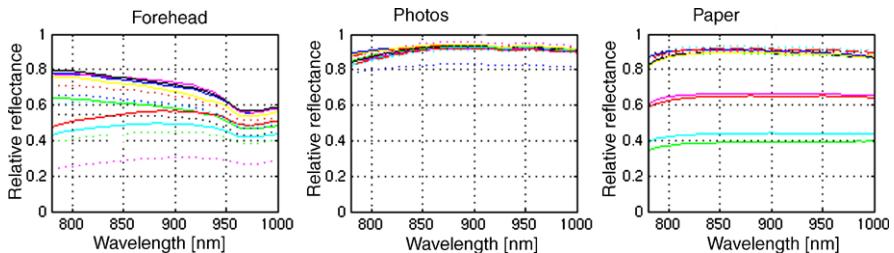
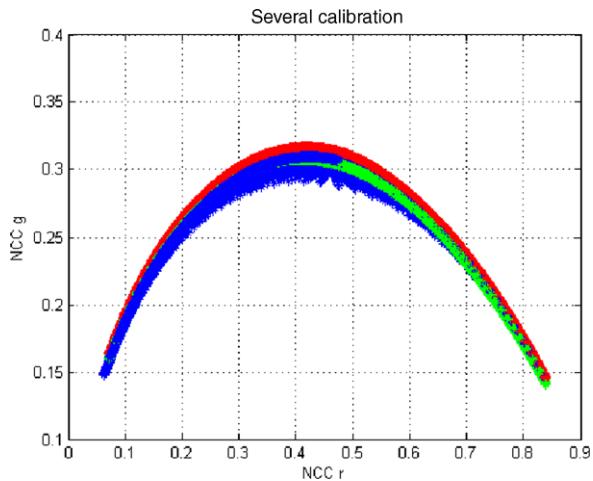
Figure 9.8 shows simulated skin chromaticities using only one calibration. The chromaticity range obtained depends on the calibration light and the color temperature range of the prevailing illumination. The possible range of skin colors (locus [44]) is affected by the amount of calibrations. Figure 9.8 shows that different white balancing illuminants have dissimilar ranges of possible skin chromaticities and produce separate skin locus. When the loci of all different calibrations are gathered together, a bigger locus is obtained, as shown in Fig. 9.9. Of course, the illumination range as well as different camera settings affect the locus size.

## 9.4 Separating Sources of Skin Data

Many materials, like inks and dyes, are used to imitate the appearance of skin. Some studies have been already done to examine how well the imitation works and how the real skin can be separated from imitation.

The skin data can come from different sources like real faces, photos or print [37]. The source cannot often be determined from normal RGB data, so spectral data is needed. An interesting spectral data region is near infrared. Figure 9.10 shows near infrared spectra for real faces, facial skin from photos and facial skin from a print of three different skin complexions. The spectra from photos and prints, which are flat, are clearly different from that of real faces. Thus simple ratio between two channels can be used to separate real skin from other sources. The level difference in real spectra between different complexions start to diminish as a degree of wavelength. Skin complexion groups are separable in print spectra, but not in photo spectra.

**Fig. 9.9** The skin locus formed with all prevailing illumination/white balancing combinations



**Fig. 9.10** Near infrared skin spectra from real faces (*left*) , photos (*middle*) and paper (*right*)

The skin color appearance for mannequins is also sought after, but it clearly is different from real skin [1]. Kim et al. [24] have studied the differences between masked fake faces and real skin. They concluded that wavelengths of 685 nm and 850 nm can be used to discriminate them.

## 9.5 Modeling Skin Colors

Skin color model is a description of possible skin tones. To create such a model, one has to first select the color space in which the model is formed, then the mathematical model to describe the possible skin colors, and finally, the data upon which the model is defined. The performance of the model depends on all these factors and is a trade-off between generality of the model and accuracy for a certain image.

Skin detection methods have been compared in several studies using different data [22, 35, 46]. The studies disagree, which might be because the optimality of the model depends on its purpose, data, material and modeling parameters.

### 9.5.1 Behavior of Skin Complexions at Different Color Spaces Under Varying Illumination

Color space in which skin data is processed, has also an effect on detection. Not all color spaces are equal: they can map RGB values differently, which can be used to separate certain colors. Even a mixture of color spaces can be used like in [47], at least for canonical or nearly canonical images.

As mentioned earlier, a color space conversion does not remove chromaticity shifts due to illumination or effects caused by noise. In fact, noise can be detrimental for low RGB values or near thresholds. The brightness control or lack of it can have a strong effect on the possible skin chromaticities. If there is no automatic brightness or gain controller, it is possible for one channel to have low values or even underclipping. Therefore, the skin colors have been studied under varying illuminations [34].

RGB coordinates are device-oriented, but they can be converted into human vision oriented spaces like XYZ or CIE Lab. A correct conversion requires an illumination-dependent transform matrix, including also the effect of device characteristics. Of course, there exist general transforms matrices. None of the matrix transforms reduce the effect of changing light since it has already affected RGBs.

The more device oriented color spaces can be classified, based on the conversion method, into two groups: those using linear transforms from RGB and those obtained via non-linear transforms. For example, linear transform based color spaces are: I1I2I3, YES, YIQ, YUV, YCrCb (Rec. 601–5 and 709). Among the nonlinear transforms are: NCC rgb, modified rgb, natural logarithm ln-chromaticity, P1P2, I1I2I3, ratios between channels (G/R, B/R, and B/G), HSV, HSL, modified ab, TLS and Yuv.

Overlap between different skin complexions vary in color spaces. In [34], the overlaps between two complexions (pale and yellowish) were compared in different color spaces and across different cameras: the overlaps between them were reasonably high in all color spaces (ranging from 50–75 percent) when using different canonical images. When using both canonical and uncanonical images, the overlap still increased due to the fact that more colors fall into the region. However, when comparing skin data from different cameras, the overlaps between skin RGBs were smaller and dependent on the cameras used in comparison. Therefore, one can argue that color spaces and cameras used do have an effect on skin detection and thus for face recognition.

### 9.5.2 Color Spaces for Skin

Several color spaces have been suggested for general skin color modeling, but thus far, none of them has been shown to be superior to the other. The list of comparison studies for color spaces can be found, for example, in [33] or [22]. However,

it seems that those spaces in which intensity is not separated so clearly from chromaticity are similar to RGB. The separation can be evaluated using linear or linearized RGB data: RGB is transformed into color space using substitution  $R \rightarrow cR$ ,  $G \rightarrow cG$ , and  $B \rightarrow cB$ , in which  $c$  describes a uniform change in the intensity levels. If the factor  $c$  does not cancel out for chromaticity descriptors, the separation is incomplete.

Normalized color coordinates (NCC) are quite often used in modeling, and they separates the intensity and chromaticity. To avoid the intensity changes, only the chromaticity coordinates are used. In [46], different color spaces are compared in terms of efficiency and transferability of the model. The performance of NCC and CIE xy was superior to several other skin color models. It was also shown in [55] that NCC has a good discrimination power. More details of color spaces for skin detection can be found in [33] or [22].

A color can be uniquely defined in by its intensity and two chromaticity coordinates since  $r + g + b = 1$ . The chromaticity coordinates for NCC color space are defined as

$$r = \frac{R}{R + G + B}, \quad (9.3)$$

$$g = \frac{G}{R + G + B}. \quad (9.4)$$

The intensity is canceled from chromaticity coordinates since they are calculated by dividing the descriptor value of the channel by the sum of all descriptor values (intensity) at that pixel.

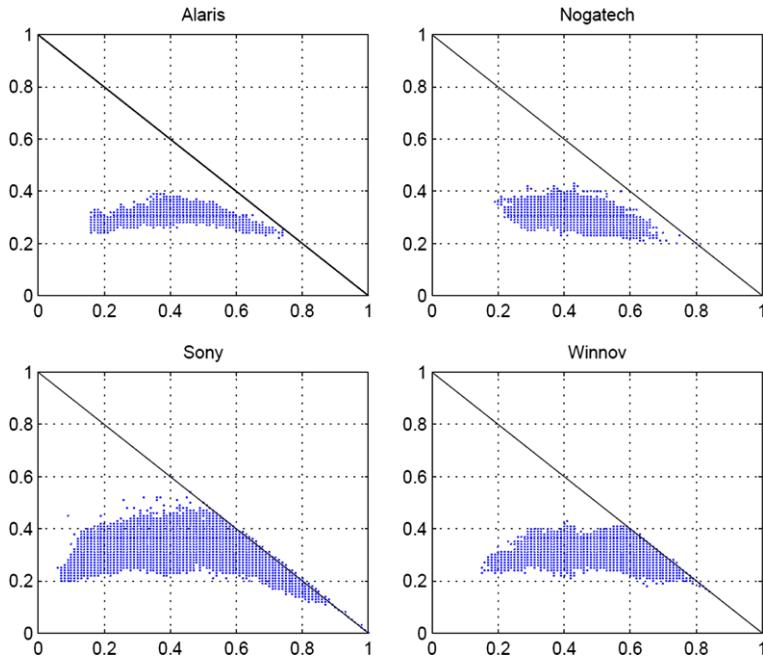
The modeling can be done using only the chromaticity coordinates to reduce the effect of illumination intensity changes, which are common in videos and images. Some models do include intensity (like in [14]), but more data is needed to construct the model and computational costs are increased due to a third component.

### 9.5.3 Skin Color Model and Illumination

Section 9.3 showed that illumination affects skin color both in canonical and uncanonical images. What is more, this dependency is camera-specific: the camera sensors and internal image preprocessing of the camera affect the color production and thus on the end results (see Fig. 9.11). Therefore, creating a universal model is difficult.

Many face detection algorithms assume that the images are taken under canonical or near canonical conditions. For many data sets, this is true. An example of this kind of image data set is a set of personal photos.

When the illumination varies, the previous approaches have a high risk of failure. Of course, the images can be subjected to color correction or color constancy algorithm, but sometimes this can lead even more serious color distortions [35].



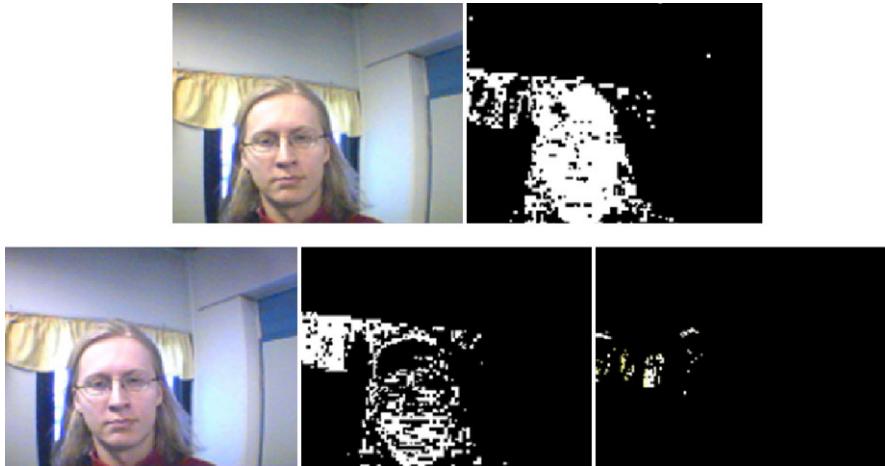
**Fig. 9.11** The camera and its properties determine the skin locus, as indicated by the loci of four cameras. However, some regions are common to all, most notable the region of skin tones

Color correction based approach has been suggested, for example, by Hsu et al. [15]: the colors in image are corrected so that the skin would appear in skin tones and after this segment the image using skin color model. The color correction is based on a pixel with a high brightness value which are assumed to belong to a white object. These pixels are used to calculate correction coefficient which are applied to the image. This approach can fail for many reasons like data loss due to saturation, or if a pixel with high brightness belongs to a nonwhite object. The latter case is demonstrated in Fig. 9.12.

For a more general skin model, one should use the knowledge of illumination changes, calibration and camera settings like in the skin locus-based approach [43]. The drawback of this model is that it is not so specific as canonical models—more color tones are included. Thus, more nonskin objects will be considered skin candidates. Since color itself is rarely enough to determine whether the target is skin or not, the face candidates are in case subjected for further processing.

#### 9.5.4 Mathematical Models for Skin Color

The model for skin color can be either a mathematically defined area in color space or a statistical approach in which a probability to belong skin is attached to color



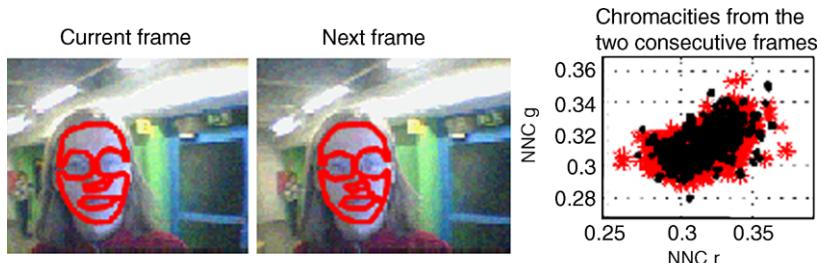
**Fig. 9.12** The *upper row* displays the color segmentation results using Hsu et al. model [15] without the color correction part. The *lower row* shows the segmentation with their color correction method. The color correction fails because the yellow curtains have the highest brightness values and is assumed to be a white object

tones. The model may be fixed or adaptive, and in the latter case, the update depends whether it is applied on single images or video frames. A more detailed review can be found, for example, in [33] or [22].

The area based approach uses a spatial constraint in the color space to define possible skin areas. The shape of the constraint can be simple thresholds like in [3] or a more complex shaped function like in [15]. Generally no thresholding is done, since the colors that fall inside the area are considered skin. These models often assume that skin has or can be corrected to have skin tone appearance. An exception is the skin locus in which the illumination changes are included in the model.

It is possible to adapt the model even for single images (e.g., [3, 26, 45]) although the successfulness depends on the validity of assumptions behind the adaptation criteria. The adaptation schema generally use a general skin model obtained from a representative image set and after that fine-tune into an image specific model. For example, in Cho et al. [3], the fine-tuning phase assumes that the skin color histogram is unimodal and skin color occurs mainly on real skin areas. This approach can fail if the image has dominant skin-colored, nonfacial object or the histogram is not unimodal.

The challenge of the probability-based approach is to be able to reliably find the probability distribution of skin colors. This requires collecting a representative data set of images for forming the model. An example of a statistical model is the one presented by Jones and Rehg [21]. They calculate the histogram and Gaussian models using over 1 billion labeled pixels. Many other statistical models like SOM or neural networks has been suggested and a review of them can be found, for example, in [33] or [22]. In addition to the statistical model, one has to determine the threshold limit for separating the skin from nonskin. It is difficult to automatically



**Fig. 9.13** Two consecutive frames are taken from a video sequence (the *first* and *second image from the left*). The facial skin areas of the frames are manually extracted and their skin RGB values are then converted to the NCC chromaticity space. The chromaticities from these two frames are marked with different colors in the rightmost image. As can be observed from the rightmost image, the chromaticities overlap significantly

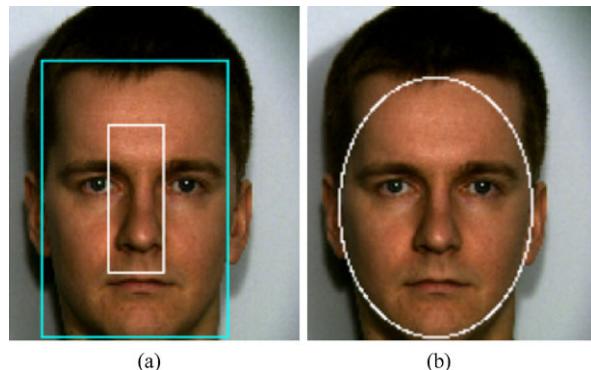
find the threshold value because the probability model found may not be valid for all images.

#### 9.5.4.1 Video Sequences

The processing of video sequences is similar to that of single, independent images. Thus, the skin detection presented earlier can also be used for videos. The fixed skin color models are suitable for videos in which changes in illumination are minimal. Generally, this is not the case and the skin color models need to be updated. The model adaptation relies often on the dependencies between consecutive frames, which is true for many videos: The consecutive frames often exhibit sequential dependency. This can be observed in Fig. 9.13: the overlap between the chromaticities from two consecutive frames is significant.

If the illumination changes between images are slow (no abrupt, drastic object color changes) or the person moves in a nonuniform illumination field slowly enough, the skin color model can adapt to the color changes. This required some constraint for selecting the pixels used in the model update. Three different adaptive schemes have been suggested: two of them use spatial constraints [39, 57] (see Fig. 9.14) and one skin locus [35]. The basic idea is the same: to use some constraint to select the pixels for model updating. The spatial constraints use different ideas to select candidate pixels from a located face: the method of Raja et al. [39] updates the skin color model using pixels inside the localized face area. The pixels are selected from an area which is 1/3 of the localization area and 1/3 from the localization boundaries. Yoo and Oh [57] argued that the localization should resemble the shape of the object (face) and they used all pixels inside the elliptical face localization. The skin locus can be used in two ways: either the whole locus or partial locus is used to select skin colored pixels from the localized face and its near surroundings.

**Fig. 9.14** Spatial constraints suggested for adaptive skin color modeling: the *left image* shows the method suggested by Raja et al. [39]. The *outer box* indicates the localized face while the pixels inside the *inner box* are used for model updating. The *image on the right* shows elliptical constraint by Yoo and Oh [57]



There are many possible methods for updating the skin color model, but perhaps a common method is the moving average, as presented in (9.5):

$$\check{M} = \frac{(1 - \alpha) * M_t + \alpha * M_{t-1}}{\max((1 - \alpha) * M_t + \alpha * M_{t-1})}, \quad (9.5)$$

where  $\check{M}$  is a new, refreshed model,  $M$  is the model,  $t$  is the frame number and  $\alpha$  is a weighting factor. Quite often, the weighting factor is set to 0.5 to get equal emphasis on the skin color model of current and previous frames. The moving average method provides a smooth transition between models from different frames. It also reduces the effect of noise, which can change pixel color without any variation in external factors and thus be detrimental to the models.

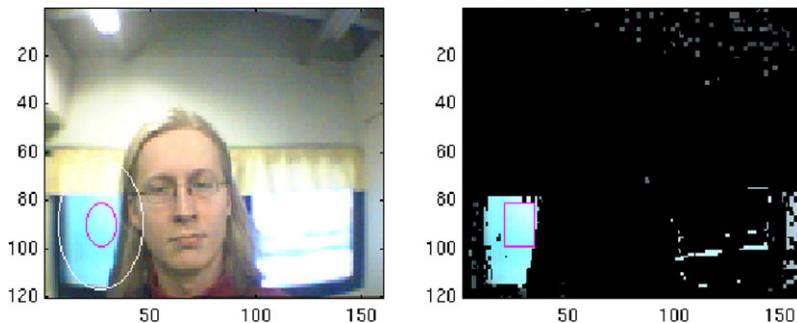
However, the spatial constraint models have been shown to be very sensitive to localization errors, therefore, they can easily adapt to nonskin objects [35]. The failure due to these constraints can happen even under a fairly moderate illumination change. In Fig. 9.15, Raja et al.'s method has failed while tracking a face on a video sequence and the skin color model is adapted to nonskin colored target, as shown in this image.

The constraint suggested by Raja et al. easily fails under a nonuniform illumination field change, as demonstrated in Fig. 9.16. The model is updated using the pixel inside the localization and therefore, it can adapt only to global illumination changes, but not to the nonuniform illumination field variation.

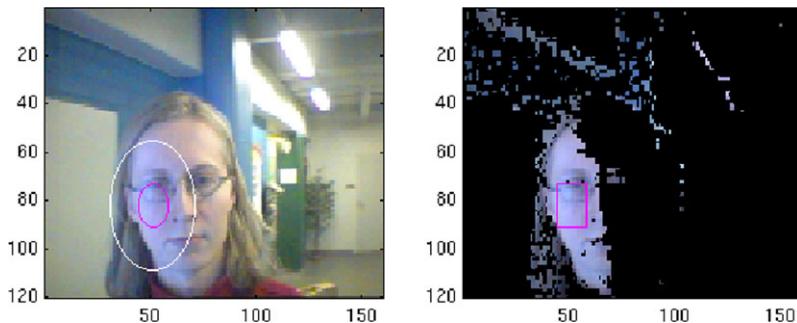
The correct localization of face is not so sensitive for a skin locus based approach since the nonskin colored pixels can be filtered out. Large skin colored objects connected to the face are problematic and cues other than color are needed to solve this.

## 9.6 Color Cue for Face Detection

As mentioned above, color is a useful cue for face detection as it can greatly reduce the search area by selecting only the skin-like regions. However, it is obvious that



**Fig. 9.15** The face tracking based on Raja et al.'s method failed and adapted to a nonfacial target. The *left image* displays the “localized face”. The *right image* shows the pixels selected by the current skin color model. The *red box* shows the pixels used for refreshing the model



**Fig. 9.16** The constraint suggested by Raja et al.'s selects a nonrepresentative set of skin pixels

the use of skin color only is not enough to distinguish between faces and other objects with a skin-like appearance (such as hands, wood, etc.). Therefore, other procedures are needed to verify whether the selected regions are (or contain) faces or not. Depending on the robustness of the skin model and changes in the illumination conditions, one can notice two cases:

- Case #1: The initial skin color detection step produces consistently reliable results. The skin color model is valid for the illumination conditions, the camera and its settings. The skin color model can be designed either for stable, controlled illumination (typical case) or for variable illumination (skin locus). In such cases, it is generally enough to consider each connected resultant component from the skin detection as a face candidate. Then, one can verify the “faceness” of the candidate by simple and fast heuristics.
- Case #2: The initial skin color detection step produces unsatisfactory results or even fails. In this case, the skin color model does not correspond to the prevailing illumination, used camera or settings of the camera. One can hope that the results would indicate the locations of the faces, but their size estimation is too unreliable. Therefore, a different method for face detection (either an appearance-based

or feature-based one) should be used when searching for the faces in and around the detected skin regions.

In both cases, the use of color accelerates the detection process. In the following, we review some methods based on color information for detecting faces. Most of the color-based face detectors start by determining the skin pixels which are then grouped using connected component analysis. Then, for each connected component, the best fit ellipse is computed using geometric moments, for example. The skin components which verify some shape and size constraints are selected as face candidates. Finally, features (such as eyes and mouth) are searched for inside each face candidate based on the observation that holes inside the face candidate are due to these features being different from skin color. Therefore, most of the color-based face detection methods mainly differ in the selection of the color space and the design of the skin model. In this context, as seen in Sect. 9.5, many methods for skin modeling in different color spaces have been proposed. For comparison studies, refer to [35, 46] and [34].

Among the works using color for face detection is Hsu et al.'s system which consists of two major modules: (1) face localization for finding face candidates, and (2) facial feature detection for verifying detected face candidates [15]. For finding the face candidates, the skin tone pixels are labeled using an elliptical skin model in the  $YC_bC_r$  color space, after applying a lighting compensation technique. The detected skin tone pixels are iteratively segmented using local color variance into connected components which are then grouped into face candidates. Then, the facial feature detection module constructs eye, mouth and face boundary maps to verify the face candidates. Good detection results have been reported on several test images. However, no comparative study has been made thus far.

In [7], Garcia and Tziritas presented another approach for detecting faces in color images. First, color clustering and filtering using approximations of the  $YC_bC_r$  and  $HSV$  skin color subspaces are applied to the original image, providing quantized skin color regions. Then a merging stage is iteratively performed on the set of homogeneous skin color regions in the color quantized image, in order to provide a set of face candidates. Finally, constraints related to shape and size of faces are applied, and face intensity texture is analyzed by performing a wavelet packet decomposition on each face area candidate in order to detect human faces. The authors have reported a detection rate of 94.23% and a false dismissal rate of 5.76% on a data set of 100 images containing 104 faces. Though the method can handle nonconstrained scene conditions, such as the presence of a complex background and uncontrolled illumination, its main drawback lies on that fact that it is computationally expensive due to its complicated segmentation algorithm and time-consuming wavelet packet analysis.

Sobottka and Pitas presented a method for face localization and facial feature extraction using shape and color [42]. First, color segmentation in HSV space is performed to locate skin-like regions. After facial feature extraction, connected component analysis and best fit ellipse calculation, a set of face candidates are obtained. To verify the “faceness” of each candidate, a set of eleven lowest-order-geometric



**Fig. 9.17** Examples of face detection results using the color-based face detector in [10]

moments is computed and used as inputs to a neural network. The authors reported a detection rate of 85% on a test set of 100 images.

In [11], Haiyuan et al. presented a different approach for detecting faces in color images. Instead of searching for facial features to verify the face candidates, the authors modeled the face pattern as a composition of a skin part and a hair part. They made two fuzzy models to describe the skin color and hair color in CIE XYZ color space. The two models are used to extract the skin color regions and the hair color regions which are compared with the prebuilt head-shape models by using a fuzzy theory based pattern-matching method to detect the faces.

In [10], Hadid et al. presented an efficient color-based face detector, using the skin locus model to extract skin-like region candidates, and then performing the selection by simple yet efficient refining stages. After ellipse fitting and orientation normalization, a set of criteria (face symmetry, presence of some facial features, variance of pixel intensities and connected component arrangement) are evaluated to keep only facial regions. The refining stages are organized in a cascade to achieve high accuracy and to keep the system fast. The system was able to detect faces and deal with different conditions (size, orientation, illumination and complex background). Figure 9.17 shows some detection examples performed by the system under different conditions.

Several other approaches using color information for detecting and tracking faces and facial features in still images and video sequences have been proposed [13, 54].



**Fig. 9.18** Examples of face detection results using the color-based face detector in [9]

It appears that most of the methods have not been tested under practical illumination changes (usually only mild changes are considered), which makes them belonging to the first category (Case #1) described above.

More recently, to detect faces in natural and unconstrained environments, Hadid and Pietikänen [9] proposed an approach which considers the fact that color is a very powerful and useful cue for face detection, but unfortunately, it may also produce unsatisfactory results or even fail. The proposed approach consists of first preprocessing the images to find the potential skin regions, avoiding thus scanning the whole image when searching for faces, and then performing an exhaustive search in and around the detected skin regions. The exhaustive search is performed using a two-stage SVM based approach, exploiting the discrimination power of the Local Binary Patterns (LBP) features. The obtained results are interesting in the sense that the proposed approach inherits the speed from the color-based methods and the efficiency from the gray scale-based ones. Some detection results are shown in Fig. 9.18.

One problem of color-based face detectors lies in the fact that they are generally camera specific. Most of the methods have reported their results on specific and limited data sets and this fact does not facilitate performing a comparative analysis between the methods. Among the attempts to define a standard protocol and a common database for testing color-based face detector is the work of Sharma and Reilly [41].

Currently, most methods for face detection rely only on gray scale information even when color images are available. Generally these methods scan the images at all possible locations and scales and then classify the sub-windows either as face or nonface, yielding in more robust but also computationally more expensive processing methods, especially with large-sized images. Among robust approaches based only on gray scale information is Viola and Jones's approach [49]. The approach uses Haar-like features and AdaBoost as a fast training algorithm. AdaBoost is used to select the most prominent features among a large number of extracted features and construct a strong classifier from boosting a set of weak classifiers. Such systems

generally run in real-time for small-sized images (e.g.,  $240 \times 320$  pixels), but tend to be slow for larger images. Including other cues such color or motion information may thus be very useful for speeding-up the detection process.

## 9.7 Color Cue for Face Recognition

The role of color information in the recognition of nonface objects has been the subject of much debate. However, there has only been a small amount of work which examines its contribution to face recognition. Most of the work has only focused on the luminance structure of the face, thus ignoring color cues, due to several reasons.

The first reason lies in the lack of evidence from human perception studies about the role of color in face recognition. Indeed, a notable study in this regard was done in [23], in which the authors found that the observers were able to quite normally process even those faces that had been subjected to hue-reversals. Color seemed to contribute no significant recognition advantages beyond the luminance information. In another piece of work [56], it is explained that the possible reason for a lack of observed color contribution in these studies is the availability of strong shape cues which make the contribution of color not very evident. The authors then investigated the role of color by designing experiments in which the shape cues were progressively degraded. They concluded that the luminance structure of the face is undoubtedly of great significance for recognition, but that color cues are not entirely discarded by the face recognition process. They suggested that color does play a role under degraded conditions by facilitating low-level facial image analysis such as better estimations of the boundaries, shape and sizes of facial features [56].

A second possible reason for a lack of work on color-based face recognition relates to the difficulties of associating illumination with white balancing of cameras. Indeed, as discussed in Sect. 9.3, illumination is still a challenging problem in automatic face recognition, therefore, there is no need to further complicate the task.

A third possible reason for ignoring color cues in the development of automatic recognition systems is the lack of color image databases<sup>1</sup> available for the testing of the proposed algorithms, in addition to the unwillingness to develop methods which cannot be used with the already existing monochrome databases and applications.

However, the few attempts to use color in automatic face recognition includes the work conducted by Torres et al. [48] who extended the eigenface approach to color by computing the principal components from each color component independently in three different color spaces (RGB, YUV and HSV). The final classification is achieved using a weighted sum of the Mahalanobis distances computed for each color component. In their experiments using one small database (59 images), the authors noticed performance improvements for the recognition rates when using YUV (88.14%) and HSV (88.14%) color spaces, while a RGB color space provided the

---

<sup>1</sup>Note that recently some color image databases have finally been collected (e.g., the color FERET database and the FRGC version 2 database).

same results (84.75%) when using  $R$ ,  $G$  or  $B$  separately and exactly the same results as using the luminance  $Y$  only. Therefore, they concluded that color is important for face recognition. However, the experiments are very limited, as only one small face database is used and the simple eigenface approach is tested.

In another piece of work that deals with color for face recognition [20], it has been argued that a performance enhancement could be obtained if a suitable conversion from color images to a monochromatic form would be adopted. The authors derived a transformation from color to gray-scale images using three different methods (PCA, linear regression and genetic algorithms). They compared their results with those obtained after converting the color images to a monochromatic form by using a simple transformation  $I = \frac{R+G+B}{3}$ , and they noticed a performance enhancement of 4% to 14% using a database of 280 images. However, the database considered in the experiments is rather small, thus, one should test the generalization performance of the proposed transformation on a larger set of images from different sources.

In [40], Rajapakse et al. considered an approach based on Nonnegative Matrix Factorization (NMF) and compared the face recognition results using color and gray scale images. On a test set of 100 face images, the authors have claimed a performance enhancement when using also color information for recognition.

In [19], Jones has attempted to extend the Gabor-based approach for face recognition to color images by defining the concept of quaternions (four component hypercomplex numbers). On a relatively limited set of experiments, the author has reported a performance enhancement on the order of 3% to 17% when using the proposed quaternion Gabor-based approach instead of the conventional monochromatic Gabor-based method.

Very recently, color face recognition has been revisited by many researchers, with an aim to discover the efficient use of color for boosting the face recognition performance. For instance, inspired by the psychophysical studies indicating that color does play a role in recognizing faces under degraded conditions, Choi et al. [58] carried out extensive experiments and studied the effect of color information on the recognition of low-resolution face images (e.g., less than  $20 \times 20$  pixels). By comparing the performance of grayscale and color features, the results showed that color information can significantly improve the recognition performance.

Yang et al. [55] compared the discriminative power of several color spaces for face recognition and found out that different color spaces display different discriminating power. Experiments on a large scale face recognition grand challenge (FRGC) problem also revealed that the RGB and XYZ color spaces are weaker than the I1I2I3, YUV, YIQ color spaces for face recognition. The authors proposed then color space normalization techniques for enhancing the discriminative power of different color spaces.

For color based face verification, Chan et al. [2] proposed a discriminative descriptor encoding the color information of the face images. The descriptor is formed by projecting the local face image acquired by multispectral LBP operators, into LDA space. The overall similarity score is obtained by fusing local similarity scores of the regional descriptors. The method has been tested on the XM2VTS and FRGC 2.0 databases with very promising results.

Liu and his colleagues extensively investigated the problem of color face recognition and reported very good results on FRGC database (Version 2 Experiment 4) [27–30, 52, 53]. For instance, in [27], the authors first derived new (uncorrelated, independent and discriminating) color spaces from the RGB color space by means of linear transformations. Then, vectors are formed in these color spaces by concatenating their component images to form augmented pattern vectors, whose dimensionality is reduced by PCA. Finally, an enhanced Fisher model (EFM) is used for recognition. The obtained results are better than those of methods using grayscale or RGB color images. In [29], the authors considered a hybrid color space by combining the  $R$  component image of the RGB color space and the chromatic components  $I$  and  $Q$  of the YIQ color space. Experiments on the Face Recognition Grand Challenge (FRGC) version 2 Experiment 4 showed the hybrid color space significantly improves face recognition performance due to the complementary characteristics of its component images. Since most of the experiments conducted by Liu and his team were mainly using the FRGC database, it is of interest to see how well the proposed methods generalize to other databases and settings.

## 9.8 Conclusions

Color is a useful cue in facial image analysis. Its use for skin segmentation and face detection is probably the most obvious, while its contribution to face recognition is not very clear. The first important issues when planning the use of color in facial image analysis are the selection of a color space and the design of a skin model. Several approaches have been proposed for these purposes, but unfortunately, there is no optimal choice. The choice made depends on the requirement of the application and also on the environment (illumination conditions, camera calibration, etc.).

Once a skin model has been defined, the contribution of color to face detection, not surprisingly, plays an important role in pre-processing the images and in the selection of the skin-like areas. Then, other refining stages can also be launched in order to find faces among skin-like regions. Color-based face detectors could be significantly much faster than other detectors which are based solely on gray-scale information, especially with large-sized images.

In relation to the contribution of color to face recognition, the issue is still under debate and among the open questions are: is color information useful for face recognition at all? If yes, how the three different spectral channels of face images should be combined to take advantages of the color information? What is the optimal color space which provides the highest discriminative power, etc.? The current results suggest that color cue has not yet shown its full potential and need further investigation. Therefore, it perhaps makes sense for current automatic face recognition systems not to rely on color for recognition because its contribution is not well established yet.

**Acknowledgements** The financial support of the Academy of Finland is gratefully acknowledged. J.B. Martinkauppi thanks Field NIRce which was sponsored by Botnia Atlantica, a project of the European Regional Development Fund.

## References

1. Angelopoulou, E., Molana, R., Daniilidis, K.: Multispectral skin color modeling. In: Proc. IEEE Computer Society's Computer Vision and Pattern Recognition, pp. 635–642, December 2001
2. Chan, C., Kittler, J.V., Messer, K.: Multi-scale local binary pattern histograms for face recognition. In: ICB07, pp. 809–818 (2007)
3. Cho, K., Jang, J., Hong, K.: Adaptive skin-color filter. *Pattern Recognit.* **34**(5) (2001)
4. Do, H.C., You, J., Chien, S.: Skin color detection through estimation and conversion of illuminant color using sclera region of eye under varying illumination. In: Proc. 18th International Conference on Pattern Recognition, pp. 327–330, August 2006
5. Edwards, E.A., Duntley, S.: The pigments and color of living human skin. *Am. J. Anat.* **65**(1), 1–33 (1939)
6. Funt, B., Barnard, K., Martin, L.: Is machine colour constancy good enough. In: Proceedings of 5th European Conference on Computer Vision, pp. 445–459, June 1998
7. Garcia, C., Tziritas, G.: Face detection using quantized skin color regions merging and wavelet packet analysis. *IEEE Trans. Multimed.* **1**(3), 264–277 (1999)
8. Graf, H.P., Chen, T., Petajan, E., Cosatto, E.: Locating faces and facial parts. In: Proceedings of 1st International Workshop Automatic Face and Gesture Recognition, pp. 41–46, May 1995
9. Hadid, A., Pietikäinen, M.: A hybrid approach to face detection under unconstrained environments. In: Proc. 18th International Conference on Pattern Recognition (ICPR), vol. 1, p. 4, Hong Kong (2006)
10. Hadid, A., Pietikäinen, M., Martinkauppi, B.: Color-based face detection using skin locus model and hierarchical filtering. In: 16th International Conference on Pattern Recognition, pp. 196–200, Quebec, August 2002
11. Haiyuan, W., Qian, C., Yachida, M.: Face detection from color images using a fuzzy pattern matching method. *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(6), 557–563 (1999)
12. Harwood, L.A.: A chrominance demodulator ic with dynamic flesh correction. *IEEE Trans. Consum. Electron.* **CE-22**, 111–117 (1976)
13. Hjelmas, E., Low, B.K.: Face detection: A survey. *Comput. Vis. Image Underst.* **83**(3), 236–274 (2001)
14. Hsu, R.L.: Face detection and modeling for recognition. PhD thesis, Michigan State University (2002)
15. Hsu, R.-L., Abdel-Mottaleb, M., Jain, A.K.: Face detection in color images. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(5) (2002)
16. Hsu, E., Mertens, T., Paris, S., Avidan, S., Durand, F.: Light mixture estimation for spatially varying white balance. *ACM Trans. Graph. (TOG)* **27**(3) (2008)
17. Hunke, M., Waibel, A.: Face locating and tracking for human-computer interaction. In: Proceedings of 1994 Conference Record of the Twenty-Eighth Asilomar Conference on Signals, Systems and Computers, pp. 1277–1281 October 1994
18. Imai, F.H., Tsumura, N., Haneishi, H., Miyake, Y.: Principal component analysis of skin color and its application to colorimetric reproduction on CRT display and hardcopy. *J. Imaging Sci. Technol.* **40**(5) (1996)
19. Jones, C.F.: Color face recognition using quaternionic Gabor wavelets. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia (2005)
20. Jones, C.F., Abbott, A.L.: Optimization of color conversion for face recognition. *EURASIP J. Appl. Signal Process.* **4**, 522–529 (2004)
21. Jones, M., Rehg, J.: Statistical color models with application to skin detection. *Int. J. Comput. Vis.* **46**(1) (2002)
22. Kakumanu, P., Makrigiannis, S., Bourbakis, N.: A survey of skin-color modeling and detection methods. *Pattern Recognit.* **40**(3) (2007)
23. Kemp, R., Pike, G., White, P., Musselman, A.: Perception and recognition of normal and negative faces: the role of shape from shading and pigmentation cues. *Perception* **25**(1), 37–52 (1996)

24. Kim, Y.S., Na, J., Yoon, S., Yi, J.: Masked fake face detection using radiance measurements. *JOSA-A* **26**(4) (2009)
25. Lee, E., Ha, Y.: Automatic flesh tone reappearance for color enhancement in TV. *IEEE Trans. Consum. Electron.* **43**(4), 1153–1159 (1997)
26. Li, B., Xue, X., Fan, J.: A robust incremental learning framework for accurate skin region segmentation in color images. *Pattern Recognit.* **40**(12) (2007)
27. Liu, C.: Learning the uncorrelated, independent, and discriminating color spaces for face recognition. *IEEE Trans. Inf. Forensics Secur.* **3**(2), 213–222 (2008)
28. Liu, Z., Liu, C.: Fusion of the complementary discrete cosine features in the yiq color space for face recognition. *Comput. Vis. Image Underst.* **111**(3), 249–262 (2008)
29. Liu, Z., Liu, C.: A hybrid color and frequency features method for face recognition. *IEEE Trans. Image Process.* **17**(10), 1975–1980 (2008)
30. Liu, Z., Liu, C.: Robust face recognition using color information. In: ICB, pp. 122–131 (2009)
31. Marszalec, E., Martinkauppi, B., Soriano, M., Pietikäinen, M.: A physics-based face database for color research. *J. Electron. Imaging* **9**(1), 32–38 (2000)
32. Martinkauppi, B., Finlayson, G.: Designing a simple 3-channel camera for skin detection. In: Proc. the 12th Color Imaging Conference: Color Science and Engineering: Systems, Technologies, and Applications, pp. 151–156, November 2004
33. Martinkauppi, B., Pietikäinen, M.: Facial skin color modeling. In: Li, S.Z., Jain, A.K. (eds.) *Handbook of Face Recognition*, pp. 109–131. Springer, Berlin (2005)
34. Martinkauppi, B., Soriano, M., Laaksonen, M.: Behavior of skin color under varying illumination seen by different cameras in different color spaces. In: *Machine Vision in Industrial Inspection IX*. Proc. SPIE, vol. 4301, pp. 102–113 (2001)
35. Martinkauppi, B., Soriano, M., Pietikäinen, M.: Comparison of skin color detection and tracking methods under varying illumination. *J. Electron. Imaging* **14**(4) (2005)
36. Martinkauppi, B., Hadid, A., Pietikäinen, M.: Color cue in facial image analysis. In: Lukac, R., Plataniotis, K. (eds.) *Color Image Processing: Methods and Applications*, pp. 285–308. CRC Press, Boca Raton (2006)
37. Martinkauppi, B., Lehtonen, J., Parkkinen, J.: Near-infrared images of skin. In: Proc. 4th European Conference on Colour in Graphics, Imaging, and Vision, 10th International Symposium on Multispectral Colour Science, pp. 508–511, June 2008
38. Nakai, H., Manabe, Y., Inokuchi, S.: Simulation and analysis of spectral distribution of human skin. In: Proc. 14th International Conference on Pattern Recognition, pp. 1065–1067 (1998)
39. Raja, Y., McKenna, S., Gong, G.: Tracking and segmenting people in varying lighting conditions using colour. In: Proceedings of IEEE 3rd International Conference on Automatic Face and Gesture Recognition, pp. 228–233, April 1998
40. Rajapakse, M., Tan, J., Rajapakse, J.: Color channel encoding with NMF for face recognition. In: IEEE Conference on Image Processing, vol. 3, pp. 2007–2010 (2004)
41. Sharma, P., Reilly, R.: A colour face image database for benchmarking of automatic face detection algorithms. In: EC-VIP-MC 2003 4th EURASIP Conference focused on Video/Image Processing and Multimedia Communications, pp. 423–428 (2003)
42. Sobottka, K., Pitas, I.: Face localization and facial feature extraction based on shape and color information. In: IEEE Conference on Image Processing, vol. 3, pp. 483–486 (1996)
43. Soriano, M., Martinkauppi, B., Huovinen, S., Laaksonen, M.: Adaptive skin color modeling using the skin locus for selecting training pixels. *Pattern Recognit.* **36**(3), 681–690 (2003)
44. Störring, M., Andersen, H.J., Granum, E.: Physics-based modelling of human skin colour under mixed illuminants. *J. Robot. Auton. Syst.* **35**(3–4), 131–142 (2001)
45. Sun, H.: Skin detection for single images using dynamic skin color modeling. *Pattern Recognit.* **43**(4) (2010)
46. Terrillon, J.C., Shirazi, M., Fukamachi, H., Akamatsu, S.: Comparative performance of different skin chrominance models and chrominance spaces for the automatic detection of human face in color images. In: IEEE Int. Conf. on Automatic Face and Gesture Recognition, pp. 54–61 (2000)
47. Tomaschitz, J.A., Facon, J.: Skin detection applied to multi-racial images. In: Proc. 16th International Conference on Systems, Signals and Image Processing IWSSIP, pp. 1–3, June 2009

48. Torres, L., Reutter, J., Lorente, L.: The importance of the color information in face recognition. In: IEEE Conference on Image Processing, vol. 3, pp. 627–631 (1999)
49. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. Conf. Computer Vision and Pattern Recognition, pp. 511–518 (2001)
50. Wyszecki, G., Stiles, W.S. (eds.): *Color Science Concepts and Methods, Quantitative Data and Formulae*, 2nd edn. Wiley, New York (2000)
51. Yang, M.H., Ahuja, N.: Detecting human faces in color images. In: Proceedings of International Conference on Image Processing, pp. 127–130 (1998)
52. Yang, J., Liu, C.: A general discriminant model for color face recognition. In: ICCV, pp. 1–6 (2007)
53. Yang, J., Liu, C.: Color image discriminant models and algorithms for face recognition. *IEEE Trans. Neural Netw.* **19**(12), 2088–2098 (2008)
54. Yang, M.-H., Kriegman, D.J., Ahuja, N.: Detecting faces in images: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**, 34–58 (2002)
55. Yang, J., Liu, C., Zhang, L.: Color space normalization: Enhancing the discriminating power of color spaces for face recognition. *Pattern Recognit.* **43**(4), 1454–1466 (2010)
56. Yip, A.W., Sinha, P.: Contribution of color to face recognition. *Perception* **31**(8), 995–1003 (2002)
57. Yoo, T., Oh, I.: A fast algorithm for tracking human faces based on chromaticity histograms. *Pattern Recognit. Lett.* **20**(10) (1999)
58. Young, C.J., Man, R.Y., Plataniotis, K.N.: Color face recognition for degraded face images. *IEEE Trans. Syst. Man Cybern., Part B, Cybern.* **39**(5), 1217–1230 (2009)



# Chapter 10

## Face Aging Modeling

Unsang Park and Anil K. Jain

### 10.1 Introduction

Face recognition accuracy is typically limited by the large intra-class variations caused by factors such as pose, lighting, expression, and age [16]. Therefore, most of the current work on face recognition is focused on compensating for the variations that degrade face recognition performance. However, facial aging has not received adequate attention compared to other sources of variations such as pose, lighting, and expression.

Facial aging is a complex process that affects both the shape and texture (e.g., skin tone or wrinkles) of a face. The aging process appears in different manifestations in different age groups, gender and ethnicity. While facial aging is mostly represented by the facial growth in younger age groups (e.g., below 18 years of age), it is mostly represented by relatively large texture changes and minor shape changes (e.g., due to the change of weight or stiffness of skin) in older age groups (e.g., over 18 years of age). Therefore, an age invariant face recognition scheme needs to be able to compensate for both types of aging process.

Some of the face recognition applications where age invariance or correction is required include (i) identifying missing children, (ii) screening for watch list, and (iii) multiple enrollment detection problems. These three scenarios have two common characteristics: (i) a significant age difference exists between probe and gallery images (images obtained at verification and enrollment stages, respectively) and (ii) an inability to obtain a user's face image to update the template (gallery). Identifying missing children is one of the most apparent applications where age compensation is needed to improve the recognition performance. In screening applications, aging is a major source of difficulty in identifying suspects in a watch list. Repeat

---

U. Park (✉) · A.K. Jain  
Michigan State University, East Lansing, MI 48824, USA  
e-mail: [parkunsa@cse.msu.edu](mailto:parkunsa@cse.msu.edu)

A.K. Jain  
e-mail: [jain@cse.msu.edu](mailto:jain@cse.msu.edu)

offenders commit crimes at different time periods in their lives, often starting as a juvenile and continuing throughout their lives. It is not unusual to encounter a time lapse of ten to twenty years between the first (enrollment) and subsequent (verification) arrests. Multiple enrollment detection for issuing government documents such as driver licenses and passports is a major problem that various government and law enforcement agencies face in the facial databases that they maintain. Face or some other types of biometric traits (e.g., fingerprint or iris) are the only ways to reliably detect multiple enrollments.

Ling et al. [10] studied how age differences affect the face recognition performance in a real passport photo verification task. Their results show that the aging process does increase the recognition difficulty, but it does not surpass the challenges posed due to change in illumination or expression. Studies on face verification across age progression [19] have shown that: (i) simulation of shape and texture variations caused by aging is a challenging task, as factors like life style and environment also contribute to facial changes in addition to biological factors, (ii) the aging effects can be best understood using 3D scans of human head, and (iii) the available databases to study facial aging are not only small but also contain uncontrolled external and internal variations (e.g., pose, illumination, expression, and occlusion). It is due to these reasons that the effect of aging in facial recognition has not been as extensively investigated as other factors that lead to large intra-class variations in facial appearance.

Some biological and cognitive studies on face aging process have also been conducted, see [18, 25]. These studies have shown that cardioidal strain is a major factor in the aging of facial outlines. Such results have also been used in psychological studies, for example, by introducing aging as caricatures generated by controlling 3D model parameters [12]. Patterson et al. [15] compared automatic aging simulation results with forensic sketches and showed that further studies in aging are needed to improve face recognition techniques. A few seminal studies [20, 24] have demonstrated the feasibility of improving face recognition accuracy by simulated aging. There has also been some work done in the related area of age estimation using statistical models, for example, [8, 9]. Geng et al. [7] learn a subspace of aging pattern based on the assumption that similar faces age in similar ways. Their face representation is composed of face texture and the 2D shape represented by the coordinates of the feature points as in the Active Appearance Models. Computer graphics community has also shown facial aging modeling methods in 3D domain [22], but the effectiveness of the aging model was not evaluated by conducting a face recognition test.

Table 10.1 gives a brief comparison of various methods for modeling aging proposed in the literature. The performance of these models is evaluated in terms of the improvement in the identification accuracy. When multiple accuracies were reported in any of the studies under the same experimental setup (e.g., due to different choice of probe and gallery), their average value is listed in Table 10.1; when multiple accuracies are reported under different approaches, the best performance is reported. The identification accuracies of various studies in Table 10.1 cannot be directly compared due to the differences in the database, the number of subjects

**Table 10.1** A comparison of various face aging models [13]

	Approach	Face matcher	Database (#subjects, #images) in probe and gallery	Rank-1 identification accuracy (%)	
				Original image	After aging model
Ramanathan et al. (2006) [20]	Shape growth modeling up to age 18	PCA	Private database (109, 109)	8.0	15.0
Lanitis et al. (2002) [8]	Build an aging function in terms of PCA coefficients of shape and texture	Mahalanobis distance, PCA	Private database (12, 85)	57.0	68.5
Geng et al. (2007) [7]	Learn aging pattern on concatenated PCA coefficients of shape and texture across a series of ages	Mahalanobis distance, PCA	FG-NET* (10, 10)	14.4	38.1
Wang et al. (2006) [26]	Build an aging function in terms of PCA coefficients of shape and texture	PCA	Private database (NA, 2000)	52.0	63.0
Patterson et al. (2006) [14]	Build an aging function in terms of PCA coefficients of shape and texture	PCA	MORPH+ (9, 36)	11.0	33.0
Park et al. [13]	Learn aging pattern based on PCA coefficients in separated 3D shape and texture given 2D database	FaceVACS	FG-NET** (82, 82) MORPH-Album1 <sup>++</sup> (612,612) BROWNS (4, 4)—probe (100, 100)—gallery	26.4 57.8 15.6	37.4 66.4 28.1

\*Used only a very small subset of the FG-NET database that contains a total of 82 subjects

+Used only a very small subset of the MORPH database that contains a total of 625 subjects

\*\*Used all the subjects in FG-NET

++Used all the subjects in MORPH-Album1 which have multiple images

and the underlying face recognition method used for evaluation. Usually, the larger the number of subjects and the larger the database variations in terms of age, pose, lighting and expression, the smaller the recognition performance improvement by an aging model. The identification accuracy for each approach in Table 10.1 before aging simulation indicates the difficulty of the experimental setup for the face recognition test as well as the limitations of the face matcher.

Compared with other published approaches, the aging model proposed by Park et al. [13] has the following features.

- 3D aging modeling: Includes a pose correction stage and a more realistic model of the aging pattern in the *3D domain*. Considering that the aging is a 3D process, 3D modeling is better suited to capture the aging patterns. Their method is the only viable alternative to building a 3D aging model directly, as no 3D aging database is currently available. Scanned 3D face data rather than reconstructed is used in [22], but they were not collected for aging modeling and hence, do not contain as much aging information as the 2D facial aging database.
- Separate modeling of shape and texture changes: Three different modeling methods, namely, shape modeling only, separate shape and texture modeling and combined shape and texture modeling (e.g., applying 2nd level PCA to remove the correlation between shape and texture after concatenating the two types of feature vectors) were compared. It has been shown that the separate modeling is better than combined modeling method, given the FG-NET database as the training data.
- Evaluation using a state-of-the-art commercial face matcher, FaceVACS: A state-of-the-art face matcher, FaceVACS from Cognitec [4] has been used to evaluate the aging model. Their method can thus be useful in practical applications requiring an age correction process. Even though their method has been evaluated only on one particular face matcher, it can be used directly in conjunction with any other 2D face matcher.
- Diverse Databases: FG-NET has been used for aging modeling and the aging model has been evaluated on three different databases: FG-NET (in a leave-one-person-out fashion), MORPH, and BROWNS. Substantial performance improvements have been observed on all three databases.

The rest of this Chapter is organized as follows: Sect. 10.2 introduces the preprocessing step of converting 2D images to 3D models, Sect. 10.3 describes the aging model, Sect. 10.4 presents the aging simulation methods using the aging model, and Sect. 10.5 provides experimental results and discussions. Section 10.6 summarizes the conclusions and lists some directions for future work.

## 10.2 Preprocessing

Park et al. propose to use a set of 3D face images to learn the model for recognition, because the true craniofacial aging model [18] can be appropriately formulated only in 3D. However, since only 2D aging databases are available, it is necessary to first convert these 2D face images into 3D. Major notations that are used in the following sections are defined first.

- $\mathbf{S}_{mm} = \{S_{mm,1}, S_{mm,2}, \dots, S_{mm,n_{mm}}\}$ : a set of 3D face models used in constructing the reduced morphable model.  $n_{mm}$  is the number of 3D face models.
- $\mathbf{S}_\alpha$ : reduced morphable model represented with the model parameter  $\alpha$ .
- $\mathbf{S}_{2d,i}^j = \{x_1, y_1, \dots, x_{n_{2d}}, y_{n_{2d}}\}$ : 2D facial feature points for the  $i$ th subject at age  $j$ .  $n_{2d}$  is the number of points in 2D.

- $\mathbf{S}_i^j = \{x_1, y_1, z_1, \dots, x_{n_{3d}}, y_{n_{3d}}, z_{n_{3d}}\}$ : 3D feature points for the  $i$ th subject at age  $j$ .  $n_{3d}$  is the number of points in 3D.
- $\mathbf{T}_i^j$ : facial texture for the  $i$ th subject at age  $j$ .
- $\mathbf{s}_i^j$ : reduced shape of  $\mathbf{S}_i^j$  after applying PCA on  $\mathbf{S}_i^j$ .
- $\mathbf{t}_i^j$ : reduced texture of  $\mathbf{T}_i^j$  after applying PCA on  $\mathbf{T}_i^j$ .
- $\mathbf{V}_s$ : top  $L_s$  principle components of  $\mathbf{S}_i^j$ .
- $\mathbf{V}_t$ : top  $L_t$  principle components of  $\mathbf{T}_i^j$ .
- $\mathbf{S}_{w_s}^j$ : synthesized 3D facial feature points at age  $j$  represented with weight  $w_s$ .
- $\mathbf{T}_{w_t}^j$ : synthesized texture at age  $j$  represented with weight  $w_t$ .
- $n_{mm} = 100$ ,  $n_{2d} = 68$ ,  $n_{3d} = 81$ ,  $L_s = 20$ , and  $L_t = 180$ .

In the following subsections,  $\mathbf{S}_{2d,i}^j$  is first transformed to  $\mathbf{S}_i^j$  using the reduced morphable model  $\mathbf{S}_\alpha$ . Then, 3D shape aging pattern space  $\{\mathbf{S}_{w_s}\}$  and texture aging pattern space  $\{\mathbf{T}_{w_t}\}$  are constructed using  $\mathbf{S}_i^j$  and  $\mathbf{T}_i^j$ .

### **10.2.1 2D Facial Feature Point Detection**

Manually marked feature points are used in aging model construction. However, in the test stage the feature points need to be detected automatically. The feature points on 2D face images are detected using the conventional Active Appearance Model (AAM) [3, 23]. AAM models for the three databases are trained separately, the details of which are given below.

#### **10.2.1.1 FG-NET**

Face images in the FG-NET database have already been (manually) marked by the database provider with 68 feature points. These feature points are used to build the aging model. Feature points are also automatically detected and the face recognition performance based on manual and automatic feature point detection methods are compared. The training and feature point detection are conducted in a cross-validation fashion.

#### **10.2.1.2 MORPH**

Unlike the FG-NET database, a majority of face images in the MORPH database belong to African-Americans. These images are not well represented by the AAM model trained on the FG-NET database due to the differences in the cranial structure between the Caucasian and African-American populations. Therefore, a subset of images (80) in the MORPH database are labeled as a training set for the automatic feature point detector in the MORPH database.

### 10.2.1.3 BROWNS

The entire FG-NET database is used to train the AAM model for detecting feature points on images in the BROWNS database.

### 10.2.2 3D Model Fitting

A simplified deformable model based on Blanz and Vetter's model [2] is used as a generic 3D face model. For efficiency, the number of vertices in the 3D morphable model is drastically reduced to 81, 68 of which correspond to salient features present in the FG-NET database, while the other 13 delineate the forehead region. Following [2], PCA was performed on the simplified shape sample set,  $\{S_{mm}\}$ . The mean shape  $\bar{S}_{mm}$ , the eigenvalues  $\lambda_l$ 's, and unit eigenvectors  $\mathbf{W}_l$ 's of the shape covariance matrix are obtained. Only the top  $L$  ( $= 30$ ) eigenvectors are used, again for efficiency and stability of the subsequent fitting algorithm performed on the possibly very noisy dataset. A 3D face shape can then be represented using the eigenvectors as

$$\mathbf{S}_\alpha = \bar{\mathbf{S}}_{mm} + \sum_{l=1}^L \alpha_l \mathbf{W}_l, \quad (10.1)$$

where the parameter  $\boldsymbol{\alpha} = [\alpha_l]$  controls the shape, and the covariance of  $\boldsymbol{\alpha}$ 's is the diagonal matrix with  $\lambda_i$  as the diagonal elements. A description is given below on how to transform the given 2D feature points  $\mathbf{S}_{2d,i}^j$  to the corresponding 3D points  $\mathbf{S}_i^j$  using the reduced morphable model  $\mathbf{S}_\alpha$ .

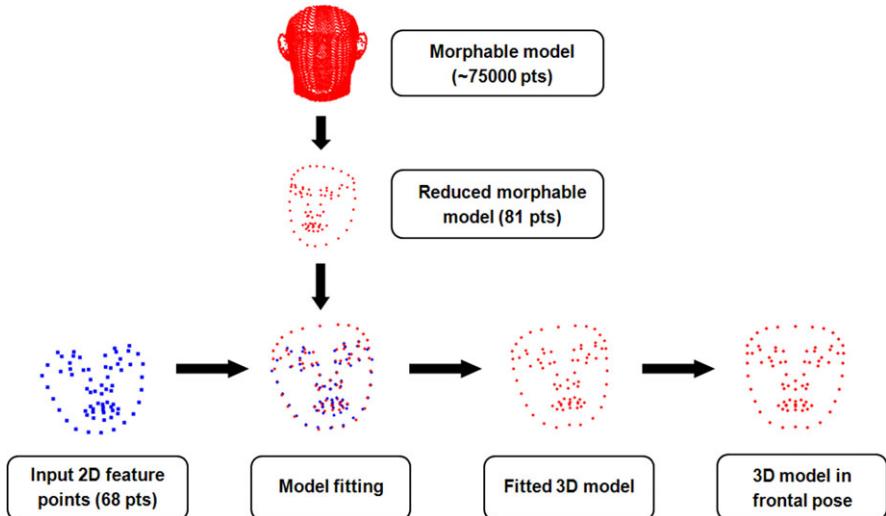
Let  $E(\cdot)$  be the overall error in fitting the 3D model of one face to its corresponding 2D feature points, where

$$E(\mathbf{P}, \mathbf{R}, \mathbf{t}, \boldsymbol{\alpha}, \{\alpha_l\}_{l=1}^L) = \|\mathbf{S}_{i,2d}^j - \mathbf{T}_{\mathbf{P}, \mathbf{R}, \mathbf{t}, \boldsymbol{\alpha}}(\mathbf{S}_\alpha)\|^2. \quad (10.2)$$

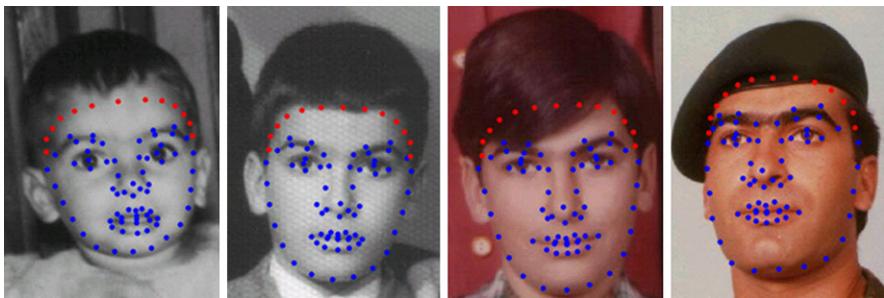
Here,  $\mathbf{T}(\cdot)$  represents a transformation operator performing a sequence of operations, that is, rotation, translation, scaling, projection, and selecting  $n_{2d}$  points out of  $n_{3d}$  that have correspondences. To simplify the procedure, an orthogonal projection  $\mathbf{P}$  is used.

In practice, the 2D feature points that are either manually labeled or automatically generated by AAM are noisy, which means overfitting these feature points may produce undesirable 3D shapes. This issue is addressed by introducing a Tikhonov regularization term to control the Mahalanobis distance of the shape from the mean shape. Let  $\sigma$  be the empirically estimated standard deviation of the energy  $E$  induced by the noise in the location of the 2D feature points. The regularized energy is defined as

$$E' = E/\sigma^2 + \sum_{l=1}^L \alpha_l^2 / \lambda_l. \quad (10.3)$$



**Fig. 10.1** 3D model fitting process using the reduced morphable model [13]



**Fig. 10.2** Four example images with manually labeled 68 points (blue) and the automatically recovered 13 points (red) for the forehead region [13]

To minimize the energy term defined in (10.3), all the  $\alpha_l$ 's are initialized to 0, the rotation matrix  $\mathbf{R}$  is set to the identity matrix and translation vector  $\mathbf{t}$  is set to 0, and the scaling factor  $a$  is set to match the overall size of the 2D and 3D shapes. Then,  $\mathbf{R}$ ,  $\mathbf{T}$ , and  $\alpha$  are iteratively updated until convergence. There are multiple ways to find the optimal pose given the current  $\alpha$ . In these tests, it was found that first estimating the best  $2 \times 3$  affine transformation followed by a QR decomposition to get the rotation works better than running a quaternion based optimization using Rodriguez's formula [17]. Note that  $\mathbf{t}_z$  is fixed to 0, as an orthogonal projection is used.

Figure 10.1 illustrates the 3D model fitting process to acquire the 3D shape. The associated texture is then retrieved by warping the 2D image. Figure 10.2 shows the manually labeled 68 points and automatically recovered 13 points that delineate the forehead region.

## 10.3 Aging Pattern Modeling

Following [7], the aging pattern is defined as an array of face models from a single subject indexed by the age. This model construction differs from [7] mainly in that the shape and texture are separately modeled at different ages using the shape (aging) pattern space and the texture (aging) pattern space, respectively, because the 3D shape and the texture images are less correlated than the 2D shape and texture that they used in [7]. The two pattern spaces as well as the adjustment of the 3D shape are described below.

### 10.3.1 Shape Aging Pattern

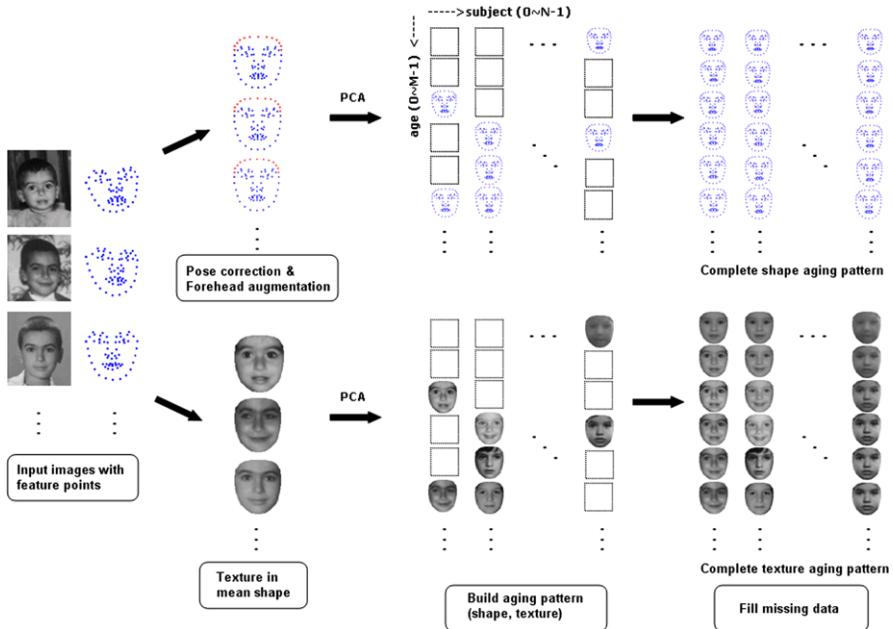
Shape pattern space captures the variations in the internal shape changes and the size of the face. The pose corrected 3D models obtained from the pre-processing phase are used for constructing the shape pattern space. Under age 19, the key effects of aging are driven by the increase of the cranial size, while at later ages the facial growth in height and width is very small [1]. To incorporate the growth pattern of the cranium for ages under 19, the overall size of 3D shape is rescaled according to the average anthropometric head width found in [5].

PCA is applied over all the 3D shapes,  $\mathbf{S}_i^j$  in the database irrespective of age  $j$  and subject  $i$ . All the mean subtracted  $\mathbf{S}_i^j$  are projected on to the subspace spanned by the columns of  $\mathbf{V}_s$  to obtain  $\mathbf{s}_i^j$  as

$$\mathbf{s}_i^j = \mathbf{V}_s^T (\mathbf{S}_i^j - \bar{\mathbf{S}}), \quad (10.4)$$

which is an  $L_s \times 1$  vector.

Assuming that there are  $n$  subjects at  $m$  ages, the basis of the shape pattern space is then assembled as an  $m \times n$  matrix with vector entries (or alternatively as an  $m \times n \times L_s$  tensor), where the  $j$ th row corresponds to age  $j$  and the  $i$ th column corresponds to subject  $i$ , and the entry at  $(j, i)$  is  $\mathbf{s}_i^j$ . The shape pattern basis is initialized with the projected shapes  $\mathbf{s}_i^j$  from the face database (as shown in the third column of Fig. 10.3). Then, missing values are filled using the available values along the same column (i.e., for the same subject). Three different methods are tested for the filling process: linear, Radial Basis Function (RBF), and a variant of RBF ( $v$ -RBF). Given available ages  $a_i$  and the corresponding shape feature vectors  $s_i$ , a missing feature value  $s_x$  at age  $a_x$  can be estimated by  $s_x = l_1 \times s_1 + l_2 \times s_2$  in linear interpolation, where  $s_1$  and  $s_2$  are shape feature vectors corresponding to the ages  $a_1$  and  $a_2$  that are closest from  $a_x$ , and  $l_1$  and  $l_2$  are weights inversely proportional to the distance from  $a_x$  to  $a_1$  and  $a_2$ . In the  $v$ -RBF process, each feature is replaced by a weighted sum of all available features as  $s_x = \sum_i \phi(a_x - a_i) s_i / (\sum \phi(a_x - a_i))$ , where  $\phi(\cdot)$  is a RBF function defined by a Gaussian function. In the RBF method, the mapping function from age to the shape feature vector is calculated by  $s_x =$



**Fig. 10.3** 3D aging model construction [13]

$\sum_i r_i \phi(a_x - a_i) / (\sum \phi(a_x - a_i))$  for each available age and feature vector  $a_i$  and  $s_i$ , where  $r_i$ 's are estimated based on the known scattered data. Any missing feature vector  $s_x$  at age  $x$  can thus be obtained.

The shape aging pattern space is defined as the space containing all the linear combinations of the patterns of the following type (expressed in PCA basis):

$$\mathbf{s}_{w_s}^j = \bar{\mathbf{s}}^j + \sum_{i=1}^n (\mathbf{s}_i^j - \bar{\mathbf{s}}^j) w_{s,i}, \quad 0 \leq j \leq m-1. \quad (10.5)$$

Note that the weight  $w_s$  in the linear combination above is not unique for the same aging pattern. The regularization term can be used in the aging simulation described below to resolve this issue. Given a complete shape pattern space, mean shape  $\bar{\mathbf{S}}$  and the transformation matrix  $\mathbf{V}_s$ , the shape aging model with weight  $w_s$  is defined as

$$\mathbf{S}_{w_s}^j = \bar{\mathbf{S}} + \mathbf{V}_s \mathbf{s}_{w_s}^j, \quad 0 \leq j \leq m-1. \quad (10.6)$$

### 10.3.2 Texture Aging Pattern

The texture pattern  $T_i^j$  for subject  $i$  at age  $j$  is obtained by mapping the original face image to frontal projection of the mean shape  $\bar{\mathbf{S}}$  followed by a column-wise

concatenation of the image pixels. After applying PCA on  $T_i^j$ , the transformation matrix  $V_t$  and the projected texture  $t_i^j$  are calculated. The same filling procedure is used as in the shape pattern space to construct the complete basis for the texture pattern space using  $t_i^j$ . A new texture  $\mathbf{T}_{w_t}^j$  can be similarly obtained, given an age  $j$  and a set of weights  $w_t$  as

$$\mathbf{t}_{w_t}^j = \bar{\mathbf{t}}^j + \sum_{i=1}^n (\mathbf{t}_i^j - \bar{\mathbf{t}}^j) w_{t,i}, \quad 0 \leq j \leq m-1, \quad (10.7)$$

$$\mathbf{T}_{w_t}^j = \bar{\mathbf{T}} + \mathbf{V}_t \mathbf{t}_{w_t}^j, \quad 0 \leq j \leq m-1. \quad (10.8)$$

Figure 10.3 illustrates the aging model construction process for both shape and texture pattern spaces.

### 10.3.3 Separate and Combined Shape & Texture Modeling

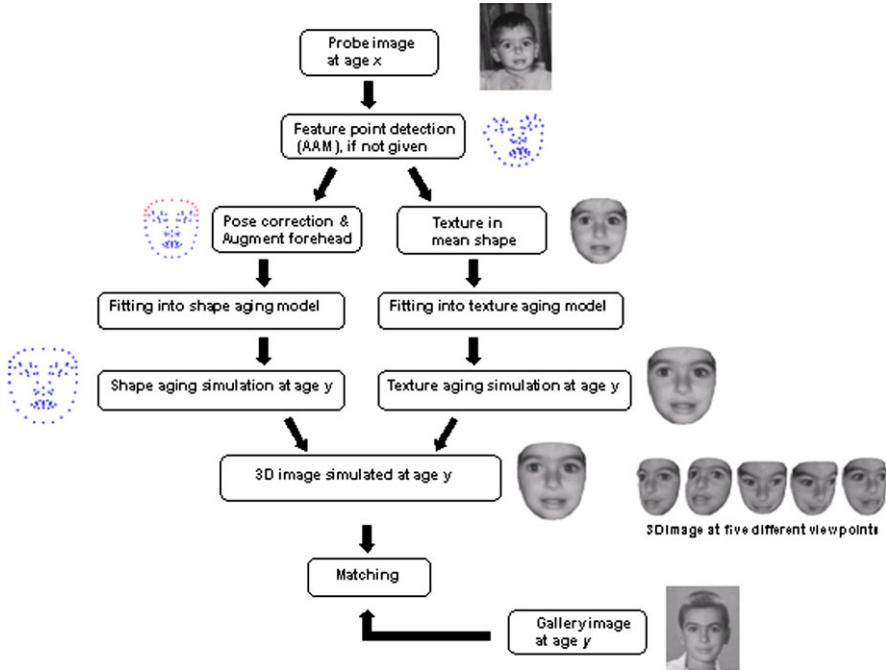
Given  $s_i^j$  and  $t_i^j$ , they can be used directly for the aging modeling or another step of PCA on the new concatenated feature vector  $C_i^j = [s_i^{j^T} \ t_i^{j^T}]^T$  can be applied. Applying PCA on  $C_i^j$  will generate a set of new Eigen vectors,  $c_i^j$  [3]. The modeling using  $s_i^j$  and  $t_i^j$  is called as “separate shape and texture modeling” and  $c_i^j$  as a “combined shape and texture modeling.”

## 10.4 Aging Simulation

Given a face image of a subject at a certain age, aging simulation involves the construction of the face image of that subject adjusted to a different age. Given a 2D image at age  $x$ , the 3D shape,  $\mathbf{S}_{\text{new}}^x$  and the texture  $\mathbf{T}_{\text{new}}^x$  are first produced by following the preprocessing step described in Sect. 10.2, and then they are projected to the reduced spaces to get  $\mathbf{s}_{\text{new}}^x$  and  $\mathbf{t}_{\text{new}}^x$ . Given a reduced 3D shape  $\mathbf{s}_{\text{new}}^x$  at age  $x$ , a weighting vector,  $w_s$ , that generates the closest possible weighted sum of the shapes at age  $x$ , can be obtained as:

$$\hat{w}_s = \underset{c_- \leq w_s \leq c_+}{\operatorname{argmin}} \| \mathbf{s}_{\text{new}}^x - \mathbf{s}_{w_s}^x \|^2 + r_s \| w_s \|^2, \quad (10.9)$$

where  $r_s$  is the weight of a regularizer to handle the cases when multiple solutions are obtained or when the linear system used to obtain the solution has a large condition number. Each element of weight vector,  $w_{s,i}$  is constrained within  $[c_-, c_+]$  to avoid strong domination by a few shape basis vectors.



**Fig. 10.4** Aging simulation from age  $x$  to  $y$  [13]

Given  $\hat{w}_s$ , age adjusted shape can be obtained at age  $y$  by carrying  $\hat{w}_s$  over to the shapes at age  $y$  and transforming the shape descriptor back to the original shape space as

$$\mathbf{S}_{\text{new}}^y = \mathbf{S}_{\hat{w}_s}^y = \bar{\mathbf{S}} + \mathbf{V}_s \mathbf{s}_{\hat{w}_s}^y. \quad (10.10)$$

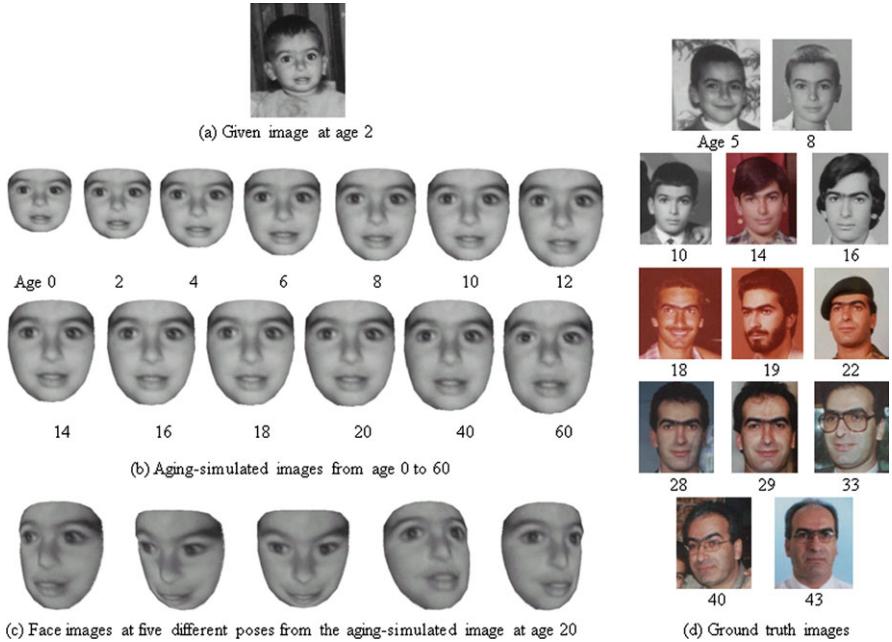
The texture simulation process is similarly performed by first estimating  $\hat{w}_t$  as

$$\hat{w}_t = \underset{c_- \leq w_t \leq c_+}{\operatorname{argmin}} \| \mathbf{t}_{\text{new}}^x - \mathbf{t}_{w_t}^x \|^2 + r_t \| w_t \|^2, \quad (10.11)$$

and then, propagating the  $\hat{w}_t$  to the target age  $y$  followed by the back projection to get

$$\mathbf{T}_{\text{new}}^y = \mathbf{T}_{\hat{w}_t}^y = \bar{\mathbf{T}} + \mathbf{V}_t \mathbf{t}_{\hat{w}_t}^y. \quad (10.12)$$

The aging simulation process is illustrated in Fig. 10.4. Figure 10.5 shows an example of aging simulated face images from a subject at age 2 in the FG-NET database. Figure 10.6 exhibits the example input images, feature point detection, pose-corrected and age-simulated images from a subject in the MORPH database. The pseudocodes of shape aging pattern space construction and simulation are given in Algorithms 10.1, 10.2, 10.3, and 10.4.

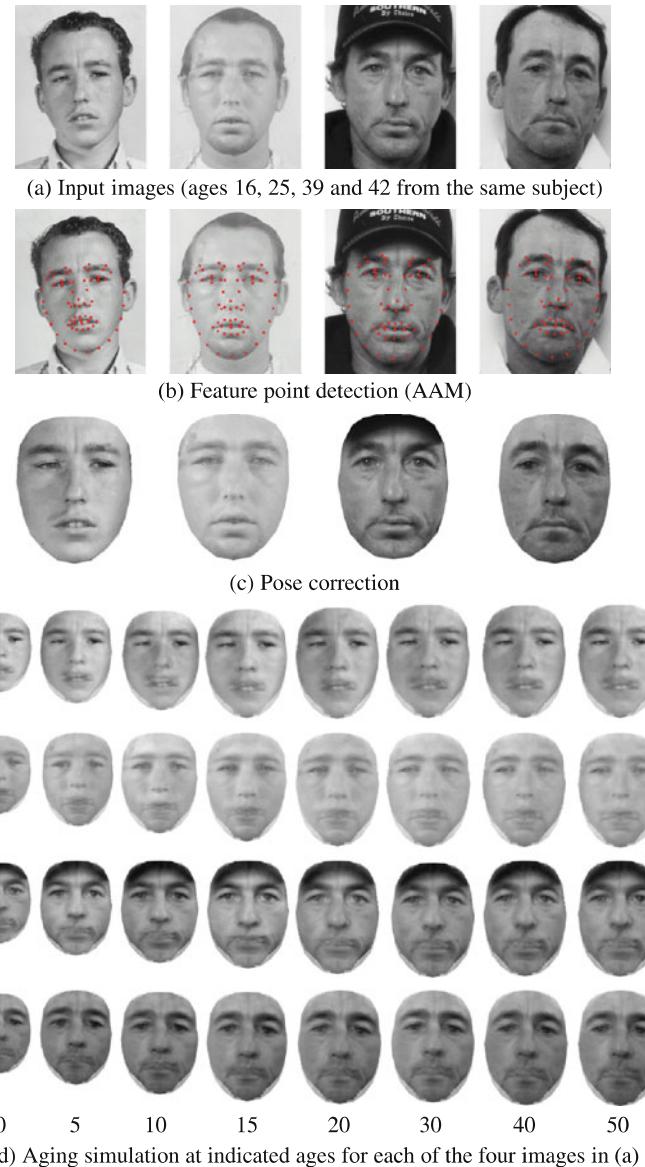


**Fig. 10.5** An example aging simulation in the FG-NET database [13]

## 10.5 Experimental Results

### 10.5.1 Database

There are two well known public domain databases to evaluate facial aging models: FG-NET [6] and MORPH [21]. The FG-NET database contains 1002 face images of 82 subjects ( $\sim 12$  images/subject) at different ages, with the minimum age being 0 (<12 months) and the maximum age being 69. There are two separate databases in MORPH: Album1 and Album2. MORPH-Album1 contains 1690 images from 625 different subjects ( $\sim 2.7$  images/subject). MORPH-Album2 contains 15 204 images from 4039 different subjects ( $\sim 3.8$  images/subject). Another source of facial aging data can be found in the book by Nixon and Galassi [11]. This is a collection of pictures of four sisters taken every year over a period of 33 years from 1975 to 2007. A new database, called, “BROWNS” are constructed by scanning 132 pictures of the four subjects (33 per subject) from the book to evaluate the aging model. Since it is desirable to have as many subjects and as many images at different ages per subject as possible, the FG-NET database is more useful for aging modeling than MORPH or BROWNS. The age separation observed in MORPH-Album1 is in the range 0–30 and that in MORPH-Album2 is less than 5. Therefore, MORPH-Album1 is more useful in evaluating the aging model than MORPH-Album2. A subset of MORPH-Album1, 1655 images of all the 612 subjects whose images at different ages are available, is used for the experiments. The complete FG-NET database has been



**Fig. 10.6** Example aging simulation process in the MORPH database [13]

used for model construction and then it is evaluated on FG-NET (in a leave-one-person-out fashion), MORPH-Album1 and BROWNS. Figure 10.7 shows multiple sample images of one subject from each of the three databases. The number of subjects, number of images, and number of images at different ages per subject for the three databases used in the aging study [13] are summarized in Table 10.2.

---

**Algorithm 10.1: 3D SHAPE AGING PATTERN CONSTRUCTION()**

---

*Input:*  $S_{2d} = \{S_{1,2d}^0, \dots, S_{i,2d}^j, \dots, S_{n,2d}^{m-1}\}$   
*Output:*  $s_i^j, i = 1, \dots, n, j = 0, \dots, m - 1$   
 $i \leftarrow 1, j \leftarrow 0$   
**while**  $i \leq n \& j \leq m - 1$   
  **if**  $S_{i,2d}^j$  is available  
     $k \leftarrow 1, E \leftarrow$  fitting error between  $S_{i,2d}^j$  and  $S_\alpha$   
    **while**  $k < \tau \& E < \theta$   
      **do** { update pose  $(a, R, t)$  (3D model parameters,  $\alpha$ , fixed)  
        **do** { update 3D model parameters (pose fixed)  
           $k \leftarrow k + 1$ , update  $E$   
      } }  
     $S_i^j \leftarrow S_\alpha$   
  Calculate eigenvalue  $\lambda_s$  and eigenvector and  $\mathbf{V}_s$  from  $S_i^j - \bar{S}$   
   $i \leftarrow 1, j \leftarrow 0$   
  **while**  $i \leq n \& j \leq m - 1$   
    **if**  $S_i^j$  is available  
       $s_i^j \leftarrow \mathbf{V}^T(S_i^j - \bar{S})$   
      Fill  $(i, j)$ -th shape pattern by  $s_i^j$   
    **else** Fill  $(i, j)$ -th shape pattern space, using interpolation along the column

---



---

**Algorithm 10.2: TEXTURE AGING PATTERN CONSTRUCTION()**

---

*Input:*  $S = \{S_1^0, \dots, S_i^j, \dots, S_n^{m-1}\}, T = \{T_1^0, \dots, T_i^j, \dots, T_n^{m-1}\},$   
*Pose* =  $\{P_1^1, \dots, P_i^j, \dots, P_n^m\}$   
*Output:*  $t_i^j, i = 1, \dots, n, j = 0, \dots, m - 1$   
Construct mean shape  $\bar{S}$   
 $i \leftarrow 1, j \leftarrow 0$   
**while**  $i \leq n \& j \leq m - 1$   
  **do** { if  $T_{i,2d}^j$  is available  
    Warp texture  $T_i^j$  from  $S_i^j$  with pose  $P_i^j$  to  $\bar{S}$   
  Calculate eigenvalue  $\lambda_t$  and eigenvector  $\mathbf{V}_t$  from  $(T_i^j - \bar{T})$   
   $i \leftarrow 1, j \leftarrow 0$   
  **while**  $i \leq n \& j \leq m - 1$   
    **if**  $T_i^j$  is available  
       $t_i^j \leftarrow \mathbf{V}^T(T_i^j - \bar{T})$   
      Fill  $(i, j)$ -th texture pattern by  $t_i^j$   
    **else** Fill  $(i, j)$ -th texture pattern space, using interpolation along the column

---

**Algorithm 10.3: AGE SIMULATION FOR SHAPE()**


---

*Input:*  $s = \{s_1^0, \dots, s_n^{m-1}\}, S_{\text{new}}^x$

*Output:*  $S_{\text{new}}^y$

Estimate  $w_s$  by (10.9)

Calculate  $S_{\text{new}}^y$  by (10.10)

---

**Algorithm 10.4: AGE SIMULATION FOR TEXTURE()**


---

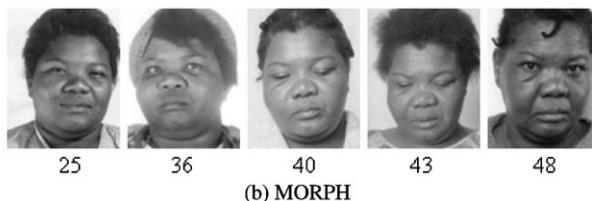
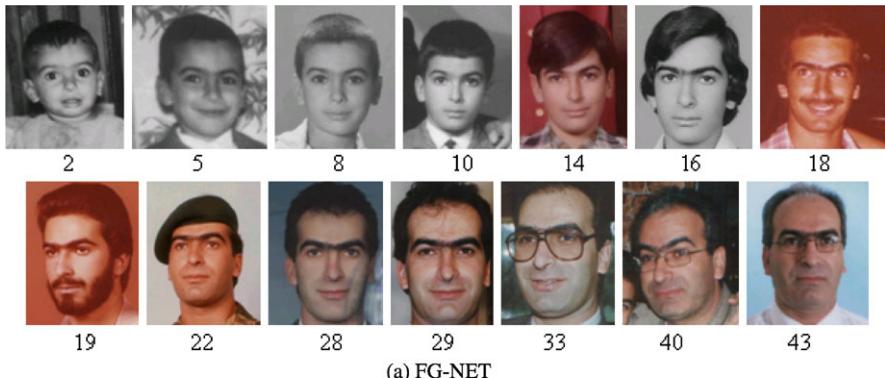
*Input:*  $t = \{t_1^0, \dots, t_n^{m-1}\}, T_{\text{new}}^x$

*Output:*  $T_{\text{new}}^y$

Estimate  $w_t$  by (10.11)

Calculate  $T_{\text{new}}^y$  by (10.12)

---



**Fig. 10.7** Example images in **a** FG-NET and **b** MORPH databases. Multiple images of one subject in each of the three databases are shown at different ages. The age value is given below each image

### 10.5.2 Face Recognition Tests

The performance of the aging model is evaluated by comparing the face recognition accuracy of a state-of-the-art matcher before and after aging simulation. The probe set,  $P = \{p_1^{x_1}, \dots, p_n^{x_n}\}$ , is constructed by selecting one image  $p_i^{x_i}$  for each subject  $i$  at age  $x_i$  in each database,  $i \in \{1, \dots, n\}$ ,  $x_i \in \{0, \dots, 69\}$ . The gallery set  $G =$

**Table 10.2** Databases used in aging modeling [13]

Database	#subjects	#images	Average #images per subject
FG-NET	82	1002	12
MORPH	Album1	625	1690
	Album2	4039	15204
BROWNS	4	132	33

**Table 10.3** Probe and gallery data used in face recognition tests [13]

Database	Probe			Gallery		
	#images	#subjects	Age group	#images	#subjects	Age group
FG-NET	82	82	{0, 5, ..., 30}	82	82	$x^* + \{5, 10, \dots, 30\}$
MORPH	612	612	{15, 20, ..., 30}	612	612	$x + \{5, 10, \dots, 30\}$
BROWNS	4	4	{15, 20, ..., 30}	100	100	$x + \{5, 10, \dots, 30\}$

\*  $x$  is the age of the probe group

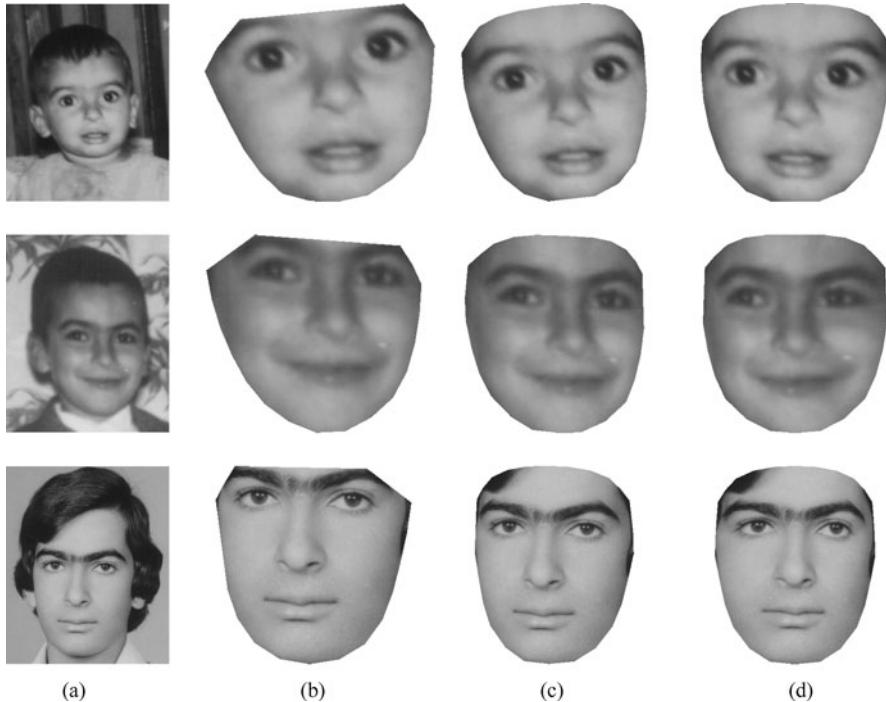
$\{g_1^{y_1}, \dots, g_n^{y_n}\}$  is similarly constructed. A number of different probe and gallery age groups are also constructed from the three databases to demonstrate the model's effectiveness in different periods of the aging process.

Aging simulation is performed in both aging and de-aging directions for each subject  $i$  in the probe and each subject  $j$  in the gallery as  $(x_i \rightarrow y_j)$  and  $(y_j \rightarrow x_i)$ . Table 10.3 summarizes the probe and gallery data sets used in the face recognition test [13].

Let  $P$ ,  $P_f$  and  $P_a$  denote the probe, the pose-corrected probe, and the age-adjusted probe set, respectively. Let  $G$ ,  $G_f$  and  $G_a$  denote the gallery, the pose-corrected gallery, and age-adjusted gallery set, respectively. All age-adjusted images are generated (in a leave-one-person-out fashion for FG-NET) using the shape and texture pattern space. The face recognition test is performed on the following probe-gallery pairs:  $P-G$ ,  $P-G_f$ ,  $P_f-G$ ,  $P_f-G_f$ ,  $P_a-G_f$  and  $P_f-G_a$ . The identification rate for the probe-gallery pair  $P-G$  is the performance on original images without applying any aging model. The accuracy obtained by fusion of  $P-G$ ,  $P-G_f$ ,  $P_f-G$  and  $P_f-G_f$  matchings is regarded as the performance after pose correction. The accuracy obtained by fusion of all the pairs  $P-G$ ,  $P-G_f$ ,  $P_f-G$ ,  $P_f-G_f$ ,  $P_a-G_f$  and  $P_f-G_a$  represents the performance after aging simulation. A simple score-sum based fusion is used in all the experiments.

### 10.5.3 Effects of Different Cropping Methods

The performance of the face recognition system is evaluated with different face cropping methods. An illustration of the cropping results obtained by different approaches is shown in Fig. 10.8. The first column shows the input face image and



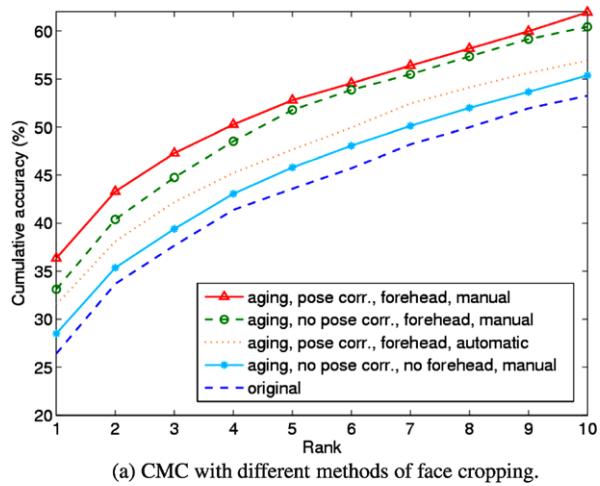
**Fig. 10.8** Example images showing different face cropping methods: **a** original, **b** no-forehead and no pose correction, **c** no pose correction with forehead, **d** pose correction with forehead [13]

the second column shows the cropped face obtained using the 68 feature points provided in the FG-NET database without pose correction. The third column shows the cropped face obtained with the additional 13 points (total 81 feature points) for forehead inclusion without any pose correction. The last column shows the cropping obtained by the 81 feature points, with pose correction.

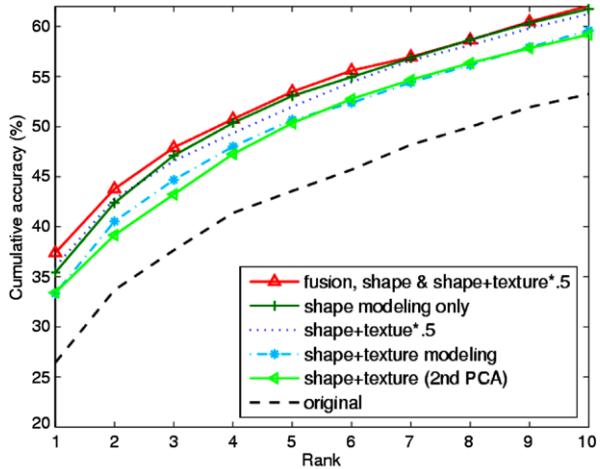
Figure 10.9(a) shows the face recognition performance on FG-NET using only shape modeling based on different face cropping methods and feature point detection methods. Face images with pose correction that include the forehead show the best performance. This result shows that the forehead does influence the face recognition performance, although it has been a common practice to remove the forehead in AAM based feature point detection and subsequent face modeling [3, 8, 26]. Therefore, the aging simulation is evaluated with the model that contains the forehead region with pose correction.

Note that, the performance difference between nonfrontal and frontal pose is as expected, and that the performance using automatically detected feature points is lower than that of manually labeled feature points. However, the performance with automatic feature point detection is still better than that of matching original images before applying the aging modeling.

**Fig. 10.9** Cumulative Match Characteristic (CMC) curves with different methods of face cropping and shape & texture modeling [13]



(a) CMC with different methods of face cropping.



(b) CMC with different methods of shape & texture modeling.

#### 10.5.4 Effects of Different Strategies in Employing Shape and Texture

Most of the existing face aging modeling techniques use either only shape or a combination of shape and texture [7, 8, 14, 20, 26]. Park et al. have tested the aging model with shape only, separate shape and texture, and combined shape and texture modeling. In the test of the combined scheme, shape and the texture are concatenated and a second stage of principle component analysis is applied to remove the possible correlation between shape and texture as in the AAM face modeling technique.

Figure 10.9(b) shows the face recognition performance of different approaches to shape and texture modeling. Consistent performance drop has been observed in face recognition performance when the texture is used together with the shape. The best performance is observed by combining shape modeling and shape + texture modeling using score level fusion. When simulating the texture, the aging simulated texture and the original texture have been blended with equal weights. Unlike shape, texture is a higher dimensional vector that can easily deviate from its original identity after the aging simulation. Even though performing aging simulation on texture produces more realistic face images, it can easily lose the original face-based identity information. The blending process with the original texture reduces the deviation and generates better recognition performance. In Fig. 10.9(b), shape + texture modeling represent separate modeling of shape and texture, shape + texture  $\times 0.5$  represents the same procedure but with the blending of the simulated texture with the original texture. The fusion of shape and shape + texture  $\times 0.5$  strategy is used for the following aging modeling experiments.

### **10.5.5 Effects of Different Filling Methods in Model Construction**

Park et al. tried a few different methods of filling missing values in aging pattern space construction (see Sect. 10.3.1): linear,  $v$ -RBF, and RBF. The rank-one accuracies are obtained as 36.12%, 35.19%, and 36.35% in shape + texture  $\times 0.5$  modeling method for linear,  $v$ -RBF, and RBF methods, respectively. The linear interpolation method is used in the rest of the experiments for the following reasons: (i) performance difference is minor, (ii) linear interpolation is computationally efficient, and (iii) the calculation of RBF based mapping function can be ill-posed.

Figure 10.10 provides the Cumulative Match Characteristic (CMC) curves with original, pose-corrected and aging simulated images in FG-NET, MORPH and BROWNS, respectively. It can be seen that there are significant performance improvement after aging modeling and simulation in all the three databases. The amount of improvement due to aging simulation is more or less similar with those of other studies as shown in Table 10.1. However, Park et al. used FaceVACS, a state-of-the-art face matcher, which is known to be more robust against internal and external facial variations (e.g., pose, lighting, expression, etc.) than simple PCA based matchers. They argued that the performance gain using FaceVACS is more realistic than that of a PCA matcher reported in other studies. Further, unlike other studies, they have used the entire FG-NET and MORPH-Album1 in the experiments. Another unique attribute of their studies is that the model is built on FG-NET and then evaluated on independent databases MORPH and BROWNS.

Figure 10.11 presents the rank-one identification accuracy for each of the 42 different age pair groups of probe and gallery in the FG-NET database. The aging process can be separated as growth and development ( $age \leq 18$ ) and adult aging process ( $age > 18$ ). The face recognition performance is somewhat lower in the growth process where more changes occur in the facial appearance. However, the

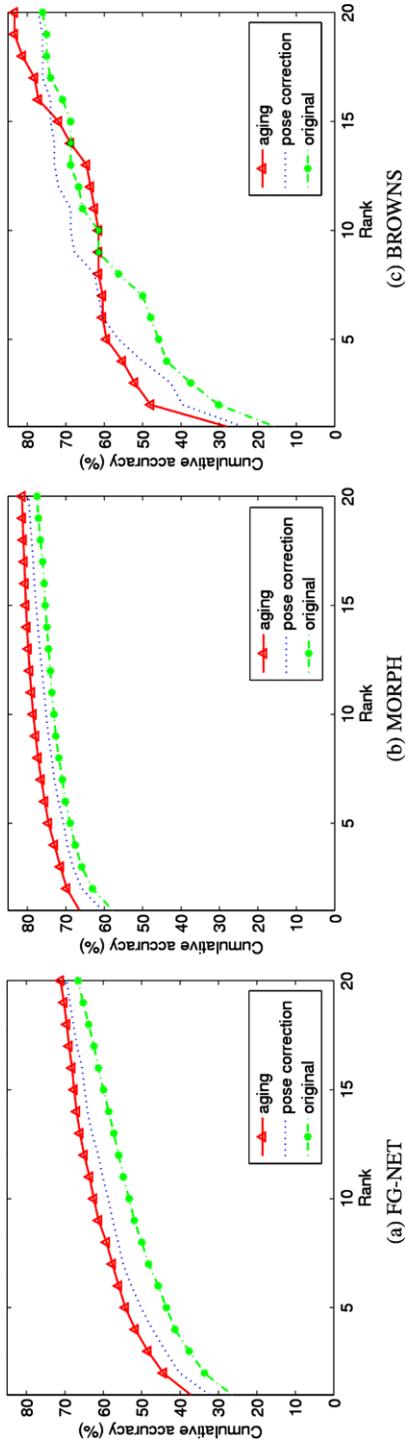
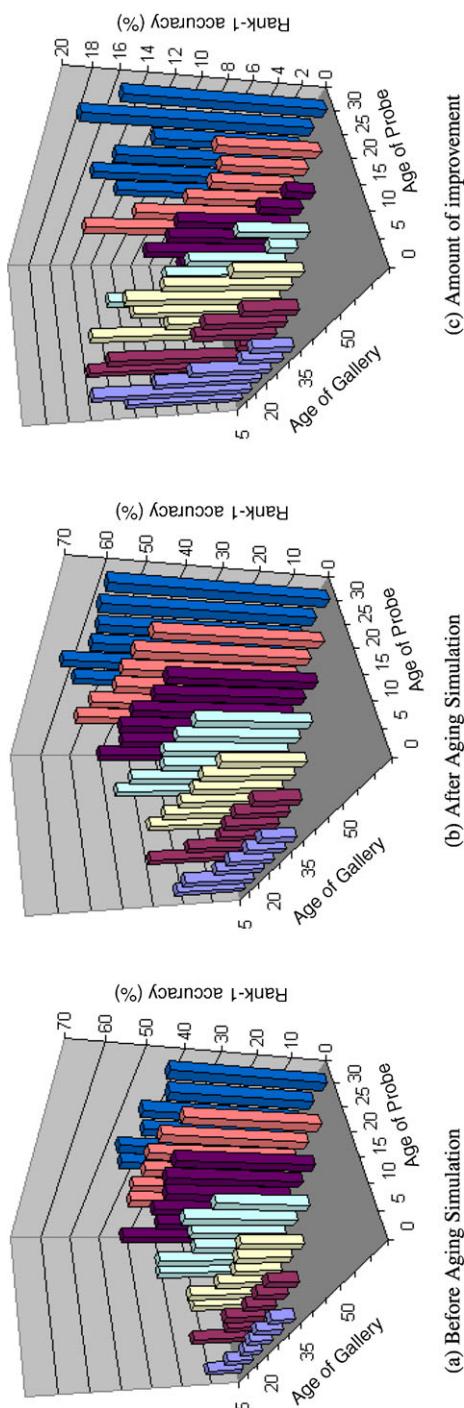
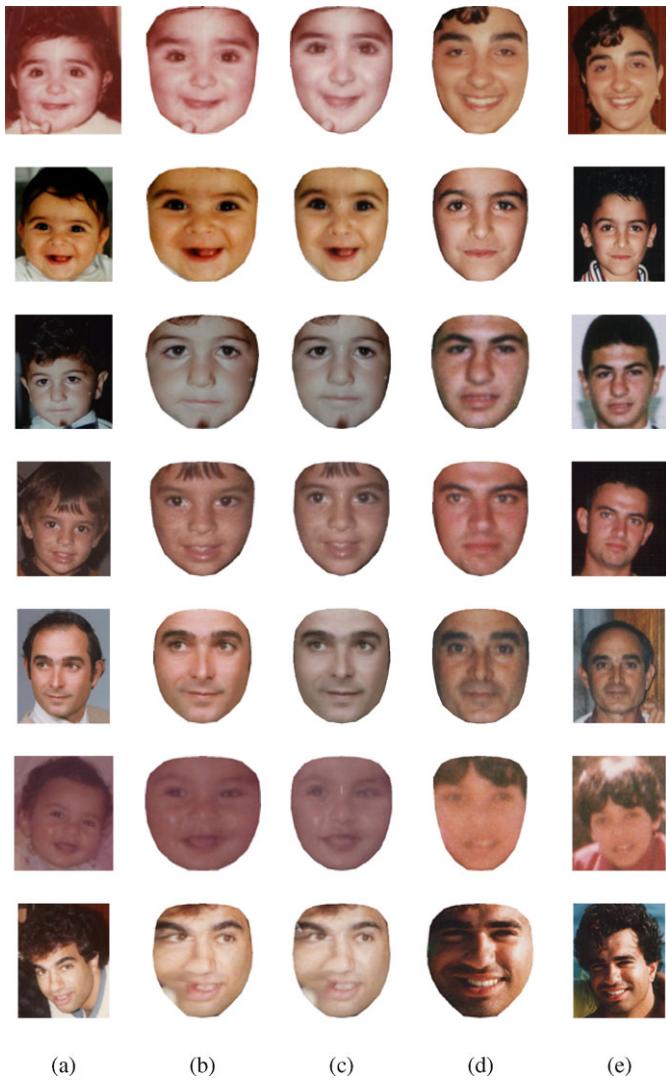


Fig. 10.10 Cumulative Match Characteristic (CMC) curves [13]



**Fig. 10.11** Rank-one identification accuracy for each probe and gallery age group: **a** before aging simulation, **b** after aging simulation, and **c** the amount of improvements after aging simulation [13]



**Fig. 10.12** Example matching results before and after aging simulation for seven different subjects: **a** probe, **b** pose-corrected probe, **c** age-adjusted probe, **d** pose-corrected gallery, and **e** gallery. The ages in each (probe, gallery) pair are (0, 18), (0, 9), (4, 14), (3, 20), (30, 49), (0, 7) and (23, 31), respectively, from the top to the bottom row [13]

aging process provides performance improvements in both of the age groups,  $\leq 18$  and  $> 18$ . The average recognition results for age groups  $\leq 18$  are improved from 17.3% to 24.8% and those for age groups  $> 18$  are improved from 38.5% to 54.2%.

Matching results for seven subjects in FG-NET are demonstrated in Fig. 10.12. The face recognition fails without aging simulation for all these subjects but succeeds with aging simulations for the first five of the seven subjects. The aging sim-

ulation fails to provide correct matchings for the last two subjects, possibly due to poor texture quality (for the sixth subject) or large pose and illumination variation (for the seventh subject).

The aging model construction takes about 44 s. The aging model is constructed off-line, therefore its computation time is not a major concern. In the recognition stage, the entire process, including feature points detection, aging simulation, enrollment and matching takes about 12 s per probe image. Note that the gallery images are preprocessed off-line. All computation times are measured on a Pentium 4, 3.2 GHz, 3 G-Byte RAM machine.

## 10.6 Conclusions

A 3D facial aging model and simulation method for age-invariant face recognition has been described. It is shown that extension of shape modeling from 2D to 3D domain gives additional capability of compensating for pose and, potentially, lighting variations. Moreover, the use of 3D model appears to provide a more powerful modeling capability than 2D age modeling because the changes in human face configuration occur primarily in 3D domain. The aging model has been evaluated using a state-of-the-art commercial face recognition engine (FaceVACS). Face recognition performances have been improved on three different publicly available aging databases. It is shown that the method is capable of handling both growth and developmental adult face aging effects.

Exploring different (nonlinear) methods for building aging pattern space, given noisy 2D or 3D shape and texture data, with cross validation of the aging pattern space and aging simulation results in terms of face recognition performance can further improve simulated aging. Age estimation is crucial if a fully automatic age invariant face recognition system is needed.

## References

1. Albert, A., Ricanek, K., Patterson, E.: The aging adult skull and face: A review of the literature and report on factors and processes of change. UNCW Technical Report, WRG FSC-A (2004)
2. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: Proc. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH), pp. 187–194 (1999)
3. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 681–685 (2001)
4. FaceVACS Software Developer Kit, Cognitec Systems GmbH. <http://www.cognitec-systems.de>
5. Farkas, L.G. (ed.): Anthropometry of the Head and Face. Lippincott Williams & Wilkins, Baltimore (1994)
6. FG-NET Aging Database. <http://www.fgnet.rsunit.com>
7. Geng, X., Zhou, Z.-H., Smith-Miles, K.: Automatic age estimation based on facial aging patterns. IEEE Trans. Pattern Anal. Mach. Intell. **29**, 2234–2240 (2007)
8. Lanitis, A., Taylor, C.J., Cootes, T.F.: Toward automatic simulation of aging effects on face images. IEEE Trans. Pattern Anal. Mach. Intell. **24**(4), 442–455 (2002)

9. Lanitis, A., Draganova, C., Christodoulou, C.: Comparing different classifiers for automatic age estimation. *IEEE Trans. Syst. Man Cybern., Part B, Cybern.* **34**(1), 621–628 (2004)
10. Ling, H., Soatto, S., Ramanathan, N., Jacobs, D.: A study of face recognition as people age. In: Proc. IEEE Int. Conf. on Computer Vision (ICCV), pp. 1–8 (2007)
11. Nixon, N., Galassi, P.: The Brown Sisters, Thirty-Three Years. The Museum of Modern Art, NY, USA (2007)
12. O'Toole, A., Vetter, T., Volz, H., Salter, E.: Three-dimensional caricatures of human heads: distinctiveness and the perception of facial age. *Perception* **26**, 719–732 (1997)
13. Park, U., Tong, Y., Jain, A.K.: Age invariant face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(5), 947–954 (2010)
14. Patterson, E., Ricanek, K., Albert, M., Boone, E.: Automatic representation of adult aging in facial images. In: Proc. Int. Conf. on Visualization, Imaging, and Image Processing, IASTED, pp. 171–176 (2006)
15. Patterson, E., Sethuram, A., Albert, M., Ricanek, K., King, M.: Aspects of age variation in facial morphology affecting biometrics. In: Proc. IEEE Conf. on Biometrics: Theory, Applications, and Systems (BTAS), pp. 1–6 (2007)
16. Phillips, P.J., Scruggs, W.T., O'Toole, A.J., Flynn, P.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: FRVT 2006 and ICE 2006 large-scale results. Technical Report NISTIR 7408, National Institute of Standards and Technology
17. Pighin, F., Szeliski, R., Salesin, D.H.: Modeling and animating realistic faces from images. *Int. J. Comput. Vis.* **50**(2), 143–169 (2002)
18. Pittenger, J.B., Shaw, R.E.: Aging faces as visco-elastic events: Implications for a theory of nonrigid shape perception. *J. Exp. Psychol. Hum. Percept. Perform.* **1**, 374–382 (1975)
19. Ramanathan, N., Chellappa, R.: Face verification across age progression. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 462–469 (2005)
20. Ramanathan, N., Chellappa, R.: Modeling age progression in young faces. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 387–394 (2006)
21. Ricanek, K.J., Tesafaye, T.: Morph: A longitudinal image database of normal adult age-progression. In: Proc. Int. Conf. on Automatic Face and Gesture Recognition (FG), pp. 341–345 (2006)
22. Scherbaum, K., Sunkel, M., Seidel, H.-P., Blanz, V.: Prediction of individual non-linear aging trajectories of faces. *Comput. Graph. Forum* **26**(3), 285–294 (2007)
23. Stegmann, M.B.: The AAM-API: An open source active appearance model implementation. In: Proc. Int. Conf. Medical Image Computing and Computer-Assisted Intervention (MICCAI), pp. 951–952 (2003)
24. Suo, J., Min, F., Zhu, S., Shan, S., Chen, X.: A multi-resolution dynamic model for face aging simulation. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1–8 (2007)
25. Thompson, D.W.: On growth and form (1992)
26. Wang, J., Shang, Y., Su, G., Lin, X.: Age simulation for face recognition. In: Proc. Int. Conf. on Pattern Recognition (ICPR), pp. 913–916 (2006)

## **Part II**

# **Face Recognition Techniques**



# Chapter 11

## Face Detection

Stan Z. Li and Jianxin Wu

### 11.1 Introduction

Face detection is the first step in automated face recognition. Its reliability has a major influence on the performance and usability of the entire face recognition system. Given a single image or a video, an ideal face detector should be able to identify and locate all the present faces regardless of their position, scale, orientation, age, and expression. Furthermore, the detection should be done irrespectively of extraneous illumination conditions and the image and video content.

Face detection can be performed based on several cues: skin color (for faces in color images and videos), motion (for faces in videos), facial/head shape, facial appearance, or a combination of these parameters. Most successful face detection algorithms are appearance-based without using other cues. The processing is done as follows: An input image is scanned at all possible locations and scales by a subwindow. Face detection is posed as classifying the pattern in the subwindow as either face or nonface. The face/nonface classifier is learned from face and nonface training examples using statistical learning methods.

This chapter presents appearance-based and learning-based methods.<sup>1</sup> It will highlight AdaBoost-based methods because so far they are the most successful ones in terms of detection accuracy and speed. Effective postprocessing methods are also described. Experimental results are provided.

---

<sup>1</sup>The reader is referred to a review article [50] for other earlier face detection methods.

S.Z. Li (✉)

Center for Biometrics and Security Research & National Laboratory of Pattern Recognition,  
Institute of Automation, Chinese Academy of Sciences, Beijing, China  
e-mail: [szli@cbsr.ia.ac.cn](mailto:szli@cbsr.ia.ac.cn)

J. Wu

School of Computer Engineering, Nanyang Technological University, Singapore, Singapore  
e-mail: [jxwu@ntu.edu.sg](mailto:jxwu@ntu.edu.sg)



**Fig. 11.1** Face (*top*) and nonface (*bottom*) examples

## 11.2 Appearance and Learning-Based Approaches

With appearance-based methods, face detection is treated as a problem of classifying each scanned subwindow as one of two classes (that is, face and nonface). Appearance-based methods avoid difficulties in modeling 3D structures of faces by considering possible face appearances under various conditions. A face/nonface classifier may be learned from a training set composed of face examples taken under possible conditions as would be seen in the running stage and nonface examples as well (see Fig. 11.1 for a random sample of 10 face and 10 nonface subwindow images). Building such a classifier is possible because pixels on a face are highly correlated, whereas those in a nonface subwindow present much less regularity.

However, large variations brought about by changes in facial appearance, lighting, and expression make the face manifold or face/nonface boundaries highly complex [4, 37, 40]. Changes in facial view (head pose) further complicate the situation. A nonlinear classifier is needed to deal with the complicated situation. The speed is also an important issue for realtime performance.

Great research effort has been made for constructing complex yet fast classifiers and much progress has been achieved since 1990s. Turk and Pentland [41] describe a detection system based on principal component analysis (PCA) subspace or eigenface representation. Whereas only likelihood in the PCA subspace is considered in the basic PCA method, Moghaddam and Pentland [23] also consider the likelihood in the orthogonal complement subspace; using that system, the likelihood in the image space (the union of the two subspaces) is modeled as the product of the two likelihood estimates, which provide a more accurate likelihood estimate for the detection. Sung and Poggio [38] first partition the image space into several face and nonface clusters and then further decompose each cluster into the PCA and null subspaces. The Bayesian estimation is then applied to obtain useful statistical features. The system of Rowley et al.'s [31] uses retinally connected neural networks. Through a sliding window, the input image is examined after going through an extensive preprocessing stage. Osuna et al. [24] train a nonlinear support vector machine to classify face and nonface patterns, and Yang et al. [51] use the SNoW (Sparse Network of Winnows) learning architecture for face detection. In these systems, a bootstrap algorithm is used iteratively to collect meaningful non-face examples from images that do not contain any faces for retraining the detector. Schneiderman and Kanade [34] use multiresolution information for different levels of wavelet transform. A nonlinear classifier is constructed using statistics of products of histograms computed from face and nonface examples. The system of five

view detectors takes about 1 minute to detect faces for a  $320 \times 240$  image over only four octaves of candidate size [34]. The speed is later improved in [35] to five seconds for an image of size  $240 \times 256$  using a Pentium II at 450 MHz.

Recent progresses in face detection mostly are made within the cascade detector framework proposed by Viola and Jones [43, 45], which provides fast and robust face detection system. Three major components contribute to the cascade face detector: an over-complete set of local features that can be evaluated quickly, an AdaBoost based method to build strong nonlinear classifiers from the weak local features, and a cascade detector architecture that leads to realtime detection speed.

An over-complete set of rectangle features, which are simple scalar Haar wavelet-like features, are shown to be effective in distinguishing faces from nonfaces. Viola and Jones make use of several techniques [7, 36] for effective computation of a large number of such features under varying scale and location, which is important for realtime performance. A single Haar-like feature, however, is far from enough to build a powerful nonlinear classifier. The AdaBoost algorithm is used to solve the following three fundamental problems: (1) selecting effective features from a large feature set; (2) constructing weak classifiers, each of which is based on one of the selected features; and (3) boosting the weak classifiers to construct a strong classifier. Moreover, the simple-to-complex cascade of classifiers makes the computation even more efficient, which follows the principles of pattern rejection [3, 8] and coarse-to-fine search [2, 10]. Their system is the first realtime frontal-view face detector, and it runs at about 15 frames per second on a  $384 \times 288$  image [45].

Various improvements have been proposed for the cascade detector, including reducing the training and testing time, and achieving higher detection accuracies. Extensions of the simple Haar-like feature set have been proposed to introduce more complex local features (for example, in [14, 21, 27]). The original cascade detector takes weeks of training time [45]. More local features lead to higher detection accuracies, but also incur even higher time and storage requirements. A strategy is introduced by Wu et al. in [47] that reduces the training time to a few hours by using a precomputation strategy. The speedup of [47] is achieved by reducing the training time of weak classifiers. An alternative strategy by Pham and Cham [27] uses one dimensional Gaussian distributions to model faces and nonfaces for a single feature, and saves more than half of the training time compared to [47].

Variants of the discrete AdaBoost algorithm used in [45] have been shown to improve the trained nonlinear classifiers, for example, using real-valued variants of AdaBoost [21]. Face detection poses an asymmetric learning problem, because we usually have only thousands of face training examples, but billions of nonfaces. It is important to specifically deal with this asymmetric learning goal. Asymmetric boosting [44] and linear asymmetric classifier (LAC) [47] are two examples that achieve higher detection performances using asymmetric learning methods.

The cascade structure has been altered for faster detection speed. Instead of evaluating all the weak classifiers in a strong classifier, the strong classifier can make a decision prematurely (evaluating only a subset of weak classifiers), for example, in the soft cascade method [5]. This “multi-exit” strategy [28, 49] usually leads to higher detection performance besides reducing testing time.

The ability to deal with nonfrontal faces is important for many real applications because approximately 75% of the faces in home photos are nonfrontal [16]. A reasonable treatment for the multiview face detection problem is the view-based method [26], in which several face models are built, each describing faces in a certain view range. This way, explicit 3D face modeling is avoided. Feraud et al. [9] adopt the view-based representation for face detection and use an array of five detectors, with each detector responsible for one facial view. Wiskott et al. [46] build elastic bunch graph templates for multiview face detection and recognition. Huang et al. [13] use SVMs to estimate the facial pose. The algorithm of Schneiderman and Kanade [34] consists of an array of five face detectors in the view-based framework.

Li et al. [17–19] present a multiview face detection system. A new boosting algorithm, called FloatBoost, is proposed to incorporate Floating Search [29] into AdaBoost (RealBoost). An extended Haar feature set is proposed for dealing with out-of-plane (left-right) rotation. A modified cascade detector (following the coarse-to-fine and simple-to-complex principle) is designed for the fast detection of multi-view faces. This work leads to the first realtime multiview face detection system. It runs at 200 ms per image ( $320 \times 240$  pixels) on a Pentium-III CPU of 700 MHz.

Huang et al. [14] presents a similar solution that detects full-range in-plane and out-of-plane rotated faces. The main contributions of [14] include a manually designed new cascade architecture, a new set of local features called granular features, and a new multi-class boosting learning algorithm called Vector Boosting.

Given that the cascade detector based on boosting learning methods has achieved the best performance to date in terms of both accuracy and speed, our presentation in the following sections focuses on this thread of research efforts. Strategies are also described for efficient detection of multiview faces.

### 11.3 AdaBoost-Based Methods

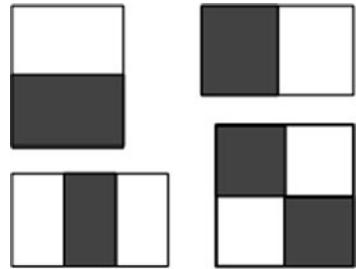
For AdaBoost learning, a complex nonlinear *strong classifier*  $H_M(x)$  is constructed as a linear combination of  $M$  simpler, easily constructible *weak classifiers* in the following form [11]

$$H_M(x) = \sum_{m=1}^M \alpha_m h_m(x) \quad (11.1)$$

where  $x$  is a pattern to be classified,  $h_m(x)$  are the  $M$  weak classifiers,  $\alpha_m \geq 0$  are the combining coefficients in  $\mathbb{R}$ . In the discrete version,  $h_m(x)$  takes a discrete value in  $\{-1, +1\}$ , whereas in the real-valued version, the output of  $h_m(x)$  is a number in  $\mathbb{R}$ .  $H_M(x)$  is real-valued, but the prediction of class label for  $x$  is obtained as  $\hat{y}(x) = \text{sign}[H_M(x)]$ .

The AdaBoost learning procedure is aimed at learning a sequence of best weak classifiers  $h_m(x)$  and the best combining weights  $\alpha_m$ . A set of  $N$  labeled training examples  $\{(x_1, y_1), \dots, (x_N, y_N)\}$  is assumed available, where  $y_i \in \{+1, -1\}$  is the class label for the example  $x_i \in \mathbb{R}^n$ . A distribution  $[w_1, \dots, w_N]$  of the training

**Fig. 11.2** Four types of rectangular Haar wavelet-like features. A feature is a scalar calculated by summing up the pixels in the white region and subtracting those in the dark region



examples, where  $w_i$  is associated with a training example  $(x_i, y_i)$ , is computed and updated during the learning to represent the distribution of the training examples. After iteration  $m$ , harder-to-classify examples  $(x_i, y_i)$  are given larger weights  $w_i^{(m)}$ , so that at iteration  $m + 1$ , more emphasis is placed on these examples. AdaBoost assumes that a procedure is available for learning a weak classifier  $h_m(x)$  from the training examples, given the distribution  $[w_i^{(m)}]$ .

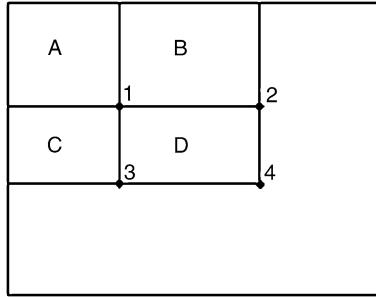
In Viola and Jones's face detection work [43, 45],<sup>2</sup> a weak classifier  $h_m(x) \in \{-1, +1\}$  is obtained by thresholding on a scalar feature  $z_k(x) \in \mathbb{R}$  selected from an overcomplete set of Haar wavelet-like features [25, 39]. In the real-valued versions of AdaBoost, such as RealBoost and LogitBoost, a real-valued weak classifier  $h_m(x) \in \mathbb{R}$  can also be constructed from  $z_k(x) \in \mathbb{R}$  [19, 21, 33]. The following discusses how to generate candidate weak classifiers.

### 11.3.1 Local Features

Viola and Jones propose four basic types of scalar features for face detection [25, 45], as shown in Fig. 11.2. Such a block feature is located in a subregion of a subwindow and varies in shape (aspect ratio), size, and location inside the subwindow. For a subwindow of size  $20 \times 20$ , there can be tens of thousands of such features for varying shapes, sizes and locations. Feature  $k$ , taking a scalar value  $z_k(x) \in \mathbb{R}$ , can be considered a transform from the  $n$ -dimensional space ( $n = 400$  if a face example  $x$  is of size  $20 \times 20$ ) to the real line. These scalar numbers form an overcomplete feature set for the intrinsically low-dimensional face pattern.

One Haar-like feature can be viewed as a mask consisting of three values: 1 for those pixels in the white region of the feature,  $-1$  for those dark region pixels in the feature, and 0 for those pixels outside of the feature region. The mask is of the same size as the subwindow. If we stack pixels of a subwindow into a vector  $x$ , and stack the mask associated with a feature into a vector  $m$ , this particular feature will have feature value  $m^T x$ .

<sup>2</sup>Viola and Jones [43, 45] used  $h_m(x) \in \{0, 1\}$ . Our notation is slightly different from but equivalent to theirs.



**Fig. 11.3** The sum of the pixels within rectangle  $D$  can be computed with four array references. The value of the integral image at location 1 is the sum of the pixels in rectangle  $A$ . The value at location 2 is  $A + B$ , at location 3 is  $A + C$ , and at location 4 is  $A + B + C + D$ . The sum within  $D$  can be computed as  $(4 + 1) - (2 + 3)$ . From Viola and Jones [43], © 2001 IEEE, with permission

These Haar-like features are interesting for two reasons: (1) powerful face/non-face classifiers can be constructed based on these features (see later); and (2) they can be computed efficiently [36] using the summed-area table [7] or integral image [43] technique.

The integral image  $\Pi(x, y)$  at location  $x, y$  contains the sum of the pixels above and to the left of  $x, y$ , defined as [43]

$$\Pi(x, y) = \sum_{x' \leq x, y' \leq y} I(x, y). \quad (11.2)$$

The image can be computed in one pass over the original image using the following pair of recurrences

$$S(x, y) = S(x, y - 1) + I(x, y), \quad (11.3)$$

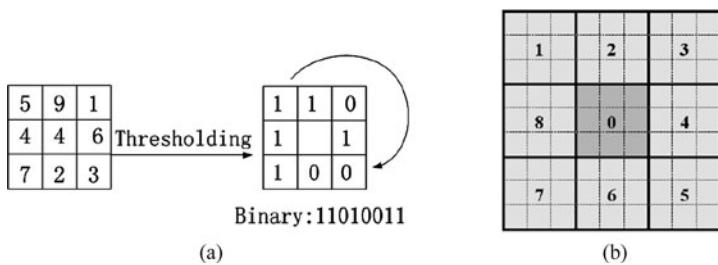
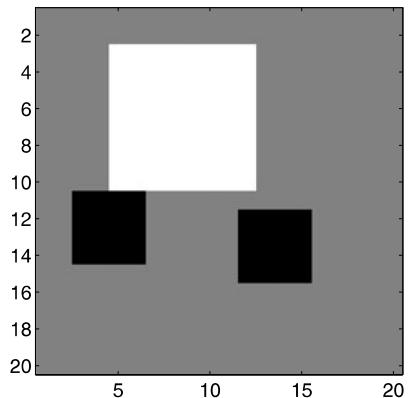
$$\Pi(x, y) = \Pi(x - 1, y) + S(x, y), \quad (11.4)$$

where  $S(x, y)$  is the cumulative row sum,  $S(x, -1) = 0$  and  $\Pi(-1, y) = 0$ . Using the integral image, any rectangular sum can be computed in four array references, as illustrated in Fig. 11.3. The use of integral images leads to enormous savings in computation for features at varying locations and scales.

With the integral images, the intensity variation within a rectangle  $D$  of any size and any location can be computed efficiently; for example  $V_D = \sqrt{V * V}$  where  $V = (4 + 1) - (2 + 3)$  is the sum within  $D$ , and a simple intensity normalization can be done by dividing all the pixel values in the subwindow by the variation.

Equation (11.4) shows that the feature value  $m^T x$  can be evaluated extremely fast when the rectangular structures in the mask  $m$  is utilized. Recently, extended sets of Haar-like features have been proposed for improving detection accuracy [27], dealing with out-of-plane head rotation [14, 19] and for in-plane head rotation [14, 21]. The extended features are carefully designed such that special structures exist in their corresponding masks, thus the feature values can be computed quickly using ideas similar to (11.4).

**Fig. 11.4** One example  
granular feature



**Fig. 11.5** Illustration of LBP (a) and MB-LBP (b)

Figure 11.4 shows an example of the granular features used in [14]. The masks corresponding to granular features have this special structure: most of the mask values are 0, while a few nonoverlapping square subregions in the mask have values of all +1 (or all -1). The square subregions are confined to be of size  $1 \times 1$ ,  $2 \times 2$ ,  $4 \times 4$ , or  $8 \times 8$ , so that the integral image trick can be used to quickly compute granular features.

The subwindow in Fig. 11.4 is of size  $20 \times 20$ , thus a feature is equivalent to a  $20 \times 20$  mask, in which the gray pixels correspond to the value 0, the  $8 \times 8$  subregion correspond to the value +1 in the mask, and the two  $4 \times 4$  subregions correspond to the value -1. By precomputing three integral images for size 2, 4, and 8 subregions correspondingly, the granular feature values can be quickly computed.

Recently other local features are also proposed for usage in face detection, among which the local binary pattern feature (LBP) has exhibited promising results.<sup>3</sup> As illustrated in Fig. 11.5(a), the original LBP operator compares a pixel with its 8 neighbors, generating a single bit '1' if the neighboring pixel has higher intensity values (and a bit '0' if otherwise). The LBP value is then a combination of these 8 bits. The multi-block LBP (MB-LBP) generalize LBP by comparing the average

<sup>3</sup>The modified Census Transform feature [15] is also used for face detection, and is very similar to LBP.

intensity of pixels within a region instead of comparing single pixel intensities (illustrated in Fig. 11.5(b)).

MB-LBP is used in [20] and [52] for face detection and recognition, which shows that MB-LBP can produce improved performance than Haar-like features in face detection. The MB-LBP features can also be computed efficiently using integral images [20].

### 11.3.2 Learning Weak Classifiers

As mentioned earlier, the AdaBoost learning procedure is aimed at learning a sequence of weak classifiers  $h_m(x)$  and the combining weights  $\alpha_m$  in (11.1). It solves the following three fundamental problems: (1) learning effective features from a large feature set; (2) constructing weak classifiers, each of which is based on one of the selected features; and (3) boosting the weak classifiers to construct a strong classifier.

AdaBoost assumes that a “weak learner” procedure is available. The task of the procedure is to select the most significant feature from a set of candidate features, given the current strong classifier learned thus far, and then construct the best weak classifier and combine it into the existing strong classifier. Here, the “significance” is with respect to some given criterion (see below).

In the case of discrete AdaBoost, the simplest type of weak classifiers is a “stump.” A stump is a single-node decision tree. When the feature is real-valued, a stump may be constructed by thresholding the value of the selected feature at a certain threshold value; when the feature is discrete-valued, it may be obtained according to the discrete label of the feature. A more general decision tree (with more than one node) composed of several stumps leads to a more sophisticated weak classifier.

For discrete AdaBoost, a stump may be constructed in the following way. Assume that we have constructed  $M - 1$  weak classifiers  $\{h_m(x)|m = 1, \dots, M - 1\}$  and we want to construct  $h_M(x)$ . The stump  $h_M(x) \in \{-1, +1\}$  is determined by comparing the selected feature  $z_{k^*}(x)$  with a threshold  $\tau_{k^*}$  as follows

$$h_M(x) = \begin{cases} +1 & \text{if } z_{k^*} > \tau_{k^*}, \\ -1 & \text{otherwise.} \end{cases} \quad (11.5)$$

In this form,  $h_M(x)$  is determined by two parameters: the type of the scalar feature  $z_{k^*}$  and the threshold  $\tau_{k^*}$ . The two may be determined by some criterion, for example, (1) the minimum weighted classification error, or (2) the lowest false alarm rate given a certain detection rate.

Supposing we want to minimize the weighted classification error with real-valued features, then we can choose a threshold  $\tau_k \in \mathbb{R}$  for each feature  $z_k$  to minimize the corresponding weighted error made by the stump with this feature; we then choose the best feature  $z_{k^*}$  among all  $k$  that achieves the lowest weighted error.

- 
0. (Input)
- (1) Training examples  $\mathcal{X} = \{(x_1, y_1), \dots, (x_N, y_N)\}$ , and example weights  $(w_1, \dots, w_N)$ , where  $N = a + b$ ; of which  $a$  examples have  $y_i = +1$  and  $b$  examples have  $y_i = -1$ .
  - (2) The mask  $m$  corresponds to a feature.
1. (Initialization)
- Compute the feature values,  $v_1, \dots, v_N$ , where  $v_i = m^T x_i$ .  
Sort the feature values as  $v_{i_1} \leq \dots \leq v_{i_N}$ , such that  $(i_1, \dots, i_N)$  is a permutation of  $(1, \dots, N)$ .  
 $\varepsilon \leftarrow \sum_{y_i=-1} w_i$
2. (Updates)
- For  $k = 1, \dots, N$ :
- ```

if  $y_{i_k} = -1$  then
     $\varepsilon \leftarrow \varepsilon - w_{i_k}, \varepsilon_i \leftarrow \varepsilon$ 
else
     $\varepsilon \leftarrow \varepsilon + w_{i_k}, \varepsilon_i \leftarrow \varepsilon$ 

```
3. (Output)
- $$k = \arg \min_{1 \leq i \leq N} \varepsilon_i.$$
- Optimal threshold  $\tau^*$ :  $\tau^* = m^T \tau_{i_k}$ .
- 

**Fig. 11.6** Finding the optimal threshold of a weak classifier

Wu et al. show in [47] that only  $N + 1$  possible  $\tau_k$  values need to be evaluated, and evaluating each  $\tau_k$  is only  $O(1)$  if we sort the feature values for  $z_k$  beforehand. This method to find the optimal threshold of a weak classifier is illustrated in Fig. 11.6.

Suppose the features are sorted in the order  $v_{i_1} \leq \dots \leq v_{i_N}$ , and  $\tau_{k_1}$  and  $\tau_{k_2}$  satisfy that  $v_{i_j} < \tau_{k_1}, \tau_{k_2} < v_{i_{j+1}}$ , then setting the threshold to either  $\tau_{k_1}$  or  $\tau_{k_2}$  will result in the same weighted error. Thus, in Fig. 11.6 only the feature values (plus  $-\infty$ ) are considered as possible thresholds. Given the weighted error at  $\tau_k = v_{i_j}$ , a simple update is sufficient to compute the error when  $\tau_k = v_{i_{j+1}}$ , because at most one example has changed its classification result.

Most of the computations in Fig. 11.6 is spent in the initialization part. One important observation in [47] is that the feature values need to be computed and sorted only once, because they do not change during the AdaBoost process even though the weights  $w$  change at each iteration. By storing the sorted feature values for all features in a table, the AdaBoost training time is reduced from weeks ([45]) to hours ([47]).

More features usually lead to higher detection accuracy [27]. However, it also means that the table of sorted feature values may be too large to be stored in the main memory. Pham and Cham construct weak classifiers using the mean and standard deviation of feature values. Given a feature, its associated mask  $m$ , and the AdaBoost weights  $w^{(M-1)}$ , the average feature value is  $\sum_i w_i^{(M-1)} m^T x_i$ , where  $x_i$  is a set of training examples.

The integral image trick can be used to accelerate the computation of the mean feature value and standard deviation, that is, providing a way to utilize the structures in the mask  $m$  and computes  $m^T x$  quickly. Let  $x$  be an image subwindow in the stacked vector form and  $y$  be the corresponding integral image, it is clear from (11.2) that the transformation that generates  $y$  from  $x$  is linear, that is, there exists a square

matrix  $B$  such that  $y = Bx$ , and  $m^T x = m^T B^{-1} y$ . The average feature value is then [27]

$$\sum_i w_i^{(M-1)} m^T x_i = m^T B^{-1} \left( \sum_i w_i^{(M-1)} y_i \right). \quad (11.6)$$

In (11.6), the weighted average integral image  $\sum_i w_i^{(M-1)} y_i$  can be computed in linear time, and the transformed mask  $m^T B^{-1}$  is sparse because of the structure in the mask  $m$ . Thus, the average feature value can be computed very quickly. Similarly, the (weighted) standard deviation can be quickly computed, too.

The faces are then modeled as a one-dimensional Gaussian distribution  $N(\mu_+, \sigma_+^2)$ , where  $\mu_+$  and  $\sigma_+$  are computed from face examples. Nonfaces are modeled by  $N(\mu_-, \sigma_-^2)$  similarly. The best threshold that separate two 1-d Gaussians can then be solved in a closed form.

This method is faster than examining all possible  $\tau_k$  values, and has a much smaller storage requirement [27]. It is reported in [27] that the training speed is about two times faster than the algorithm presented in Fig. 11.6. This method has much less storage requirements and thus can train a strong classifier with more local features.

A decision stump is simple but may not fully utilize the information contained in a feature. A more complex weak classifier can be constructed by using piece-wise decision functions [14, 22]: dividing the range of feature values into  $k$  non-overlapping cells, and learn a decision function (for example, a decision stump) for every cell. Piece-wise decision functions take longer training time but usually have higher discrimination power than simple decision stumps.

Supposing that we want to achieve the lowest false alarm rate given a certain detection rate, we can set a threshold  $\tau_k$  for each  $z_k$  so a specified detection rate (with respect to  $w^{(M-1)}$ ) is achieved by  $h_M(x)$  corresponding to a pair  $(z_k, \tau_k)$ . Given this, the false alarm rate (also with respect to  $w^{(M-1)}$ ) due to this new  $h_M(x)$  can be calculated. The best pair  $(z_{k^*}, \tau_{k^*})$  and hence  $h_M(x)$  is the one that minimizes the false alarm rate.

There is still another parameter that can be tuned to balance between the detection rate and the false alarm rate: The class label prediction  $\hat{y}(x) = \text{sign}[H_M(x)]$  is obtained by thresholding the strong classifier  $H_M(x)$  at the default threshold value 0. However, it can be done as  $\hat{y}(x) = \text{sign}[H_M(x) - T_M]$  with another value  $T_M$ , which can be tuned for the balance.

The form of (11.5) is for Discrete AdaBoost. In the case of real-valued versions of AdaBoost, such as RealBoost and LogitBoost, a weak classifier should be real-valued or output the class label with a probability value. For the real-value type, a weak classifier may be constructed as the log-likelihood ratio computed from the histograms of the feature value for the two classes. (See the literature for more details [17–19].) For the latter, it may be a decision stump or tree with probability values attached to the leaves [21].

### 11.3.3 Learning Strong Classifiers Using AdaBoost

AdaBoost learns a sequence of weak classifiers  $h_m$  and boosts them into a strong one  $H_M$  effectively by minimizing the upper bound on classification error achieved by  $H_M$ . The bound can be derived as the following exponential loss function [32]

$$J(H_M) = \sum_i e^{-y_i H_M(x_i)} = \sum_i e^{-y_i \sum_{m=1}^M \alpha_m h_m(x)} \quad (11.7)$$

where  $i$  is the index for training examples. AdaBoost constructs  $h_m(x)$  ( $m = 1, \dots, M$ ) by stagewise minimization of (11.7). Given the current  $H_{M-1}(x) = \sum_{m=1}^{M-1} \alpha_m h_m(x)$ , and the newly learned weak classifier  $h_M$ , the best combining coefficient  $\alpha_M$  for the new strong classifier  $H_M(x) = H_{M-1}(x) + \alpha_M h_M(x)$  minimizes the cost

$$\alpha_M = \arg \min_{\alpha} J(H_{M-1}(x) + \alpha_m h_M(x)). \quad (11.8)$$

The minimizer is

$$\alpha_M = \log \frac{1 - \varepsilon_M}{\varepsilon_M} \quad (11.9)$$

where  $\varepsilon_M$  is the weighted error rate

$$\varepsilon_M = \sum_i w_i^{(M-1)} 1[\text{sign}(H_M(x_i)) \neq y_i] \quad (11.10)$$

where  $1[C]$  is 1 if  $C$  is true but 0 otherwise.

Each example is reweighted after an iteration that is,  $w_i^{(M-1)}$  is updated according to the classification performance of  $H_M$ :

$$\begin{aligned} w_i^{(M)} &= w_i^{(M-1)} \exp(-y_i \alpha_M h_M(x_i)) \\ &= \exp(-y_i H_M(x_i)) \end{aligned} \quad (11.11)$$

which is used for calculating the weighted error or another cost for training the weak classifier in the next round. This way, a more difficult example is associated with a larger weight so it is emphasized more in the next round of learning. The algorithm is summarized in Fig. 11.7.

### 11.3.4 Alternative Feature Selection Methods

In boosting based methods (cf. Fig. 11.7), the weak classifiers  $h_M$  and their related weights  $\alpha_M$  are determined simultaneously:  $h_M$  is chosen to minimize certain objective value (for example, weighted error rate of the feature) and  $\alpha_M$  is a function

- 
0. (Input)
    - (1) Training examples  $\mathcal{X} = \{(x_1, y_1), \dots, (x_N, y_N)\}$ ,  
where  $N = a + b$ ; of which  $a$  examples have  $y_i = +1$  and  $b$  examples have  $y_i = -1$ .
    - (2) The number  $M$  of weak classifiers to be combined.
  1. (Initialization)  
 $w_i^{(0)} = \frac{1}{2a}$  for those examples with  $y_i = +1$  or  $w_i^{(0)} = \frac{1}{2b}$  for those examples with  $y_i = -1$ .
  2. (Forward inclusion)
    - For  $m = 1, \dots, M$ :
      - (1) Choose optimal  $h_m$  to minimize the weighted error.
      - (2) Choose  $\alpha_m$  according to (11.9).
      - (3) Update  $w_i^{(m)} \leftarrow w_i^{(m)} \exp[-y_i \alpha_m h_m(x_i)]$  and normalize to  $\sum_i w_i^{(m)} = 1$ .
  3. (Output)
    - Classification function:  $H_M(x)$  as in (11.1).
    - Class label prediction:  $\hat{y}(x) = \text{sign}[H_M(x)]$ .
- 

**Fig. 11.7** AdaBoost learning algorithm

of the objective value. Wu et al. [47] show that if these tasks (learning  $h_M$  and setting  $\alpha_M$ ) are decoupled into two sequential steps, a more accurate strong classifier  $H_M$  can be obtained.

Different methods can be used to select features and train weak classifiers. Besides AdaBoost and other boosting variants, [47] showed that a greedy Forward Feature Selection (FFS) method can successfully select a subset of features from a large feature pool and learn corresponding weak classifiers. In boosting methods, a feature is selected if its corresponding weak classifier has minimum weighted error rate. FFS uses a different selection criterion that is directly related to the strong classifier's performance. In FFS, if a partial strong classifier  $H_{M-1}$  is already constructed, a feature  $h_{M^*}$  is selected in iteration  $M$  only if it leads to highest strong classifier accuracy, that is,  $H_{M-1} \cup h_{M^*}$  has the highest accuracy among all possible  $h_M$ . FFS uses majority vote (that is,  $\alpha_i = 1$  for all  $i$ ). A table of feature values are stored to ensure fast weak classifier training. FFS trains faster than the AdaBoost method, and achieves comparable but slightly lower detection accuracy than AdaBoost.

In fact, it is shown that AdaBoost is a sequential forward search procedure using the greedy selection strategy to minimize a certain margin on the training set [32]. Conceptually, FFS and AdaBoost shares the greedy feature selection idea, although different objective functions are used to guide the greedy search procedures.

A crucial heuristic assumption used in such a sequential forward search procedure is the monotonicity (that is, that addition of a new weak classifier to the current set does not decrease the value of the performance criterion). The premise offered by the sequential procedure in AdaBoost or FFS breaks down when this assumption is violated. Floating Search [29] is a sequential feature selection procedure with backtracking, aimed to deal with nonmonotonic criterion functions for feature selection. The sequential forward floating search (SFFS) methods [29] adds or deletes a single ( $\ell = 1$ ) feature and then backtracks  $r$  steps, where  $r$  depends on the current situation.

- 
0. (Input)
    - (1) Training examples  $\mathcal{X} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}$ ,  
where  $N = a + b$ ; of which  $a$  examples have  $y_i = +1$  and  $b$  examples have  $y_i = -1$ .
    - (2) The maximum number  $M_{\max}$  of weak classifiers.
    - (3) The cost function  $J(H_M)$ , and the maximum acceptable cost  $J^*$ .
  1. (Initialization)
    - (1)  $w_i^{(0)} = \frac{1}{2a}$  for those examples with  $y_i = +1$  or  $w_i^{(0)} = \frac{1}{2b}$  for those examples with  $y_i = -1$ .
    - (2)  $J_m^{\min} = \text{max-value}$  (for  $m = 1, \dots, M_{\max}$ ),  $M = 0$ ,  $\mathcal{H}_0 = \{\}$ .
  2. (Forward inclusion)
    - (1)  $M \leftarrow M + 1$ .
    - (2) Learn  $h_M$  and  $\alpha_M$ .
    - (3) Update  $w_i^{(M)} \leftarrow w_i^{(M-1)} \exp[-y_i \alpha_M h_M(x_i)]$ , normalize to  $\sum_i w_i^{(M)} = 1$ .
    - (4)  $\mathcal{H}_M = \mathcal{H}_{M-1} \cup \{h_M\}$ ;  
If  $J_M^{\min} > J(H_M)$ , then  $J_M^{\min} = J(H_M)$ .
  3. (Conditional exclusion)
    - (1)  $h' = \arg \min_{h \in \mathcal{H}_M} J(H_M - h)$ .
    - (2) If  $J(H_M - h') < J_{M-1}^{\min}$ , then
      - (a)  $\mathcal{H}_{M-1} = \mathcal{H}_M - h'$ .  
 $J_{M-1}^{\min} = J(H_M - h')$ ;  $M = M - 1$ .
      - (b) If  $h' = h_{m'}$ , then  
recalculate  $w_i^{(j)}$  and  $h_j$  for  $j = m', \dots, M$ .
      - (c) Go to 3.(1).
    - (3) Else
      - (a) If  $M = M_{\max}$  or  $J(\mathcal{H}_M) < J^*$ , then go to 4.
      - (b) Go to 2.(1).
  4. (Output)
 

Classification function:  $H_M(x)$  as in (11.1).  
Class label prediction:  $\hat{y}(x) = \text{sign}[H_M(x)]$ .

---

**Fig. 11.8** FloatBoost algorithm

The FloatBoost Learning procedure is shown in Fig. 11.8. It is composed of several parts: the training input, initialization, forward inclusion, conditional exclusion, and output. In step 2 (forward inclusion), the currently most significant weak classifiers are added one at a time, which is the same as in AdaBoost. In step 3 (conditional exclusion), FloatBoost removes the least significant weak classifier from the set  $\mathcal{H}_M$  of current weak classifiers, subject to the condition that the removal leads to a lower cost than  $J_{M-1}^{\min}$ . Supposing that the weak classifier removed was the  $m'$ th in  $\mathcal{H}_M$ , then  $h_{m'}, \dots, h_{M-1}$  and the  $\alpha_m$ 's must be relearned. These steps are repeated until no more removals can be done.

### 11.3.5 Asymmetric Learning Methods

The face detection (and other object detection) problem is a rare-event detection problem [47], in the sense that the face (or target object) only occupies a small

number of subwindows while the nonface (or nonobject) subwindows are on the order of millions even in a small-sized image. This asymmetric nature of the classifier learning problem is long recognized and methods have been proposed to build a strong classifier that takes into account the asymmetry property.

Viola and Jones proposed the AsymBoost method [44], which is a modification of the AdaBoost algorithm. The essence of AsymBoost is to focus more on positive examples by changing the weight update rule (11.11) to

$$w_i^{(M)} = C \exp(-y_i H_M(x_i)), \quad (11.12)$$

where  $C = (\sqrt{K})^{(1/T)}$  if  $y_i > 0$  and  $C = (\sqrt{K})^{(-1/T)}$  if  $y_i < 0$ , and  $K > 1$  is a parameter that measures that level of asymmetry. We assume that the boosting learning procedure will repeat  $T$  rounds. In the  $T$  rounds of AdaBoost algorithm, positive examples are continuously assigned higher weights than negative examples.

Wu et al. propose another asymmetric learning method. Wu et al. [47] shows that it is advantageous to adjust the values of  $\alpha_i$  after the features are selected, according to the cascade detection framework. Assuming the false alarm rate of all strong classifiers in a cascade is 0.5, a 20-node cascade will have a  $10^{-6}$  false alarm rate if we assume the strong classifiers reject nonfaces independent to each other. Thus, Wu et al. propose the following learning goal for the strong classifiers in a cascade: “*for every node, design a classifier with very high (e.g. 99.9%) detection rate and only moderate (e.g., 50%) false positive rate.*” Linear Asymmetric Classifier (LAC) is designed to find the  $\alpha$  that achieve this goal.

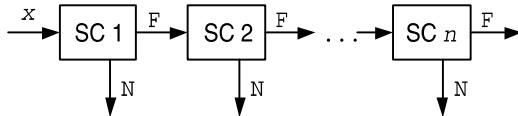
Given weak classifiers  $h_1, h_2, \dots, h_M$ , an example  $x$  is mapped to a vector of responses  $\mathbf{h}(x) = (h_1(x), h_2(x), \dots, h_M(x))$ . LAC computes the distributions of the vector  $\mathbf{h}(x)$ :  $\mu_+$  and  $\Sigma_+$  are mean and covariance matrix of  $\mathbf{h}(x)$  when  $x$  is the set of faces. Similarly,  $\mu_-$  and  $\Sigma_-$  are the mean and covariance matrix computed using nonfaces. It is showed in [47] that the following LAC solution vector  $\alpha^* \in \mathbb{R}^M$  is globally optimal for the cascade learning goal under certain reasonable assumptions:

$$\alpha^* = \Sigma_+^{-1}(\mu_+ - \mu_-). \quad (11.13)$$

Another way to set the  $\alpha$  vector is to use the Fisher’s Discriminant Analysis (FDA). Experiments in [47] show that using LAC or FDA to set the  $\alpha$  vector consistently improve cascade detection accuracy, no matter the weak classifiers are selected and trained using AdaBoost or FFS.

### 11.3.6 Cascade of Strong Classifiers

A boosted strong classifier effectively eliminates a large portion of nonface subwindows while maintaining a high detection rate. Nonetheless, a single strong classifier may not meet the requirement of an extremely low false alarm rate (for example,  $10^{-6}$  or even lower). A solution is to arbitrate between several detectors (strong classifier) [31], for example, using the “AND” operation.



**Fig. 11.9** A cascade of  $n$  strong classifiers (SC). The input is a subwindow  $x$ . It is sent to the next SC for further classification only if it has passed all the previous SCs as the face (F) pattern; otherwise it exits as nonface (N).  $x$  is finally considered to be a face when it passes all the  $n$  SCs

Viola and Jones [43, 45] further extend this idea by training a cascade consisting of a cascade of strong classifiers, as illustrated in Fig. 11.9. A strong classifier is trained using bootstrapped nonface examples that pass through the previously trained cascade. Usually, 10 to 20 strong classifiers are cascaded. For face detection, subwindows that fail to pass a strong classifier are not further processed by the subsequent strong classifiers. This strategy can significantly speed up the detection and reduce false alarms, with a little sacrifice of the detection rate.

Various improvements have also been proposed to the cascade structure. Xiao et al. argue that historical information is useful during the cascade training [48], that is, we should not ignore the information contained in the strong classifiers  $SC_1, \dots, SC_{M-1}$  when we train the  $M$ -th strong classifier  $SC_M$  (which is the practice in Fig. 11.7). The Boosting Chain framework is proposed in [48] to incorporate such historical information: the strong classifier  $SC_{M-1}$  is treated as the first “weak classifier” in  $SC_M$ . This modification to the cascade framework reduces the number of required weak classifiers and increases detection accuracy [6, 48].

Another attempt to modify the cascade framework is the soft cascade method [5] or similar ideas [28, 49]. Soft cascade is an extreme cascade structure: a “monolithic” strong classifier composed of multiple weak classifiers, much similar to the strong classifier in the cascade framework. Let  $c_1(x), \dots, c_T(x)$  be the weak classifiers that form a soft cascade:

$$H_T(x) = \sum_{i=1}^T c_i(x). \quad (11.14)$$

A soft cascade associates a rejection threshold  $r_t$  for every partial strong classifier  $H_t(x) = \sum_{i=1}^t c_i(x)$ . If  $H_t(x) < r_t$ , the input subwindow  $x$  is rejected as nonface and the weak classifiers  $c_{t+1}, \dots, c_T$  are not evaluated. In other words, a soft cascade is similar to a cascade structure that requires only 1 weak classifier per node. However, since historical information is preserved in  $H_t$ , soft cascades achieve high detection performances.

After the weak classifiers  $c_1, \dots, c_T$  are trained, the soft cascade method rearranges the order of these weak classifiers. This step is carried out using a separate set of validation examples. An optimal ordering of weak classifiers and rejection thresholds  $r_t$  are chosen to minimize both the detection errors and testing time computational costs [5].

Similar ideas are proposed to improve the original cascade framework in [6]. Suppose that a cascade consists of strong classifiers  $SC_1, \dots, SC_M$ , where  $SC_i$  is

trained using the AdaBoost method. By adjusting the threshold of the strong classifier, we can get different strong classifier performance in terms of false alarm rates and detection rates. This threshold is usually determined manually by setting a fixed goal for either detection or false alarm rate, which not necessarily leads to optimal cascade detection performance. Brubaker et al. proposed a “two-point” algorithm to automatically find optimal thresholds of strong classifiers. The two-point algorithm uses less weak classifiers than fixed goal AdaBoost (and thus faster detection speed), and achieves higher detection performances.

## 11.4 Dealing with Head Rotations

Multiview face detection should be able to detect nonfrontal faces. Face detection methods usually handle two major types of head rotation: (1) out-of-plane (left-right) rotation; (2) in-plane rotation.

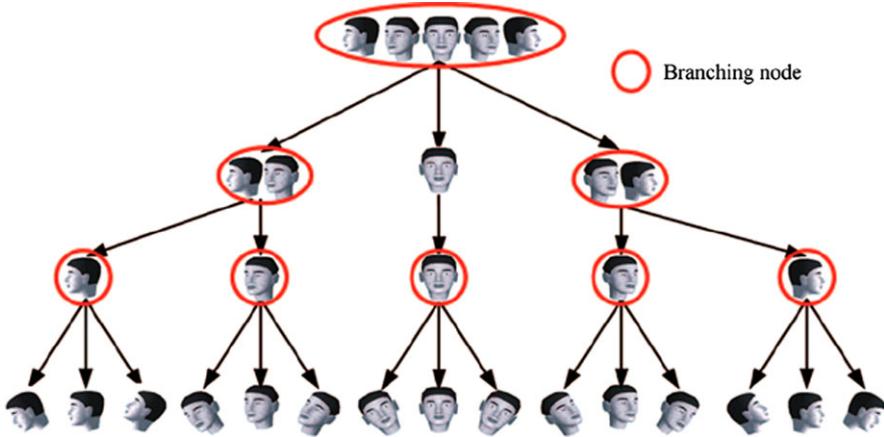
Rowley et al. [30] propose to use two neural network classifiers for detection of frontal faces subject to in-plane rotation. The first is the router network, trained to estimate the orientation of an assumed face in the subwindow, though the window may contain a nonface pattern. The inputs to the network are the intensity values in a preprocessed  $20 \times 20$  subwindow. The angle of rotation is represented by an array of 36 output units, in which each unit represents an angular range. With the orientation estimate, the subwindow is derotated to make the potential face upright. The second neural network is a normal frontal, upright face detector.

Within the cascade detector framework, detector-pyramids have been proposed to detect and merge faces in different poses and have achieved the state-of-the-art detection performance.

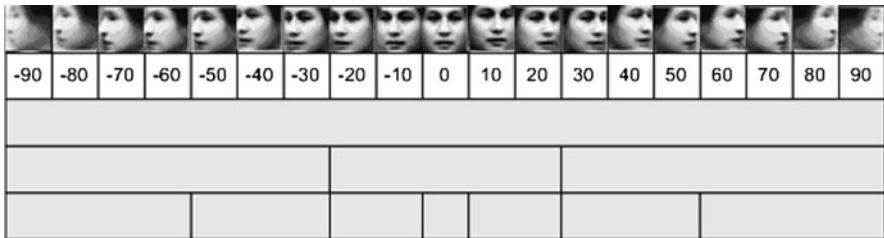
### 11.4.1 Hierarchical Organization of Multi-view Faces

The Width-First-Search structure (Fig. 11.10) by Huang et al. in [14] handles in-plane and out-of-plane rotations simultaneously. Huang et al. [14] manually divides the face range into 15 different poses, and arranges such poses in a four level tree structure. The top level tree node includes all face poses. The second level contains 3 nodes, which correspond to left profile, frontal, and right profile faces. The third level further refines to 5 nodes, where left and right profile faces are split into 2 different nodes based on the out-of-plane rotation angle. The first 3 levels handle out-of-plane rotations. Each node in the third level is split to 3 nodes in the final level, handling different in-plane rotation angles.

The tree structure in [14] handles out-of-plane rotation in  $\Theta = [-90^\circ, +90^\circ]$  and in-plane rotation in  $\Phi_2 = [-45^\circ, +45^\circ]$ . The full in-plane rotation range  $\Phi = [-180^\circ, +180^\circ]$  is covered by rotating the features  $90^\circ$ ,  $180^\circ$ , and  $270^\circ$ . It is noticed that for these specific rotation angles, rotating the features is equivalent to rotate the



**Fig. 11.10** Illustration of the structure for detecting multi-view faces. From Huang et al. [14], © 2007 IEEE, with permission



**Fig. 11.11** Out-of-plane view partition. Out-of-plane head rotation (row 1), the facial view labels (row 2), and the coarse-to-fine view partitions at the three levels of the detector-pyramid (rows 3 to 5)

mask  $m$  associated with a feature by a corresponding angle. Rotating features is more efficient than rotating images.

A similar hierarchy is used by Li et al. [17, 19] to handle out-of-plane rotation in  $[-90^\circ, +90^\circ]$ , shown in Fig. 11.11. It is worth noting that the face-pose hierarchy in Fig. 11.11 does not handle in-plane rotations. The leaf detectors are designed to handle in-plane rotations in the range  $[-15^\circ, +15^\circ]$ . The full in-plane rotation in  $\Phi = [-45^\circ, +45^\circ]$  is dealt with by also applying the detector-pyramid on the rotated test images ( $\pm 30^\circ$ ).

### 11.4.2 From Face-Pose Hierarchy to Detector-Pyramid

A detector-pyramid that detects multi-view faces can be derived directly from the face-pose hierarchy. Every node in Fig. 11.10 corresponds to a strong classifier. For example, the root node determines whether a subwindows contains a left profile,

**Table 11.1** Desired vectors for different face poses and non-faces in Vector Boosting training

| Cases              | Desired output vector                       |
|--------------------|---------------------------------------------|
| Left profile face  | (1, 0, 0, 0)                                |
| Frontal face       | (0, 1, 0, 0)                                |
| Right profile face | (0, 0, 1, 0)                                |
| Nonfaces           | (−1, 0, 0, 0), (0, −1, 0, 0), (0, 0, −1, 0) |



**Fig. 11.12** Merging from different channels. From left to right: Outputs of frontal, left, and right view channels and the final result after the merge

frontal, or right profile face. It is important to note that multiple children nodes of the same parent node can be activated simultaneously (that is, a width-first-search of a tree structure). Huang et al. made this choice based on the following argument: different face poses are jointly competing with nonfaces, and the discrimination among them is less important (except in the final level). This hypothesis requires nodes in the top 3 levels to be multi-class classifiers, and a Vector Boosting algorithm is proposed in [14] to satisfy this special requirement.

For example, a partial profile face with a 45° out-of-plane rotation angle can be detected by both the right profile face node and the frontal face node in the second layer. In order to achieve higher detection accuracy, it is reasonable to further examine the two sub-trees rooted at both nodes. In the Vector Boosting classifier for the root node, the desired output is a vector and the desired vectors for different cases are summarized in Table 11.1.

At the final level, the single node with the highest confidence is chosen as the detected face pose. In order to get a classifier with very low false alarm rate, the classifier in the leaf node of the width-first-search tree in [14] is in fact a cascade detector.

Figure 11.11 leads to another detector pyramid. Instead of using a multi-class boosting algorithm that generates a vector output, [17] uses  $k$  binary RealBoost strong classifiers if a node has  $k$  children nodes. Multiple children of a node can be activated (that is, further examining the subtree rooted at a child node) if more than one binary RealBoost classifiers output positively. At the final level, multiple leaf nodes (corresponding to different face poses) can be active for one subwindow. Different from [14], faces detected by the seven channels at the final level of Fig. 11.11 are merged to obtain the final result. This is illustrated in Fig. 11.12.

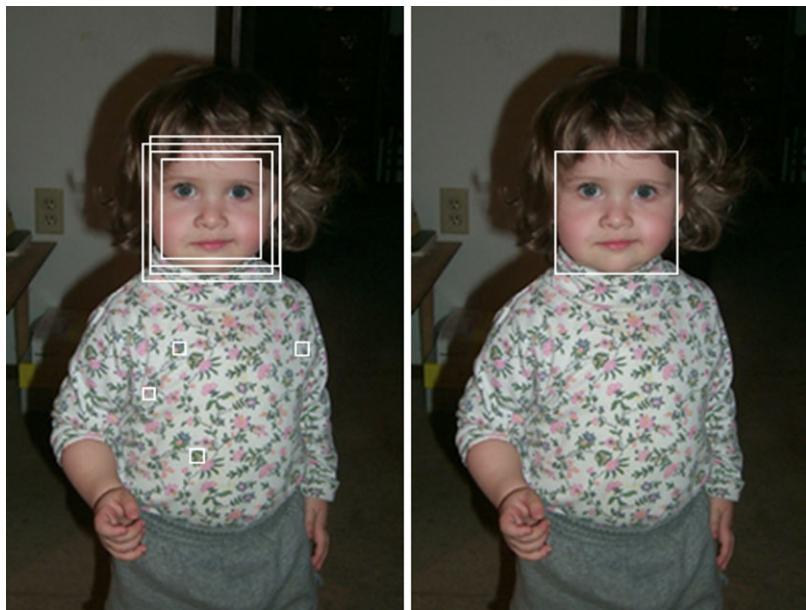


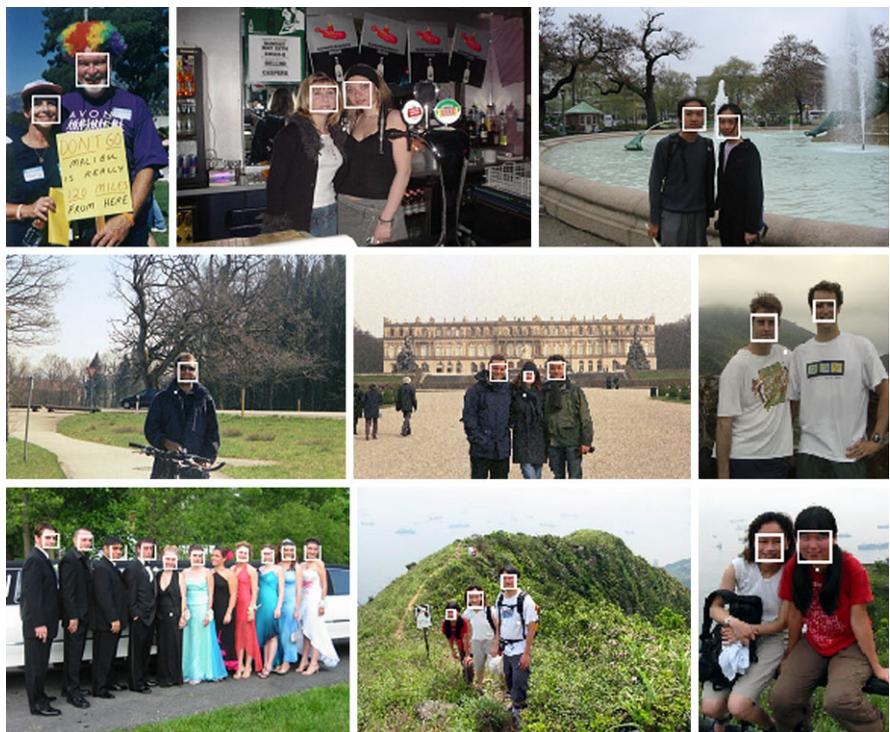
Fig. 11.13 Merging multiple detections

## 11.5 Postprocessing

A single face in an image may be detected several times at close locations or on multiple scales. False alarms may also occur but usually with less consistency than multiple face detections. The number of multiple detections in a neighborhood of a location can be used as an effective indication for the existence of a face at that location. This assumption leads to a heuristic for resolving the ambiguity caused by multiple detections and eliminating many false detections. A detection is confirmed if the number of multiple detections is greater than a given value; and given the confirmation, multiple detections are merged into a consistent one. This is practiced in most face detection systems [31, 38]. Figure 11.13 gives an illustration. The image on the left shows a typical output of initial detection, where the face is detected four times with four false alarms on the cloth. On the right is the final result after merging. After the postprocessing, multiple detections are merged into a single face and the false alarms are eliminated. Figures 11.14 and 11.15 show some typical frontal and multiview face detection examples; the multiview face images are from the Carnegie Mellon University (CMU) face database [42].

## 11.6 Performance Evaluation

The result of face detection from an image is affected by the two basic components: the face/nonface classifier and the postprocessing (merger). To understand



**Fig. 11.14** Results of frontal face detection

how the system works, it is recommended that the two components be evaluated separately [1], with two types of test data. The first consists of face icons of a fixed size (as are used for training). This process aims to evaluate the performance of the face/nonface classifier (preprocessing included), without being affected by merging. The second type of test data consists of normal images. In this case, the face detection results are affected by both trained classifier and merging; the overall system performance is evaluated.

### 11.6.1 Performance Measures

The face detection performance is primarily measured by two rates: the correct detection rate (which is 1 minus the miss rate) and the false alarm rate. The performance can be observed by plotting on the receiver operating characteristic (ROC) curves.

The false alarm rate is computed as the percentage of the subwindows that are nonfaces but wrongly classified as faces. However, the number of false detections (remaining after merging multiple detections) is a better suited metric because it

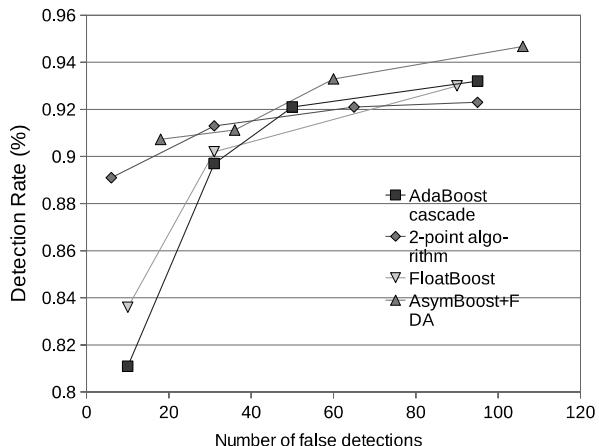


**Fig. 11.15** Results of multiview face detection

reflects the effect of postprocessing and directly links to the final output of a face detection system. Although the false alarm rate is usually positively correlated with the number of false detections, recently more authors are reporting number of false detections (after postprocessing) in the X-axis of the ROC curves. Figure 11.16 shows several examples of the ROC curves, including four recent methods evaluated on the benchmark MIT-CMU frontal face dataset [42].

An ideal face detection system should have a detection rate of 100%, with a false alarm rate of 0, though none of the current systems can achieve this generally. In practical systems, increasing the detection rate is usually accompanied by an increase in the false alarm rate. In the case where a confidence function is used to

**Fig. 11.16** Typical ROC curve for face detection on the MIT-CMU frontal face dataset. The algorithms include the AdaBoost implementation in [6], the 2-point algorithm [6], FloatBoost [17], and AsymBoost + FDA [47]



distinguish between the face and nonface subwindows, with the high output value indicating the detection of face and low value nonface, a trade-off between the two rates can be made by adjusting the decisional threshold. In the case of the AdaBoost learning method, the threshold for (11.1) is learned from the training face icons and bootstrapped nonface icons, so a specified rate (usually the false alarm rate) is under control for the training set. Remember that performance numbers of a system are always with respect to the data sets used; two algorithms or systems cannot be compared directly unless the same data sets are used.

### 11.6.2 Comparison of Cascade-Based Detectors

As the cascade-based methods (with local features) have so far provided the best face detection solutions in terms of the statistical rates and the speed, the following provides a comparative evaluation on different boosting algorithms (DAB: discrete AdaBoost; RAB: real AdaBoost; and GAB: gentle AdaBoost), different training sets preparations, and different weak classifiers. The results provide empirical references for face detection engineers.

- **Boosting Algorithms.** Three 20-stage cascade classifiers were trained with DAB, RAB, and GAB using the Haar-like feature set of Viola and Jones [43, 45] and stumps as the weak classifiers. It is reported that GAB outperformed the other two boosting algorithms [21]. Also, a smaller rescaling factor for scanning images was beneficial for a high detection rate.
- **Weak Classifiers.** Stumps are the simplest tree type of weak classifiers (WCs) that can be used in discrete AdaBoost. A stump is a single-node tree that does not allow learning dependence between features. In general,  $n$  split nodes are needed to model dependence between  $n - 1$  variables. It is reported in [6] that using a decision tree as weak classifier, the cascade achieves up to 15% higher detection

**Table 11.2** Average number of features evaluated per nonface subwindow of size  $20 \times 20$  (reproduced from Lienhart et al. [21])

| AdaBoost type | Number of splits |       |       |       |
|---------------|------------------|-------|-------|-------|
|               | 1                | 2     | 3     | 4     |
| DAB           | 45.09            | 44.43 | 31.86 | 44.86 |
| GAB           | 30.99            | 36.03 | 28.58 | 35.40 |
| RAB           | 26.28            | 33.16 | 26.73 | 35.71 |

rate with the same number of false detections. Other complex weak classifier (for example, piece-wise decision functions) can also increase detection performance.

- **Detection Speed.** Table 11.2 compares the CART tree weak classifiers of varying number of nodes in terms of the effectiveness of rejecting nonface subwindows. RAB is the most effective. It is also reported in [6] that RAB has the fastest detection speed. Reusing historical information [48] is confirmed to be effective in reducing testing time by [6].
- **Haar-like and Other Local Features.** The experiments in [21] and other works (for example, [14, 27]) suggest that whereas the larger Haar-like feature set makes it more complex in both time and memory in the boosting learning phase, gain is obtained in the detection phase. Using approximately the same training time, [27] (with 295 920 local features) reported about 5% lower false alarms than the method in Fig. 11.6 (with 40 000 local features). Other local features such as MB-LBP also exhibits excellent detection results [52].
- **Subwindow Size.** Different subwindow sizes, ranging from  $16 \times 16$  up to  $32 \times 32$ , have been used on face detection. The experiments [21] show that a subwindow size of  $20 \times 20$  achieves the highest detection rate at an absolute number of false alarms between 5 and 100 on the CMU test set of frontal faces. A subwindow size of  $24 \times 24$  worked better for false alarms fewer than five.

## 11.7 Conclusions

Face detection is the first step in automated face recognition and has applications in biometrics and multimedia management. Owing to the complexity of the face and nonface manifolds, highly accurate face detection with a high detection rate and low false alarm rate has been challenging. Now this difficult problem has almost been solved to meet the minimum requirements of most practical applications, because of the advances in face recognition research and machine learning.

Boosting-based face detection methods [14, 17, 19, 21, 43, 45, 47] have been the most effective of all those developed so far. In terms of detection and false alarm rates, they are comparable to the neural network method of Rowley et al. [31], but are several times faster.

Regarding the boosting based approach, the following conclusions can be drawn in terms of feature sets, boosting algorithms, weak classifiers, subwindow sizes, and training set sizes according to reported studies [14, 17, 19, 21, 43, 45, 47]:

- An over-complete set of Haar-like features are effective for face detection. The use of the integral image method makes computation of these features efficient and achieves scale invariance. Extended Haar-like features help detect nonfrontal faces.
- AdaBoost learning can select best subset from a large feature set and construct a powerful nonlinear classifier.
- The cascade structure significantly improves the detection speed and effectively reduces false alarms, with a little sacrifice of the detection rate.
- Selecting the weak classifiers and learning the weights that combine those weak classifiers can be decoupled.
- Alternative feature selection methods can be used to reduce training time (for example, FFS), or to achieve lower error rate or detection time (for example, FloatBoost).
- Asymmetry needs to be taken care of in learning classifiers for face detection. Weights that are specifically learned to satisfy learning goals in the cascade framework (for example, using LAC) improve detection accuracy.
- Less aggressive versions of AdaBoost, such as GentleBoost and LogitBoost, may be preferable to discrete and real AdaBoost in dealing with training data containing outliers [12].
- Representationally, more complex weak classifiers such as small CART trees can model second-order and/or third-order dependencies, and may be beneficial for the nonlinear task of face detection.

Although face detection technology is now sufficiently mature to meet the minimum requirements of many practical applications, much work is still needed before automatic face detection can achieve performance comparable to the human performance. The Haar + AdaBoost approach is effective and efficient. However, the current approach has almost reached its power limit. Within such a framework, improvements may be possible by designing additional sets of features that are complementary to the existing ones and adopting more advanced learning techniques, which could lead to more complex classifiers while avoiding the overfitting problem.

**Acknowledgements** This work was partially supported by the Chinese National Natural Science Foundation Project #61070146, the National Science and Technology Support Program Project #2009BAK43B26, and the AuthenMetric R&D Funds (2004–2011). The work was also partially supported by the TABULA RASA project (<http://www.tabularasa-euproject.org>) under the Seventh Framework Programme for research and technological development (FP7) of the European Union (EU), grant agreement #257289.

## References

1. Alvira, M., Rifkin, R.: An empirical comparison of SNoW and svms for face detection. Technical Report AI Memo 2001-004 & CBCL Memo 193, MIT (2001)
2. Amit, Y., Geman, D., Wilder, K.: Joint induction of shape features and tree classifiers. IEEE Trans. Pattern Anal. Mach. Intell. **19**, 1300–1305 (1997)

3. Baker, S., Nayar, S.: Pattern rejection. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 544–549 (1996)
4. Bichsel, M., Pentland, A.P.: Human face recognition and the face image set's topology. CVGIP, Image Underst. **59**, 254–261 (1994)
5. Bourdev, L.D., Brandt, J.: Robust object detection via soft cascade. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. II, pp. 236–243 (2005)
6. Brubaker, S.C., Wu, J., Sun, J., Mullin, M.D., Rehg, J.M.: On the design of cascades of boosted ensembles for face detection. Int. J. Comput. Vis. **77**(1–3), 65–86 (2008)
7. Crow, F.: Summed-area tables for texture mapping. In: SIGGRAPH, vol. 18(3), pp. 207–212 (1984)
8. Elad, M., Hel-Or, Y., Keshet, R.: Pattern detection using a maximal rejection classifier. Pattern Recognit. Lett. **23**, 1459–1471 (2002)
9. Feraud, J., Bernier, O., Collobert, M.: A fast and accurate face detector for indexation of face images. In: Proc. Fourth IEEE Int. Conf on Automatic Face and Gesture Recognition, Grenoble (2000)
10. Fleuret, F., Geman, D.: Coarse-to-fine face detection. Int. J. Comput. Vis. **20**, 1157–1163 (2001)
11. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. **55**(1), 119–139 (1997)
12. Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. Technical report, Department of Statistics, Sequoia Hall, Stanford University, July 1998
13. Huang, J., Shao, X., Wechsler, H.: Face pose discrimination using support vector machines (SVM). In: Proceedings of International Conference Pattern Recognition, Brisbane, Queensland, Australia (1998)
14. Huang, C., Ai, H., Li, Y., Lao, S.: High-performance rotation invariant multiview face detection. IEEE Trans. Pattern Anal. Mach. Intell. **29**(4), 671–686 (2007)
15. Küblbeck, C., Ernst, A.: Face detection and tracking in video sequences using the modified census transformation. Image Vis. Comput. **24**(6), 564–572 (2006)
16. Kuchinsky, A., Pering, C., Creech, M.L., Freeze, D., Serra, B., Gwizdka, J.: FotoFile: A consumer multimedia organization and retrieval system. In: Proceedings of ACM SIG CHI'99 Conference, Pittsburg, May 1999
17. Li, S.Z., Zhang, Z.: FloatBoost learning and statistical face detection. IEEE Trans. Pattern Anal. Mach. Intell. **26**(9), 1112–1123 (2004)
18. Li, S.Z., Zhang, Z.Q., Shum, H.-Y., Zhang, H.: FloatBoost learning for classification. In: Proceedings of Neural Information Processing Systems, Vancouver (2002)
19. Li, S.Z., Zhu, L., Zhang, Z.Q., Blake, A., Zhang, H., Shum, H.: Statistical learning of multi-view face detection. In: Proceedings of the European Conference on Computer Vision, vol. 4, pp. 67–81, Copenhagen, Denmark, 28 May–2 June 2002
20. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block local binary patterns for face recognition. In: International Conference on Biometrics, pp. 828–837 (2007)
21. Lienhart, R., Kuranov, A., Pisarevsky, V.: Empirical analysis of detection cascades of boosted classifiers for rapid object detection. MRL Technical Report, Intel Labs, December 2002
22. Liu, C., Shum, H.-Y.: Kullback–Leibler boosting. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. I, pp. 587–594 (2003)
23. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object representation. IEEE Trans. Pattern Anal. Mach. Intell. **7**, 696–710 (1997)
24. Osuna, E., Freund, R., Girosi, F.: Training support vector machines: An application to face detection. In: CVPR, pp. 130–136 (1997)
25. Papageorgiou, C.P., Oren, M., Poggio, T.: A general framework for object detection. In: Proceedings of IEEE International Conference on Computer Vision, pp. 555–562, Bombay (1998)
26. Pentland, A.P., Moghaddam, B., Starner, T.: View-based and modular eigenspaces for face recognition. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 84–91 (1994)

27. Pham, M.-T., Cham, T.-J.: Fast training and selection of Haar features using statistics in boosting-based face detection. In: Proceedings of IEEE International Conference on Computer Vision (2007)
28. Pham, M.-T., Hoang, V.-D.D., Cham, T.-J.: Detection with multi-exit asymmetric boosting. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2008)
29. Pudil, P., Novovicova, J., Kittler, J.: Floating search methods in feature selection. *Pattern Recognit. Lett.* **15**(11), 1119–1125 (1994)
30. Rowley, H., Baluja, S., Kanade, T.: Rotation invariant neural network-based face detection. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (1998)
31. Rowley, H.A., Baluja, S., Kanade, T.: Neural network-based face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(1), 23–28 (1998)
32. Schapire, R., Freund, Y., Bartlett, P., Lee, W.S.: Boosting the margin: A new explanation for the effectiveness of voting methods. *Ann. Stat.* **26**(5), 1651–1686 (1998)
33. Schneiderman, H.: A statistical approach to 3D object detection applied to faces and cars (CMU-RI-TR-00-06). PhD thesis, RI (2000)
34. Schneiderman, H., Kanade, T.: A statistical method for 3D object detection applied to faces and cars. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2000)
35. Schneiderman, H., Kanade, T.: Object detection using the statistics of parts. *Int. J. Comput. Vis.* **56**(3), 151–177 (2004)
36. Simard, P.Y., Bottou, L., Haffner, P., Cun, Y.L.: Boxlets: a fast convolution algorithm for signal processing and neural networks. In: Kearns, M., Solla, S., Cohn, D. (eds.) *Advances in Neural Information Processing Systems*, vol. 11, pp. 571–577. MIT Press, Cambridge (1998)
37. Simard, P.Y., Cun, Y.A.L., Denker, J.S., Victorri, B.: Transformation invariance in pattern recognition—tangent distance and tangent propagation. In: Orr, G.B., Muller, K.-R. (eds.) *Neural Networks: Tricks of the Trade*. Springer, New York (1998)
38. Sung, K.-K., Poggio, T.: Example-based learning for view-based human face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(1), 39–51 (1998)
39. Tieu, K., Viola, P.: Boosting image retrieval. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 228–235 (2000)
40. Turk, M.: A random walk through eigenspace. *IEICE Trans. Inf. Syst.* **E84-D**(12), 1586–1695 (2001)
41. Turk, M.A., Pentland, A.P.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–86 (1991)
42. Various Face Detection Databases. [www.ri.cmu.edu/projects/project\\_419.html](http://www.ri.cmu.edu/projects/project_419.html)
43. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii (2001)
44. Viola, P.A., Jones, M.J.: Fast and robust classification using asymmetric AdaBoost and a detector cascade. In: *Advances in Neural Information Processing Systems*, vol. 14, pp. 1311–1318 (2001)
45. Viola, P., Jones, M.J.: Robust real-time face detection. *Int. J. Comput. Vis.* **57**(2), 137–154 (2004)
46. Wiskott, L., Fellous, J., Kruger, N., v. d. Malsburg, C.: Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 775–779 (1997)
47. Wu, J., Brubaker, S.C., Mullin, M.D., Rehg, J.M.: Fast asymmetric learning for cascade face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(3), 369–382 (2008)
48. Xiao, R., Zhu, L., Zhang, H.J.: Boosting chain learning for object detection. In: Proceedings of IEEE International Conference on Computer Vision, pp. 709–714 (2003)
49. Xiao, R., Zhu, H., Sun, H., Tang, X.: Dynamic cascades for face detection. In: Proceedings of IEEE International Conference on Computer Vision (2007)

50. Yang, M.-H., Kriegman, D., Ahuja, N.: Detecting faces in images: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(1), 34–58 (2002)
51. Yang, M.-H., Roth, D., Ahuja, N.: A SNoW-based face detector. In: *Proceedings of Neural Information Processing Systems*, pp. 855–861 (2000)
52. Zhang, L., Chu, R., Xiang, S., Liao, S., Li, S.Z.: Face detection based on multi-block LBP representation. In: *International Conference on Biometrics*, pp. 11–18 (2007)



# Chapter 12

## Facial Landmark Localization

Xiaoqing Ding and Liting Wang

### 12.1 Introduction

Face detection and recognition is a vibrant area of biometrics with active research and commercial efforts over the last 20 years. The task of face detection is to search faces in images, reporting their positions by a bounding box. Recent studies [19, 31] have shown that face detection has already been a state-of-the-art technology in both accuracy and speed. However, face detection is not sufficient to acquire facial landmarks, for example, eye contours, mouth corners, nose, eyebrows, etc. This is therefore the task of facial landmark localization which aims to find the accurate positions of the facial feature points as illustrated in Fig. 12.1. It is a fundamental and significant work in face-related areas, for example, face recognition, face cartoon/sketch, face pose estimate, model-based face tracking, eye/mouth motion analysis, 3D face reconstruction, etc.

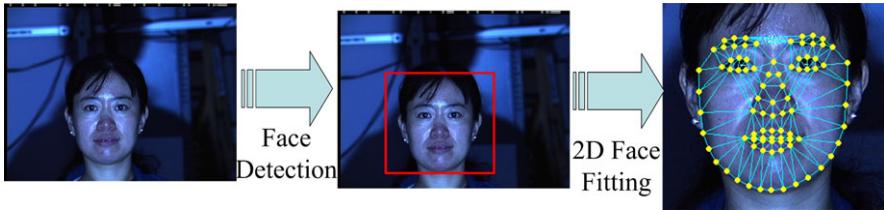
There is a wide variety of works related to facial landmark localization. The early researches extract facial landmarks without a global model. Facial landmarks, such as the eye corners and centers, the mouth corners and center, the nose corners, chin and cheek borders are located based on geometrical knowledge. The first step consists of the establishment of a rectangular search region for the mouth and a rectangular search region for the eyes. The borders are extracted by applying corner detection algorithm such as SUSAN border extraction algorithm [17]. Such methods are fast, however, they could not deal with faces of large variation in appearance due to pose, rotation, illumination and background changes.

---

X. Ding (✉) · L. Wang

State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China  
e-mail: [dingxq@tsinghua.edu.cn](mailto:dingxq@tsinghua.edu.cn)

L. Wang  
e-mail: [wangltmail@tsinghua.edu.cn](mailto:wangltmail@tsinghua.edu.cn)



**Fig. 12.1** Facial landmark localization

Different with the earlier model-independent algorithm, some researches focus on model-dependent algorithm. Hsu and Jain [18] propose an approach which represents human faces semantically via facial components such as eyes, mouth, face outline, and the hair outline. Each facial component is encoded by a closed (or open) snake that is drawn from a 3D generic face model. The face shape model here is not based on statistical learning and it still could not deal with faces of large variation in appearance due to pose, rotation, illumination and background changes.

With the prominent successful research of Active Shape Model (ASM) [7, 9, 3, 8] and Active Appearance Model (AAM) [4–6, 10–13], face shape is well modeled as a linear combination of principal modes (major eigenvectors) learned from the training face shapes. By learning statistical distribution of shapes and textures from training database, a deformable shape model is built. The boundary of objects with similar shapes to those in the training set could be extracted by fitting this deformable model to images. Depending on the different tasks, ASM and AAM can be built in different ways. On one hand, we might construct a person specific ASM or AAM across pose, illumination, and expression. Such a person-specific model might be useful for interactive user interface applications including head pose estimation, gaze estimation etc. On the other hand, we might construct ASM or AAM to fit any face, including faces unseen in training set. Evidence suggests that the performance of the person-specific facial landmark localization is substantially better than the performance of generic facial landmark localization. As indicated in [15], Gross's experimental results confirm that generic facial landmark localization is far harder than person-specific facial landmark localization and the performance degrades quickly when fitting to images which are unseen in the training set.

In recent years, there are several improved research works based on the framework of AAM. Papandreou and Maragos [27] introduce two enhancements to inverse-compositional AAM matching algorithms in order to overcome the limitation when inverse-compositional AAM matching algorithms are used in conjunction with models exhibiting significant appearance variation, such as AAMs trained on multiple-subject human face images. Liu Xiaoming [25, 26] proposes a discriminative framework to greatly improve the robustness, accuracy and efficiency of face alignment for unseen data. Liebelt et al. [24] develop an iterative multi-level algorithm that combines AAM fitting and robust 3D shape alignment. Xiao et al. [33] also develop the research work of combining 2D AAM and 3D Morphable Model (3DMM). Hamsici and Martinez [16] derive a new approach carries the advantages of AAM and 3DMM that can model nonlinear changes in examples without the

need of a pre-alignment step. Lee and Kim [21] propose a tensor-based AAM that can handle a variety of subjects, poses, expressions, and illuminations in the tensor algebra framework. They reported Tensor-based AAM reduced the fitting error of the conventional AAM by about two pixels and the computation time by about 0.6 second.

There are also several improved research works based on the framework of ASM. Tu et al. [30] propose a hierarchical CONDENSATION framework to estimate the face configuration parameter under the framework of ASM. Jiao et al. [20] present a W-ASM, in which Gabor wavelet features are used for modeling local image structure. Zhang and Ai [34] propose an AdaBoost discriminative framework which improves the accuracy, efficiency, and robustness of ASM. The same research works are also carried on by Li and Ito [23] who describe a modeling method by using AdaBoosted histogram classifiers. Brunet et al. [2] define a new criterion to select landmarks that have good generalization properties. Vogler et al. [32] combine the ASM with 3D deformable model which governs the overall shape, orientation and location.

In the following, we will introduce a coarse-to-fine facial landmark localization algorithm which uses discriminant learning to remedy the generalization problems based on the framework of Active Shape Model.

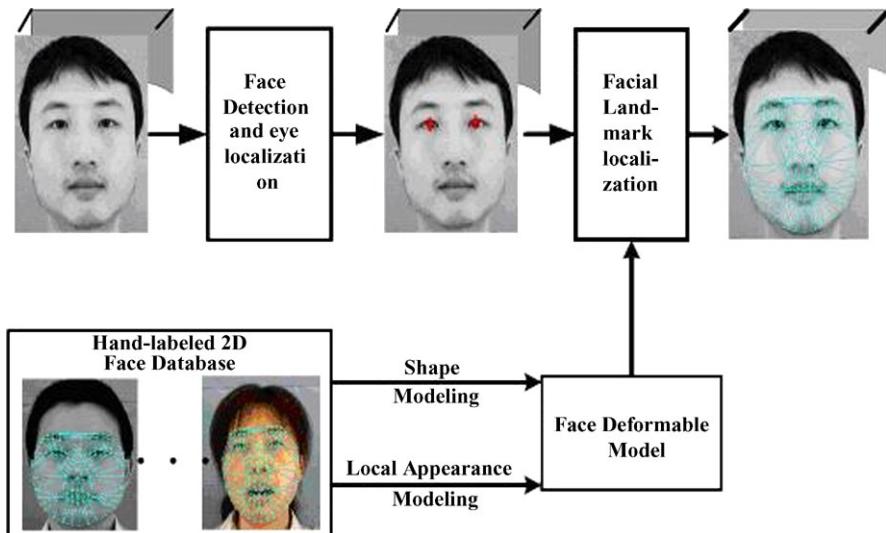
## 12.2 Framework for Landmark Localization

This facial landmark localization framework consists of training and locating procedures, as illustrated in Fig. 12.2.

The training procedure is building a face deformable model via shape modeling and local appearance modeling. This procedure needs a great amount of hand labeled data. The locating procedure consists of firstly the face detection, the eye localization and then the facial landmark localization based on the face deformable model. In the eye localization procedure, we will introduce a robust and precise eye localization method, and then adopt this method to precisely locate the eye position. The eye localization method is real-time. In the facial landmark localization procedure, a random forest embedded active shape model is adopted. In the following paragraphs, they will be presented and discussed in detail.

## 12.3 Eye Localization

The eye localization is a crucial step towards automatic face recognition and facial landmark localization due to the fact that these face related applications need to normalize faces, measure the relative positions or extract features according to eye positions. Like other problems of object detection under complex scene such as face detection, car detection, eye patterns also have large variation in appearance due to various factors, such as size, pose, rotation, the closure and opening of eyes,



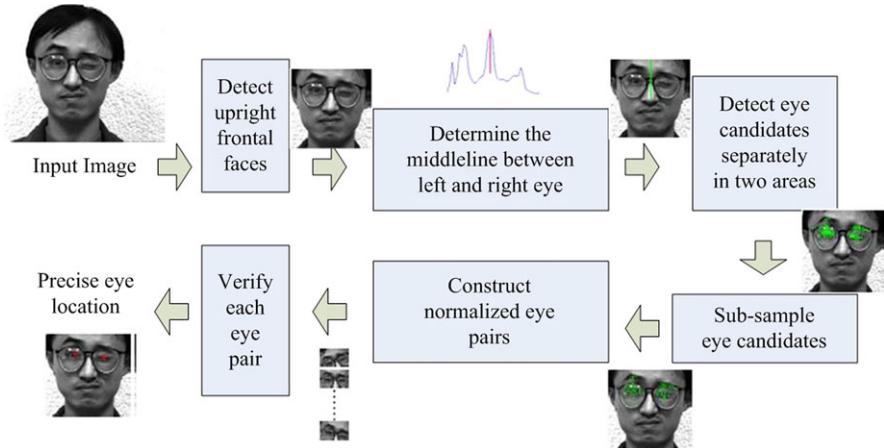
**Fig. 12.2** Facial landmark localization processing framework

illumination conditions, the reflection of glasses and the occlusion by hairs etc. Even having found the positions of faces grossly, robustly and precisely locating the eye's center is still a challenging task. A variety of eye detection and tracking algorithms have been proposed in recent years, but most of them can only deal with part of these variations or be feasible under some constraints. We have devised a novel approach for precisely locating eyes in face areas under a probabilistic framework. The experimental results demonstrate that our eye localization method can robustly cope with different eye variations and achieve higher detection rate on diverse test sets.

The block diagram of the proposed method is shown in Fig. 12.3. When a rough face region is presented to the system, mean projection function and variance projection function [14] are adopted for determining the midline between the left and right eye. Then in the two areas, the appearance-based eye detector is used to find eye candidates separately. All the eye candidates are subsampled according to their probabilities. The remaining left and right eye candidates are paired. All the possible eye pairs are classified by an appearance-based eye-pair classifier. The most probable eye pairs are taken as the locations of left and right eyes.

### 12.3.1 Midline of Eyes

For an upright frontal face, the vertical midline between left and right eye is near the bridge of nose. According to the observations that the change of gray intensity on eye area is more obvious than bridge of nose and the eye area is often darker than the bridge of nose, vertical mean and variance projection function [14] are



**Fig. 12.3** Flowchart of the precise eye localization method under probabilistic framework

used. Suppose  $I(x, y)$  is the intensity of a pixel at location  $(x, y)$ , the vertical mean projection function  $\text{MPF}_v(x)$  and vertical variance projection function  $\text{VPF}_v(x)$  of  $I(x, y)$  in intervals  $[y_1, y_2]$  can be defined respectively, as:

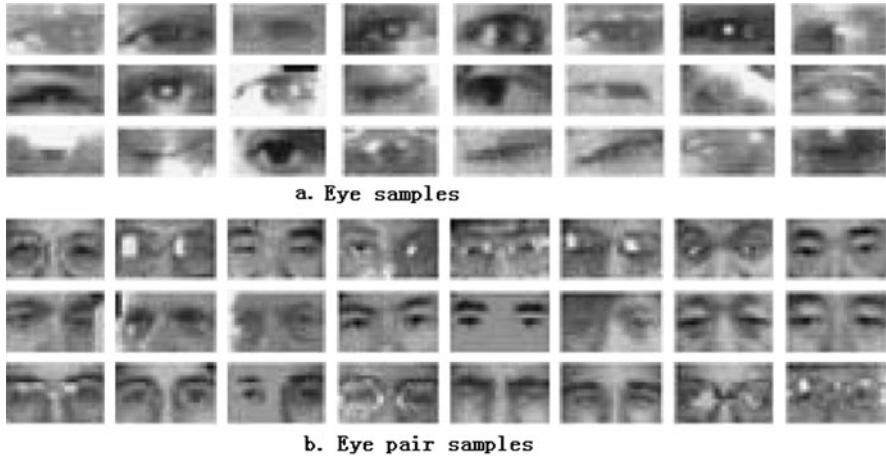
$$\text{MPF}_v(x) = \frac{1}{y_2 - y_1} \sum_{y=y_1}^{y_2} I(x, y) \quad (12.1)$$

$$\text{VPF}_v(x) = \sqrt{\frac{1}{y_2 - y_1} \sum_{y=y_1}^{y_2} [I(x, y) - \text{MPF}_v(x)]^2} \quad (12.2)$$

Applying the two functions to upper half of a face region, an obvious response around the bridge of nose will be obtained (Fig. 12.3b). So the position of vertical midline separating the left eye from right eye can be estimated. An appearance-based eye detector will be applied in the two areas separately.

### 12.3.2 Eye Candidate Detection

We used standard AdaBoost training methods combined with Viola and Jones's cascade approach to build appearance based eye detector. The cascade structure enables the detector to rule out most of the face areas as eye with a few tests and allows computational resources to be concentrated on the more challenging parts of the images. The features used in AdaBoost training process are Haar basis vectors [31] as elementary features. For an eye sample with size of  $24 \times 12$ , there are about 40 000 features in total. There are in total 6800 eye samples in the positive training set, some of which can be seen in Fig. 12.4. All the eye samples are cropped from faces



**Fig. 12.4** Positive training examples for AdaBoost

with the eye center being the center of the example. The negative examples are obtained by a bootstrap process [28]. All the samples are processed with gray scale normalization and size normalization to  $24 \times 12$  pixels. In this step, we avoid making premature decisions about the precise location of an eye. By contrast, we just exclude most background and give all the candidates with the probabilities at the expense of some false positives. The face regions most easily confused with eyes are eyebrows, thick frames of glasses, etc. In Fig. 12.3c, the center of every detected candidate is denoted by a dot in the face area.

### 12.3.3 Eye Candidate Subsampling

Because the local appearance of eyes is not nearly as distinctive as that of the whole face, some spurious eyes such as eyebrows, spectacle frames would be found, and true eyes would be located in different scales and near positions (in Fig. 12.3d). If all the candidates were considered in the next step, the processing time would be too long (e.g., for 40 left eye candidates and 40 right eye candidates, we have 1600 eye pairs in next step). To merge the candidates in a neighborhood, we subsample candidates with a factor of  $N$  in horizontal and vertical direction according to the probabilities (as shown in Fig. 12.3d).  $N$  is adjusted according to the face width. After the subsampling step, the number of eye pairs is reduced to 1/3 or less of the original amount.

### 12.3.4 Eye-Pair Classification

To exclude spurious and inaccurate eye candidates, we build an eye-pair classifier in a similar way constructing the eye detector described above. Each eye-pair sample includes the bounding rectangle around the left and right eyes with a small amount of space above and below the eyes, some of which can be seen in Fig. 12.4.

Negative eye-pair examples are collected also using bootstrap method. All the samples are normalized to  $25 \times 15$  pixels. In the test stage, for every pair of candidates in our list we figure out all possible pairings such that a priori information on inter ocular distances is satisfied. Then we use the affine warp to normalize the pair's region so that its left and right eye center positions line up with the left and right eye center positions of training data. The probability of the pairing constituting a true eye-pair is estimated. The average position of the 3 most probable eye-pairs' eye-center is considered as the precise position of the eye center of the face.

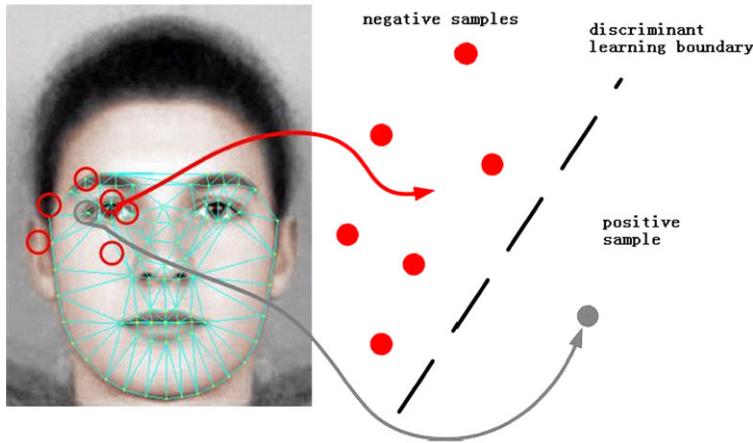
## 12.4 Random Forest Embedded ASM

With the prominent successful research of ASM and AAM, face shape is well modeled as a linear combination of principal modes (major eigenvectors) learned from the training face shapes. Here, we define shape as a series of coordinates of facial feature points. Facial landmark localization, thus, can be solved under the framework of ASM. Both the methods consist of three steps, shape modeling, distance measurement and global optimization. The facial landmark localization method is described as fitting the 2D face model to the novel face image. 2D face model is a deformable model based on random forest embedded key point recognition under the framework of ASM. We name our 2D face model as Random Forest Embedded Active Shape Model (RFE-ASM). The novelty is that this method embeds the discriminant learning into ASM. In our method, the 2D face model is represented by 88 landmarks; therefore, it can describe eyes, eyebrows, nose, mouth and cheek. Each landmark is accurately recognized by a fast classifier, which is trained from the appearance around this landmark. The proposed 2D face model embedding discriminant learning is illustrated in Fig. 12.5. Our facial landmark localization using RFE-ASM is presented in the following. Firstly, face shape is modeled and the fitting problem is defined as an optimization problem; then, distance between shape fitting results and the novel face image should be measured; finally, best fit should be optimized and facial landmark localization is performed by optimizing all the defined 88 facial feature points.

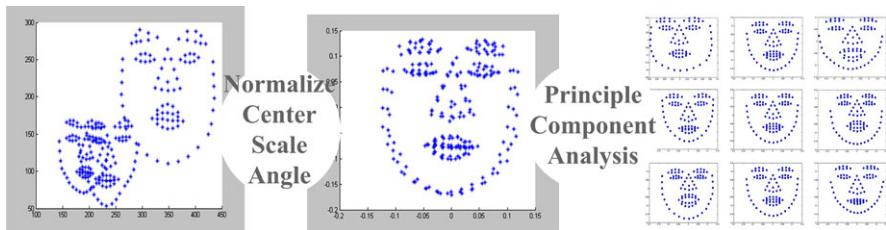
### 12.4.1 Shape Modeling

We define shape as a series of coordinates of facial feature points as:

$$X_i = [x_{i1}, y_{i1}, x_{i2}, y_{i2} \dots x_{ij}, y_{ij} \dots x_{i88}, y_{i88}]^T. \quad (12.3)$$



**Fig. 12.5** 2D face model embedding discriminant learning



**Fig. 12.6** 2D shape modeling

It is the sequence of hand-labeled 88 points in the image lattice. We manually label 88 points for each face image in the training set. The manually labeled face images are used to train the face model. With the trained model, we can automatically locate 88 facial feature points of the face images which are unseen in the training set. 2D face shape is firstly normalized (center, scale, angle) and then well modeled as a linear combination of principal modes (major eigenvectors) learned from the training face shapes as illustrated in Fig. 12.6.

Principal component analysis (PCA) is used to represent the normalized shape as a vector  $b$  in the low-dimensional shape eigenspace spanned by  $k$  principal modes (major eigenvectors) learned from the training shapes. A new shape  $X$  could be linearly obtained from shape eigenspace  $P$  with shape parameter vector  $b$ , and then transformed by center, scale and angle, presented by geometry parameter  $a$  as shown in:

$$X = T_a(\bar{X} + Pb), \quad (12.4)$$

$$a = (X_t, Y_t, s, \theta), \quad (12.5)$$

$$T_a \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} X_t \\ Y_t \end{pmatrix} + \begin{pmatrix} s \cos \theta & -s \sin \theta \\ s \sin \theta & s \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (12.6)$$

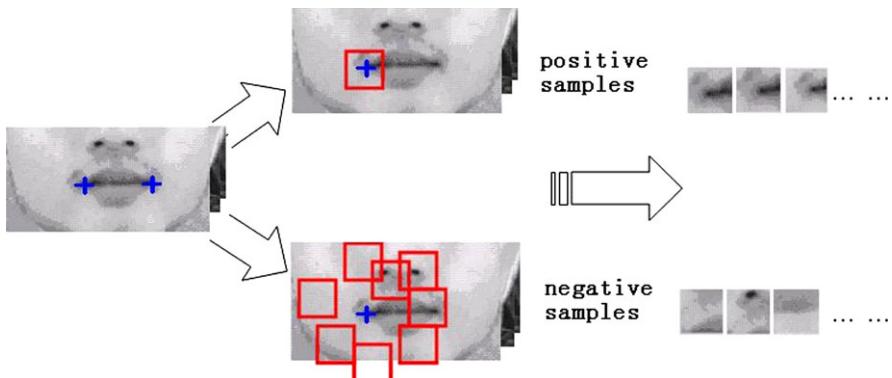
$s$  is scale factor,  $\theta$  is angle factor, and  $X_t$ ,  $Y_t$  are horizontal and vertical shift variables. 2D face modeling builds 2D face deformable model based on a large training set. Such 2D face model needs two parameters (geometry parameter  $a$  and shape parameter  $b$ ) to present a face shape. Facial landmark localization algorithm is thus defined as the method to find the best geometry parameter  $a$  and shape parameter  $b$  for a novel face image.

### 12.4.2 Distance Measurement

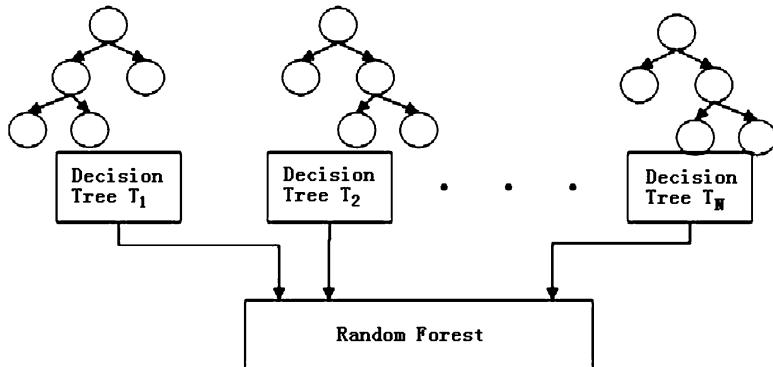
In conventional ASM, local image features around each landmark are modeled as the first derivatives of the sampled profiles perpendicular to the landmark contour. However, this approach ignores the difference between landmarks and their nearby backgrounds. This study proposes to add key point recognition into ASM by embedding discriminant learning as illustrated in Fig. 12.5. Each landmark is accurately recognized by a fast classifier, which is trained from the appearance around this landmark. Several classification algorithms, such as SVM or neural networks, could have been chosen. Among those, Lepetit and Fua [22] have found random forest to be eminently suitable because it is robust and fast, while remaining reasonably easy to train. The proposed method is under the framework of ASM with embedded random forest learning, so called RFE-ASM.

Random forest classifier is trained to recognize each landmark. The samples are collected on a large training set. All the samples are cropped from faces (the distance between the center of the left eye and the center of the right eye is normalized into 60 pixels). Positive samples are the  $32 \times 32$  image patches of all the training images with the center at the ground-truth landmark position. While negative samples are the  $32 \times 32$  image patches of all the training images with the center inside the  $40 \times 40$ , but outside the  $5 \times 5$  region from the ground-truth landmark position. As illustrated in Fig. 12.7, we take an example of the left mouth corner point. To find the left mouth corner point accurately, we train one random forest classifier for this landmark. All the samples are cropped from face images.

Random forest is a classifier combination method. A random forest consists of  $N$  binary trees. Each node of a binary tree is a weak classifier. The structure of random forest combines all the weak classifiers into a strong classifier. The output of random forest classifier is the voting of each binary tree. Figure 12.8 depicts a random forest. It consists of  $N$  decision trees. Each decision tree is trained by a completely random approach. For each decision tree,  $T_n$ , the samples are selected randomly from the training sample pool. It is a subset of all the training samples. After  $N$  trees are trained, the final decision combines all the outputs of  $T_1 T_2 \dots T_N$  by considering the average of all  $N$  outputs. Figure 12.9 depicts a generic tree. Each node contains a simple comparison of the intensity in a pair of points that split the space of image patches to be classified. The training step aims to get an estimate

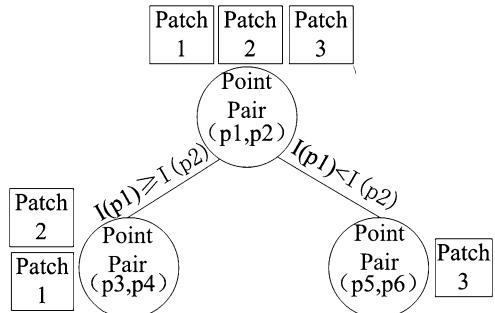


**Fig. 12.7** To train one random forest for left mouth corner point, this figure shows an example of positive and negative sample collection



**Fig. 12.8** Random forest combines the outputs of all decision trees as a classifier fusion method

**Fig. 12.9** A generic binary tree in random forest: each node contains a simple comparison of the intensity in a pair of points that split the space of image patches to be classified



based on training data of the posterior distribution over the classes in each leaf (the end node of a binary tree, which does not have child nodes).

In this training case, a random forest consists of multiple binary trees so that each tree yields a different partition of the space of image patches. Each node of a binary tree stores the best point pair, which is the weak classifier. The weak classifier is the comparison of intensity in a pair of points as in:

$$h = \begin{cases} 1 & \text{if } I(p_1) \geq I(p_2), \\ 0 & \text{otherwise.} \end{cases} \quad (12.7)$$

Each node chooses the best point pair as the best weak classifier and the random forest combines the results of each weak classifier to a strong classifier as

$$\hat{F}(p) = \arg \max_c p_c(p) \quad (12.8)$$

$$= \arg \max_c (1/N) \sum_{n=1 \dots N} p_{n,p}(f(p) = c) \quad (12.9)$$

where  $N$  is the total number of binary trees and  $n$  specifies one binary tree;  $p$  is the image patch to be classified;  $c$  is the label of class such that when  $c = 0$ , the image patch does not belong to the landmark and when  $c = 1$ , the image patch belongs to the landmark;  $p_{n,p}$  is the probability classified by the  $n$ th binary tree that the image patch  $p$  belongs to the landmark.

By dropping the image patch down the tree and performing a determined point pair comparison at each node, the image patch is sent to one side or the other (each node of a binary tree has two splits). When it reaches a leaf, it is assigned probabilities of belonging to a class depending on the distribution stored in the leaf. Responses of all the binary trees are combined during classification to achieve a better recognition rate than a single tree could. The distance of the novel face image and the 2D face model is thus measured by the output of random forest classifiers. The point that is more similar to the landmark will get a bigger random forest output probability.

### 12.4.3 Global Optimization

Facial landmark localization aims to find the best fit of the 88 points in the novel face image with the 2D face model. A new shape  $X$  could be obtained with geometry parameter  $a$  and shape parameter vector  $b$ . For each landmark, a random forest classifier gives the result measuring the distance of one point belonging to the landmark. 2D face shape consists of 88 facial feature points; therefore, all random forest results should be embedded into the global optimization. The global optimization objective function proposed is:

$$(\hat{a}, \hat{b}) = \arg \min_{a,b} \left( (Y - T_a(\bar{X} + Pb))^T W (Y - T_a(\bar{X} + Pb)) + k \sum_{j=1}^t b_j^2 / \sigma_j^2 \right) \quad (12.10)$$

where  $W$  is the output of random forest classifier and is embedded into the global optimization objective function to weigh 88 facial points. The shape parameter vector  $b$  is restricted to the vector space spanned by the training database.

The optimization includes the following steps:

1. Initialization:  $a$  is initialized according to face detection bounding box and two eyes positions. PCA shape parameter  $b$  is initialized to 0.
2. Finding new shape candidate:  $Y \leftarrow \hat{Y}$  Random Forest output in the nearby location of the last  $Y$ .
3.  $a \leftarrow \hat{a}$ .  $\hat{a} = \min_a((Y - T_a(\bar{X} + Pb))^T W(Y - T_a(\bar{X} + Pb)))$ .
4.  $b \leftarrow \hat{b}$ .  $\hat{b} = \min_b((Y - T_a(\bar{X} + Pb))^T W(Y - T_a(\bar{X} + Pb))) + k \cdot \sum_{j=1}^t b_j^2 / \sigma_j^2$ .
5. If  $\|\hat{a} - a\| + \|\hat{b} - b\| < \epsilon$ , stop. else, go to 2.

## 12.5 Experiments

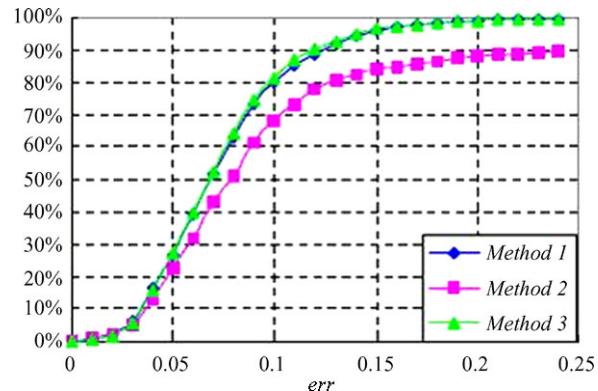
### 12.5.1 Eye Localization

The training set is drawn from FERET, ARData, Bern, BioID, ORL, OCRFace database, and a total of 6800 eyes and 18 000 eye-pairs are cropped and normalized for training. The experimental test set consists of Yale (15 persons, 165 images), AeroInfo Face (165 persons, 3740 images), Police Face (30 persons, 448 images), and a total of 4353 faces are involved in the evaluation of localization performance and the influence of eye locations on face recognition. In the training databases, FERET, ARData, BioID, Bern, ORL are open databases, and OCRFace, built by our lab, consists of 1448 face images with different views, expressions and glasses. Among the test databases, Yale is an open database, which features extremely unbalanced lightening and thick glasses; Police Face, provided by the First Research Institute of Ministry of Public Security of China, features strong glaring of glasses and large pose variation; AeroInfo, provided by the Aerospace Information Co. Ltd. of China, features a large variety of illumination, expression, pose, face size, and complex background. The three test sets are from diverse sources to cover different eye variations in view angles, sizes, illumination, and glasses. Experiments based on such diverse sets should be able to test the generalization performance of our eye localization algorithm.

To evaluate the precision of eye localization, a scale independent localization criterion [29] is used. This relative error measure compares the automatic localization result with the manually marked locations of each eye. Let  $C_l$  and  $C_r$  be the manually extracted left and right eye positions,  $C'_l$  and  $C'_r$  be the detected positions,  $d_l$  be the Euclidean distance between  $C'_l$  and  $C_l$ ,  $d_r$  be the Euclidean distance between  $C'_r$  and  $C_r$ ,  $d_{lr}$  be the Euclidean distance between the ground truth eye centers. Then the relative error of this detection is defined as follows:

$$\text{err} = \frac{\max(d_l, d_r)}{d_{lr}}. \quad (12.11)$$

**Fig. 12.10** Cumulative distribution of localization errors of three methods on test set

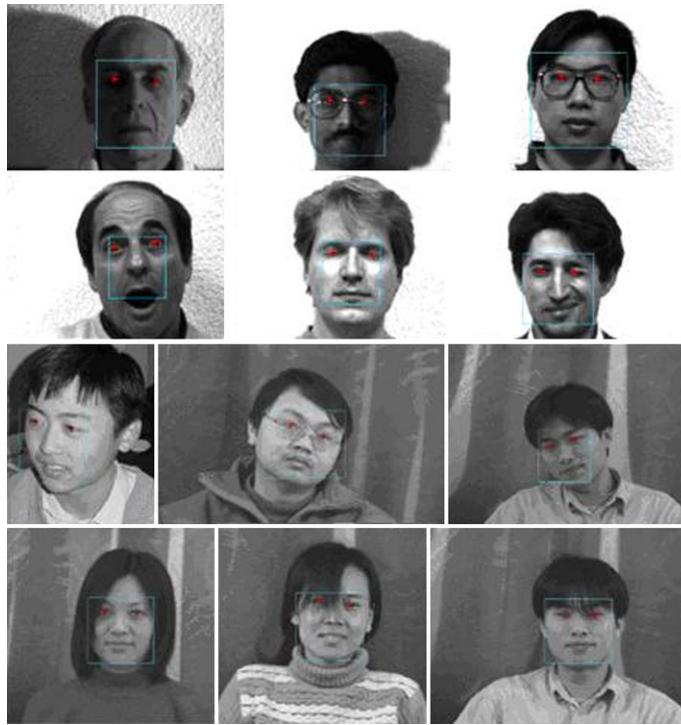


Three different eye localization methods are implemented and evaluated on the test set. Method 1: The proposed algorithm in this chapter. Method 2: Similar to the method proposed in [1]. After grossly locating the face area and determining the midline between left and right eye, connected components analysis and projection analysis are applied to the two areas separately to locate the eye center position. Method 3: Different from Method 1 only in that the step, subsampling eye candidates, is omitted.

The cumulative distribution function of localization error of three methods is shown in Fig. 12.10. From the figure, we can see that method 1 and method 3 achieve similar performance and about 99.1% of the test samples are with localization error below 0.20. Both are superior to method 2. But method 1 is 2–3 times faster than method 3. So the subsampling step does not degrade the location precision, but enhances the localization speed. The average processing time per face of method 1 on a PIV2.4 GHz PC system is 60 ms without special code optimization. In Fig. 12.11, we offer some examples out of the test sets for visual examination. The system appears to be robust to the presence of unbalanced illumination, eyeglasses, partial occlusion and even significant pose changes. This generalization ability is likely a consequence of the combination of local appearance and global appearance under probabilistic framework. Specially, in local appearance, the illumination influence can be effectively removed through local gray scale normalization; in global appearance, the influence caused by face rotation in image plane can be effectively removed through the aligning. We also compared method 1 with other newly published systems. In paper [35], the detection was considered to be correct if  $\text{err} < 0.25$ . Their detection performance on JAFFE database was 97.18%. We evaluate method 1 on JAFFE under the same test protocol. The detection rate of our method is 98.6% if  $\text{err} < 0.10$ , and the detection rate is 100% if  $\text{err} < 0.12$ .

### 12.5.2 Random Forest Embedded ASM

In order to verify our algorithm, experiments have been conducted on a large data set consisting of 3244 images from four databases for training. We collect and con-



**Fig. 12.11** Some eye localization results from test sets

struct the THFaceID database including 334 male and female aging from young to old with various facial expressions. The Yale database, FRGC database and JAFFE database are all publicly available. The Yale database includes illumination changes and facial expression changes; The FRGC database also includes facial expression and illumination changes under controlled and uncontrolled situations; The JAFFE database includes expression changes. All the 3244 images are manually labeled with 88 points as the ground truth landmarks. Test set 1 is constructed by the THFaceID database including 200 persons, totally 600 images. Test set 2 is IMM database. This method automatically detects faces and locates eye positions. The eye localization is used as the initialization for parameter optimization procedure. After initialization, the faces are aligned by the generic face deformable model trained before. The accuracy is measured by

$$e = \sum_{i=1}^{88} \|P_a - P_m\|_2 / (88 \cdot d_e). \quad (12.12)$$

We call it the relative error  $e$ , which is the point to point error between the face alignment results  $P_a$  and manually labeled ground-truth  $P_m$  when the distance of left and right eye  $d_e$  is normalized to 60 pixels.

**Table 12.1** Relative error of the algorithm

| Number of random trees | Relative error on our database | Relative error on IMM database |
|------------------------|--------------------------------|--------------------------------|
| 1                      | 6.79                           | 7.28                           |
| 5                      | 3.84                           | 4.41                           |
| 10                     | 3.76                           | 4.28                           |
| 30                     | 3.76                           | 4.23                           |
| 50                     | 3.77                           | 4.27                           |
| 80                     | 3.82                           | 4.30                           |
| 100                    | 3.82                           | 4.27                           |

**Table 12.2** Speed of the algorithm

| Number of random trees | Speed (ms) | Computer configuration           |
|------------------------|------------|----------------------------------|
| 1                      | 24         | Intel Core2, 2.66 GHz, 3.25G RAM |
| 5                      | 30         |                                  |
| 10                     | 37         |                                  |
| 30                     | 69         |                                  |
| 50                     | 102        |                                  |
| 80                     | 150        |                                  |
| 100                    | 184        |                                  |

**Fig. 12.12** Results of facial landmark localization towards video sequences

Table 12.1 shows the test results on Test set 1 and 2. Table 12.2 shows the speed of the algorithm. Figure 12.12 shows facial landmark localization algorithm towards video sequences.

**Table 12.3** Face recognition results in different facial landmark localization results

| Number of random trees | PCA dimension after reduction | Recognition rate (%) |
|------------------------|-------------------------------|----------------------|
| 10                     | 500                           | 63.7                 |
| 20                     | 500                           | 64.9                 |
| 30                     | 600                           | 65.1                 |
| 40                     | 500                           | 64.3                 |
| 50                     | 600                           | 63.7                 |
| 60                     | 500                           | 63.8                 |
| 80                     | 500                           | 64.0                 |
| 100                    | 600                           | 63.6                 |

In order to make sure what precision will make facial landmark localization meaningful in applications, the face recognition experiment on FRGC-V2 database is carried on. This experiment shows the relation of facial landmark localization's precision and face recognition rate. Gabor features are extracted at each facial landmark and put together as the whole feature vector. The training samples are 222 individuals from FRGC-V2 database, each individual has 10 images. The testing samples are 466 individuals from FRGC-V2 database, each individual has one image as face template. 466 individuals has totally 8014 images for testing. After feature extraction, PCA is used as the dimension reduction, LDA is used as the discriminant learning, and normalized correlation classifier is used. In addition, this experiment does not do illumination preprocessing. The face recognition results are listed in Table 12.3.

## 12.6 Conclusions

We have presented a facial landmark localization algorithm. Incorporating random forest classifier into ASM, this method works well when fitting to images which are unseen in the training set. Moreover, it runs in real time.

Face Recognition Vendor Test 2006 has shown that face recognition can achieve high accuracy under controlled conditions, for example, when the testing face samples are frontal. However, when face pose changes largely, the performance of existing methods drop drastically. The same difficulties are found in the literature of facial landmark localization. A reasonable way to improve multi-view facial landmark localization is to use 3D face morphable model. With the development of our further research, our studies will focus on fast and robust facial landmark localization algorithm by combining 2D deformable model with 3D morphable model.

**Acknowledgements** The author is indebted to the National Basic Research Program of China (973 program) under Grant No. 2007CB311004 for supporting this work, to Dr. Yong Ma for his works on face detection and eye localization, to Mr. Liu Ding who kindly helped do the face recognition experiment of this paper.

## References

1. Baskan, S., Atalay, V.: Projection based method for segmentation of human face and its evaluation. *Pattern Recognit. Lett.* **23**, 1623–1629 (2002)
2. Brunet, N., Perez, F., de la Torre, F.: Learning good features for active shape models. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 206–211. IEEE Computer Society Press, Los Alamitos (2009)
3. Cootes, T.F., Taylor, C.J.: Active shape model search using local grey-level models: A quantitative evaluation. In: 4th British Machine Vision Conference, pp. 639–648. BMVA Press, Guildford (1993)
4. Cootes, T.F., Taylor, C.J.: Combining elastic and statistical models of appearance variation. In: Proceeding of 6th European Conference on Computer Vision, vol. 1, pp. 149–163. Springer, Berlin (2000)
5. Cootes, T.F., Taylor, C.J.: Constrained active appearance models. In: Proceeding of 8th International Conference on Computer Vision, vol. 1, pp. 748–754. IEEE Computer Society Press, Los Alamitos (2001)
6. Cootes, T.F., Taylor, C.J.: An algorithm for tuning an active appearance model to new data. In: Proceeding of British Machine Vision Conference, vol. 3, pp. 919–928. BMVA Press, Guildford (2006)
7. Cootes, T.F., Taylor, C.J., Cooper, D., Graham, J.: Training models of shape from sets of examples. In: 3rd British Machine Vision Conference, pp. 9–18. BMVA Press, Guildford (1992)
8. Cootes, T.F., Taylor, C.J., Lanitis, A.: Active shape models: Evaluation of a multi-resolution method for improving image search. In: 5th British Machine Vision Conference, pp. 327–336. BMVA Press, Guildford (1994)
9. Cootes, T.F., Taylor, C., Cooper, D., Graham, J.: Active shape models—their training and their applications. *Comput. Vis. Image Underst.* **61**(1), 38–59 (1995)
10. Cootes, T.F., Walker, K.N., Taylor, C.J.: View-based active appearance models. In: Proceeding of 4th International Conference on Automatic Face and Gesture Recognition, pp. 227–232. IEEE Computer Society Press, Los Alamitos (2000)
11. Cootes, T.F., Wheeler, G., Walker, K., Taylor, C.J.: Coupled-view active appearance models. In: 11th British Machine Vision Conference, pp. 52–61. BMVA Press, Guildford (2000)
12. Cootes, T.F., Edwards, G., Taylor, C.: Active appearance models. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(6), 681–685 (2001)
13. Cootes, T.F., Twining, C.J., Petrovic, V., Schedestowitz, R., Taylor, C.J.: Groupwise construction of appearance models using piece-wise affine deformations. In: Proceeding of British Machine Vision Conference, vol. 2, pp. 879–888. BMVA Press, Guildford (2005)
14. Feng, G.C., Yuen, P.C.: Multi-cues eye detection on gray intensity image. *Pattern Recognit.* **34**, 1033–1046 (2001)
15. Gross, R., Matthews, I., Baker, S.: Generic vs. person specific active appearance models. *Image Vis. Comput.* **23**(1), 1080–1093 (2005)
16. Hamsici, O., Martinez, A.: Active appearance models with rotation invariant kernels. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1003–1009. IEEE Computer Society Press, Los Alamitos (2009)
17. Hess, M., Martinez, G.: Facial feature extraction based on the smallest univalue segment assimilating nucleus (susan) algorithm. In: Proceedings of Picture Coding Symposium, vol. 1, pp. 261–266. IEEE Computer Society Press, Los Alamitos (2004)
18. Hsu, R.L., Jain, A.K.: Generating discriminating cartoon faces using interacting snakes. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(11), 1388–1398 (2003)
19. Huang, C., Ai, H.Z., Li, Y., Lao, S.H.: High performance rotation invariant multiview face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(4), 671–686 (2007)
20. Jiao, F., Li, S., Shum, H., Schuurmans, D.: Face alignment using statistical models and wavelet features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. I:321–327. IEEE Computer Society Press, Los Alamitos (2003)

21. Lee, H.-S., Kim, D.: Tensor-based aam with continuous variation estimation: Application to variation-robust face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**, 1102–1116 (2009)
22. Lepetit, V., Fua, P.: Keypoint recognition using randomized trees. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1465–1479 (2006)
23. Li, Y., Ito, W.: Shape parameter optimization for adaboosted active shape model. In: *IEEE International Conference on Computer Vision*, vol. 1, pp. 251–258 (2005)
24. Liebelt, J., Xiao, J., Yang, J.: Robust aam fitting by fusion of images and disparity data. In: *Proceedings of IEEE Computer Vision and Pattern Recognition*, vol. II, pp. 2483–2490. IEEE Computer Society Press, Los Alamitos (2006)
25. Liu, X.: Generic face alignment using boosted appearance model. In: *Proceedings of IEEE Computer Vision and Pattern Recognition*, p. 1079 (2007)
26. Liu, X.: Discriminative face alignment. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**, 1941–1954 (2009)
27. Papandreou, G., Maragos, P.: Adaptive and constrained algorithms for inverse compositional active appearance model fitting. In: *Proceedings of Computer Vision and Pattern Recognition*, p. 1 (2008)
28. Sung, K.K., Poggio, T.: Example based learning for view-based human face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**, 39–51 (1995)
29. Tu, Z., Chen, X., Yuille, A.L., Zhu, S.-C.: Image parsing: Unifying segmentation, detection, and recognition. In: *Proceeding of International Conference on Computer Vision*, vol. 1, p. 18. IEEE Computer Society Press, Los Alamitos (2003)
30. Tu, J., Zhang, Z., Zeng, Z., Huang, T.: Face localization via hierarchical condensation with fisher boosting feature selection. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. II, pp. 719–724. IEEE Computer Society Press, Los Alamitos (2004)
31. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple feature. In: *Proceedings of Computer Vision and Pattern Recognition*, vol. 1, pp. 511–518. IEEE Computer Society Press, Los Alamitos (2001)
32. Vogler, C., Li, Z., Kanaujia, A., Goldenstein, S., Metaxas, D.: The best of both worlds: Combining 3d deformable models with active shape models. In: *IEEE International Conference on Computer Vision*, pp. 1–7 (2007)
33. Xiao, J., Baker, S., Matthews, I., Kanade, T.: Real-time combined 2d+3d active appearance models. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 535–542. IEEE Computer Society Press, Los Alamitos (2004)
34. Zhang, L., Ai, H.: Multi-view active shape model with robust parameter estimation. In: *International Conference on Pattern Recognition*, vol. 4, pp. 469–468 (2006)
35. Zhou, Z.H., Geng, X.: Projection functions for eye detection. *Pattern Recognit.* **37**(5), 1049–1056 (2004)

# Chapter 13

## Face Tracking and Recognition in Video

Rama Chellappa, Ming Du, Pavan Turaga, and Shaohua Kevin Zhou

### 13.1 Introduction

Faces are expressive three dimensional objects. Information useful for recognition tasks can be found both in the geometry and texture of the face and also facial motion. While geometry and texture together determine the ‘appearance’ of the face, motion encodes behavioral cues such as idiosyncratic head movements and gestures which can potentially aid in recognition tasks. Traditional face recognition systems have relied on a gallery of still images for learning and a probe of still images for recognition. While the advantage of using motion information in face videos has been widely recognized, computational models for video based face recognition have only recently gained attention.

In this chapter, we consider applications where one is presented with a video sequence—either in a single camera setting or a multi-camera setting—and the goal is to recognize the person in the video. The gallery could consist of either still-images or could be videos themselves.

Video is a rich source of information in that it can lead to potentially better representations by offering more views of the face. Further, the role of facial motion for face perception has been well documented. Psychophysical studies [26] have

---

R. Chellappa (✉) · M. Du · P. Turaga

Department of Electrical and Computer Engineering, Center for Automation Research, University of Maryland, College Park, MD 20742, USA

e-mail: [rama@umiacs.umd.edu](mailto:rama@umiacs.umd.edu)

M. Du

e-mail: [mingdu@umiacs.umd.edu](mailto:mingdu@umiacs.umd.edu)

P. Turaga

e-mail: [pturaga@umiacs.umd.edu](mailto:pturaga@umiacs.umd.edu)

S.K. Zhou

Siemens Corporate Research, 755 College Road East, Princeton, NJ 08540, USA

e-mail: [kzhou@scr.siemens.com](mailto:kzhou@scr.siemens.com)

found evidence that when both structure and dynamics information is available, humans tend to rely more on dynamics under nonoptimal viewing conditions (such as low spatial resolution, harsh illumination conditions etc.). Dynamics also aids in recognition of familiar faces [31]. If one were to ignore temporal dependencies, a video sequence can be considered as a collection of still images; so still-image-based recognition algorithms can always be applied. The properties of video sequences that can be exploited are (1) temporal correlations, (2) idiosyncratic dynamic information, and (3) availability of multiple views. Video thus proves useful in various tasks—it can be used to generate better appearance models, mitigate effects of non-cooperative viewing conditions, localize a face using motion, model facial behavior for improved recognition, generate better models of face shape from multiple views, etc.

The rest of the chapter is organized as follows. In Sect. 13.2, we describe the utility of videos in enhancing performance of image-based recognition tasks. In Sect. 13.3, we discuss a joint tracking-recognition framework that allows for using the motion information in a video to better localize and identify the person in the video using still galleries. In Sect. 13.4, we discuss how to jointly capture facial appearance and dynamics to obtain a parametric representation for video-to-video recognition. In Sect. 13.5, we discuss recognition in multi-camera networks where the probe and gallery both consist of multi-camera videos. Finally in Sect. 13.6, we present concluding remarks and directions for future research.

## 13.2 Utility of Video

**Frame-Based Fusion** An immediate possible utilization of temporal information for video-based face recognition is to fuse the results obtained by a 2D face recognition algorithm on each frame of the sequence. The video sequence can be seen as an unordered set of images to be used for both training and testing phases. During testing one can use the sequence as a set of probes, each of them providing a decision regarding the identity of the person. Appropriate fusion techniques can then be applied to provide the final identity. Perhaps the most frequently used fusion strategy in this case is majority voting [24, 34].

In [28], Park et al. adopt three matchers for frame-level face recognition: Face-VACS, PCA and correlation. They use the sum rule (with min-max normalization) to fuse results obtained from the three matchers and the maximum rule to fuse results of individual frames. In [21], the concept of identity surface is proposed to represent the hyper-surface formed by projecting face patterns of an individual to the feature vector space parameterized with respect to pose. This surface is learned from gallery videos. In testing stage, model trajectories are synthesized on the identity surfaces of enrolled subjects after the pose parameters of probe video have been estimated. Every point on the trajectory corresponds to a frame of the video and trajectory distance is defined as a weighted sum of point-wise distances. The model trajectory that yields minimum distance to the probe video's trajectory gives the final identification result. Based on the result that images live approximately in a bilinear

space of motion and illumination variables, Xu et al. estimate these parameters for each frame of a probe video sequence with a registered 3D generic face model [38]. They then replace the generic model with a person-specific model of each subject in the gallery to synthesize video sequences with the estimated illumination and motion parameters. Frame-wise comparison is conducted between the synthesized videos and the probe video. A synthesized video is considered as a winner if one of its frames yield the smallest distance across all frames and all the subjects in the gallery.

**Ensemble Matching** Without recourse to modeling temporal dynamics, one can consider a video as an ensemble of images. Recent methods have focused on utilizing image-ensembles for object and face recognition [4, 15, 17, 41]. For example, it was shown by Jacobs et al. that the illumination cone of a convex Lambertian surface can be approximated by a 9-dimensional linear subspace [5]. Motivated by this, the set of face images of the same person under varying illumination conditions is frequently modeled as a linear subspace of 9-dimensions [19]. In such applications, an object ‘category’ consists of image-sets of several ‘instances’. A common approach in such applications is to approximate the image-space of a single face/object under these variations as a linear subspace [14, 15]. A simplistic model for object appearance variations is then a mixture of subspaces. In [41], Zhou and Chellappa study the problem of measuring similarity between two ensembles by projecting the data into a Reproducing Kernel Hilbert Space (RKHS). The ensemble distance is then characterized as the probabilistic distance (Chernoff distance, Bhattacharyya distance, Kullback–Leibler (KL) divergence etc.) in RKHS.

**Appearance Modeling** Most face recognition approaches rely on a model of appearance for each individual subject. The simplest appearance model is a static image of the person. Such appearance models are rather limited in utility in video-based face recognition tasks where subjects may be imaged under varying viewpoints, illuminations, expressions etc. Thus, instead of using a static image as an appearance model, a sufficiently long video which encompasses several variations in facial appearance can lend itself to building more robust appearance models. Several methods have been proposed for extracting more descriptive appearance models from videos. For example, a facial video is considered as a sequence of images sampled from an ‘appearance manifold’ in [20]. In principle, the appearance manifold of a subject contains all possible appearances of the subject. In practice, the appearance manifold for each person is estimated from training data of videos. For ease of estimation, the appearance manifold is considered to be a collection of affine subspaces, where each subspace encodes a set of similar appearances of the subject. Temporal variations of appearances in a given video sequence are then modeled as transitions between the appearance subspaces. This method is robust to large appearance changes if sufficient 3D view variations and illumination variations are available in the training set. Further, the tracking problem can be integrated into this framework by searching for a bounding-box on the test image that minimizes the distance of the cropped region to the learnt appearance manifold.

In a related work, [3] represents the appearance variations due to shape and illumination on human faces, using the assumption that the ‘shape-illumination manifold’ of all possible illuminations and head poses is generic for human faces. This means that the shape-illumination manifold can be estimated using a set of subjects exclusive of the test set. They show that the effects of face shape and illumination can be learnt using Probabilistic PCA from a small, unlabeled set of video sequences of faces in randomly varying lighting conditions. Given a novel sequence, the learnt model is used to decompose the face appearance manifold into albedo and shape-illumination manifolds, producing the classification decision using robust likelihood estimation.

### 13.3 Still Gallery vs. Video Probes

Following Phillips et al. [29], we define a still-to-video scenario as follows. The gallery consists of still facial templates, and the probe set consists of video sequences containing the facial region. Though significant research has been conducted on still-to-still recognition, research efforts on still-to-video recognition are relatively fewer owing to the following challenges [40] in typical surveillance applications: poor video quality, significant illumination and pose variations, and low image resolution. Most existing video-based recognition systems [9, 40] attempt the following: The face is first detected and then tracked over time. Only when a frame satisfying certain criteria (size, pose) is acquired, recognition is performed using still-to-still recognition technique. For this, the face part is cropped from the frame and transformed or registered using appropriate transformations. This tracking-then-recognition approach attempts to resolve uncertainties in tracking and recognition sequentially and separately and requires a criterion for selecting good frames and estimation of parameters for registration. Also, still-to-still recognition does not effectively exploit temporal information.

We will assume that a certain feature representation for spatio-temporal patterns of moving faces has been made. We will also assume that there exists a set of hidden parameters, constituting the state vector, which govern how the spatio-temporal patterns evolve in time. The state vector encodes information such as motion parameters which can be used for tracking and identity parameters that can be used for recognition. Given a set of features, we need inference algorithms for estimating these hidden parameters. The three basic components of the model are the following.

- A motion equation governing the kinematic behavior of the tracking motion vector
- An identity equation governing the temporal evolution of the identity variable
- An observation equation establishing a link between the motion vector and the identity variable

We denote the gallery as  $\mathcal{I} = \{I_1, I_2, \dots, I_N\}$ , indexed by the identity variable  $n$ , which lies in a finite sample space  $\mathcal{N} = \{1, 2, \dots, N\}$ . And we denote the identity,

motion vector, and the observation at time  $t$  as  $n_t$ ,  $\theta_t$  and  $z_t$ , respectively. Using the Sequential Importance Sampling (SIS) [12, 18, 22] technique, the joint posterior distribution of the motion vector and the identity variable [i.e.,  $p(n_t, \theta_t | z_{0:t})$ ] is estimated at each time instant and then propagated to the next time instant governed by motion and identity equations. The marginal distribution of the identity variable [i.e.,  $p(n_t | z_{0:t})$ ] is estimated to provide the recognition result.

The recognition model consists of the following components.

- *Motion equation*

In its most general form, the motion model can be written as

$$\theta_t = g(\theta_{t-1}, u_t); \quad t \geq 1 \quad (13.1)$$

where  $u_t$  is noise in the motion model, whose distribution determines the motion state transition probability  $p(\theta_t | \theta_{t-1})$ . The function  $g(., .)$  characterizes the evolving motion, and it could be a function learned offline or given a priori. One of the simplest choices is an additive function (i.e.,  $\theta_t = \theta_{t-1} + u_t$ ), which leads to a first-order Markov chain.

The choice of  $\theta_t$  is dependent on the application. Affine motion parameters are often used when there is no significant pose variation available in the video sequence. However, if a three-dimensional (3D) face model is used, 3D motion parameters should be used accordingly.

- *Identity equation*

Assuming that the identity does not change as time proceeds, we have

$$n_t = n_{t-1}; \quad t \geq 1. \quad (13.2)$$

In practice, one may assume a small transition probability between identity variables to increase the robustness.

- *Observation equation*

By assuming that the transformed observation is a noise-corrupted version of some still template in the gallery, the observation equation can be written as

$$\mathcal{T}_{\theta_t}\{z_t\} = I_{n_t} + v_t; \quad t \geq 1 \quad (13.3)$$

where  $v_t$  is observation noise at time  $t$ , whose distribution determines the observation likelihood  $p(z_t | n_t, \theta_t)$ , and  $\mathcal{T}_{\theta_t}\{z_t\}$  is a transformed version of the observation  $z_t$ . This transformation could be geometric, photometric, or both. However, when confronting difficult scenarios, one should use a more sophisticated likelihood function as discussed in [43].

- *Statistical independence*

We assume statistical independence between all noise variables  $u_t$  and  $v_t$ .

- *Prior distribution*

The prior distribution  $p(n_0 | z_0)$  is assumed to be uniform.

$$p(n_0 | z_0) = \frac{1}{N}; \quad n_0 = 1, 2, \dots, N. \quad (13.4)$$

In our experiments,  $p(\theta_0|z_0)$  is assumed to be Gaussian: its mean comes from an initial detector or manual input and its covariance matrix is manually specified.

Using an overall state vector  $x_t = (n_t, \theta_t)$ , (13.1) and (13.2) can be combined into one state equation (in a normal sense) that is completely described by the overall state transition probability

$$p(x_t | x_{t-1}) = p(n_t | n_{t-1})p(\theta_t | \theta_{t-1}). \quad (13.5)$$

Given this model, our goal is to compute the posterior probability  $p(n_t | z_{0:t})$ . It is in fact a probability mass function (PMF), as  $n_t$  only takes values from  $\mathcal{N} = \{1, 2, \dots, N\}$ , as well as a marginal probability of  $p(n_t, \theta_t | z_{0:t})$ , which is a mixed distribution. Therefore, the problem is reduced to computing the posterior probability.

### 13.3.1 Posterior Probability of Identity Variable

The evolution of the posterior probability  $p(n_t | z_{0:t})$  as time proceeds is interesting to study, as the identity variable does not change by assumption [i.e.,  $p(n_t | n_{t-1}) = \delta(n_t - n_{t-1})$ , where  $\delta(\cdot)$  is a discrete impulse function at zero, that is,  $\delta(x) = 1$  if  $x = 0$ ; otherwise  $\delta(x) = 0$ ]. Using time recursion, Markov properties, and statistical independence embedded in the model, one can derive the following expressions:

$$\begin{aligned} & p(n_{0:t}, \theta_{0:t} | z_{0:t}) \\ &= p(n_{0:t-1}, \theta_{0:t-1} | z_{0:t-1}) \frac{p(z_t | n_t, \theta_t)p(n_t | n_{t-1})p(\theta_t | \theta_{t-1})}{p(z_t | z_{0:t-1})} \\ &= p(n_0, \theta_0 | z_0) \prod_{i=1}^t \frac{p(z_i | n_i, \theta_i)p(n_i | n_{i-1})p(\theta_i | \theta_{i-1})}{p(z_i | z_{0:i-1})} \\ &= p(n_0 | z_0)p(\theta_0 | z_0) \prod_{i=1}^t \frac{p(z_i | n_i, \theta_i)\delta(n_i - n_{i-1})p(\theta_i | \theta_{i-1})}{p(z_i | z_{0:i-1})}. \end{aligned} \quad (13.6)$$

Therefore, by marginalizing over  $\theta_{0:t}$  and  $n_{0:t-1}$ , we obtain the marginal posterior distribution for the identity  $j$ .

$$\begin{aligned} p(n_t = j | z_{0:t}) &= p(n_0 = j | z_0) \int_{\theta_0} \cdots \int_{\theta_t} p(\theta_0 | z_0) \\ &\quad \times \prod_{i=1}^t \frac{p(z_i | j, \theta_i)p(\theta_i | \theta_{i-1})}{p(z_i | z_{0:i-1})} d\theta_t \cdots d\theta_0. \end{aligned} \quad (13.7)$$

Thus,  $p(n_t = j | z_{0:t})$  is determined by the prior distribution  $p(n_0 = j | z_0)$  and the product of the likelihood functions  $\prod_{i=1}^t p(z_i | j, \theta_i)$ . If a uniform prior is assumed, then  $\prod_{i=1}^t p(z_i | j, \theta_i)$  is the only determining factor.

### 13.3.2 Sequential Importance Sampling Algorithm

Consider a general time series state space model fully determined by (1) the overall state transition probability  $p(x_t | x_{t-1})$ ; (2) the observation likelihood  $p(z_t | x_t)$ ; and (3) prior probability  $p(x_0)$  and statistical independence among all noise variables. We wish to compute the posterior probability  $p(x_t | z_{0:t})$ .

If the model is linear with Gaussian noise, it is analytically solvable by a Kalman filter, which essentially propagates the mean and variance of a Gaussian distribution over time. For nonlinear and non-Gaussian cases, an extended Kalman filter and its variants have been used to arrive at an approximate analytic solution [2]. Recently, the SIS technique, a special case of the Monte Carlo method [12, 18, 22] has been used to provide a numerical solution and propagate an arbitrary distribution over time.

The essence of the Monte Carlo method is to represent an arbitrary probability distribution  $\pi(x)$  closely by a set of discrete samples. It is ideal to draw i.i.d. samples  $\{x^{(m)}\}_{m=1}^M$  from  $\pi(x)$ . However, it is often difficult to implement, especially for nontrivial distributions. Instead, a set of samples  $\{x^{(m)}\}_{m=1}^M$  is drawn from an importance function  $g(x)$ ; then a weight

$$w^{(m)} = \pi(x^{(m)})/g(x^{(m)}) \quad (13.8)$$

is assigned to each sample. This technique is called importance sampling. It can be shown [22] that the importance sample set  $\mathcal{S} = \{(x^{(m)}, w^{(m)})\}_{m=1}^M$  is properly weighted to the target distribution  $\pi(x)$ . To accommodate a video, importance sampling is used in a sequential fashion, which leads to SIS. SIS propagates  $\mathcal{S}_{t-1}$  according to the sequential importance function, say  $g(x_t | x_{t-1})$ , and calculates the weight using

$$w_t = w_{t-1} p(z_t | x_t) p(x_t | x_{t-1}) / g(x_t | x_{t-1}). \quad (13.9)$$

In the CONDENSATION algorithm,  $g(x_t | x_{t-1})$  is taken to be  $p(x_t | x_{t-1})$  and (13.9) becomes

$$w_t = w_{t-1} p(z_t | x_t). \quad (13.10)$$

In fact, (13.10) is implemented by first resampling the sample set  $\mathcal{S}_{t-1}$  according to  $w_{t-1}$  and then updating the weight  $w_t$  using  $p(z_t | x_t)$ . For a complete description of the SIS method, refer to Doucet et al. [12] and Liu and Chen [22].

In the context of video-based face recognition, the posterior probability  $p(n_t, \theta_t | z_{0:t})$  is represented by a set of indexed and weighted samples

$$\mathcal{S}_t = \{(n_t^{(m)}, \theta_t^{(m)}, w_t^{(m)})\}_{m=1}^M \quad (13.11)$$

with  $n_t$  as the above index. We can sum the weights of the samples belonging to the same index  $n_t$  to obtain a proper sample set  $\{n_t, \beta_{n_t}\}_{n_t=1}^N$  with respect to the posterior PMF  $p(n_t | z_{0:t})$ . Straightforward implementation of the CONDENSATION algorithm for simultaneous tracking and recognition is not efficient in terms of its computational load. We refer the reader to [42] for a more detailed treatment of this issue.

### 13.3.3 Experimental Results

In this section, we describe the still-to-video scenarios used in our experiments and model choices, followed by a discussion of results. Two databases are used in the still-to-video experiments.

Database-0 was collected outside a building. We mounted a video camera on a tripod and requested subjects to walk straight toward the camera to simulate typical scenarios for visual surveillance. Database-0 includes one face gallery and one probe set. The probe contains 12 videos, one for each individual.

In Database-1, we have video sequences with subjects walking in a slant path toward the camera. There are 30 subjects, each having one face template. The face gallery is shown in Fig. 13.1. The probe contains 30 video sequences, one for each subject. Figure 13.1 shows some frames extracted from one probe video. As far as imaging conditions are concerned, the gallery is quite different from the probe, especially in terms of lighting. This is similar to the “FC” test protocol of the FERET test [29]. These images/videos were collected as part of the HumanID project by the National Institute of Standards and Technology and University of South Florida researchers.

#### 13.3.3.1 Results for Database-0

We now consider affine transformation. Specifically, the motion is characterized by  $\theta = (a_1, a_2, a_3, a_4, t_x, t_y)$ , where  $\{a_1, a_2, a_3, a_4\}$  are deformation parameters and  $\{t_x, t_y\}$  are 2D translation parameters. It is a reasonable approximation because there is no significant out-of-plane motion as the subjects walk toward the camera. Regarding the photometric transformation, only the zero-mean-unit-variance operation is performed to compensate partially for contrast variations. The complete transformation  $\mathcal{T}_\theta\{z\}$  is processed as follows. Affine transform  $z$  using  $\{a_1, a_2, a_3, a_4\}$ , crop out the interested region at position  $\{t_x, t_y\}$  with the same size as the still template in the gallery, and perform the zero-mean-unit-variance operation.

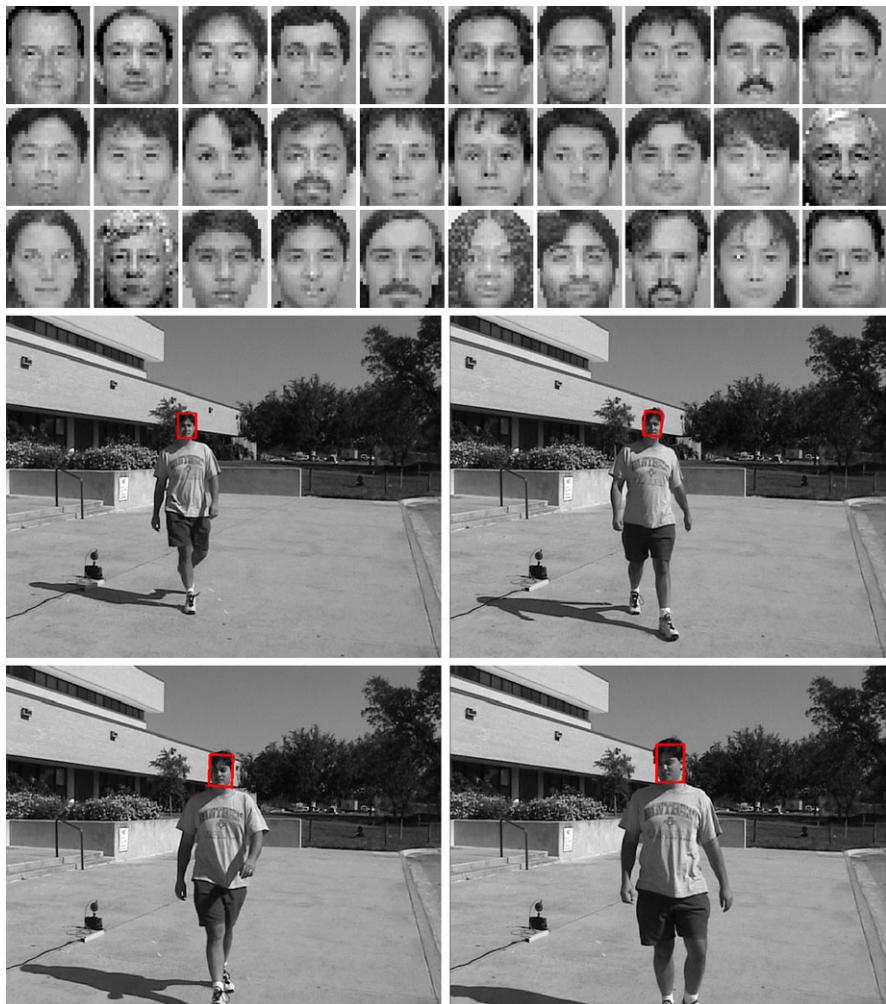
A time-invariant first-order Markov Gaussian model with constant velocity is used for modeling motion transition. Given that the subject is walking toward the camera, the scale increases with time. However, under perspective projection, this increase is no longer linear, causing the constant-velocity model to be not optimal. However, experimental results show that so long as the samples of  $\theta$  can cover the motion, this model is sufficient.

The likelihood measurement is simply set as a “truncated” Laplacian:

$$p_1(z_t | n_t, \theta_t) = L(\|\mathcal{T}_{\theta_t}\{z_t\} - I_{n_t}\|; \sigma_1, \tau_1) \quad (13.12)$$

where  $\|\cdot\|$  is sum of absolute distance,  $\sigma_1$  and  $\lambda_1$  are manually specified, and

$$L(x; \sigma, \tau) = \begin{cases} \sigma^{-1} \exp(-x/\sigma) & \text{if } x \leq \tau\sigma, \\ \sigma^{-1} \exp(-\tau) & \text{otherwise.} \end{cases} \quad (13.13)$$



**Fig. 13.1** Database-1. *First row:* the face gallery with image size of  $30 \times 26$ . *Second and third rows:* four frames in one probe video with image size of  $720 \times 480$ ; the actual face size ranged from approximately  $20 \times 20$  in the first frame to  $60 \times 60$  in the last frame. Note the significant illumination variations between the probe and the gallery

Gaussian distribution is widely used as a noise model, accounting for sensor noise and digitization noise among others. However, given the observation equation:  $v_t = \mathcal{P}_{\theta_t}\{z_t\} - I_{n_t}$ , the dominant part of  $v_t$  becomes the high-frequency residual if  $\theta_t$  is not proper; and it is well known that the high-frequency residual of natural images is more Laplacian-like. The “truncated” Laplacian is used to give a “surviving” chance for samples to accommodate abrupt motion changes.

Table 13.1 summarizes the average recognition performance and computational time of the CONDENSATION and the proposed algorithm when applied to

**Table 13.1** Recognition performance of algorithms when applied to Database-0

| Algorithm                             | CONDENSATION | Proposed    |
|---------------------------------------|--------------|-------------|
| Recognition rate within top one match | 100%         | 100%        |
| Time per frame                        | 7 seconds    | 0.5 seconds |

**Table 13.2** Performances of algorithms when applied to Database-1

| Case                             | Case 1 | Case 2 | Case 3 | Case 4 | Case 5 |
|----------------------------------|--------|--------|--------|--------|--------|
| Tracking accuracy                | 83%    | 87%    | 93%    | 100%   | NA     |
| Recognition within top 1 match   | 13%    | NA     | 83%    | 93%    | 57%    |
| Recognition within top 3 matches | 43%    | NA     | 97%    | 100%   | 83%    |

Database-0. Both algorithms achieved 100% recognition rate with top match. However, the proposed algorithm is more than 10 times faster than the CONDENSATION algorithm.

### 13.3.3.2 Results on Database-1

**Case 1: Tracking and Recognition Using Laplacian Density** We first investigate the performance using the same setting as described in Sect. 13.3.3.1. Table 13.2 shows that the recognition rate is poor: only 13% are correctly identified using the top match. The main reason is that the “truncated” Laplacian density is not able to capture the appearance difference between the probe and the gallery, indicating a need for more effective appearance modeling. Nevertheless, the tracking accuracy is reasonable, with 83% successfully tracked because we are using multiple face templates in the gallery to track the specific face in the probe video. After all, faces in both the gallery and the probe belong to the same class of human face, and it seems that the appearance change is within the class range.

**Case 2: Pure Tracking Using Laplacian Density** In Case 2, we measure the appearance change within the probe video as well as the noise in the background. To this end, we introduce a dummy template  $T_0$ , a cut version in the first frame of the video. Define the observation likelihood for tracking as

$$q(z_t | \theta_t) = L(\| \mathcal{T}_{\theta_t}\{z_t\} - T_0 \|; \sigma_2, \tau_2) \quad (13.14)$$

where  $\sigma_2$  and  $\tau_2$  are set manually. The other setting, such as motion parameter and model, is the same as in Case 1. We still can run the CONDENSATION algorithm to perform pure tracking. Table 13.2 shows that 87% are successfully tracked by this simple tracking model, which implies that the appearance within the video remains similar.

**Case 3: Tracking and Recognition Using Probabilistic Subspace Density** As mentioned in Case 1, we need a new appearance model to improve the recognition accuracy. Of the many approaches suggested in the literature, we decided to use the approach suggested by Moghaddam et al. [25] because of its computational efficiency and high recognition accuracy. However, here we model only the intrapersonal variations.

We need at least two facial images for one identity to construct the intrapersonal space (IPS). Apart from the available gallery, we crop out the second image from the video ensuring no overlap with the frames actually used in probe videos.

We then fit a probabilistic subspace density on top of the IPS. It proceeds as follows: A regular PCA is performed for the IPS. Suppose the eigensystem for the IPS is  $\{(\lambda_i, e_i)\}_{i=1}^d$ , where  $d$  is the number of pixels and  $\lambda_1 \geq \dots \geq \lambda_d$ . Only top  $r$  principal components corresponding to top  $r$  eigenvalues are then kept while the residual components are considered isotropic. The density is written as follows

$$Q(x) = \left\{ \frac{\exp(-\frac{1}{2} \sum_{i=1}^r \frac{y_i^2}{\lambda_i})}{(2\pi)^{r/2} \prod_{i=1}^r \lambda_i^{1/2}} \right\} \left\{ \frac{\exp(-\frac{\varepsilon^2}{2\rho})}{(2\pi\rho)^{(d-r)/2}} \right\} \quad (13.15)$$

where the principal components  $y_i$ , the reconstruction error  $\varepsilon^2$ , and the isotropic noise variance  $\rho$  are defined as

$$y_i = e_i^T x, \quad \varepsilon^2 = \|x\|^2 - \sum_{i=1}^r y_i^2, \quad \rho = (d-r)^{-1} \sum_{i=r+1}^d \lambda_i. \quad (13.16)$$

It is easy to write the likelihood as follows:

$$p_2(z_t | n_t, \theta_t) = Q_{\text{IPS}}(\mathcal{T}_{\theta_t}\{z_t\} - I_{n_t}). \quad (13.17)$$

Table 13.2 lists the performance using this new likelihood measurement. It turns out that the performance is significantly better than in Case 1, with 93% tracked successfully and 83% correctly recognized within the top match. If we consider the top three matches, 97% are correctly identified.

**Case 4: Tracking and Recognition Using Combined Density** In Case 2, we studied appearance changes within a video sequence. In Case 3, we studied the appearance change between the gallery and the probe. In Case 4, we attempt to take advantage of both cases by introducing a combined likelihood defined as follows.

$$p_3(z_t | n_t, \theta_t) = p_2(z_t | n_t, \theta_t)q(z_t | \theta_t). \quad (13.18)$$

Again, all other settings are the same as in Case 1. We now obtain the best performance so far: no tracking error, 93% are correctly recognized as the first match, and no error in recognition when the top three matches are considered.

**Case 5: Still-to-Still Face Recognition** We also performed an experiment for still-to-still face recognition. We selected the probe video frames with the best frontal face view (i.e., biggest frontal view) and cropped out the facial region by normalizing with respect to the eye coordinates manually specified. It turns out that the recognition result is 57% correct for the top match and 83% for the top three matches. Clearly, Case 4 is the best among all.

## 13.4 Video Gallery vs. Video Probes

Here we describe a parametric model for appearance and dynamics to understand the manifold structures of these models, which are then used to devise joint appearance and dynamic based recognition algorithms.

### 13.4.1 Parametric Model for Appearance and Dynamic Variations

A wide variety of spatio-temporal data have often been modeled as realizations of dynamical models. Examples include dynamic textures [11], human joint angle trajectories [6] and silhouettes [37]. A well-known dynamical model for such time-series data is the autoregressive and moving average (ARMA) model. Linear dynamical systems represent a class of parametric models for time-series. A wide variety of time series data such as dynamic textures, human joint angle trajectories, shape sequences, video based face recognition etc., are frequently modeled as autoregressive and moving average (ARMA) models [1, 6, 11, 37]. Let  $f(t)$  be a sequence of features extracted from a video indexed by time  $t$ . The ARMA model parametrizes the evolution of the features  $f(t)$  using the following equations:

$$f(t) = Cz(t) + w(t) \quad w(t) \sim N(0, R), \quad (13.19)$$

$$z(t+1) = Az(t) + v(t) \quad v(t) \sim N(0, Q) \quad (13.20)$$

where,  $z \in \mathbb{R}^d$  is the hidden state vector,  $A \in \mathbb{R}^{d \times d}$  the transition matrix and  $C \in \mathbb{R}^{p \times d}$  the measurement matrix.  $f \in \mathbb{R}^p$  represents the observed features while  $w$  and  $v$  are noise components modeled as normal with 0 mean and covariances  $R \in \mathbb{R}^{p \times p}$  and  $Q \in \mathbb{R}^{d \times d}$ , respectively.

For high-dimensional time-series data (dynamic textures etc), the most common approach is to first learn a lower-dimensional embedding of the observations via PCA, and learn temporal dynamics in the lower-dimensional space. Closed form solutions for learning the model parameters ( $A, C$ ) from the feature sequence ( $f_{1:T}$ ) have been proposed by [11, 27] and are widely used in the computer vision community. Let observations  $f(1), f(2), \dots, f(\tau)$ , represent the features for the time indices  $1, 2, \dots, \tau$ . Let  $[f(1), f(2), \dots, f(\tau)] = U\Sigma V^T$  be the singular value decomposition of the data. Then  $\hat{C} = U$ ,  $\hat{A} = \Sigma V^T D_1 V (V^T D_2 V)^{-1} \Sigma^{-1}$ , where  $D_1 = [00; I_{\tau-1} 0]$  and  $D_2 = [I_{\tau-1} 0; 00]$ .

The model parameters  $(A, C)$  do not lie in a vector space. The transition matrix  $A$  is only constrained to be stable with eigenvalues inside the unit circle. The observation matrix  $C$  is constrained to be an orthonormal matrix. For comparison of models, the most commonly used distance metric is based on subspace angles between column-spaces of the observability matrices [10]. For the ARMA model of (13.20), starting from an initial condition  $z(0)$ , it can be shown that the *expected* observation sequence is given by

$$E \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ \vdots \\ \vdots \end{bmatrix} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ \vdots \end{bmatrix} z(0) = O_\infty(M)z(0). \quad (13.21)$$

Thus, the expected observation sequence generated by a time-invariant model  $M = (A, C)$  lies in the column space of the extended *observability* matrix given by

$$O_\infty^T = [C^T, (CA)^T, (CA^2)^T, \dots, (CA^n)^T, \dots]. \quad (13.22)$$

In experimental implementations, we approximate the extended observability matrix by the finite observability matrix as is commonly done [33]

$$O_m^T = [C^T, (CA)^T, (CA^2)^T, \dots, (CA^{m-1})^T]. \quad (13.23)$$

The size of this matrix is  $mp \times d$ . The column space of this matrix is a  $d$ -dimensional subspace of  $\mathbb{R}^{mp}$ , where  $d$  is the dimension of the state-space  $z$  in (13.20).  $d$  is typically of the order of 5–10.

Thus, given a database of videos, we estimate the model parameters as described above for each video. The finite observability matrix is computed as in (13.23). To represent the subspace spanned by the columns of this matrix, we store *an* orthonormal basis computed by Gram-Schmidt orthonormalization. Since, a subspace is a point on a *Grassmann* manifold [35, 36], a linear dynamical system can be alternately identified as a point on the Grassmann manifold corresponding to the column space of the observability matrix. The goal now is to devise methods for classification and recognition using these model parameters. Given a set of videos for a given class, we would like to compute a parametric or non-parametric class-conditional density. Then, the maximum likelihood classification for each test instance can be performed using these class conditional distributions. To enable these, we need to understand the geometry of the Grassmann manifold.

### 13.4.2 The Manifold Structure of Subspaces

The set of all  $d$ -dimensional linear subspaces of  $\mathbb{R}^n$  is called the Grassmann manifold which will be denoted as  $\mathcal{G}_{n,d}$ . The set of all  $n \times d$  orthonormal matrices is

called the Stiefel manifold and shall be denoted as  $\mathcal{S}_{n,d}$ . As discussed in the applications above, we are interested in computing statistical models over the Grassmann manifold. Let  $U_1, U_2, \dots, U_k$  be some previously estimated points on  $\mathcal{S}_{n,d}$  and we seek their sample mean, an average, for defining a probability model on  $\mathcal{S}_{n,d}$ . Recall that these  $U_i$ s are tall, orthogonal matrices. It is easy to see that the Euclidean sample mean  $\frac{1}{k} \sum_{i=1}^k U_i$  is not a valid operation, because the resultant mean does not have the property of orthonormality. This is because  $\mathcal{S}_{n,d}$  is not a vector space. Similarly, many of the standard tools in estimation and modeling theory do not directly apply to such spaces but can be adapted by accounting for the underlying nonlinear geometry.

A subspace is stored as an orthonormal matrix which forms a basis for the subspace. As mentioned earlier, orthonormal matrices are points on the Stiefel manifold. However, since the choice of basis for a subspace is not unique, any notion of distance and statistics should be invariant to this choice. This requires us to interpret each point on the Grassmann manifold as an equivalence of points on the Stiefel manifold, where all orthonormal matrices that span the same subspace are considered equivalent. This interpretation is more formally described as a *quotient* interpretation that is, the Grassmann manifold is considered a quotient space of the Stiefel manifold. Quotient interpretations allow us to extend the results of the base manifold such as tangent spaces, geodesics etc to the new quotient manifold. In our case, it turns out that the Stiefel manifold itself can be interpreted as a quotient of a more basic manifold—the special orthogonal group  $SO(n)$ . A quotient of Stiefel is thus a quotient of  $SO(n)$  as well.

A point  $U$  on  $\mathcal{S}_{n,d}$  is represented as a tall-thin  $n \times d$  orthonormal matrix. The corresponding equivalence class of  $n \times d$  matrices  $[U] = UR$ , for  $R \in GL(d)$  is called the Procrustes representation of the Stiefel manifold. Thus, to compare two points in  $\mathcal{G}_{n,d}$ , we simply compare the smallest squared distance between the corresponding equivalence classes on the Stiefel manifold according to the Procrustes representation. Given matrices  $U_1$  and  $U_2$  on  $\mathcal{S}_{n,d}$ , the smallest squared Euclidean distance between the corresponding equivalence classes is given by

$$d_{\text{Procrust}}^2([U_1], [U_2]) = \min_R \text{tr}(U_1 - U_2 R)^T(U_1 - U_2 R) \quad (13.24)$$

$$= \min_R \text{tr}(R^T R - 2U_1^T U_2 R + I_k). \quad (13.25)$$

When  $R$  varies over the orthogonal group  $O(d)$ , the minimum is attained at  $R = H_1 H_2^T = A(A^T A)^{-1/2}$ , where  $A = H_1 D H_2^T$  is the singular value decomposition of  $A$ . We refer the reader to [8] for proofs and alternate cases. Given several examples from a class  $(U_1, U_2, \dots, U_n)$  on the manifold, the class conditional density can be estimated using an appropriate kernel function. We first assume that an appropriate choice of a divergence on the manifold has been made such as the one above. For the Procrustes measure, the density estimate is given by [8] as

$$\hat{f}(U; M) = \frac{1}{n} C(M) \sum_{i=1}^n K[M^{-1/2}(I_k - U_i^T U_i M^{-1})M^{-1/2}] \quad (13.26)$$

where  $K(T)$  is the kernel function,  $M$  is a  $d \times d$  positive definite matrix which plays the role of the kernel width or a smoothing parameter.  $C(M)$  is a normalizing factor chosen so that the estimated density integrates to unity. The matrix valued kernel function  $K(T)$  can be chosen in several ways. We have used  $K(T) = \exp(-\text{tr}(T))$  in all the experiments reported in this chapter. In this non-parametric method for density estimation, the choice of kernel width  $M$  becomes important. Thus, though this is a non-iterative procedure, the optimal choice of the kernel width can have a large impact on the final results. In general, there is no standard way to choose this parameter except for cross-validation. In the experiments reported here, we use  $M = I$ , the  $d \times d$  identity matrix.

In addition to such nonparametric methods, there are principled methods to devise parametric densities on manifolds. Here, we simply refer the reader to [36] for mathematical details. In brief, using the tangent structure of the manifold, it is possible to define the well-known parametric densities such as multi-variate Gaussian, mixture-of-Gaussians etc., on the tangent spaces and wrap them back to the manifold. Densities defined in such a manner are called ‘wrapped’-densities. In the experiments section, we use a wrapped-Gaussian to model class-condition densities on the Grassmann manifold. This is compared to the simpler nonparametric method described above.

### 13.4.3 Video-Based Face Recognition Experiments

We performed a recognition experiment on the NIST’s Multiple Biometric Grand Challenge (MBGC) dataset. The MBGC Video Challenge dataset consists of a large number of subjects walking towards a camera in a variety of illumination conditions. Face regions are manually tracked and a sequence of cropped images is obtained. There were a total of 143 subjects with the number of videos per subject ranging from 1 to 5. In our experiments, we took subsets of the dataset which contained at least 2 sequences per person denoted as  $S_2$ , at least 3 sequences per person denoted as  $S_3$  etc. Each of the face-images was first preprocessed to zero-mean and unity variance. In each of these subsets, we performed a leave-one-out testing. The results of the leave one out testing are shown in Table 13.3. Also reported are the total number of distinct subjects and the total number of video sequences in each of the subsets. In the comparisons, we show results using the ‘arc-length’ metric between subspaces [13]. This metric computes the subspace angles between two subspaces and takes the Frobenius norm of the angles as a distance measure [13]. We also show comparisons with the Procrustes measure, the Kernel density estimate with  $M = I$  and a parametric wrapped Gaussian density on the manifold. The wrapped Gaussian is estimated on the tangent-plane centered at the mean-point of the dataset. The mean, more formally defined as the Karcher mean, is defined as the point that minimizes the sum of squared geodesic distances to all other points. The tangent-plane being a vector space allows the use of multi-variate statistics to define class-conditional densities. We refer the reader to [36] for mathematical details.

**Table 13.3** Comparison of video based face recognition approaches using (a) Subspace Angles + Arc-length metric, (b) Procrustes Distance, (c) kernel density, (d) Wrapped Normal on Tangent Plane

| Subset | Distinct Subjects | Total Sequences | Arc-length Metric | Procrustes Metric | Kernel density | Wrapped Normal |
|--------|-------------------|-----------------|-------------------|-------------------|----------------|----------------|
| $S_2$  | 143               | 395             | 38.48             | 43.79             | 39.74          | <b>63.79</b>   |
| $S_3$  | 55                | 219             | 48.85             | 53.88             | 50.22          | <b>74.88</b>   |
| $S_4$  | 54                | 216             | 48.61             | 53.70             | 50.46          | <b>75</b>      |
| Avg.   |                   |                 | 45.31%            | 50.45%            | 46.80%         | <b>71.22%</b>  |

As can be seen, statistical methods outperform nearest-neighbor based approaches. As one would expect, the results improve when more examples per class are available. Since the optimal kernel-width is not known in advance, this might explain the relatively poor performance of the kernel density method. More examples of statistical inference on the Grassmann manifold for image and video-based recognition can be found in [35].

## 13.5 Face Recognition in Camera Network

Video-based face recognition algorithms exploit information temporally across the video sequence to improve recognition performance. With camera networks, we can capture multi-view videos which allow us to further integrate information spatially across view angles. It is worth noting that this is different from traditional face recognition of single-camera videos in which various face poses exhibit. In that case, one usually needs to model the dynamics of pose changes in the training phase and estimate pose in the testing phase. For example, in [20], Lee et al. train a representation for the face appearance manifold. The manifold consists of locally linear subspaces for different poses. A transition probability matrix is also trained to characterize the temporal dynamics for this representation. In [23], the dynamics are encoded in the learned Hidden Markov Models (HMMs). The mean observations of hidden states are shown to represent facial images at various poses. These approaches are designed to work with a single camera.

On the other hand, in camera network deployments there are multiple images of the face in different poses at a given time instant. These images could include a mix of frontal and nonfrontal images of the face, or, in some cases, a mix of nonfrontal images (see Fig. 13.2). Videos captured in such a mode have natural advantages in providing persistent sensing over a large area and stronger cues for handling pose variations. Nonetheless, if we do not leverage the collaboration among cameras, the power of multi-view data over single-views cannot be fully exploited. For example, if we extend the single-view video-based methods, such as [20] and [23], to a camera network, they have to function in such a mode that cameras do not collaborate with each other except at the final fusion stage.

**Fig. 13.2** Images acquired by a multi-camera network. Each column corresponds to a different camera, and each row corresponds to a different time instant and subject. Note that, under unconstrained acquisition, it is entirely possible that none of the images are frontal in spite of using five cameras to observe the subject [32]



In general, there are some principles one should follow in developing a video-based face recognition algorithm for camera networks: First, the method should be able to collaboratively utilize information collected by multiple cameras and arrive at a multi-view representation from it, as opposed to perform recognition for each view individually and then fusing the result. Second, the method should be able to tackle pose variations effectively, as this is the major concern of a multi-view face recognition system. Third, the method should work on data whose acquisition conditions are as close to practical surveillance situations as possible. These conditions include: reasonable distance between subject and cameras, relatively low resolution in the face region, uncontrolled pose variations, uncontrolled subject motion, and possible interruptions in acquisition (say, the subject moves out of the field of view of a camera) etc.

Next, we will introduce a video-based face tracking and recognition framework following these principles. The system first tracks a subject's head from multi-view videos and back-projects textures to a spherical head-model. Then a rotation-invariant feature based on spherical harmonic (SH) transform is constructed from the texture maps. Finally, video-based recognition is achieved through measurement of ensemble similarity.

### 13.5.1 Face Tracking from Multi-view Videos

The tracker is set in a Sequential Importance Resampling (SIR) (particle filtering) framework, which can be broken down into a description of its state space, the state transition model and the observation model. To fully describe the position and pose of a 3D object, we usually need a 6-D representation ( $\mathbb{R}^3 \times \text{SO}(3)$ ), where the 3-D real vector space is used to represent the object's location, and the special orthogonal group  $\text{SO}(3)$  is used to represent the object's rotation. In our work, we model the human head as a sphere and perform pose-robust recognition. This enables us to explore in 3-D state space  $\mathcal{S} = \mathbb{R}^3$ . Each state vector  $\mathbf{s} = [x, y, z]$  represents the 3-D position of a sphere's center, disregarding the orientation. The radius of the sphere is assumed to be known through an initialization step. The low dimensionality of

the state space contributes to the reliability of the tracker, since for SIR, even a large number of particles will necessarily be sparse in high dimensional space.

The state transition model  $P(\mathbf{s}_t | \mathbf{s}_{t-1})$  is set as a Gaussian distribution  $\mathcal{N}(\mathbf{s}_t | \mathbf{s}_{t-1}, \sigma^2 \mathbf{I})$ . We have found that the tracking result is relatively insensitive to the specific value of  $\sigma$  and fixed it at 50 mm (our external camera calibration is metric). The observations for the filter are histograms extracted from the multi-view video frames  $I_t^j$ , where  $j$  is the camera index and  $t$  is the frame index. Histogram features are invariant to rotations and thus fit the circumstance of reduced state space. To adopt this feature, we need to back-project  $I_t^j$  onto the spherical head model and establish the histogram over the texture map. The observation likelihood is modeled as follows:

$$P(O_t | \mathbf{s}_t^{(i)}) = P(I_t^1, I_t^2, \dots, I_t^K | \mathbf{s}_t^{(i)}) \propto 1 - D(H(M_{t,i}), H_{\text{template}}), \quad (13.27)$$

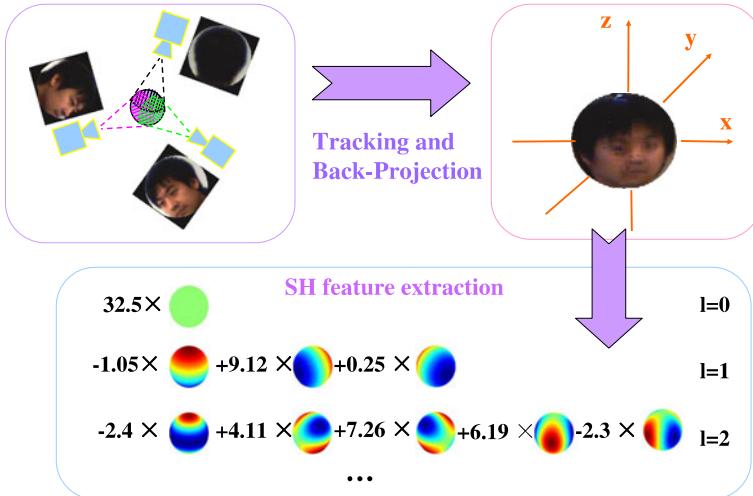
where  $\mathbf{s}_t^{(i)}$  is the  $i$ th particle at the  $t$ th frame;  $H(M_{t,i})$  is the histogram of the texture map built from the particle  $\mathbf{s}_t^{(i)}$ ;  $H_{\text{template}}$  is the histogram of template texture map. The template texture map is computed after initializing the head position in the first frame, then updated by back-projecting the head region in the image, which is fixed by the maximum a posteriori (MAP) estimate onto the sphere model. The  $D(H_1, H_2)$  function calculates the Bhattacharyya distance between two normalized histograms.

We now describe the procedure for obtaining texture map on the surface of the head model. First, we uniformly sample the spherical surface. Then for the  $j$ th camera, the world coordinates of sample points  $[x_n, y_n, z_n]$ ,  $n = 1, 2, \dots, N$  are transformed into coordinates in that camera's reference frame  $[x_n^{C_j}, y_n^{C_j}, z_n^{C_j}]$  to determine their visibility in that camera's view. Only unoccluded points (i.e., those satisfying  $z_n^{C_j} \leq z_0^{C_j}$ , where  $z_0^{C_j}$  is the distance from the head center to the  $j$ th camera center) are projected onto the image plane. By relating these model surface points  $[x_n, y_n, z_n]$  to the pixels at their projected image coordinates  $I(x_n^{P_j}, y_n^{P_j})$ , we build the texture map  $M^j$  of the visible hemisphere for the  $j$ th camera view. This continues until we have transformed the texture maps obtained from all camera views to the spherical model. Points in the overlapped region are fused using a weighting strategy, based on representing the texture map of the  $j$ th camera view as a function of locations of surface points  $M^j(x, y, z)$ . We assign the function value at point  $[x_n, y_n, z_n]$  a weight  $W_{n,j}$ , according to the point's proximity to the projection center. This is based on the fact that, on the rim of a sphere, a large number of surface points tend to project to the same pixel, so image pixels corresponding to those points are not suitable for back-projection. The intensity value at the point  $[x_n, y_n, z_n]$  of the resulting texture map will be:

$$M(x_n, y_n, z_n) = M^{j_{\max}}(x_n, y_n, z_n), \quad (13.28)$$

where

$$j_{\max} = \arg \max W_{n,j}, \quad j = 1, 2, \dots, K. \quad (13.29)$$



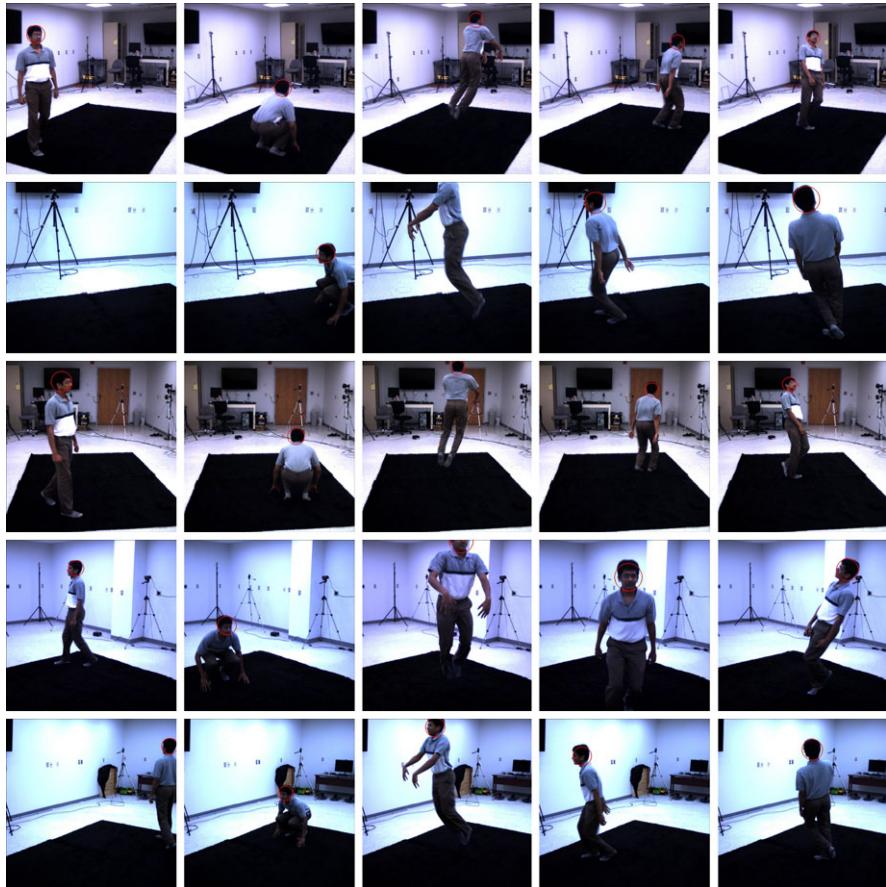
**Fig. 13.3** Feature extraction. We first obtain the texture map of the human head on the surface of a spherical model through back projection of multi-view images captured by the camera network, then represent it with spherical harmonics

The texture mapping and back-projection processes are illustrated in the left part of Fig. 13.3.

Figure 13.4 shows an example of our pose-free tracking result for a multi-view video sequence. The video sequence has 500 frames. The tracker is able to stably track all the frames without failure, despite the considerably abrupt motions and the frequent occurrences of rotation, translation and scaling of the human head as shown. Sometimes the subject's head is outside the field-of-view of certain cameras. Though subjects usually do not undergo such extreme motion in real-world surveillance videos, this example clearly illustrates the reliability of our tracking algorithm. In our experiments, the tracker handles all the captured videos without difficulty. The occasionally observed inaccuracies in bounding circles are mostly due to the difference between sphere and the exact shapes of human heads. Successful tracking enables the subsequent recognition task.

### 13.5.2 Pose-Free Feature Based on Spherical Harmonics

In this section, we describe the procedure for extracting a rotation-invariant feature from the texture map obtained in Sect. 13.5.1. The process is illustrated in Fig. 13.3. According to the Spherical Harmonics (SH) theory, SHs form a set of orthonormal basis functions over the unit sphere, and can be used to linearly expand any square-integrable function on  $S^2$ . SH representation has been used for matching 3D shapes [16] due to its properties related to the rotation group. In the vision community,



**Fig. 13.4** Sample tracking results for a multi-view video sequence. 5 views are shown here. Each row of images is captured by the same camera. Each column of images corresponds to the same time-instant

following the work of Basri and Jacobs [5], researchers have used SH to understand the impact of illumination variations in face recognition [30, 39].

The general SH representation is used to analyze complex functions (For description of general SH, please refer to [5] or [30]). However, the spherical function determined by the texture map are real functions, and thus we consider real spherical harmonics (or Tesselated SH):

$$Y_l^m(\theta, \phi) = \begin{cases} Y_{l0} & \text{if } m = 0, \\ \frac{1}{\sqrt{2}}(Y_{lm} + (-1)^m Y_{l,-m}) & \text{if } m > 0, \\ \frac{1}{\sqrt{2}i}(Y_{l,-m} - (-1)^m Y_{lm}) & \text{if } m < 0 \end{cases} \quad (13.30)$$

where  $Y_{lm}(\cdot, \cdot)$  denotes the general SH basis function of degree  $l \geq 0$  and order  $m$  in  $(-l, -l+1, \dots, l-1, l)$ . Note that here we are using the spherical coordinate system.  $\theta \in (0, \pi)$  and  $\phi \in (0, 2\pi)$  are the zenith angle and azimuth angle, respectively. The Real SHs are also orthonormal and they share most of the major properties of the general Spherical Harmonics. From now on, the word “Spherical Harmonics” shall refer only to the Real SHs. As in Fourier expansion, the SH expansion coefficients  $f_l^m$  of function  $f(\theta, \phi)$  can be computed as:

$$f_l^m = \int_{\theta} \int_{\phi} f(\theta, \phi) Y_l^m(\theta, \phi) d\theta d\phi. \quad (13.31)$$

The expansion coefficients have a very important property which is directly related to our ‘pose-free’ face recognition application:

**Proposition** *If two functions defined on  $S^2$ :  $f(\theta, \phi)$  and  $g(\theta, \phi)$  are related by a rotation  $R \in SO(3)$ , that is,  $g(\theta, \phi) = R(f(\theta, \phi))$ , and their SH expansion coefficients are  $f_l^m$  and  $g_l^m$  ( $l = 0, 1, \dots$  and  $m = -l \dots l$ ), respectively, the following relationship exists:*

$$g_l^m = \sum_{m'=-l}^l D_{mm'}^l f_l^{m'} \quad (13.32)$$

and the  $D_{mm'}^l$ s satisfy:

$$\sum_{m'=-l}^l (D_{mm'}^l)^2 = 1. \quad (13.33)$$

In other words, after rotation, the SH expansion coefficients at a certain degree  $l$  are actually linear combinations of those before the rotation, and coefficients at different degrees do not affect each other. This proposition is a direct result of the following lemma [7, 16]:

**Lemma** *Denote  $E_l$  the subspace spanned by  $Y_l^m(\theta, \phi)$ ,  $m = -l \dots l$ , then  $E_l$  is an irreducible representation for the rotation group  $SO(3)$ .*

Thus, given a texture map  $f(\theta, \phi)$  and its corresponding SH coefficient  $\{f_l^m, l = 0, 1, \dots, m = -l, \dots, l\}$ , we can formulate the energy vector associated with  $f(\theta, \phi)$  as  $e_f = (\|f_0\|_2, \|f_1\|_2, \|f_l\|_2, \dots)$ , where  $f_l$  is the vector of all  $f_l^m$  at degree  $l$ . Equation (13.33) guarantees that  $e_f$  keeps unchanged when the texture map is rotated, and this enables pose-robust face recognition. We refer to  $e_f$  as the SH Energy feature. Note that this is different from the energy feature defined in [16]. In practice, we further normalize the SH energy feature with regard to total energy. This is the same as assuming that all the texture maps have the same total energy, and somehow function as an illumination-normalized signature. Although this also means that skin color information is not used for recognition, it proves to work very well in experiments.



**Fig. 13.5** Comparison of the reconstruction qualities of head/face texture map with different number of spherical harmonic coefficients. The images from left to right are: the original 3D head/face texture map, the texture map reconstructed from 40-degree, 30-degree and 20-degree SH coefficients, respectively [32]

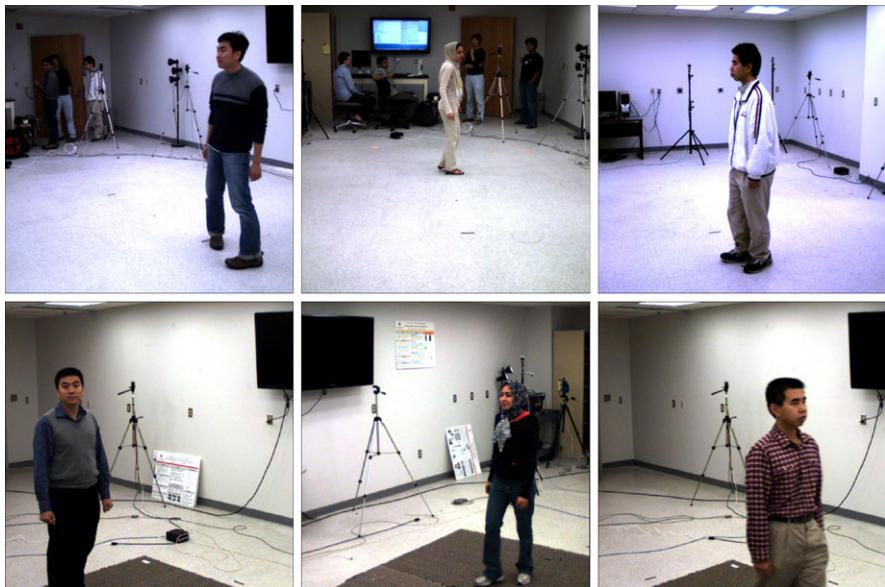
The remaining issue concerns with obtaining a suitable band-limited approximation with SH for our application. In Fig. 13.5, we show a 3D head texture map and its reconstructed version with 20, 30 and 40 degree SH transform, respectively. The ratio of computation time for the 3 cases is roughly 1:5:21. (The exact time varies with configuration of the computer, for example, on a PC with Xeon 2.13 GHz CPU, it takes roughly 1.2 seconds to do a 20 degree SH transform for 18 050 points.) We have observed that the 30-degree transform achieves the best balance between approximation precision and computational cost.

### 13.5.3 Measure Ensemble Similarity

Given two multi-view video sequences with  $m$  and  $n$  frames (Every “frame” is actually a group of images, each captured by a camera in the network.), respectively, we generate 2 ensembles of feature vectors, respectively. They may contain different number of vectors. To achieve video-level recognition, we are interested in measuring the similarity between these two sets of vectors. Now, we calculate the ensemble similarity as the limiting Bhattacharyya distance in RKHS following [41]. In experiments, we measure the ensemble similarity between feature vectors of a probe video and those of all the gallery videos. The gallery video with the shortest distance to the probe is considered as the best match. For detailed derivations and explanation of limiting Bhattacharyya distance in the RKHS, please refer to [41].

### 13.5.4 Experiments

Most existing “multi-view” still or video face databases, such as PIE, Yale-B, the oriental face data, M2VTS etc., target recognition-across-pose algorithms, so they are not applicable to our multi-view to multi-view matching algorithm. The data we used in this work are multi-view video sequences captured with 4 or 5 video cameras in an indoor environment, collected at 3 different sessions: one for building



**Fig. 13.6** Example of gallery and probe video frames. Images on the top are gallery frames, and those on the bottom are probe frames of the same subjects. Many subjects look differently in gallery and probe

a gallery and the other two for constructing probes. To test the robustness of our recognition algorithm, we arranged the second session to be one week after the first one, and the third 6 months after the second. The appearance of some subjects change significantly between the sessions. The database enrolls 25 subjects. Each subject has 1 gallery video and most subjects have 2 probe videos. Each video is 100 to 200 frames in length. Since each video sequence is captured with multiple cameras, it is equivalent to 4 or 5 videos in the single camera case. Figure 13.6 shows some example frames from gallery and probe video sequences. This data set poses great challenges to multi-view face recognition algorithms.

#### 13.5.4.1 Feature Comparison

We associate 5 different kinds of features with different classifiers to compare their performance in image-based face recognition systems. By “image-based face recognition” we mean that each frame is treated as gallery or probe individually and no video-level fusion of results is performed. As a result, the recognition rate is computed by counting the number of correctly classified frames, not videos. The inputs to all these face recognition systems are based on the same tracking results. For any system based on feature of raw image intensity value, we use only the head region that is cropped by a circular mask as provided by the tracking result. All the head images are scaled to  $30 \times 30$ . For the PCA features, Eigenvectors that preserve the

**Table 13.4** Comparison of recognition performance

| Feature              | NN           | KDE          | SVM-Linear   | SVM-RBF      |
|----------------------|--------------|--------------|--------------|--------------|
| Intensity PCA        | 49.7%        | 39.0%        | 49.2%        | 57.8%        |
| Intensity LDA        | 50.5%        | 27.2%        | 33.1%        | 40.7%        |
| SH PCA               | 33.6%        | 30.9%        | 31.2%        | 44.2%        |
| SH Energy            | 55.3%        | 47.9%        | 50.3%        | 67.1%        |
| Normalized SH Energy | <b>60.8%</b> | <b>64.7%</b> | <b>78.2%</b> | <b>86.0%</b> |

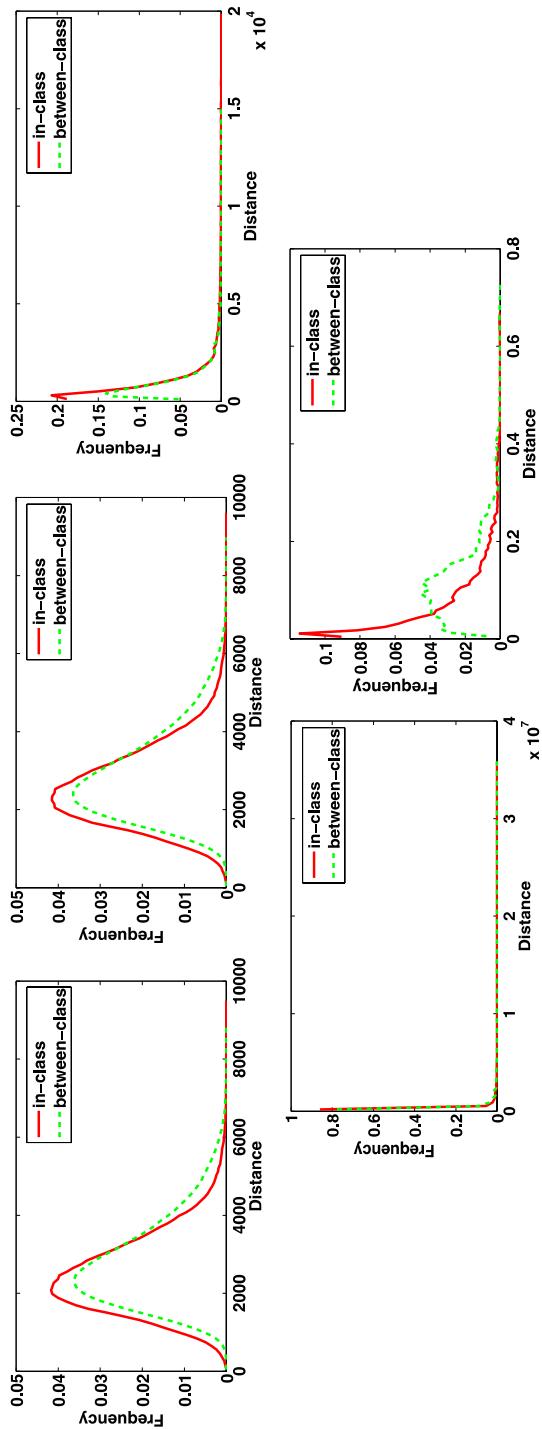
**Table 13.5** KL divergence of in-class and between-class distances for different features

| Intensity | Intensity + PCA | SH + PCA | SH Energy | Normalized SH Energy |
|-----------|-----------------|----------|-----------|----------------------|
| 0.1454    | 0.1619          | 0.2843   | 0.1731    | <b>1.1408</b>        |

top 95% energy are kept. For the SH-based feature, we perform a 30-degree SH transform. Here, we would like to emphasize that since both gallery and probe are captured when subjects are performing free motion, the poses exhibited in images of any view are arbitrary and keep changing. This is significantly different from the settings of most existing multi-view face databases. The results are shown in Table 13.4. As we can see, the performance of the proposed feature exceeds that of other features by a large margin in all cases. Note that we do not fuse the results of different views for non-SH-based features.

To quantitatively verify the proposed feature's discrimination power, we then conducted the following experiment. We calculate distances for each unordered pair of feature vectors  $\{x_i, x_j\}$  in the gallery. If  $\{x_i, x_j\}$  belongs to the same subject, then the distance is categorized as being *in-class*. Otherwise, the distance is categorized as being *between-class*. We approximate the distribution of the two kinds of distances as histograms.

Intuitively, if a feature has good discrimination power, then the in-class distances evaluated using that feature tends to take smaller values compared to the between-class distances. If the two distributions mix together, then this feature is not good for classification. We use the symmetric KL divergence  $\text{KL}(p\|q) + \text{KL}(q\|p)$  to evaluate the difference between the two distributions. We summarize the values of KL divergence for the 5 features in Table 13.5 and plot the distributions in Fig. 13.7. As clearly shown, the in-class distances for normalized SH energy feature are concentrated in the low value bins, while the between-class ones tend to have higher values, and their modes are obviously separated from each other. For all other features, the between-class distances do not show a clear trend of being larger than the in-class ones, and their distributions are just mixed. The symmetric KL-divergence also suggests the same.



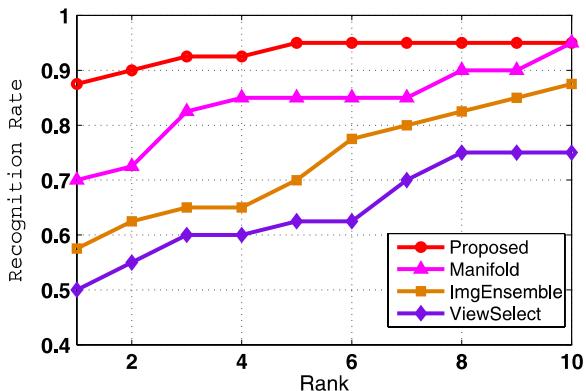
**Fig. 13.7** Comparison of the discriminant power of the 5 features. *First row*, from left to right: intensity value + PCA, SH, SH + PCA. *Second row*, from left to right: SH Energy, Normalized SH energy. The green curve is between-class distance distribution and the red one is in-class distance distribution. Number of bins is 100 [32]

### 13.5.4.2 Video-Based Recognition

In this experiment, we compare the performance of 4 video-level recognition systems: (1) Ensemble-similarity-based algorithm as proposed in [41] for cropped face images. The head images in a video are automatically cropped by a circular mask as provided by the tracking results and scaled to 30 by 30. Then we calculate the limiting Bhattacharyya distance between gallery and probe videos in RKHS for recognition. The kernel is RBF. If a video has  $n$  frames and it is captured by  $k$  cameras, then there are  $k \times n$  head (face) images in the ensemble. (2) View-selection-based algorithm. We first train a PCA subspace for frontal-view face. The training images are a subset of the Yale B database and are scaled to 30 by 30. We then use this subspace to pick frontal-view face images from our gallery videos. We construct a frontal-view face PCA subspace for each individual. For every frame of a probe video, we first compute the “frontalness” of the subject’s face in each view according to its distance to the general PCA model. The view which best matches the model is selected and fitted to the individual PCA subspaces of all the subjects. After classification of all the frames has been finished, recognition result for the video is obtained through majority voting. (3) video-based face recognition algorithm using probabilistic appearance manifold as proposed in [20]. We use 8 planes for local manifold model and set the probability of remaining the same pose to be 0.7 in the pose transition probability matrix. We first use this algorithm to process each view of a probe video. To fuse results of different views we use majority voting. If there is a tie in views’ voting, we pick the one with smaller Hausdorff distance as the winner. (4) Normalized SH energy feature + ensemble similarity. This algorithm is as described in Sect. 13.5.2 and Sect. 13.5.3.

We plot the cumulative recognition rate curve in Fig. 13.8. Note that the numbers shown here should not be compared with those in the previous image-based recognition experiment to draw misleading conclusions, as these two sets of recognition rates are not convertible to each other. The view-selection method heavily relies on the availability of frontal-view face images, however, in the camera network case, the frontal pose may not appear in any view of the cameras. As a result, it does not perform well in this multi-view to multi-view matching experiment. Rather than the ad-hoc majority voting fusion scheme adopted by the view-selection algorithm, the manifold-based algorithm and the image-ensemble-based algorithm use more reasonable strategies to combine classification results of individual frames. Moreover, they both have certain ability to handle pose variations, especially the manifold-based one. However, because they are designed to work with a single camera, they are single-view in nature. Repeating these algorithms for each view does not fully utilize the multi-view information. On the other hand, the proposed method is multi-view in nature and is based on a pose-free feature, so it performs noticeably better than the other 3 algorithms in this experiment.

**Fig. 13.8** Cumulative recognition rate of the 4 video-based face recognition algorithms



## 13.6 Conclusions

Video offers several advantages for face recognition, in terms of motion information and availability of more views. We reviewed several techniques that exploit video by either fusing information on a per-frame basis, considering them as image-ensembles, or by learning better appearance models. However, the availability of video opens interesting questions of how to exploit the temporal correlation for better tracking of faces, how to exploit behavioral cues available from video, and how to fuse the multiple views afforded by a camera network. Also, algorithms need to be derived that allow for matching a probe video to a still or video gallery. We showed applications involving such scenarios and discussed the issues involved in designing algorithms for such scenarios. There are several future research directions that are promising. While there are several studies that suggest that humans can recognize faces in non-cooperative conditions [26]—poor resolution, bad lighting etc.—if motion and dynamic information is available. This capability has been difficult to describe mathematically and replicate in an algorithm. If this phenomenon can be modeled mathematically, it could lead to more accurate surveillance and biometric systems. The role of familiarity in face recognition and the role that motion plays in recognition of familiar faces, while well known in psychology and neuroscience literature [31], is yet another avenue that has been challenging to model mathematically and replicate algorithmically.

**Acknowledgements** Supported by a MURI Grant N00014-08-1-0638 from the Office of Naval Research. The authors would like to thank Dr. Aswin Sankaranarayanan for helpful discussions related to Sect. 13.5.

## References

1. Aggarwal, G., Roy-Chowdhury, A., Chellappa, R.: A system identification approach for video-based face recognition. In: International Conference on Pattern Recognition, Cambridge, UK, August 2004

2. Anderson, B., Moore, J.: Optimal Filtering. Prentice Hall, Englewood Cliffs (1979)
3. Arandjelovic, O., Cipolla, R.: Face recognition from video using the generic shape-illumination manifold. In: European Conference on Computer Vision, pp. 27–40 (2006)
4. Arandjelovic, O., Shakhnarovich, G., Fisher, J., Cipolla, R., Darrell, T.: Face recognition with image sets using manifold density divergence. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 581–588, San Diego, USA, June 2005
5. Basri, R., Jacobs, D.: Lambertian reflectance and linear subspaces. In: Proceedings of IEEE International Conference on Computer Vision, vol. 2, pp. 383–390 (2001)
6. Bissacco, A., Chiuso, A., Ma, Y., Soatto, S.: Recognition of human gaits. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 52–57, Hawaii, USA, December 2001
7. Brocker, T., Dieck, T.: Representations of Compact Lie Groups. Springer, Berlin (2003)
8. Chikuse, Y.: Statistics on Special Manifolds. Lecture Notes in Statistics. Springer, New York (2003)
9. Choudhury, T., Clarkson, B., Jebara, T., Pentland, A.: Multimodal person recognition using unconstrained audio and video. In: Proc. of Intl. Conf. on Audio- and Video-Based Person Authentication, pp. 176–181 (1999)
10. Cock, K.D., Moor, B.D.: Subspace angles between ARMA models. *Syst. Control Lett.* **46**, 265–270 (2002)
11. Doretto, G., Chiuso, A., Wu, Y.N., Soatto, S.: Dynamic textures. *Int. J. Comput. Vis.* **51**(2), 91–109 (2003)
12. Doucet, A., Godsill, S., Andrieu, C.: On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.* **10**(3), 197–209 (2000)
13. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.* **20**(2), 303–353 (1999)
14. Fan, W., Yeung, D.-Y.: Locally linear models on face appearance manifolds with application to dual-subspace based classification. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1384–1390, New York, NY, USA, June 2006
15. Hamm, J., Lee, D.D.: Grassmann discriminant analysis: a unifying view on subspace-based learning. In: International Conference on Machine Learning, pp. 376–383, Helsinki, Finland, June 2008
16. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3d shape descriptors. In: Proceedings of Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, pp. 156–164 (2003)
17. Kim, T.K., Kittler, J., Cipolla, R.: Discriminative learning and recognition of image set classes using canonical correlations. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(6), 1005–1018 (2007)
18. Kitagawa, G.: Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Comput. Graph. Stat.* **5**, 1–25 (1996)
19. Lee, K.-C., Ho, J., Kriegman, D.J.: Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(5), 684–698 (2005)
20. Lee, K.-C., Ho, J., Yang, M.-H., Kriegman, D.J.: Visual tracking and recognition using probabilistic appearance manifolds. *Comput. Vis. Image Underst.* **99**(3), 303–331 (2005)
21. Li, Y., Gong, S., Liddell, H.: Video-based online face recognition using identity surfaces. In: Proceedings of the IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, pp. 40–46 (2001)
22. Liu, J., Chen, R.: Sequential Monte Carlo for dynamic systems. *J. Am. Stat. Assoc.* **93**, 1031–1041 (1998)
23. Liu, X., Chen, T.: Video-based face recognition using adaptive hidden Markov model. In: Proceedings of Computer Vision and Pattern Recognition, vol. 1, pp. 340–345 (2003)
24. Liu, X., Chen, T., Thornton, S.M.: Eigenspace updating for non-stationary process and its application to face recognition. *Pattern Recognit.* **36**, 1945–1959 (2003)
25. Moghaddam, B., Pentland, A.: Probabilistic visual learning for object representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**, 696–710 (1997)

26. O'Toole, A.J., Roark, D., Abdi, H.: Recognizing moving faces: a psychological and neural synthesis. *Trends Cogn. Sci.* **6**, 261–266 (2002)
27. Overschee, P.V., Moor, B.D.: Subspace algorithms for the stochastic identification problem. *Automatica* **29**(3), 649–660 (1993)
28. Park, U., Jain, A.K., Ross, A.: Face recognition in video: adaptive fusion of multiple matchers. In: Proceedings of IEEE Computer Society Workshop on Biometrics (In Conjunction with CVPR), pp. 1–8 (2007)
29. Philipps, P., Moon, H., Rivzi, S., Ross, P.: The Feret evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 1090–1104 (2000)
30. Ramamoorthi, R.: Analytic pca construction for theoretical analysis of lighting variability in images of a Lambertian object. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(10), 1322–1333 (2002)
31. Roark, D.A., Barrett, S.E., O'Toole, A.J., Abdi, H.: Learning the moves: The effect of familiarity and facial motion on person recognition across large changes in viewing format. *Perception* **761–773** (2006)
32. Ross, A.A., Nandakumar, K., Jain, A.K.: Handbook of Multibiometrics. International Series on Biometrics. Springer, New York (2006)
33. Saisan, P., Doretto, G., Wu, Y.N., Soatto, S.: Dynamic texture recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 58–63, Hawaii, USA, December 2001
34. Shakhnarovich, G., Fisher, J.W., Darrell, T.: Face recognition from long-term observations. In: Proceedings of the European Conference on Computer Vision, vol. 3, pp. 851–865, May 2002
35. Turaga, P., Veeraraghavan, A., Chellappa, R.: Statistical analysis on Stiefel and Grassmann manifolds with applications in computer vision. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8, Alaska, USA, June 2008
36. Turaga, P., Veeraraghavan, A., Srivastava, A., Chellappa, R.: Statistical analysis on manifolds and its applications to video analysis. In: Schonfeld, D., Shan, C., Tao, D., Wang, L. (eds.) *Video Search and Mining. Studies in Computational Intelligence*, Chap. 5. Springer, Berlin (2010)
37. Veeraraghavan, A., Roy-Chowdhury, A., Chellappa, R.: Matching shape sequences in video with an application to human movement analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(12), 1896–1909 (2005)
38. Xu, Y., Roy-Chowdhury, A., Patel, K.: Pose and illumination invariant face recognition in video. In: Proceedings of IEEE Computer Society Workshop on Biometrics (In Conjunction with CVPR), pp. 1–7 (2007)
39. Zhang, L., Samaras, D.: Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(3), 351–363 (2006)
40. Zhao, W.Y., Chellappa, R., Rosenfeld, A., Phillips, P.: Face recognition: a literature survey. *ACM Comput. Surv.* **35** (2003)
41. Zhou, S.K., Chellappa, R.: From sample similarity to ensemble similarity: probabilistic distance measures in reproducing kernel Hilbert space. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(6), 917–929 (2006)
42. Zhou, S., Krueger, V., Chellappa, R.: Probabilistic recognition of human faces from video. *Comput. Vis. Image Underst.* **91**, 214–245 (2003)
43. Zhou, S., Chellappa, R., Moghaddam, B.: Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Trans. Image Process.* (2004)



# Chapter 14

## Face Recognition at a Distance

Frederick W. Wheeler, Xiaoming Liu, and Peter H. Tu

### 14.1 Introduction

Face recognition, and biometric recognition in general, have made great advances in the past decade. Still, the vast majority of practical biometric recognition applications involve cooperative subjects at close range. Face Recognition at a Distance (FRAD) has grown out of the desire to automatically recognize people out in the open, and without their direct cooperation. The face is the most viable biometric for recognition at a distance. It is both openly visible and readily imaged from a distance. For security or covert applications, facial imaging can be achieved without the knowledge of the subject. There is great interest in iris at a distance, however it is doubtful that iris will outperform face with comparable system complexity and cost. Gait information can also be acquired over large distances, but face will likely continue to be a more discriminating identifier.

In this chapter, we will review the primary driving applications for FRAD and the challenges still faced. We will discuss potential solutions to these challenges and review relevant research literature. Finally, we will present a few specific activities to advance FRAD capabilities and discuss expected future trends. For the most part, we will focus our attention on issues that are unique to FRAD. Some of the main challenges of FRAD are shared by many other face recognition applications, and are thoroughly covered in other dedicated chapters of this book.

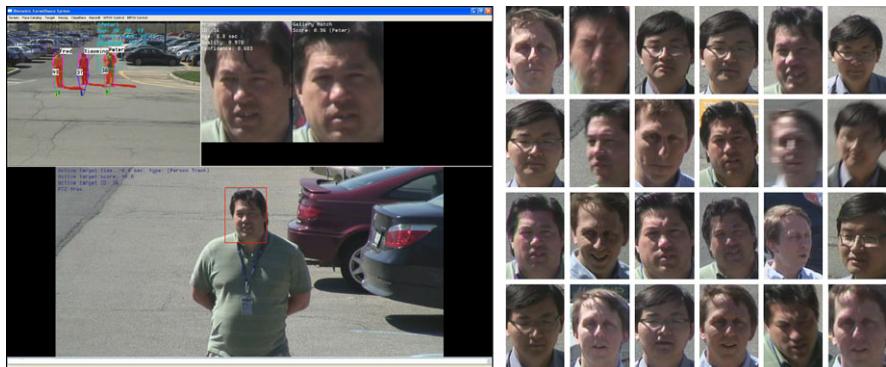
Distance itself is not really the fundamental motivating factor for FRAD. The real motivation is to work over large coverage areas without subject cooperation.

---

F.W. Wheeler (✉) · X. Liu · P.H. Tu  
Visualization and Computer Vision Lab, GE Global Research, Niskayuna, NY 12309, USA  
e-mail: [wheeler@ge.com](mailto:wheeler@ge.com)

X. Liu  
e-mail: [liux@ge.com](mailto:liux@ge.com)

P.H. Tu  
e-mail: [tu@ge.com](mailto:tu@ge.com)



**Fig. 14.1** On the left, a face recognition at a distance application showing tracked and identified subjects in wide field-of-view video (*upper left*), high-resolution narrow field-of-view video from an automatically controlled PTZ camera (*bottom*), and a detected and recognized facial image (*upper right*). On the right, some of the facial images captured by the system over a few minutes, selected to show the variation in facial image quality

The nature of the activity of subjects and the size of the coverage area can vary considerably with the application and this impacts the degree of difficulty. Subjects may be sparse and standing or walking along predictable trajectories, or they may be crowded, moving in a chaotic manner, and occluding each other. The coverage area may range from a few square meters at a doorway or choke point, to a transportation terminal, building perimeter, city block, or beyond. Practical solutions do involve image capture from a distance, but the field might be more accurately called *face recognition of noncooperative subjects over a wide area*. Figure 14.1 shows a FRAD system operating in a parking lot. There are two primary difficulties faced by FRAD. First, acquiring facial images from a distance. Second, recognizing the person in spite of imperfections in the captured data.

There are a wide variety of commercial, security, defense and marketing applications of FRAD. Some of the most important potential applications include:

- Access control: Unlock doors when cleared persons approach.
- Watch-list recognition: Raise an alert when a person of interest, such as a known terrorist, local offender or disgruntled ex-employee is detected in the vicinity.
- White-list recognition: Raise an alert whenever a person not cleared for the area is detected.
- Rerecognition: Recognize people recently imaged by a nearby camera for automatic surveillance with long-range persistent tracking.
- Event logging: For each person entering a region, catalog the best facial image.
- Marketing: Understand long-term store customer activities and behavior.

The *Handbook of Remote Biometrics* [53] also contains chapters on FRAD. The focus in that book is somewhat complementary, covering system issues and a more detailed look at illumination levels, optics and image sensors for face imaging at distances up to the 100–300 m range and beyond with both theoretical analysis and practical design advice.

### 14.1.1 Primary Challenges

In the ideal imaging conditions for 2D face recognition, the subject is illuminated in a uniform manner, is facing a color camera with a neutral expression and the image has a resolution with 200 or more pixels eye-to-eye. These conditions are easily achieved with a cooperative subject at close range.

With FRAD, the subject is by definition not at close range, but perhaps more importantly, the level of cooperation is reduced. Applications of FRAD for cooperative subjects are at best unusual and rare. In typical FRAD applications, subjects are not cooperative, and this is the scenario that is assumed in most research work on FRAD. Noncooperative subjects may be either unaware that facial images are being collected, or aware but unconcerned, perhaps due to acclimation. That is, they are neither actively cooperating with the system, nor trying to evade the system.

A much more challenging situation occurs when subjects are actively evasive. A subject may attempt to evade face capture and recognition by obscuring their face with a hat, glasses or other adornments, or by deliberately looking away from cameras or downward. In such situations it might still be beneficial for a system to automatically determine that the subject is evasive.

In a sense, FRAD is not a specific core technology or basic research problem. It can be viewed as an application and a system design problem. Some of the challenges in that design are specific to FRAD, but many are broader face recognition challenges that are discussed and addressed throughout this book. The main challenges of FRAD are concerned with the capture of facial images that have the best quality possibly, and with processing and face recognition that is robust to the remaining imperfections. These challenges can be organized into a few categories, which we discuss below.

The first challenge of FRAD is simply acquiring facial images for subjects who may be 10–30 m or more away from the sensor. Some of the optics issues to consider are lens parameters, exposure time, and the effect on the image when any compromise is made.

### 14.1.2 Optics and Light Intensity

As subject distance increases, a primary issue is the selection or adjustment of the camera lens to maintain field of view and image intensity. As the distance from the camera to the subject is increased the focal length of the camera lens must be increased proportionally if we are to maintain the same field of view, or image sampling resolution. That is, if the subject distance is doubled, then the focal length,  $F$ , must be doubled to maintain a facial image size of, say, 200 pixels eye-to-eye.

The light intensity a lens delivers to the image sensor is proportional to the f-number. The f-number,  $N$ , of a lens is the focal length divided by the diameter of the entrance pupil,  $D$ , or  $N = F/D$ . To maintain image brightness, and thus contrast and signal to noise ratio, the f-number must be maintained. So, if the subject

distance is doubled and the focal length is doubled, then the f-number of that particular lens must be maintained by doubling the pupil diameter. Of course, an image sensor with greater sensitivity may also be used to compensate for a reduction in light intensity from the lens.

If the pupil diameter for a lens is already restricted with an adjustable aperture stop, then increasing the pupil diameter to maintain the f-number is simply a matter of adjusting that setting. Adjustments to the pupil diameter are usually described in terms of the resulting f-number, and are typically called f-stops. The f-stops are defined as the f-number at the image when the lens is focused at infinity or very far away.

However, as subject distance is increased, eventually an adjustable aperture will be fully open and the pupil aperture will be limited by the size of the lens itself. So, imaging faces well at larger distances generally requires larger lenses, which have larger glass elements, are heavier and more expensive. Another drawback to increasing the diameter of the lens is a reduction in depth of field (DOF), the range over which objects are well focused. However, DOF is inherently larger at larger object distances, so this is often less of a concern for FRAD.

### ***14.1.3 Exposure Time and Blur***

There are a number of different types of image distortion that can be of concern when capturing faces at a distance. If an appropriate lens is selected, then a facial image captured at a distance can be as bright as an image captured at close range. However, this is not always the situation. Lenses with large diameters are expensive or simply may not be in place. When the lens does not have a low enough f-number (large enough aperture relative to the focal length), the amount of light reaching the image sensor will be too low and the image SNR will be reduced. If the image is amplified to compensate, sensor noise will be amplified as well and the resulting image will be noisy. Without amplification, the image will be dark.

If the light intensity at the sensor is too low, the exposure time for each image can be increased to compensate. However, this introduces a trade-off with motion blur. The subjects being imaged are generally in motion. If the exposure time is long enough that the motion of the subjects is significant during the exposure, then some degree of blurring will occur.

FRAD systems often utilize active pan-tilt camera control. Mechanical vibration of the camera can be significant in such systems and is another source of blur. At very long distances, atmospheric distortion and haze can also contribute to image distortion.

### ***14.1.4 Image Resolution***

In FRAD applications that lack an optimal optical system to provide an ideal image to the sensor, resolution of the resulting facial image can be low. In some cases,

it may be desired to recognize people in video from a stationary camera where facial image resolution is low due to subject distance. An active camera system with automatic pan, tilt and zoom may simply reach its capture distance limit, but one may still want to recognize people at greater distances. No matter how the optical system is designed, there is always some further desired subject distance and in these cases facial image resolution will be reduced. Facial recognition systems that deal with low-resolution facial images are certainly desirable.

### ***14.1.5 Pose, Illumination and Expression***

The Pose, Illumination and Expression (PIE) challenges are not unique to FRAD. There are many other face recognition applications that share these challenges. The PIE challenges are, however, somewhat customized and more pronounced in FRAD.

In many FRAD applications, it is desirable to mount cameras well above people's heads, as is done for most ordinary security cameras. This allows for the imaging of people's faces over a wider area with less occlusion. A disadvantage is that the viewing angle of faces has a slight downward tilt, often called the "surveillance perspective."

The pan angle (left-right) of faces in FRAD applications can in the worst cases be completely arbitrary. In open areas where there are no regular travel directions, this will be the case. Corridors and choke points are more favorable situations, generally limiting the directions in which people are facing. People tend to face the direction of their travel. When faces can be oriented in many directions, the use of a distributed set of active pan-tilt-zoom cameras can help [25]. Still, the variation of facial capture pan angle can be high.

There is some hope for this inherent pose problem with FRAD, and it is observation time. A wide-area active camera system may be able to observe and track a walking subject for 5–10 seconds or more. A stationary or loitering person may be observed for a much longer period of time. A persistent active face capture system, may eventually opportunistically capture a facial image of any particular subject with a nearly straight-on pose angle.

Most FRAD applications are deployed outdoors with illumination conditions that are perhaps the most challenging. Illumination is typically from sunlight or distributed light fixtures. The direction and intensity of the sunlight will change with the time of day, the weather and the seasons. Over the capture region there may be large objects such as trees and buildings that both block and reflect light, and alter the color of ambient light, increasing the variation of illumination over the area.

Subjects who are not trying to evade the system and who are not engaged in conversation will for the most part have a neutral expression. Of the PIE set of challenges, the expression issue is generally less of a concern for FRAD.

### 14.1.6 Approaches

There are two basic approaches to FRAD: high-definition stationary cameras, and active camera systems. FRAD generally means face recognition not simply at a distance, but over a wide area. Unfortunately, a wide camera viewing area that captures the entire coverage area results in low image resolution. Conversely, a highly zoomed camera that yields high-resolution facial images has a narrow field of view.

#### 14.1.6.1 High-Definition Stationary Camera

If a FRAD capture sensor is to operate over a 20 m wide area with a single camera and we require 100 pixels across captured faces, then we would need a camera with about 15 000 pixels of horizontal resolution. Assuming an ordinary sensor aspect ratio, this is a 125 *megapixel* sensor and not currently practical. If we were to use a high-definition 1080 by 1920 pixel camera to image the full 20 m wide area, a face would be imaged with a resolution of about 13 by 7 pixels.

If the coverage region is not too large, say 2 m across, then a single stationary high-definition 1080 by 1920 camera could image faces in the region with about 100 pixels eye-to-eye. This is not very high resolution for face recognition, but may be sufficient for verification or low-risk applications. If the desired coverage area grows, and stationary cameras are still used, then the camera resolution would have to increase, or multiple cameras would be required.

#### 14.1.6.2 Active-Vision Systems

FRAD is more often addressed with a multi-camera system where one or more Wide field Of view (WFOV) cameras view a large area with low-resolution and one or more Narrow Field Of View (NFOV) cameras are actively controlled to image faces with high resolution using pan, tilt and zoom (PTZ) commands. Through the use of face detection or person detection, and possibly tracking, the location of people is determined from the WFOV video. The NFOV are targeted to detected or tracked people through pan and tilt control, and possibly also adaptive zoom control. The WFOV and NFOV cameras are sometimes called the master and slave cameras, and NFOV cameras are often simply called PTZ cameras.

This is also often described as a “foveated imaging” or “focus-of-attention” approach and it somewhat mimics the human visual system, where a wide angular range is monitored with relatively low resolution and the eyes are actively directed toward areas of interest for more detailed resolution. This is especially the case when the WFOV and NFOV cameras are co-located.

What we describe here is a prototypical approach. There are of course many possible modifications and improvements. A single camera may be used with a system that enables switching between WFOV and NFOV lenses, with the additional challenge that wide-field video coverage will not be continuous. A low-zoom WFOV

camera may not be stationary, but could instead pan and tilt with a high-zoom NFOV camera, more like the human eye or a finderscope. Instead of using a WFOV camera, a single NFOV camera could continuously scan a wide area by following a pan and tilt angle pattern. One could use many WFOV cameras and many NFOV cameras in a single cooperative network. Clearly, there are many options for improving upon the prototypical approach, with various advantages and disadvantages.

When multiple subjects are present a multi-camera system must decide somehow how to schedule its NFOV camera or cameras. It needs to determine when to point the camera at each subject. There has been considerable research effort put into this NFOV resource allocation and scheduling problem, which becomes more complicated as more NFOV cameras are utilized. Some of the factors that a scheduling algorithm may account for include: which subjects are facing one of the NFOV cameras, the number of times each subject's face has been captured thus far, the quality and resolution of those images, the direction of travel and speed of each subject, and perhaps the specific location of each subject. An NFOV target scheduling algorithm accounts for some desired set of factors such as these and determines when and where to direct the NFOV cameras using their pan, tilt and zoom controls.

### ***14.1.7 Literature Review***

#### **14.1.7.1 Databases**

Most test databases for face recognition contain images or video captured at close range with cooperative subjects. They are thus best suited for training and testing face recognition for access control applications. However, there are a few datasets that are more suitable for face recognition at a distance development and evaluation.

The database collected at the University of Texas at Dallas (UTD) for the DARPA Human ID program [40] includes close-up still images and video of subjects and also video of persons walking toward a still camera from distances of up to 13.6 m and video of persons talking and gesturing from approximately 8 m. The collection was performed indoors, but in a large open area with one wall made entirely of glass, approximating outdoor lighting conditions. A fairly low zoom factor was used in this collection.

Yao et al. [58] describe the University of Tennessee, Knoxville Long Range High Magnification (UTK-LRHM) face database of moderately cooperative subjects at distances between 10 m and 20 m indoors, and extremely long distances between 50 m and 300 m outdoors. Indoor zoom factors are between 3 and 20, and outdoor zoom factors range up to 284. Imaging at such extremes can result in distortion due to air temperature and pressure gradients, and the optical system used exhibits additional blur at such magnifications.

The NIST Multiple Biometric Grand Challenge (MBGC) is focused on face and iris recognition with both still images and video and has sponsored a series of challenge problems. In support of the unconstrained face recognition challenges, this

program has collected high-definition and standard definition outdoor video of subjects walking toward the camera and standing at ranges of up to about 10 m. Since subjects were walking toward the camera, frontal views of their faces were usually visible. MBGC is also making use of the DARPA Human ID data described above [39].

It is important to remember that as distance increases, people also have increased difficulty in recognizing faces. Face recognition results from Pittsburgh Pattern Recognition on MBGC uncontrolled video datasets (of the subjects walking toward the camera) are comparable to and in some cases superior to face recognition results by humans on the same data [39].

Each of these databases captures images or video with stationary cameras. FRAD sensors are generally real-time actively controlled camera systems. Such hardware systems are difficult to test offline. The evaluation of active camera systems with shared or standardized datasets is not feasible because real-time software and hardware integration aspects are not modeled. Components of these systems, such as face detection, person detection, tracking and face recognition itself can be tested in isolation on appropriate shared datasets. But interactions between the software and hardware components can only be fully tested on live action scenes. Virtual environments can also be used to test many aspects of an active-vision system [43–45] (Sect. 14.1.7.3).

#### 14.1.7.2 Active-Vision Systems

There have been a great many innovations and systems developed for wide-area person detection and tracking to control NFOV cameras to capture facial images at a distance. We review a selected group of publications in this section, in approximate chronological order. A few of the systems described here couple face capture to face recognition. All are motivated by this possibility.

In some very early work in this area, Stillman et al. [52] developed an active camera system for person identification using two WFOV cameras and two NFOV cameras. This real-time system worked under some restricted conditions, but over a range of several meters, detected people based on skin color, triangulated 3D locations, and pointed NFOV cameras at faces. A commercial face recognition system then identified the individuals. Interestingly, the motivation for this effort, at its time in history, was to make an intelligent computing environment more aware of people present in order to improve the interaction. No mention is made of security.

Greiffenhagen et al. [21] describe a dual camera face capture system where the WFOV camera is an overhead omnidirectional camera and the NFOV has pan, tilt and zoom control. The authors use a systematic engineering methodology in the design of this real-time system, and perform a careful statistical characterization of components of the system so that measurement uncertainty can be carried through and a face capture probability of 0.99 can be guaranteed. While face recognition was not applied to the captured images, they are of sufficient quality for recognition. The system handles multiple subjects as long as the probability of occlusion is small.

Zhou et al. [64] have developed their Distant Human Identification (DHID) system to collect biometric information of humans at a distance, for face recognition and gait. A single WFOV camera has a 60° field of view and enables tracking of persons out to a distance of 50 m. A combination of background subtraction, temporal differencing, optical flow and color-based blob detection is used for person detection and tracking. The system aims to capture short zoomed-in video sequences for gait recognition and relatively high-resolution facial images. Person detections from the WFOV video are used to target the NFOV camera initially, and then the NFOV tracks the subject based only on the NFOV video data.

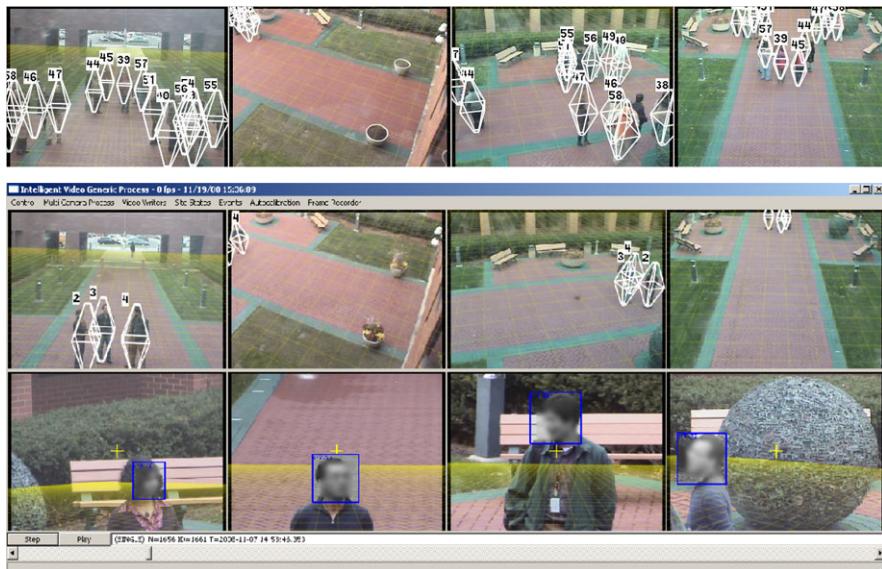
Marchesotti et al. [34] have also developed a two-camera face capture at a distance system. Persons are detected and tracked using a blob detector in the WFOV video, and an NFOV camera is panned and tilted to acquire short video clips of subject faces.

A *face cataloger* system has been developed and described by Hampapur et al. [22]. This system uses two widely separated WFOV cameras with overlapping views of a 20 ft. by 19 ft. lab space. To detect persons, a 2D multi-blob tracker is applied to the video from each WFOV camera and these outputs are combined by a 3D multi-blob tracker to determine 3D head locations in a calibrated common coordinate system. An active camera manager then directs the two pan-tilt NFOV cameras to capture facial images. In this system, a more aggressive zoom factor is used when subjects are moving more slowly. The authors experimentally demonstrate a trade-off between NFOV zoom and the probability of successfully capturing a facial image. When the NFOV zoom factor is higher, the required pointing accuracy is greater, so there is a higher likelihood of missing the subject. This would be a general trend with all active camera systems. Later work described by Senior et al. [51] simplified and advanced the calibration procedure and was demonstrated outdoors.

Bagdanov et al. [2] have developed a method for capturing facial images over a wide area with a single active pan-tilt camera. In this approach, reinforcement learning is employed to discover the camera control actions that maximize the chance of acquiring a frontal face image given the current appearance of object motion as seen from the camera in its home position. Essentially, the system monitors object motion with the camera in its home position and over time learns *if, when and where* to zoom in for a facial image. A frontal face detector applied after each attempt guides the learning. Some benefits are that this approach uses a single camera and requires no camera calibration.

Prince [41, 42], Elder [19] et al. have developed a system that addresses collection and pose challenges. They make use of a foveated sensor using a stationary camera with a 135° field of view and a foveal camera with a 13° field of view. Faces are detected via the *stationary* camera video using motion detection, background modeling and skin detection. A pan and tilt controller directs the *foveal* camera to detected faces. Since the system detects people based on face detection it naturally handles partially occluded and stationary people.

Davis et al. [16, 17] have developed methods to automatically scan a wide area and detect persons with a single PTZ camera. Their approach detects human behavior and learns the frequency with which humans appear across the entire coverage



**Fig. 14.2** Multi-camera person tracking with crowding (*above*) and person tracking and active NFOV face capture (*below*) (© 2009 IEEE, used with permission [61])

region so that scanning is done efficiently. Such a strategy might be used to create a one-camera active face capture system, or to greatly increase the coverage area if used as a WFOV camera.

Krahnstover et al. [25] have developed a face capture at a distance framework and prototype system. Four fixed cameras with overlapping viewpoints are used to pervasively track multiple subjects in a 10 m by 30 m region. Tracking is done in a real-world coordinate frame, which drives the targeting and control of four separate PTZ cameras that surround the monitored region. The PTZ cameras are scheduled and controlled to capture high-resolution facial images, which are then associated with tracker IDs. An optimization procedure schedules target assignments for the PTZ cameras with the goal of maximizing the number of facial images captured while maximizing facial image quality. This calculation is based on the apparent subject pose angle and distance. This opportunistic strategy tends to capture facial images when subjects are facing one of the PTZ cameras.

Bellotto et al. [5] describe an architecture for active multi-camera surveillance and face capture where trackers associated with each camera, and high-level reasoning algorithms communicate via an SQL database. Information from persons detected by WFOV trackers can be used to assign actively controlled NFOV cameras to particular subjects. Once NFOV cameras are viewing a face, the face is tracked and the NFOV camera follows the face with a velocity control system.

In Yu et al. [61], the authors have used this system [25] to monitor groups of people over time, associate an identity with each tracked person and record the degree of close interaction between identified individuals. Figure 14.2 shows person tracking and face capture from this system. This allows for the construction of a social

network that captures the interactions, relationships and leadership structure of the subjects under surveillance.

#### 14.1.7.3 NFOV Resource Allocation

Face capture with active cameras is faced with the problem of resource allocation. Given a limited number of NFOV cameras and a large number of potential targets, it becomes necessary to predict feasible periods of time in the future, during which a target could be captured by a NFOV camera at the desired resolution and pose, followed by scheduling the NFOV cameras based on these feasible temporal windows. Lim et al. [27, 28] address the former problem by constructing what is known as a “Task Visibility Interval” that encapsulates the required information. For the latter, these authors then utilize these “Task Visibility Intervals” to schedule NFOV camera assignments.

Bimbo and Pernici [6] have addressed the NFOV scheduling problem for capturing face images with an active camera network. They formulate the problem as a Kinetic Traveling Salesman Problem (KTSP) to determine how to acquire as many targets as possible.

A variety of NFOV scheduling policies have been developed and evaluated by Costello et al. [15] as well.

Qureshi and Terzopoulos [43–45] have developed an extensive virtual environment simulator for a large train station with behaviorally realistic autonomous pedestrians who move about without colliding, and carry out tasks such as waiting in line, buying tickets, purchasing food and drinks, waiting for trains and proceeding to the concourse area. The video-rendering engine handles occlusions, and models camera jitter and imperfect color response. The purpose is to develop and test active camera control and scheduling systems with many WFOV cameras and many NFOV cameras on a scale where real-world experiments would be prohibitively expensive. With such a simulator, visual appearance will never be perfect. However, this system allows the setup and evaluation of person tracking tasks and camera scheduling algorithms with dozens of subjects and cameras over a very large area with perfect ground truth. Then, the exact same scenario can be executed again with a change in any algorithm or aspect of the camera set-up.

#### 14.1.7.4 Very Long Distances

Yao et al. [58, 60] have explored face recognition at considerable distances, using their UTK-LRHM face database. For indoor data, with a gallery of 55 persons and a commercial face recognition system, they show a decline in recognition rate from 65.5% to 47.3% as the zoom factor goes from 1 to 20 and the subject distance is increased to maintain an eye-to-eye image resolution of 60 pixels. It is also shown that the recognition rate at a zoom factor of 20 can be raised back up to 65.5% with wavelet-based deblurring.

Yao et al. [59, 60] have used a super-resolution approach based on frequency domain registration and cubic-spline interpolation on facial images from the UTK-LRHM face database [58] and found considerable benefit in some circumstances. Super-resolution seems most effective when facial images begin at a low-resolution. For facial images with about 35 pixels eye-to-eye, super-resolution increased the recognition rate from 10% to 30% with a 55-person gallery. Super-resolution followed by unsharp masking further increased recognition rate to 38% and yielded a cumulative match characteristic performance almost as high as when optical zoom alone was used to double the facial image resolution.

#### 14.1.7.5 3D Imaging

Most 3D face capture systems use the stereo or structured light approach [9]. Stereo capture systems use two cameras with a known geometric relationship. The distance to feature points detected in each camera's image is then found via triangulation. Structured light systems use a light projector and a camera, also with a known geometric relationship. The light pattern is detected in the camera's image and 3D points are determined. Each system is characterized by a *baseline* distance, between the stereo cameras or between the light projector and camera. With either approach, the accuracy of the triangulated 3D data degrades with subject distance if the baseline distance is held constant. To maintain 3D reconstruction accuracy as subject distance increases, the baseline distance must be increased proportionally. This prohibits a physically compact system and is a fundamental challenge to 3D face capture at a distance with these methods. However, there are some newer systems under development that are overcoming the baseline challenge for 3D face capture at a distance. Below we review a few 3D face capture systems designed to operate at large distances.

Medioni et al. [35–37] have addressed FRAD for noncooperative individuals with a single camera approach and 3D face reconstruction. They propose a system where an ultra-high resolution 3048 by 4560 pixel camera is used by switching readout modes. Bandwidth limitations generally prevent the readout of the full resolution of such cameras at 30 Hz. However, full-frame low-resolution fast-frame-rate readouts can be used for person detection and tracking and partial-frame high-resolution readouts can be used to acquire a series of facial images of a detected and tracked person. Person detection is accomplished without background modeling, using an edgelet feature-based detector. This work emphasizes the 3D reconstruction of faces with 100 pixels eye-to-eye using shape from motion on data acquired with a prototype of the envisioned system. 3D reconstructions are performed at distances of up to 9 m. Though current experiments show 2D face recognition outperforming 3D, the information may be fused, or the 3D data may enable pose correction.

Rara et al. [46, 47] acquire 3D facial shape information at distances up to 33 m using a stereo camera pair with a baseline of 1.76 m. An Active Appearance Model localizes facial landmarks from each view and triangulation yields 3D landmark positions. The authors can achieve a 100% recognition rate at 15 m, though the gallery

size is 30 subjects and the collection environment is cooperative and controlled. It is noted that depth information at such long distances with this modest baseline can be quite noisy and may not significantly contribute to recognition accuracy.

Redman et al. [48] and colleagues at Lockheed Martin Coherent Technologies have developed a 3D face imaging system for biometrics using Fourier Transform Profilometry. This involves projecting a sinusoidal fringe pattern onto the subject's face using short eye-safe laser bursts and imaging the illuminated subject with a camera that is offset laterally from the light source. Fourier domain processing of the image can recover a detailed 3D image. In a sense, this falls into the class of structured light approaches, but with a small baseline requirement. A current test system is reported to capture a 3D facial image at 20 m subject distance with a range error standard deviation of about 0.5 mm and a baseline distance of only 1.1 m.

Redman et al. [49] have also developed 3D face imaging systems based on digital holography, with both multiple-source and multiple wavelength configurations. With multiple-wavelength holography, a subject is imaged two or more times, each time illuminated with a laser tuned to a different wavelength, in the vicinity of 1617 nm for this system. The laser illumination is split to create a reference beam, which is mixed with the received beam. The interference between these beams is the hologram that is imaged by the sensor. The holograms at each wavelength are processed to generate the 3D image. The multi-wavelength holographic system has been shown to capture a 3D facial image at a 100 m subject distance with a range error of about 1–2 mm, though this has been performed in a lab setting and not with live subjects. With this approach there is zero baseline distance. The only dependence of accuracy on subject distance is atmospheric transmission loss.

Andersen et al. [1] have also developed a 3D laser radar and applied it to 3D facial image capture. This approach uses a time-of-flight strategy to range measurement, with a rapidly pulsed (32.4 kHz) green Nd:YAG laser and precisely timed camera shutter. 50–100 reflectivity images are captured and processed to produce a 3D image. This system has been used to capture 3D facial images at distances up to 485 m. Each 3D image capture takes a few seconds. Though at this stage not many samples have been collected, the RMS range error is about 2 mm at 100 m subject distance and about 5 mm at 485 m subject distance. Atmospheric turbulence, vibrations and system errors are the factors that limit the range of this system.

The Fourier Transform Profilometry and Digital Holography approaches operate at large distance, but do not naturally handle a large capture region. Coupled with a WFOV video camera and person detection and tracking system, these systems could be used to capture 3D facial images over a wide area.

#### 14.1.7.6 Face and Gait Fusion

For recognition at a distance, face and gait are a natural pair. In most situations, a sensor used for FRAD will also be acquiring video suitable for gait analysis. Liu et al. [31] exploit this and develop fusion algorithms to show a significant improvement in verification performance by using multi-modal fusion gait and face recognition with facial images collected outdoors at a modest standoff distance.



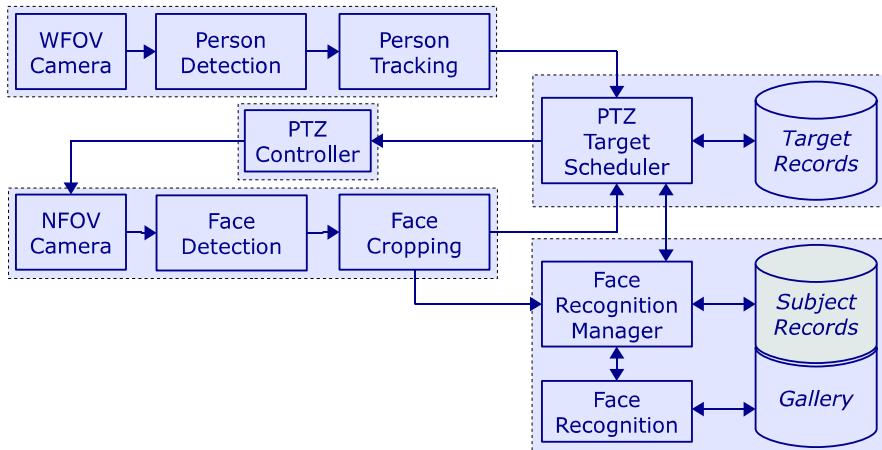
**Fig. 14.3** The Biometric Surveillance System, a portable test and demonstration system on a wheeled cart with two raised camera nodes (*left*), and a close-up view of one node (*right*)

Zhou et al. [62, 63] have recognized that the best gait information comes from profile views, and have thus focused on fusing gait information with profile facial images. In initial work [62], face recognition was done using curvature features of just the face profile. Later efforts [63] use the whole side-view of the face and also enhanced the resolution of the side-view using multi-frame super-resolution, motivated to use all available information. For 45 subjects imaged at a distance of 10 ft, the best recognition rates are 73.3% for single-frame face, 91.1% for multi-frame enhanced face, 93.3% for gait, and 97.8% for fused face and gait. Facial profile super-resolution gives a considerable improvement, as does fusion.

## 14.2 Face Capture at a Distance

GE Global Research and Lockheed Martin have developed a FRAD system called the Biometric Surveillance System [56]. The system features reliable ground-plane tracking of subjects, predictive targeting, a target priority scoring system, interfaces to multiple commercial face recognition systems, many configurable operating modes, an auto-enrollment mechanism, and network-based sharing of auto-enrollment data for re-identification. Information about tracking, target scoring, target selection, target status, attempted recognition, successful recognition, and enrollments are displayed in a highly animated user interface (Fig. 14.1 on page 354).

The system uses one or more networked nodes where each node has a co-located WFOV and NFOV camera (Fig. 14.3). Each camera is a Sony EVI-HD1, which features several video resolution and format modes and integrated pan, tilt, zoom and focus, all controllable via a VISCA™ serial interface. The WFOV camera is operated in NTSC video mode, producing 640 by 480 video frames at 30 Hz. The pan, tilt and zoom settings of the WFOV camera are held fixed. The NFOV camera is configured for 1280 by 720 resolution video at 30 Hz, and its pan, tilt and zoom setting are actively controlled. Matrox® frame grabbers are used to transfer video streams to a high-end but standard workstation.



**Fig. 14.4** System diagram showing main computational components of the Biometric Surveillance System

A system diagram is shown in Fig. 14.4. The stationary WFOV camera is used to detect and track people in its field of view. The WFOV camera is calibrated to determine its internal and external parameters, which include the focal length, principal point, location, and orientation. This defines a mapping between real-world metric coordinates and the WFOV camera image. Since the camera is stationary, a background subtraction approach is used for moving object detection. The variation of each color component of each pixel is learned and adapted using a non-parametric distribution. Grayscale imagery may be used as well, but color increases detection rate. Whenever a pixel does not match this model, it is declared a foreground pixel. From the camera calibration information and the assumption that people are walking upright on the ground plane, feasible sizes and locations of persons within the image plane are established. Blobs of foreground pixels that conform to these feasible sizes are detected persons. A ground-plane tracker based on an extended Kalman filter is applied to detected persons [7, 24]. The use of the Kalman filter makes the tracker robust to intermittent occlusions and provides the velocity, travel direction and predicted locations of subjects [54].

The automatically controlled NFOV camera is also calibrated with respect to the real-world coordinate system, when it is in its home position with its pan and tilt angles at  $0^\circ$  and its zoom factor set to 1. Further calibration of the NFOV camera determines how pan, tilt and zoom settings affect its field of view. An important part of this calibration is the camera's *zoom point*, or the pixel location that always points to the same real-world point as the zoom factor is changed. The zoom point is not necessarily the exact center of the image, and even a small offset can affect targeting accuracy when a high zoom is used for distant subjects. Effectively, this collective calibration data allows for the specification of a region in the WFOV image, and determines the pan, tilt and zoom settings to make that region the full image for the NFOV camera.

**Table 14.1** The factor and clipping range used for each parameter to score targets

| Parameter        | Factor | Clipping Range |
|------------------|--------|----------------|
| Direction cosine | 10     | [−8, 8]        |
| Speed (m/s)      | 10     | [0, 20]        |
| Capture attempts | −2     | [−5, 0]        |
| Face captures    | −1     | [−5, 0]        |
| Times recognized | −5     | [−15, 0]       |

### 14.2.1 Target Selection

When multiple persons are present, the system must determine which subject to target for high-resolution face capture. From the WFOV person tracker, it is straightforward to determine the distance to a subject, the degree to which a subject is facing (or at least moving toward) the cameras, and the speed of the subject. Further, because the person tracker is generally quite reliable, a record can be kept for each tracked subject. This subject record includes the number of times we have targeted the subject, the number of times we have successfully captured a facial image and the number of times the subject has been successfully identified by the face recognition algorithm. All of this information is used by the target selection mechanism.

Detected and tracked persons are selected for high-resolution facial capture based on a priority scoring mechanism. A score is produced for each tracked subject, and the subject with the highest score is selected as the next target. Several parameters are used in the scoring process, and for each parameter, a multiplicative factor is applied and the result is clipped to a certain range. For example, the subject's speed in m/s is multiplied by the factor 10.0, clipped to the range [0, 20] and added to the score. Table 14.1 shows the complete set of parameters and factors currently in use, though not yet optimized. The direction cosine parameter is the cosine of the angle between the subject's direction of travel and the line from the subject to the NFOV camera. This parameter indicates the degree to which the subject is facing the NFOV camera. The net overall effect of this process is to favor subjects moving more quickly toward the cameras who have not yet been satisfactorily imaged. In practice, a target selection strategy like this causes the system to move from subject to subject, with a tendency to target subjects from which we are most likely to get new and useful facial images.

When a subject is selected, the system uses the Kalman filter tracker to predict the location of the subject's face at a specific target time about 0.5–1.0 s in the future. The NFOV camera will point to this location and hold until the target time has passed. This gives the camera time to complete the pan and tilt change, and time for vibration to settle. Facial images are captured when the subject moves through the narrow field-of-view as predicted. We have already discussed the trade-off between zoom factor and probability of successfully capturing a facial image. This system uses an adaptive approach. If there have been no face captures for the subject, then the initial face resolution goal will be a modest 30 pixels eye-to-eye. However, each

time a facial image is successfully captured at a particular resolution, the resolution goal is increased by 20%. Each subject tends to be targeted and imaged many times by the system, so the facial image resolution goal rapidly increases. For a particular target, this resolution goal and the subject distance determines the zoom factor of the NFOV camera. The subject distance is also used to set the focus distance of the NFOV camera.

### ***14.2.2 Recognition***

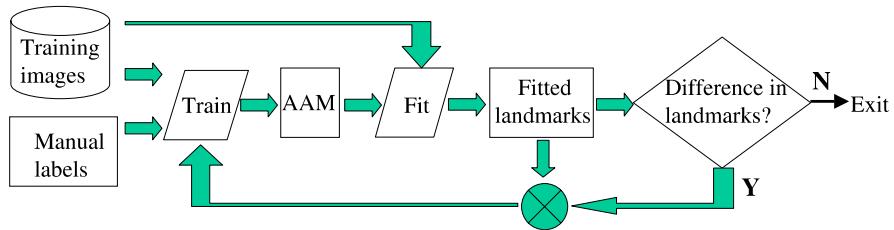
The NFOV video is processed on a per-frame basis. In each frame, the Pittsburgh Pattern Recognition FT SDK is utilized to detect faces. If there is more than one detection, we use only the most central face in the image, since it is more likely to be the face of the targeted subject. A detected face is cropped from the full frame image and passed to the face recognition manager. The target scheduler is also informed of the face capture, so the subject record can be updated.

When the face recognition manager receives a new facial image, a facial image capture record is created and the image is stored. Facial images can be captured by the system at up to about 20 Hz, but face recognition generally takes 0.5–2 s per image, depending on the algorithm. Recognition cannot keep up with capture, so the face recognition algorithm is operated asynchronously. The system can be interfaced to Cognitec FaceVACS®, Identix FaceIt®, Pittsburgh Pattern Recognition FTR, or an internal research face recognition system. In a processing loop, the face recognizer is repeatedly applied to the most recently captured facial image not yet processed, and results are stored in the facial image capture record. Face recognition can use a stored gallery of images, manual enrollments, automatic enrollments or any combination.

The face recognition manager queries the target scheduler to determine which tracker subject ID a facial image came from, based on the capture time of the image. With this information, subject records are created, keyed by the tracker ID number, and the facial image capture records are associated with them.

The auto-enrollment feature of this system makes use of these subject records. This is a highly configurable rule-based process. A typical rule is that a subject is an auto-enroll candidate if one face capture has a quality score exceeding a threshold, one face capture has a face detection threshold exceeding a threshold, recognition has been attempted at least 4 times and has never succeeded, and the most recent face capture was at least 4 seconds ago. If a subject is an auto-enroll candidate, the facial image with the highest quality score is selected and enrolled in the face recognition gallery, possibly after an optional user confirmation.

In indoor and outdoor trials, the capabilities of the system have been evaluated [56]. Test subjects walked in the vicinity of the system in an area where up to about 8 other nonsubjects were also walking in view. An operator recorded the subject distance at the first person detection, first face capture and first successful



**Fig. 14.5** Diagram of the AAM enhancement scheme (© 2006 Xiaoming Liu, et al., used with permission [33])

face recognition with a gallery of 262 subjects. In this experiment, the mean distance to initial person detection was 37 m, the mean distance to initial facial image capture was 34 m and the mean distance to recognition was 17 m.

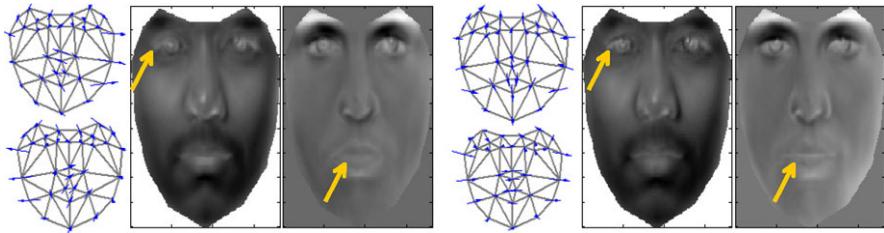
### 14.3 Low-Resolution Facial Model Fitting

Face alignment is a process of overlaying a deformable template on a face image to obtain the locations of facial features. Being an active research topic for over two decades [12], face alignment has many applications, such as face recognition, expression analysis, face tracking and animation, etc. Within the considerable prior work on face alignment, *Active Appearance Models (AAMs)* [14] have been one of the most popular approaches. However, the majority of existing work focuses on fitting the AAM to facial images with moderate to high quality [26, 29, 30, 57]. With the popularity of surveillance cameras and greater needs for FRAD, methods to effectively fit an AAM to low-resolution facial images are of increasing importance. This section addresses this particular problem and presents our solutions for it.

Little work has been done in fitting AAMs to low-resolution images. Cootes et al. [13] proposed a multi-resolution Active Shape Model. Dedeoglu et al. [18] proposed integrating the image formulation process into the AAM fitting scheme. During the fitting, both image formulation parameters and model parameters are estimated in a united framework. The authors also showed the improvement of their method compared to fitting with a single high-resolution AAM. We will show that as an alternative fitting strategy, a multi-resolution AAM has far better fitting performance than a high-resolution AAM.

#### 14.3.1 Face Model Enhancement

One requirement for AAM training is to manually position the facial landmarks for all training images. This is a time-consuming and error-prone operation, which certainly affects face modeling. To tackle the problem of labeling error, we develop an AAM enhancement scheme (see Fig. 14.5). Starting with a set of training images



**Fig. 14.6** The 6th and 7th shape basis and the 1st and 4th appearance basis before (*left*) and after enhancement (*right*). After enhancement, more symmetric shape variation is observed, and certain facial areas appear sharper (© 2006 Xiaoming Liu, et al., used with permission [33])

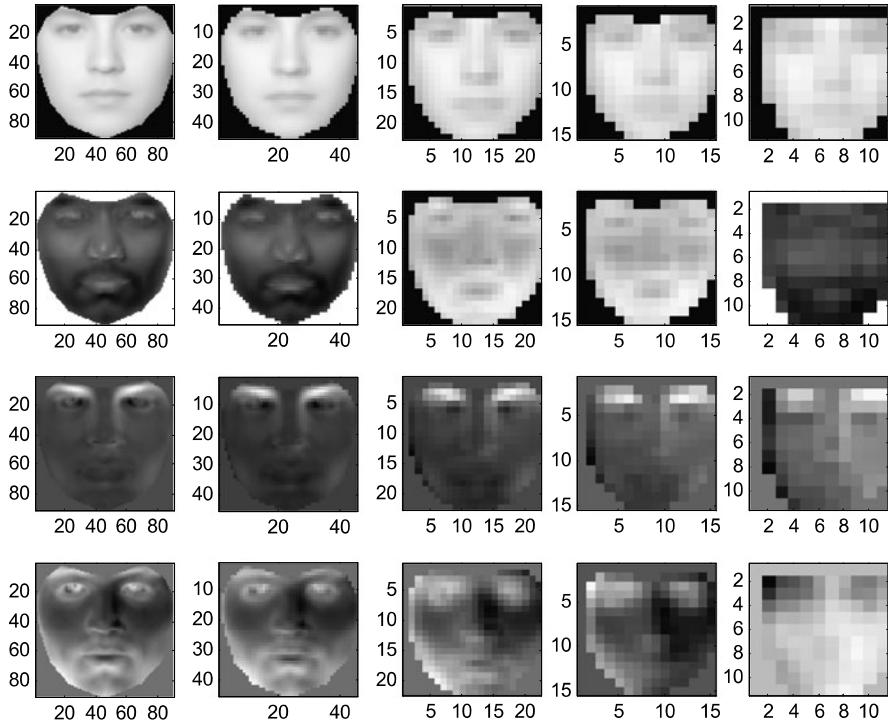
and manual labels, an AAM is trained using the above method. Then the AAM is fit to the same training images using the Simultaneous Inverse Compositional (SIC) algorithm, where the manual labels are used as the initial location for fitting. This fitting yields new landmark positions for the training images. This process is iterated. This new landmark set is used for face modeling again, followed by model fitting using the new AAM. The iteration continues until there is no significant difference between the landmark locations of the consecutive iterations. In the face modeling of each iteration, the basis vectors for both the appearance and shape models are chosen such that 98% and 99% of the energy are preserved, respectively.

With the refined landmark locations, the resulting AAM is improved as well. As shown in Fig. 14.6, the variation of landmarks around the outer boundary of the cheek becomes more symmetric after enhancement. Also, certain facial areas, such as the left eye boundary of the 1st appearance basis and the lips of 4th appearance basis, are visually sharper after enhancement, because the training images are better aligned thanks to improved landmark location accuracy.

Another benefit of this enhancement is improved compactness of the face model. In our experiments, the numbers of appearance and shape basis vectors reduce from 220 and 50 to 173 and 14, respectively. There are at least two benefits of a more compact AAM. One is that fewer shape and appearance parameters need to be estimated during model fitting. Thus the minimization process is less likely to become trapped in a local minimum, and fitting robustness is improved. The other is that model fitting can be performed faster because the computation cost directly depends on the dimensionality of the shape and appearance models.

### 14.3.2 Multi-Resolution AAM

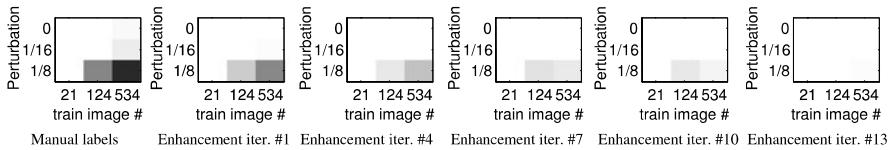
The traditional AAM algorithm makes no distinction with respect to the resolution of the test images being fit. Normally the AAM is trained using the full resolution of the training images, which is called a *high-resolution AAM*. When fitting a high-resolution AAM to a low-resolution image, an up-sampling step is involved in interpolating the observed image and generating a warped input image,  $I(\mathbf{W}(\mathbf{x}; \mathbf{P}))$ . This



**Fig. 14.7** The appearance models of a multi-res AAM: Each column shows the mean and first 3 basis vectors at relative resolutions 1/2, 1/4, 1/8, 1/12 and 1/16 respectively (© 2006 Xiaoming Liu, et al., used with permission [33])

can cause problems because a high-resolution AAM has high frequency components that a low-resolution image does not contain. Thus, even with perfect estimation of the model parameters, the warped image will always have high frequency residual with respect to the high-resolution model instance, which, at a certain point, will overwhelm the residual due to the model parameter errors. Hence, fitting becomes problematic.

The basic idea of applying multi-resolution modeling to AAM is straightforward. Given a set of facial images, we down-sample them into low-resolution images at multiple scales. We then train an AAM using the down-sampled images at each resolution. We call the pyramid of AAMs a *multi-res AAM*. For example, Fig. 14.7 shows the appearance models of a multi-res AAM at relative resolutions 1/2, 1/4, 1/8, 1/12 and 1/16. Comparing the AAMs at different resolutions, we can see that the AAMs at lower resolutions have more blurring than the AAMs at higher resolutions. Also, the AAMs at lower resolutions have fewer appearance basis vectors compared to the AAMs at higher resolutions, which will benefit the fitting. The landmarks used for training the AAM for the highest resolution are obtained using the enhancement scheme above. The mean shapes of a multi-res AAM differ



**Fig. 14.8** The convergence rate of fitting using an AAM trained from manual labels, and AAM after enhancement iteration number 1, 4, 7, 10 and 13. The brightness of the block is proportional to the convergence rate. Continuing improvement of fitting performance is observed as the enhancement process progresses (© 2006 Xiaoming Liu, et al., used with permission [33])

only by a scaling factor, while the shape basis vectors from different scales of the multiple-resolution AAM are exactly the same.

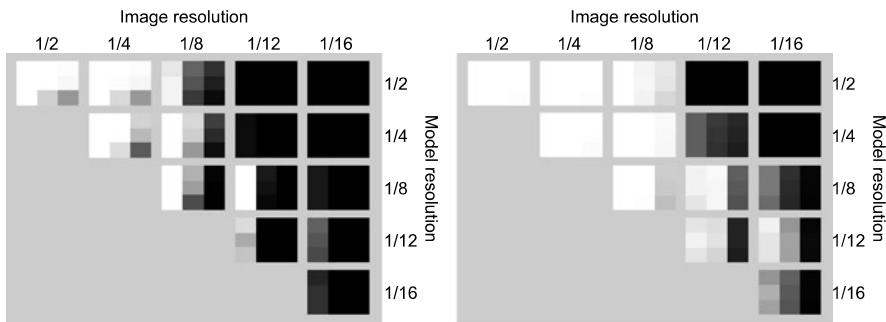
### 14.3.3 Experiments

Our experiments are conducted on a subset of the ND1 face database [10], which contains 534 images from 200 subjects. Regarding the fitting performance measurement, we use the convergence rate (CR) with respect to different levels of perturbation on the initial landmark locations. The fitting is converged if the average mean squared error between the estimated landmarks and the ground-truth is less than a threshold. Given the true landmarks of one image, we randomly deviate each landmark within a rectangular area up to a certain range, and the projection of the perturbed landmarks in the shape model is used as the initial shape parameters. Three different perturbation ranges,  $R$ , are used: 0, 1/16, and 1/8 of the facial height.

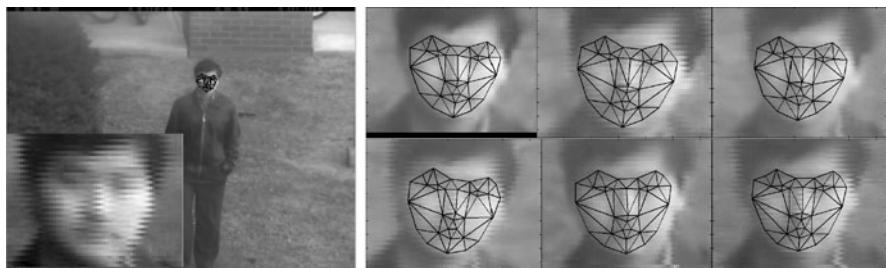
Another varying factor is the number of images/subjects in the training set. When multiple images of one subject are used for training an AAM, the resulting AAM is considered as a person-specific AAM. When the number of subjects in the training set is large, the resultant AAM is a generic AAM. The more subjects used, the more generic the AAM is. Using the ND1 database, we test the modeling with three different population sizes, where the numbers of images are 21, 124, 534, and the corresponding numbers of subjects are 5, 25, 200, respectively.

Figure 14.8 shows the CR of AAM fitting after a varying number of model enhancement iterations. The leftmost plot shows the CR using an AAM trained from manual labels only, with varying population size and perturbation window size. Each element represents the CR, which is computed using the same training set as test images. There are some non-converged cases when more generic models are used with a larger perturbation window size. The rest of the plots show the CR using the AAM trained after 1, 4, 7, 10 and 13 iterations of the enhancement algorithm. Continuing improvement of fitting performance is observed with additional enhancement iterations. After the model enhancement is completed, the fitting process converges for all testing cases, no matter how generic the model or how large the perturbation of the initialization.

The second experiment is to test the fitting performance of a multi-res AAM on images with different resolutions. The same dataset and test scheme are used as in



**Fig. 14.9** The convergence rate of fitting a multi-res AAM trained with manual labels (*left*) and enhanced landmarks (*right*) to images with different resolution. Each 3 by 3 block has the same axes as in Fig. 14.8 (© 2006 Xiaoming Liu, et al., used with permission [33])



**Fig. 14.10** Fitting a multi-res AAM to an outdoor surveillance video: One sample frame with zoom in facial area (*left*) and six zoom in frames overlaid with fitting results (*right*), (© 2006 Xiaoming Liu, et al., used with permission [33])

the previous experiment, except that the different resolutions of down-sampled training images are also used as test images for fitting. Model fitting is conducted with all combinations of AAM resolution and image resolution, where the model resolution no less than the image resolution. As shown in Fig. 14.9, the AAM trained with enhanced landmarks performs much better than the AAM trained from manual labels. Also, for low-resolution images, the best fitting performance is obtained when the model resolution is slightly higher than the facial image resolution, which is far better than fitting using the AAM with the highest model resolution. This shows that the additional appearance detail in the higher resolution AAM seems to confuse the minimization process and results in degraded fitting performance.

The last experiment is model fitting on a surveillance video captured using a PTZ camera positioned about 20 m from the subject. Sample fitting results using a multi-res AAM are shown in Fig. 14.10. Although the frame size is 480 by 640 pixels, the facial area is not only at a low resolution, but also suffers from strong blurring, specular, and interlacing effects, which makes fitting a very challenging task. Our multi-res AAM continuously fits around 100 frames and provides reasonable results.

However, the high-resolution AAM only successfully fits the first 4 frames in this sequence.

## 14.4 Facial Image Super-Resolution

In situations where lack of image resolution is an impediment to recognition, multi-frame image super-resolution can be a differentiator that makes FRAD possible. In typical FRAD systems facial images are captured with a video camera, and many images of the face are recorded over a short period of time. While conventional face recognition systems operate on a single image, it is desirable to improve recognition performance by using all available image data. There are a few approaches that may be taken. In this section we will describe our super-resolution approach [55], Chap. 13 covers the direct use of video for recognition, and multi-sample fusion [50] is another viable approach.

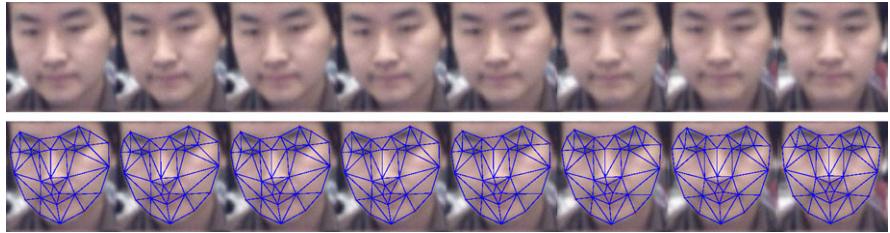
Super-resolution is the process of producing one high-resolution image from multiple low-resolution images of the same object or scene [8, 11, 32]. Resolution improvement can come from dealiasing, deblurring and noise reduction. A key aspect of super-resolution processing is the exploitation of the fact that the object or camera moves between video frames, so that image pixels in different frames have sub-pixel offsets and thus contain new information.

In this section, we describe a method for the super-resolution of faces from video. Super-resolution algorithms generally have two parts: frame-to-frame registration, and the super-resolution image processing itself. In our method, registration is accomplished with an Active Appearance Model (AAM) designed specifically for the shape of the face and its motion [33]. To solve for the super-resolved image, we define and optimize a cost function with an  $L_1$  data fidelity component and a Bilateral Total Variation (BTV) regularization term, as described by Farsiu [20].

Most image super-resolution algorithms use a parameterized whole-image transformation for registration, such as a homography or rigid translation [23]. This is suitable when the scene is planar or the perspective distortion due to depth and camera motion is insignificant. Facial images have been super-resolved quite dramatically by Baker and Kanade [4], but the motion model used is translation-only and does not account for 3D face shape, effectively assuming that the subject is always facing the camera. To deal with the nonrigid motion of moving faces, optical flow has been used for the registration step [3]. While optical flow certainly can track facial motion, it is computationally complex and its generality brings the risk of overfitting. The super-resolution approach by Mortazavian et al. [38], like that described here, also uses a facial model fitting approach for registration.

### 14.4.1 Registration and Super-Resolution

Given video of a subject we fit an AAM [33] to the face in each frame. The AAM defines 33 landmark positions that are the vertices of 49 triangles over the face as



**Fig. 14.11** Faces from 8 video frames showing the fitted AAM shape model. The fitted AAM will allow frame-to-frame registration even as the face rotates (© 2009 IEEE, used with permission [55])

seen in Fig. 14.11. The registration of the face between any two frames is then a piecewise affine transformation, with an affine transformation for each triangle defined by the corresponding vertices. A set of about  $N = 10$  consecutive frames are combined to produce the super-resolved image.

We will describe the super-resolution algorithm using linear algebra notation, as if each image has its pixel values in a single vector. The computation is actually carried out with operations on 2D pixel arrays. The super-resolution process uses an image formation model relating each of the input frames  $Y_i$ , to an unknown super-resolution image,  $X$ , with twice the pixel resolution. The image formation process accounts for the face motion, camera blur and detector sampling. For each input frame,  $F_i$  is the registration operator that warps  $X$  to be aligned with  $Y_i$ , but at twice the resolution. The camera blur operator,  $H$ , applies the Point Spread Function (PSF). For most installed surveillance cameras it is difficult to determine the true PSF, so we assume a Gaussian shaped PSF with hand selected width,  $\sigma$ . Finally, the sampling operation of the detector is represented by the sparse matrix  $D$  that extracts every other pixel in each dimension, yielding an image that should match our real observed image. If we let  $V_i$  represent additive pixel intensity noise, the complete linear image formation process is then,

$$Y_i = DH F_i X + V_i.$$

This is the forward model of our observed low-resolution images  $Y_i$  given the unknown high-resolution image  $X$ . Our goal is to solve the inverse problem to recover  $X$ .

The super-resolved image  $X$  is produced by optimizing a cost function that is the  $L_1$  norm of the difference between the model of the observations and the actual observations, plus a regularization term,  $\Psi(X)$ ,

$$\hat{X} = \operatorname{argmin}_X \left[ \sum_{i=1}^N \|DH F_i X - Y_i\|_1 + \lambda \Psi(X) \right].$$

The  $L_1$  norm is used in the data fidelity portion of the cost function for robustness against incorrect modeling assumptions and registration errors. For the regulariza-

tion term, we use Bilateral Total Variation (BTV) [20],

$$\Psi(X) = \sum_{l=-P}^P \sum_{m=-P}^P \alpha^{|m|+|l|} \|X - S_x^l S_y^m X\|_1.$$

Here  $S_x^l$  and  $S_y^m$  are operators that shift the image in the  $x$  and  $y$  direction by  $l$  and  $m$  pixels. With BTV, the neighborhood over which absolute pixel difference constraints are applied can be larger (with  $P > 1$ ) than for Total Variation (TV). The size of the neighborhood is controlled by parameter  $P$  and the constraint strength decay is controlled by  $\alpha \in (0, 1)$ .  $L_1$ -based regularization methods such as BTV or TV are selected to preserve edges. By contrast,  $L_2$ -based Tikhonov regularization is effectively a smoothness constraint, which is counter to our goal of increased resolution.

To solve for the super-resolution image,  $X$  is first initialized using straightforward warping and averaging. A steepest descent search using the analytic gradient of the cost function is used [55]. With the original frames normalized to a pixel range of  $[0, 1]$ , we have found that setting the regularization strength parameter  $\lambda$  to 0.025 gives the best visual results and we use that value for the experiments presented here.

#### 14.4.2 Results

Figure 14.12 shows sample super-resolution results, including: (a) the face from the original video frame; (b) that single frame restored with a Wiener filter; and (c) the result of multi-frame super-resolution using  $N = 10$  consecutive frames. The increase in sharpness and clarity is visually apparent. In another experiment with a gallery of 700 images from unique subjects, we tested 138 facial video clips collected from three individuals with surveillance cameras that had an eye-to-eye distance ranging from 17 to 48 pixels. With this challenging data collected at about 10 m, super-resolution processing instead of using just single frames increased the rank-1 recognition rate from 50% to 56%.

#### 14.5 Conclusions

Face recognition at a distance is a challenging problem with a large number of beneficial applications. We have reviewed the primary challenges, approaches and research literature on this topic, and we have described some specific work that we have carried out to create a prototype FRAD system, fit alignment models to faces at very low resolutions and super-resolve facial images. Still, there are a great many open issues that may lead to enhanced functionality or new applications. We conclude this chapter by highlighting a number of these potential future avenues of research.



**Fig. 14.12** Example original video frames, Wiener filter results, and super-resolution results with enlarged views of the left eye. In the Wiener filter results, artifacts in the face region are primarily due to enhancement of interlacing artifacts. Some ringing due to circular convolution is present, but only near the image edges. The increased resolution and clarity in the super-resolution results is clearly visible (© 2009 IEEE, used with permission [55])

Commercially available face recognition systems are typically designed and trained for access control applications with high-quality facial images. The need for facial recognition algorithms that work well under FRAD imaging conditions is well understood, and this is an active research area. It will be key to understand which facial features are present and absent in facial images collected at a distance so that recognition algorithms can focus on those that remain. If face recognition can be performed fast enough, then immediate recognition results, or even face quality analysis can be utilized more actively for NFOV resource allocation and active capture control loops. The use of incremental fusion of face recognition results during rapid video capture of faces may also make active capture systems more efficient.

One of the challenges of FRAD is subject pose. Strategies to attract the attention of subjects to certain locations near cameras may help this issue in some situations. In all of the active vision systems, we have discussed both the WFOV and NFOV cameras are stationary. The use of cameras on movable platforms, such as guide wires or robots could enable much more effective facial image collection and open a new surveillance and biometric identification paradigm.

**Acknowledgements** Section 14.2 of this report was prepared by GE Global Research as an account of work sponsored by Lockheed Martin Corporation. Information contained in this report constitutes technical information which is the property of Lockheed Martin Corporation. Neither

GE nor Lockheed Martin Corporation, nor any person acting on behalf of either; a. Makes any warranty or representation, expressed or implied, with respect to the use of any information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or b. Assume any liabilities with respect to the use of, or for damages resulting from the use of, any information, apparatus, method, or process disclosed in this report. Sections 14.3 and 14.4 were supported in part by award #2005-IJ-CX-K060 awarded by the National Institute of Justice, Office of Justice Programs, US Department of Justice. The opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the Department of Justice.

## References

1. Andersen, J.F., Busck, J., Heiselberg, H.: Long distance high accuracy 3-D laser radar and person identification. In: Kamerman, G.W. (ed.) *Laser Radar Technology and Applications X*, vol. 5791, pp. 9–16. SPIE, Bellingham (2005)
2. Bagdanov, A., Bimbo, A., Nunziati, W., Pernici, F.: Learning foveal sensing strategies in unconstrained surveillance environments. In: AVSS (2006)
3. Baker, S., Kanade, T.: Super resolution optical flow. Tech. Rep. CMU-RI-TR-99-36, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA (1999)
4. Baker, S., Kanade, T.: Limits on super-resolution and how to break them. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(9), 1167–1183 (2002)
5. Bellotto, N., Sommerlade, E., Benfold, B., Bibby, C., Reid, I., Roth, D., Fernández, C., Gool, L.V., González, J.: A distributed camera system for multi-resolution surveillance. In: Proc. of the ACM/IEEE Intl. Conf. on Distributed Smart Cameras (ICDSC) (2009)
6. Bimbo, A.D., Pernici, F.: Towards on-line saccade planning for high-resolution image sensing. *Pattern Recognit. Lett.* **27**(15), 1826–1834 (2006)
7. Blackman, S., Popoli, R.: *Design and Analysis of Modern Tracking Systems*. Artech House, Norwood (1999)
8. Borman, S.: Topics in multiframe superresolution restoration. Ph.D. thesis, University of Notre Dame, Notre Dame, IN (2004)
9. Bowyer, K.W., Chang, K., Flynn, P.: A survey of approaches and challenges in 3D and multimodal 3D + 2D face recognition. *Comput. Vis. Image Underst.* **101**(1), 1–15 (2006)
10. Chang, K., Bowyer, K., Flynn, P.: Face recognition using 2D and 3D facial data. In: Proc. ACM Workshop on Multimodal User Authentication, pp. 25–32 (2003)
11. Chaudhuri, S. (ed.): *Super-Resolution Imaging*, 3rd edn. Kluwer Academic, Dordrecht (2001)
12. Cootes, T., Cooper, D., Tylor, C., Graham, J.: A trainable method of parametric shape description. In: BMVC, pp. 54–61 (1991)
13. Cootes, T., Taylor, C., Lanitis, A.: Active shape models: Evaluation of a multi-resolution method for improving image search. In: BMVC, vol. 1, pp. 327–336 (1994)
14. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(6), 681–685 (2001)
15. Costello, C.J., Diehl, C.P., Banerjee, A., Fisher, H.: Scheduling an active camera to observe people. In: Proc. of the ACM Intl. Workshop on Video Surveillance and Sensor Networks, pp. 39–45 (2004)
16. Davis, J., Morison, A., Woods, D.: An adaptive focus-of-attention model for video surveillance and monitoring. *Mach. Vis. Appl.* **18**(1), 41–64 (2007)
17. Davis, J., Morison, A., Woods, D.: Building adaptive camera models for video surveillance. In: WACV (2007)
18. Dedeoglu, G., Baker, S., Kanade, T.: Resolution-aware fitting of active appearance models to low-resolution images. In: ECCV (2006)
19. Elder, J.H., Prince, S., Hou, Y., Sizintsev, M., Oleviskiy, Y.: Pre-attentive and attentive detection of humans in wide-field scenes. *Int. J. Comput. Vis.* **72**, 47–66 (2007)

20. Farsiu, S., Robinson, M.D., Elad, M., Milanfar, P.: Fast and robust multiframe super-resolution. *IEEE Trans. Image Process.* **13**(10), 1327–1344 (2004)
21. Greiffenhangen, M., Ramesh, V., Comaniciu, D., Niemann, H.: Statistical modeling and performance characterization of a real-time dual camera surveillance system. In: *CVPR* (2000)
22. Hampapur, A., Pankanti, S., Senior, A., Tian, Y.L., Brown, L., Bolle, R.: Face cataloger: multi-scale imaging for relating identity to location. In: *AVSS*, pp. 13–20 (2003)
23. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge (2000)
24. Krahnstoever, N., Tu, P., Sebastian, T., Perera, A., Collins, R.: Multi-view detection and tracking of travelers and luggage in mass transit environments. In: Proc. Ninth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS) (2006)
25. Krahnstoever, N., Yu, T., Lim, S.N., Patwardhan, K., Tu, P.: Collaborative real-time control of active cameras in large scale surveillance systems. In: Proc. Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (M2SFA2) (2008)
26. Liang, L., Wen, F., Xu, Y., Tang, X., Shum, H.: Accurate face alignment using shape constrained Markov network. In: *CVPR* (2006)
27. Lim, S.N., Davis, L.S., Mittal, A.: Constructing task visibility intervals for a surveillance system. *ACM Multimedia Systems Journal* **12**(3) (2006)
28. Lim, S.N., Davis, L., Mittal, A.: Task scheduling in large camera network. In: *ACCV* (2007)
29. Liu, X.: Discriminative face alignment. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(11), 1941–1954 (2009)
30. Liu, X.: Video-based face model fitting using adaptive active appearance model. *Image Vis. Comput.* **28**(7), 1162–1172 (2010)
31. Liu, Z., Sarkar, S.: Outdoor recognition at a distance by fusing gait and face. *Image Vis. Comput.* **25**(6), 817–832 (2007)
32. Liu, K.R., Kang, M.G., Chaudhuri, S. (eds.): *IEEE Signal Processing Magazine*, Special edition: Super-Resolution Image Reconstruction, vol. 20, no. 3. IEEE (2003)
33. Liu, X., Tu, P.H., Wheeler, F.W.: Face model fitting on low resolution images. In: *BMVC* (2006)
34. Marchesotti, L., Marcenaro, L., Regazzoni, C.: Dual camera system for face detection in unconstrained environments. In: *ICIP* (2003)
35. Medioni, G., Choi, J., Kuo, C.H., Choudhury, A., Zhang, L., Fidaleo, D.: Non-cooperative persons identification at a distance with 3D face modeling. In: *BTAS* (2007)
36. Medioni, G., Fidaleo, D., Choi, J., Zhang, L., Kuo, C.H., Kim, K.: Recognition of non-cooperative individuals at a distance with 3D face modeling. In: 2007 IEEE Workshop on Automatic Identification Advanced Technologies, pp. 112–117 (2007)
37. Medioni, G., Choi, J., Kuo, C.H., Fidaleo, D.: Identifying noncooperative subjects at a distance using face images and inferred three-dimensional face models. *IEEE Trans. Syst. Man Cybern., Part A, Syst. Hum.* **39**(1), 12–24 (2009)
38. Mortazavian, P., Kittler, J., Christmas, W.: A 3-D assisted generative model for facial texture super-resolution. In: *BTAS*, pp. 1–7 (2009)
39. NIST Multiple Biometric Grand Challenge. <http://face.nist.gov/mbgc>
40. O'Toole, A., Harms, J., Snow, S., Hurst, D., Pappas, M., Ayyad, J., Abdi, H.: A video database of moving faces and people. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(5), 812–816 (2005)
41. Prince, S., Elder, J., Hou, Y., Sizinstev, M., Olevsky, E.: Towards face recognition at a distance. In: Proc. of the IET Conf. on Crime and Security, pp. 570–575 (2006)
42. Prince, S., Elder, J., Warrell, J., Felisberti, F.: Tied factor analysis for face recognition across large pose differences. *IEEE Trans. Pattern Anal. Mach. Intell.* **30**(6), 970–984 (2008)
43. Qureshi, F., Terzopoulos, D.: Surveillance in virtual reality: System design and multi-camera control. In: *CVPR*, pp. 1–8 (2007)
44. Qureshi, F., Terzopoulos, D.: Multi-camera control through constraint satisfaction for persistent surveillance. In: *AVSS*, pp. 211–218 (2008)
45. Qureshi, F., Terzopoulos, D.: Smart camera networks in virtual reality. *Proc. IEEE* **96**(10), 1640–1656 (2008)

46. Rara, H., Elhabian, S., Ali, A., Miller, M., Starr, T., Farag, A.: Distant face recognition based on sparse-stereo reconstruction. In: ICIP, pp. 4141–4144 (2009)
47. Rara, H., Elhabian, S., Ali, A., Miller, M., Starr, T., Farag, A.: Face recognition at-a-distance based on sparse-stereo reconstruction. In: CVPR Workshop on Biometrics, pp. 27–32 (2009)
48. Redman, B., Höft, T., Grow, T., Novotny, J., McCumber, P., Rogers, N., Hoening, M., Kubala, K., Sibell, R., Shald, S., Uberna, R., Havermann, R., Sandalphon, D.: Low-cost, stand-off, 2D+3D face imaging for biometric identification using Fourier transform profilometry. In: 2009 Military Sensing Symposia (MSS) Specialty Group on Active E-O Systems, vol. 1. Las Vegas, NV (2009)
49. Redman, B., Marron, J., Seldomridge, N., Grow, T., Höft, T., Novotny, J., Thurman, S.T., Embrey, C., Bratcher, A., Kendrick, R.: Stand-off 3D face imaging and vibrometry for biometric identification using digital holography. In: 2009 Military Sensing Symposia (MSS) Specialty Group on Active E-O Systems, vol. 1. Las Vegas, NV (2009)
50. Ross, A.A., Nandakumar, K., Jain, A.K. (eds.): *Handbook of Multibiometrics*. Springer, Berlin (2006)
51. Senior, A., Hampapur, A., Lu, M.: Acquiring multi-scale images by pan-tilt-zoom control and automatic multi-camera calibration. In: WACV, vol. 1, pp. 433–438 (2005)
52. Stillman, S., Tanawongsuwan, R., Essa, I.: A system for tracking and recognizing multiple people with multiple cameras. In: Proc. of 2nd Intl. Conf. on Audio-Vision-based Person Authentication, pp. 96–101 (1998)
53. Tistarelli, M., Li, S.Z., Chellappa, R. (eds.): *Handbook of Remote Biometrics for Surveillance and Security*. Springer, Berlin (2009)
54. Tu, P.H., Doretto, G., Krahnstoever, N.O., Perera, A.G.A., Wheeler, F.W., Liu, X., Rittscher, J., Sebastian, T.B., Yu, T., Harding, K.G.: An intelligent video framework for homeland protection. In: Proc. of SPIE Defense & Security Symposium, Conference on Unattended Ground, Sea, and Air Sensor Technologies and Applications IX. Orlando, FL (2007)
55. Wheeler, F.W., Liu, X., Tu, P.H.: Multi-frame super-resolution for face recognition. In: BTAS (2007)
56. Wheeler, F.W., Weiss, R.L., Tu, P.H.: Face recognition at a distance system for surveillance applications. In: BTAS (2010)
57. Yan, S., Liu, C., Li, S.Z., Zhang, H., Shum, H.Y., Cheng, Q.: Face alignment using texture-constrained active shape models. *Image Vis. Comput.* **21**(1), 69–75 (2003)
58. Yao, Y., Abidi, B., Kalka, N., Schmid, N., Abidi, M.: High magnification and long distance face recognition: Database acquisition, evaluation, and enhancement. In: Proc. Biometrics Symposium (2006)
59. Yao, Y., Abidi, B., Kalka, N.D., Schmid, N., Abidi, M.: Super-resolution for high magnification face images. In: Prabhakar, S., Ross, A.A. (eds.) *Proceedings of the SPIE, Biometric Technology for Human Identification IV*, vol. 6539. Orlando, FL (2007)
60. Yao, Y., Abidi, B.R., Kalka, N.D., Schmid, N.A., Abidi, M.A.: Improving long range and high magnification face recognition: Database acquisition, evaluation, and enhancement. *Comput. Vis. Image Underst.* **111**(2), 111–125 (2008)
61. Yu, T., Lim, S.N., Patwardhan, K., Krahnstoever, N.: Monitoring, recognizing and discovering social networks. In: CVPR (2009)
62. Zhou, X., Bhanu, B.: Feature fusion of face and gait for human recognition at a distance in video. In: ICPR, vol. 4, pp. 529–532 (2006)
63. Zhou, X., Bhanu, B.: Integrating face and gait for human recognition at a distance in video. *IEEE Trans. Syst. Man Cybern., Part B, Cybern.* **37**(5), 1119–1137 (2007)
64. Zhou, X., Collins, R., Kanade, T., Metes, P.: A master-slave system to acquire biometric imagery of humans at distance. In: ACM International Workshop on Video Surveillance (2003)



# Chapter 15

## Face Recognition Using Near Infrared Images

Stan Z. Li and Dong Yi

### 15.1 Introduction

Face recognition should be based on intrinsic factors of the face, such as, the 3D shape and the albedo of the facial surface. Extrinsic factors that include illumination, eyeglasses, and hairstyle, are irrelevant to biometric identity, and hence their influence should be minimized. Out of all these factors, variation in illumination is a major challenge and needs to be tackled first.

Conventional visual (VIS) image based face recognition systems, academic and commercial, are compromised in accuracy by changes in environmental illumination, even for cooperative user applications in an indoor environment. In an in-depth study on influence of illumination changes on face recognition [1], Adini et al. examined several distance measures and local image operators, including Gabor filters, local directive filters, and edge maps, which were considered to be relatively insensitive to illumination changes for face recognition. Several conclusions were made: (i) lighting conditions, and especially light angle, drastically change the appearance of a face; (ii) when comparing unprocessed images, the changes between images of a person under different illumination conditions are larger than those between images of two persons under the same illumination; (iii) all the local filters under study, are not capable overcoming variations due to changes in illumination direction. The influence of illumination is also shown in evaluations such as Face Recognition Vendor Test [17].

Near infrared (NIR) based face recognition [11–14], as opposed to the conventional visible light (VIS) based methods, is an effective approach for overcoming the impact of illumination changes on face recognition. It uses a special purpose

---

S.Z. Li (✉) · D. Yi

Center for Biometrics and Security Research & National Laboratory of Pattern Recognition,  
Institute of Automation, Chinese Academy of Sciences, Beijing, China  
e-mail: [szli@cbsr.ia.ac.cn](mailto:szli@cbsr.ia.ac.cn)

D. Yi  
e-mail: [dyi@cbsr.ia.ac.cn](mailto:dyi@cbsr.ia.ac.cn)

imaging device to capture front-lighted NIR face images [3–5], normalizing the illumination direction. Using a proper face feature representation, such as Local Binary Pattern (LBP) [2, 9, 18], variation in the illumination strength is also overcome. These lead to a complete illumination-invariant face representation. Problems caused by uncontrolled environmental illumination are minimized thereby, and difficulties in building the face matching engine are alleviated. The NIR approach usually achieves significantly higher performance than the VIS approach for cooperative user application scenarios in uncontrolled illumination environment. NIR face recognition products and integrated systems have been in the market and are used in many applications (refer to Chap. 1).

In this chapter, we introduce the NIR face recognition approach, describe the design of active NIR face imaging system, illustrate how to derive from NIR face image an illumination invariant face representation, and provide a learning based method for face feature selection and classification. Experiments are presented.

## 15.2 Active NIR Imaging System

The key aspect in the NIR face recognition approach is a special purpose NIR image capture hardware system [14]. Its goal is to overcome the problem arising from uncontrolled environmental light and produce face images of a good illumination condition for face recognition. Good illumination means (i) that the lighting to the face is from the frontal direction and (ii) that the face image has suitable pixel intensities.

To achieve illumination from the frontal direction, active NIR illuminators, for example, space light-emitting diodes (LEDs) around the camera lens, are used to illuminate the face from the front such that front-lighted NIR face images are acquired. This is similar to a camera with a flash light but the NIR lights work in the invisible spectrum of NIR, being nonintrusive to human eyes.

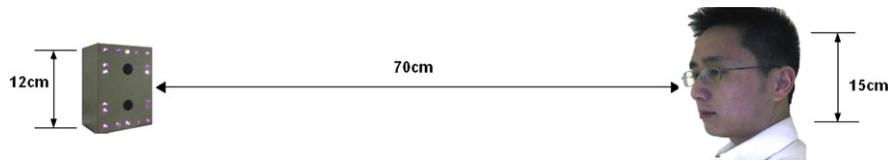
The following are the main requirements for the NIR imaging system:

1. The active NIR lights should be nonintrusive to human eyes.
2. The direction of the NIR lighting to the face should be fixed.
3. The active NIR light signals arriving at the camera sensor should override the signals from other light sources in the environment.

Here, the NIR lights mean the active NIR lights from the NIR imaging system, excluding NIR components in the environment such as sunlight and light bulbs.

This selective capture (of NIR light from the imaging system) can be achieved by the following methods:

1. Choose illuminators such as LEDs in an invisible spectrum. While a 850 nm LED light looks a dim dark red, 940 nm is entirely invisible.
2. Mount the NIR LEDs around the camera lens so as to illuminate from the frontal direction.



**Fig. 15.1** An active NIR imaging device and the face



**Fig. 15.2** Color images (*top*) captured by a color camera versus NIR images (*bottom*) captured by an NIR imaging system. While unfavorable lighting is obvious in the color face images, it is almost unseen in the NIR face images

3. Choose the NIR LEDs to be powerful enough to override environmental light sources that may affect the system. The most favorable or least challenging environment is when there is total darkness and the least favorable or most challenging is sunlight in summer. A short camera exposure should be used to avoid over-exposure when the LED power is high.
4. Use an optical filter to minimize lights from the environment. One option is to use a long pass filter that filters out visible components of environmental lights. A better but more expensive option, is to use a narrow band pass filter that matches the wavelength of the chosen active LEDs.

Figure 15.1 illustrates a hardware device and its positional relationship with the face. The device consists of 18 NIR LEDs, an NIR camera, a VIS color camera, and the casing. The NIR LEDs and the NIR camera are for NIR face image acquisition. The hardware and the face are relatively positioned in such a way that the lighting is frontal and the NIR rays provide nearly homogeneous illumination on the face, which is an excellent illumination condition for face recognition. The VIS image may be used for visual feedback and human computer interaction (HCI). The imaging device works at a rate of 30 frames per second with the USB 2.0 protocol for  $640 \times 480$  images.

Figure 15.2 depicts images of a face illuminated by NIR LED lights from the front, a lamp aside and environmental lights. We can see the following: (i) the lighting conditions are likely to cause problems for face recognition with the color im-

ages; (ii) the NIR images, with the visible light composition cut off by the filter, are mostly frontal-lighted (by the NIR lights), with minimum influence from the side lighting. Based on the two set of images that were obtained the following observations can be made.

In outdoor environments, the sunlight contains very a strong NIR component, much stronger than what can be overridden by the NIR imaging system described above. The NIR imaging hardware must be enhanced to overcome the influence from the sunlight. The key factor is the ratio between the power of the controlled active NIR illumination and the power from other light sources (recorded by the image sensor).

One solution for such an enhanced NIR imaging system is provided in [21]. The system uses a powerful NIR illumination flash and synchronizes it with image sensor exposure. The main points of the enhanced imaging system are summarized as follows:

1. Use a powerful NIR illuminator, such as a narrow band NIR laser generator.
2. Set very short imaging exposure time, for example, 50  $\mu$ s.
3. Synchronize the time of NIR flash output to the exposure window of sensor.
4. Use a narrow band-pass optical filter that matches the wavelength of the chosen active NIR flash.

The enhanced NIR imaging (ENIR) system may include an additional processing step to further reduce the influence from strong and uncontrolled lights in the environment. This is done by taking difference of two successive frames. The first frame is captured by illuminating the subject's face with an active light source and the second is captured after turning this light source off. The second frame is used to represent the face subject under ambient static lights. The difference operation is performed to reduce or eliminate the strong NIR component in the ambient light or sunlight [10], and thus output an image of the face illuminated by the active NIR lighting in the frontal direction. As a result, this ENIR imaging system not only provides appropriate active frontal lighting but also minimizes ambient light as well as outdoor sunlight.

Figure 15.3 shows some images captured when the active light source on/off and the finally output of the ENIR camera under the sunlight. From Fig. 15.3(c), we can see that ENIR imaging hardware can work properly under the sunlight. Figure 15.10 shows face images captured by the ENIR imaging system in the sunlight. It demonstrates the effect of reducing NIR component from sunlight.

### 15.3 Illumination Invariant Face Representation

In this section, we first provide an analysis using a Lambertian surface imaging model to show that the NIR images contain the most relevant, intrinsic information about a face, subject only to a multiplying constant or a monotonic transform due to lighting intensity changes. We then present an Local Binary Pattern (LBP) based representation to amend the degree of freedom of the monotonic transform to achieve an illumination invariant representation for face recognition applications.



**Fig. 15.3** From left to right: **a** Face image in mixture of sunlight and active NIR light; **b** face image in sunlight only, **c** image difference (**a**)–(**b**)

### 15.3.1 Modeling of Active NIR Images

According to the Lambertian model, an image  $I(x, y)$  under a point light source is formed according to the following criterion:

$$I(x, y) = \rho(x, y)\mathbf{n}(x, y)\mathbf{s} \quad (15.1)$$

where,  $\rho(x, y)$  is the albedo of the facial surface material at point  $(x, y)$ ,  $\mathbf{n} = (n_x, n_y, n_z)$  is the surface normal (a unit row vector) in 3D space, and  $\mathbf{s} = (s_x, s_y, s_z)$  is the lighting direction (a column vector, with magnitude). Albedo  $\rho(x, y)$  reflects the photometric properties of facial skin and hairs.  $\mathbf{n}(x, y)$  is the geometric shape of the face. The most important factor that affects the face recognition performance is the direction of the incident lighting relative to the face surface normal. The product of  $\rho(x, y)$  and  $\mathbf{n}(x, y)$  is the intrinsic property of the face at a fixed pose and is the only factor needed for face detection and recognition. Therefore,  $\mathbf{s}$  is the extrinsic property that should be removed. Assume  $\mathbf{s} = \kappa\mathbf{s}^0$ , where  $\kappa$  is a multiplying constant that is introduced to account for possible changes in the strength of the lighting caused by changes in the distance between the face and the LED lights, and  $\mathbf{s}^0 = (s_x^0, s_y^0, s_z^0)$  is a unit column vector of the lighting direction. Let  $\theta(x, y)$  be the incident angle between the lighting and the face surface normal at point  $(x, y)$ , then  $\cos\theta(x, y) = \mathbf{n}(x, y)\mathbf{s}^0$ . Equation (15.1) can be expressed as

$$I(x, y) = \kappa\rho(x, y)\cos\theta(x, y). \quad (15.2)$$

A less restrictive modeling of constant  $\kappa$  would be to use a monotonic transform instead of a constant. It's evident we see that the face image  $\rho(x, y)\cos\theta(x, y)$  changes as the lighting direction changes, given albedo  $\rho(x, y)$  and 3D shape  $\mathbf{n}(x, y)$  fixed. The present hardware design is aimed at preserving the intrinsic property while minimizing variation due to the extrinsic factor of environmental lights. When the active NIR lighting is from the (nearly) frontal direction (see Fig. 15.1),

**Fig. 15.4** LBP code for  $3 \times 3$  window

| Local Window                                                                                                                                                 | Thresholded | Weights |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|---------|---|----|------|---|----|----|----|-------------------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|---|-----|---|---|---|---|------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---|---|----|-----|---|----|----|-----|
| <table border="1"> <tr><td>18</td><td>15</td><td>8</td></tr> <tr><td>21</td><td>(18)</td><td>6</td></tr> <tr><td>27</td><td>23</td><td>22</td></tr> </table> | 18          | 15      | 8 | 21 | (18) | 6 | 27 | 23 | 22 | <table border="1"> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>(0)</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table> | 1 | 0 | 0 | 1 | (0) | 0 | 1 | 1 | 1 | <table border="1"> <tr><td>8</td><td>4</td><td>2</td></tr> <tr><td>16</td><td>(0)</td><td>1</td></tr> <tr><td>32</td><td>64</td><td>128</td></tr> </table> | 8 | 4 | 2 | 16 | (0) | 1 | 32 | 64 | 128 |
| 18                                                                                                                                                           | 15          | 8       |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 21                                                                                                                                                           | (18)        | 6       |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 27                                                                                                                                                           | 23          | 22      |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 1                                                                                                                                                            | 0           | 0       |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 1                                                                                                                                                            | (0)         | 0       |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 1                                                                                                                                                            | 1           | 1       |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 8                                                                                                                                                            | 4           | 2       |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 16                                                                                                                                                           | (0)         | 1       |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |
| 32                                                                                                                                                           | 64          | 128     |   |    |      |   |    |    |    |                                                                                                                                                       |   |   |   |   |     |   |   |   |   |                                                                                                                                                            |   |   |   |    |     |   |    |    |     |

LBP String = (0001111)  
LBP Code =  $0+0+0+8+16+32+64+128=248$

that is,  $s^0 = (0, 0, 1)$ , the image can be approximated by

$$I(x, y) = \kappa \rho(x, y) n_z(x, y), \quad (15.3)$$

where,  $n_z(x, y)$  is the z component of the surface normal that can be acquired by a range imaging system. An active NIR image  $I(x, y)$  combines information about both surface normal component  $n_z(x, y)$  and albedo map  $\rho(x, y)$  and, therefore, provides the required intrinsic property about a face for face recognition.

### 15.3.2 Compensation for Monotonic Transform

Given a face, the constant  $\kappa$  is the only factor affecting the intensity of the face image. This monotonic transform in intensity could be calibrated by histogram equalization, histogram specification, or some monotonic transform invariant features.

The degree of freedom in  $\kappa$  or in a monotonic transform can be compensated by using LBP features to achieve an illumination invariant representation of faces. The basic form of the LBP operator is illustrated in Fig. 15.4. The binary bits describing a local  $3 \times 3$  sub-window are generated by thresholding the 8 pixels in the surrounding locations by the gray value of its center; the feature vector is formed by concatenating the thresholded binary bits in an anticlockwise manner. There are a total of 256 possible values and, hence, 256 LBP patterns denoted by such an LBP code; each value represents a type of LBP local pattern. Such a basic form of LBP can be extended to multi-scale LBP,  $LBP_{(P,R)}$ , where  $R$  is the radius of the circle surrounding the center and  $P$  is the number of pixels on the circle. An LBP  $(P, R)$  string is called uniform, denoted by  $LBP_{(P,R)}^{u2}$ , if the neighboring bits (the circular sense) contain at most two bitwise transitions from 0 to 1 or vice versa (see [18] for details).

From the analysis, we see that the NIR imaging and LBP features together lead to an illumination invariant representation of faces. In other words, applying the LBP operator to an active NIR image generates illumination invariant features for faces. The illumination invariant face representation provides great advantages for face recognition in varying illumination.

An LBP-based face matching method is described in [2, 9]. In this method, the image is divided into  $7 \times 7 = 49$  blocks. An LBP histogram is calculated for

each block. A  $\chi^2$  distance is calculated between two histograms for matching and a weighted sum of the  $\chi^2$  distance is then used for matching between two face images. The method is shown to achieve very good results on the FERET database. However, such a method still lacks optimality in terms of the block division and the weights.

Recently, several LBP variants are proposed for face recognition, such as Multi-scale Block LBP (MB-LBP) [15], Local Ternary Pattern (LTP) [19] and so on. These features are also robust to intensity monotonic changes and can be used in NIR face recognition.

The following describes a procedure for extracting LBP histogram features:

### 1. Computing Base LBP Features

- Computing  $LBP_{8,1}^{u_2}$  codes for every pixel location in the image.

### 2. LBP Code Histogramming

- A histogram of the base LBP codes is computed over a local region centered at each pixel, each histogram bin being the number of occurrences of the corresponding LBP code in the local region. There are 59 bins for  $LBP_{8,1}^{u_2}$ .
- An LBP histogram is considered as a set of 59 individual features.

### 3. Gathering LBP Histograms

- For a face image of size  $W \times H$ , with the interior area of size  $W' \times H'$ , the total number of LBP histogram features is  $D = W' \times H' \times 59$  (number of valid pixel locations times the number of LBP histogram bins).

For example, if  $W \times H = 120 \times 142$  and a local region for histogramming is a rectangle of size  $16 \times 20$ , the interior area is of size  $W' \times H' = 104 \times 122$  pixels. Then there are a total of  $104 \times 122 \times 59 = 748\,592$  elements in the LBP histogram feature pool.

## 15.4 NIR Face Classification

Of the large number of LBP histogram features present in a feature pool. Some are useful for face recognition, some are not so useful, and some may be contradictory. They must be selected or weighted to achieve the best performance. In this section, we present an AdaBoost [7, 20] based learning method for selecting best LBP features and constructing a face classifier.

Given a training set of LBP features of faces subject to image noise, slight pose changes, and alignment errors, the learning method finds a good set of discriminative features among a large number of candidates and then build a strong classifier based on the selected features. Once trained, the classifier is able to recognize faces without having to be retrained when a new individual client is added.

### 15.4.1 AdaBoost Based Feature Selection

As an AdaBoost procedure essentially learns a two-class classifier, we convert the multi-class problem into a two-class problem using the idea of intra- and extra-class differences [16]. Two face examples are considered as intra-class if they are of the same person, or extra-class, otherwise. However, in terms of facial features, the difference data are derived between samples in the features rather than in the original image space, that is, a difference is taken between the facial features (for example, LBP histograms) of two face examples. In this manner, a training set of positive (intra-class) and negative (extra-class) training examples can be obtained for the learning procedure.

Assume that a training set of  $N$  examples is given as positive and negative classes,  $\mathbf{S} = (x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$  where  $x_i$  is a training example (the difference between two feature vectors) and  $y_i \in \{+1, -1\}$  is the class label. The AdaBoost procedure can be used to learn a set of best  $T$  features stagewise and thereby constructs a sequence of  $T$  weak classifiers,  $h_t(x) \in \{+1, -1\}$ , and linearly combine the weak classifiers in an optimal way into a stronger classifier,

$$H(x) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(x)\right), \quad (15.4)$$

where  $\alpha_t \in \mathbf{R}$  are the combining weights. The AdaBoost learning procedure is originally aimed at deriving  $\alpha_t$  and  $h_t(x)$  so that an upper error bound is minimized [7]. The reader is referred to [7, 20] for AdaBoost learning.

AdaBoost assumes that a procedure is available for learning a weak classifier  $h_t(x)$  from the training examples weighted by the current distribution  $w_t$ . We use a weak classifier based on a single scalar feature, that is, an LBP histogram bin value. Therefore, when AdaBoost constructs a  $h_t(x)$ , it need to select a good feature for it. In this way, AdaBoost can provide a good subset of features.

In the test phase, the learned  $H(x)$  can be used to classify face images. The difference is calculated between the selected features of the two face images. A weak decision  $h_t$  can be made in terms of each selected feature. The weak decisions are linearly combined with the weights  $\alpha_t$  to give the predict value  $H(x)$ . The final decision can be made by comparing  $H(x)$  with a threshold value. If greater, the two face images are considered as belonging to the same person (intra-class), otherwise, they are belonging to different persons (inter-class). Moreover, a cascade of AdaBoost classifiers [20] can be constructed to cope with complex distributions of two classes.

### 15.4.2 LDA Classifier

LDA reduces the dimensionality by linearly projecting the original feature vector in high dimensional space to a lower dimensional subspace such that the ratio of

within-class scatter over between-class scatter is minimized. This minimized the classification error when the distributions of the class data are Gaussian [8]. The high dimensional data may be preprocessed using the PCA transform to make the within-class scatter matrix nonsingular, before LDA is applied. The basis images of such a combined projection  $\mathbf{P}$  of PCA and LDA is called Fisherfaces [6]. More advanced forms of LDA, such as direct LDA or regularized LDA, could be used to obtain  $\mathbf{P}$ . The input of LDA, in our context, is the space of selected features.

Given two input vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$  in the space of selected features, their LDA projections are calculated as  $\mathbf{v}_1 = \mathbf{Px}_1$  and  $\mathbf{v}_2 = \mathbf{Px}_2$  and the following cosine score (or called “cosine distance” in some of the literature) is used for the matching:

$$H(\mathbf{v}_1, \mathbf{v}_2) = (\mathbf{v}_1 \cdot \mathbf{v}_2) / \|\mathbf{v}_1\| \|\mathbf{v}_2\|. \quad (15.5)$$

In the test phase, the projections  $\mathbf{v}_1$  and  $\mathbf{v}_2$  are computed from two input vectors  $\mathbf{x}_1$  and  $\mathbf{x}_2$ , the other for the input face image and one for an enrolled face image. By comparing the score  $H(\mathbf{v}_1, \mathbf{v}_2)$  with a threshold, a decision can be made whether  $\mathbf{x}_1$  and  $\mathbf{x}_2$  belong to the same person.

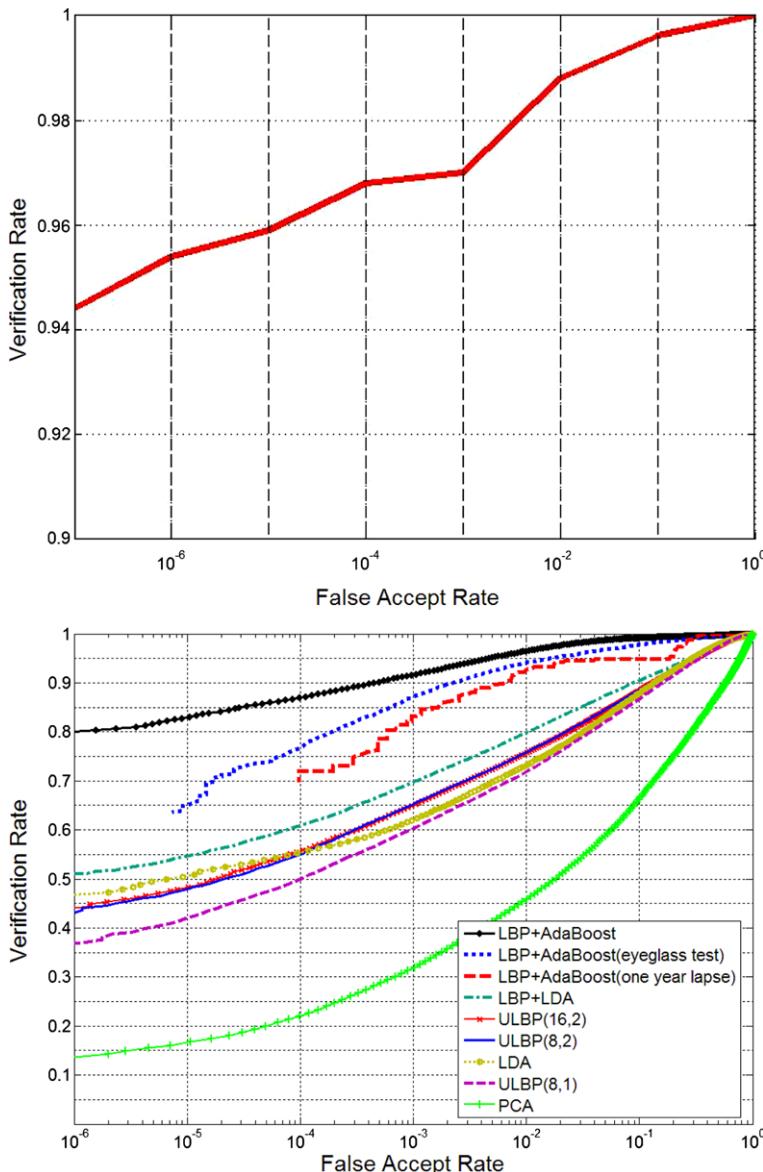
## 15.5 Experiments

In this section, results with the NIR face recognition system of [14] are presented to illustrate the advantages of the NIR face recognition method. Performances of LBP + AdaBoost and LBP + LDA NIR face matching engines are compared with several existing baseline and face matching engines. Case studies regarding effects of eyeglasses, time lapse, and weak illumination are reported. Finally, results on a data set collected under sunlight in an outdoor environment are presented to demonstrate the performance of ENIR system [21].

### 15.5.1 Basic Evaluation

In the training phase, the training set of positive examples were derived from intra-class pairs of LBP histogram features, the negative set from extra-class pairs, each example being a 748 592 dimensional vector. There were  $10^4$  face images of about 1000 persons, 10 images each person, all Chinese. A training set of about  $45 \times 10^3$  positive and  $5 \times 10^7$  negative examples were collected from the training images. A cascade of 5 strong classifiers were trained, with about 1500 weak classifiers. The ROC curves for the training set are shown on the top of Fig. 15.5, where the FAR is reduced to below  $10^{-7}$  with an accuracy of 94.4%.

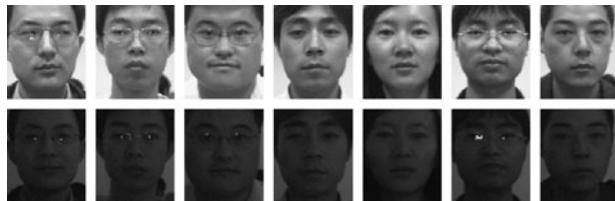
A technology evaluation was done with a test set of 3237 images. The test set contained 35 persons, with 80 to 100 images per person. None of the test images were in the training set. This generated 149 217 intra-class (positive) and 5 088 249



**Fig. 15.5** *Top:* ROC curve of LBP + AdaBoost method for face verification on the training set. *Bottom:* ROC curves for various compared methods

extra-class (negative) pairs. Several other methods were included in the evaluation (for comparison), using the same set of training and test images. They were: (i) PCA on the NIR images (with Mahalanobis distance), (ii) LDA on the NIR images (with cosine distance), (iii) the LBP + LDA method, (iv) the original LBP

Visible Light Face Images under Controlled and Weak Light Condition



LBP+AdaBoost Matching Scores for intra- and extra-class pairs

|        | Ctrl 1        | Ctrl 2        | Ctrl 3        | Ctrl 4        | Ctrl 5        | Ctrl 6        | Ctrl 7        |
|--------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Weak 1 | <b>0.4888</b> | 0.4831        | 0.4751        | 0.4689        | 0.4764        | 0.4745        | 0.4658        |
| Weak 2 | 0.4831        | <b>0.5131</b> | 0.4578        | 0.5202        | 0.4926        | 0.5014        | 0.4709        |
| Weak 3 | 0.4808        | 0.4588        | <b>0.4804</b> | 0.4724        | 0.4814        | 0.4653        | 0.4619        |
| Weak 4 | 0.4531        | 0.4563        | 0.4415        | <b>0.4688</b> | 0.4570        | 0.4582        | 0.4570        |
| Weak 5 | 0.4892        | 0.4904        | 0.4757        | 0.4893        | <b>0.5275</b> | 0.4780        | 0.4824        |
| Weak 6 | 0.4648        | 0.4835        | 0.4768        | 0.5085        | 0.4864        | <b>0.4935</b> | 0.4691        |
| Weak 7 | 0.4686        | 0.4683        | 0.4636        | 0.4708        | 0.4887        | 0.4687        | <b>0.5068</b> |

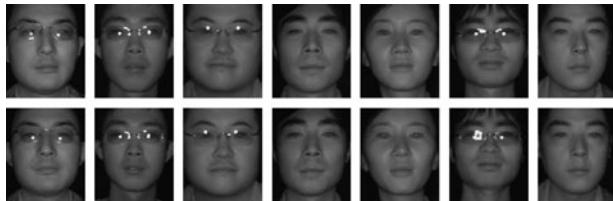
**Fig. 15.6** *Top:* Images captured under controlled illumination (*row 1*) and under weak light conditions (*row 2*), each column belonging to the same person. *Bottom:* LBP + AdaBoost matching scores

method developed by Ahonen et al. [2] and Hadid et al. [9] ( $\chi^2$  distances between LBP histograms in  $7 \times 7$  image blocks) with three operators:  $LBP_{(8,1)}^{u2}$ ,  $LBP_{(8,2)}^{u2}$  and  $LBP_{(16,2)}^{u2}$ . On the bottom of Fig. 15.5 shows the ROC curves derived from the scores for the intra- and extra-class pairs. By the VR values at FAR = 0.1%, the compared methods can be ranked in order of decreasing VR as: LBP + AdaBoost (VR = 91.8%), LBP + LDA (69.9%),  $LBP_{(8,2)}^{u2}$  (65.29%),  $LBP_{(16,2)}^{u2}$  (65.29%), Image + LDA (62.4%),  $LBP_{(8,1)}^{u2}$  (60.7%), and Image + PCA (32.0%). See later for explanations of the “LBP + AdaBoost (eyeglass test)” and “LBP + AdaBoost (one year lapse)” curves.

### 15.5.2 Weak Illumination

Figures 15.6 and 15.7 present case studies to compare performance of visible light (VIS) and NIR image based face matching methods under weak illumination. The LBP + AdaBoost classifier for VIS images was trained using VIS images, whereas the one for NIR images was the one using for other tests. In the tables, the diagonal entries (in bold font) are for the intra-class pairs between controlled and weak illumination. For the VIS case, the mean and variance are 0.4970 and 0.0201 for intra-class pairs, and 0.4747 and 0.0154 for extra-class pairs. There are several cases of mismatch because the intra-class scores are not necessarily higher than the extra-class ones. In contrast, the NIR solution well separates the two classes, with

Active NIR Light Face Images under Controlled and Weak Light Conditions



LBP+AdaBoost Matching Scores for intra- and extra-class pairs

|        | Ctrl 1        | Ctrl 2        | Ctrl 3        | Ctrl 4        | Ctrl 5        | Ctrl 6        | Ctrl 7        |
|--------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Weak 1 | <b>0.6202</b> | 0.3262        | 0.3963        | 0.3354        | 0.3492        | 0.3747        | 0.3558        |
| Weak 2 | 0.3551        | <b>0.6573</b> | 0.3226        | 0.3213        | 0.3637        | 0.4054        | 0.3442        |
| Weak 3 | 0.3886        | 0.3489        | <b>0.7144</b> | 0.3244        | 0.3759        | 0.3249        | 0.3139        |
| Weak 4 | 0.3902        | 0.3062        | 0.3545        | <b>0.6812</b> | 0.4123        | 0.3069        | 0.3144        |
| Weak 5 | 0.4135        | 0.3289        | 0.4507        | 0.3851        | <b>0.6882</b> | 0.3780        | 0.3510        |
| Weak 6 | 0.3459        | 0.4163        | 0.3520        | 0.3292        | 0.2996        | <b>0.6948</b> | 0.2849        |
| Weak 7 | 0.3697        | 0.2847        | 0.2831        | 0.3374        | 0.3656        | 0.2778        | <b>0.6162</b> |

**Fig. 15.7** *Top:* Images captured under controlled illumination (*row 1*) and under weak light conditions (*row 2*), each column belonging to the same person. *Bottom:* LBP + AdaBoost matching scores

the mean and variance of 0.6675 and 0.0377 for intra-class pairs, and 0.3492 and 0.0403 for extra-class pairs, and correctly matches all the pairs.

### 15.5.3 Eyeglasses

This section presents a case analysis of influence of eye glasses on face matching, as shown by the images, and the score table in Fig. 15.8. In the tables, the diagonal entries (in bold font) are for the intra-class pairs without and with glasses (that is, between the two images in the same column); the lower triangle entries for the extra-class no-glass pairs (that is, between two different images in the first row); and the upper triangle for the extra-class glass pairs (that is, between two different images in the second row). The mean and variance of correlations (not shown here due to page limit) are 0.9306 and 0.0419 for intra-class pairs, and 0.7985 and 0.0761 for extra-class pairs of either wearing no glasses or wearing glasses. Compared with correlation, LBP + AdaBoost matching engine can well separate between the two classes—the intra-class scores are consistently higher than those of extra-class scores.

Statistics were also obtained using 1500 images of 30 subjects, 50 images per subject of which 25 are with glasses and 25 without. The no-eyeglass images were used as the gallery set and the eyeglass images as the probe set. The ROC curve is labeled “LBP + AdaBoost (eyeglass test)” in the bottom of Fig. 15.5 (the portion for FAR smaller than  $10^{-5}$  is unavailable because of the limited data points). At FAR =



LBP+AdaBoost Matching Scores for intra- and extra-class pairs

|       | With 1        | With 2        | With 3        | With 4        | With 5        | With 6        | With 7        |        |
|-------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|--------|
| W/O 1 | <b>0.6537</b> | 0.2880        | 0.3276        | 0.3251        | 0.4056        | 0.3510        | 0.3422        | With 1 |
| W/O 2 | 0.2205        | <b>0.6468</b> | 0.1730        | 0.2859        | 0.2545        | 0.2740        | 0.3856        | With 2 |
| W/O 3 | 0.3487        | 0.1838        | <b>0.6565</b> | 0.3209        | 0.3261        | 0.2666        | 0.3231        | With 3 |
| W/O 4 | 0.2464        | 0.2744        | 0.2654        | <b>0.7092</b> | 0.3331        | 0.2986        | 0.3680        | With 4 |
| W/O 5 | 0.3425        | 0.2945        | 0.3352        | 0.2896        | <b>0.6120</b> | 0.3191        | 0.3704        | With 5 |
| W/O 6 | 0.2896        | 0.2100        | 0.2205        | 0.2871        | 0.2699        | <b>0.6366</b> | 0.2346        | With 6 |
| W/O 7 | 0.2915        | 0.3708        | 0.3197        | 0.3958        | 0.3397        | 0.3056        | <b>0.6541</b> | With 7 |
|       | W/O 1         | W/O 2         | W/O 3         | W/O 4         | W/O 5         | W/O 6         | W/O 7         |        |

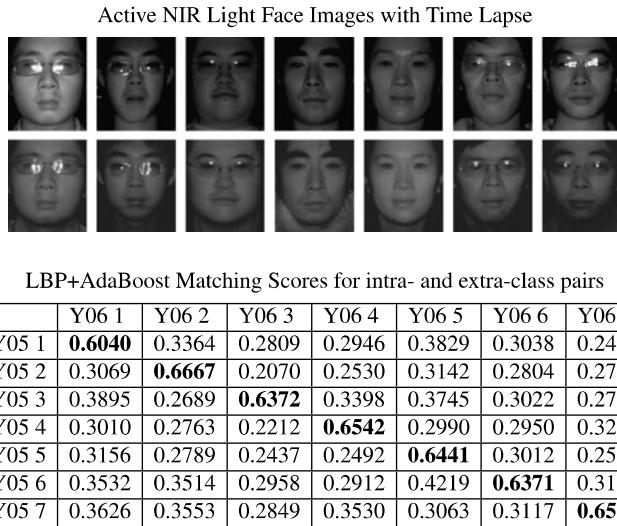
**Fig. 15.8** Analysis on effects of glasses. *Top:* Images without glasses (*row 1*) and with glasses (*row 2*), each column belonging to the same person. *Bottom:* LBP + AdaBoost matching scores

0.1%, the VR was 87.1%, as opposed to 91.8% of the “LBP + AdaBoost” curve for the no-eyeglasses vs. no-eyeglasses and eyeglasses vs. eyeglasses comparisons.

#### 15.5.4 Time Lapse

Tests were performed to evaluate effect of time lapse on NIR face recognition. Figure 15.9 presents a case analysis of time lapse effect on the matching scores. The NIR images of 7 individuals were acquired in Spring 2005 and Spring 2006, respectively. The table shows the matching scores produced by the LBP + AdaBoost classifier trained on active NIR images. The mean and variance of the scores are 0.6421 and 0.0198 for intra-class pairs (in bold font), and 0.3045 and 0.0462 for extra-class pairs. The LBP + AdaBoost matching engine well separates between the two classes, the intra-class scores are consistently higher than the extra-class ones.

Statistics were also obtained using 750 images of 30 persons, 25 images per person; of the 25 images, 10 were captured one year ago and used as the gallery set, and 15 were current images used as the probe set. The ROC curve is labeled “LBP + AdaBoost (one year lapse)” in the bottom of Fig. 15.5 (the portion for FAR smaller than  $10^{-4}$  is unavailable because of the limited data points). At FAR = 0.1%, the VR was 83.24%, as opposed to 91.8% for images of no significant time lapse (the “LBP + AdaBoost” curve).



**Fig. 15.9** Analysis on effects of time lapse. *Top*: NIR face images of spring 2005 (*row 1*) and spring 2006 (*row 2*), each column belonging to the same person. *Bottom*: LBP + AdaBoost matching scores where Y05 and Y06 denote Spring 2005 and Spring 2006, respectively

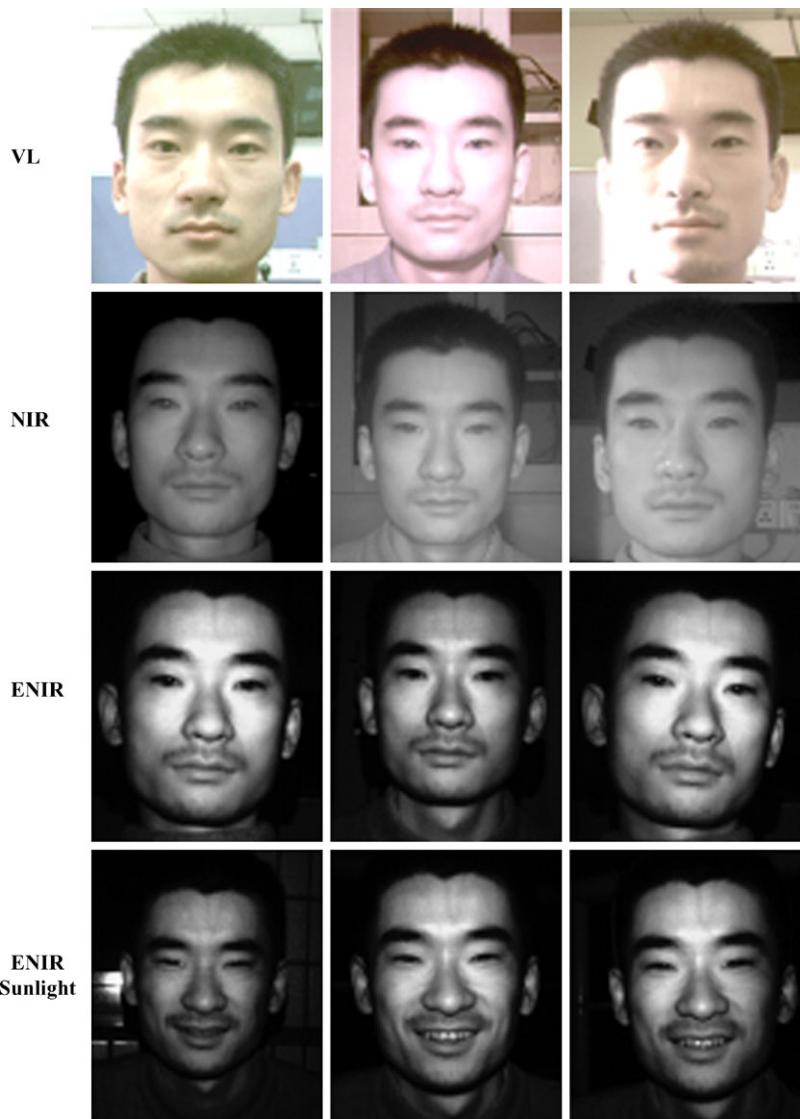
### 15.5.5 Outdoor Environment

These following case studies were performed to evaluate the robustness of ENIR [21] system in an outdoor environment and compare its performance to VIS and NIR systems. To this end, a independent test set containing 20 persons is collected under the following lighting conditions:

1. Normal indoor frontal lighting
2. Strong indoor frontal lighting
3. Strong indoor non-frontal lighting
4. Outdoor frontal sunlight
5. Outdoor side sunlight
6. Outdoor back sunlight

where the strong indoor lighting is provided by a 1000 W tungsten-halogen lamp with a color temperature of 2500–3000 K. While capturing face images, the lamp is placed at 3.5–4 m away from the subject. The lamp contains a wide range of NIR compositions and can be used to simulate sunlight in an indoor environment.

The VIS and NIR cameras are saturated easily by sunlight, and hence the test set does not include VIS and NIR images outdoor. The number of VIS and NIR face images in the set are  $4$  (indoor conditions)  $\times 20$  (person)  $\times 10$  (images/person) = 800. The number of ENIR face images is  $800 + 3$  (outdoor conditions)  $\times 20$  (person)  $\times 10$  (images/person) = 1400. The resolution of VIS, NIR and ENIR images are all  $640 \times 480$ .



**Fig. 15.10** Face images in the outdoor test set. *Rows 1–3:* VIS, NIR and ENIR images under 3 indoor lighting conditions. *Row 4:* ENIR images captured outdoors

Figure 15.10 shows some face images in the test set. We can see that VIS face images are unstable to the direction and strength of the strong lighting, NIR images are less unstable, and ENIR images are the most stable ones under various lighting conditions.

In the experiments, the face images under normal indoor frontal lighting are used as gallery and the other images are used as probe. Two protocols are used to compare the three imaging systems:

1. For indoor images: comparing ROCs of all the 3 systems.
2. For outdoor images: evaluating ROC of the ENIR system only, while the other systems could not function to a satisfactory extent.

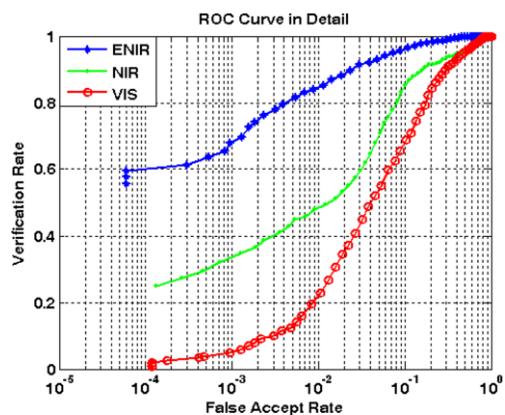
Figure 15.11 shows the ROC curves for all experiments, in which the ENIR system is the most stable one under all lighting conditions. In Fig. 15.11(a), the performance of VIS and NIR systems dropped significantly under strong frontal halogen lamp light, whereas the ENIR system still has relatively high verification rate ( $VR = 69\%$  when  $FAR = 0.001$  and  $VR = 85\%$  when  $FAR = 0.01$ ). Similar trend is also shown in Fig. 15.11(b). Figure 15.11(c) shows the results of the ENIR system under outdoor sunlight in three different directions. The highest verification rate is  $VR = 50\% @ FAR = 0.001$  and  $VR = 69\% @ FAR = 0.01$  when the sunlight is from the subjects' back direction. From the above figures, we can see the performance under frontal lighting is always worse when compared to nonfrontal lighting. A possible explanation is that human eyes are easily disturbed by the frontal lighting, and this can cause a significant change in facial expression, which is another challenging problem in face recognition.

## 15.6 Conclusions

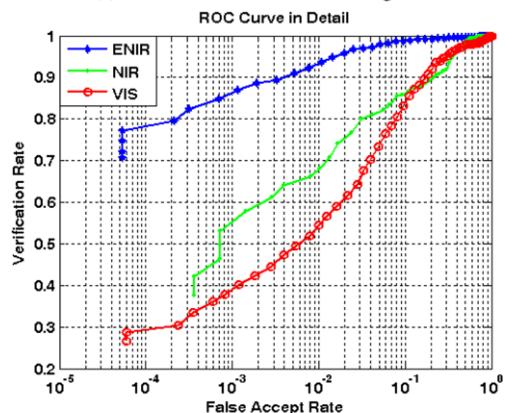
We have presented an effective solution for overcoming the problems caused by illumination variation that severely affects the performance of face recognition systems. The solution consists of active NIR imaging hardware, more efficient algorithms, and a novel system design. An illumination invariant face representation is obtained by extracting LBP features from NIR images. The AdaBoost procedure is used to learn a powerful face recognition engine based on the invariant representation. Highly accurate and fast face recognition systems can be built thereby. Extensive experiments show the robustness of the present solution in terms of image properties, illumination changes, ethnic groups, and advantages over existing methods. An enhanced solution, using a newly developed ENIR imaging device, is presented to deal with strong NIR composition in ambient light such as in the sunlight. The results show that the ENIR system performed significantly better than the VIS and NIR systems in adverse illumination environment. This approach results in face recognition products that perform well for 1-to-many identification for cooperative user applications.

**Acknowledgements** This work was partially supported by the Chinese National Natural Science Foundation Project #61070146, the National Science and Technology Support Program Project #2009BAK43B26, and the AuthenMetric R&D Funds (2004–2011). The work was also partially supported by the TABULA RASA project (<http://www.tabularasa-euproject.org>) under the Seventh Framework Programme for research and technological development (FP7) of the European Union (EU), grant agreement #257289.

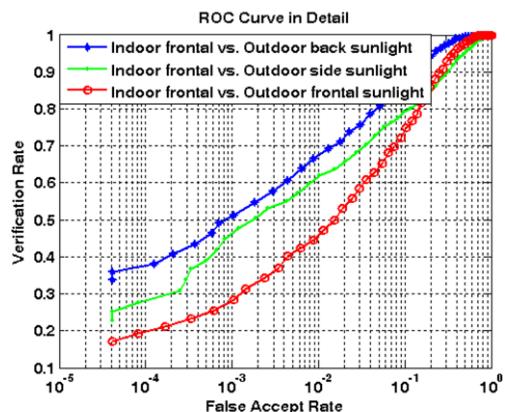
**Fig. 15.11** ROC curves of NIR, VIS and ENIR systems on the outdoor test set



(a) Indoor frontal vs. Indoor strong frontal.



(b) Indoor frontal vs. Indoor strong non-frontal.



(c) ENIR system under outdoor sunlight.

## References

1. Adini, Y., Moses, Y., Ullman, S.: Face recognition: The problem of compensating for changes in illumination direction. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 721–732 (1997)
2. Ahonen, T., Hadid, A., Pietikainen, M.: Face recognition with local binary patterns. In: Proceedings of the European Conference on Computer Vision, pp. 469–481, Prague, Czech Republic (2004)
3. AuthenMetric Co. Ltd.: A method for face image acquisition using active lighting. Patent Application No. 200310121340.1, 12 December 2003
4. AuthenMetric Co. Ltd.: A method for face image acquisition and a method and system for face recognition. Patent Application No. PCT/CN2004/000482, 14 May 2004
5. AuthenMetric Co. Ltd.: An image acquisition apparatus for face recognition. Patent Application No. 200520022878.1, 22 March 2005
6. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. In: Proceedings of the European Conference on Computer Vision, pp. 45–58 (1996)
7. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of online learning and an application to boosting. *J. Comput. Syst. Sci.* **55**(1), 119–139 (1997)
8. Fukunaga, K.: Introduction to Statistical Pattern Recognition, 2nd edn. Academic Press, Boston (1990)
9. Hadid, A., Pietikainen, M., Ahonen, T.: A discriminative feature space for detecting and recognizing faces. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 797–804 (2004)
10. Hizem, W., Krichen, E., Ni, Y., Dorizzi, B., Garcia-Salicetti, S.: Specific sensors for face recognition. In: Proceedings of IAPR International Conference on Biometric, vol. 3832, pp. 47–54 (2006)
11. Li, S.Z., His Face Team: AuthenMetric F1: a highly accurate and fast face recognition system. In: ICCV2005—Demos, 15–21 October 2005
12. Li, S.Z., Chu, R.F., Ao, M., Zhang, L., He, R.: Highly accurate and fast face recognition using near infrared images. In: Proceedings of IAPR International Conference on Biometric (ICB-2006), pp. 151–158, Hong Kong, January 2006
13. Li, S.Z., Zhang, L., Liao, S.C., Zhu, X.X., Chu, R.F., Ao, M., He, R.: A near-infrared image based face recognition system. In: Proceedings of 7th IEEE International Conference Automatic Face and Gesture Recognition (FG-2006), pp. 455–460, Southampton, UK, 10–12 April 2006
14. Li, S.Z., Chu, R., Liao, S., Zhang, L.: Illumination invariant face recognition using near-infrared images. *IEEE Trans. Pattern Anal. Mach. Intell.* **26** (2007) (Special issue on Bio-metrics: Progress and Directions)
15. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block local binary patterns for face recognition. In: ICB, pp. 828–837 (2007)
16. Moghaddam, B., Nastar, C., Pentland, A.: A Bayesian similarity measure for direct image matching. Media Lab Tech Report No. 393, MIT, August 1996
17. NIST. Face Recognition Vendor Tests (FRVT). <http://www.frvt.org>
18. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(7), 971–987 (2002)
19. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Trans. Image Process.* **19**(6), 1635–1650 (2010)
20. Viola, P., Jones, M.: Robust real time object detection. In: IEEE ICCV Workshop on Statistical and Computational Theories of Vision. Vancouver, Canada, 13 July 2001
21. Yi, D., Liu, R., Chu, R., Liu, D., Wang, R., Li, S.Z.: Outdoor face recognition using enhanced near infrared imaging. In: Proceedings of IAPR International Conference on Biometric, Seoul, Korea, August 2007

# Chapter 16

## Multispectral Face Imaging and Analysis

Andreas Koschan, Yi Yao, Hong Chang, and Mongi Abidi

### 16.1 Introduction

This chapter addresses the advantages of using multispectral narrow-band images for face recognition, as opposed to conventional broad-band images obtained by color or monochrome cameras (see also the chapter for a discussion of color in face analysis). Narrow-band images are by definition taken over a very small range of wavelengths, while broad-band images average the information obtained over a wide range of wavelengths. There are two primary reasons for employing multispectral imaging for face recognition.

First, we believe that there is distinctive facial information contained in certain narrow spectral bands which can be acknowledged and employed to enhance face recognition performance in comparison to broad-band color or black and white images. Broad-band imaging has the potential to degrade this information that is embedded in the narrow-band image due to the integration process over a wide range of wavelengths during the formation of the image.

Second, multispectral images can separate the illumination information from the reflectance of objects, so that we can use this illumination information to normalize the images. In contrast, it is nearly impossible to separate and employ the illumination distribution information from broad-band images. To verify the effectiveness of

---

A. Koschan (✉) · H. Chang · M. Abidi

Imaging, Robotics, and Intelligent Systems Lab, University of Tennessee, Knoxville, TN 37996, USA

e-mail: [akoschan@utk.edu](mailto:akoschan@utk.edu)

H. Chang

e-mail: [hchang2@utk.edu](mailto:hchang2@utk.edu)

M. Abidi

e-mail: [abidi@utk.edu](mailto:abidi@utk.edu)

Y. Yao

Visualization and Computer Vision Lab, GE Global Research, Niskayuna, NY 12309, USA

e-mail: [yi.yao@ge.com](mailto:yi.yao@ge.com)

multippectral images for improving face recognition, two sequential procedures are taken into account: first, multippectral face image acquisition and second, spectral band selection.

To reduce information redundancy among multippectral images, complexity-guided distance-based band selection is introduced which uses a model selection criterion for an automatic selection. This selection can simplify the imaging process by reducing the number of multippectral images to be taken under a given illumination. In other words, the goal is to identify a small set of optimal multippectral bands to be taken under a given illumination as opposed to acquiring a large set of multippectral bands over the entire visible spectrum.

The performance of selected bands outperforms the conventional images by up to 15%. From the significant performance improvement via complexity-guided distance-based band selection, we conclude that specific facial information carried in certain narrow-band spectral images can enhance face recognition performance compared to broad-band images. In addition, the algorithm is equally useful and successful in a wide variety of recognition schemes.

## 16.2 Multippectral Imaging

Multippectral imaging is a technique that provides images of a scene at multiple wavelengths and can generate precise optical spectra at every pixel. A *multippectral image* is a collection of several monochrome images of the same scene, each of them taken with additional receptors sensitive to other frequencies of the visible light, or to frequencies beyond visible light, like the infrared region of electromagnetic continuum. Each image is referred to as a *band* or a *channel*. Multippectral imaging produces a three-dimensional image cube with two spatial dimensions (horizontal and vertical) and one spectral dimension. The spectral dimension contains spectral information for each pixel on the multippectral cube. A *multippectral image* can be represented as

$$C(x, y) = (\mu_1(x, y), \mu_2(x, y), \dots, \mu_{N_B}(x, y))^T (\mu_1, \mu_2, \dots, \mu_{N_B})^T. \quad (16.1)$$

The signal strength  $u_k(x, y)$  of a camera sensor in a certain wavelength range,  $\lambda_{\min}$  to  $\lambda_{\max}$ , can be represented as

$$u_k(x, y) = \int_{\lambda_{\min}}^{\lambda_{\max}} R(x, y, \lambda) L(x, y, \lambda) S_k(x, y, \lambda) d\lambda, \quad (16.2)$$

with  $k = 1, \dots, N_B$ , where  $N_B = 1$  for monochromatic images and  $N_B = 3$  for three-channel color images. The parameters  $(x, y)$  indicate the pixel location in the image.  $R(x, y, \lambda)$  is the spectral surface reflectance of the object,  $L(x, y, \lambda)$  is the spectral distribution of the illumination, and  $S_k(x, y, \lambda)$  is the spectral sensitivity of the camera corresponding to channel  $k$ . The entire possible integration wavelength range can be in the visible spectrum, 400–720 nm, or in addition may include infrared spectrum depending on the camera design.

While a monochrome image,  $N_B = 1$ , has only one band, which is represented as a gray-value image, a multispectral image consists of at least three bands,  $N_B \geq 3$ . Thus, the image value of a pixel in a multispectral image is represented by vectors with  $N_B$  components, as opposed to scalar image values representing pixels in a monochrome image. Although a color image with three bands constitutes in theory the simplest form of a multispectral image, the term is more commonly used for images with more than three bands. One example would be a four-band image using the three *RGB* bands and an additional band beyond the visible spectrum, like in the infrared (IR). Satellites usually take several images from frequency bands in the visible and nonvisible range. No common agreement exists yet on the definition of the term *hyperspectral image*. However, the term is commonly used for images with more than a hundred bands,  $N_B > 100$ . While *multi* in multispectral means *many* spectral bands, the *hyper* in hyperspectral means *over* as in *more than many* and refers to the large number of measured wavelength bands.

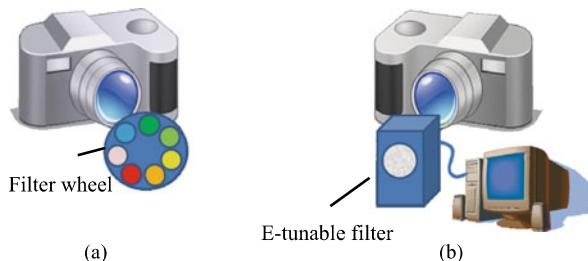
Multispectral imaging can enhance and expand the capability of detecting materials as well as the spatial distributions. For example, spin-offs from NASA's multi- and hyperspectral imaging remote sensing technology, developed for earth resources monitoring, are techniques that combine and integrate spectral with spatial methods. Such techniques are finding use, for example, in medicine, agriculture, manufacturing, and forensics, to mention a few. Multispectral or hyperspectral sensors collect the electromagnetic spectrum at dozens or hundreds of wavelength ranges in the visible and near infrared spectra. Spectral resolution of a multispectral sensor is higher and is defined as a measure of the narrowest spectral wavelength that can be resolved by a sensor. Due to hardware limitations (most of the color cameras are RGB cameras) all the spectral information is converted to RGB triplets during image acquisition. It is a projection from an infinite-dimensional color space to a three-dimensional space, or many-to-one projection, which results in colors with different spectral distributions giving the same RGB response. Colors with the same tristimulus data but with different power distributions are called metameristic colors.

The compromise between tristimulus data collection and spectrographic information is the employment of multiple (more than three) color filters with narrow bandwidth mounted either between the lens and the sensor of the camera or in front of the camera lens. The most common procedure is to place the filters in front of the camera lens. Such an apparatus allows to obtain spatial information about the imaged scene at high spectral resolution. The collection and processing of 2D images of the same scene under many spectral (often narrow band-pass) filters, particularly in the visible range, is often referred to as multispectral imaging.

### 16.2.1 Multispectral Imaging Using Rotating Wheels

In recent years, modern spectral image capture systems tend to rely on combinations of CCD cameras with various types of narrow or broad band filters. The images are then processed using common high-capacity computers with software developed to

**Fig. 16.1** Multispectral imaging systems. **a** Camera with rotating wheel and **b** camera with electronically tunable filter in front of lens



properly treat the spectral data. Therefore, capturing multispectral images can be accomplished by swapping narrow band-pass glass filters in front of the camera lens. It is common for such filters to be mounted in a filter wheel. Nowadays, color filters with minimum bandwidth of approximately 10 nm are available off-the-shelf. Different filters and combinations were proposed for different applications. Figure 16.1 illustrates the principle of mounting a filter wheel in front of the camera lens.

Ohta et al. [33] used a film-based system for multispectral image capture. Their system used a mechanical rotating filter wheel with eight gelatin filters to image rigid objects. Only rays within a small wavelength band experience constructive interference and pass through the interference filters. In such use, interference filters offer a large aperture, large field of view, and good optical quality. Tominaga proposed a camera system with six color filters [45], which had six spectral channels of the color filters' fixed wavelength bands.

Fixed-filter systems have intrinsic restrictions: (1) the selection of color filters and the number of filters are limited; (2) filters with a narrow band-pass are difficult to build; (3) moving parts are necessary to select the filters since it is a mechanical system. Due to the latter restriction, time on the order of seconds can be required to step filters in a preset sequence and vibrations of the imaging system may occur. These systems are commonly employed for multispectral imaging of rigid objects where image acquisition time can be long.

### 16.2.2 Multispectral Imaging Using Electronically Tunable Filters

A faster and more flexible way of multispectral imaging involves electronically tunable filters (ETFs). A tunable filter is a device whose spectral transmission can be electronically controlled through the application of voltage or acoustic signals. In addition, the large aperture and imaging capability of these devices represent a distinct advantage over conventional dispersive spectral analysis techniques. Unlike conventional filter wheels, there are no moving parts and no discontinuity in the spectral transmission range, thus providing finer spectral sampling, and rapid and random switching between color bands. Also, ETFs are light weight, making them attractive for airborne or remote sensor platforms.

Electronically tunable filters offer the fastest, most accurate and flexible color filtering techniques that are currently available. The majority of ETFs can be classified

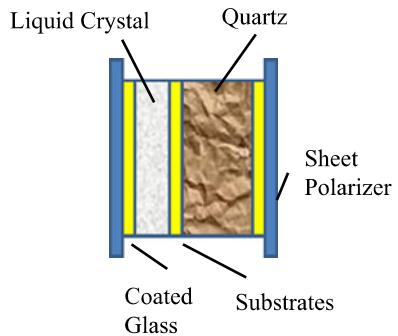
under three operational categories: (1) Acousto-Optical Filters based on diffraction, (2) Fabry–Perot Filters based on optical interference, and (3) Liquid Crystal Filters based on birefringence. Although each of the three types of ETFs is based on different principles of optics, each of them is successful in selecting individual band pass over a continuum of spectral ranges with high speed and accuracy. Figure 16.1 shows two different designs of multispectral imaging systems: (a) a camera with a rotating wheel and (b) a camera with an electronically tunable filter in front of the lens.

The operation of the *Acousto-Optic tunable filter (AOTF)* is based on the interaction of electromagnetic and acoustic waves. The main module of an AOTF is an optically transparent crystal that possesses a certain combination of optical and acoustic properties. While the incoming light falls on the crystal, a radio-frequency acoustic wave is sent to the crystal simultaneously. It is used in creating a refractive index wave within the crystal. The incident beam when passing through the refractive index wave breaks into its component wavelengths. In the end, a single wavelength of light is selected for transmission. Proper design makes one of these wavelengths much more prominent and that becomes the output color of the filter. The wavelength of the filtered light is selected by changing the frequency of the acoustic wave. AOTFs are lightweight and very fast spectral filtering devices. One disadvantage of such devices is the requirement that the incident light be collimated [16].

Another category of electronically tunable filters applies the principle of *optical interference*. A Fabry–Perot cavity is the basic component consisting of two parallel planar surfaces, whose inner face is coated with partially transparent films of high reflectivity, enclosing a rectangular volume of air or some dielectric material. Light enters through one of the partially transparent mirrors and is multiply reflected within the cavity. The multiply transmitted rays interact with each other, creating optical interference effects, which result in the transmission, through the opposite semitransparent mirror, of only one particular wavelength and its harmonics. To block the unwanted harmonics, often two cavities in a row are employed, constituting a *dual tunable Fabry–Perot (DTFP)* device [39]. *Electro-optic Fabry–Perot (EOFP)* devices adjust the bandpass spectrum by varying the refractive index of the cavity through the application of electric potential. Recently, *liquid crystals (LCFP)* are employed as cavity medium. On average, single-cavity ETFs can select the output wavelength out of an input range that is no larger than 100 nm wide. Thus, a cascade of Fabry–Perot cavities is needed in order to have an EOFP that can analyze the entire visible spectrum. Such designs are more costly and have a lower transmission rate (20–50% instead of 90% for a single cavity [39]).

Recently, the Applied Spectral Imaging SpectraCube has been introduced, which is an interferometry-based portable digital camera. This camera is based on the idea that if interference of the color signal is created and measured, the spectrum of the original signal can be recovered applying the *inverse Fourier transform*. With this device, a full 2D array of spectra is captured at once and, unlike filter-based systems, a single exposure is acquired. The spectral resolution of this device can be set higher than most filter-based systems (e.g., about 4 nm), but it also comes

**Fig. 16.2** Principle design of a Liquid Crystal Tunable Filter element (after [14])



at a high expense. Moreover, the single-image acquisition time ranges from 30 to 150 seconds (depending on spatial and spectral resolution and aperture) [13].

The third and most commonly used category of filter devices is *liquid crystal tunable filters (LCTFs)*, which use electrically controlled liquid crystal elements to select a specific visible wavelength of light for transmission through the filter at the rejection of all others. A typical LCTF is built using a stack of polarizers and tunable retardation (birefringent) liquid crystal plates (cp. Fig. 16.2). The LCTF is polarization sensitive. Switching speed is limited by relaxation time of the crystal and is of the order of  $\sim 50$  ms. Special devices can be designed for fast switching ( $\sim 5$  ms) through a short sequence of wavelengths. Spectral resolution, or band pass, of the LCTF is typically of the order of several nm, although a narrower band pass can also be constructed [14].

For a multispectral imaging system employing a LCTF, the camera response  $u_{\lambda_k}$  corresponding to band  $k$  centered at wavelength  $\lambda_k$  within the range,  $\lambda_{k,\min}$  to  $\lambda_{k,\max}$ , can be represented as (compare (16.2))

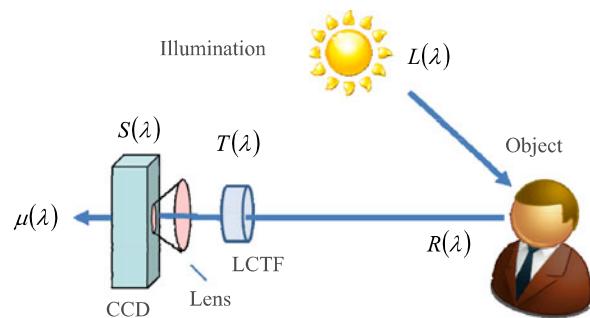
$$u_{\lambda_k} = \int_{\lambda_{k,\min}}^{\lambda_{k,\max}} R_{\lambda_k}(\lambda) L_{\lambda_k}(\lambda) S_{\lambda_k}(\lambda) T_{\lambda_k}(\lambda) d\lambda \quad (16.3)$$

for  $k = 1, 2, \dots, N_B$  where  $k$  indicates the  $k$ th spectral band,  $N_B$  is the total number of bands, and  $T_{\lambda_k}$  is the spectral transmittance of the LCTF.  $(x, y)$  is omitted for simplicity.  $R_{\lambda_k}$  is the spectral surface reflectance of the object (here the face),  $L_{\lambda_k}$  is the spectral distribution of the illumination, and  $S_{\lambda_k}$  is the spectral sensitivity of the camera corresponding to band  $k$ . the imaging process is illustrated in Fig. 16.3.

Nowadays, a considerable variety of Liquid Crystal Tunable Filters, Acousto-Optic Tunable Filters, and Electro-Optic Fabry-Perot is available in the market. Most of them have comparable performance characteristics. Table 16.1 lists typical characteristics of ETFs (after [39]).

Spectroradiometers are a precise alternative to filter-based systems. After light passes through the shutter, it is directed to a concave diffraction grating that breaks up the signal into a photosensitive array and focuses the diffracted signal onto a photosensitive array. These devices have a very high spectral resolution, precision, and stability [13]. Nevertheless, one disadvantage of spectroradiometers is that they

**Fig. 16.3** The camera response is the result of integration of all the factors involved, including the spectral distribution of illumination, reflectance of the object, the transmittance of the filter, and the spectral response of the camera



**Table 16.1** Typical tunable filter characteristics (after [39])

| Attributes                        | AOTF                      | EOFP                          | LCTF                      |
|-----------------------------------|---------------------------|-------------------------------|---------------------------|
| Operating spectral range          | 200–5000 nm               | 400–1550 nm                   | 400–1800 nm               |
| Max width of tunable range        | 700 nm (vis + NIR)        | 100 nm                        | 450 nm (vis + NIR)        |
|                                   | 3900 nm (MIR)             |                               | 950 nm (MIR)              |
| Min. output bandwidth             | 0.4 nm                    | 0.05 nm                       | 5 nm                      |
| Max. output bandwidth             | 50 nm                     | 30 nm                         | 10 nm                     |
| Mean error in central wavelength  | 1 nm (varies with l)      | 1 nm (varies with l)          | 0.5 nm                    |
| Average transmission rate         | 98%                       | 20–50%                        | 20–50%                    |
| Transmission rate over wavelength | constant                  | increases with wavelength     | increases with wavelength |
| Out of band transmission          | 0.05–0.1%                 | 0.5–1%                        | 0.01–0.05%                |
| Tunability time                   | ~15–30 ms                 | ~40 ms (LCFP)<br>~4 ms (DTFP) | ~50 ms                    |
| Incident light limitations        | requires collimated light | none                          | none                      |

measure only single points. Therefore, it is nearly impossible to use them to capture a full scene.

Multispectral imaging systems with electronically tunable filters have been used by several research groups [17, 20, 21, 34, 35, 40, 45]. The Munsell Color Science Laboratory initiated efforts with multispectral images using a LCTF over the visible spectrum, especially for high resolution art portrait reconstruction [20, 21]. They also acquired the Lippmann2000 database [40] that contains spectral images of several objects including faces from 4 Caucasians and 3 East-Asians. This data was acquired by a film camera with approximately 15 to 25 second lapses between exposures and 16 exposures for each person, under flash lighting.

Pan et al. [34, 35] acquired spectral images over the near infrared spectrum (700–1000 nm) and demonstrated that spectral images of faces acquired in the near infrared range can be used to recognize individuals. Until now, not much research has been done using multispectral imaging in the visible domain to address the problem of face recognition, especially with respect to changes in illumination conditions.

The multispectral databases mentioned above either have very few data records or are not in the visible spectrum. In addition, these datasets were not compared with conventional face images by recognition engines.

### **16.2.3 Multispectral Band Selection**

When a large amount of multispectral data has to be analyzed, it is very desirable to reduce the initial information without losing classification accuracy to a significant degree. This reduction can be achieved by two methodologies: feature extraction [4, 11, 12, 18, 23, 24, 29–31, 36, 41, 42, 44] or band selection [6, 26–28, 38, 43, 46].

In feature extraction, a new and reduced data set representing the transformed initial information is obtained, whereas in band selection a subset of relevant data from the original bands is chosen. Compared to feature extraction, band-selection methods identify a subset of the original spectral bands that contains most of the characteristics. Selecting a subset of relevant bands from the original set allows the process of image acquisition to be reduced to a certain number of bands instead of dealing with the entire set of data and, therefore, simplifying image acquisition and analysis.

Feature extraction transfers the data into a lower dimensional space. Features are extracted from the original spectral bands to construct a lower-dimension feature space. Thereby the original data are transformed into the destination feature space through projections such as Projection Pursuit (PP) [31], Principal Component Analysis (PCA) [12], locally linear embedding [41], Isomap [44] and subspace theory [18, 36, 42], wavelet transform [4, 23], and Independent Component Analysis (ICA) [11, 30]. These projections preserve most desired information but change the physical meaning of each spectral band. These methods rank the influence of each single spectral band on the new lower dimensional space. Bands with the highest influence are considered to include more information and are therefore selected [1, 7]. It is nearly impossible to predict the best dimension required for dimension reduction without significant loss of information. In addition, the data is transformed and no longer exists as original data. Some crucial and critical information may have been compromised and distorted.

The goal of band selection is to minimize information loss from the information preservation point of view. Mutual information is a good candidate for band selection as a measure of independence between random variables. For example, Sotoca et al. [43] used conditional entropy as an approximation of mutual information to measure the independent information carried by one band given a sub-band set. Peng et al. [38] argued that maximizing the relevance between each individual band and the class is equivalent to the maximum dependency criterion if one band is selected at a time. Their method needs samples from inside the class, as well as outside of the class, to evaluate the relevance of each individual band with the class. Thus, it is not applicable if no samples outside the class are available. Basically, the method employs an unsupervised band selection criterion to obtain the relevant spectral bands

from a set of sample images, while minimizing the dependent information between spectral bands and maximizing the conditional entropies of the selected bands [43].

In general, the problem of subset selection using numerical techniques for model selection requires two components: a search algorithm and an evaluation criterion.

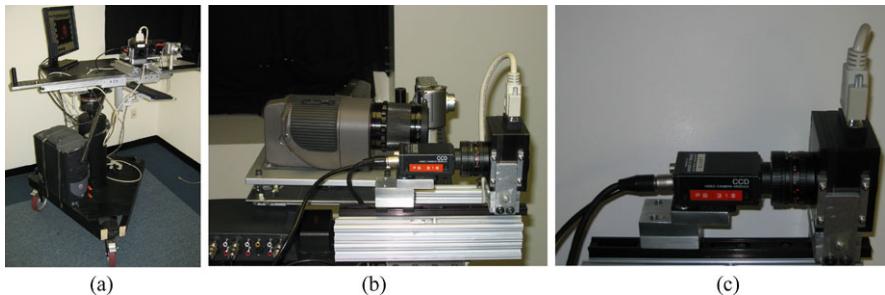
First, an algorithm is needed for the efficient search of the solution space, such as greedy and genetic algorithms. Exhaustive search [25] over the entire feature data set and the branch-and-bound algorithm [5] has rarely been used in the analysis of high-dimension data due to the computational costs even though they lead to an optimal solution. Heuristic search methods such as hill climbing, backward elimination, forward selection, and stepwise selection are commonly used.

Second, a criterion or measure is needed for the comparison and evaluation of competing models to guide the search. The criterion can be based on human perception, which makes it hardware dependent, participant dependent, quite subjective, and also time consuming. Other criteria mentioned in the literature include first or second spectral derivatives [1]. Entropy is interpreted as a measure of stability of each individual wavelength band. Low entropy values correspond to low uncertainties and thus bands with small entropy are considered good feature bands. Bassett and Shen [2] used entropy to measure the difference between different classes.

Commonly, researchers apply band selection with the consideration of classification outputs. In [1], the issue of hyperspectral bands and method selection using unsupervised and supervised methods driven by classification accuracy and computational cost is addressed. Their formulation is more general by optimizing over several methods and the combinations of supervised and unsupervised methods evaluations. One goal of band selection is to identify a reflectance feature that remains invariant when the viewing conditions change. Wang and Angelopoulou [47] propose a technique for extracting color information that is invariant to geometry and incident illumination. They examine the rate of change in reflected intensity with respect to wavelength over the visible part of the electromagnetic spectrum. For diffuse surfaces the only factor that contributes to variations over the wavelength is the albedo of the surface independent of the particular model of reflectance.

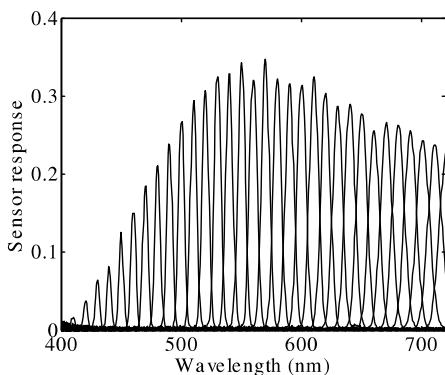
### 16.3 The IRIS-M<sup>3</sup> Face Database

A multispectral and multi-illuminant face database was acquired at the IRIS Lab to support research in multispectral image analysis for face recognition. The database includes indoor and outdoor images taken under controlled and uncontrolled illumination situations. During image acquisition a liquid crystal tunable filter in the visible spectrum was used, which provided narrow band filters at different wavelengths between 400 nm and 720 nm. The multispectral face database has noteworthy characteristics. It is the first database with registered images in the visible, multispectral and thermal modalities, coupled with spectral distributions of the illumination sources used during acquisition. Participants were imaged under various illumination conditions such as halogen light, fluorescent light and day light. Another interesting feature is the large number of multispectral bands available in the database with 25 bands per participant and a total of 82 participants.

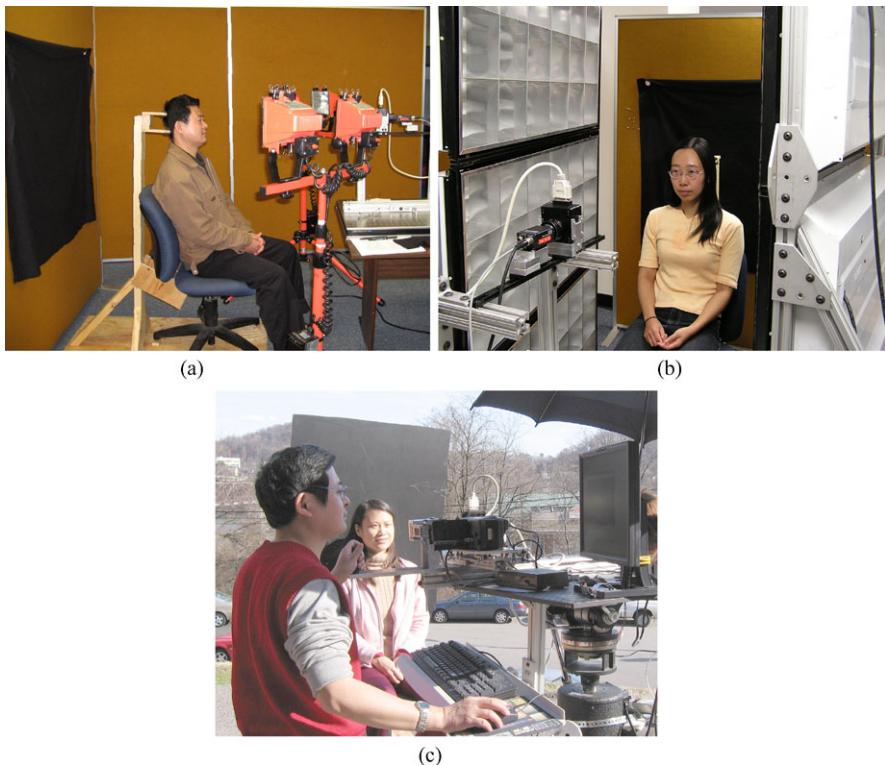


**Fig. 16.4** **a** The all-inclusive multimodal and multispectral mobile imaging system, **b** lateral view of the multimodal imaging system, and **c** the multispectral imaging components (from [10])

**Fig. 16.5** Narrow-band transmittances of the LCTF from 400 nm to 720 nm with 10 nm increments

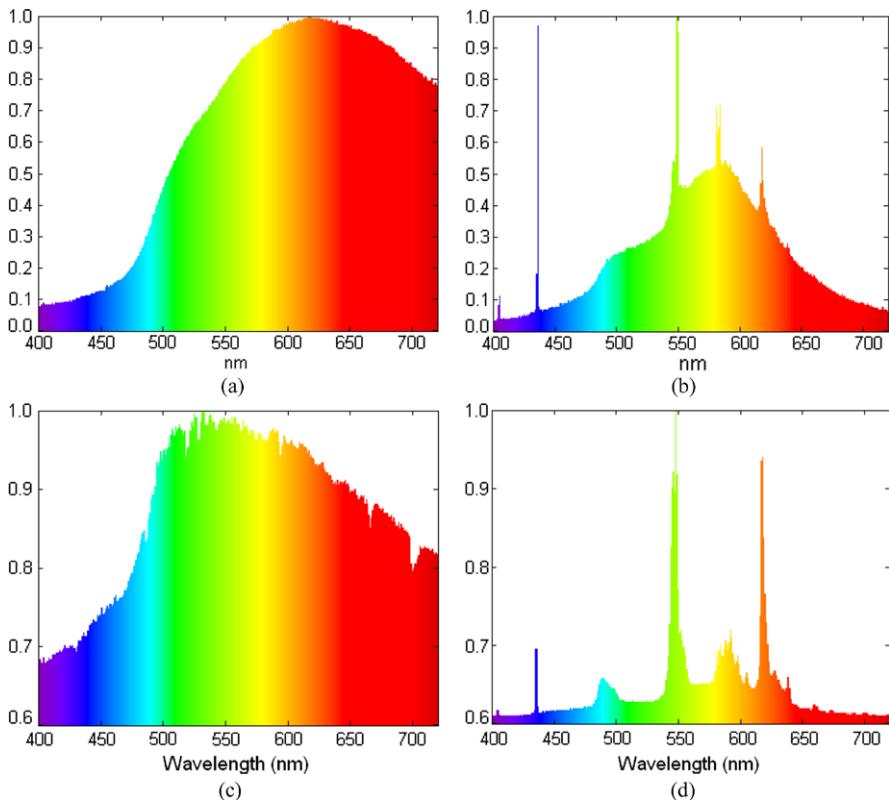


The multispectral imaging system shown in Fig. 16.4 is integrated on a translational platform to acquire well-aligned face images in a short period of time. This allows the participants to maintain their expression and pose. The mobile imaging system shown in Fig. 16.4(a) consists of a multispectral imaging module, a digital RGB camera, a near infrared camera, a spectrometer, a frame grabber and an onboard computer. Figure 16.4(b) shows the lateral view of the multimodal imaging system. The multispectral imaging components shown in Fig. 16.4(c), consist of a Sony XC-75 monochrome camera and a VariSpec liquid crystal tunable filter, which can electronically tune a narrow band filter centered at various wavelengths in visible spectrum. The LCTF provides narrowband filters with a full width-at-half-maximum bandwidth of 7 nm. A maximum of 331 narrow-band multispectral images can be acquired by continuously tuning the LCTF. The aperture of the LCTF is 35 mm and the field of view is  $\pm 7^\circ$ . A wide angle lens is mounted on the monochrome camera (Sony XC-75) and this is coupled with the LCTF through a hardware interconnection. The camera auto-gain is set to 0 dB in order to acquire raw data. The black current of the Sony XC-75 is measured by covering the lens and reading the pixel values of black images. After averaging, the typical black current is 4 to 5 values out of 256. The transmittance of the used LCTF is different at different wavelengths as shown in Fig. 16.5.



**Fig. 16.6** Three illumination setups, **a** quadruple halogen lights with a pair on each side, **b** a pair of fluorescent light panels, and **c** daylight with side illumination (from [10])

Three datasets were acquired with three different illumination scenarios: halogen light, fluorescent light and daylight. The three illumination setups are shown in Fig. 16.6. The quadruple halogen lights with a pair on each side of the participant are shown in Fig. 16.6(a). The second illumination setup was a pair of fluorescent light panels (fluorescent-1) shown in Fig. 16.6(b). We assume that the indoor illuminations, halogen light and fluorescent light, are homogeneously distributed on the face and have stable spectral power when they are lit. The daylight face data was acquired with side illumination due to the fact that many participants were unable to maintain pose or expression with bright sun light shining directly into their eyes. We grouped the outdoor data acquisition into 8 different sessions according to weather conditions and acquisition time. The weather conditions ranged from sunny to cloudy and the passing clouds caused rapid changes in lighting conditions. An outdoor data acquisition setup with side illumination is shown in Fig. 16.6(c). For comparison an additional set of images was acquired with a Canon A80 under another type of fluorescent light (fluorescent-2). The illuminations are characterized during the database collection via the use of a light meter and a spectrometer. An

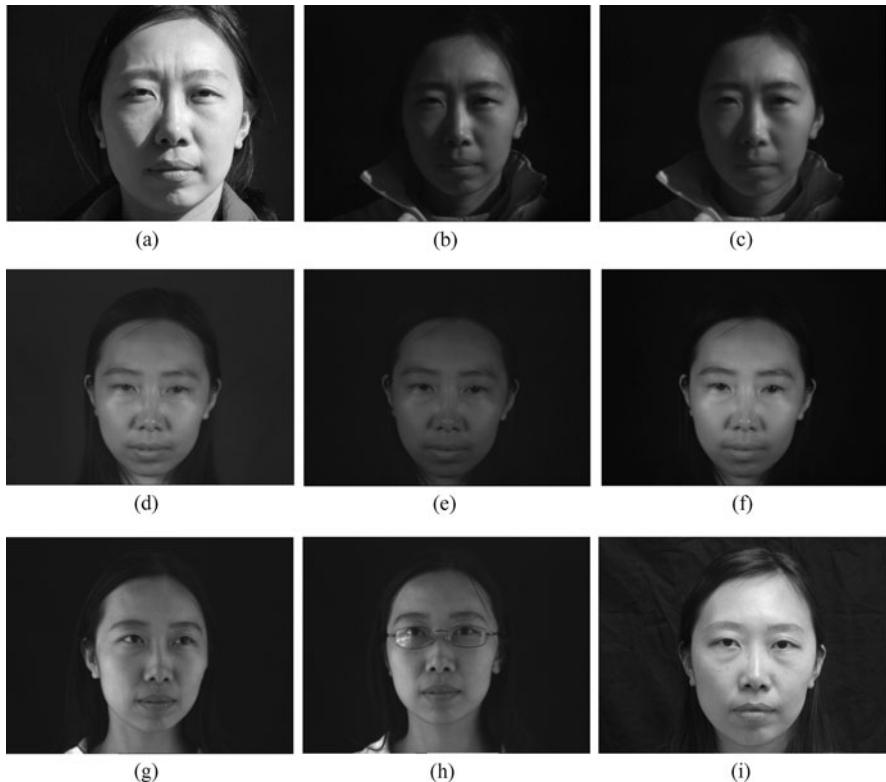


**Fig. 16.7** Normalized spectral power distribution of **a** halogen light **b** fluorescent light, **c** day light, and **d** another fluorescent light (from [10])

EasyView30 light meter was used to measure the illuminance and an Ocean Optics USB2000 Spectrometer was used to measure the irradiance of the illuminant.

The spectral power distributions (SPDs) of four different illuminants used during the data acquisition process are shown in Fig. 16.7. The SPD of halogen light (a) is very smooth and the peak is at the orange part of the spectrum. The SPD of the fluorescent light panel (b) is spiky at certain wavelengths. The day light depicted in Fig. 16.7(c) tends to have more green and blue components in comparison to the other two illuminants. A second fluorescent light was used during the image acquisition process which has a different SPD than the first fluorescent light (see Fig. 16.7(d)).

There are a total of 82 participants of different ethnic groups, ages, facial hair characteristics, and genders in the database with 2624 face images. The corresponding illumination information for each image is recorded using the spectrometer. The image resolution is 640 by 480 pixels and the eye-to-eye distance is about 120 pixels. The database was collected in 11 sessions between August 2005 and May 2006 with some participants being photographed multiple times. Figure 16.8 shows sam-



**Fig. 16.8** Sample images in a data record in the IRIS-M<sup>3</sup> database; **a** side illumination under daylight, **b** band 640 nm multispectral image under daylight, **c** band 720 nm spectral image under daylight, **d** under indoor halogen light, **e** band 640 nm spectral image under indoor halogen light, **f** band 720 nm spectral image under indoor halogen light, **g** under fluorescent with slightly facing her left, **h** under fluorescent with glasses, and **i** under another fluorescent light (from [10])

ples from one data record in the IRIS-M<sup>3</sup> database with variations in lighting conditions and elapsed time. The database contains 76% males and 24% females; the ethnic diversity was defined as a collection of 57% Caucasian, 23% Asian (Chinese, Japanese, Korean and similar ethnicity), 12% Asian Indian, and 8% of African Descent. Figure 16.9 illustrates by example the demographics of the database including different ethnicities, age groups, facial hair characteristics, and genders.

## 16.4 Complexity-Guided Distance-Based Band Selection

In this section, a complexity-guided distance-based band selection method is introduced as a key step in multispectral image processing aimed at improved face recognition. Let the total number of multispectral bands be  $N_B$  and  $\lambda_k$  denotes the central wavelength of the  $k$ th band. The complete set of multispectral bands



**Fig. 16.9** Examples of 6 subjects, **a** male Asian under fluorescent light, **b** female Caucasian under fluorescent light, **c** male Caucasian under fluorescent light, **d** female of African Descent under fluorescent light, **e** male Asian Indian under daylight, **f** female Asian under daylight, **g** male Caucasian under daylight and **h** female of African Descent under daylight (from [10])

is  $\mathcal{B} = \{\lambda_k \mid k = 1, \dots, N_B\}$ . The proposed method automatically searches for an optimal subset  $\mathcal{B}_{\text{opt}} \subseteq \mathcal{B}$  such that the fused images from  $\mathcal{B}_{\text{opt}}$  can outperform conventional broad-band images. To achieve such a goal, the proposed method explores a divergence based distance measure and defines a new redundancy measure to quantitatively describe the correlation among multiple spectral bands. The probabilistic distance measure insures that the selected subset is sufficient for improving recognition rate whereas the redundancy measure insures that the subset is necessary with a minimum number of selected bands.

First, let us define the genuine and imposter sets of each spectral band. In multi-spectral face recognition, a gallery consists of a set of samples  $\{g_1, \dots, g_N\}$ , where  $N$  is the total number of subjects in the gallery. When a probe image  $p_j^k$  collected at the  $k$ th spectral band is presented to a system, it is compared with all the samples in the gallery. The comparison between a probe  $p_j^k$  and each gallery sample  $g_i$  produces a similarity score  $S_{ij}^k$ . These similarity scores can be divided into two groups

for each spectral band, referred to as the genuine  $\mathcal{G}_k$  and imposter  $\mathcal{I}_k$  sets. The genuine and imposter sets are defined as:  $\mathcal{G}_k = \{S_{ij}^k \mid i = j\}$  and  $\mathcal{I}_k = \{S_{ij}^k \mid i \neq j\}$ , respectively. In other words, the genuine set contains the similarity scores with probe and gallery images from the same subject whereas the imposter set consists of similarity scores with the probe and gallery images from different subjects.

Given the genuine and imposter sets of the multiple spectral bands, we first estimate the distributions or equivalently the probability density functions (PDFs) of their respective similarity scores,  $\hat{p}_{G,k}(x)$  and  $\hat{p}_{I,k}(x)$ , and then compute the distance between these two PDFs producing a distance measure  $Q_k$  for each spectral band. In the algorithm proposed in [8, 9], the  $N_{\text{opt}}$  spectral bands with the highest  $Q_k$  values are selected. Two drawbacks are observed. Firstly, the number of bands  $N_{\text{opt}}$  is user specified, which cannot be optimized automatically. Secondly, bands with the highest  $N_{\text{opt}}$  distance measures are selected disregarding the latent redundancy among these bands. In this chapter, we investigate appropriate approaches to select a subset with minimum redundancy and to automatically determine the optimal number of sub-bands.

In [8, 9], it has been demonstrated that an important criterion of selecting the appropriate spectral bands is the separation between the similarity scores of the genuine and imposter set. Therefore, we obtain a set of ordered sub-bands according to the distance measure:  $\mathcal{B} = \{\lambda_{k_i} \mid i = 1, \dots, N_B\}$  with  $Q_{k_1} \geq Q_{k_{i+1}}$ , which means that the first and last sub-bands in  $\mathcal{B}$  have the highest and lowest distance measures, respectively. We start with the spectral band that yields the highest distance measure  $\lambda_{k_1}$ :  $\mathcal{B}_1 = \{\lambda_{k_1}\}$ . The questions are how to select the next band that brings in the most information and when to stop the selection process. These two questions correspond to the problems of selecting the optimal subset and deriving the optimal number of sub-bands. To answer these two questions, we define a quantitative measure that describes the redundancy among bands.

At the  $m$ th iteration and given the current selected subset  $\mathcal{B}_m$ , we obtain the set of candidate sub-bands:  $\overline{\mathcal{B}} = \mathcal{B} - \mathcal{B}_m$ . According to the order of the sub-bands in  $\overline{\mathcal{B}}$ , we add one sub-band  $\lambda_{k_i}$  into  $\mathcal{B}_m$ :  $\mathcal{B}_{m,k_i} = \mathcal{B}_m \cup \{\lambda_{k_i}\}$  and evaluate the redundancy measure of the augmented set:  $R(\mathcal{B}_{m,k_i})$ . We increase  $i$  if the redundancy measure decreases and stop if the redundancy measure begins to increase or all the candidate sub-bands in  $\overline{\mathcal{B}}$  have been examined. If  $R(\mathcal{B}_{m,k_i}) < R(\mathcal{B}_m)$ , the newly elected band  $\lambda_{k_i}$  is added to  $\mathcal{B}_m$  forming a new band set  $\mathcal{B}_{m+1} = \mathcal{B}_{m,k_i}$  and the process iterates. Otherwise, output the current  $\mathcal{B}_{\text{opt}} = \mathcal{B}_m$  as the optimal subset. Algorithm 16.1 describes the detailed steps of the proposed band selection algorithm.

In the election process at each iteration, we start from the sub-band with the highest distance measure in the candidate set and search for the appropriate sub-band according to the descending order of the distance measure. The search is stopped once the redundancy measure of the augmented set begins to increase. In so doing, we are able to locate the sub-band that produces a relatively high distance measure and a relatively low redundancy measure simultaneously.

In the following sections, we will describe the major steps in the proposed algorithm, namely PDF estimation, distance measure computation, and the definition of the redundancy measure.

---

**Algorithm 16.1:** Multispectral band selection
 

---

**Input:**  $\mathcal{G}_k, \mathcal{I}_k$

Compute PDFs of the genuine and imposter sets,  $\hat{p}_{G,k}(x)$  and  $\hat{p}_{I,k}(x)$   
 Compute the probabilistic distance measure  $Q_k$  between  $\hat{p}_{G,k}(x)$  and  
 $\hat{p}_{I,k}(x)$

Obtain  $\mathcal{B} = \{\lambda_{k_i} \mid i = 1, \dots, N_B\}$  with  $Q_{k_i} \geq Q_{k_{i+1}}$   
 Initialize  $R(\Phi) = \infty$ ,  $\mathcal{B}_0 = \Phi$ ,  $\mathcal{B}_1 = \{\lambda_{k_1}\}$ , and  $m = 1$

**while**  $R(\mathcal{B}_m) < R(\mathcal{B}_{m-1})$  **do**

  Initialize  $i = 2$

  Obtain  $\overline{\mathcal{B}} = \mathcal{B} - \mathcal{B}_m$

  Obtain  $\mathcal{B}_{m,k_1} = \mathcal{B}_m \cup \{\lambda_{k_1}\}$  and  $\mathcal{B}_{m,k_2} = \mathcal{B}_m \cup \{\lambda_{k_2}\}$  with  $\lambda_{k_1}, \lambda_{k_2} \in \overline{\mathcal{B}}$

  Compute  $R(\mathcal{B}_{m,k_1})$  and  $R(\mathcal{B}_{m,k_2})$

**while**  $R(\mathcal{B}_{m,k_i}) < R(\mathcal{B}_{m,k_{i-1}})$  **do**

    Obtain  $\mathcal{B}_{m,k_{i+1}} = \mathcal{B}_m \cup \{\lambda_{k_{i+1}}\}$  with  $\lambda_{k_{i+1}} \in \overline{\mathcal{B}}$

    Compute  $R(\mathcal{B}_{m,k_{i+1}})$

    Increase  $i$  by one

**end while**

  Add  $\lambda_{k_{i-1}}$  to  $\mathcal{B}_m$ :  $\mathcal{B}_{m+1} = \mathcal{B}_m \cup \{\lambda_{k_{i-1}}\}$

  Increase  $m$  by one

**end while**

**Output:**  $\mathcal{B}_{\text{opt}} = \mathcal{B}_{m-1}$

---

### 16.4.1 Kernel Density Estimation

Kernel density estimation (KDE) is used to obtain the probability density function because the underlying density can be estimated without assuming a particular form or structure [37]. Formally, kernel estimators smooth out the contribution of each observed data point over a local neighborhood of that data point. Letting  $K()$  denote the Kernel function and  $h$  its smoothing parameter/bandwidth, the estimated density at any point  $x$  is given by [48]:

$$\hat{p}(x) = \frac{1}{Nh} \sum_{i=1}^N K\left(\frac{x - x_i}{h}\right). \quad (16.4)$$

Recall that  $N$  is the total number of subjects in the gallery and that  $S_{ij}^k$  denotes the similarity score between the gallery sample of the  $i$ th subject and the probe of the  $j$ th subject collected from the  $k$ th spectral band. From the similarity scores of various subjects, the distributions of the genuine and imposter sets,  $\hat{p}_{G,k}(x)$  and  $\hat{p}_{I,k}(x)$ , are estimated using KDE:

$$\hat{p}_{G,k}(x) = \frac{1}{Nh_{G,k}} \sum_{i=1}^N K\left(\frac{x - S_{ii}^k}{h_{G,k}}\right), \quad (16.5)$$

$$\hat{p}_{I,k}(x) = \frac{1}{N(N-1)h_{I,k}} \sum_{i=1}^N \sum_{j=1, j \neq i}^N K\left(\frac{x - S_{ij}^k}{h_{I,k}}\right). \quad (16.6)$$

The bandwidth parameter controls the smoothness of the density estimation and determines the trade-off between the bias and variance. Often  $h$  is chosen as to minimize the Asymptotic Mean Integrated Square Error (AMISE) [32]:

$$h_{\text{AMISE}} = \left[ \frac{\rho(K)}{N\mu(K)^2\sigma(p')} \right]^{1/3} \quad (16.7)$$

where  $\rho(K) = 2 \int_{-\infty}^{\infty} x K(x) K_I(x) dx$ ,  $\mu(K) = \int_{-\infty}^{\infty} x^2 K(x) dx$ , and  $\sigma(p') = \int_{-\infty}^{\infty} p'(x)^2 dx$  with  $K_I(x) = \int_{-\infty}^x K(x) dx$ .

### 16.4.2 Probabilistic Distance Measure

Probabilistic distance measures are used here to measure the similarity between the genuine and imposter sets. Probabilistic distance measures have been used in many research areas such as probability and statistics, pattern recognition, information theory, communication, and so on. Here, the symmetric Kullback–Leibler divergence, referred to as the Jeffrey divergence, [22], is used:

$$Q_k = \int [\hat{p}_{G,k}(x) - \hat{p}_{I,k}(x)] \log \frac{\hat{p}_{G,k}(x)}{\hat{p}_{I,k}(x)} dx. \quad (16.8)$$

### 16.4.3 Redundancy Measure

The redundancy measure is derived in the framework of multivariate model selection with penalty on correlation between sub-bands. The core process is a multivariate kernel estimation on the similarity scores of the genuine sets and the evaluation of redundancy based on information complexity.

In the previous section, the PDFs,  $\hat{p}_{G,k}(x)$  and  $\hat{p}_{I,k}(x)$ , of the  $k$ th sub-band are estimated independently of other sub-bands. To incorporate the correlation between sub-bands, we employ multivariate KDE. Given the set  $\mathcal{B}_m$ , the dimension of the multivariate KDE is the number of sub-bands under consideration,  $N_k = |\mathcal{B}_m|$ . Let  $\mathbf{s}$  denote the multivariate vector and  $\mathbf{s}_i$  represent the  $i$ th data point where  $\mathbf{s}_i = [S_{ii}^k]$  with  $\lambda_k \in \mathcal{B}_m$ . The multivariate KDE is given by:

$$\hat{p}(\mathbf{x}) = \frac{1}{N(2\pi)^{N_k/2}} |H|^{-1/2} \sum_{i=1}^N K[(\mathbf{x} - \mathbf{s}_i) H^{-1}(\mathbf{x} - \mathbf{s}_i)]. \quad (16.9)$$

If the sub-bands in  $\mathcal{B}_m$  are independent, the multivariate KDE reduces to:

$$\hat{p}(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \left[ \prod_{k=1}^{N_k} \frac{1}{h_k} K\left(\frac{x_k - S_{ii}^k}{h_k}\right) \right]. \quad (16.10)$$

To evaluate whether the above independent assumption fits the data, we borrow the information complexity (ICOMP) criterion proposed by Bozdogan [3]. The ICOMP criterion is chosen since it has the following advantages. Firstly, it allows the measurement of dependency between the random variables. Secondly, it establishes and provides a trade-off between the fit and the interaction between the parameter estimates and the residuals of a model via the measure of complexity of their respective covariances. The redundancy measure based on the ICOMP is given by:

$$R(\mathcal{B}_m) = -2 \log L(\mathbf{s}_1, \dots, \mathbf{s}_N | h_k) + 2C_{1F}(\hat{\Sigma}), \quad (16.11)$$

where  $\hat{\Sigma}$  denotes the estimated covariance matrix. The first term in the above equation evaluates the fitting error whereas the second term considers the complexity of the estimated covariance matrix, which indicates the correlation among variables. The fitting error term is given by:

$$-2 \log L(\mathbf{s}_1, \dots, \mathbf{s}_N | h_k) = \sum_{i=1}^N \log \hat{p}_{-i}(\mathbf{s}_i), \quad (16.12)$$

where  $\hat{p}_{-i}(\mathbf{x})$  is the leave-one-out estimator:

$$\hat{p}_{-i}(\mathbf{x}) = \frac{1}{N-1} \sum_{j=1, j \neq i}^N \left[ \prod_{k=1}^{N_k} \frac{1}{h_k} K\left(\frac{x_k - S_{jj}^k}{h_k}\right) \right]. \quad (16.13)$$

The  $C_{1F}(\hat{\Sigma})$  is the second order equivalent measure of the complexity of the  $C_1$  measure:

$$C_{1F}(\hat{\Sigma}) = \frac{s}{4} \frac{C_1(\hat{\Sigma})}{(\text{tr}(\hat{\Sigma})/s)^2}, \quad (16.14)$$

where the information complexity  $C_1$  is defined as:

$$C_1(\hat{\Sigma}) = \frac{s}{2} \log \left[ \frac{\text{tr}(\hat{\Sigma})}{s} \right] - \frac{1}{2} \log |\hat{\Sigma}| \quad (16.15)$$

and  $s = \text{rank}(\hat{\Sigma})$ . The estimated covariance matrix is computed by:

$$\hat{\Sigma} = \hat{F}^{-1} \hat{R} \hat{F}^{-1}, \quad (16.16)$$

where  $\hat{F}^{-1}$  is the inverse Fisher information estimation and  $\hat{R}$  is the estimated outer-product form of the Fisher information.

## 16.5 Experimental Results

In order to demonstrate the effectiveness of the band selection algorithm, six sets of experiments are designed, including simulated and real data. The experimental results demonstrate that face recognition rate can be substantially improved over that of the conventional broad-band images for both indoor and outdoor environments. In addition, a simplified multispectral face imaging system can be engineered with reduced acquisition and processing time.

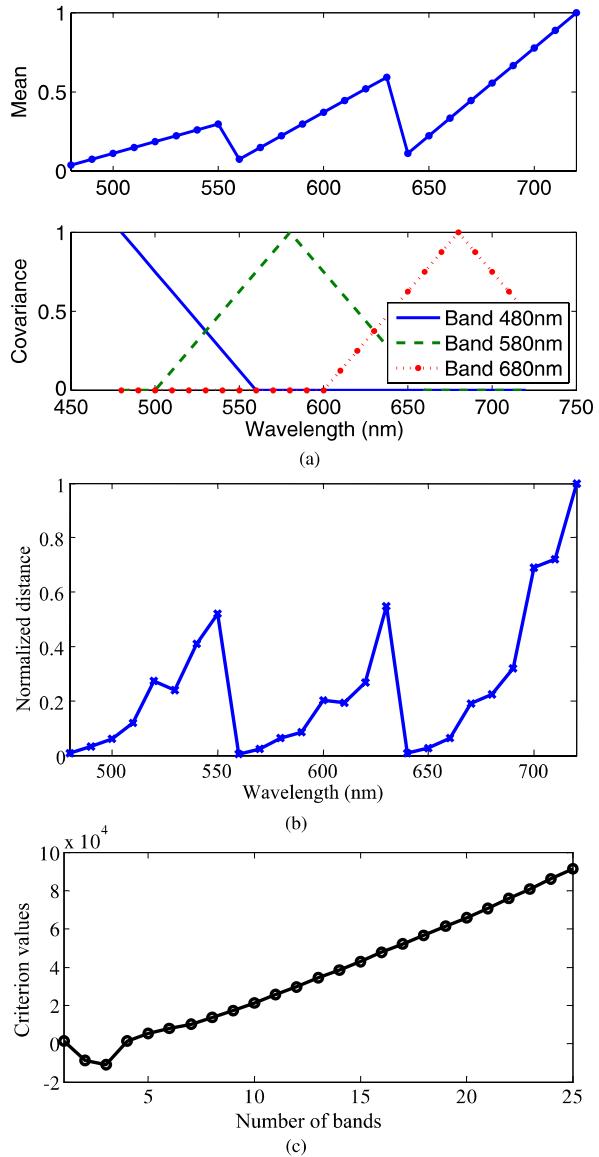
### 16.5.1 Simulated Data

In this section, the performance of the proposed band selection algorithms is studied via simulated data. Simulated data sets are deliberately included for experimentation with controlled parameters, where one can quantitatively compare the results with the known groundtruth values. As mentioned previously, the input parameters to the proposed algorithm are sets of genuine  $\mathcal{G}_k$  and imposter  $\mathcal{I}_k$  similarity scores. To achieve a close resemblance to the real data, similarity scores are computed by the Identix's FaceIt [19] recognition engine based on the IRIS-M<sup>3</sup> face database and their distributions are studied, which leads to the following main observations. (1) The distributions of the similarity scores of the imposter sets collected at different spectral locations share a similar distribution, which can be modeled as a Gaussian with zero mean and unit variance. (2) The distributions of the similarity scores of genuine sets resemble a Gaussian with comparable variances but different means. (3) The correlation between two sub-bands decreases with respect to the increase in the spectral distance between the sub-bands.

Based on these observations, the simulated data is designed as follows. The center wavelength of simulated sub-spectral bands are distributed uniformly in a spectral range of 480 nm to 720 nm with a 10 nm increment. The similarity scores of the imposter sets for all sub-bands are drawn from a Gaussian with zero mean and unit variance. In a Matlab implementation, this is done by calling the function *randn()* with default settings. Considering the correlation of the genuine similarity scores among sub-bands, the scores are drawn from a multivariate Gaussian. Its mean and covariance values determine the system's behavior and are the controlled parameters in the simulated experiments. The specific mean and covariance values used in the experiments will be given in the discussions regarding the experimental results. Once the mean and covariance values are designed, 500 samples are drawn from the multivariate Gaussian for each set. In a Matlab implementation, this is done by calling the function *mvnrnd()* with given mean and covariance.

The mean and covariance values of the simulated data for the first experiment are shown in Fig. 16.10(a). The data is designed such that the optimal number of bands is three and they are 720 nm, 630 nm, and 550 nm, depicted as the local peaks in Fig. 16.10(a). Multiple peaks are deliberately introduced in the design of this experiment to test the ability of the proposed algorithm in picking up the correct

**Fig. 16.10** **a** The mean and covariance of the simulated data. The correlation values are shown for bands 480 nm, 580 nm, and 680 nm. The correlation with a given band decreases linearly as the wavelength of the sub-band moves away from the center wavelength of the given band. **b** Normalized distance measure values for 25 bands with three local peaks. **c** Redundancy measures for different number of bands with three bands having the minimal value



number of optimal bands. In addition, the correlation of these data sets is designed such that these peaks are uncorrelated, resulting in an optimal number of sub-bands of three. The computed distance values are shown in Fig. 16.10(b), where we observe three peaks, which agrees with the actual optimal bands. This verifies the ability of the proposed algorithm in correctly locating the potential optimal bands. Figure 16.10(c) shows the redundancy measure with a minimum value at three, suggesting that the three peaks are sufficient to represent the remaining sub-bands. This

agrees with the structure of the simulated data and verifies the ability of the proposed algorithm in not only locating the optimal bands but also selecting the correct number of bands to form a sufficient and necessary subset.

Although, by design, bands 710 nm and 700 nm produce higher distance measures, they are highly correlated with band 720 nm. Therefore, they are excluded from the optimal subset selected via the newly proposed redundancy measure. In comparison, the band ranking algorithm described in [8, 9] selects 700 nm, 710 nm, and 720 nm if  $N_{\text{opt}} = 3$  is given. The more informative bands 550 nm and 630 nm are not selected. The proposed algorithm selects the optimal bands and decides the number of bands with consideration of both distance values and the correlation among the bands.

In the second experiment, the correlation among spectral bands is increased, as shown in Fig. 16.11(a). With the increased correlation, we expect to see a decreased number of selected sub-bands. Figure 16.11(b) shows the distance measure values from these 25 bands, and the corresponding redundancy measure is given in Fig. 16.11(c). Disregarding the four peaks illustrated in Fig. 16.11(b), only two sub-bands 710 nm and 530 nm are selected primarily because of the increased correlation among sub-bands.

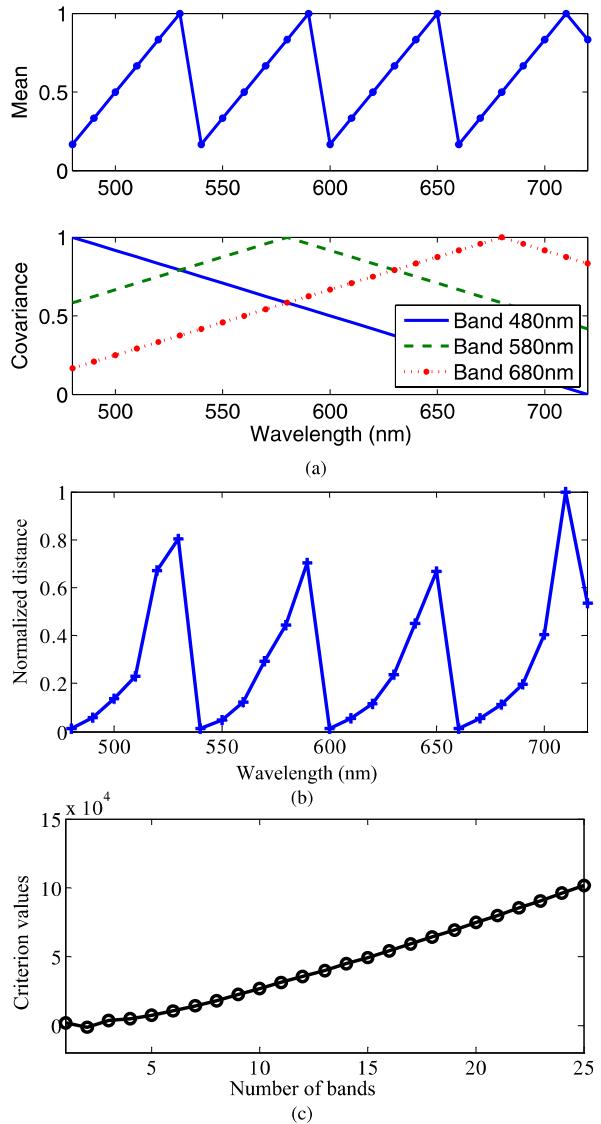
The experimental results based on simulated data clearly illustrate the power of the proposed selection process. Instead of simply ranking multiple spectral bands according to their probabilistic distance measure, the proposed algorithm can adaptively and automatically select the most representative and informative bands according to the recognition capacity of the bands and their correlations.

### **16.5.2 Real Data**

That selecting the optimal spectral bands from a series of multispectral images under given illuminations improves face recognition performance can be shown in the following experiments with real data. Four experiments are designed to investigate the recognition performances of fused images from the selected band/bands in comparison with conventional broad-band images. In real world situations, face images are frequently acquired under different lighting conditions and compared with the database images. It is reasonable and important to study the situation that the illuminations for gallery and probe images are different. The IRIS-M<sup>3</sup> face database is used because it contains gallery and probe images collected under various illuminations. According to the available lighting sources in the IRIS-M<sup>3</sup> face database, the following four sets of experiments were conducted with the gallery and probe sets collected from different illuminations. Table 16.2 lists the experimental conditions.

In these experiments, similarity scores are obtained via Identix's FaceIt [19], a well-known recognition engine. Jeffrey divergence values are normalized between 0 and 1 for clear illustration. If more than two bands are selected, for example, Haar wavelet-based pixel-level fusion [15], can be applied for the fusion of images from the selected sub-bands. Given the registered narrow-band images from the selected

**Fig. 16.11** **a** The mean and covariance of the simulated data. The correlation values are shown for bands 480 nm, 580 nm, and 680 nm. Spectral bands are highly correlated. **b** Normalized distance measure values for 25 bands with four local peaks. **c** Redundancy measures for different number of bands with two bands having the minimal value



spectral range, two-dimensional discrete wavelet decomposition is performed on each image to obtain the wavelet approximation coefficients and detail coefficients. The coefficients in inverse wavelet transform for fused image are obtained by choosing the maximum among each type of coefficients. The two-dimensional discrete wavelet inverse transform is then performed to construct the fused image.

In addition to the rank-one recognition rate, a numerical measure is used to evaluate the recognition performance. To enable quantitative comparison of the overall performances at different ranks, a mapping operation projecting the multi-index

**Table 16.2** Experimental conditions including the description of the gallery lighting, probe lighting, spectral range, and the number of sub-bands

|              | Gallery     | Probe        |                | Increment | Num. of bands |
|--------------|-------------|--------------|----------------|-----------|---------------|
|              |             | Illumination | Spectral range |           |               |
| Experiment 1 | Fluorescent | Halogen      | 480 nm–720 nm  | 10 nm     | 25            |
| Experiment 2 | Day light   | Halogen      | 480 nm–720 nm  | 10 nm     | 25            |
| Experiment 3 | Fluorescent | Day light    | 480 nm–720 nm  | 20 nm     | 13            |
| Experiment 4 | Halogen     | Day light    | 480 nm–720 nm  | 20 nm     | 13            |

CMC curve to a single number, CMCM, is defined as:

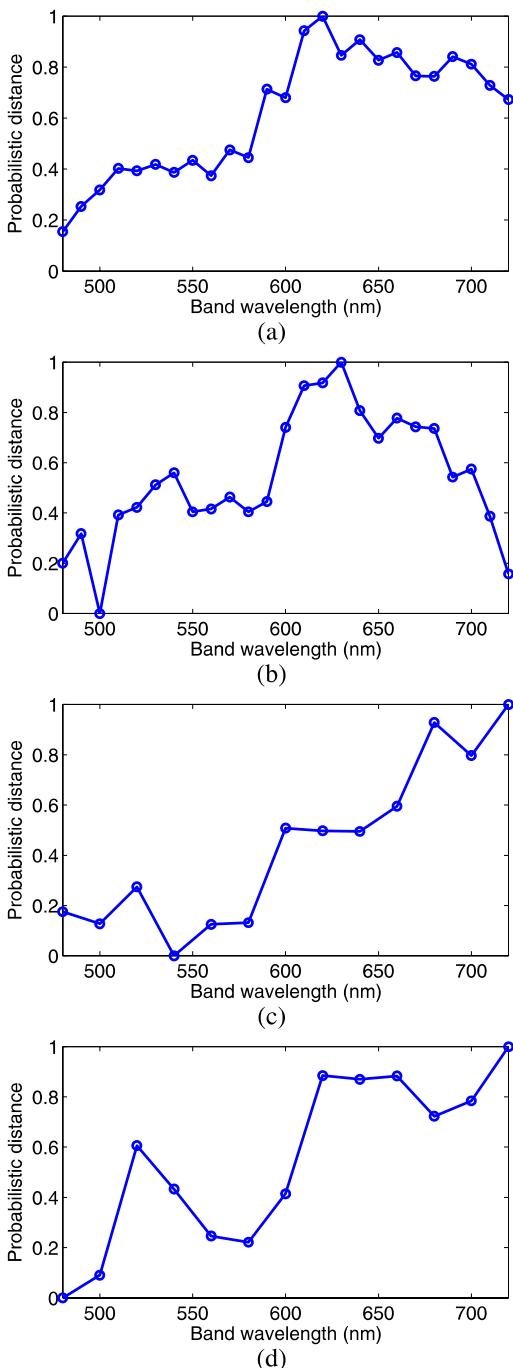
$$\text{CMCM} = \sum_{r=1}^N \frac{C_r}{rN}, \quad (16.17)$$

where  $r$  represents the rank number, and  $C_r$  denotes the number of probe images that can be correctly identified at and below rank  $r$ . Recall that  $N$  denotes the total number of subjects in the gallery. The factor  $1/r$  can be viewed as a weight, which decreases monotonously as  $r$  increases. As a result, the rank-one recognition rate is dominant and contributes the most to the value of CMCM. A better face recognition performance is indicated by a higher CMCM value, which varies between 0 and 1.

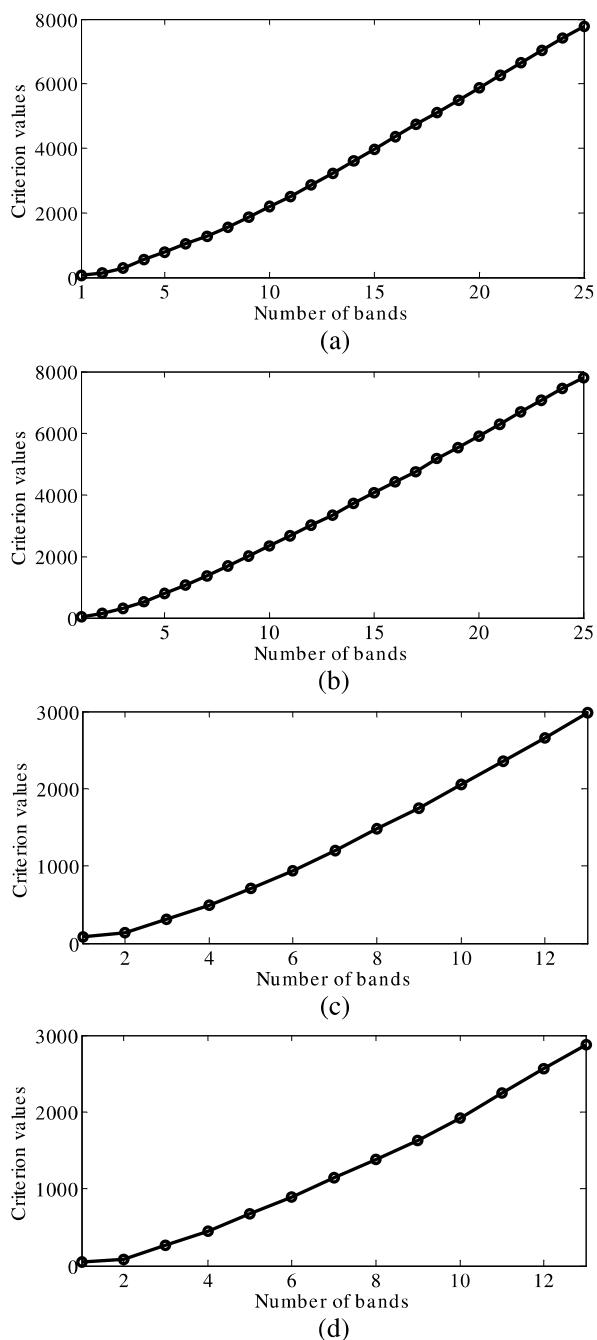
In Fig. 16.12, the normalized probability distance is given whereas the redundancy measure via ICOMP calculation of each possible number of selected bands is given in Fig. 16.13. For all experiments, the proposed algorithm selected one band. The selected bands are 610 nm, 610 nm, 720 nm, and 720 nm for the four experiments, respectively. The reason why only one band is selected lies in the fact that the correlation among all the bands is relatively high. To validate the selection results, face recognition performances including the rank-one recognition rate and CMCM values of the selected bands are tested and given in Table 16.3 in comparison with those of the conventional broad-band images. The images from a single selected band outperform the conventional broad-band monochromatic images. Taking Experiment 1 as an example, the recognition performance is improved by relatively  $9.7\%((97.14 - 88.56)/88.56 \times 100\% = 9.7\%)$  of rank-one rate and by  $4.5\%((98.57 - 94.28)/94.28 \times 100\% = 4.5\%)$  of the CMCM value.

Table 16.3 also lists the performance comparison between the proposed algorithm and the reference algorithm developed in [8, 9]. For Experiment 2 and 4, the images from a single band selected by the proposed algorithm outperform the fused images from three selected bands via the reference algorithm. For the other two experiments (Experiment 1 and 3), their performances are comparable. Note that image fusion helps to reduce image noise, which may also lead to improved recognition performance. This explains the observed comparable performance in Experiments 1 and 3. Therefore, we could conclude that the proposed algorithm is capable of identifying the most concise subset with the most informative spectral bands.

**Fig. 16.12** Normalized distance measures for multiple spectral bands:  
**a** Experiment 1,  
**b** Experiment 2,  
**c** Experiment 3, and  
**d** Experiment 4



**Fig. 16.13** Redundancy measure for different number of selected sub-bands:  
**a** Experiment 1,  
**b** Experiment 2,  
**c** Experiment 3, and  
**d** Experiment 4



**Table 16.3** Rank-one and CMCM recognition rates of the monochromatic conventional broad-band image, the fused narrow-band images via the reference algorithm [8, 9], and the selected narrow-band images via the proposed algorithm

|                     | The proposed algorithm | Broad-band | Improvement | Reference [8, 9] | Improvement |
|---------------------|------------------------|------------|-------------|------------------|-------------|
| <b>Experiment 1</b> |                        |            |             |                  |             |
| Rank-one            | 97.14                  | 88.56      | 9.7         | 97.14            | 0.0         |
| CMCM                | 98.57                  | 94.28      | 4.5         | 98.57            | 0.0         |
| <b>Experiment 2</b> |                        |            |             |                  |             |
| Rank-one            | 65.17                  | 57.15      | 15.0        | 62.86            | 3.7         |
| CMCM                | 74.06                  | 70.20      | 5.5         | 72.11            | 2.7         |
| <b>Experiment 3</b> |                        |            |             |                  |             |
| Rank-one            | 97.14                  | 94.28      | 3.0         | 97.14            | 0.0         |
| CMCM                | 97.57                  | 95.36      | 3.4         | 98.57            | -1.0        |
| <b>Experiment 4</b> |                        |            |             |                  |             |
| Rank-one            | 54.29                  | 48.57      | 11.8        | 48.57            | 11.8        |
| CMCM                | 64.79                  | 57.28      | 13.1        | 60.84            | 6.5         |

## 16.6 Conclusions

In this chapter, the fundamentals of multispectral imaging and its applications to face recognition were introduced. Variation in illumination dramatically degrades face recognition performance. Narrow-band sub-spectral images were used instead of conventional broad-band images to improve recognition performance. A spectral band selection algorithm was developed to choose the optimal band images under given illumination conditions. From the experiments, the spectral bands of 610 nm and 720 nm are the optimal choice for probes under indoor halogen light and varying daylight, respectively. The selected optimal spectral bands are consistent with those specified by physics analysis with known system configuration and illumination characteristics and result in a 3%–15% improvement in face recognition rate in comparison with that of conventional broad-band images. In addition, the optimal set ensures a minimum amount of acquisition and processing time for fast face recognition by selecting the most informative and independent sub-bands.

**Acknowledgements** This work was supported in part by the DOE University Research Program in Robotics under grant DOE-DEFG02-86NE37968 and NSF-CITeR grant 01-598B-UT.

## References

1. Bajcsy, P., Groves, P.: Methodology for hyperspectral band selection. *Photogramm. Eng. Remote Sens.* **70**, 793–802 (2004)
2. Bassett, E.M., Shen, S.S.: Information theory-based band selection for multispectral systems. In: SPIE, vol. 3118, pp. 28–35 (1997)

3. Bozdogan, H.: Akaike's information criterion and recent developments in information complexity. *J. Math. Psychol.* **44**, 62–91 (2000)
4. Bruce, L., Koger, C., Li, J.: Dimensionality reduction of hyperspectral data using discrete wavelet transform extraction. *IEEE Trans. Geosci. Remote Sens.* **40**(10), 2331–2338 (2002)
5. Brusco, M.J.: An enhanced branch-and-bound algorithm for a partitioning problem. *Br. J. Math. Stat. Psychol.* **56**, 83–92 (2003)
6. Bruzzone, L., Roli, F., Serpico, S.B.: An extension of the Jeffreys–Matusita distance to multi-class cases for feature selection. *IEEE Trans. Geosci. Remote Sens.* **33**(6), 1318–1321 (1995)
7. Chang, C.I., Du, Q., Sun, T.S., Althouse, M.L.G.: A joint band prioritization and band decorrelation approach to band selection for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **37**(6), 2631–2641 (1999)
8. Chang, H., Yao, Y., Koschan, A., Abidi, B., Abidi, M.: Spectral range selection for face recognition under various illuminations. In: *IEEE Int'l Conf. on Image Processing*, San Diego, CA, October 2008
9. Chang, H., Yao, Y., Koschan, A., Abidi, B., Abidi, M.: Improving face recognition via narrow-band spectral range selection using Jeffrey divergence. *IEEE Trans. Inf. Forensics Secur.* **4**(1), 111–122 (2009)
10. Chang, H., Koschan, A., Abidi, B., Abidi, M.: Fusing continuous spectral images for face recognition under indoor and outdoor illuminants. *Mach. Vis. Appl.* **21**, 201–215 (2010)
11. Du, H., Qi, H., Wang, X., Ramanath, R., Snyder, W.E.: Band selection using independent component analysis for hyperspectral image processing. In: *Applied Imagery Pattern Recognition Workshop*, pp. 93–98 (2003)
12. El-ghazawi, T., Kaewpijit, S., Le Moigne, J.: Parallel and adaptive reduction of hyperspectral data to intrinsic dimensionality. In: *IEEE Int'l Conf. on Cluster Computing* (2001)
13. Finlayson, G.D., Morovic, P.M., Hordley, S.D.: Using the spectracube for multispectral imaging. In: *2nd Conference on Color in Imaging, Vision and Graphics*, pp. 268–274 (2004)
14. Gat, N.: Imaging spectroscopy using tunable filters: a review. In: *SPIE*, vol. 4056, pp. 50–64 (2000)
15. Gonzalez, R., Woods, R.: *Digital Image Processing*. Prentice Hall, New York (2004)
16. Gottlieb, M.S.: Acousto-optic tunable filters. In: *Design and Fabrication of Acousto-Optic Devices*, pp. 197–284. Marcel Dekker, New York (1994)
17. Hardeberg, J.Y., Schmitt, F., Brettel, H.: Multispectral image capture using a liquid crystal tunable filters. *Opt. Eng.* **41**(10), 2532–2548 (2002)
18. Healey, G., Slater, D.: Invariant recognition in hyperspectral images. In: *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 438–443, Ft. Collins, CO, June 1999
19. <http://www.identix.com/pages/101-faceit-sdk>
20. Imai, F.H., Berns, R.S.: High-resolution multispectral image archives: a hybrid approach. In: *IS&T SID Sixth Color Imaging Conf.*, pp. 185–189 (1998)
21. Imai, F.H., Rosen, M.R., Berns, R.S.: Multi-spectral imaging of a van Gogh's self-portrait at the National Gallery of Art, Washington, D.C. In: *IS&T PICS*, pp. 185–189 (2001)
22. Jeffreys, H.: An invariant form for the prior probability in estimation problems. *Proc. R. Soc.* **186**, 453–461 (1946)
23. Jiang, L., Mann, B., Mathur, A.: Wavelet transform for dimensionality reduction in hyperspectral linear unmixing. In: *IEEE Int'l Geoscience and Remote Sensing Symposium*, vol. 6, pp. 3513–3515 (2002)
24. Jimenez, L.O., Landgrebe, D.A.: Supervised classification in high dimensional space: geometrical statistical and asymptotical properties of multivariate data. *IEEE Trans. Syst. Man Cybern.* **28**(1), 39–54 (1998)
25. Kohavi, R., John, G.H.: Wrappers for feature subset selection. *Artif. Intell.* **97**, 273–324 (1997)
26. Koller, D., Sahami, M.: Towards optimal feature selection. In: *Int'l Conf. on Machine Learning*, pp. 284–292, Bari, Italy, July 1996
27. Korycinski, D., Crawford, M.M., Barnes, J.W.: Adaptive feature selection for hyperspectral data analysis using a binary hierarchical classifier and tabu search. In: *Int'l Geoscience and Remote Sensing Symposium*, pp. 297–299 (2003)

28. Kudo, M., Sklansky, J.: Comparison of algorithms that select features for pattern classifiers. *Pattern Recognit.* **33**(1), 25–41 (2000)
29. Kuman, S., Ghosh, J., Crawford, M.M.: Best basis feature extraction algorithms for classification of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* **39**(7), 1368–1379 (2001)
30. Lennon, M., Mercier, G., Mouchot, M., Hubert-Moy, L.: Independent component analysis as a tool for the dimensionality reduction and the representation of hyperspectral image. In: IEEE Int'l Geoscience and Remote Sensing Symposium, vol. 6, pp. 2893–2895 (2001)
31. Liu, C., Wechsler, H.: A unified Bayesian framework for face recognition. In: IEEE Int'l Conf. on Image Processing, pp. 151–155, Chicago, IL, October 1998
32. Mugdadi, A.R., Munthali, E.: Relative efficiency in kernel estimation of the distribution function. *J. Stat. Res.* **37**(2), 203–218 (2003)
33. Ohta, N., Takahashi, K., Urabe, H., Miyagawa, T.: Image simulation by use of a laser color printer. *O plus E* **22**, 57–64 (1981)
34. Pan, Z., Healey, G., Prasad, M., Tromber, B.: Face recognition in hyperspectral images. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(12), 1552–1560 (2003)
35. Pan, Z., Healey, G., Prasad, M., Tromberg, B.: Hyperspectral face recognition under variable outdoor illumination. In: SPIE, vol. 5425, pp. 520–529 (2004)
36. Parkkinen, J., Oja, E., Jaaskelainen, T.: Color analysis by learning subspaces and optical processing. In: Int'l Conf. on Neural Networks (1988)
37. Parzen, E.: On estimation of a probability density function and model. *Ann. Math. Stat.* **33**(3), 1065–1076 (1962)
38. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(8), 1226–1238 (2005)
39. Poger, S., Angelopoulou, E.: Selecting components for building multispectral sensors. In: IEEE CVPR Technical Sketches (2001)
40. Rosen, M., Jiang, X.: Lippmann2000: a spectral image database under construction. In: Int'l Symposium on Multispectral Imaging and Color Reproduction for Digital Archives, pp. 117–122 (1999)
41. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**, 2323–2326 (2000)
42. Slater, D., Healey, G.: Physics-based model acquisition and identification in airborne spectral images. In: Int'l Conf. on Computer Vision, pp. 257–262, Vancouver, British Columbia, Canada, July 2001
43. Sotoca, J.M., Pla, F., Klaren, A.C.: Unsupervised band selection for multispectral images using information theory. In: Int'l Conf. on Pattern Recognition, vol. 3, pp. 510–513, Cambridge, UK, August 2004
44. Tenenbaum, J.B., Silva, V., Langford, J.C.: A global framework for nonlinear dimensionality reduction. *Science* **290**, 2319–2323 (2000)
45. Tomonaga, S.: A multi-channel vision system for estimating surface and illuminant functions. *J. Opt. Soc. Am. A* **13**, 2163–2173 (1996)
46. Vidal-Naquet, M., Ullman, S.: Object recognition with informative features and linear classification. In: IEEE Int'l Conf. on Computer Vision, pp. 281–288, Nice, France, October 2003
47. Wang, H., Angelopoulou, E.: Sensor band selection for multispectral imaging via average normalized information. *J. Real-Time Image Process.* **1**(2), 109–121 (2006)
48. Wasserman, L.: All of Statistics: A Concise Course in Statistical Inference. Springer Texts in Statistics (2005)

# Chapter 17

## Face Recognition Using 3D Images

**I.A. Kakadiaris, G. Passalis, G. Toderici, E. Efraty, P. Perakis, D. Chu,  
S. Shah, and T. Theoharis**

### 17.1 Introduction

Our face is our password—face recognition promises to revolutionize the way we identify individuals in a nonintrusive and convenient manner. Even though research in face recognition has spanned over nearly three decades, only 2D systems, with limited adoption to practical applications, have been developed so far. The primary reason behind this is the low accuracy of 2D face recognition systems in the presence of: (i) pose variations between the gallery and probe datasets, (ii) variations in lighting, and (iii) variations in the presence of expressions and/or accessories. The above conditions generally arise when noncooperative subjects are involved, which is the very case that demands accurate recognition.

Face recognition using 3D images was introduced in order to overcome these challenges. It was partly made possible by significant advances in 3D scanner technology. However, even 3D face recognition has faced significant challenges which have hindered its adoption for practical applications. The main problem of 3D face recognition is the high cost and fragility of 3D scanners. Over the last seven years, our research team has focused on exploring the usefulness of 3D data and the development of models for face recognition (under the general name URxD).

In this chapter, we present advances that aid in overcoming the challenges encountered in 3D face recognition. First, we present a fully automatic 3D face recognition system, UR3D, which has been proven to be robust under variations in expressions. The fundamental idea of this system is the description of facial data using an Annotated Face Model (AFM). The AFM is fitted to the facial scan using

---

I.A. Kakadiaris (✉) · G. Passalis · G. Toderici · E. Efraty · P. Perakis · D. Chu · S. Shah ·  
T. Theoharis

Computational Biomedicine Lab, Department of Computer Science, University of Houston,  
Houston, TX 77204, USA  
e-mail: [ioannisk@grip.cis.upenn.edu](mailto:ioannisk@grip.cis.upenn.edu)

G. Passalis · P. Perakis · T. Theoharis  
Computer Graphics Laboratory, Department of Informatics and Telecommunications, University  
of Athens, Ilisia 15784, Greece

a subdivision-based deformable model framework. The deformed model captures the details of an individual's face and represents this 3D geometry information in an efficient 2D representation by utilizing the model's parametrization. This representation is analyzed in the wavelet domain and the associated wavelet coefficients define the metadata that are used for comparing the different subjects. These metadata are both compact and descriptive. This approach that involves geometric modeling of the human face allows greater flexibility, better understanding of the face recognition issues, and requires no training.

Second, we demonstrate how pose variations are handled in 3D face recognition. The 3D scanners that are used to obtain facial data are usually nonimmersive which means that only a partial 3D scan of the human face is obtained, particularly so in noncooperative, practical conditions. Thus, there are often missing data of the frontal part of the face. This can be overcome by identifying a number of landmarks on each 3D facial scan thereby allowing correct registration with the AFM, independent of the original pose of the face. For nonfrontal scans, missing data can be added by exploiting facial symmetry, assuming that at least half of the face is visible. This is achieved by improving the subdivision-based deformable model framework to allow symmetric fitting. Symmetric fitting alleviates the missing data problem and facilitates the creation of geometry images that are pose invariant. Another alternative to tackle the missing data problem is to attempt recognition based on the facial profile; this approach is particularly useful in recognizing car drivers from side view images. In this approach, the gallery includes facial profile information under different poses, collected from subjects during enrollment. These profiles are generated by projecting the subjects' 3D face data. Probe profiles are extracted from the input images and compared to the gallery profiles.

Finally, we demonstrate how the problems related to the cost of 3D scanners can be mitigated through hybrid systems. Such systems employ 3D scanners for the enrollment of subjects, which can take place in a few specialized locations, and 2D cameras at points of authentication, which can be multiple and dispersed. It is practical to adopt this approach if hybrid systems can improve the accuracy of a 2D system. During enrollment, 2D+3D data (2D texture and 3D shape) are used to build subject-specific annotated 3D models. To achieve this, an AFM is fitted to the raw 2D+3D data using a subdivision-based deformable framework. A geometry image representation is then extracted using the parametrization of the model. During the verification phase, a single 2D image is used as the input to map the subject-specific 3D AFM. Given the pose in the 2D image, an Analytical Skin Reflectance Model (ASRM) is then applied to the gallery AFM to transfer the lighting from the probe to the texture in the gallery. The matching score is computed using the relit gallery texture and the probe texture. This hybrid method surpasses the accuracy of 2D face recognition system in difficult datasets.

### 17.1.1 3D Face Recognition

In recent years, several 3D face recognition approaches have been proposed that offer increased accuracy and resilience to pose and illumination variations when compared to 2D approaches. The limitations of 2D approaches were highlighted in the Face Recognition Vendor Test 2002 study. However, the advantages of 3D face recognition were not evident since most 3D approaches had not been extensively validated due to the non-availability of 3D databases. This is evident in the surveys of the 3D face recognition field given by Bowyer et al. [8], Chang et al. [13] and Scheenstra et al. [57]. To address this issue, NIST introduced the Face Recognition Grand Challenge and Face Recognition Vendor Test 2006 [21] and released two publicly available multimodal (3D and 2D) databases, FRGC v1 and FRGC v2.

On FRGC v1, a database that contains over 900 frontal scans without any facial expressions, Pan et al. [46] reported 95% rank-one recognition rate using a PCA approach, while Russ et al. [56] reported a 98% verification rate. Our approach achieved a 99% rank-one recognition rate [29].

On FRGC v2, a database that contains over 4000 frontal scans with various facial expressions, Chang et al. [11, 12] examined the effects of facial expressions using two different 3D recognition algorithms. They reported a 92% rank-one recognition rate. The same rank-one recognition rate (92%) was also reported by Lu et al. [40]. In their approach, a Thin Plate Spline (TPS) was used to learn expression deformation from a control group of neutral and non-neutral scans. Husken et al. [28] presented a multimodal approach that uses hierarchical graph matching (HGM). They extended their HGM approach from 2D to 3D but the reported 3D performance was poorer than the 2D equivalent. The fusion of the two approaches, however, provided competitive results, a 96.8% verification rate at 0.001 False Acceptance Rate (FAR), compared to 86.9% when using the 3D only. Al-Osaimi et al. [1] used a PCA subspace, referred to as the expression deformation model, to analyze facial deformations from 3D data. They reported an average (over ROC I, II and III experiments) verification rate of 94.2% at 0.001 FAR. Maurer et al. [43] also presented a multimodal approach tested on the FRGC v2 database, and reported a 87% verification rate at 0.01 FAR. In our initial work on this database [49], we analyzed the behavior of our approach in the presence of facial expressions. The improvements presented in our subsequent work [30] allowed us to overcome the shortcomings of this approach. Our method, using only 3D data, achieved 97% rank-one recognition and an average (over ROC I, II and III experiments) verification rate of 97.1% at 0.001 FAR.

### 17.1.2 3D Face Recognition from Partial Scans: UR3D-PS

Even though the majority of the 3D face recognition approaches focus on full frontal scans, there are several approaches that focus on partial scans (that are prone to missing data). Lu et al. [38, 39, 41], in a series of studies, presented methods to locate

the positions of the corners of the eyes and mouth, and the tips of the nose and chin, based on a fusion scheme of shape index on range maps and the “cornerness” response on intensity maps. They also developed a heuristic method based on cross-profile analysis to locate the nose tip more robustly. Candidate landmark points were filtered out using a static (nondeformable) statistical model of landmark positions. Although they report a 90% rank-one matching accuracy in an identification experiment, no claims were made with respect to the effects of pose variations.

Dibeklioglu et al. [17, 18] introduced a nose tip localization and segmentation method using curvature-based heuristic analysis to enable pose correction in a face recognition system that allows identification under significant pose variations. However, their system cannot handle facial scans with yaw rotations greater than 45°. Additionally, even though the Bosphorus database that was used consisted of 3396 facial scans, the data were obtained from only 81 subjects.

Blanz et al. [5, 6] presented an approach in which a 3D Morphable Model was fitted on 3D facial scans, which is a well-established approach for producing 3D synthetic faces from scanned data. However, face recognition testing was validated on the FRGC database that consists of frontal facial scans, and on the FERET database that contains faces under pose variations which do not exceed 40°. Bronstein et al. [10] presented a face recognition method that is capable of handling missing data. This was an extension of their previous approach [9] where they deformed the face by embedding it into a multi-dimensional space. Such an approach preserves only the intrinsic geometries of face. Since facial expressions are mainly extrinsic geometries, the result is an expression invariant representation (canonical form) of the face. They reported high recognition rates, but on a limited database of 30 subjects. Also, the database did not contain side scans. Furthermore, the scans that contained missing data were derived synthetically by randomly removing certain areas from frontal scans. In Nair and Cavallaro’s [45] work on partial 3D face matching, the face was divided into areas and only certain areas were used for registration and matching. This approach was based on an assumption that the areas of missing data can be excluded. Using a database of 61 subjects, they showed that using parts of the face rather than the whole face, yields higher recognition rates. This approach, as well as their subsequent work on 3D landmark detection, cannot be applied to missing data resulting from pose self-occlusion, especially when holes exist around the nose region. Lin et al. [36] introduced a coupled 2D and 3D feature extraction method to determine the positions of eye sockets using curvature analysis. The nose tip was considered as the extreme vertex along the normal direction of eye sockets. The method was used in an automatic 3D face authentication system but was tested on only 27 datasets with various poses and expressions. Mian et al. [44] introduced a heuristic method for nose tip detection and used it in a face recognition system. The method is based on a geometric analysis of the nose ridge contour projected on the  $x$ - $y$  plane. It is used as a preprocessing step to crop and pose correct the facial data. Even though it allows up to 90° roll variation, this approach requires yaw and pitch variation less than 15°, thus limiting the applicability to near frontal scans. Perakis et al. [50] presented methods for detecting facial landmarks and used them to match partial facial data. Local shape and curvature analysis were used to locate

candidate landmark points (eye inner and outer corners, mouth corners, and nose and chin tips). The points were identified and labeled by matching them with a statistical facial landmark model. The method addresses the problem of extreme yaw rotations and missing facial areas, and its face recognition accuracy was validated against the FRGC v2 and UND Ear databases.

### 17.1.3 3D-aided 2D Face Recognition

The literature in 3D and 2D+3D Face Recognition has rapidly increased in recent years. An excellent survey was presented by Bowyer et al. [8]. The approach proposed by Riccio and Dugelay [55] uses geometric invariants on the face to establish a correspondence between the 3D gallery face and the 2D probe. Some of the invariants were manually selected. This algorithm does not use the texture information registered with the 3D data from the scanner, and hence, does not take full advantage of the input data. Blanz and Vetter [5] employed a morphable model technique to acquire the geometry and texture of faces from 2D images. Wang et al. [67] used a spherical harmonic representation [2] with the morphable model for 2D face recognition. Toderici et al. [61] proposed a method referred to as UR2D that uses 2D+3D data to build a 3D subject-specific model for the gallery. In contrast, Wang's method uses a 2D image to build a 3D model for the gallery based on a 3D statistical morphable model. Yin and Yourst [69] used frontal and profile 2D images to construct 3D shape models. In comparison to these methods, the UR2D method is able to more accurately model the subject identity as it uses both 2D and 3D information. Smith and Hancock [58] presented an approach for albedo estimation from 2D images also based on a 3D morphable model. The normals of the fitted model were then used for the computation of shading, assuming a Lambertian reflectance model. Biswas et al. [3] proposed a method for albedo estimation for face recognition using two-dimensional images. However, their approach was based on the assumption that the image does not contain shadows, and does not handle specular light. The relighting approach of Lee et al. [34], also suffers from the self-shadowing problem. Tsakalnidou [62] proposed a relighting method designed for face recognition but this approach produced images with poorer visual quality when compared to more generic methods, especially when specular highlights over-saturate the images.

### 17.1.4 3D-aided Profile Recognition

The use of face profile for identification had attracted research interest even before the arrival of the associated computer technologies [22]. The methods for recognition using the profile curve can be classified into one of two categories: *landmark-based* methods [27, 32, 37, 68] or *global* methods [23, 31, 47, 70]. Landmark-based methods rely on the attributes associated with a set of fiducial points, and recognition uses similarity metrics based on those attributes. Global methods consider each

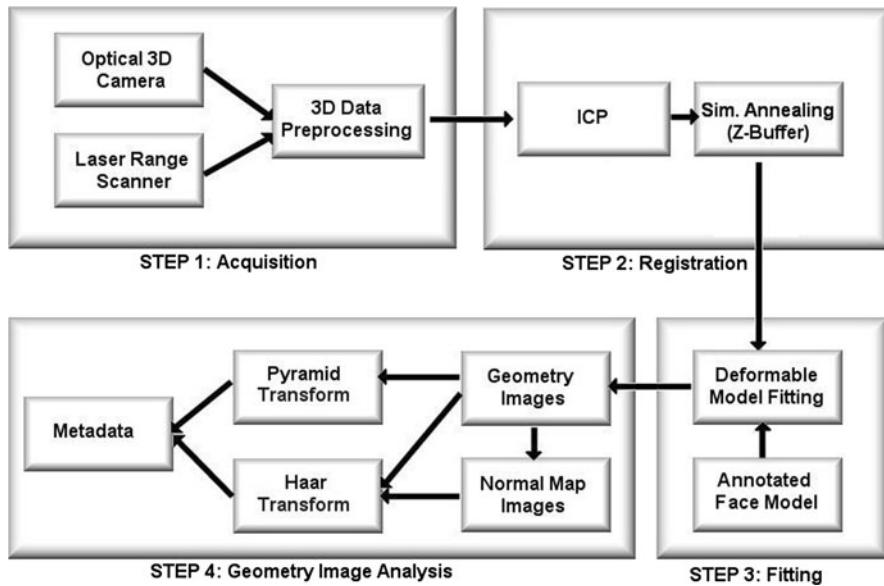
profile as a geometric object and introduce a similarity metric between homogeneous objects: all regions of a profile are treated equally.

Harmon et al. [27] defined 17 fiducial points; after aligning two profiles based on the selected landmarks, the matching was achieved by measuring the Euclidean distance of the feature vectors derived from the outlines. A 96% recognition rate was reported. Wu et al. [68] used a B-spline to locate six landmarks and extracted 24 features from the resulting segments. Liposca and Loncaric [37] used scale-space filtering to locate 12 landmarks and extracted 21 distances based on those landmarks. The Euclidean distance between the vectors of features was used for the identification.

Bhanu and Zhou [70] proposed curvature-based matching using a dynamic warping algorithm. They reported a recognition rate of almost 90% on the University of Bern Database that consisted of 30 subjects. Gao and Leung [24] introduced a method to encode profiles as attributed strings and developed an algorithm for attributed string matching. They reported nearly 100% recognition rate on the Bern database. Pan et al. [47] proposed a method that uses metrics for the comparison of probability density functions on properly rotated and normalized profile curves. Gao et al. [23, 25] proposed new formulations of the Hausdorff distance. Initially, their method was extended to match two sets of lines, while later, it was based on weighting points by their significance. In both cases, they applied their distance metric to measure the similarity of face profiles.

All these methods were designed for standard profiles only and use 2D images as gallery. Kakadiaris et al. [31] introduced the use of a 3D face model for the generation of profiles under different poses for the gallery. Modified directional Hausdorff distance of the probe profile to the gallery profile was used for identification. In addition, four different profiles under various rotation angles were used to introduce robustness to pose.

An important step in the implementation of a fully automatic system suitable for unconstrained scenarios is developing an accurate profile extractor. The majority of profile-based identification approaches do not sufficiently address this issue: instead they use manual extraction [31, 47] or very basic thresholding methods based on the assumption of indoor controlled illumination and a uniform background [7, 37, 70]. More efficient methods have been applied for near-frontal face extraction and feature localization. Among the most powerful are the methods based on the Active Shape Model (ASM), originally proposed by Cootes et al. [15]. These methods are based on recovering parameters of a statistical shape model, when a local minimum of the matching energy is found based on a search in local neighborhoods of the shape points. During the last decade, numerous modifications for the ASM have been proposed [26, 42]. The ultimate goal for most of these algorithms is alignment, therefore the shape is mostly defined by sparse set of common face landmarks visible on the frontal view, enforced by only a few additional points. For the contour extraction task, points should be densely sampled in order to approximate the curve accurately. Another known shortcoming of the ASM approach is the sensitivity to initialization, which is especially critical for ridge-like shapes.



**Fig. 17.1** Overview of the UR3D 3D face recognition method

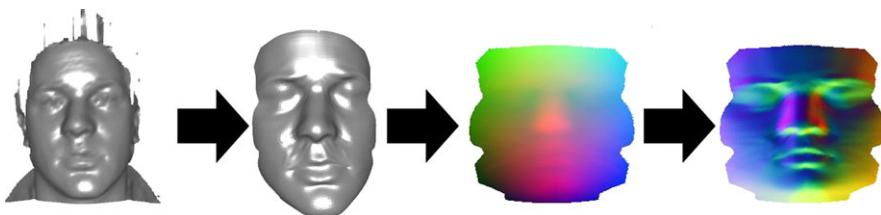
## 17.2 3D Face Recognition: UR3D

The UR3D 3D face recognition method is reviewed in this section [30]. It is a purely geometric approach as it does not require any statistical training. The AFM is deformed to capture the shape of the face of each subject. This approach represents the 3D information in an efficient 2D structure by utilizing the AFM's UV parameterization. This structure is subsequently analyzed in the wavelet domain and the spectral coefficients define the final metadata that are used for comparison among different subjects.

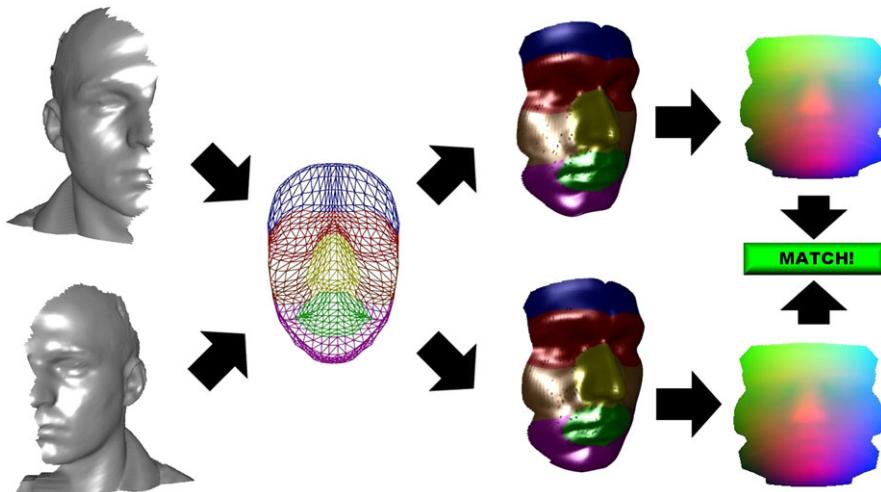
This method has the following steps (Fig. 17.1):

1. *Acquisition:* Raw 3D data are acquired from the sensor and converted to a polygonal representation using sensor-dependent preprocessing.
2. *Registration:* The data are registered to the AFM using a two-phase approach.
3. *Deformable Model Fitting:* The AFM is fitted to the data using a subdivision-based deformable model framework.
4. *Geometry Image Analysis:* Geometry and normal map images are derived from the fitted AFM and wavelet analysis is applied to extract a reduced coefficient set as metadata (Fig. 17.2).

A detailed explanation of each step can be found at [30].



**Fig. 17.2** From left to right: Facial scan → Fitted AFM → Extracted geometry image → Computed normal image



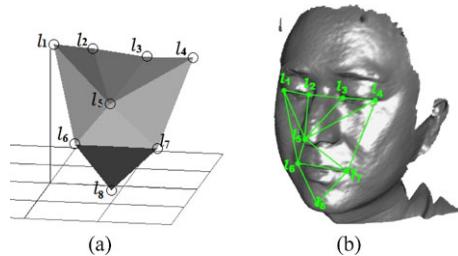
**Fig. 17.3** Interpose matching using the proposed method (left to right): Opposite side facial scans with extensive missing data, Annotated Face Model (AFM), resulting fitted AFM of each scan (facial symmetry used), extracted geometry images

### 17.3 3D Face Recognition for Partial Scans: UR3D-PS

UR3D is focused on 3D frontal facial scans and does not handle extensive missing data. In this section, the focus is shifted to 3D partial scans with missing data (such as side facial scans with large yaw rotations). The goal is to handle both frontal and side scans seamlessly thus producing a biometric signature that is pose invariant and hence, making the method more suitable for real-world applications.

The main idea of the proposed method is presented in Fig. 17.3. It allows matching among interpose facial scans and solves the missing data problem by using facial symmetry. To this end, a registration step is added that uses an automated 3D landmark detector to increase the resiliency of the registration process to large yaw rotations (common in side facial scans). Additionally, the subdivision-based deformable model framework is extended to allow symmetric fitting. Symmetric fitting alleviates the missing data problem as it derives geometry images from the AFM that

**Fig. 17.4** Depiction of:  
**a** landmark model as a 3D object; and **b** landmark model overlaid on a facial scan



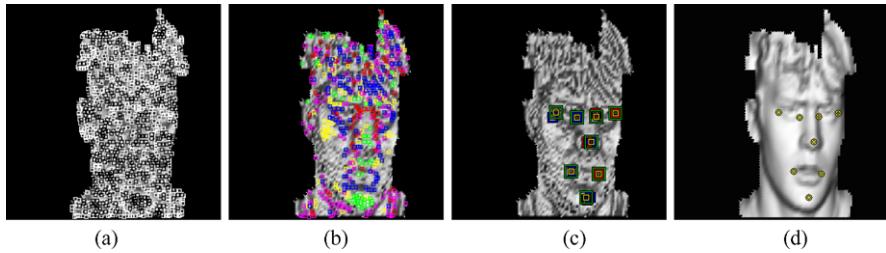
are pose invariant. Compared to the method presented in the previous section all other steps, except the registration and fitting step, remain unchanged. However, to make interpose matching more accurate, frontal facial scans are handled as a pair of independent side facial scans (left and right).

### 17.3.1 3D Landmark Detection

The proposed method (UR3D-PS) employs an improved version of the 3D landmark detection algorithm presented in [51]. Candidate interest points are extracted from the facial scans and are subsequently identified and labeled as landmarks by using a Facial Landmark Model (FLM). A set of 8 anatomical landmarks is used: right eye outer corner ( $l_1$ ), right eye inner corner ( $l_2$ ), left eye inner corner ( $l_3$ ), left eye outer corner ( $l_4$ ), nose tip ( $l_5$ ), mouth right corner ( $l_6$ ), mouth left corner ( $l_7$ ) and chin tip ( $l_8$ ) (Fig. 17.4). Note that at least five of these landmarks are always visible on side facial scans. The model with the entire set of eight landmarks will be referred to as FLM8 while the models with the reduced sets of five landmarks (left and right) will be referred to as FLM5L and FLM5R, respectively.

To create each FLM a mean shape is computed from a manually annotated training set. One hundred and fifty frontal facial scans with neutral expressions are randomly chosen from the FRGC v2 database as the training set. *Procrustes Analysis* [14, 19, 59] procedure is used to align the landmarks shape and calculate the mean shape. Subsequently, the variations of each FLM are analyzed by applying Principal Component Analysis (PCA) to the aligned landmark shapes. Aligned shape vectors form a distribution in the  $nd$  dimensional shape space, where  $n$  is the number of landmarks and  $d$  the dimension of each landmark. As described by Cootes et al. [16, 59], we can decompose this distribution and select the most significant eigenvectors of the eigenspace (*principal components*). We incorporated 15 eigenvalues (out of 24) in FLM8, which represent 99% of total shape variations of the frontal landmark shapes. Similarly, we incorporated 7 eigenvalues (out of 15) in FLM5L and FLM5R, which represent 99% of total shape variations of the left and right side landmark shapes.

The FLMs are used to detect landmarks in each facial scan as follows (depicted in Fig. 17.5):



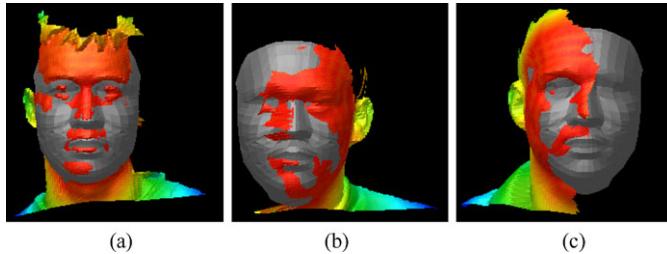
**Fig. 17.5** Results of landmark detection and selection process: **a** shape indexes maxima and minima; **b** spin image classification; **c** extracted best landmark sets; and **d** resulting landmarks

- *Extract candidate landmarks by using the Shape Index map.* After computing shape index values on a 3D facial scan, mapping to 2D space is performed to create the shape index map. Local maxima (Caps) are candidate landmarks for nose tips and chin tips and local minima (Cups) for eye corners and mouth corners. The most significant subset of points for each group (Caps and Cups) is retained.
- *Classify candidate landmarks by using Spin Image templates.* Candidate landmarks from the previous step are classified and filtered according to their relevance with five Spin Image templates. The similarity between two spin image grids  $P$  and  $Q$  is expressed by the normalized linear correlation coefficient:

$$S(P, Q) = \frac{N \sum p_i q_i - \sum p_i \sum q_i}{\sqrt{[N \sum p_i^2 - (\sum p_i)^2][N \sum q_i^2 - (\sum q_i)^2]}}$$

where  $p_i, q_i$  denotes each of the  $N$  elements of spin image grids  $P$  and  $Q$ , respectively.

- *Label Landmarks.* Using the classified candidate landmarks, feasible combinations of five landmarks are created. Subsequently, the rigid transformation that best aligns these combinations with the corresponding FLMs is computed. If the result is not consistent with FLM5L or FLM5R then the combination is filtered out. If it is consistent, the landmarks are labeled by the corresponding FLM and the combination is considered a possible solution. Possible solutions also include combinations of eight landmarks that are created from fusing two combinations of five landmarks (FLM5L and FLM5R) and are consistent with FLM8.
- *Select Final Solution.* The optimal solution (landmark combination) for each of the FLM5R, FLM5R and FLM8 is selected based on the distance from the mean shape of the corresponding FLM. To select the final solution the three optimal landmark combinations are compared using a normalized Procrustes Distance that takes into consideration the shape space dimensions.



**Fig. 17.6** AFM (gray) and facial scans (color coding: *red* means low registration error, *blue* means high registration error) superposed after registration (the scans): **a** frontal scan; **b** 45° left side scan; and **c** 60° right side scan

### 17.3.2 Partial Registration

Side facial scans with missing data cannot be registered robustly using the registration module of UR3D. To compute a rough but robust registration between the AFM and frontal or side facial scans (Fig. 17.6), the detected 3D landmarks are used. The Procrustes distance between a set of landmark points  $\mathbf{x}$  on the scan and the corresponding landmark points  $\mathbf{x}_0$  on the AFM is minimized in an iterative approach. If  $\mathbf{T}$  translates  $\mathbf{x}$  so that its centroid is at the origin (0,0,0),  $\mathbf{T}_0$  translates  $\mathbf{x}_0$  so that its centroid is at the origin (0,0,0), and  $\mathbf{R}$  is an optimal rotation that minimizes the Procrustes distance of  $\mathbf{x}$  to the reference shape  $\mathbf{x}_0$ , then, the final transformation to register a facial scan with vertices  $\mathbf{v}_i$  to the AFM is:

$$\mathbf{v}'_i = \mathbf{T}_0^{-1} \cdot \mathbf{R} \cdot \mathbf{T} \cdot \mathbf{v}_i$$

and pose is estimated from  $\mathbf{R}$ . The landmark set detected on a facial scan (frontal, right or left) determines which of the FLM8, FLM5R and FLM5L will be used. However, in practice when a frontal scan is detected, we do not use the FLM8, but we consider it as a pair of side scans (and compute two independent registrations using FLM5R and FLM5L).

To fine-tune the registration we use Simulated Annealing. Note that for side scans, only one half of the model's z-buffer is used in the objective function. The other half is excluded as it would have been registered with areas that may contain missing data. The landmark detection algorithm effectively substitutes the ICP in the registration process. Therefore, the Simulated Annealing algorithm is only allowed to produce limited translations and rotations and cannot alleviate registration errors caused by erroneous landmark detection.

### 17.3.3 Symmetric Deformable Model Fitting

We have modified the fitting module of UR3D to incorporate the notion of *symmetric fitting* in order to handle missing data. The framework can now handle the left

and right sides of the AFM independently. The idea is to use the facial symmetry to avoid the computation of the external forces on areas of possible missing data. The internal forces are not affected and remain unmodified to ensure the continuity of the fitted surface. As a result, when fitting the AFM to facial scans classified as left side (from the previous step), the external forces are computed on the left side of the AFM and mirrored to the right side (and vice versa for right side scans). Therefore, for each frontal scan, two fitted AFMs are computed: one that has the left side mirrored to the right and another that has the right side mirrored to the left. The method derives geometry and normal images from the deformed AFMs as described in the previous section.

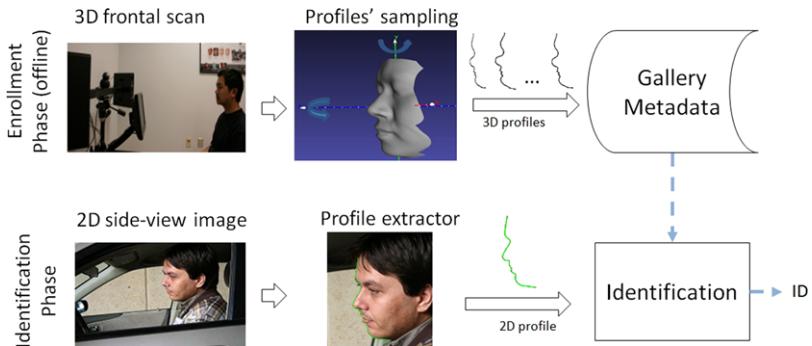
## 17.4 3D-aided Profile Recognition: URxD-PV

Until recently, research in profile-based recognition was based on comparison of *standard profiles*—the contours of side view images with yaw very close to  $-90^\circ$ . Research in 3D–3D face recognition has indicated that the profile information contains highly discriminative information [35, 48, 69], where the term “profile” is often associated with the facial area along the symmetry axis of the 3D face model. However, neither approach is capable of accurate modeling of a silhouetted face profile, as observed in a 2D image because (i) the face is not perfectly symmetric, (ii) the face is almost never at yaw equal to  $-90^\circ$  with respect to the sensor, and (iii) if the distance between camera and object is not sufficiently large, perspective projection needs to be considered (based on imaging sensor parameters). Note that, in this paper, the term “profile” always indicates the silhouette of nearly side view head images for clarity of presentation.

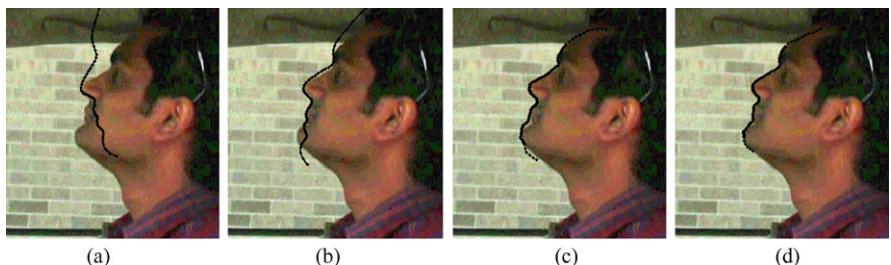
The central idea of our approach is the use 3D face models to explore the feature space of a profile under various rotations. An accurate 3D model embeds information about possible profile shapes in the probe 2D images, which allows flexibility and control over the training data. We suggest that sufficient sampling in the pose space, which corresponds to nearly side-view face images, provides robustness for a recognition task. Specifically, we propose to generate various profiles using rotations of a 3D face model. The profiles are used to train a classifier for profile-based identification. Two different types of profiles are employed in our system: (i) *3D profiles*—those generated synthetically through 3D face models to be used as training data, and (ii) *2D profiles*—those extracted from 2D images of side-view faces.

The schematic illustration of the profile-based face recognition system is depicted in Fig. 17.7 and includes *Enrollment* and *Identification* phases. The algorithmic solutions for the entire 3D-aided profile-based recognition framework including profile modeling, landmark detection, shape extraction, and classification are provided in [20].

In our approach, we treat the profile as an open curve  $\mathcal{C}$ , it may be described by a pair of arc-length parameterized 1D functions  $Y_{\mathcal{C}}(l)$  and  $X_{\mathcal{C}}(l)$ , where  $l \in [0, 1]$ . A set of  $k$  landmarks is defined by their coordinates on a parametric curve:  $\{0 = v^1 < \dots < v^k = 1\}$ . The set contains both anatomical landmarks (e.g., “chin”) and



**Fig. 17.7** Enrollment and identification phases of the proposed integrated profile-based face recognition system



**Fig. 17.8** Propagation of profile search. Depiction of **a** initial profile; **b** after two iterations; **c** after 5 iterations; and **d** final result

pseudo-landmarks (e.g., “middle of the nose”). We approximate functions  $Y_{\mathcal{C}}(l)$  and  $X_{\mathcal{C}}(l)$  by a finite set of points and obtain an equivalent *n-points shape model* as follows:

$$\mathbf{v} = [x_1, y_1, x_2, y_2, \dots, x_n, y_n]^T \in \mathbb{R}^{2n}. \quad (17.1)$$

The positions of the points are obtained through uniform arc-length sampling of the curve between a predefined subset of the landmarks. The sampling pattern is consistent for all profiles and, therefore, the coordinates of these landmarks always preserve their indices.

### 17.4.1 Profile Extraction from 2D Images

The profile extraction is based on the Active Shape Model paradigm developed by Cootes et al. [15]. It uses an iterative approach to gradually improve the fit of a given instance of the *n*-point shape to an image. Figure 17.8 depicts the shape propagation of the extractor. In the classical ASM framework, a manually labeled set of training images is needed. This dataset is used for the construction of the statistical

shape model, also known as Point Distribution Model (PDM). The same dataset is employed to design the features to guide the local search, typically by computing a pixel's likelihood belonging to a shape. In our case, no such labeled training set is available. Therefore, the main difference of our ASM framework from the classical approach is the fact that we only use available 3D shape information to guide the search of 2D profiles in the images and the local search is guided by features derived from color segmentation and edge detection operators.

### 17.4.2 Identification

During the identification phase, the matching scores between the probe profile and every profile in the gallery are computed. The decision is made according to the nearest neighbor rule. We propose to employ a modified Hausdorff distance as the matching score. For two finite point sets  $\mathcal{M} = \{m_1, \dots, m_n\}$  and  $\mathcal{T} = \{t_1, \dots, t_n\}$  with associated weights  $\{w_1^{\mathcal{M}}, \dots, w_n^{\mathcal{M}}\}$  and  $\{w_1^{\mathcal{T}}, \dots, w_n^{\mathcal{T}}\}$ , the distance is defined as:

$$\frac{1}{n} \max \left( h_{\mathcal{M}} \sum_{m_i \in \mathcal{M}} \min_{t_j \in \mathcal{T}} \|m_i - t_j\| \sqrt{w_i^{\mathcal{T}} w_j^{\mathcal{M}}}, h_{\mathcal{T}} \sum_{t_i \in \mathcal{T}} \min_{m_j \in \mathcal{M}} \|t_i - m_j\| \sqrt{w_j^{\mathcal{M}} w_i^{\mathcal{T}}} \right)$$

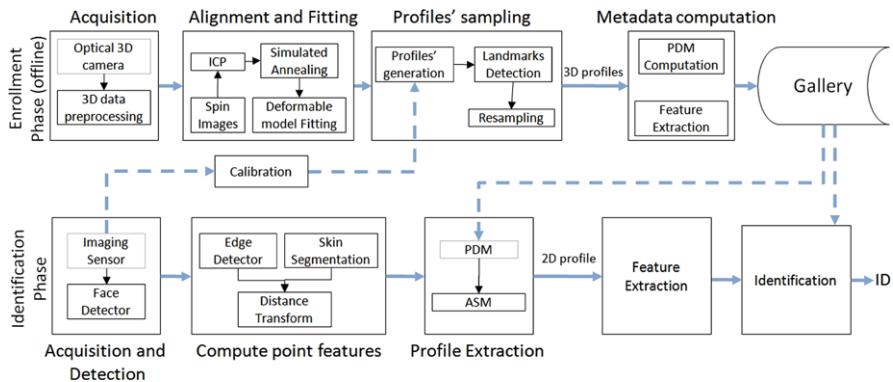
where  $\mathcal{M}$  and  $\mathcal{T}$  are probe and gallery  $n$ -point shapes and  $h_{\mathcal{M}}$  and  $h_{\mathcal{T}}$  are normalization factors of the distance between predefined landmarks to eliminate scale influence. The set of weights for a probe profile reflects the accuracy of a shape extractor (all equal 1 for manually extracted profiles). The set of weights for a gallery profile reflects prior knowledge about the discriminative properties of the various regions.

A single face profile is a weak biometric, primarily because of pose uncertainty and inaccuracies in the acquisition and extraction stages. If the sequence of frames is available, we can compensate for these uncertainties by fusing the results of recognition from multiple frames. Our assumption is that, by using video frames acquired at a low frame rate, we will be able to accumulate evidence from more poses.

### 17.4.3 Integration

The complete framework of the automatic profile-based profile recognition system is illustrated in Fig. 17.9. During the *Enrollment phase* (E) the raw data from each subject is converted to metadata and stored in the database as follows:

- E1. Acquire a facial shape with a 3D scanner and convert it to a polygonal mesh representation.
- E2. Align and fit the 3D data to a common reference model.



**Fig. 17.9** The components of an integrated 3D-aided profile recognition system (URxD-PV)

- E3. Generate multiple synthetic profiles by sampling a predefined range of rotation angles and locate a set of anatomical landmarks on them.
- E4. Derive a set of features based on the profile geometry and landmark locations from profiles and store them as metadata to be used in the identification phase.

During the *Identification phase* (I), the profile is extracted from a 2D image, and its metadata is matched with the gallery metadata as follows:

- I1. Acquire an image and compute a tight region of interest (*ROI*) that contains the face.
- I2. Compute a set of features for each pixel in the ROI, which will be used to guide the shape extraction procedure.
- I3. Extract the profile shape using the modified Active Shape Model.
- I4. Extract features (varies depending on classifier) from the profile shape.
- I5. Match/Classify the features.

## 17.5 3D-aided 2D Face Recognition: UR2D

We have developed a 3D-aided 2D face Recognition system (UR2D). Table 17.1 summarizes the different choices for the different modules in that UR2D system.

### 17.5.1 3D + 2D Enrollment

The UR2D enrollment method employs the Annotated Face Model (AFM) proposed by Kakadiaris et al. [30] to generate geometry images (regularly sampled 2D images with three channels) which encode geometric information ( $x$ ,  $y$  and  $z$  components of a vertex in  $R^3$ ). There are seven channels for the geometry images—three channels for representing the actual geometry of the face, three for representing the texture

**Table 17.1** Variations of the UR2D system

| Method name | Gallery Data | Probe Data | Geometry Image | Relighted | Distance metric | Score Normalization |
|-------------|--------------|------------|----------------|-----------|-----------------|---------------------|
| UR3D        | 3D           | 3D         | X              |           | CWSSIM          | MAD                 |
| UR2D-V-1    | 3D + 2D      | 2D         | X              | X         | GS              | E                   |
| UR2D-V-2    | 3D + 2D      | 2D         | X              | X         | CWSSIM          | E                   |
| UR2D-V-3    | 3D + 2D      | 2D         | X              |           | CWSSIM          | E                   |
| L1          | 2D           | 2D         |                |           |                 |                     |
| UR2D-V-4    | 2D           | 2D         |                |           | GS              | E                   |

---

**Algorithm 17.1:** Enrollment with 3D data

---

**Input:** 3D facial mesh, 2D facial image, subject ID.

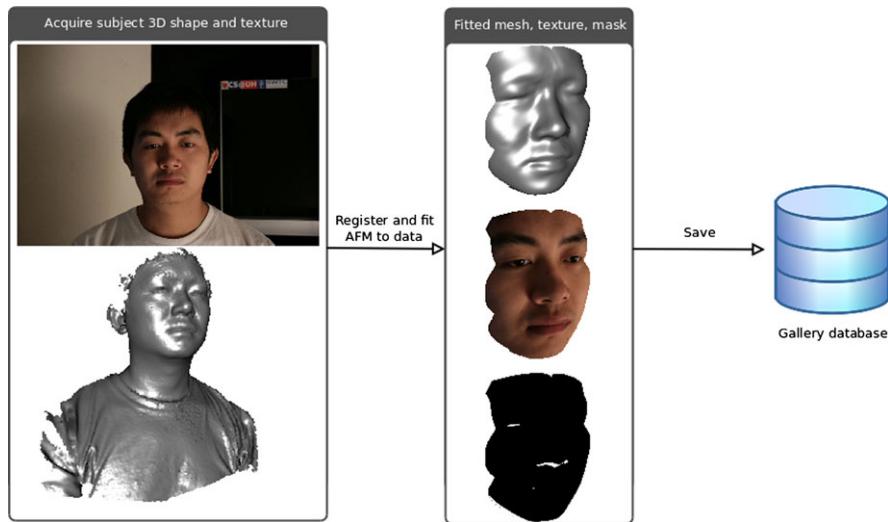
1. Preprocess the 3D facial mesh.
  2. Register AFM to the 3D facial mesh.
  3. Fit AFM to 3D facial mesh.
  4. Lift texture from the 2D facial image based on the fitted AFM.
  5. Compute visibility map.
  6. Store the fitted AFM, texture and visibility map in the gallery as metadata for subject ID.
- 

information, and one for the visibility map. For practical purposes, all experiments use a resolution of  $256 \times 256$ .

Specifically, the algorithm first fits the AFM to the input 3D data [30]. Once the fitting is complete, the AFM is represented as a geometry image. For each vertex in the geometry image, the algorithm computes the closest point on the data. The texel corresponding to this point in the data is used to create the corresponding texture image for the fitted AFM. Additionally, a visibility map is computed (Algorithm 17.1). If the closest point on the data does not have a valid texel assigned (i.e., if the 3D point was not visible to the 2D image sensor), the value one (1) is assigned to the corresponding location in the visibility map. Otherwise, it is assigned a value of zero. The enrollment pipeline is depicted in Fig. 17.10.

### 17.5.2 2D Authentication

In the authentication stage (Algorithm 17.2), the input is a 2D image. Seven fiducial landmarks (two eye inner corners, two eye outer corners, nose tip, and two nose corners) are detected using PittPatt [54]. Once the pose is estimated (using these landmarks and their corresponding locations on the AFM), the texture is mapped onto the AFM (Fig. 17.11). An analytical skin reflectance model (described in the



**Fig. 17.10** Depiction of the enrollment procedure for the UR2D algorithm. The *first column* lists the input data while the *second column* list the fitted AFM with texture on the top and without texture on the bottom

---

#### Algorithm 17.2: Authentication with 2D data

---

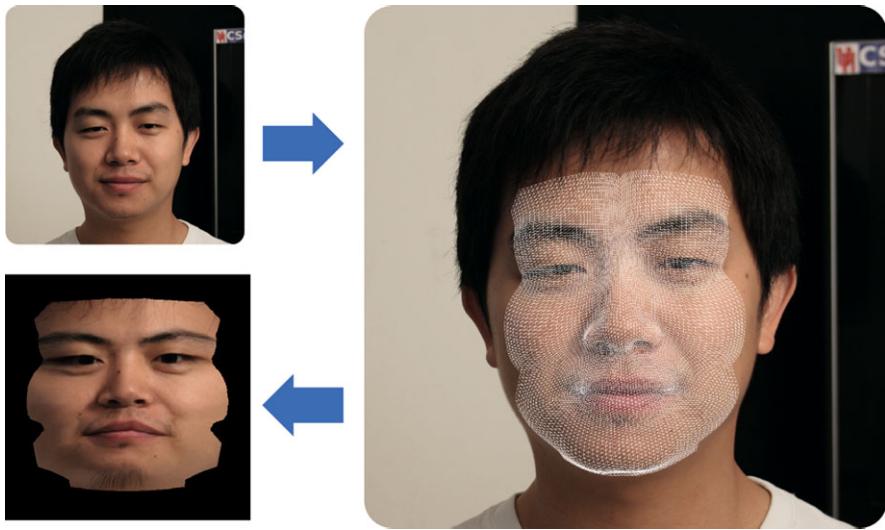
**Input:** 2D facial image and claimed subject ID.

1. Retrieve “claimed ID” AFM from the gallery.
  2. Locate the seven landmarks on the 2D facial image.
  3. Register the AFM to the 2D facial image using the corresponding landmarks (Fig. 17.11).
  4. Compute the visibility map.
  5. Bidirectionally relight the enrollment 2D facial texture to match the probe 2D facial texture.
  6. Compute the CWSSIM and GS scores between the relit texture and the probe texture.
  7. Threshold the score to make an ACCEPT/REJECT decision.
- 

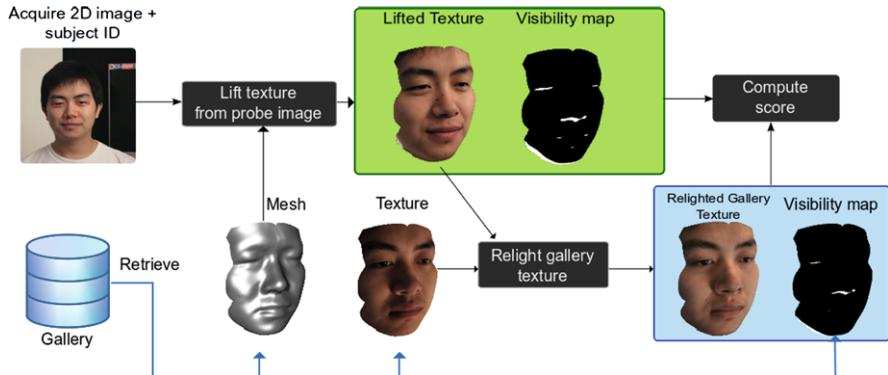
next section) is used to bidirectionally relight the *gallery* texture using the stored AFM mesh, in order to match the illumination of the probe texture (Fig. 17.12).

### 17.5.3 Skin Reflectance Model

In the case when test data is of sufficient resolution, a bidirectional surface scattering reflection distribution function (BSSRDF) should be used to model the skin reflectance. However, in most recognition systems, we deal with data of rather low

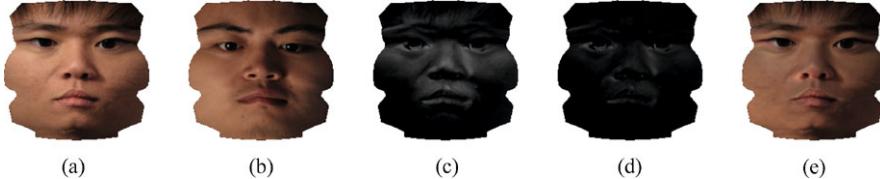


**Fig. 17.11** Converting raw 2D images to textures in the geometry image space: Raw 2D image → Fitted AFM of the same subject registered and superimposed over the image → Image converted to texture in geometry image space. The conversion is done by matching a set of landmarks on the AFM and on the 2D image



**Fig. 17.12** The authentication phase of the 3D-aided 2D face recognition system

resolutions, thus, it is safe to employ a hybrid bidirectional reflectance distribution function (BRDF) to model skin reflectance. The ASRM uses the Lambertian BRDF to model the diffuse component and the Phong BRDF to model the specular component. The Lambertian BRDF is the simplest, most widely used, physics-based model for diffuse reflectance. The model assumes that the surface is equally bright from all directions. The intensity of the light at a surface point is proportional to the angle between the surface normal and the incident light directions (denoted as  $\theta$ )  $I_d = E \cos \theta$ , where  $E$  is the intensity of the light source. The Lambertian BRDF



**Fig. 17.13** Optimization for relighting (textures are in geometry image space): **a**  $M'_T$ : texture of subject A; **b**  $M_T$ : texture of subject B; **c** texture difference between subjects (before optimization); **d** texture difference between subjects (after optimization); **e** subject A with illumination of subject B ( $I'_s + (I'_d + I'_a) \frac{M_T - I_s}{I_d + I_a}$ )

does not take into account the specular reflections caused by the oily layer of the skin. The BRDF proposed by Phong [53] can be used to accommodate for this. The intensity of the specular reflection at a surface point is  $I_s = E \cos^n \phi$ , where  $\phi$  is the angle between the view vector and the reflected light and  $n$  is a parameter that controls the size of the highlight. Note that each facial area has different specular properties, therefore the use of a specular map based on the annotation of the AFM is required [30].

#### 17.5.4 Bidirectional Relighting

The illumination parameters and the ASRM can be optimized in two different ways: estimating the albedo [4, 60] and transferring illumination (relighting). In both cases, the UR2D algorithm represents the texture in the AFM's UV space.

Generally, the texture  $M_T$  is the result of the lighting applied on the unknown albedo  $M_A$  and is given by:  $M_T = I_s + (I_d + I_a) \cdot M_A$ , where  $I_a$  is the ambient component,  $I_d$  the diffuse component and  $I_s$  the specular component (assuming white specular highlights). Solving this equation for the albedo yields:  $M_A = \frac{M_T - I_s}{I_d + I_a}$ . However, for many practical applications, the albedo itself is not required, and is used only as an intermediate step for relighting. Thus, when possible, the user should use bidirectional relighting without first estimating the albedo. This means that the optimization directly estimates the parameters for two lights (one that *removes* the illumination from the *gallery* image and one that *adds* the illumination from the *probe* image). The goal is to match the illumination conditions of a *gallery* texture to that of a *probe* texture. The following metric is minimized:

$$D = \left| M'_T - I'_s - (I'_d + I'_a) \frac{M_T - I_s}{I_d + I_a} \right|, \quad (17.2)$$

where  $I_a$ ,  $I_d$ , and  $I_s$  are the parameters of the light illuminating the *gallery*;  $I'_a$ ,  $I'_d$  and  $I'_s$  are the parameters of the second light illuminating the *probe*, while  $M'_T$  is the target texture. This process is depicted in Fig. 17.13. The relighting method is bidirectional, meaning that *probe* and *gallery* textures can be interchanged.



**Fig. 17.14** Facial scans with various expressions for a subject from the FRGC v2 database

**Table 17.2** Verification rates of our method at 0.001 FAR using different transforms on the FRGC v2 database

|         | ROC I | ROC II | ROC III |
|---------|-------|--------|---------|
| Fusion  | 97.3% | 97.2%  | 97.0%   |
| Haar    | 97.1% | 96.8%  | 96.7%   |
| Pyramid | 95.2% | 94.7%  | 94.1%   |

To improve the performance under low lighting conditions, instead of computing the difference in the RGB color space, a Hue-Saturation-Intensity (HSI) model can be used with the intensity weighed twice the amount of hue and saturation.

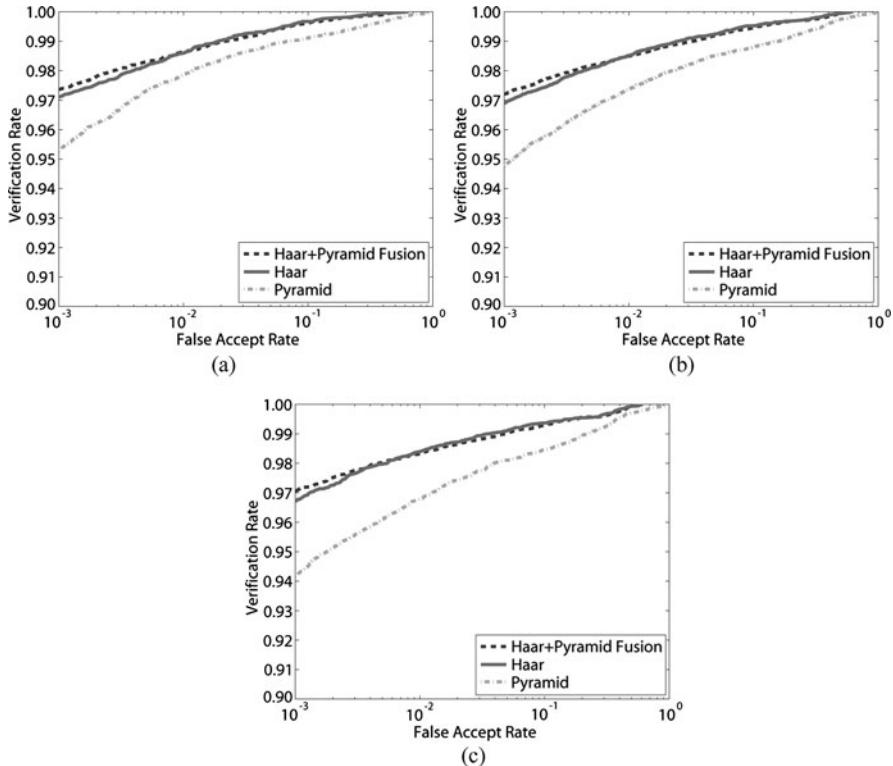
The equations above describe an ASRM for a single point light and the objective function to be minimized. The ASRM can be implemented as a Cg shader to greatly speed up the relighting process. For self-shadowing the shadow mapping technique can be used [60]. To model multiple point lights, the contribution of each light's ASRM must be summed. A full implementation of the ASRM on consumer level graphics hardware is able to bidirectionally relight a texture to a target within, on average, five seconds. Distance metrics and normalization methods are discussed in detail at [61].

## 17.6 Experimental Results

### 17.6.1 3D Face Recognition

For validation purposes, we have used the FRGC v2 [52] database, containing 4007 3D frontal facial scans of 466 persons. Figure 17.14 shows some examples of 3D facial scans from this database.

The performance is measured under a verification scenario. In order to produce comparable results, we use the three masks provided along with the FRGC v2 database. These masks, referred to as ROC I, ROC II and ROC III, are of increasing difficulty, respectively. The verification rates of our method at 0.001 False Acceptance Rate (FAR) are presented in the Table 17.2 [30]. The results are also presented using Receiver Operating Characteristic (ROC) curves (Fig. 17.15). The verification rate is measured for each wavelet transform separately, as well as for their weighted fusion. The average verification rate (over ROC I, II and III) was 97.16% for the fusion of the two transforms, 96.86% for the Haar transform and



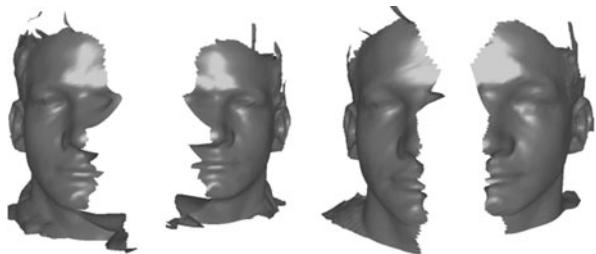
**Fig. 17.15** Performance of the proposed method using the Haar and Pyramid transforms as well as their fusion on the FRGC v2 database. Results reported using: **a** ROC I, **b** ROC II, and **c** ROC III

94.66% for the Pyramid transform. Even though the Pyramid transform is computationally more expensive it is outperformed by the simpler Haar wavelet transform. However, the fusion of the two transforms offers more descriptive power, yielding higher scores especially in the more difficult experiments (ROC II and ROC III).

### 17.6.2 3D Face Recognition for Partial Scans

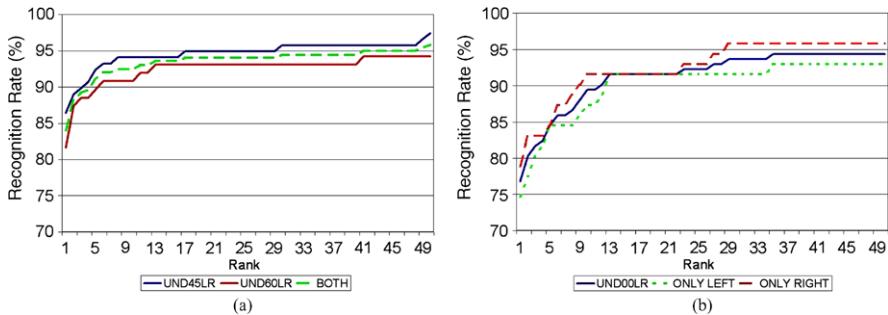
For interpose validation experiments, we combined the frontal facial scans of the FRGC v2 database with side facial scans of the UND Ear Database [63], collections F and G (Fig. 17.16). This database (which was created for ear recognition purposes) contains left and right side scans with yaw rotations of 45°, 60° and 90°. Note that for the purposes of our method, these side scans are considered partial frontal scans with extensive missing data. We use only the 45° side scans (118 subjects, 118 left and 118 right) and the 60° side scans (87 subjects, 87 left and 87 right). These data define two collections, referred to as UND45LR and UND60LR, respectively. For

**Fig. 17.16** Left and right side facial scans from the UND Ear Database



**Table 17.3** Rank-one recognition rate of our method for matching partial scans

|         | Rank-one Rate |
|---------|---------------|
| UND45LR | 86.4%         |
| UND60LR | 81.6%         |
| UND00LR | 76.8%         |



**Fig. 17.17** **a** CMC graphs for matching left (gallery) with right (probe) side scans using UND45LR, UND60LR and the combination of the two; **b** CMC graphs for matching frontal (gallery) with left, right and both (probe) side scans using UND00LR

each collection, the left side scan of a subject is considered gallery and the right is considered probe. A third collection, referred to as UND00LR, is defined as follows: the gallery set has one frontal scan for each of the 466 subjects of FRGC v2 while the probe set has a left and right  $45^\circ$  side scan from 39 subjects and a left and right  $60^\circ$  side scan from 32 subjects. Only subjects present in the gallery set were allowed in the probe set.

We evaluated the performance of our method under an identification scenario using partial scans of arbitrary sides for the gallery and probe sets. Our method can match any combination of left, right or frontal facial scans with the use of facial symmetry. For each of the three collections, the rank-one recognition rates are given in the Table 17.3 while the Cumulative Match Characteristic (CMC) graphs are depicted in Fig. 17.17.

In the cases of UND45LR and UND60LR, for each subject, the gallery set contains a single left side scan while the probe set contains a single right side scan.

Therefore facial symmetry is always used to perform identification. As expected, the  $60^\circ$  side scans yielded lower results as they are considered more challenging compared to the  $45^\circ$  side scans (see Fig. 17.17(a)). In the case of UND00LR, the gallery set contains a frontal scan for each subject, while the probe set contains left and right side scans. This scenario is very common when the enrollment of subjects is controlled but the identification is uncontrolled. In Fig. 17.17(b) the CMC graph is given (UND00LR's probe set is also split in left-only and right-only subsets). Compared to UND45LR and UND60LR, there is a decrease in the performance of our method in UND00LR. One could argue that since the gallery set consists of frontal scans (that do not suffer from missing data), the system should perform better. However, UND00LR has the largest gallery set (it includes all of the 466 subjects found in the FRGC v2 database) making it the most challenging database in our experiments with partial scans.

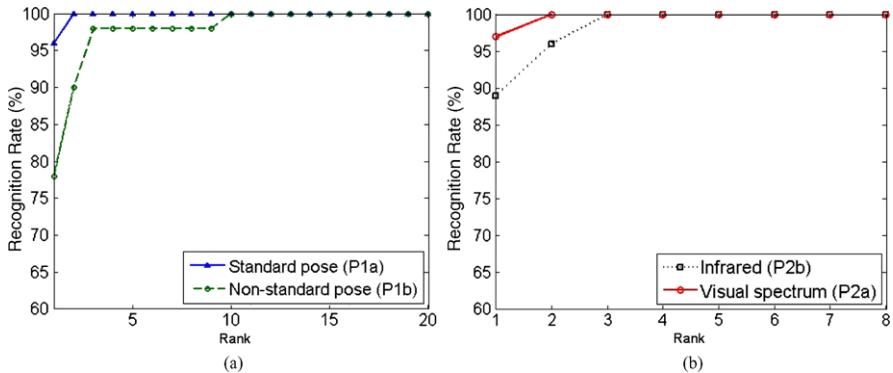
### 17.6.3 3D-aided Profile Recognition

In our experiments, we employ data from the face collection from the University of Houston [66] that contains 3D data that was acquired with a 2-pod 3dMD™ system and side-view 2D images. The acquisition environment includes both controlled (indoor, stable background) and uncontrolled (driver) scenarios. The contents of the probe cohorts P1a and P1b are single side-view images of the driver in standard and arbitrary non-standard poses, corresponding to the gallery of 50 subjects. The probe cohorts P2a and P2b are video sequences of 100 frames each, from the same scene in visual and infrared spectrum, respectively. Each sequence corresponds to one of 30 subjects in the gallery.

In the first experiment, we validate recognition performance of the system on the single-frame and the multi-frame cohorts. The CMC curves for each type of pose for the single-frame cohort are depicted in Fig. 17.18(a). The results depicted in Fig. 17.18(b) are assessing the performance of profile recognition on visual spectrum and infrared sequences.

We observe that recognition is higher for the nearly standard profiles (rank-1 recognition rate is 96%), than for nonstandard profiles (78%). This effect may be attributed to the fact that standard profiles contain more discriminative information. The drop in performance for the infrared sequence (89%) with comparison to visual spectrum sequence (97%) is attributed to the fact that it corresponds to smaller face size (about 500 pixels for P2a and only 140 pixels for P2b).

For the gallery profile sampling, we consider angles in the range  $[-110^\circ, -70^\circ]$  for yaw and  $[-25^\circ, 25^\circ]$  for roll. We do not create profiles for different pitch angles because they correspond to only the in-plane rotations and do not influence the geometry of the profile. The resolution of sampling is  $5^\circ$ . To demonstrate the sensitivity of the algorithm to the predefined range of gallery sampling angles, we compare recognition results based on the original gallery to the results based on wider or narrower ranges, where each range is reduced by  $5^\circ$  from each side. The



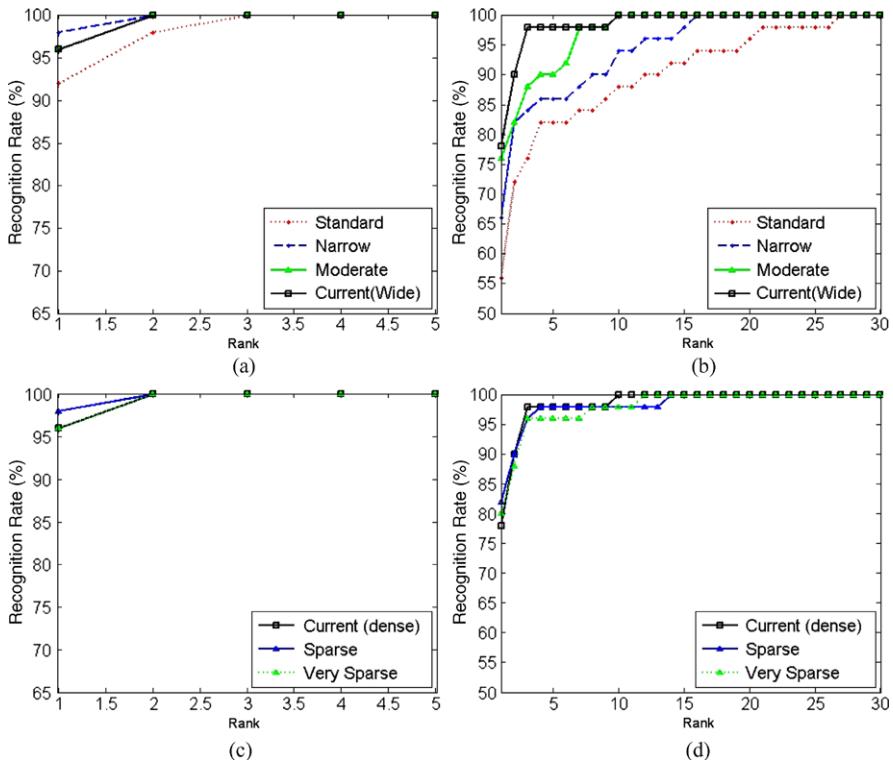
**Fig. 17.18** Recognition results on side-view single-frame and multi-frame images: **a** performance on single-frame cohorts, and **b** performance on multi-frame cohorts

outcome of this comparison is depicted in Figs. 17.19(a, b) separately for standard and nonstandard poses. In a similar manner, Figs. 17.19(c, d) depict the influences of angular sampling density on recognition by comparison of the current sampling density of  $5^\circ$  to the alternative sparser sampling densities of  $10^\circ$  and  $20^\circ$ . These experiments were applied on P1a and P1b cohorts to examine the influence on standard and nonstandard poses.

The results show a clear tendency for the widely sampled pose domain to be more robust on non-standard poses. For instance, rank-1 recognition is 78% for wide region (current settings), 76% for slightly narrower region and only 56% for the sampling region with  $10^\circ$  reduced from each side. On the other hand, narrow sampled pose domain regions will slightly outperform if we consider only nearly standard poses. For instance, sampling in the narrow region results in 98% rank-1 recognition as compared to 96% recognition for other settings (wide and moderate). However, even in this case, sampling only a single point corresponding to standard pose (ultra-narrow) is worse than other options and results in 92% rank-1 recognition for nearly standard poses and only 56% for nonstandard poses. Unlike the area of sampling region, the frequency of sampling has less influence on the performance.

#### 17.6.4 3D-aided 2D Face Recognition

**Database UHDB11 [64]** The UHDB11 database was created to analyze the impact of the variation in both pose and lighting. The database contains acquisitions from 23 subjects under six illumination conditions. For each illumination condition, the subject is asked to face four different points inside the room. This generated rotations on the Y axis. For each rotation on Y, three images are acquired with rotations on the Z axis (assuming that the Z axis goes from the back of the head to the nose, and that the Y axis is the vertical axis through the subject's head). Thus, each

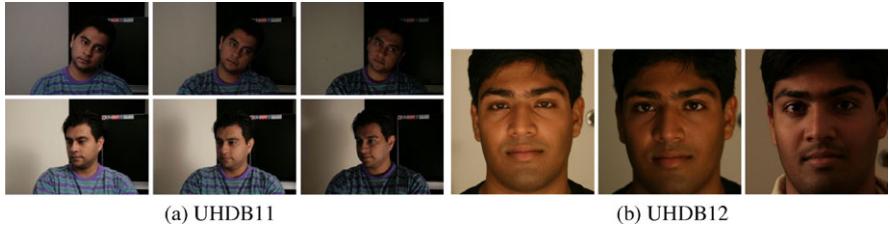


**Fig. 17.19** Recognition results using various sampling domains: **a, c** cohort P1a (nearly standard poses), and **b, d** cohort P1b (nonstandard poses)

subject is acquired under six illumination conditions, four Y rotations, and three Z rotations. For each acquisition, the subject 3D mesh is also acquired concurrently. Figure 17.20(a) depicts the variation in pose and illumination for one of the subjects from UHDB11. There are 23 subjects, resulting in 23 gallery datasets (3D plus 2D) and 1,602 probe datasets (2D only).

**Database UHDB12 [65]** The 3D data were captured using a 3dMD™ two-pod optical scanner, while the 2D data were captured using a commercial Canon™ DSLR camera. The system has six diffuse lights that allow the variation of the lighting conditions. For each subject, there is a single 3D scan (and the associated 2D texture) that is used as a gallery dataset and several 2D images that are used as probe datasets. Each 2D image is acquired under one of the six possible lighting conditions depicted in Fig. 17.20(b). There are 26 subjects, resulting in 26 gallery datasets (3D plus 2D) and 800 probe datasets (2D only).

**Authentication** We performed a variety of authentication experiments. We evaluated both relighting and unlighting. In case of unlighting, both gallery and probe images were unlit (thus, becoming albedos). In the case of relighting, the gallery



**Fig. 17.20** Examples images from database UHDB11 and database UHDB12 with variation of lighting and pose

**Fig. 17.21** ROC curve on authentication experiment on UHDB12 (varying illumination)

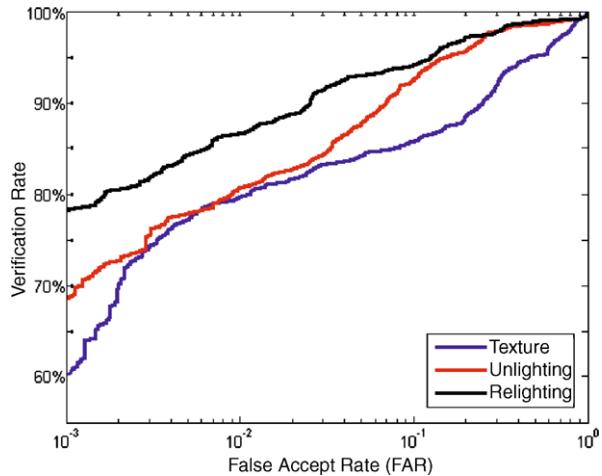
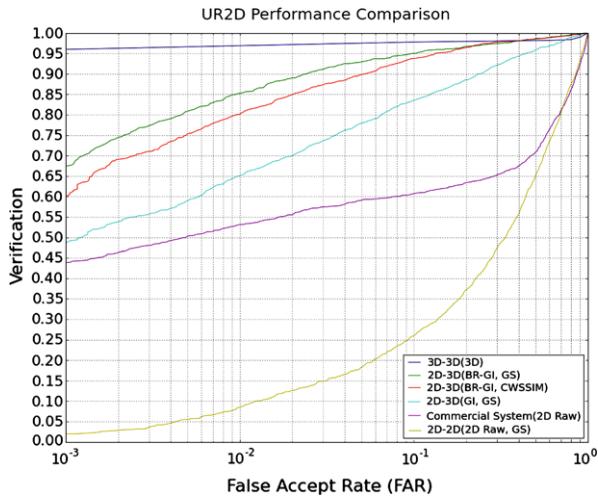


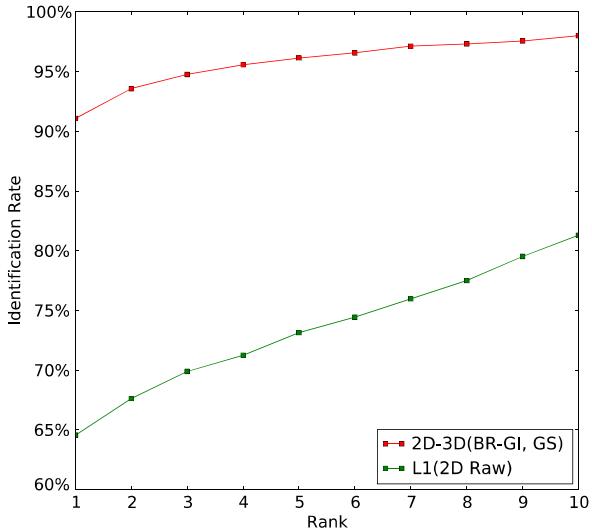
image was relit according to the probe image. The results for UHDB12 (using the UR2D algorithm, the CWSSIM metric and Z-normalization) are summarized using a Receiver Operating Characteristic (ROC) curve (Fig. 17.21). Note that face recognition benefits more from relit images than from unlit images. It achieves a 10% higher authentication rate at  $10^{-3}$  False Accept Rate (FAR) than unlighting. The performance using the raw texture is also included as a baseline. Even though these results depend on the UHDB12 and the distance metric that was used, they indicate clearly that *relighting is more suitable for face recognition than unlighting*. The reason behind this is that any unlighting method produces an albedo for which the ground truth is not known; Therefore, the optimization procedure is more prone to errors.

UHDB11 was employed to assess the robustness of the 3D-aided 2D face recognition approach with respect to both lighting and pose variation. Figure 17.22 depicts the ROC curve for UHDB11 for four different methods: (i) 3D-3D: Using the UR3D algorithm where both the gallery and probe are 3D datasets (shape only no texture) [30]; (ii) 2D-3D(BR\_GI, GS): The UR2D algorithm using bidirectionally relit images, GS distance metric, and E-normalization; (iii) 2D-3D(BR\_GI, CWSSIM): The UR2D algorithm using bidirectionally relit images, CWSSIM distance

**Fig. 17.22** ROC curve for an authentication experiment using data from UHDB11 (varying illumination and pose). Note that the Equal Error Rate which the 3D-aided 2D face recognition algorithm achieves is half that of the leading commercial product available at this time



**Fig. 17.23** Identification performance of the 3D-aided 2D face recognition approach versus the performance of a leading commercial 2D face recognition product



metric, and E-normalization; (iv) 2D-3D(GI, GS): The UR2D algorithm using raw texture from the geometry images, GS distance metric, and E-normalization; (v) 2D-2D(2D Raw, GS): Computing the GS distance metric for the raw 2D data, and E-normalization; (vi) L1(2D Raw, GS): Results from the L1 IdentityToolsSDK [33]. Note that the UR2D algorithm(BR\_GS, GS) outperforms one of the best commercial products.

**2D-3D Identification Experiment** The UHDB11 database is also used in an identification experiment. The results are provided in a Cumulative Matching Characteristic (CMC) curve on 23 subjects of UHDB11 (Fig. 17.23). From these results

it is evident that the UR2D algorithm outperforms the commercial 2D-only product throughout the entire CMC curve.

## 17.7 Conclusions

While the price of commercial 3D systems is dropping, to tap into the wealth of 2D sensors that are already economically available, we would need to employ a 3D-aided 2D recognition technique. These 3D-aided 2D recognition methods can provide promising results without the need for an expensive 3D sensor at the authentication site. The effectiveness of these methods with relighting process have been demonstrated and it has been proven to provide robust face recognition under varying pose and lighting condition.

**Acknowledgements** This work was funded in part by the US Army Research Laboratory award DWAM80750, the UH Eckhard Pfeiffer Endowment, and the Central Intelligence Agency under the DISA ENCORE contract DCA200-02-D-5014 with Unisys Corporation serving as the primary on behalf of the Government. Any opinions, findings, conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of our sponsors.

## References

1. Al-Osaimi, F., Bennamoun, M., Mian, A.: An expression deformation approach to non-rigid 3D face recognition. *Int. J. Comput. Vis.* **81**(3), 302–316 (2009)
2. Basri, R., Jacobs, D.: Lambertian reflectance and linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(2), 218–233 (2003)
3. Biswas, S., Aggarwal, G., Chellappa, R.: Robust estimation of albedo for illumination-invariant matching and shape recovery. *IEEE Trans. Pattern Anal. Mach. Intell.* **1**(8), 884–899 (2008)
4. Biswas, S., Aggarwal, G., Chellappa, R.: Robust estimation of albedo for illumination-invariant matching and shape recovery. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**, 884–899 (2009)
5. Blanz, V., Vetter, T.: Face recognition based on fitting a 3D morphable model. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(9), 1063–1074 (2003)
6. Blanz, V., Scherbaum, K., Seidel, H.-P.: Fitting a morphable model to 3D scans of faces. In: Proc. 11th IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, 14–20 October 2007
7. Bottino, A., Cumani, S.: A fast and robust method for the identification of face landmarks in profile images. *WSEAS Trans. Comput.* **7**, 1250–1259 (2008)
8. Bowyer, K., Chang, K., Flynn, P.: A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition. *Comput. Vis. Image Underst.* **101**(1), 1–15 (2006)
9. Bronstein, A., Bronstein, M., Kimmel, R.: Three-dimensional face recognition. *Int. J. Comput. Vis.* **64**(1), 5–30 (2005)
10. Bronstein, A., Bronstein, M., Kimmel, R.: Robust expression-invariant face recognition from partially missing data. In: Proc. European Conference on Computer Vision, pp. 396–408, Graz, Austria (2006)
11. Chang, K., Bowyer, K., Flynn, P.: Adaptive rigid multi-region selection for handling expression variation in 3D face recognition. In: Proc. IEEE Workshop on Face Recognition Grand Challenge Experiments, pp. 157–164, San Diego, CA, 20–25 June 2005

12. Chang, K., Bowyer, K., Flynn, P.: Effects on facial expression in 3D face recognition. In: Proc. SPIE Biometric Technology for Human Identification II, vol. 5779, pp. 132–143, Orlando, FL (2005)
13. Chang, K., Bowyer, K., Flynn, P.J.: An evaluation of multi-modal 2D+3D face biometrics. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(4), 619–624 (2005)
14. Cootes, T., Taylor, C.: Statistical models of appearance for computer vision. Technical report, University of Manchester, October 2001
15. Cootes, T., Taylor, C., Cooper, D., Graham, J.: Active shape models—their training and application. *Comput. Vis. Image Underst.* **61**(1), 38–59 (1995)
16. Cootes, T., Taylor, C., Kang, H., Petrovic, V.: Modeling facial shape and appearance. In: *Handbook of Face Recognition*, pp. 39–63. Springer, Berlin (2005)
17. Dibeklioglu, H.: Part-based 3D face recognition under pose and expression variations. Master's thesis, Bogazici University (2008)
18. Dibeklioglu, H., Salah, A., Akarun, L.: 3D facial landmarking under expression, pose and occlusion variations. In: Proc. 2nd International Conference on Biometrics Theory, Applications and Systems, Arlington, VA, 29 September–1 October 2008
19. Dryden, I., Mardia, K.: *Statistical Shape Analysis*. Wiley, New York (1998)
20. Efraty, B., Ismailov, E., Shah, S., Kakadiaris, I.: Profile-based 3d-aided face recognition. *Pattern Recognit.* (2011, in press). Corrected Proof. Available online 19 July 2011
21. Face recognition vendor test 2006 (2006). <http://www.frvt.org/FRVT2006/>
22. Galton, F.: Numerical profiles for classification and recognition. *Nature* **83**, 127–130 (1910)
23. Gao, Y.: Efficiently comparing face images using a modified Hausdorff distance. In: Proc. IEEE Conference on Vision, Image and Signal Processing, pp. 346–350, December 2003
24. Gao, Y., Leung, M.: Human face profile recognition using Attributed String. *Pattern Recognit.* **35**(2), 353–360 (2002)
25. Gao, Y., Leung, M.: Line segment Hausdorff distance on face matching. *Pattern Recognit.* **35**(2), 361–371 (2002)
26. Gu, L., Kanade, T.: 3D alignment of face in a single image. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1305–1312, New York, NY, 17–22 June 2006
27. Harmon, L.D., Khan, M.K., Lasch, R., Ramig, P.: Machine identification of human faces. *Pattern Recognit.* **2**(13), 97–110 (1981)
28. Husken, M., Brauckmann, M., Gehlen, S., von der Malsburg, C.: Strategies and benefits of fusion of 2D and 3D face recognition. In: Proc. IEEE Workshop on Face Recognition Grand Challenge Experiments, pp. 174–181, San Diego, CA, 20–25 June 2005
29. Kakadiaris, I., Passalis, G., Theoharis, T., Toderici, G., Konstantinidis, I., Murtuza, N.: Multimodal face recognition: Combination of geometry with physiological information. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1022–1029, San Diego, CA, 20–25 June 2005
30. Kakadiaris, I., Passalis, G., Toderici, G., Murtuza, M., Lu, Y., Karampatziakis, N., Theoharis, T.: Three-dimensional face recognition in the presence of facial expressions: An annotated deformable model approach. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(4), 640–649 (2007)
31. Kakadiaris, I., Abdelmunim, H., Yang, W., Theoharis, T.: Profile-based face recognition. In: Proc. 8th IEEE International Conference on Automatic Face and Gesture Recognition, pp. 1–8, Amsterdam, The Netherlands, 17–19 September 2008
32. Kaufman, G., Breeding, K.: The automatic recognition of human faces from profile silhouettes. *IEEE Trans. Syst. Man Cybern.* **6**, 113–121 (1976)
33. L1 Identity Solutions. L1 faceit SDK
34. Lee, J., Machiraju, R., Pfister, H., Moghaddam, B.: Estimation of 3D faces and illumination from single photographs using a bilinear illumination model. In: Proc. Eurographics Symposium on Rendering, pp. 73–82, Konstanz, Germany, 29 June–1 July 2005
35. Li, C., Barreto, O.: Profile-based 3D Face Registration and Recognition, vol. 3506, pp. 478–488. Springer, Berlin (2005). Chap. 10

36. Lin, T., Shih, W., Chen, W., Ho, W.: 3D face authentication by mutual coupled 3D and 2D feature extraction. In: Proc. 44th ACM Southeast Regional Conference, Melbourne, FL, 10–12 March 2006
37. Liposca, Z., Loncaric, S.: A scale-space approach to face recognition from profiles. In: Proc. 8th International Conference on Computer Analysis of Images and Patterns, pp. 243–250, London, UK, September 1999
38. Lu, X., Jain, A.: Multimodal facial feature extraction for automatic 3D face recognition. Technical Report MSU-CSE-05-22, Michigan State University, October 2005
39. Lu, X., Jain, A.: Automatic feature extraction for multiview 3D face recognition. In: Proc. 7th International Conference on Automatic Face and Gesture Recognition, Southampton, UK, 10–12 April 2006
40. Lu, X., Jain, A.: Deformation modeling for robust 3D face matching. In: Proc. IEEE Computer Vision and Pattern Recognition, pp. 1377–1383, New York, NY, 17–22 June 2006
41. Lu, X., Jain, A., Colbry, D.: Matching 2.5D face scans to 3D models. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(1), 31–43 (2006)
42. Mahoor, M., Abdel-Mottaleb, M.: Facial features extraction in color images using enhanced active shape model. In: Proc. 7th International Conference on Automatic Face and Gesture Recognition, pp. 144–148, Washington, DC, USA, 2–6 April 2006
43. Maurer, T., Guigonis, D., Maslov, I., Pesenti, B., Tsaregorodtsev, A., West, D., Medioni, G.: Performance of Geometrix ActiveIDTM 3D face recognition engine on the FRGC data. In: Proc. IEEE Workshop on Face Recognition Grand Challenge Experiments, San Diego, CA, 20–25 June 2005
44. Mian, A., Bennamoun, M., Owen, R.: An efficient multimodal 2D-3D hybrid approach to automatic face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(11), 1927–1943 (2007)
45. Nair, P., Cavallaro, A.: Matching 3D faces with partial data. In: Proc. British Machine Vision Conference, Leeds, UK, 1–4 September 2008
46. Pan, G., Han, S., Wu, Z., Wang, Y.: 3D face recognition using mapped depth images. In: Proc. IEEE Workshop on Face Recognition Grand Challenge Experiments, pp. 175–181, San Diego, CA, 20–25 June 2005
47. Pan, G., Zheng, L., Wu, Z.: Robust metric and alignment for profile-based face recognition: An experimental comparison. In: Proc. 7th IEEE Workshop on Applications of Computer Vision, vol. 1, pp. 117–122, January 2005
48. Papathodorou, T., Rueckert, D.: 3D face recognition. In: Face Recognition, pp. 417–446. I-Tech Education and Publishing, July 2007
49. Passalis, G., Kakadiaris, I., Theoharis, T., Toderici, G., Murtuza, N.: Evaluation of 3D face recognition in the presence of facial expressions: An annotated deformable model approach. In: Proc. IEEE Workshop on Face Recognition Grand Challenge Experiments, vol. 3, p. 171, San Diego, CA, 20–25 June 2005
50. Perakis, P., Passalis, G., Theoharis, T., Toderici, G., Kakadiaris, I.: Partial matching of interpose 3D facial data for face recognition. In: Proc. 3rd IEEE International Conference on Biometrics: Theory, Applications and Systems, Arlington, VA, 28–30 September 2009
51. Perakis, P., Theoharis, T., Passalis, G., Kakadiaris, I.: Automatic 3D facial region retrieval from multi-pose facial datasets. In: Proc. Eurographics Workshop on 3D Object Retrieval, pp. 37–44, Munich, Germany, 30 March–3 April 2009
52. Phillips, P., Flynn, P., Scruggs, T., Bowyer, K., Chang, J., Hoffman, K., Marques, J., Min, J., Worek, W.: Overview of the face recognition grand challenge. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 947–954, San Diego, CA (2005)
53. Phong, B.: Illumination for computer generated pictures. *Commun. ACM* **18**(6), 311–317 (1975)
54. Pittsburgh Pattern Recognition. PittPatt face tracking & recognition software development kit (2009)
55. Riccio, D., Dugelay, J.-L.: Geometric invariants for 2D/3D face recognition. *Pattern Recognit. Lett.* **28**(14), 1907–1914 (2007)

56. Russ, T., Koch, K., Little, C.: 3D facial recognition: A quantitative analysis. In: Proc. 45th Annual Meeting of the Institute of Nuclear Materials Management, pp. 338–344, July 2004
57. Scheenstra, A., Ruifrok, A., Veltkamp, R.C.: A survey of 3d face recognition methods. In: Proc. in Lecture Notes in Computer Science, pp. 891–899 (2005)
58. Smith, W., Hancock, E.: Estimating the albedo map of the face from a single image. In: Proc. IEEE International Conference on Image Processing, vol. 3, pp. 780–783, Genoa, Italy, 11–14 September 2005
59. Stegmann, M.B., Gomez, D.D.: A brief introduction to statistical shape analysis. Technical report, Technical University of Denmark, March 2002
60. Toderici, G., Passalis, G., Theoharis, T., Kakadiaris, I.: An automated method for human face modeling and relighting with application to face recognition. In: Proc. Workshop on Photometric Analysis For Computer Vision, Rio de Janeiro, Brazil, 14–21 October 2007
61. Toderici, G., Passalis, G., Zafeiriou, S., Tzimiropoulos, G., Petrou, M., Theoharis, T., Kakadiaris, I.: Bidirectional relighting for 3D-aided 2D face recognition. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2010)
62. Tsalakanidou, F., Malassiotis, S., Strintzis, M.: A 2D + 3D face identification system for surveillance applications. In: Proc. IEEE International Conference on Advanced Video and Signal based Surveillance, pp. 194–199, London, UK, 5–7 September 2007
63. U. of Notre Dame. University of Notre Dame Biometrics Database (2008). <http://www.nd.edu/@cvrl/UNDBiometricsDatabase.html>
64. UH Computational Biomedicine Lab. UHDB11 face database (2009). <http://cbl.uh.edu/URxD/datasets/>
65. UH Computational Biomedicine Lab. UHDB12 face database (2009). <http://cbl.uh.edu/URxD/datasets/>
66. URxD-PV. UHDB22: CBL database for biometrics research. Available at <http://cbl.uh.edu/URxD/datasets/>
67. Wang, Y., Zhang, L., Liu, Z., Hua, G., Wen, Z., Zhang, Z., Samaras, D.: Face relighting from a single image under arbitrary unknown lighting conditions. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(11), 1968–1984 (2009)
68. Wu, C., Huang, J.: Human face profile recognition by computer. *Pattern Recognit.* **23**, 255–259 (1990)
69. Yin, L., Yourst, M.: 3D face recognition based on high-resolution 3D face modeling from frontal and profile views. In: Proc. ACM SIGMM Workshop on Biometrics Methods and Applications, pp. 1–8, New York, NY, 8 November 2003
70. Zhou, X., Bhanu, B.: Human recognition based on face profiles in video. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, June 2005



# Chapter 18

## Facial Action Tracking

Jörgen Ahlberg and Igor S. Pandzic

### 18.1 Introduction

The problem of facial action tracking has been a subject of intensive research in the last decade. Mostly, this has been with such applications in mind as face animation, facial expression analysis, and human-computer interfaces. In order to create a face animation from video, that is, to capture the facial action (facial motion, facial expression) in a video stream, and then animate a face model (depicting the same or another face), a number of steps have to be performed. Naturally, the face has to be detected, and then some kind of model has to be fitted to the face. This can be done by aligning and deforming a 2D or 3D model to the image, or by localizing a number of facial landmarks. Commonly, these two are combined. The result must in either case be expressed as a set of parameters that the face model in the receiving end (the face model to be animated) can interpret.

Depending on the application, the model and its parameterization can be simple (e.g., just an oval shape) or complex (e.g., thousands of polygons in layers simulating bone and layers of skin and muscles). We usually wish to control appearance, structure, and motion of the model with a small number of parameters, chosen so as to best represent the variability likely to occur in the application. We discriminate here between rigid face/head tracking and tracking of facial action. The former is typically employed to robustly track the faces under large pose variations, using a rigid face/head model (that can be quite non-face specific, e.g., a cylinder). The latter here refers to tracking of facial action and facial features, such as lip and eyebrow

---

J. Ahlberg (✉)

Division of Information Systems, Swedish Defence Research Agency (FOI), P.O. Box 1165,  
583 34 Linköping, Sweden  
e-mail: [jorahl@foi.se](mailto:jorahl@foi.se)

I.S. Pandzic

Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, 10000 Zagreb,  
Croatia  
e-mail: [igor.pandzic@fer.hr](mailto:igor.pandzic@fer.hr)

motion. The treatment in this chapter is limited to tracking of facial action (which, by necessity, includes the head tracking).

The parameterization can be dependent or independent on the model. Examples of model independent parameterizations are MPEG-4 Face Animation Parameters (see Sect. 18.2.3) and FACS Action Units (see Sect. 18.2.2). These parameterizations can be implemented on virtually any face model, but leaves freedom to the model and its implementation regarding the final result of the animation. If the transmitting side wants to have full control of the result, more dense parameterizations (controlling the motion of every vertex in the receiving face model) are used. Such parameterizations can be implemented by MPEG-4 Facial Animation Tables (FAT), morph target coefficients, or simply by transmitting all vertex coordinates at each time step.

Then, the face and its landmark/features/deformations must be tracked through an image (video) sequence. Tracking of faces has received significant attention for quite some time but is still not a completely solved problem.

### **18.1.1 Previous Work**

A plethora of face trackers are available in the literatures, and only a few of them can be mentioned here. They differ in how they model the face, how they track changes from one frame to the next, if and how changes in illumination and structure are handled, if they are susceptible to drift, and if real-time performance is possible. The presentation here is limited to monocular systems (in contrast to stereo-vision) and 3D tracking.

#### **18.1.1.1 Rigid Face/Head Tracking**

Malciu and Prêteux [36] used an ellipsoidal (alternatively an ad hoc Fourier synthesized) textured wireframe model and minimized the registration error and/or used the optical flow to estimate the 3D pose. LaCascia et al. [31] used a cylindrical face model with a parameterized texture being a linear combination of texture warping templates and orthogonal illumination templates. The 3D head pose was derived by registering the texture map captured from the new frame with the model texture. Stable tracking was achieved via regularized, weighted least-squares minimization of the registration error.

Basu et al. [7] used the Structure from Motion algorithm by Azerbayejani and Pentland [6] for 3D head tracking, refined and extended by Jebara and Pentland [26] and Ström [56] (see below). Later rigid head trackers include notably the works by Xiao et al. [64] and Morency et al. [40].

### 18.1.1.2 Facial Action Tracking

In the 1990s, there were many approaches to non-rigid face tracking. Li et al. [33] estimated face motion in a simple 3D model by a combination of prediction and a model-based least-squares solution to the optical flow constraint equation. A render-feedback loop was used to combat drift. Eisert and Girod [16] determined a set of animation parameters based on the MPEG-4 Facial Animation standard (see Sect. 18.2.3) from two successive video frames using a hierarchical optical flow based method. Tao et al. [58] derived the 3D head pose from 2D-to-3D feature correspondences. The tracking of nonrigid facial features such as the eyebrows and the mouth was achieved via a probabilistic network approach. Pighin et al. [50] derived the face position (rigid motion) and facial expression (nonrigid motion) using a continuous optimization technique. The face model was based on a set of 3D face models.

In this century, DeCarlo and Metaxas [11] used a sophisticated face model parameterized in a set of deformations. Rigid and nonrigid motion was tracked by integrating optical flow constraints and edge-based forces, thereby preventing drift. Wiles et al. [63] tracked a set of hyperpatches (i.e., representations of surface patches invariant to motion and changing lighting).

Gokturk et al. [22] developed a two-stage approach for 3D tracking of pose and deformations. The first stage learns the possible deformations of 3D faces by tracking stereo data. The second stage simultaneously tracks the pose and deformation of the face in the monocular image sequence using an optical flow formulation associated with the tracked features. A simple face model using 19 feature points was utilized.

As mentioned, Ström [56] used an Extended Kalman Filter (EKF) and Structure from Motion to follow the 3D rigid motion of the head. Ingemars and Ahlberg extended the tracker to include facial action [24]. Ingemars and Ahlberg combined two sparse texture models, based on the first frame and (dynamically) on the previous frame respectively, in order to get accurate tracking and no drift. Lefèvre and Odobez used a similar idea, but separated the texture models more, and used Nelder–Mead optimization [42] instead of an EKF (see Sect. 18.4).

As follow-ups to the introduction of Active Appearance Models, there were several appearance-based tracking approaches. Ahlberg and Forchheimer [4, 5] represented the face using a deformable wireframe model with a statistical texture. A simplified Active Appearance Model was used to minimize the registration error. Because the model allows deformation, rigid and non-rigid motions are tracked. Dornaika and Ahlberg [12, 14] extended the tracker with a step based on random sampling and consensus to improve the rigid 3D pose estimation. Fanelli and Fratarcangeli [18] followed the same basic strategy, but exploited the Inverse Compositional Algorithm by Matthews and Baker [37]. Zhou et al. [65] and Dornaika and Davoine [15] combined appearance models with a particle filter for improved 3D pose estimation.

### 18.1.2 Outline

This chapter explains the basics of parametric face models used for face and facial action tracking as well as fundamental strategies and methodologies for tracking. A few tracking algorithms serving as pedagogical examples are described in more detail. The chapter is organized as follows: In Sect. 18.2 parametric face modeling is described. Various strategies for tracking are discussed in Sect. 18.3, and a few tracker examples are described in Sects. 18.4–18.6. In Sect. 18.6.3 some examples of commercially available tracking systems are give.

## 18.2 Parametric Face Modeling

There are many ways to parameterize and model the appearance and behavior of the human face. The choice depends on, among other things, the application, the available resources, and the display device. Statistical models for analyzing and synthesizing facial images provide a way to model the 2D appearance of a face. Here, other modeling techniques for different purposes are mentioned as well.

What all models have in common is that a compact representation (few parameters) describing a wide variety of facial images is desirable. The parameter sets can vary considerably depending on the variability being modeled. The many kinds of variability being modeled/parameterized include the following.

- *Three-dimensional motion and pose*—the dynamic, 3D position and rotation of the head. Nonrigid face/head tracking involves estimating these parameters for each frame in the video sequence.
- *Facial action*—facial feature motion such as lip and eyebrow motion. Estimated by nonrigid tracking.
- *Shape and feature configuration*—the shape of the head, face and the facial features (e.g., mouth, eyes). This could be estimated by some alignment or facial landmark localization methods.
- *Illumination*—the variability in appearance due to different lighting conditions.
- *Texture and color*—the image pattern describing the skin.
- *Expression*—muscular synthesis of emotions making the face look, for example, happy or sad.

For a head tracker, the purpose is typically to extract the 3D motion parameters and be invariant to all other parameters. Whereas, for example, a user interface being sensitive to the mood of the user would need a model extracting the expression parameters, and a recognition system should typically be invariant to all but the shape and texture parameters.

### 18.2.1 Eigenfaces

Statistical texture models in the form of *eigenfaces* [30, 53, 60] have been popular for facial image analysis. The basic idea is that a training set of facial images are collected and registered, each image is reshaped into a vector, and a principal component analysis (PCA) is performed on the training set. The principal components are called eigenfaces. A facial image (in vector form),  $\mathbf{x}$ , can then be approximated by a linear combination,  $\hat{\mathbf{x}}$ , of these eigenfaces, that is,

$$\mathbf{x} \approx \hat{\mathbf{x}} = \bar{\mathbf{x}} + \boldsymbol{\Phi}_x \xi, \quad (18.1)$$

where  $\bar{\mathbf{x}}$  is the average of the training set,  $\boldsymbol{\Phi}_x = (\boldsymbol{\phi}_1 | \boldsymbol{\phi}_2 | \dots | \boldsymbol{\phi}_t)$  contains the eigenfaces, and  $\xi$  is a vector of weights or eigenface parameters. The parameters minimizing  $\|\mathbf{x} - \hat{\mathbf{x}}\|^2$  are given by

$$\xi = \boldsymbol{\Phi}_x^T (\mathbf{x} - \bar{\mathbf{x}}). \quad (18.2)$$

Commonly, some kind of image normalization is performed prior to eigenface computation.

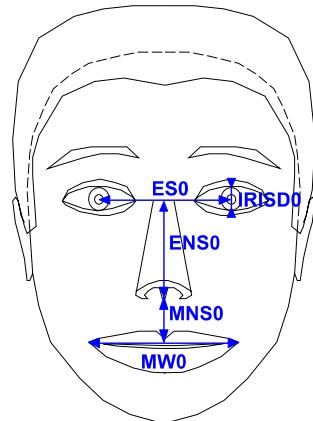
The space spanned by the eigenfaces is called the *face subspace*. Unfortunately, the manifold of facial images has a highly nonlinear structure and is thus not well modeled by a linear subspace. Linear and nonlinear techniques are available in the literature and often used for face recognition. For face tracking, it has been more popular to linearize the face manifold by warping the facial images to a standard pose and/or shape, thereby creating *shape-free* [10], *geometrically normalized* [55], or *shape-normalized* images and eigenfaces (texture templates, texture modes) that can be warped to any face shape or texture-mapped onto a wireframe face model.

### 18.2.2 Facial Action Coding System

During the 1960s and 1970s, a system for parameterizing minimal facial actions was developed by psychologists trying to analyze facial expressions. The system was called the *Facial Action Coding System* (FACS) [17] and describes each facial expression as a combination of around 50 *Action Units* (AUs). Each AU represents the activation of one facial muscle.

The FACS has been a popular tool not only for psychology studies but also for computerized facial modeling (an example is given in Sect. 18.2.5). There are also other models available in the literatures, for example, Park and Waters [49] described modeling skin and muscles in detail, which falls outside the scope of this chapter.

**Fig. 18.1** Face Animation Parameter Units (FAPU)



### 18.2.3 MPEG-4 Facial Animation

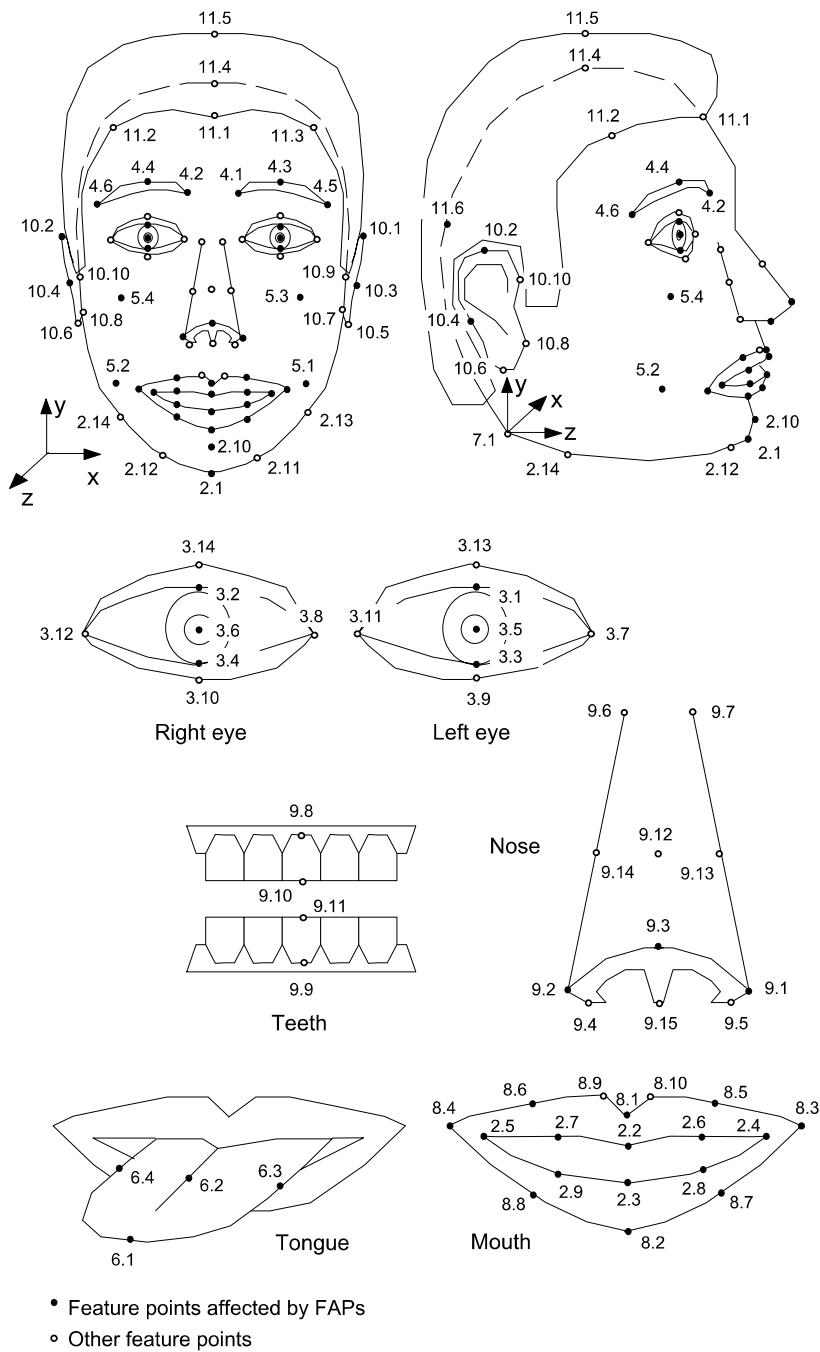
MPEG-4, since 1999 an international standard for coding and representation of audiovisual objects, contains definitions of face model parameters [41, 47].

There are two sets of parameters: *Facial Definition Parameters* (FDPs), which describe the static appearance of the head, and *Facial Animation Parameters* (FAPs), which describe the dynamics.

MPEG-4 defines 66 low-level FAPs and two high-level FAPs. The low-level FAPs are based on the study of minimal facial actions and are closely related to muscle actions. They represent a complete set of basic facial actions, and therefore allow the representation of most natural facial expressions. Exaggerated values permit the definition of actions that are normally not possible for humans, but could be desirable for cartoon-like characters.

All low-level FAPs are expressed in terms of the *Face Animation Parameter Units* (FAPUs), illustrated in Fig. 18.1. These units are defined in order to allow interpretation of the FAPs on any face model in a consistent way, producing reasonable results in terms of expression and speech pronunciation. They correspond to distances between key facial features and are defined in terms of distances between the MPEG-4 facial Feature Points (FPs, see Fig. 18.2). For each FAP it is defined on which FP it acts, in which direction it moves, and which FAPU is used as the unit for its movement. For example, FAP no. 3, open\_jaw, moves the Feature Point 2.1 (bottom of the chin) downwards and is expressed in MNS (mouth-nose separation) units. The MNS unit is defined as the distance between the nose and the mouth (see Fig. 18.1) divided by 1024. Therefore, in this example, a value of 512 for the FAP no. 3 means that the bottom of the chin moves down by half of the mouth-nose separation. The division by 1024 is introduced in order to have the units sufficiently small that FAPs can be represented in integer numbers.

The two high-level FAPs are *expression* and *viseme*. *expression* can contain two out of a predefined list of six basic expressions: joy, sadness, anger, fear, disgust and surprise. Intensity values allow to blend the two expressions. Similarly,

**Fig. 18.2** Facial Feature Points (FP)

`viseme` can contain two out of a predefined list of 14 visemes, and a blending factor to blend between them.

The neutral position of the face (when all FAPs are 0) is defined as follows:

- The coordinate system is right-handed; head axes are parallel to the world axes.
- Gaze is in the direction of Z axis.
- All face muscles are relaxed.
- Eyelids are tangent to the iris.
- The pupil is one third of IRISD0.
- Lips are in contact—the line of the lips is horizontal and at the same height of lip corners.
- The mouth is closed and the upper teeth touch the lower ones.
- The tongue is flat, horizontal with the tip of tongue touching the boundary between upper and lower teeth (feature point 6.1 touching 9.11, see Fig. 18.2).

All FAPs are expressed as displacements from the positions defined in the neutral face.

The FDPs describe the static shape of the face by the 3D coordinates of each feature point and the texture as an image with the corresponding texture coordinates.

#### 18.2.4 Computer Graphics Models

When synthesizing faces using computer graphics (for user interfaces [25], web applications [45], computer games, or special effects in movies), the most common model is a *wireframe model* or a *polygonal mesh*. The face is then described as a set of vertices connected with lines forming polygons (usually triangles). The polygons are shaded or texture-mapped, and illumination is added. The texture could be parameterized or fixed—in the latter case facial appearance is changed by moving the vertices only. To achieve life-like animation of the face, a large number (thousands) of vertices and polygons are commonly used. Each vertex can move in three dimensions, so the model requires a large number of degrees of freedom. To reduce this number, some kind of parameterization is needed.

A commonly adopted solution is to create a set of *morph targets* and blend between them. A morph target is a predefined set of vertex positions, where each morph target represents, for example, a facial expression or a viseme. Thus, the model shape is defined by the morph targets  $\mathbf{A}$  and controlled by the parameter vector  $\alpha$

$$\begin{aligned} \mathbf{g} &= \mathbf{A}\alpha, \\ \sum \alpha_i &= 1. \end{aligned} \tag{18.3}$$

The  $3N$ -dimensional vector  $\mathbf{g}$  contains the 3D coordinates of the  $N$  vertices; the columns of the  $3N \times M$ -matrix  $\mathbf{A}$  contain the  $M$  morph targets; and  $\alpha$  contains the  $M$  morph target coefficients. To limit the required computational complexity, most  $\alpha_i$  values are usually zero.

Morph targets are usually manually created by artists. This is a time consuming process so automatic methods have been devised to copy a set of morph targets from one face model to another [20, 43, 46].

To render the model on the screen, we need to find a correspondence from the model coordinates to image pixels. The projection model (see Sect. 18.2.6), which is not defined by the face model, defines a function  $P(\cdot)$  that projects the vertices on the image plane

$$(u, v) = P(x, y, z). \quad (18.4)$$

The image coordinate system is typically defined over the range  $[-1, 1] \times [-1, 1]$  or  $[0, w - 1] \times [0, h - 1]$ , where  $(w, h)$  is the image dimensions in pixels.

To texturize the model, each vertex is associated with a (prestored) texture coordinate  $(s, t)$  defined on the unit square. Using some interpolating scheme (e.g., piecewise affine transformations), we can find a correspondence from any point  $(x, y, z)$  on the model surface to a point  $(s, t)$  in the texture image and a point  $(u, v)$  in the rendered image. Texture mapping is executed by copying (interpolated) pixel values from the texture  $\mathbf{I}_t(s, t)$  to the rendered image of the model  $\mathbf{I}_m(u, v)$ . We call the coordinate transform  $T_{\mathbf{u}}(\cdot)$ , and thus

$$\mathbf{I}_m(u, v) = T_{\mathbf{u}}[\mathbf{I}_t(s, t)]. \quad (18.5)$$

While morphing is probably the most commonly used facial animation technique, it is not the only one. Skinning, or bone-based animation, is the process of applying one or more transformation matrices, called bones, to the vertices of a polygon mesh in order to obtain a smooth deformation, for example, when animating joints such as elbow or shoulder. Each bone has a weight that determines its influence on the vertex, and the final position of the vertex is the weighted sum of the results of all applied transformations. While the main purpose of skinning is typically body animation, it has been applied very successfully to facial animation. Unlike body animation, where the configuration of bones resembles the anatomical human skeleton, for facial animation the bones rig is completely artificial and bears almost no resemblance to human skull. Good starting references for skinning are [29, 39]. There are numerous other approaches to facial animation, ranging from ones based on direct modeling of observed motion [48], to pseudo muscle models [35] and various degrees of physical simulation of bone, muscle and tissue dynamics [27, 59, 61].

More on computerized facial animation can be found in [47, 49]. Texture mapping is treated in [23].

### 18.2.5 Candide: A Simple Wireframe Face Model

Candide is a simple face model that has been a popular research tool for many years. It was originally created by Rydfalk [52] and later extended by Welsh [62] to cover the entire head (Candide-2) and by Ahlberg [2, 3] to correspond better to

MPEG-4 facial animation (Candide-3). The simplicity of the model makes it a good pedagogic example.

Candide is a wireframe model with 113 vertices connected by lines forming 184 triangular surfaces. The geometry (shape, structure) is determined by the 3D coordinates of the vertices in a model-centered coordinate system ( $x, y, z$ ). To modify the geometry, Candide-1 and Candide-2 implement a set of Action Units from FACS. Each AU is implemented as a list of vertex displacements, an *Action Unit Vector* (AUV), describing the change in face geometry when the Action Unit is fully activated. The geometry is thus parameterized as

$$\mathbf{g}(\alpha) = \bar{\mathbf{g}} + \Phi_a \alpha, \quad 0 \leq \alpha_i \leq 1 \quad (18.6)$$

where the resulting vector  $\mathbf{g}$  contains the  $(x, y, z)$  coordinates of the vertices of the model,  $\bar{\mathbf{g}}$  is the standard shape of the model, and the columns of  $\Phi_a$  are the AUVs. The  $\alpha_i$  values are the Action Unit activation levels.

Candide-3 is parameterized slightly different than the previous versions, generalizing the AUVs to animation modes (implementing AUs or FAPs) and adding shape modes. The parameterization is

$$\mathbf{g}(\alpha, \sigma) = \bar{\mathbf{g}} + \Phi_a \alpha + \Phi_s \sigma. \quad (18.7)$$

The difference between  $\alpha$  and  $\sigma$  is that the shape parameters control the static deformations that cause individuals to differ from each other. The animation parameters control the dynamic deformations due to facial action.

This kind of linear model is, in different variations, a common way to model facial geometry. For example, PCA found a matrix that described 2D shape and animation modes combined, Gokturk et al. [22] estimated 3D animation modes using a stereo-vision system, and Caunce et al. [9] created a shape model from models adapted to profile and frontal photos.

To change the pose, the model coordinates are rotated, scaled and translated so that

$$\mathbf{g}(\alpha, \sigma, \pi) = s \mathbf{R} \mathbf{g}(\alpha, \sigma) + \mathbf{t} \quad (18.8)$$

where  $\pi$  contains the six pose/global motion parameters plus a scaling factor.

### 18.2.6 Projection Models

The function  $(u, v) = P(x, y, z)$ , above, is a general projection model representing the camera. There are various projection models from which to chose, each with a set of parameters that may be known (calibrated camera) or unknown (uncalibrated camera). In most applications, the camera is assumed to be at least partly calibrated. We stress that only simple cases are treated here, neglecting such camera parameters as skewness and rotation. For more details, consult a computer vision textbook like [54].

- The simplest projection model is the *orthographic projection*—basically just throwing away the  $z$ -coordinate. Parameters are pixel size  $(a_u, a_v)$  and principal point  $(c_u, c_v)$ . The projection can be written

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a_u & 0 & 0 & c_u \\ 0 & a_v & 0 & c_v \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}. \quad (18.9)$$

- The most common camera model is the *perspective projection*, which can be expressed as

$$\begin{cases} (u', v', w') = \mathbf{P}(x, y, z, 1)^T, \\ (u, v) = (u'/w', v'/w') \end{cases} \quad (18.10)$$

where

$$\mathbf{P} = \begin{pmatrix} a_u & 0 & 0 & c_x \\ 0 & a_v & 0 & c_y \\ 0 & 0 & 1/f & c_z \end{pmatrix}, \quad (18.11)$$

$(c_x, c_y, c_z)$  is the focus of expansion (FOE), and  $f$  is the focal length of the camera;  $(a_u, a_v)$  determines the pixel size. Commonly, a simple expression for  $\mathbf{P}$  is obtained where  $(c_x, c_y, c_z) = \mathbf{0}$  and  $a_u = a_v = 1$  are used. In this case, (18.10) is simply

$$(u, v) = \left( f \frac{x}{z}, f \frac{y}{z} \right). \quad (18.12)$$

- The *weak perspective projection* is an approximation of the perspective projection suitable for an object where the internal depth variation is small compared to the distance  $z_{\text{ref}}$  from the camera to the object.

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a_u/z_{\text{ref}} & 0 & 0 & c_x \\ 0 & a_v/z_{\text{ref}} & 0 & c_y \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}. \quad (18.13)$$

### 18.3 Tracking Strategies

A face tracking system estimates the rigid or non-rigid motion of a face through a sequence of image frames. In the following, we discuss the two-frame situation, where we have an estimation of the model parameters  $\hat{\mathbf{p}}_{k-1}$  in the old frame, and the system should estimate the parameters  $\hat{\mathbf{p}}_k$  in the new frame (i.e., how to transform the model from the old frame to the new frame).

As mentioned in the introduction, we discuss only monocular tracking here, that is, we disregard stereo vision systems.

### 18.3.1 Motion-Based vs. Model-Based Tracking

Tracking systems can be said to be either *motion-based* or *model-based*, also referred to as *feed-forward* or *feed-back* motion estimation. A *motion-based* tracker estimates the displacements of pixels (or blocks of pixels) from one frame to another. The displacements might be estimated using optical flow methods (giving a dense optical flow field), block-based motion estimation methods (giving a sparse field but using less computational power), or motion estimation in a few image patches only (giving a few motion vectors only but at very low computational cost).

The estimated motion field is used to compute the motion of the object model using, for example, least-squares, Kalman filtering, or some optimization method. The motion estimation in such a method is consequently dependent on the pixels in two frames; the object model is used only for transforming the 2D motion vectors to 3D object model motion. The problem with such methods is the *drifting* or the *long sequence motion problem*. A tracker of this kind accumulates motion errors and eventually loses track of the face. Examples of such trackers can be found in the literature [7, 8, 51].

A *model-based* tracker, on the other hand, uses a model of the object's appearance and tries to change the object model's pose (and possibly shape) parameters to fit the new frame. The motion estimation is thus dependent on the object model and the new frame—the old frame is not regarded except for constraining the search space. Such a tracker does not suffer from drifting; instead, problems arise when the model is not strong or flexible enough to cope with the situation in the new frame. Trackers of this kind can be found in certain articles [5, 31, 33, 56]. Other trackers [11, 13, 22, 36] are motion-based but add various model-based constraints to improve performance and combat drift.

### 18.3.2 Model-Based Tracking: First Frame Models vs. Pre-trained Models

In general, the word *model* refers to any prior knowledge about the 3D structure, the 3D motion/dynamics, and the 2D facial appearance. One of the main issues when designing a model-based tracker is the appearance model. An obvious approach is to capture a reference image of the object from the first frame of the sequence. The image could then be geometrically transformed according to the estimated motion parameters, so one can compensate for changes in scale and rotation (and possibly nonrigid motion). Because the image is captured, the appearance model is deterministic, object-specific, and (potentially) accurate. Thus, trackers of this kind can be precise, and systems working in real time have been demonstrated [22, 32, 56, 63].

A drawback with such a first frame model is the lack of flexibility—it is difficult to generalize from one sample only. This can cause problems with changing appearance due to variations in illumination, facial expression, and other factors. Another

drawback is that the initialization is critical; if the captured image was for some reason not representative for the sequence (due to partial occlusion, facial expression, or illumination) or simply not the correct image (i.e., if the object model was not correctly aligned in the first frame) the tracker does not work well. Such problems can usually be solved by manual interaction but may be hard to automate.

Note that the model could be renewed continuously (sometimes called an *online* model), so the model always is based on the previous frame. In this way the problems with flexibility are reduced, but the tracker is then motion-based and might drift.

Another property is that the tracker does not need to know what it is tracking. This could be an advantage—the tracker can track different kinds of objects—or a disadvantage. A relevant example is when the goal is to extract some higher level information from a human face, such as facial expression or lip motion. In that case we need a tracker that identifies and tracks specific facial features (e.g., lip contours or feature points).

A different approach is a *pre-trained model-based tracker*. Here, the appearance model relies on previously captured images combined with knowledge of which parts or positions of the images correspond to the various facial features. When the model is transformed to fit the new frame, we thus obtain information about the estimated positions of those specific facial features.

The appearance model may be person specific or general. A specific model could, for example, be trained on a database containing images of one person only, resulting in an accurate model for this person. It could cope, to some degree, with the illumination and expression changes present in the database. A more general appearance model could be trained on a database containing many faces in different illuminations and with different facial expressions. Such a model would have a better chance to enable successful tracking of a previously unseen face in a new environment, whereas a specific appearance model presumably would result in better performance on the person and environment for which it was trained. Trackers using pre-trained models of appearance can be found in the literature [5, 31].

### 18.3.3 Appearance-Based vs. Feature-Based Tracking

An *appearance-based* or *featureless* or *generative model-based* tracker matches a model of the entire facial appearance with the input image, trying to exploit all available information in the model as well as the image. Generally, we can express this as follows:

Assume a parametric generative face model and an input image  $\mathbf{I}$  of a face from which we want to estimate a set of parameters. The parameters to be extracted should form a subset of parameter set controlling the model. Given a vector  $\mathbf{p}$  with  $N$  parameters, the face model can generate an image  $\mathbf{I}_m(\mathbf{p})$ . The principle of analysis-by-synthesis then states that the best estimates of the facial parameters are the ones

minimizing the distance between the generated image and the input image

$$\mathbf{p}^* = \arg \min_{\mathbf{p}} \delta[\mathbf{I}, \mathbf{I}_m(\mathbf{p})] \quad (18.14)$$

for some distance measure  $\delta(\cdot)$ .

The problem of finding the optimal parameters is a high-dimensional ( $N$  dimensions) search problem and thus of high computational complexity. By using clever search heuristics, we can reduce the search time. The trackers described in [5, 31, 33, 36] are appearance-based.

A *feature-based* tracker, on the other hand, chooses a few facial features that are, supposedly, easily and robustly tracked. Features such as color, specific points or patches, and edges can be used. Typically, a tracker based on feature points tries to estimate the 2D position of a set of points and from these points to compute the 3D pose of the face. Feature-based trackers can be found in [11, 12, 26, 56, 63].

In the following sections, we describe three trackers found in the literature. They represent the classes mentioned above.

## 18.4 Feature-Based Tracking Example

The tracker described in this section tracks a set of feature points in an image sequence and uses the 2D measurements to calculate the 3D structure and motion of the face and the facial features. The tracker is based on the structure from motion (SfM) algorithm by Azerbayejani and Pentland [6]. The (rigid) face tracker was then developed by Jebara and Pentland [26] and further by Ström et al. [56, 57]. The tracker was later extended as to handle non-rigid motion, and thus track facial action by Ingemars and Ahlberg [24].

With the terminology above, it is the first frame model-based and feature-based tracker. We stress that the presentation here is somewhat simplified.

### 18.4.1 Face Model Parameterization

The head tracker designed by Jebara and Pentland [26] estimated a model as a set of points with no surface. Ström et al. [57] extended the system to include a 3D wireframe face model (Candide). A set of feature points are placed on the surface of the model, not necessarily coinciding with the model vertices. The 3D face model enables the system to predict the surface angle relative to the camera as well as self-occlusion. Thus, the tracker can predict when some measurements should not be trusted.

### 18.4.1.1 Pose Parameterization

The pose in the  $k$ th frame is parameterized with three rotation angles  $(r_x, r_y, r_z)$ , three translation parameters  $(t_x, t_y, t_z)$ , and the inverse focal length  $\phi = 1/f$  of the camera.<sup>1</sup>

Azerbayejani and Pentland [6] chose to use a perspective projection model where the origin of the 3D coordinate system is placed in the center of the image plane instead of at the focal point, that is, the FOE is set to  $(0, 0, 1)$  (see Sect. 18.2.6). This projection model has several advantages; for example, there is only one unknown parameter per feature point (as becomes apparent below).

Thus, the 2D (projected) screen coordinates are computed as

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{1}{1 + z\phi} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (18.15)$$

### 18.4.1.2 Structure Parameterization

The structure of the face is represented by the image coordinates  $(u_0, v_0)$  and the depth values  $z_0$  of the feature points in the first frame. If the depths  $z_0$  are known, the 3D coordinates of the feature points can be computed for the first frame as

$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = (1 + z_0\phi) \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} \quad (18.16)$$

and for all following frames as

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{R} \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (18.17)$$

where  $\mathbf{R}$  is the rotation matrix created. For clarity of presentation, the frame indices on all the parameters are omitted.

All put together, the model parameter vector is

$$\mathbf{p} = (t_x, t_y, t_z, r_x, r_y, r_z, \phi, z_1, \dots, z_N)^T \quad (18.18)$$

where  $N$  is the number of feature points and  $r_x, r_y, r_z$  are used to update  $\mathbf{R}$ . Combining (18.16), (18.17), and (18.15), we get a function from the parameter vector to screen coordinates

$$(u_1, v_1, \dots, u_N, v_N)^T = h_k(\mathbf{p}). \quad (18.19)$$

Note that the parameter vector has  $N + 7$  degrees of freedom, and that we get  $2N$  values if we measure the image coordinates for each feature point. Thus, the problem of estimating the parameter vector from image coordinates is overconstrained when  $N > 7$ .

---

<sup>1</sup>In practice, the  $z$ -translation should be parameterized by  $\zeta = t_z\phi$  instead of  $t_z$  for stability reasons.

### 18.4.2 Parameter Estimation Using an Extended Kalman Filters

A Kalman filter is used to estimate the dynamic changes of a state vector of which a function can be observed. When the function is nonlinear, we must use an extended Kalman filter (EKF). The literature on Kalman filtering is plentiful [21, 28, 54, 55], so we only summarize the approach here.

In our case, the state vector is the model parameter vector  $\mathbf{p}$  and we observe, for each frame, the screen coordinates  $\mathbf{u}_k = h_k(\mathbf{p}_k)$ . Because we cannot measure the screen coordinates exactly, measurement noise  $\mathbf{v}_k$  is added as well. We can summarize the dynamics of the system as

$$\begin{cases} \mathbf{p}_{k+1} = f_k(\mathbf{p}_k) + \mathbf{w}_k, & \mathbf{w}_k \sim N(\mathbf{0}, \mathbf{W}_k), \\ \hat{\mathbf{u}}_k = h_k(\mathbf{p}_k) + \mathbf{v}_k, & \mathbf{v}_k \sim N(\mathbf{0}, \mathbf{V}_k) \end{cases} \quad (18.20)$$

where  $f_k(\cdot)$  is the dynamics function,  $h_k(\cdot)$  is the measurement function, and  $\mathbf{w}$  and  $\mathbf{v}$  are zero-mean Gaussian random variables with known covariances  $\mathbf{W}_k$  and  $\mathbf{V}_k$ .

The job for the EKF is to estimate the state vector  $\mathbf{p}_k$  given the measurement vector  $\hat{\mathbf{u}}_k$  and the previous estimate  $\hat{\mathbf{p}}_{k-1}$ . Choosing the trivial dynamics function  $f_k(\mathbf{p}_k) = \mathbf{p}_k$ , the state estimate is updated using

$$\hat{\mathbf{p}}_k = \hat{\mathbf{p}}_{k-1} + \mathbf{K}_k [\hat{\mathbf{u}}_k - h_k(\hat{\mathbf{p}}_{k-1})] \quad (18.21)$$

where  $\mathbf{K}_k$  is the Kalman gain matrix. It is updated every time step depending on  $f_k(\cdot)$ ,  $h_k(\cdot)$ ,  $\mathbf{W}_k$ , and  $\mathbf{V}_k$ . The covariances are given by initial assumptions or estimated during the tracking.

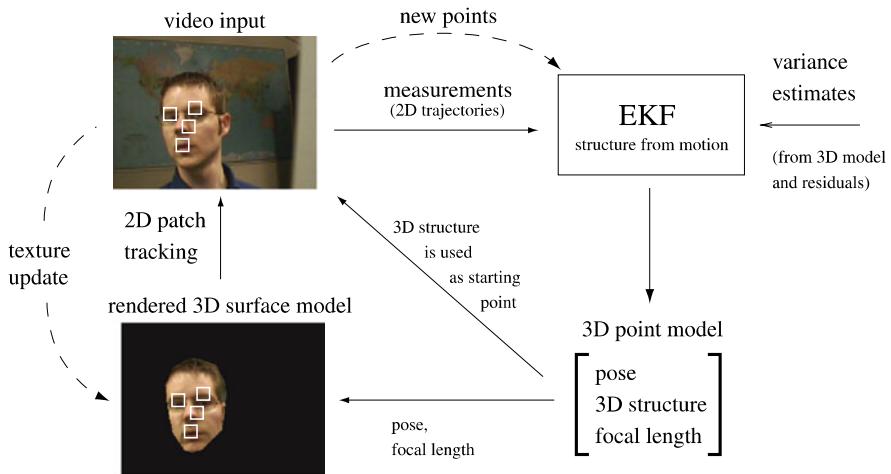
### 18.4.3 Tracking Process

The tracker must be initialized, manually or by using a face detection algorithm. The model reference texture is captured from the first frame, and feature points are automatically extracted. To select feature points that could be reliably tracked, points where the determinant of the Hessian

$$\det(H) = \begin{vmatrix} \mathbf{I}_{xx}(x, y) & \mathbf{I}_{xy}(x, y) \\ \mathbf{I}_{xy}(x, y) & \mathbf{I}_{yy}(x, y) \end{vmatrix} \quad (18.22)$$

is large are used. The function `cvGoodFeaturesToTrack` in OpenCV [44] implements a few variations. The determinant is weighted with the cosine of the angle between the model surface normal and the camera direction. The number of feature points to select is limited only by the available computational power and the real-time requirements.

The initial feature point depth values  $(z_1, \dots, z_N)$  are given by the 3D face model. Then, for each frame, the model is rendered using the current model parameters. Around each feature point, a small patch is extracted from the rendered image.



**Fig. 18.3** Patches from the rendered image (*lower left*) are matched with the incoming video. The two-dimensional feature point positions are fed to the EKF, which estimates the pose information needed to render the next model view. For clarity, only 4 of 24 patches are shown. Illustration courtesy of J. Ström

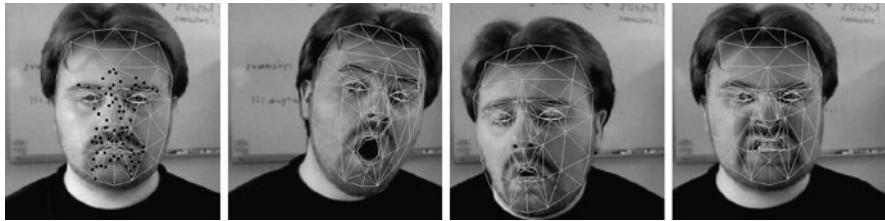
The patches are matched with the new frame using a zero-mean normalized cross correlation. The best match, with subpixel precision, for each feature point is collected in the measurement vector  $\hat{\mathbf{u}}_k$  and fed into the EKF update equation (18.21).

Using the face model and the values from the normalized template matching, the measurement noise covariance matrix can be estimated making the EKF rely on some measurements more than others. Note that this also tells the EKF in which directions in the image the measurements are reliable. For example, a feature point on an edge (e.g., the mouth outline) can reliably be placed in the direction perpendicular to the edge but less reliably along the edge. The system is illustrated in Fig. 18.3.

#### 18.4.4 Tracking of Facial Action

Ingemars and Ahlberg [24] extended the tracker to track facial action as well. A set of states was added to the state vector corresponding to the animation parameters of Candidate-3. In order to be able to track these, there must be feature points within the area that is influenced by each estimated animation parameter, and the number of feature points must be higher. However, the automatic feature point selection can still be used, since the observation function can be calculated online by interpolation of the known observation functions of the face model vertices.

In order to be both accurate and to handle drift, the tracker combines feature point patches from the first frame of the sequence with patches dynamically extracted from the previous frame. Examples are shown in Fig. 18.4.



**Fig. 18.4** Tracking example, feature-based tracking. The *leftmost image* shows the automatically extracted tracking points. Images courtesy of N. Ingemars

## 18.5 Appearance-Based Tracking Example

In this section, we describe a statistical model-based and appearance-based tracker estimating the 3D pose and deformations of the face. It is based on the Active Appearance Models (AAMs) described in Chap. 5. The tracker was first presented by Ahlberg [1], and was later improved in various ways as described in Sect. 18.5.4.

To use the AAM search algorithm, we must first decide on a geometry for the face model, its parameterization of shape and texture, and how we use the model for warping images. Here, we use the face model Candide-3 described in Sect. 18.2.5.

### 18.5.1 Face Model Parameterization

#### 18.5.1.1 Geometry Parameterization

The geometry (structure)  $\mathbf{g}(\sigma, \alpha)$  of the Candide model is parameterized according to (18.7). There are several techniques to estimate the shape parameters  $\sigma$  (e.g., an AAM search or facial landmark localization method). When adapting a model to a video sequence, the shape parameters should be changed only in the first frame(s)—the head shape does not vary during a conversation—whereas the pose and animation parameters naturally change at each frame. Thus, during the tracking process we can assume that  $\sigma$  is fixed (and known), and let the shape depend on  $\alpha$  only

$$\begin{cases} \bar{\mathbf{g}}_\sigma = \bar{\mathbf{g}} + \Phi_s \sigma, \\ \mathbf{g}_\sigma(\alpha) = \bar{\mathbf{g}}_\sigma + \Phi_a \alpha. \end{cases} \quad (18.23)$$

#### 18.5.1.2 Pose Parameterization

To perform global motion (i.e., pose change), we need six parameters plus a scaling factor according to (18.8). Since the Candide model is defined only up to scale, we can adopt the weak perspective projection and combine scale and  $z$ -translation in one parameter. Thus, using the 3D translation vector  $\mathbf{t} = (t_x, t_y, t_z)^T$  and the three Euler angles for the rotation matrix  $\mathbf{R}$ , our pose parameter vector is

$$\pi = (r_x, r_y, r_z, , t_x, t_y, t_z)^T. \quad (18.24)$$

### 18.5.1.3 Texture Parameterization

We use a statistical model of the texture and control the texture with a small set of texture parameters  $\xi$ . The model texture vector  $\hat{\mathbf{x}}$  is generated according to (18.1) (see Sect. 18.2.1). The synthesized texture vector  $\hat{\mathbf{x}}$  has for each element a corresponding  $(s, t)$  coordinate and is equivalent to the texture image  $\mathbf{I}_t(s, t)$ : the relation is just lexicographic ordering,  $\hat{\mathbf{x}} = L[\mathbf{I}_t(s, t)]$ .  $\mathbf{I}_t(s, t)$  is mapped on the wireframe model to create the generated image  $\mathbf{I}_m(u, v)$  according to (18.5).

The entire appearance of the model can now be controlled using the parameters  $(\xi, \pi, \alpha, \sigma)$ . However, as we assume that the shape  $\sigma$  is fixed during the tracking session, and the texture  $\xi$  depends on the other parameters, the parameter vector we optimize below is

$$\mathbf{p} = (\pi^T, \alpha^T)^T. \quad (18.25)$$

### 18.5.2 Tracking Process

The tracker should find the optimal adaptation of the model to each frame in a sequence as described in Sect. 18.3.3. That is, we wish to find the parameter vector  $\mathbf{p}_k^*$  that minimizes the distance between image  $\mathbf{I}_m$  generated by the model and each frame  $\mathbf{I}_k$ . Here, an iterative solution is presented, and as an initial value of  $\mathbf{p}$  we use  $\hat{\mathbf{p}}_{k-1}$  (i.e., the estimated parameter vector from the previous frame).

Instead of directly comparing the generated image  $\mathbf{I}_m(u, v)$  to the input image  $\mathbf{I}_k(u, v)$ , we back-project the input image to the model's parametric surface coordinate system  $(s, t)$  using the inverse of the texture mapping transform  $T_u$

$$\mathbf{I}_{u(\mathbf{p})}(s, t) = T_{u(\mathbf{p})}^{-1}[\mathbf{I}_k(u, v)], \quad (18.26)$$

$$\mathbf{x}(\mathbf{p}) = L[\mathbf{I}_{u(\mathbf{p})}(s, t)]. \quad (18.27)$$

We then compare the normalized input image vector  $\mathbf{x}(\mathbf{p})$  to the generated model texture vector  $\hat{\mathbf{x}}(\mathbf{p})$ .  $\hat{\mathbf{x}}(\mathbf{p})$  is generated in the face subspace as closely as possible to  $\mathbf{x}(\mathbf{p})$  (see (18.1)), and we compute a residual image  $\mathbf{r}(\mathbf{p}) = \mathbf{x}(\mathbf{p}) - \hat{\mathbf{x}}(\mathbf{p})$ . The process from input image to residual, is illustrated in Fig. 18.5.

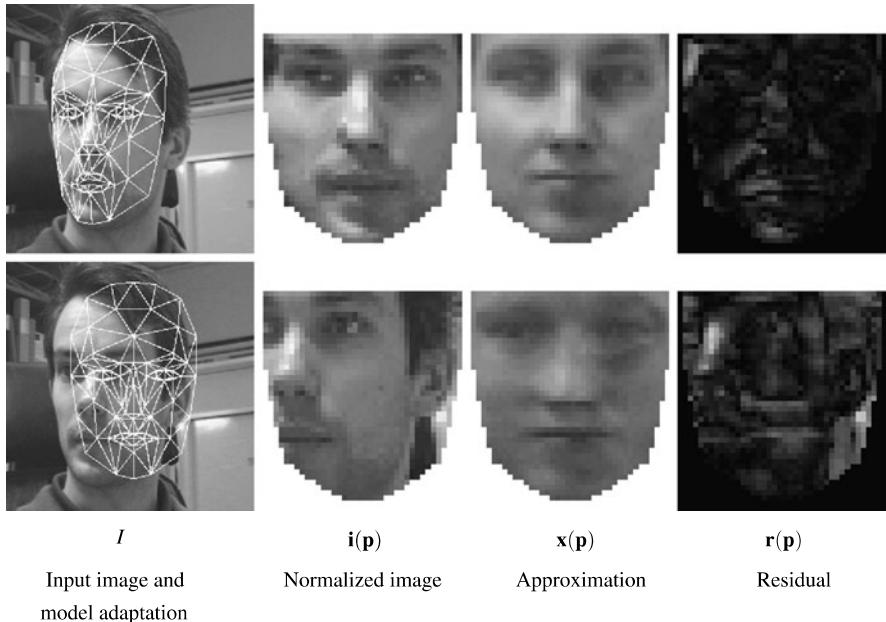
As the distance measures according to (18.14), we use the squared norm of the residual image

$$\delta(\mathbf{I}_k, \mathbf{I}_m(\mathbf{p})) = \|\mathbf{r}(\mathbf{p})\|^2. \quad (18.28)$$

From the residual image, we also compute the update vector

$$\Delta \mathbf{p} = -\mathbf{U}\mathbf{r}(\mathbf{p}) \quad (18.29)$$

where  $\mathbf{U} = (\frac{\delta \mathbf{r}}{\delta \mathbf{p}})^\dagger$  is the precomputed active appearance update matrix (i.e., the pseudo-inverse of the estimated gradient matrix  $\frac{\delta \mathbf{r}}{\delta \mathbf{p}}$ ). It is created by numeric differentiation, systematically displacing each parameter and computing an average over the training set.



**Fig. 18.5** Analysis-synthesis process. A good and a bad model adaptation. The more similar the normalized image and its approximation is, the better the model adaptation is

We then compute the estimated new parameter vector as

$$\hat{\mathbf{p}}_k = \mathbf{p} + \Delta \mathbf{p}. \quad (18.30)$$

In most cases, the model fitting is improved if the entire process is iterated a few times.

### 18.5.3 Tracking Example

To illustrate, a video sequence of a previously unseen person was used and the Candide-3 model was manually adapted to the first frame of the sequence by changing the pose parameters and the static shape parameters (recall that the shape parameter vector  $\sigma$  is assumed to be known). The model parameter vector  $\mathbf{p}$  was then iteratively optimized for each frame, as is shown in Fig. 18.6.

Note that this, quite simple, tracker needs some more development in order to be robust to varying illumination, strong facial expressions, and large head motion. Moreover, it is very much depending on the training data.



**Fig. 18.6** Tracking example, appearance-based tracking. Every tenth frame shown

#### 18.5.4 Improvements

In 2002, Dornaika and Ahlberg [12, 13] introduced a feature/appearance-based tracker. The head pose is estimated using a RANdom SAmpling Consensus (RANSAC) technique [19] combined with a texture consistency measure to avoid drifting. Once the 3D head pose  $\pi$  is estimated, the facial animation parameters  $\alpha$  can be estimated using the scheme described in Sect. 18.5.

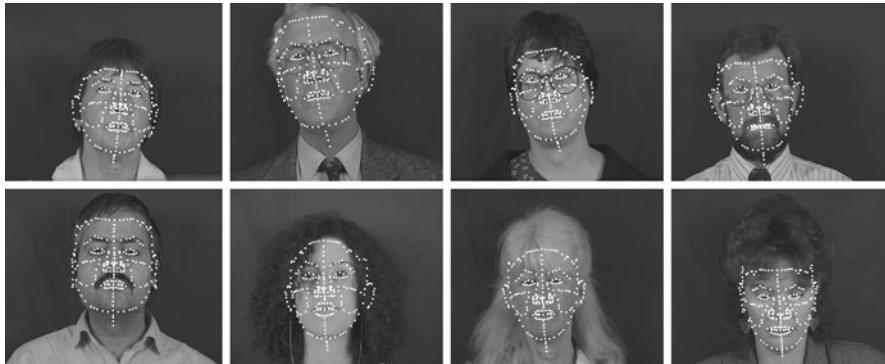
In 2004, Matthews and Baker [37] proposed the Inverse Compositional Algorithm which allows for an accurate and fast fitting, actually reversing the roles of the input image and the template in the well-known, but slow, Lucas–Kanade Image Alignment algorithm [34]. Supposing that the appearance will not change much among different frames, it can be “projected out” from the search space. Fanelli and Fratarvangeli [18] incorporated the Inverse Compositional Algorithm in an AAM-based tracker to improve robustness.

Zhou et al. [65] and Dornaika and Davoine [15] combined appearance models with a particle filter for improved 3D pose estimation.

## 18.6 Fused Trackers

### 18.6.1 Combining Motion- and Model-Based Tracking

In order to exploit the advantages of the various tracking strategies mentioned above, modern state-of-the-art trackers often combine them. For example, Lefèvre and Odobez [32] used the Candide model and two sets of feature points. The first set is called the “trained set”, that is, a set of feature points trained from the first frame of the video sequence. The second set is the “adaptive set”, that is continuously adapted during tracking. Using only one of these sets (the first corresponding to first frame model-based and the second to motion-based, according to the terminology used here) results in a certain kinds of tracking failures. While tracking using the adaptive method (motion-based tracking) is more precise, it is also prone to drift. Lefèvre and Odobez devised a hybrid method exploiting the advantages of both the adaptive



**Fig. 18.7** Face model alignment examples. Images courtesy of A. Caunce

and the trained method. Compared to the method by Ingemars and Ahlberg [24] described above, Lefèvre and Odobez made a more thorough investigation of how to exploit the two methods. Another major difference is the choice of methods for 3D pose estimation from 2D measurements (EKF and Nelder–Mead downhill simplex optimization [42], respectively). Notably, Lefèvre and Odobez demonstrated stable tracking under varying lighting conditions.

### 18.6.2 Combining Appearance- and Feature-Based Tracking

Another recent development was published by Caunce et al. [9], combining the feature-based and appearance-based approaches. In the tradition of the Manchester group (where the AAMs originated), the work was started with thorough statistical model-building of shape and texture of the face. Using profile and frontal photographies, a 3D shape model was adapted to each training subject and a PCA performed. A set of texture patches were extracted from the mean texture, and used to adapt the 3D model to a new image in a two stage process. The first stage consists of the rigid body adaptation (pose, translation) and the second step the facial shape (restrained by the pretrained model). Moreover, a scale hierarchy was used.

Thus, this method is primarily aimed at facial landmark localization/face model alignment, but is used (by Caunce et al.) for tracking of facial action as well by letting the parameter estimated from the previous frame serving as the initial estimation. Example results are shown in Fig. 18.7 using XM2VTSDB [38] imagery.

### 18.6.3 Commercially Available Trackers

Naturally, there are a number of commercial products providing head, face and facial action tracking. Below, some of these are mentioned.

One of the best commercial facial trackers was developed in the 1990s by a US company called Eyematic. The company went bankrupt and the technology was taken over by another company which subsequently sold it to Google.

The Australian company Seeing Machines is offering a face tracker API called faceAPI that tracks head motion, lips and eyebrows.

Visage Technologies (Sweden) provides a statistical appearance-based tracker as well as an feature-based tracker in an API called visage|SDK.

The Japanese company Omron developed a suite of face sensing technology called OKAO Vision that includes face detection, tracking of the face, lips, eyebrows and eyes, as well as estimation of attributes such as gender, ethnicity and age.

Image Metrics (UK) provides high-end performance-based facial animation service to film and game producers based on their in-house facial tracking algorithms.

Mova (US) uses a radically different idea in order to precisely capture high-density facial surface data using phosphorescent makeup and fluorescent lights.

Other companies with face tracking products include Genemation (UK), OKI (Japan), SeeStorm (Russia) and Visual Recognition (Netherlands). Face and facial feature tracking is also used in end-user products such as Fix8 by Mobicnex (US) and the software bundled with certain Logitech webcams.

## 18.7 Conclusions

We have described some strategies for tracking and distinguished between model- and motion-based tracking as well as between appearance- and feature-based tracking. Whereas motion-based trackers may suffer from drifting, model-based trackers do not have that problem. Appearance- and feature-based trackers follow different basic principles and have different characteristics.

Two trackers have been described, one feature-based and one appearance-based. Both trackers are all model-based and thus do not suffer from drifting. Improvements found in the literature are discussed for both trackers.

Trackers combining the described tracking strategies, presumably representing the state-of-the-art, have been described, and commercially available tackers have been briefly mentioned.

## References

1. Ahlberg, J.: An active model for facial feature tracking. In: Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), Tampere, Finland, pp. 63–67 (2001)
2. Ahlberg, J.: Candide 3—an updated parameterized face. Technical Report LiTH-ISY-R-2326, Linköping University, Sweden (2001)
3. Ahlberg, J.: Model-based coding—extraction, coding and evaluation of face model parameters. PhD thesis, Linköping University, Sweden (2002)

4. Ahlberg, J.: An active model for facial feature tracking. *EURASIP J. Appl. Signal Process.* **2002**(6), 566–571 (2002)
5. Ahlberg, J., Forchheimer, R.: Face tracking for model-based coding and face animation. *Int. J. Imaging Syst. Technol.* **13**(1), 8–22 (2003)
6. Azerbayejani, A., Pentland, A.: Recursive estimation of motion, structure, and focal length. *IEEE Trans. Pattern Anal. Mach. Intell.* **17**(6), 562–575 (1995)
7. Basu, S., Essa, I., Pentland, A.: Motion regularization for model-based head-tracking. In: *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, Vienna, Austria, pp. 611–616 (1996)
8. Black, M.J., Yacoob, Y.: Recognizing facial expressions in image sequences using local parameterized models of image motion. *Int. J. Comput. Vis.* **25**(1), 23–48 (1997)
9. Caunce, A., Cristinacce, D., Taylor, C., Cootes, T.: Locating facial features and pose estimation using a 3D shape model. In: *Proceedings of the International Symposium on Visual Computing*, Las Vegas, USA (2009)
10. Craw, I., Cameron, P.: Parameterising images for recognition and reconstruction. In: *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 367–370. Springer, London (1991)
11. DeCarlo, D., Metaxas, D.: Optical flow constraints on deformable models with applications to face tracking. *Int. J. Comput. Vis.* **38**(72), 99–127 (2000)
12. Dornaika, F., Ahlberg, J.: Face model adaptation using robust matching and the active appearance algorithm. In: *Proceedings of the IEEE International Workshop on Applications of Computer Vision (WACV)*, pp. 3–7, Orlando, FL (2002)
13. Dornaika, F., Ahlberg, J.: Face and facial feature tracking using deformable models. *Int. J. Image Graph.* **4**(3), 499–532 (2004)
14. Dornaika, F., Ahlberg, J.: Fast and reliable active appearance model search for 3D face tracking. *IEEE Trans. Syst. Man Cybern., Part B, Cybern.* **34**(4), 1838–1853 (2004)
15. Dornaika, F., Davoine, F.: On appearance based face and facial action tracking. *IEEE Trans. Circuits Syst. Video Technol.* **16**(9), 1107–1124 (2006)
16. Eisert, P., Girod, B.: Model-based estimation of facial expression parameters from image sequences. In: *Proceedings of the International Conference on Image Processing (ICIP)*, pp. 418–421, Santa Barbara, CA, USA (1997)
17. Ekman, P., Friesen, W.V.: *Facial action coding system*. Consulting Psychologists Press, Palo Alto (1977)
18. Fanelli, G., Fratarcangeli, M.: A non-invasive approach for driving virtual talking heads from real facial movements. In: *Proceedings of the IEEE 3DTV Conference*, Kos, Greece, May 2007
19. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **24**(6), 381–395 (1981)
20. Fratarcangeli, M., Schaerf, M., Forchheimer, R.: Facial motion cloning with radial basis functions in mpeg-4 fba. *Graph. Models* **69**(2), 106–118 (2007)
21. Gelb, A.: *Applied Optimal Estimation*. MIT Press, Cambridge (1974)
22. Gokturk, S.B., Bouguet, J.Y., Grzeszczuk, R.: A data-driven model for monocular face tracking. In: *Proceedings of the International Conference on Computer Vision (ICCV)*, pp. 701–708, Vancouver, Canada (2001)
23. Heckbert, P.: Survey of texture mapping. *IEEE Comput. Graph. Appl.* (1986)
24. Ingemars, N., Ahlberg, J.: Feature-based face tracking using extended Kalman filtering. In: *Proceedings of the Swedish Symposium on Image Analysis (SSBA)*, pp. 141–144, Linköping, Sweden (2007)
25. InterFace. European Union 5th Framework Programme project 2000–2002
26. Jebara, T., Pentland, A.: Parameterized structure from motion for 3D adaptive feedback tracking of faces. In: *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 144–150. San Juan, Puerto Rico (1997)
27. Kahler, K., Haber, J., Seidel, H.-P.: Geometry-based muscle modeling for facial animation. In: *Proceedings of Graphics Interface*, pp. 37–46 (2001)

28. Kalman, R.E.: A new approach to linear filtering and prediction problems. *J. Basic Eng.* **82D**, 25–45 (1960)
29. Kavan, L., Zara, J.: Spherical blend skinning: A real-time deformation of articulated models. In: *Proceedings I3D*, New York: ACM SIGGRAPH, pp. 9–16
30. Kirby, M., Sirovich, L.: Application of the Karhunen–Loeve procedure for the characterization of human faces. *IEEE Trans. Pattern Anal. Mach. Intell.* **12**(1), 103–108 (1990)
31. La Cascia, M., Sclaroff, S., Athitsos, V.: Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3D models. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(4), 322–336 (2000)
32. Lefevre, S., Odobez, J.-M.: Structure and appearance features for robust 3d facial actions tracking. In: *International Conference on Multimedia and Expo*, New York, NY, USA (2009)
33. Li, H., Roivanen, P., Forchheimer, R.: 3-D motion estimation in model-based facial image coding. *IEEE Trans. Pattern Anal. Mach. Intell.* **15**(6), 545–555 (1993)
34. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 674–679, April 1981
35. Magnenat-Thalmann, N., Primeau, N.E., Thalmann, D.: Abstract muscle actions procedures for human face animation. *Vis. Comput.* **3**(5), 290–297 (1988)
36. Malciu, M., Prêteux, F.: A robust model-based approach for 3d head tracking in video sequences. In: *Proceedings of the International Conference on Face and Gesture Recognition*, pp. 169–174. Grenoble, France (2000)
37. Matthews, I., Baker, S.: Active appearance models revisited. *Int. J. Comput. Vis.* **60**(2), 135–164 (2004)
38. Messer, K., Matas, J., Kittler, J., Luettin, J., Maitre, G.: XM2VTSDB: The Extended M2VTS Database. In: *Proceedings of the Second International Conference on Audio and Video-based Biometric Person Authentication (AVBPA)*, Washington, DC (1999)
39. Mohr, A., Gleicher, M.: Building efficient, accurate character skins from examples. In: *Proceedings of SIGGRAPH*, pp. 562–568, San Diego, CA, 27–31 July 2003. ACM, New York (2003)
40. Morency, L.-P., Whitehill, J., Movellan, J.R.: Generalized adaptive view-based appearance model: Integrated framework for monocular head pose estimation. In: *Proceedings of the International Conference on Face and Gesture Recognition*, pp. 1–8, Amsterdam, The Netherlands (2008)
41. Moving Picture Experts Group. ISO/IEC 14496—MPEG-4 international standard (1999). [www.cselt.it/mpeg](http://www.cselt.it/mpeg)
42. Nelder, J.A., Mead, R.: A simplex method for function minimization. *Comput. J.* **7**(4), 308–313 (1965)
43. Noh, J., Neumann, U.: Expression cloning. In: *Proceedings of SIGGRAPH*, Los Angeles, CA (2001)
44. OpenCV. <http://opencv.willowgarage.com>
45. Pandzic, I.S.: Life on the web. *Softw. Focus* **2**(2), 52–59 (2001)
46. Pandzic, I.S.: Facial motion cloning. *Graph. Models* **65**(6), 385–404 (2003)
47. Pandzic, I.S., Forchheimer, R. (eds.): *MPEG-4 Facial Animation: The Standard, Implementations, and Applications*. Wiley, Chichester (2002)
48. Parke, F.I.: Parameterized models for facial animation. *IEEE Comput. Graph. Appl.* **2**(9), 61–68 (1982)
49. Parke, F.I., Waters, K.: *Computer Facial Animation*, 2nd edn. AK Peters, Wellesley (2008)
50. Pighin, F., Szeliski, S., Salesin, S.: Resynthesizing facial animation through 3d model-based tracking. In: *Proceedings of the International Conference on Computer Vision (ICCV)*, pp. 143–150, Kerkyra, Greece (1999)
51. Roivanen, P.: Motion estimation in model-based coding of human faces. Licentiate thesis, Linköping University, Sweden (1990)
52. Rydfalk, M.: Candide, a parameterized face. Technical Report LiTH-ISY-I-866, Linköping University, Sweden (1987)

53. Sirovich, L., Kirby, M.: Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am.* **4**(3), 519–524 (1987)
54. Sonka, M., Hlavac, V., Boyle, R.: *Image Processing, Analysis, and Machine Vision*. PWS, Pacific Grove (1998)
55. Ström, J.: Model-based head tracking and coding. PhD thesis, Linköping University, Sweden (2002)
56. Ström, J.: Model-based real-time head tracking. *EURASIP J. Appl. Signal Process.* **2002**(10), 1039–1052 (2002)
57. Ström, J., Jebara, T., Basu, S., Pentland, A.: Real time tracking and modeling of faces: an EKF-based analysis by synthesis approach. In: *IEEE ICCV Workshop on Modelling People*, Kerkyra, Greece (1999)
58. Tao, H., Lopez, R., Huang, T.: Tracking of face features using probabilistic network. In: *Proceedings of the International Conference on Face and Gesture Recognition*, Nara, Japan (1998)
59. Terzopoulos, D., Waters, K.: Physically-based facial modeling, analysis and animation. *J. Vis. Comput. Animat.* **1**(4), 73–80 (1990)
60. Turk, M., Pentland, A.: Eigenfaces for recognition. *Int. J. Cogn. Neurosci.* **3**(1), 71–86 (1991)
61. Waters, K.: A muscle model for animating three-dimensional facial expressions. *Comput. Graph.* **21**(4), 17–24 (1987)
62. Welsh, B.: Model-based coding of images. PhD thesis, British Telecom Research Lab (1991)
63. Wiles, C.S., Maki, A., Matsuda, N.: Hyperpatches for 3D model acquisition. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(12), 1391–1403 (2001)
64. Xiao, J., Moriyama, T., Kanade, T., Cohn, J.: Robust full motion recovery of head for facial expression analysis. *Int. J. Imaging Syst. Technol.* **13**, 85–94 (2003)
65. Zhou, S., Chellappa, R., Moghaddam, B.: Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Trans. Image Process.* **13**(11), 1491–1506 (2004)

# Chapter 19

## Facial Expression Recognition

Yingli Tian, Takeo Kanade, and Jeffrey F. Cohn

### 19.1 Introduction

Facial expressions are the facial changes in response to a person's internal emotional states, intentions, or social communications. Facial expression analysis has been an active research topic for behavioral scientists since the work of Darwin in 1872 [21, 26, 29, 83]. Suwa et al. [90] presented an early attempt to automatically analyze facial expressions by tracking the motion of 20 identified spots on an image sequence in 1978. After that, much progress has been made to build computer systems to help us understand and use this natural form of human communication [5, 7, 8, 17, 23, 32, 43, 45, 57, 64, 77, 92, 95, 106–108, 110].

In this chapter, facial expression analysis refers to computer systems that attempt to automatically analyze and recognize facial motions and facial feature changes from visual information. Sometimes the facial expression analysis has been confused with emotion analysis in the computer vision domain. For emotion analysis, higher level knowledge is required. For example, although facial expressions can convey emotion, they can also express intention, cognitive processes, physical effort, or other intra- or interpersonal meanings. Interpretation is aided by context, body gesture, voice, individual differences, and cultural factors as well as by facial configuration and timing [11, 79, 80]. Computer facial expression analysis systems need to analyze the facial actions regardless of context, culture, gender, and so on.

---

Y. Tian (✉)

Department of Electrical Engineering, The City College of New York, New York, NY 10031,  
USA

e-mail: [ytian@ccny.cuny.edu](mailto:ytian@ccny.cuny.edu)

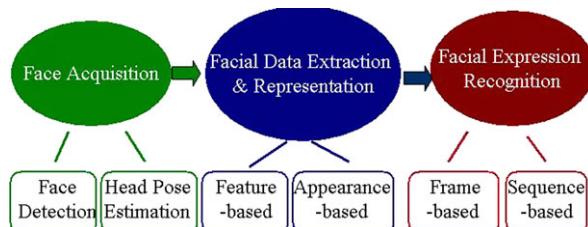
T. Kanade

Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA  
e-mail: [tk@cs.cmu.edu](mailto:tk@cs.cmu.edu)

J.F. Cohn

Department of Psychology, University of Pittsburgh, Pittsburgh, PA 15260, USA  
e-mail: [jeffcohn@pitt.edu](mailto:jeffcohn@pitt.edu)

**Fig. 19.1** Basic structure of facial expression analysis systems



The accomplishments in the related areas such as psychological studies, human movement analysis, face detection, face tracking, and recognition make the automatic facial expression analysis possible. Automatic facial expression analysis can be applied in many areas such as emotion and paralinguistic communication, clinical psychology, psychiatry, neurology, pain assessment, lie detection, intelligent environments, and multimodal human computer interface (HCI).

## 19.2 Principles of Facial Expression Analysis

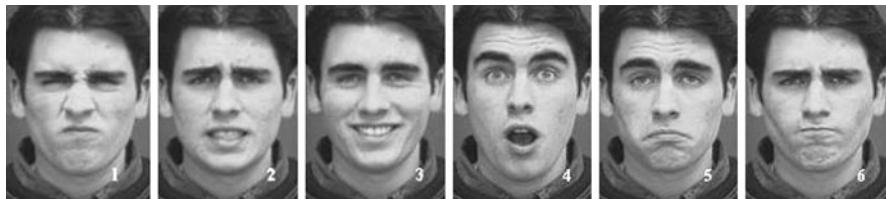
### 19.2.1 Basic Structure of Facial Expression Analysis Systems

Facial expression analysis includes both measurement of facial motion and recognition of expression. The general approach to automatic facial expression analysis (AFAEA) consists of three steps (Fig. 19.1): face acquisition, facial data extraction and representation, and facial expression recognition.

Face acquisition is a processing stage to automatically find the face region for the input images or sequences. It can be a detector to detect face for each frame or just detect face in the first frame and then track the face in the remainder of the video sequence. To handle large head motion, the head finder, head tracking, and pose estimation can be applied to a facial expression analysis system.

After the face is located, the next step is to extract and represent the facial changes caused by facial expressions. In facial feature extraction for expression analysis, there are mainly two types of approaches: geometric feature-based methods and appearance-based methods. The geometric facial features present the shape and locations of facial components (including mouth, eyes, brows, nose, etc.). The facial components or facial feature points are extracted to form a feature vector that represents the face geometry. With appearance-based methods, image filters, such as Gabor wavelets, are applied to either the whole-face or specific regions in a face image to extract a feature vector. Depending on the different facial feature extraction methods, the effects of in-plane head rotation and different scales of the faces can be eliminated by face normalization before the feature extraction or by feature representation before the step of expression recognition.

Facial expression recognition is the last stage of AFEA systems. The facial changes can be identified as facial action units or prototypic emotional expressions (see Sect. 19.3.1 for definitions). Depending on whether the temporal information is



**Fig. 19.2** Emotion-specified facial expression (posed images from database [49]). 1, disgust; 2, fear; 3, joy; 4, surprise; 5, sadness; 6, anger

used, in this chapter we classify a recognition approach as frame-based or sequence-based.

### 19.2.2 *Organization of the Chapter*

This chapter introduces recent advances in facial expression analysis. The first part discusses general structure of AFEA systems. The second part describes the problem space for facial expression analysis. This space includes multiple dimensions: level of description, individual differences in subjects, transitions among expressions, intensity of facial expression, deliberate versus spontaneous expression, head orientation and scene complexity, image acquisition and resolution, reliability of ground truth, databases, and the relation to other facial behaviors or nonfacial behaviors. We note that most work to date has been confined to a relatively restricted region of this space. The last part of this chapter is devoted to a description of more specific approaches and the techniques used in recent advances. They include the techniques for face acquisition, facial data extraction and representation, facial expression recognition, and multimodal expression analysis. The chapter concludes with a discussion assessing the current status, future possibilities, and open questions about automatic facial expression analysis.

## 19.3 Problem Space for Facial Expression Analysis

### 19.3.1 *Level of Description*

With few exceptions [17, 23, 34, 95], most AFEA systems attempt to recognize a small set of prototypic emotional expressions as shown in Fig. 19.2 (i.e., disgust, fear, joy, surprise, sadness, anger). This practice may follow from the work of Darwin [21] and more recently Ekman and Friesen [27, 28] and Izard et al. [48] who proposed that emotion-specified expressions have corresponding prototypic facial expressions. In everyday life, however, such prototypic expressions occur relatively infrequently. Instead, emotion more often is communicated by subtle changes in one

**Table 19.1** FACS action units (AU). AUs with “\*\*” indicate that the criteria have changed for this AU, that is, AU 25, 26, and 27 are now coded according to criteria of intensity (25A-E), and AU 41, 42, and 43 are now coded according to criteria of intensity

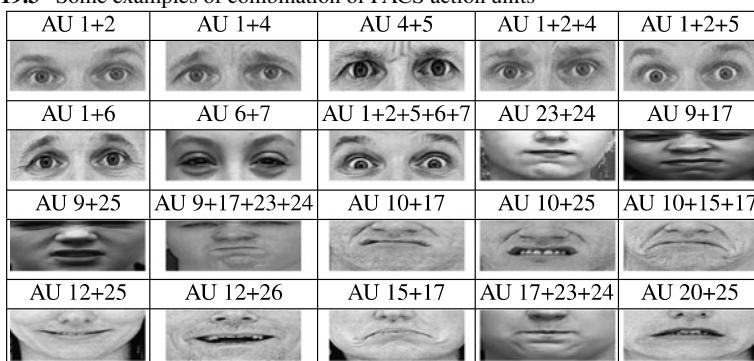
| Upper Face Action Units |                     |                     |                   |               |               |
|-------------------------|---------------------|---------------------|-------------------|---------------|---------------|
| AU 1                    | AU 2                | AU 4                | AU 5              | AU 6          | AU 7          |
|                         |                     |                     |                   |               |               |
| Inner Brow Raiser       | Outer Brow Raiser   | Brow Lowerer        | Upper Lid Raiser  | Cheek Raiser  | Lid Tightener |
| *AU 41                  | *AU 42              | *AU 43              | AU 44             | AU 45         | AU 46         |
|                         |                     |                     |                   |               |               |
| Lid Droop               | Slit                | Eyes Closed         | Squint            | Blink         | Wink          |
| Lower Face Action Units |                     |                     |                   |               |               |
| AU 9                    | AU 10               | AU 11               | AU 12             | AU 13         | AU 14         |
|                         |                     |                     |                   |               |               |
| Nose Wrinkler           | Upper Lip Raiser    | Nasolabial Deepener | Lip Corner Puller | Cheek Puffer  | Dimpler       |
| AU 15                   | AU 16               | AU 17               | AU 18             | AU 20         | AU 22         |
|                         |                     |                     |                   |               |               |
| Lip Corner Depressor    | Lower Lip Depressor | Chin Raiser         | Lip Puckerer      | Lip Stretcher | Lip Funneler  |
| AU 23                   | AU 24               | *AU 25              | *AU 26            | *AU 27        | AU 28         |
|                         |                     |                     |                   |               |               |
| Lip Tightener           | Lip Pressor         | Lips Part           | Jaw Drop          | Mouth Stretch | Lip Suck      |

or a few discrete facial features, such as tightening of the lips in anger or obliquely lowering the lip corners in sadness [12]. Change in isolated features, especially in the area of the eyebrows or eyelids, is typical of paralinguistic displays; for instance, raising the brows signals greeting [25]. To capture such subtlety of human emotion and paralinguistic communication, automated recognition of fine-grained changes in facial expression is needed. The facial action coding system (FACS: [29]) is a human-observer-based system designed to detect subtle changes in facial features. Viewing videotaped facial behavior in slow motion, trained observers can manually FACS code all possible facial displays, which are referred to as action units and may occur individually or in combinations.

FACS consists of 44 action units. Thirty are anatomically related to contraction of a specific set of facial muscles (Table 19.1) [22]. The anatomic basis of the remaining 14 is unspecified (Table 19.2). These 14 are referred to in FACS as miscellaneous actions. Many action units may be coded as symmetrical or asymmetrical. For action units that vary in intensity, a 5-point ordinal scale is used to measure the degree of muscle contraction. Table 19.3 shows some examples of combinations of FACS action units.

**Table 19.2** Miscellaneous actions

| AU | Description      |
|----|------------------|
| 8  | Lips toward      |
| 19 | Tongue show      |
| 21 | Neck tighten     |
| 29 | Jaw thrust       |
| 30 | Jaw sideways     |
| 31 | Jaw clench       |
| 32 | Bite lip         |
| 33 | Blow             |
| 34 | Puff             |
| 35 | Cheek suck       |
| 36 | Tongue bulge     |
| 37 | Lip wipe         |
| 38 | Nostril dilate   |
| 39 | Nostril compress |

**Table 19.3** Some examples of combination of FACS action units

Although Ekman and Friesen proposed that specific combinations of FACS action units represent prototypic expressions of emotion, emotion-specified expressions are not part of FACS; they are coded in separate systems, such as the emotional facial action system (EMFACS) [41]. FACS itself is purely descriptive and includes no inferential labels. By converting FACS codes to EMFACS or similar systems, face images may be coded for emotion-specified expressions (e.g., joy or anger) as well as for more molar categories of positive or negative emotion [65].

### 19.3.2 Individual Differences in Subjects

Face shape, texture, color, and facial and scalp hair vary with sex, ethnic background, and age [33, 119]. Infants, for instance, have smoother, less textured skin and often lack facial hair in the brows or scalp. The eye opening and contrast between iris and sclera differ markedly between Asians and Northern Europeans, which may affect the robustness of eye tracking and facial feature analysis more generally. Beards, eyeglasses, or jewelry may obscure facial features. Such individual differences in appearance may have important consequences for face analysis. Few attempts to study their influence exist. An exception was a study by Zlochower et al. [119], who found that algorithms for optical flow and high-gradient component detection that had been optimized for young adults performed less well when used in infants. The reduced texture of infants' skin, their increased fatty tissue, juvenile facial conformation, and lack of transient furrows may all have contributed to the differences observed in face analysis between infants and adults.

In addition to individual differences in appearance, there are individual differences in expressiveness, which refers to the degree of facial plasticity, morphology, frequency of intense expression, and overall rate of expression. Individual differences in these characteristics are well established and are an important aspect of individual identity [61] (these individual differences in expressiveness and in biases for particular facial actions are sufficiently strong that they may be used as a biometric to augment the accuracy of face recognition algorithms [19]). An extreme example of variability in expressiveness occurs in individuals who have incurred damage either to the facial nerve or central nervous system [75, 99]. To develop algorithms that are robust to individual differences in facial features and behavior, it is essential to include a large sample of varying ethnic background, age, and sex, which includes people who have facial hair and wear jewelry or eyeglasses and both normal and clinically impaired individuals.

### 19.3.3 Transitions Among Expressions

A simplifying assumption in facial expression analysis is that expressions are singular and begin and end with a neutral position. In reality, facial expression is more complex, especially at the level of action units. Action units may occur in combinations or show serial dependence. Transitions from action units or combination of actions to another may involve no intervening neutral state. Parsing the stream of behavior is an essential requirement of a robust facial analysis system, and training data are needed that include dynamic combinations of action units, which may be either additive or nonadditive.

As shown in Table 19.3, an example of an additive combination is smiling (AU 12) with mouth open, which would be coded as AU 12 + 25, AU 12 + 26, or AU 12 + 27 depending on the degree of lip parting and whether and how far the mandible was lowered. In the case of AU 12 + 27, for instance, the facial analysis

system would need to detect transitions among all three levels of mouth opening while continuing to recognize AU 12, which may be simultaneously changing in intensity.

Nonadditive combinations represent further complexity. Following usage in speech science, we refer to these interactions as co-articulation effects. An example is the combination AU 12 + 15, which often occurs during embarrassment. Although AU 12 raises the cheeks and lip corners, its action on the lip corners is modified by the downward action of AU 15. The resulting appearance change is highly dependent on timing. The downward action of the lip corners may occur simultaneously or sequentially. The latter appears to be more common [85]. To be comprehensive, a database should include individual action units and both additive and nonadditive combinations, especially those that involve co-articulation effects. A classifier trained only on single action units may perform poorly for combinations in which co-articulation effects occur.

#### ***19.3.4 Intensity of Facial Expression***

Facial actions can vary in intensity. Manual FACS coding, for instance, uses a 3- or more recently a 5-point intensity scale to describe intensity variation of action units (for psychometric data, see Sayette et al. [82]). Some related action units, moreover, function as sets to represent intensity variation. In the eye region, action units 41, 42, and 43 or 45 can represent intensity variation from slightly drooped to closed eyes. Several computer vision researchers proposed methods to represent intensity variation automatically. Essa and Pentland [32] represented intensity variation in smiling using optical flow. Kimura and Yachida [50] and Lien et al. [56] quantified intensity variation in emotion-specified expression and in action units, respectively. These authors did not, however, attempt the more challenging step of discriminating intensity variation within types of facial actions. Instead, they used intensity measures for the more limited purpose of discriminating between different types of facial actions. Tian et al. [94] compared manual and automatic coding of intensity variation. Using Gabor features and an artificial neural network, they discriminated intensity variation in eye closure as reliably as human coders did. Recently, Bartlett and colleagues [5] investigated action unit intensity by analyzing facial expression dynamics. They performed a correlation analysis to explicitly measure the relationship between the output margin of the learned classifiers and expression intensity. Yang et al. [111] converted the problem of intensity estimation to a ranking problem, which is modeled by the RankBoost. They employed the output ranking score for intensity estimation. These findings suggest that it is feasible to automatically recognize intensity variation within types of facial actions. Regardless of whether investigators attempt to discriminate intensity variation within facial actions, it is important that the range of variation be described adequately. Methods that work for intense expressions may generalize poorly to ones of low intensity.

### 19.3.5 Deliberate Versus Spontaneous Expression

Most face expression data have been collected by asking subjects to perform a series of expressions. These directed facial action tasks may differ in appearance and timing from spontaneously occurring behavior [30]. Deliberate and spontaneous facial behavior are mediated by separate motor pathways, the pyramidal and extrapyramidal motor tracks, respectively [75]. As a consequence, fine-motor control of deliberate facial actions is often inferior and less symmetrical than what occurs spontaneously. Many people, for instance, are able to raise their outer brows spontaneously while leaving their inner brows at rest; few can perform this action voluntarily. Spontaneous depression of the lip corners (AU 15) and raising and narrowing the inner corners of the brow (AU 1 + 4) are common signs of sadness. Without training, few people can perform these actions deliberately, which incidentally is an aid to lie detection [30]. Differences in the temporal organization of spontaneous and deliberate facial actions are particularly important in that many pattern recognition approaches, such as hidden Markov modeling, are highly dependent on the timing of the appearance change. Unless a database includes both deliberate and spontaneous facial actions, it will likely prove inadequate for developing face expression methods that are robust to these differences.

### 19.3.6 Head Orientation and Scene Complexity

Face orientation relative to the camera, the presence and actions of other people, and background conditions may influence face analysis. In the face recognition literature, face orientation has received deliberate attention. The FERET database [76], for instance, includes both frontal and oblique views, and several specialized databases have been collected to try to develop methods of face recognition that are invariant to moderate change in face orientation [100]. In the face expression literature, use of multiple perspectives is rare; and relatively less attention has been focused on the problem of pose invariance. Most researchers assume that face orientation is limited to in-plane variation [3] or that out-of-plane rotation is small [57, 68, 77, 95]. In reality, large out-of-plane rotation in head position is common and often accompanies change in expression. Kraut and Johnson [54] found that smiling typically occurs while turning toward another person. Camras et al. [10] showed that infant surprise expressions often occur as the infant pitches her head back. To develop pose invariant methods of face expression analysis, image data are needed in which facial expression changes in combination with significant non-planar change in pose. Some efforts have been made to handle large out-of-plane rotation in head position [5, 20, 97, 104].

Scene complexity, such as background and the presence of other people, potentially influences accuracy of face detection, feature tracking, and expression recognition. Most databases use image data in which the background is neutral or has a

consistent pattern and only a single person is present in the scene. In natural environments, multiple people interacting with each other are likely, and their effects need to be understood. Unless this variation is represented in training data, it will be difficult to develop and test algorithms that are robust to such variation.

### ***19.3.7 Image Acquisition and Resolution***

The image acquisition procedure includes several issues, such as the properties and number of video cameras and digitizer, the size of the face image relative to total image dimensions, and the ambient lighting. All of these factors may influence facial expression analysis. Images acquired in low light or at coarse resolution can provide less information about facial features. Similarly, when the face image size is small relative to the total image size, less information is available. NTSC cameras record images at 30 frames per second, The implications of down-sampling from this rate are unknown. Many algorithms for optical flow assume that pixel displacement between adjacent frames is small. Unless they are tested at a range of sampling rates, the robustness to sampling rate and resolution cannot be assessed.

Within an image sequence, changes in head position relative to the light source and variation in ambient lighting have potentially significant effects on face expression analysis. A light source above the subject's head causes shadows to fall below the brows, which can obscure the eyes, especially for subjects with pronounced bone structure or hair. Methods that work well in studio lighting may perform poorly in more natural lighting (e.g., through an exterior window) when the angle of lighting changes across an image sequence. Most investigators use single-camera setups, which is problematic when a frontal orientation is not required. With image data from a single camera, out-of-plane rotation may be difficult to standardize. For large out-of-plane rotation, multiple cameras may be required. Multiple camera setups can support three dimensional (3D) modeling and in some cases ground truth with which to assess the accuracy of image alignment. Pantic and Rothkrantz [70] were the first to use two cameras mounted on a headphone-like device; one camera is placed in front of the face and the other on the right side of the face. The cameras are moving together with the head to eliminate the scale and orientation variance of the acquired face images.

Image resolution is another concern. Professional grade PAL cameras, for instance, provide very high resolution images. By contrast, security cameras provide images that are seriously degraded. Although postprocessing may improve image resolution, the degree of potential improvement is likely limited. Also the effects of post processing for expression recognition are not known. Table 19.4 shows a face at different resolutions. Most automated face processing tasks should be possible for a  $69 \times 93$  pixel image. At  $48 \times 64$  pixels the facial features such as the corners of the eyes and the mouth become hard to detect. The facial expressions may be recognized at  $48 \times 64$  and are not recognized at  $24 \times 32$  pixels. Algorithms that work well at optimal resolutions of full face frontal images and studio lighting

**Table 19.4** A face at different resolutions. All images are enlarged to the same size. At  $48 \times 64$  pixels the facial features such as the corners of the eyes and the mouth become hard to detect. Facial expressions are not recognized at  $24 \times 32$  pixels [97]

| Face Process | 96 x 128 | 69 x 93 | 48 x 64 | 24 x 32 |
|--------------|----------|---------|---------|---------|
| Detect?      | Yes      | Yes     | Yes     | Yes     |
| Pose?        | Yes      | Yes     | Yes     | Yes     |
| Recognize?   | Yes      | Yes     | Yes     | Maybe   |
| Features?    | Yes      | Yes     | Maybe   | No      |
| Expressions? | Yes      | Yes     | Maybe   | No      |

can be expected to perform poorly when recording conditions are degraded or images are compressed. Without knowing the boundary conditions of face expression algorithms, comparative performance is difficult to assess. Algorithms that appear superior within one set of boundary conditions may perform more poorly across the range of potential applications. Appropriate data with which these factors can be tested are needed.

### 19.3.8 Reliability of Ground Truth

When training a system to recognize facial expression, the investigator assumes that training and test data are accurately labeled. This assumption may or may not be accurate. Asking subjects to perform a given action is no guarantee that they will. To ensure internal validity, expression data must be manually coded, and the reliability of the coding verified. Interobserver reliability can be improved by providing rigorous training to observers and monitoring their performance. FACS coders must pass a standardized test, which ensures (initially) uniform coding among international laboratories. Monitoring is best achieved by having observers independently code a portion of the same data. As a general rule, 15% to 20% of data should be comparison-coded. To guard against drift in coding criteria [62], restandardization is important. When assessing reliability, coefficient kappa [36] is preferable to raw percentage of agreement, which may be inflated by the marginal frequencies of codes. Kappa quantifies interobserver agreement after correcting for the level of agreement expected by chance.

### 19.3.9 Databases

Because most investigators have used relatively limited data sets, the generalizability of different approaches to facial expression analysis remains unknown. In most data sets, only relatively global facial expressions (e.g., joy or anger) have been considered, subjects have been few in number and homogeneous with respect to age

and ethnic background, and recording conditions have been optimized. Approaches to facial expression analysis that have been developed in this way may transfer poorly to applications in which expressions, subjects, contexts, or image properties are more variable. In the absence of comparative tests on common data, the relative strengths and weaknesses of different approaches are difficult to determine. In the areas of face and speech recognition, comparative tests have proven valuable [76], and similar benefits would likely accrue in the study of facial expression analysis. A large, representative test-bed is needed with which to evaluate different approaches. We list several databases for facial expression analysis in Sect. 19.4.5.

### 19.3.10 Relation to Other Facial Behavior or Nonfacial Behavior

Facial expression is one of several channels of nonverbal communication. Contraction of the muscle *zygomaticus major* (AU 12), for instance, is often associated with positive or happy vocalizations, and smiling tends to increase vocal fundamental frequency [16]. Also facial expressions often occur during conversations. Both expressions and conversations can cause facial changes. Few research groups, however, have attempted to integrate gesture recognition broadly defined across multiple channels of communication [44, 45]. An important question is whether there are advantages to early rather than late integration [38]. Databases containing multimodal expressive behavior afford the opportunity for integrated approaches to analysis of facial expression, prosody, gesture, and kinetic expression.

### 19.3.11 Summary and Ideal Facial Expression Analysis Systems

The problem space for facial expression includes multiple dimensions. An ideal facial expression analysis system has to address all these dimensions, and it outputs accurate recognition results. In addition, the ideal facial expression analysis system must perform automatically and in real-time for all stages (Fig. 19.1). So far, several systems can recognize expressions in real time [53, 68, 97]. We summarize the properties of an ideal facial expression analysis system in Table 19.5.

## 19.4 Recent Advances

For automatic facial expression analysis, Suwa et al. [90] presented an early attempt in 1978 to analyze facial expressions by tracking the motion of 20 identified spots on an image sequence. Considerable progress had been made since 1990 in related technologies such as image analysis and pattern recognition that make AFEA possible. Samal and Iyengar [81] surveyed the early work (before 1990) about automatic recognition and analysis of human face and facial expression. Two survey papers

**Table 19.5** Properties of an ideal facial expression analysis system

| Robustness        |                                                           |
|-------------------|-----------------------------------------------------------|
| Rb1               | Deal with subjects of different age, gender, ethnicity    |
| Rb2               | Handle lighting changes                                   |
| Rb3               | Handle large head motion                                  |
| Rb4               | Handle occlusion                                          |
| Rb5               | Handle different image resolution                         |
| Rb6               | Recognize all possible expressions                        |
| Rb7               | Recognize expressions with different intensity            |
| Rb8               | Recognize asymmetrical expressions                        |
| Rb9               | Recognize spontaneous expressions                         |
| Automatic process |                                                           |
| Am1               | Automatic face acquisition                                |
| Am2               | Automatic facial feature extraction                       |
| Am3               | Automatic expression recognition                          |
| Real-time process |                                                           |
| Rt1               | Real-time face acquisition                                |
| Rt2               | Real-time facial feature extraction                       |
| Rt3               | Real-time expression recognition                          |
| Autonomic Process |                                                           |
| An1               | Output recognition with confidence                        |
| An2               | Adaptive to different level outputs based on input images |

summarized the work (before year 1999) of facial expression analysis [35, 69]. Recently, Zeng et al. [114] surveyed the work (before year 2007) for affect recognition methods including audio, visual and spontaneous expressions. In this chapter, instead of giving a comprehensive survey of facial expression analysis literature, we explore the recent advances in facial expression analysis based on four problems: (1) face acquisition, (2) facial feature extraction and representation, (3) facial expression recognition, and (4) multimodal expression analysis. In addition, we list the public available databases for expression analysis.

Many efforts have been made for facial expression analysis [4, 5, 8, 13, 15, 18, 20, 23, 32, 34, 35, 37, 45, 58–60, 67, 69, 70, 87, 95, 96, 102, 104, 107, 110–115, 117]. Because most of the work are summarized in the survey papers [35, 69, 114], here we focus on the recent research in automatic facial expression analysis which tends to follow these directions:

- Build more robust systems for face acquisition, facial data extraction and representation, and facial expression recognition to handle head motion (in-plane and out-of-plane), occlusion, lighting changes, and lower intensity of expressions
- Employ more facial features to recognize more expressions and to achieve a higher recognition rate
- Recognize facial action units and their combinations rather than emotion-specified expressions
- Recognize action units as they occur spontaneously
- Develop fully automatic and real-time AFEA systems
- Analyze emotion portrayals by combining multimodal features such as facial expression, vocal expression, gestures, and body movements

### **19.4.1 Face Acquisition**

With few exceptions, most AFEA research attempts to recognize facial expressions only from frontal-view or near frontal-view faces [51, 70]. Kleck and Mendolia [51] first studied the decoding of profile versus full-face expressions of affect by using three perspectives (a frontal face, a 90° right profile, and a 90° left profile). Forty-eight decoders viewed the expressions from 64 subjects in one of the three facial perspectives. They found that the frontal faces elicited higher intensity ratings than profile views for negative expressions. The opposite was found for positive expressions. Pantic and Rothkrantz [70] used dual-view facial images (a full-face and a 90° right profile) which are acquired by two cameras mounted on the user's head. They did not compare the recognition results by using only the frontal view and the profile. So far, it is unclear how many expressions can be recognized by side-view or profile faces. Because the frontal-view face is not always available in real environments, the face acquisition methods should detect both frontal and nonfrontal view faces in an arbitrary scene.

To handle out-of-plane head motion, face can be obtained by face detection, 2D or 3D face tracking, or head pose detection. Nonfrontal view faces are warped or normalized to frontal view for expression analysis.

#### **19.4.1.1 Face Detection**

Many face detection methods have been developed to detect faces in an arbitrary scene [47, 55, 72, 78, 86, 89, 101]. Most of them can detect only frontal and near-frontal views of faces. Heisele et al. [47] developed a component-based, trainable system for detecting frontal and near-frontal views of faces in still gray images. Rowley et al. [78] developed a neural network based system to detect frontal-view face. Viola and Jones [101] developed a robust real-time face detector based on a set of rectangle features.

To handle out-of-plane head motion, some researchers developed face detectors to detect face from different views [55, 72, 86]. Pentland et al. [72] detected faces by

using the view-based and modular eigenspace method. Their method runs real-time and can handle varying head positions. Schneiderman and Kanade [86] proposed a statistical method for 3D object detection that can reliably detect human faces with out-of-plane rotation. They represent the statistics of both *object* appearance and *nonobject* appearance using a product of histograms. Each histogram represents the joint statistics of a subset of wavelet coefficients and their position on the object. Li et al. [55] developed an AdaBoost-like approach to detect faces with multiple views. A detail survey about face detection can be found in paper [109].

Some facial expression analysis systems use the face detector which developed by Viola et al. [101] to detect face for each frame [37]. Some systems [18, 67, 94–96, 104] assume that the first frame of the sequence is frontal and expressionless. They detect faces only in the first frame and then perform feature tracking or head tracking for the remaining frames of the sequence.

#### 19.4.1.2 Head Pose Estimation

In a real environment, out-of-plane head motion is common for facial expression analysis. To handle the out-of-plane head motion, head pose estimation can be employed. The methods for estimating head pose can be classified as 3D model-based methods [1, 91, 98, 104] and 2D image-based methods [9, 97, 103, 118].

**3D Model-Based Method** Many systems employ a 3D model based method to estimate head pose [4, 5, 15, 18, 20, 67, 102, 104]. Bartlett et al. [4, 5] used a canonical wire-mesh face model to estimate face geometry and 3D pose from hand-labeled feature points. In papers [15, 102], the authors used an explicit 3D wireframe face model to track geometric facial features defined on the model [91]. The 3D model is fitted to the first frame of the sequence by manually selecting landmark facial features such as corners of the eyes and mouth. The generic face model, which consists of 16 surface patches, is warped to fit the selected facial features. To estimate the head motion and deformations of facial features, a two-step process is used. The 2D image motion is tracked using template matching between frames at different resolutions. From the 2D motions of many points on the face model, the 3D head motion then is estimated by solving an overdetermined system of equations of the projective motions in the least-squares sense [15].

In paper [104], a cylindrical head model is used to automatically estimate the 6 degrees of freedom (dof) of head motion in realtime. An active appearance model (AAM) method is used to automatically map the cylindrical head model to the face region, which is detected by face detection [78], as the initial appearance template. For any given frame, the template is the head image in the previous frame that is projected onto the cylindrical model. Then the template is registered with the head appearance in the given frame to recover the full motion of the head. They first use the iteratively reweighted least squares technique [6] to deal with nonrigid motion and occlusion. Second, they update the template dynamically in order to deal with gradual changes in lighting and self-occlusion. This enables the system to work well



**Fig. 19.3** Example of 3D head tracking, including re-registration after losing the head

even when most of the face is occluded. Because head poses are recovered using templates that are constantly updated and the pose estimated for the current frame is used in estimating the pose in the next frame, errors would accumulate unless otherwise prevented. To solve this problem, the system automatically selects and stores one or more frames and associated head poses from the tracked images in the sequence (usually including the initial frame and pose) as references. Whenever the difference between the estimated head pose and that of a reference frame is less than a preset threshold, the system rectifies the current pose estimate by re-registering this frame with the reference. The reregistration prevents errors from accumulating and enables the system to recover head pose when the head reappears after occlusion, such as when the head moves momentarily out of the camera's view. On-line tests suggest that the system could work robustly for an indefinite period of time. It was also quantitatively evaluated in image sequences that include maximum pitch and yaw as large as 40 and 75 degrees, respectively. The precision of recovered motion was evaluated with respect to the ground truth obtained by a precise position and orientation measurement device with markers attached to the head and found to be highly consistent (e.g., for maximum yaw of 75 degrees, absolute error averaged 3.86 degrees). An example of the 3D head tracking is shown in Fig. 19.3 including reregistration after losing the head. More details can be found in paper [104].

**2D Image-Based Method** To handle the full range of head motion for expression analysis, Tian et al. [97] detected the head instead of the face. The head detection uses the smoothed silhouette of the foreground object as segmented using background subtraction and computing the *negative curvature minima* (NCM) points of the silhouette. Other head detection techniques that use silhouettes can be found elsewhere [42, 46].

**Table 19.6** Definitions and examples of the three head pose classes: frontal or near frontal view, side view or profile, and others, such as back of the head or occluded faces. The expression analysis process is applied to only the frontal and near-frontal view faces [9, 97]

| Poses       | Frontal or near frontal                                                           | Side view or profile                  | Others                    |
|-------------|-----------------------------------------------------------------------------------|---------------------------------------|---------------------------|
| Definitions | Both eyes and lip corners are visible                                             | One eye or one lip corner is occluded | No enough facial features |
| Examples    |  |                                       |                           |

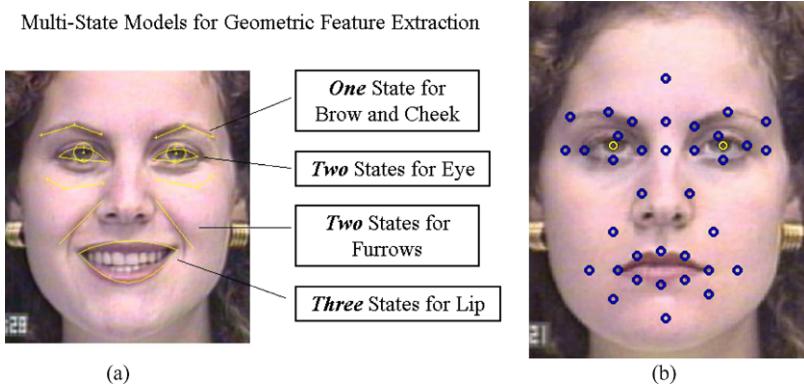
After the head is located, the head image is converted to gray-scale, histogram-equalized, and resized to the estimated resolution. Then a three-layer neural network (NN) is employed to estimate the head pose. The inputs to the network are the processed head image. The outputs are the three head poses: (1) frontal or near frontal view, (2) side view or profile, (3) others, such as back of the head or occluded face (Table 19.6). In the frontal or near frontal view, both eyes and lip corners are visible. In the side view or profile, at least one eye or one corner of the mouth becomes self-occluded because of the head. The expression analysis process is applied only to the frontal and near-frontal view faces. Their system performs well even with very low resolution of face images.

#### 19.4.2 Facial Feature Extraction and Representation

After the face is obtained, the next step is to extract facial features. Two types of features can be extracted: geometric features and appearance features. Geometric features present the shape and locations of facial components (including mouth, eyes, brows, and nose). The facial components or facial feature points are extracted to form a feature vector that represents the face geometry. The appearance features present the appearance (skin texture) changes of the face, such as wrinkles and furrows. The appearance features can be extracted on either the whole-face or specific regions in a face image.

To recognize facial expressions, an AEFA system can use geometric features only [15, 20, 70], appearance features only [5, 37, 59], or hybrid features (both geometric and appearance features) [23, 95, 96, 102]. The research shows that using hybrid features can achieve better results for some expressions.

To remove the effects of variation in face scale, motion, lighting, and other factors, one can first align and normalize the face to a standard face (2D or 3D) manually or automatically [23, 37, 57, 102], and then obtain normalized feature measurements by using a reference image (neutral face) [95].



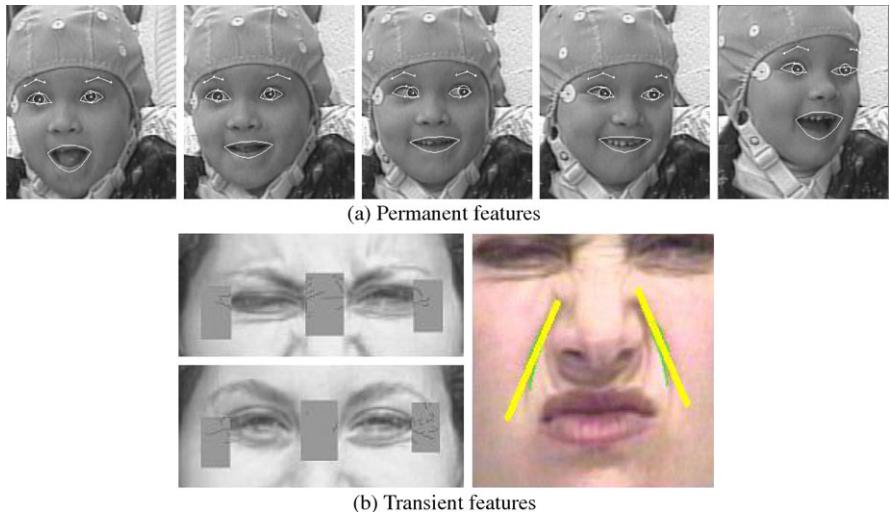
**Fig. 19.4** Facial feature extraction for expression analysis [95]. **a** Multistate models for geometric feature extraction. **b** Locations for calculating appearance features

#### 19.4.2.1 Geometric Feature Extraction

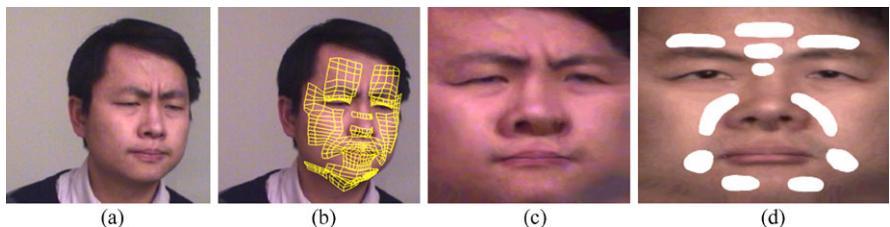
As shown in Fig. 19.4, in order to detect and track changes of facial components in near frontal face images, Tian et al. develop multi-state models to extract the geometric facial features. A three-state lip model describes the lip state: open, closed, tightly closed. A two-state model (open or closed) is used for each of the eyes. Each brow and cheek has a one-state model. Some appearance features, such as *nasolabial furrows* and *crows-feet wrinkles* (Fig. 19.5b), are represented explicitly by using two states: present and absent. Given an image sequence, the region of the face and approximate location of individual face features are detected automatically in the initial frame [78]. The contours of the face features and components then are adjusted manually in the initial frame. After the initialization, all face feature changes are automatically detected and tracked in the image sequence. The system groups 15 parameters for the upper face and 9 parameters for the lower face, which describe shape, motion, and state of face components and furrows. To remove the effects of variation in planar head motion and scale between image sequences in face size, all parameters are computed as ratios of their current values to that in the reference frame. Details of geometric feature extraction and representation can be found in paper [95].

Automatic active appearance model (AAM) mapping can be employed to reduce the manual preprocessing of the geometric feature initialization [66, 105]. Xiao et al. [104] performed the 3D head tracking to handle large out-of-plane head motion (Sect. 19.4.1) and track nonrigid features. Once the head pose is recovered, the face region is stabilized by transforming the image to a common orientation for expression recognition [18, 67].

The systems in [15, 102] use an explicit 3D wireframe face model to track geometric facial features defined on the model [91]. The 3D model is fitted to the first frame of the sequence by manually selecting landmark facial features such as corners of the eyes and mouth. The generic face model, which consists of 16 surface



**Fig. 19.5** Example results of feature extraction [95]. **a** Permanent feature extraction (eyes, brows, and mouth). **b** Transient feature extraction (crows-feet wrinkles, wrinkles at nasal root, and nasolabial furrows)



**Fig. 19.6** Example of feature extraction [102]. **a** Input video frame. **b** Snapshot of the geometric tracking system. **c** Extracted texture map. **d** Selected facial regions for appearance feature extraction [102]

patches, is warped to fit the selected facial features. Figure 19.6b shows an example of the geometric feature extraction of paper [102].

#### 19.4.2.2 Appearance Feature Extraction

Gabor wavelets [22] are widely used to extract the facial appearance changes as a set of multiscale and multiorientation coefficients. The Gabor filter may be applied to specific locations on a face [59, 94, 96, 116] or to the whole face image [4, 23, 37]. Zhang et al. [116] was the first to compare two type of features to recognize expressions, the geometric positions of 34 fiducial points on a face and 612 Gabor wavelet coefficients extracted from the face image at these 34 fiducial points. The recognition rates for six emotion-specified expressions (e.g., joy and anger) were

significantly higher for Gabor wavelet coefficients. Donato et al. [23] compared several techniques for recognizing six single upper face AUs and six lower face AUs. These techniques include optical flow, principal component analysis, independent component analysis, local feature analysis, and Gabor wavelet representation. The best performances were obtained using a Gabor wavelet representation and independent component analysis. All of these systems [23, 116] used a manual step to align each input image with a standard face image using the center of the eyes and mouth.

Tian et al. [96] studied geometric features and Gabor coefficients to recognize single AU and AU combinations. In their system, they used 480 Gabor coefficients in the upper face for 20 locations and 432 Gabor coefficients in the lower face for 18 locations (Fig. 19.4). They found that Gabor wavelets work well for single AU recognition for homogeneous subjects without head motion. However, for recognition of AU combinations when image sequences include nonhomogeneous subjects with small head motions, the recognition results are relatively poor if we use only Gabor appearance features. Several factors may account for the difference. First, the previous studies used homogeneous subjects. For instance, Zhang et al. [116] included only Japanese and Donato et al. [23] included only Euro-Americans. Tian et al. use Cohn–Kanade database which contains diverse subjects of European, African, and Asian ancestry. Second, the previous studies recognized emotion-specified expressions or only single AUs. Tian et al. tested the Gabor-wavelet-based method on both single AUs and AU combinations, including nonadditive combinations in which the occurrence of one AU modifies another. Third, the previous studies manually aligned and cropped face images. System [96] omitted this pre-processing step. In summary, using Gabor wavelets alone, recognition is adequate only for AU6, AU43, and AU0. Using geometric features alone, recognition is consistently good and shows high AU recognition rates with the exception of AU7. Combining both Gabor wavelet coefficients and geometric features, the recognition performance increased for all AUs.

In system [4], 3D pose and face geometry is estimated from hand-labeled feature points by using a canonical wire-mesh face model [73]. Once the 3D pose is estimated, faces are rotated to the frontal view and warped to a canonical face geometry. Then, the face images are automatically scaled and cropped to a standard face with a fixed distance between the two eyes. Difference images are obtained by subtracting a neutral expression face. They employed a family of Gabor wavelets at five spatial frequencies and eight orientations to a different image. Instead of specific locations on a face, they apply the Gabor filter to the whole face image. To provide robustness to lighting conditions and to image shifts they employed a representation in which the outputs of two Gabor filters in quadrature are squared and then summed. This representation is known as Gabor energy filters and it models complex cells of the primary visual cortex. Recently, Bartlett and her colleagues extend the system by using fully automatic face and eye detection. For facial expression analysis, they continue employ Gabor wavelets as appearance features [5].

Wen and Huang [102] use the ratio-image based method to extract appearance features, which is independent of a person's face albedo. To limit the effects of the

noise in tracking and individual variation, they extracted the appearance features in facial regions instead of points, and then used the weighted average as the final feature for each region. Eleven regions were defined on the geometric-motion-free texture map of the face (Fig. 19.6d). Gabor wavelets with two spatial frequency and six orientations are used to calculate Gabor coefficients. A 12-dimension appearance feature vector is computed in each of the 11 selected regions by weighted averaging of the Gabor coefficients. To track the face appearance variations, an appearance model (texture image) is trained using a Gaussian mixture model based on exemplars. Then an online adaption algorithm is employed to progressively adapt the appearance model to new conditions such as lighting changes or differences in new individuals. See [102] for details.

### 19.4.3 Facial Expression Recognition

The last step of AFEA systems is to recognize facial expression based on the extracted features. Many classifiers have been applied to expression recognition such as neural network (NN), support vector machines (SVM), linear discriminant analysis (LDA), K-nearest neighbor, multinomial logistic ridge regression (MLR), hidden Markov models (HMM), tree augmented naive Bayes, RankBoost, and others. Some systems use only a rule-based classification based on the definition of the facial actions. Here, we summarize the expression recognition methods to frame-based and sequence-based expression recognition methods. The frame-based recognition method uses only the current frame with or without a reference image (it is mainly a neutral face image) to recognize the expressions of the frame. The sequence-based recognition method uses the temporal information of the sequences to recognize the expressions for one or more frames. Table 19.7 summarizes the recognition methods, recognition rates, recognition outputs, and the databases used in the most recent systems. For the systems that used more classifiers, the best performance for person-independent test has been selected.

**Frame-Based Expression Recognition** Frame-based expression recognition does not use temporal information for the input images. It uses the information of current input image with/without a reference frame. The input image can be a static image or a frame of a sequence that is treated independently. Several methods can be found in the literature for facial expression recognition such as *neural networks* [95, 96, 116], *support vector machines* [4, 37], *linear discriminant analysis* [17], *Bayesian network* [15], and *rule-based classifiers* [70].

Tian et al. [96] employed a neural network-based recognizer to recognize FACS AUs. They used three-layer neural networks with one hidden layer to recognize AUs by a standard back-propagation method [78]. Separate networks are used for the upper and lower face. The inputs can be the normalized geometric features, the appearance feature, or both. The outputs are the recognized AUs. The network is trained to respond to the designated AUs whether they occur alone or in combination. When

**Table 19.7** FACS AU or expression recognition of recent advances. SVM, support vector machines; MLR, multinomial logistic ridge regression; HMM, hidden Markov models; BN, Bayesian network; GMM, Gaussian mixture model; RegRankBoost, RankBoost with l1 regularization

| Systems  | Recognition methods         | Recognition rate | Recognized outputs                               | Databases                          |
|----------|-----------------------------|------------------|--------------------------------------------------|------------------------------------|
| [94–96]  | Neural network (frame)      | 95.5%            | 16 single AUs and their combinations             | Ekman–Hager [31], Cohn–Kanade [49] |
| [18, 67] | Rule-based (sequence)       | 100%<br>57%      | Blink, nonblink<br>Brow up, down, and non-motion | Frank–Ekman [40]                   |
| [37]     | SVM + MLR (frame)           | 91.5%            | 6 Basic expressions                              | Cohn–Kanade [49]                   |
| [5]      | Adaboost + SVM (sequence)   | 80.1%            | 20 facial actions                                | Frank–Ekman [40]                   |
| [15]     | BN + HMM (frame & sequence) | 73.22%<br>66.53% | 6 Basic expressions<br>6 Basic expressions       | Cohn–Kanade [49]<br>UIUC–Chen [14] |
| [102]    | NN + GMM (frame)            | 71%              | 6 Basic expressions                              | Cohn–Kanade [49]                   |
| [111]    | RegRankBoost (frame)        | 88%              | 6 Basic expressions                              | Cohn–Kanade [49]                   |

AUs occur in combination, multiple output nodes are excited. To our knowledge, system of [96] was the first system to handle AU combinations. Although several other systems tried to recognize AU combinations [17, 23, 57], they treated each combination as if it were a separate AU. More than 7000 different AU combinations have been observed [83], and a system that can handle AU combinations is more efficient. A overall recognition rate of 95.5% had been achieved for neutral expression and 16 AUs whether they occurred individually or in combinations.

In [37], a two-stage classifier was employed to recognize neutral expression and six emotion-specified expressions. First, SVMs were used for the pairwise classifiers, that is, each SVM is trained to distinguish two emotions. Then they tested several approaches, such as nearest neighbor, a simple voting scheme, and multinomial logistic ridge regression (MLR) to convert the representation produced by the first stage into a probability distribution over six emotion-specified expressions and neutral. The best performance at 91.5% was achieved by MLR.

Wen and Huang [102] also employed a two-stage classifier to recognize neutral expression and six emotion-specified expressions. First, a neural network is used to classify *neutral* and *nonneutral-like* [93]. Then Gaussian mixture models (GMMs) were used for the remaining expressions. The overall average recognition rate was 71% for a people-independent test.

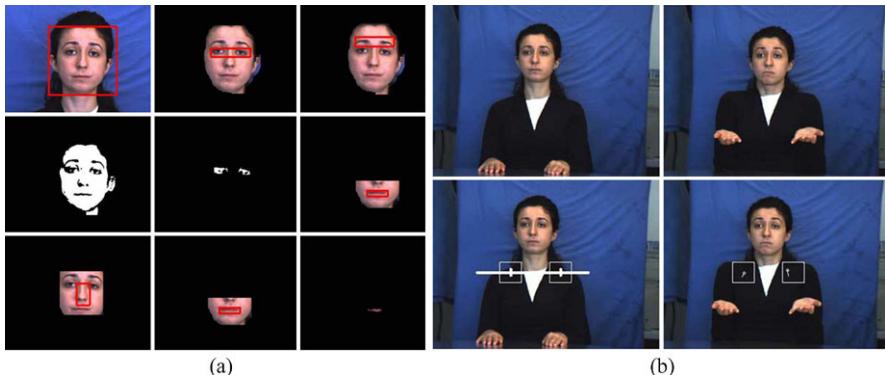
Yang et al. [111] employ RankBoost with l1 regularization for expression recognition. They also estimate the intensity of expressions by using the output ranking scores. For six emotion-specified expressions in Cohn–Kanade database, they achieved 88% recognition rate.

**Sequence-Based Expression Recognition** The sequence-based recognition method uses the temporal information of the sequences to recognize the expressions of one or more frames. To use the temporal information, the techniques such as HMM [4, 15, 17, 57], recurrent neural networks [52, 77], and rule-based classifier [18] were employed in facial expression analysis. The systems of [4, 15, 18] employed a sequence-based classifier. Note that the systems of [4] and [18] are comparative studies for FACS AU recognition in spontaneously occurring behavior by using the same database [40]. In that database, subjects were ethnically diverse, AUs occurred during speech, and out-of-plane motion and occlusion from head motion and glasses were common. So far, only several systems tried to recognize AUs or expression in spontaneously occurring behavior [4, 5, 18, 97].

The system [18] employed a rule-based classifier to recognize AUs of eye and brow in spontaneously occurring behavior by using a number of frames in the sequence. The algorithm achieved an overall accuracy of 98% for three eye behaviors: blink (AU 45), flutter, and no blink (AU 0). *Flutter* is defined as two or more rapidly repeating blinks (AU 45) with only partial eye opening (AU 41 or AU 42) between them. 100% accuracy is achieved between blinks and non-blanks. Accuracy across the three categories in the brow region (brow-up, brow-down, nonbrow motion) was 57%. The number of brow-down actions was too small for reliable point estimates. Omitting brow-down from the analysis, recognition accuracy would increase to 80%. Human FACS coders had similar difficulty with brow-down, agreeing only about 50% in this database. The small number of occurrences was no doubt a factor for FACS coders as well. The combination of occlusion from eyeglasses and correlation of forward head pitch with brow-down complicated FACS coding.

System [4] first employed SVMs to recognize AUs by using Gabor representations. Then they used hidden Markov models (HMMs) to deal with AU dynamics. HMMs were applied in two ways: (1) taking Gabor representations as input, and (2) taking the outputs of SVM as input. When they use Gabor representations as input to train HMMs, the Gabor coefficients were reduced to 100 dimensions per image using PCA. Two HMMs, one for blinks and one for nonblinks were trained and tested using leave-one-out cross-validation. A best performance of 95.7% recognition rate was obtained using five states and three Gaussians. They achieved a 98.1% recognition rate for blink and non-blink using SVM outputs as input to train HMMs for five states and three Gaussians. Accuracy across the three categories in the brow region (brow-up, brow-down, nonbrow motion) was 70.1% (HMMs trained on PCA-reduced Gabors) and 66.9% (HMMs trained on SVM outputs) respectively. Omitting brow-down, the accuracy increases to 90.9% and 89.5%, respectively.

Cohen et al. [15] first evaluated Bayesian network (frame-based) classifiers such as Gaussian naive Bayes (NB-Gaussian), Cauchy naive Bayes (NB-Cauchy), and tree-augmented-naive Bayes (TAN), focusing on changes in distribution assumptions and feature dependency structures. They also proposed a new architecture of HMMs to segment and recognize neutral and six emotion-specified expressions from video sequences. For the person-independent test in the Cohn–Kanade database [49], the best performance at recognition rate of 73.2% was achieved by the TAN classifier. See details in Cohen et al. [15].



**Fig. 19.7** Example of the face and body feature extraction employed in the FABO system [45]. **a** Face features. **b** Body features—shoulder extraction procedure. Shoulder regions found and marked on the neutral frame (*first row*), estimating the movement within the shoulder regions using optical flow (*second row*)

#### 19.4.4 Multimodal Expression Analysis

Facial expression is one of several modes of nonverbal communication. The message value of various modes may differ depending on context and may be congruent or discrepant with each other. Recently, several researchers integrated facial expression analysis with other modes such as gesture, prosody, and speech [20, 44, 45, 84]. Cohn et al. [20] investigated the relation between facial actions and vocal prosody for depression detection. They achieved the same accuracy rate at 79% by using facial actions and vocal prosody respectively. No results are reported for combination. Gunes and Piccardi [45] combined facial actions and body gestures for 9 expression recognition. They found that recognition from fused face and body modalities performs better than that from the face or the body modality alone.

For facial feature extraction in [45], following frame-by-frame face detection, a combination of appearance (e.g., wrinkles) and geometric features (e.g., feature points) is extracted from the face videos. A reference frame with neutral expression is employed for feature comparison. For body feature extraction and tracking, they detected and tracked head, shoulders and hands by using meanshift method from the body videos. Figure 19.7 shows examples of the face and body feature extraction in [45]. A total of 152 features for face modality and 170 features for body modality were used for the detection of face and body temporal segments with various classifiers including both frame-based and sequence-based methods. They tested the system on FABO database [44] and achieved recognition rate at 35.22% by only using face features and 76.87% by only using body features. The recognition rate increased to 85% with combination of both face and body features. More details can be found at [45].

**Table 19.8** Summary of databases for facial expression analysis

| Databases           | Images/<br>Videos | Subjects | Expressions                                                | Neutral | Spontaneous | Multimodal   | 3D data |
|---------------------|-------------------|----------|------------------------------------------------------------|---------|-------------|--------------|---------|
| Cohn-Kanade<br>[49] | videos            | 210      | basic<br>expressions<br>single AUs<br>AU combina-<br>tions | yes     | no          | frontal face | no      |
| FABO<br>[44]        | videos            | 23       | 9 expressions<br>hand gestures                             | yes     | no          | frontal face | no      |
| JAFFE<br>[59]       | images            | 10       | 6 basic<br>expressions                                     | yes     | no          | frontal face | no      |
| MMI<br>[71]         | images<br>videos  | 19       | single AUs<br>AU combina-<br>tions                         | yes     | no          | frontal face | no      |
| RU-FACS<br>[5]      | videos            | 100      | AU combina-<br>tions<br>AU                                 | yes     | yes         | 4 face poses | no      |
| BU-3DFE<br>[112]    | static            | 100      | 6 basic<br>expressions                                     | yes     | no          | face         | yes     |
| BU-4DFE<br>[113]    | dynamic           | 101      | 6 basic<br>expressions                                     | yes     | no          | face         | yes     |

#### 19.4.5 Databases for Facial Expression Analysis

Standard databases play important roles to train, evaluate, and compare different methods and systems for facial expression analysis. There are some public available databases (images or videos) of expression analysis for conducting comparative tests [5, 24, 40, 44, 49, 59, 63, 71, 74, 88, 112, 113]. In this chapter, we summarize several common used standard databases for facial expression analysis in Table 19.8.

Cohn-Kanade AU-Coded Face Expression Database (Cohn-Kanade) [49] is the most commonly used comprehensive database in research on automated facial expression analysis. In Cohn-Kanade database, facial behavior was recorded for two views of faces (frontal view and 30-degree view) in 210 adults between the ages of 18 and 50 years. They were 69% female, 31% male, 81% Euro-American, 13% Afro-American, and 6% other groups. In the database, 1917 image sequences from frontal view videos for 182 subjects have been FACS coded for either target action units or the entire sequence. Japanese Female Facial Expression (JAFFE) Database [59] contains 213 images of 6 basic facial expressions and neutral posed by 10 Japanese female subjects. It is the first downloadable database for facial expression analysis. MMI Facial Expression Database (MMI) [71] contains more than 1500 samples of both static images and image sequences of faces from 19 subjects in frontal and profile views displaying various facial expressions of emotion, single AUs, and AU combinations. It also includes the identification of the temporal

segments (onset, apex, offset) of shown AU and emotion facial displays. The Bi-modal Face and Body Gesture Database (FABO) [44] contains image sequences captured by two synchronized cameras (one for frontal view facial actions, and another for frontal view upper body gestures as shown in Fig. 19.7) from 23 subjects. The database is coded to neutral and nine general expressions (uncertainty, anger, surprise, fear, anxiety, happiness, disgust, boredom, and sadness) based on facial actions and body gestures. The RU-FACS Spontaneous Expression Database (RU-FACS) [5] is a dataset of spontaneous facial behavior with rigorous FACS coding. The dataset consists of 100 subjects participating in a ‘false opinion’ paradigm with speech-related mouth movements and out-of-plane head rotations from four views of face (frontal, left 45°, right 45°, and up about 22°). To date, image sequences from frontal view of 33 subjects have been FACS-coded. The database is being prepared for release. The Binghamton University 3D Facial Expression Database (BU-3DFE) [112] contains 2500 3D facial expression models including neutral and 6 basic expressions from 100 subjects. Associated with each 3D expression model, there are two corresponding facial texture images captured at two views (about +45° and -45°). The BU-4DFE database [113] is extended from a static 3D space (BU-3DFE database) to a dynamic 3D space at a video rate of 25 frames per second. BU-4DFE database contains 606 3D facial expression sequences captured from 101 subjects. Associated with each 3D expression sequence, there is a facial texture video with high resolution of 1040 × 1329 pixels per frame.

## 19.5 Open Questions

Although many recent advances and successes in automatic facial expression analysis have been achieved, as described in the previous sections, many questions remain open, for which answers must be found. Some major points are considered here.

### 1. *How do humans correctly recognize facial expressions?*

Research on human perception and cognition has been conducted for many years, but it is still unclear how humans recognize facial expressions. Which types of parameters are used by humans and how are they processed? By comparing human and automatic facial expression recognition we may be able to advance our understanding of each and discover new ways of improving automatic facial expression recognition.

### 2. *Is it always better to analyze finer levels of expression?*

Although it is often assumed that more fine-grained recognition is preferable, the answer depends on both the quality of the face images and the type of application. Ideally, an AFEA system should recognize all action units and their combinations. In high quality images, this goal seems achievable; emotion-specified expressions then can be identified based on emotion prototypes identified in the psychology literature. For each emotion, prototypic action units have been identified. In lower quality image data, only a subset of action units and emotion-specified expression may be recognized. Recognition of emotion-specified expressions directly may be needed. We seek systems that become “self aware”

about the degree of recognition that is possible based on the information of given images and adjust processing and outputs accordingly. Recognition from coarse-to-fine, for example from emotion-specified expressions to subtle action units, depends on image quality and the type of application. Indeed, for some purposes, it may be sufficient that a system is able to distinguish between positive, neutral, and negative expression, or recognize only a limited number of target action units, such as brow lowering to signal confusion, cognitive effort, or negative affect.

### *3. Is there any better way to code facial expressions for computer systems?*

Almost all the existing works have focused on recognition of facial expression, either emotion-specified expressions or FACS coded action units. The emotion-specified expressions describe expressions at a coarse level and are not sufficient for some applications. Although the FACS was designed to detect subtle changes in facial features, it is a human-observer-based system with only limited ability to distinguish intensity variation. Intensity variation is scored at an ordinal level; the interval level measurement is not defined and anchor points may be subjective. Challenges remain in designing a computer-based facial expression coding system with more quantitative definitions.

### *4. How do we obtain reliable ground truth?*

Whereas some approaches have used FACS, which is a criterion measure widely used in the psychology community for facial expression analysis, most vision-based work uses emotion-specified expressions. A problem is that emotion-specified expressions are not well defined. The same label may apply to very different facial expressions, and different labels may refer to the same expressions, which confounds system comparisons. Another problem is that the reliability of labels typically is unknown. With few exceptions, investigators have failed to report interobserver reliability and the validity of the facial expressions they have analyzed. Often there is no way to know whether subjects actually showed the target expression or whether two or more judges would agree that the subject showed the target expression. At a minimum, investigators should make explicit labeling criteria and report interobserver agreement for the labels. When the dynamics of facial expression are of interest, temporal resolution should be reported as well. Because intensity and duration measurements are critical, it is important to include descriptive data on these features as well. Unless adequate data about stimuli are reported, discrepancies across studies are difficult to interpret. Such discrepancies could be due to algorithms or to errors in ground truth determination.

### *5. How do we recognize facial expressions in real life?*

Real-life facial expression analysis is much more difficult than the posed actions studied predominantly to date. Head motion, low resolution input images, absence of a neutral face for comparison, and low intensity expressions are among the factors that complicate facial expression analysis. Recent works in 3D modeling of spontaneous head motion and action unit recognition in spontaneous facial behavior are exciting developments. How elaborate a head model is required to be in such work is as yet a research question. A cylindrical model

is relatively robust and has proven effective as a part of blink detection system [104], but highly parametric, generic, or even custom-fitted head models may prove necessary for more complete action unit recognition.

Most works to date have used a single, passive camera. Although there are clear advantages to approaches that require only a single passive camera or video source, multiple cameras are feasible in a number of settings and can be expected to provide improved accuracy. Active cameras can be used to acquire high resolution face images [46]. Also, the techniques of super-resolution can be used to obtain higher resolution images from multiple low resolution images [2]. At present, it is an open question how to recognize expressions in situations in which a neutral face is unavailable, expressions are of low intensity, or other facial or nonverbal behaviors, such as occlusion by the hands, are present.

#### 6. *How do we best use the temporal information?*

Almost all works have emphasized recognition of discrete facial expressions, regardless of being defined as emotion-specified expressions or action units. The timing of facial actions may be as important as their configuration. Recent work by our group has shown that intensity and duration of expression vary with context and that the timing of these parameters is highly consistent with automatic movement [85]. Related work suggests that spontaneous and deliberate facial expressions may be discriminated in terms of timing parameters [19], which is consistent with neuropsychological models [75] and may be important to lie detection efforts. Attention to timing is also important in guiding the behavior of computer avatars. Without veridical timing, believable avatars and ones that convey intended emotions and communicative intents may be difficult to achieve.

#### 7. *How may we integrate facial expression analysis with other modalities?*

Facial expression is one of several modes of nonverbal communication. The message value of various modes may differ depending on context and may be congruent or discrepant with each other. An interesting research topic is the integration of facial expression analysis with that of gesture, prosody, and speech. Combining facial features with acoustic features would help to separate the effects of facial actions due to facial expression and those due to speech-related movements. The combination of facial expression and speech can be used to improve speech recognition and multimodal person identification [39].

## 19.6 Conclusions

Five recent trends in automatic facial expression analysis are (1) diversity of facial features in an effort to increase the number of expressions that may be recognized; (2) recognition of facial action units and their combinations rather than more global and easily identified emotion-specified expressions; (3) more robust systems for face acquisition, facial data extraction and representation, and facial expression recognition to handle head motion (both in-plane and out-of-plane), occlusion, lighting change, and low intensity expressions, all of which are common in spontaneous facial behavior in naturalistic environments; (4) fully automatic and real-time AFEA

systems; and (5) combination of facial actions with other modes such as gesture, prosody, and speech. All of these developments move AFEA toward real-life applications. Several databases that addresses most problems for deliberate facial expression analysis have been released to researchers to conduct comparative tests of their methods. Databases with ground-truth labels, preferably both action units and emotion-specified expressions, are needed for the next generation of systems, which are intended for naturally occurring behavior (spontaneous and multimodal) in real-life settings. Work in spontaneous facial expression analysis is just now emerging and potentially will have significant impact across a range of theoretical and applied topics.

**Acknowledgements** We sincerely thank Zhen Wen and Hatice Gunes for providing pictures and their permission to use them in this chapter.

## References

1. Ahlberg, J., Forchheimer, R.: Face tracking for model-based coding and face animation. *Int. J. Imaging Syst. Technol.* **13**(1), 8–22 (2003)
2. Baker, S., Kanade, T.: Limits on super-resolution and how to break them. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(9), 1167–1183 (2002)
3. Bartlett, M., Hager, J., Ekman, P., Sejnowski, T.: Measuring facial expressions by computer image analysis. *Psychophysiology* **36**, 253–264 (1999)
4. Bartlett, M., Braathen, B., Littlewort-Ford, G., Hershey, J., Fasel, I., Marks, T., Smith, E., Sejnowski, T., Movellan, J.R.: Automatic analysis of spontaneous facial behavior: A final project report. Technical Report INC-MPLab-TR-2001.08, Machine Perception Lab, Institute for Neural Computation, University of California, San Diego (2001)
5. Bartlett, M., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., Movellan, J.: Automatic recognition of facial actions in spontaneous expressions. *J. Multimed.* **1**(6), 22–35 (2006)
6. Black, M.: Robust incremental optical flow. PhD thesis, Yale University (1992)
7. Black, M., Yacoob, Y.: Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In: Proc. of International Conference on Computer Vision, pp. 374–381 (1995)
8. Black, M., Yacoob, Y.: Recognizing facial expressions in image sequences using local parameterized models of image motion. *Int. J. Comput. Vis.* **25**(1), 23–48 (1997)
9. Brown, L., Tian, Y.-L.: Comparative study of coarse head pose estimation. In: IEEE Workshop on Motion and Video Computing, Orlando (2002)
10. Camras, L., Lambrecht, L., Michel, G.: Infant surprise expressions as coordinative motor structures. *J. Nonverbal Behav.* **20**, 183–195 (1966)
11. Carroll, J., Russell, J.: Do facial expression signal specific emotions? *J. Pers. Soc. Psychol.* **70**, 205–218 (1996)
12. Carroll, J., Russell, J.: Facial expression in Hollywood’s portrayal of emotion. *J. Pers. Soc. Psychol.* **72**, 164–176 (1997)
13. Chang, Y., Hu, C., Feris, R., Turk, M.: Manifold based analysis of facial expression. *Image Vis. Comput.* **24**(6), 605–614 (2006)
14. Chen, L.: Joint processing of audio-visual information for the recognition of emotional expressions in human-computer interaction. PhD thesis, University of Illinois at Urbana-Champaign, Department of Electrical Engineering (2000)
15. Cohen, I., Sebe, N., Cozman, F., Cirelo, M., Huang, T.: Coding, analysis, interpretation, and recognition of facial expressions. *J. Comput. Vis. Image Underst.* (2003). Special Issue on Face Recognition

16. Cohn, J., Katz, G.: Bimodal expression of emotion by face and voice. In: ACM and ATR Workshop on Face/Gesture Recognition and Their Applications, pp. 41–44 (1998)
17. Cohn, J., Zlochower, A., Lien, J., Kanade, T.: Automated face analysis by feature point tracking has high concurrent validity with manual faces coding. *Psychophysiology* **36**, 35–43 (1999)
18. Cohn, J., Kanade, T., Moriyama, T., Ambadar, Z., Xiao, J., Gao, J., Immura, H.: A comparative study of alternative faces coding algorithms. Technical Report CMU-RI-TR-02-06, Robotics Institute, Carnegie Mellon University, Pittsburgh, November 2001
19. Cohn, J., Schmidt, K., Gross, R., Ekman, P.: Individual differences in facial expression: stability over time, relation to self-reported emotion, and ability to inform person identification. In: Proceedings of the International Conference on Multimodal User Interfaces (ICMI 2002), pp. 491–496 (2002)
20. Cohn, J., Kreuz, T., Yang, Y., Nguyen, M., Padilla, M., Zhou, F., Fernando, D.: Detecting depression from facial actions and vocal prosody. In: International Conference on Affective Computing and Intelligent Interaction (ACII2009) (2009)
21. Darwin, C.: *The Expression of Emotions in Man and Animals*. Murray, London (1872), reprinted by University of Chicago Press, 1965
22. Daugmen, J.: Complete discrete 2d Gabor transforms by neural networks for image analysis and compression. *IEEE Trans. Acoust. Speech Signal Process.* **36**(7), 1169–1179 (1988)
23. Donato, G., Bartlett, M., Hager, J., Ekman, P., Sejnowski, T.: Classifying facial actions. *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(10), 974–989 (1999)
24. Douglas-Cowie, E., Cowie, R., Schroeder, M.: The description of naturally occurring emotional speech. In: International Conference of Phonetic Sciences (2003)
25. Eihl-Eihesfeldt, I.: *Human Ethology*. Aldine de Gruyter, New York (1989)
26. Ekman, P.: The Argument and Evidence about Universals in Facial Expressions of Emotion, vol. 58, pp. 143–164. Wiley, New York (1989)
27. Ekman, P.: Facial expression and emotion. *Am. Psychol.* **48**, 384–392 (1993)
28. Ekman, P., Friesen, W.: *Pictures of Facial Affect*. Consulting Psychologist, Palo Alto (1976)
29. Ekman, P., Friesen, W.: *The Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, San Francisco (1978)
30. Ekman, P., Rosenberg, E.E.: *What the Face Reveals*. Oxford University, New York (1997)
31. Ekman, P., Hager, J., Methvin, C., Irwin, W.: Ekman–Hager facial action exemplars. Human Interaction Laboratory, University of California, San Francisco
32. Essa, I., Pentland, A.: Coding, analysis, interpretation, and recognition of facial expressions. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 757–763 (1997)
33. Farkas, L., Munro, I.: *Anthropometric Facial Proportions in Medicine*. Charles C Thomas, Springfield (1987)
34. Fasel, B., Luttin, J.: Recognition of asymmetric facial action unit activities and intensities. In: Proceedings of International Conference of Pattern Recognition (2000)
35. Fasel, B., Luttin, J.: Automatic facial expression analysis: Survey. *Pattern Recognit.* **36**(1), 259–275 (2003)
36. Fleiss, J.: *Statistical Methods for Rates and Proportions*. Wiley, New York (1981)
37. Ford, G.: Fully automatic coding of basic expressions from video. Technical Report INC-MPLab-TR-2002.03, Machine Perception Lab, Institute for Neural Computation, University of California, San Diego (2002)
38. Fox, N., Reilly, R.: Audio-visual speaker identification. In: Proc. of the 4th International Conference on Audio- and Video-Based Biometric Person Authentication (2003)
39. Fox, N., Gross, R., de Chazal, P., Cohn, J., Reilly, R.: Person identification using multimodal features: speech, lip, and face. In: Proc. of ACM Multimedia Workshop in Biometrics Methods and Applications (WBMA 2003), CA (2003)
40. Frank, M., Ekman, P.: The ability to detect deceit generalizes across different types of high-stake lies. *Pers. Soc. Psychol.* **72**, 1429–1439 (1997)
41. Friesen, W., Ekman, P.: Emfac-s-7: emotional facial action coding system. Unpublished manuscript, University of California at San Francisco (1983)

42. Fujiyoshi, H., Lipton, A.: Real-time human motion analysis by image skeletonization. In: Proc. of the Workshop on Application of Computer Vision (1998)
43. Fukui, K., Yamaguchi, O.: Facial feature point extraction method based on combination of shape extraction and pattern matching. *Syst. Comput. Jpn.* **29**(6), 49–58 (1998)
44. Gunes, H., Piccardi, M.: A bimodal face and body gesture database for automatic analysis of human nonverbal affective behavior. In: International Conference on Pattern Recognition (ICPR), pp. 1148–1153 (2006)
45. Gunes, H., Piccardi, M.: Automatic temporal segment detection and affect recognition from face and body display. *IEEE Trans. Syst. Man Cybern., Part B, Cybern.* **39**(1), 64–84 (2009)
46. Hampapur, A., Pankanti, S., Senior, A., Tian, Y., Brown, L., Bolle, R.: Face cataloger: multi-scale imaging for relating identity to location. In: Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance (2003)
47. Heisele, B., Serre, T., Pontil, M., Poggio, T.: Component-based face detection. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognit. (CVPR) (2001)
48. Izard, C., Doughterty, L., Hembree, E.A.: A system for identifying affect expressions by holistic judgments. Unpublished Manuscript, University of Delaware (1983)
49. Kanade, T., Cohn, J., Tian, Y.-L.: Comprehensive database for facial expression analysis. In: Proceedings of International Conference on Face and Gesture Recognition, pp. 46–53 (2000)
50. Kimura, S., Yachida, M.: Facial expression recognition and its degree estimation. In: Proc. of the International Conference on Computer Vision and Pattern Recognition, pp. 295–300 (1997)
51. Kleck, R., Mendolia, M.: Decoding of profile versus full-face expressions of affect. *J. Non-verbal Behav.* **14**(1), 35–49 (1990)
52. Kobayashi, H., Tange, K., Hara, F.: Dynamic recognition of six basic facial expressions by discrete-time recurrent neural network. In: Proc. of the International Joint Conference on Neural Networks, pp. 155–158 (1993)
53. Kobayashi, H., Tange, K., Hara, F.: Real-time recognition of six basic facial expressions. In: Proc. IEEE Workshop on Robot and Human Communication, pp. 179–186 (1995)
54. Kraut, R., Johnson, R.: Social and emotional messages of smiling: an ethological approach. *J. Pers. Soc. Psychol.* **37**, 1539–1523 (1979)
55. Li, S., Gu, L.: Real-time multi-view face detection, tracking, pose estimation, alignment, and recognition. In: IEEE Conf. on Computer Vision and Pattern Recognition Demo Summary (2001)
56. Lien, J.-J., Kanade, T., Cohn, J., Li, C.: Subtly different facial expression recognition and expression intensity estimation. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 853–859 (1998)
57. Lien, J.-J., Kanade, T., Cohn, J., Li, C.: Detection, tracking, and classification of action units in facial expression. *J. Robot. Auton. Syst.* **31**, 131–146 (2000)
58. Lucey, S., Wang, Y., Cox, M., Sridharan, S., Cohn, J.: Efficient constrained local model fitting for non-rigid face alignment. *Image Vis. Comput.* **27**(12), 1804–1813 (2009)
59. Lyons, M., Akamasku, S., Kamachi, M., Gyoba, J.: Coding facial expressions with Gabor wavelets. In: Proceedings of International Conference on Face and Gesture Recognition (1998)
60. Mahoor, M., Cadavid, S., Messinger, D., Cohn, J.: A framework for automated measurement of the intensity of non-posed facial action units. In: IEEE Workshop on CVPR for Human Communicative Behavior Analysis, pp. 74–80 (2009)
61. Manstead, A.: Expressiveness as an Individual Difference, pp. 285–328. Cambridge University Press, Cambridge (1991)
62. Martin, P., Bateson, P.: Measuring Behavior: An Introductory Guide. Cambridge University Press, Cambridge (1986)
63. Martinez, A., Benavente, R.: The ar face database. CVC Technical Report, No. 24 (1998)
64. Mase, K.: Recognition of facial expression from optical flow. *IEICE Trans. Electron.* **74**(10), 3474–3483 (1991)
65. Matias, R., Cohn, J., Ross, S.: A comparison of two systems to code infants' affective expression. *Dev. Psychol.* **25**, 483–489 (1989)

66. Matthews, I., Baker, S.: Active appearance models revisited. *Int. J. Comput. Vis.* **60**(2), 135–164 (2004)
67. Moriyama, T., Kanade, T., Cohn, J., Xiao, J., Ambadar, Z., Gao, J., Imanura, M.: Automatic recognition of eye blinking in spontaneously occurring behavior. In: Proceedings of the 16th International Conference on Pattern Recognition (ICPR '2002), vol. 4, pp. 78–81 (2002)
68. Moses, Y., Reynard, D., Blake, A.: Determining facial expressions in real time. In: Proc. of Int. Conf. On Automatic Face and Gesture Recognition, pp. 332–337 (1995)
69. Pantic, M., Rothkrantz, L.: Automatic analysis of facial expressions: the state of the art. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(12), 1424–1445 (2000)
70. Pantic, M., Rothkrantz, L.: Expert system for automatic analysis of facial expression. *Image Vis. Comput.* **18**(11), 881–905 (2000)
71. Pantic, M., Valstar, M., Rademaker, R., Maat, L.: Web-based database for facial expression analysis. In: International conference on Multimedia and Expo (ICME05) (2005)
72. Pentland, A., Moghaddam, B., Starner, T.: View-based and modular eigenspaces for face recognition. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 84–91 (1994)
73. Pighin, F., Szeliski, H., Salesin, D.: Synthesizing realistic facial expressions from photographs. In: Proc of SIGGRAPH (1998)
74. Pilz, K., Thornton, I., Bühlhoff, H.: A search advantage for faces learned in motion. In: Experimental Brain Research (2006)
75. Rinn, W.: The neuropsychology of facial expression: a review of the neurological and psychological mechanisms for producing facial expressions. *Psychol. Bull.* **95**, 52–77 (1984)
76. Rizvi, S., Phillips, P., Moon, H.: The Feret verification testing protocol for face recognition algorithms. In: Proceedings of the Third International Conference on Automatic Face and Gesture Recognition, pp. 48–55 (1998)
77. Rosenblum, M., Yacoob, Y., Davis, L.: Human expression recognition from motion using a radial basis function network architecture. *IEEE Trans. Neural Netw.* **7**(5), 1121–1138 (1996)
78. Rowley, H., Baluja, S., Kanade, T.: Neural network-based face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(1), 23–38 (1998)
79. Russell, J.: Culture and the categorization. *Psychol. Bull.* **110**, 426–450 (1991)
80. Russell, J.: Is there universal recognition of emotion from facial expression? A review of the cross-cultural studies. *Psychol. Bull.* **115**, 102–141 (1991)
81. Samal, A., Iyengar, P.: Automatic recognition and analysis of human faces and facial expressions: A survey. *Pattern Recognit.* **25**(1), 65–77 (1992)
82. Sayette, M., Cohn, J., Wertz, J., Perrott, M., Parrott, D.: A psychometric evaluation of the facial action coding system for assessing spontaneous expression. *J. Nonverbal Behav.* **25**, 167–186 (2001)
83. Scherer, K., Ekman, P.: Handbook of Methods in Nonverbal Behavior Research. Cambridge University Press, Cambridge (1982)
84. Scherer, K., Ellgring, H.: Multimodal expression of emotion: Affect programs or componential appraisal patterns? *Emotion* **7**(1), 158–171 (2007)
85. Schmidt, K., Cohn, J.F., Tian, Y.-L.: Signal characteristics of spontaneous facial expressions: Automatic movement in solitary and social smiles. *Biol. Psychol.* (2003)
86. Schneiderman, H., Kanade, T.: A statistical model for 3d object detection applied to faces and cars. In: IEEE Conference on Computer Vision and Pattern Recognition. IEEE, New York (2000)
87. Shan, C., Gong, S., McOwan, P.: Facial expression recognition based on local binary patterns: A comprehensive study. *Image Vis. Comput.* **27**(6), 803–816 (2009)
88. Sim, T., Baker, S., Bsat, M.: The cmu pose, illumination, and expression database. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(12), 1615–1618 (2003)
89. Sung, K., Poggio, T.: Example-based learning for view-based human face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(1), 39–51 (1998)

90. Suwa, M., Sugie, N., Fujimora, K.: A preliminary note on pattern recognition of human emotional expression. In: International Joint Conference on Pattern Recognition, pp. 408–410 (1978)
91. Tao, H., Huang, T.: Explanation-based facial motion tracking using a piecewise Bezier volume deformation model. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition (1999)
92. Terzopoulos, D., Waters, K.: Analysis of facial images using physical and anatomical models. In: IEEE International Conference on Computer Vision, pp. 727–732 (1990)
93. Tian, Y.-L., Bolle, R.: Automatic detecting neutral face for face authentication. In: Proceedings of AAAI-03 Spring Symposium on Intelligent Multimedia Knowledge Management, CA (2003)
94. Tian, Y.-L., Kanade, T., Cohn, J.: Eye-state action unit detection by Gabor wavelets. In: Proceedings of International Conference on Multi-modal Interfaces (ICMI 2000), pp. 143–150, September 2000
95. Tian, Y.-L., Kanade, T., Cohn, J.: Recognizing action units for facial expression analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(2), 1–19 (2001)
96. Tian, Y.-L., Kanade, T., Cohn, J.: Evaluation of Gabor-wavelet-based facial action unit recognition in image sequences of increasing complexity. In: Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition (FG'02), Washington, DC (2002)
97. Tian, Y.-L., Brown, L., Hampapur, A., Pankanti, S., Senior, A., Bolle, R.: Real world real-time automatic recognition of facial expressions. In: Proceedings of IEEE Workshop on Performance Evaluation of Tracking and Surveillance, Graz, Austria (2003)
98. Toyama, K.: “Look, ma—no hands!” hands-free cursor control with real-time 3d face tracking. In: Proc. Workshop on Perceptual User Interfaces (PUI'98) (1998)
99. VanSwearingen, J., Cohn, J., Bajaj-Luthra, A.: Specific impairment of smiling increases severity of depressive symptoms in patients with facial neuromuscular disorders. *J. Aesthet. Plast. Surg.* **23**, 416–423 (1999)
100. Vetter, T.: Learning novel views to a single face image. In: Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, pp. 22–29 (1995)
101. Viola, P., Jones, M.: Robust real-time object detection. In: International Workshop on Statistical and Computational Theories of Vision—Modeling, Learning, Computing, and Sampling (2001)
102. Wen, Z., Huang, T.: Capturing subtle facial motions in 3d face tracking. In: Proc. of Int. Conf. on Computer Vision (2003)
103. Wu, Y., Toyama, K.: Wide-range person and illumination-insensitive head orientation estimation. In: Proceedings of International Conference on Automatic Face and Gesture Recognition, pp. 183–188 (2000)
104. Xiao, J., Moriyama, T., Kanade, T., Cohn, J.: Robust full-motion recovery of head by dynamic templates and re-registration techniques. *Int. J. Imaging Syst. Technol.* (2003)
105. Xiao, J., Baker, S., Matthews, I., Kanade, T.: Real-time combined 2d + 3d active appearance models. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 535–542 (2004)
106. Yacoob, Y., Black, M.: Parameterized modeling and recognition of activities. In: Proc. 6th IEEE Int. Conf. on Computer Vision, pp. 120–127, Bombay (1998)
107. Yacoob, Y., Davis, L.: Recognizing human facial expression from long image sequences using optical flow. *IEEE Trans. Pattern Anal. Mach. Intell.* **18**(6), 636–642 (1996)
108. Yacoob, Y., Lam, H.-M., Davis, L.: Recognizing faces showing expressions. In: Proc. Int. Workshop on Automatic Face- and Gesture-Recognition, pp. 278–283, Zurich, Switzerland (1995)
109. Yang, M., Kriegman, D., Ahuja, N.: Detecting faces in images: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(1) (2002)
110. Yang, P., Liu, Q., Metaxas, D.: Facial expression recognition using encoded dynamic features. In: International conference on Computer Vision and Pattern Recognition (CVPR) (2008)

111. Yang, P., Liu, Q., Cui, X., Metaxas, D.: Rankboost with l1 regularization for facial expression recognition and intensity estimation. In: International conference on Computer Vision (ICCV) (2009)
112. Yin, L., Wei, X., Sun, Y., Wang, J., Rosato, M.: A 3d facial expression database for facial behavior research. In: International Conference on Automatic Face and Gesture Recognition, pp. 211–216 (2006)
113. Yin, L., Chen, X., Sun, Y., Worm, T., Reale, M.: A high-resolution 3d dynamic facial expression database. In: International Conference on Automatic Face and Gesture Recognition (2008)
114. Zeng, Z., Pantic, G.R.M., Huang, T.: A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(1), 39–58 (2009)
115. Zhang, Y., Ji, Q.: Facial expression recognition with dynamic Bayesian networks. *IEEE Trans. Pattern Anal. Mach. Intel.* **27**(5) (2005)
116. Zhang, Z., Lyons, M., Schuster, M., Akamatsu, S.: Comparison between geometry-based and Gabor-wavelets-based facial expression recognition using multi-layer perceptron. In: International Workshop on Automatic Face and Gesture Recognition, pp. 454–459 (1998)
117. Zhang, Y., Ji, Q., Zhu, Z., Yi, B.: Dynamic facial expression analysis and synthesis with mpeg-4 facial animation parameters. *IEEE Trans. Circuits Syst. Video Technol.* **18**(10), 1383–1396 (2008)
118. Zhao, L., Pingali, G., Carlbom, I.: Real-time head orientation estimation using neural networks. In: Proc of the 6th International Conference on Image Processing (2002)
119. Zlochower, A., Cohn, J., Lien, J., Kanade, T.: A computer vision based method of facial expression analysis in parent-infant interaction. In: International Conference on Infant Studies, Atlanta (1998)



# Chapter 20

## Face Synthesis

Yang Wang, Zicheng Liu, and Baining Guo

### 20.1 Introduction

How to synthesize photorealistic images of human faces has been a fascinating yet difficult problem in computer graphics. Here, the term “face synthesis” refers to synthesis of still images as well as synthesis of facial animations. In general, it is difficult to draw a clean line between the synthesis of still images and that of facial animations. For example, the technique of synthesizing facial expression images can be directly used for generating facial animations, and most of the facial animation systems involve the synthesis of still images. In this chapter, we focus more on the synthesis of still images and skip most of the aspects that mainly involve the motion over time.

Face synthesis has many interesting applications. In the film industry, people would like to create virtual human characters that are indistinguishable from the real ones. In games, people have been trying to create human characters that are interactive and realistic. There are commercially available products [18, 19] that allow people to create realistic looking avatars that can be used in chat rooms, e-mail, greeting cards, and teleconferencing. Many human-machine dialog systems use realistic-looking human faces as visual representation of the computer agent that interacts with the human user. Face synthesis techniques have also been used for talking head compression in the video conferencing scenario.

---

Y. Wang (✉)  
Carnegie Mellon University, Pittsburgh, PA 15213, USA  
e-mail: [wangy@cs.cmu.edu](mailto:wangy@cs.cmu.edu)

Z. Liu  
Microsoft Research, Redmond, WA 98052, USA  
e-mail: [zliu@microsoft.com](mailto:zliu@microsoft.com)

B. Guo  
Microsoft Research Asia, Beijing 100080, China  
e-mail: [bainguo@microsoft.com](mailto:bainguo@microsoft.com)

The techniques of face synthesis can be useful for face recognition too. Romdhani et al. [47, 48] used their three dimensional (3D) face modeling technique for face recognition with different poses and lighting conditions. Qing et al. [44] used the face relighting technique as proposed by Wen et al. [59] for face recognition under a different lighting environment. Wang et al. [57] used the 3D spherical harmonic morphable model (SHBMM), an integration of spherical harmonics into the morphable model framework, for face recognition under arbitrary pose and illumination conditions. Many face analysis systems use an analysis-by-synthesis loop where face synthesis techniques are part of the analysis framework.

In this chapter, we review recent advances on face synthesis including 3D face modeling, face relighting, and facial expression synthesis.

## 20.2 Face Modeling

In the past a few years, there has been a lot of work on the reconstruction of face models from images [12, 23, 27, 41, 47, 52, 67]. There are commercially available software packages [18, 19] that allow a user to construct their personalized 3D face models. In addition to its applications in games and entertainment, face modeling techniques can also be used to help with face recognition tasks especially in handling different head poses (see Romdhani et al. [48] and Chap. 10). Face modeling techniques can be divided into three categories: face modeling from an image sequence, face modeling from two orthogonal views, and face modeling from a single image. An image sequence is typically a video of someone’s head turning from one side to the other. It contains a minimum of two views. The motion between each two consecutive views is relatively small, so it is feasible to perform image matching.

### 20.2.1 Face Modeling from an Image Sequence

Given an image sequence, one common approach for face modeling typically consists of three steps: image matching, structure from motion, and model fitting. First, two or three relatively frontal views are selected, and some image matching algorithms are used to compute point correspondences. The selection of frontal views are usually done manually. Point correspondences are computed either by using dense matching techniques such as optimal flow or feature-based corner matching. Second, one needs to compute the head motion and the 3D structures of the tracked points. Finally, a face model is fitted to the reconstructed 3D points. People have used different types of face model representations including parametric surfaces [13], linear class face scans [5], and linear class deformation vectors [34].

Fua and Miccio [13, 14] computed dense matching using image correlations. They then used a model-driven bundle adjustment technique to estimate the motions and compute the 3D structures. The idea of the model-driven bundle adjustment is to add a regularizer constraint to the traditional bundle adjustment formulation. The

constraint is that the reconstructed 3D points can be fit to a parametric face model. Finally, they fit a parametric face model to the reconstructed 3D points. Their parametric face model contains a generic face mesh and a set of control points each controlling a local area of the mesh. By adjusting the coefficients of the control points, the mesh deforms in a linear fashion. Denote  $c_1, c_2, \dots, c_m$  to be the coefficients of the control points. Let  $R, T, s$  be the rotation, translation, and scaling parameters of the head pose. Denote the mesh of the face as  $S = S(c_1, c_2, \dots, c_m)$ . Let  $\mathcal{T}$  denote the transformation operator, which is a function of  $R, T, s$ . The model fitting can be formulated as a minimization problem

$$\min \sum_i \text{Dist}[P_i, \mathcal{T}(S)], \quad (20.1)$$

where  $P_i$  is the reconstructed 3D points, and  $\text{Dist}(P_i, \mathcal{T}(S))$  is the distance from  $P_i$  to the surface  $\mathcal{T}(S)$ .

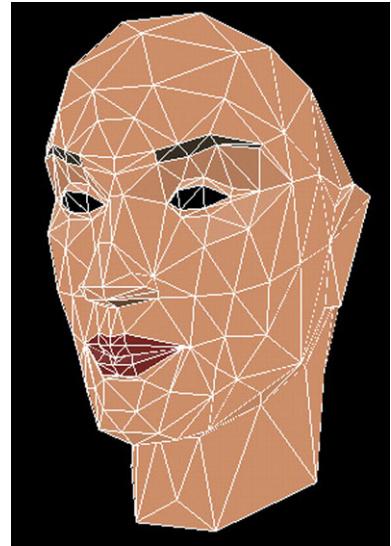
This minimization problem can be solved using an iterative closest point approach. First,  $c_1, \dots, c_m$  are initialized and fixed. For each point  $P_i$ , find its closest point  $Q_i$  on the surface  $S$ . Then solve for the pose parameters  $R, T, s$  to minimize  $\sum_i \|P_i - \mathcal{T}(Q_i)\|$  by using the quaternion-based technique [17]. The head pose parameters are then fixed. Because  $S$  is a linear function of  $c_1, \dots, c_m$ , (20.1) becomes a linear system and can be solved through a least-square procedure. At the next iteration, the newly estimated  $c_1, \dots, c_m$  are fixed, and we solve for  $R, T, s$  again.

Liu et al. [32, 34] developed a face modeling system that allows an untrained user with a personal computer and an ordinary video camera to create and instantly animate his or her face model. The user first turns his or her head from one side to the other. Then two frames pop up, and the user is required to mark five feature points (two inner eye corners, two mouth corners, and the nose top) on each view. After that, the system is completely automatic. Once the process finishes, his or her constructed face model is displayed and animated. The authors used a feature-based approach to find correspondences. It consists of three steps: (1) detecting corners in each image; (2) matching corners between the two images; (3) detecting false matches based on a robust estimation technique. The reader is referred to Liu et al. [34] for details. Compared to the optical flow approach, the feature-based approach is more robust to intensity and color variations.

After the matching is done, they used both the corner points from the image matching and the five feature points clicked by the user to estimate the camera motion. Because of the matching errors for the corner points and the inaccuracy of the user-clicked points, it is not robust to directly use these points for motion estimation. Therefore they used the physical properties of the user-clicked feature points to improve the robustness. They used the face symmetry property to reduce the number of unknowns and put reasonable bounds on the physical quantities (such as the height of the nose). In this way, the algorithm becomes significantly more robust. The algorithm's details were described by Liu and Zhang [32].

For the model fitting, they used a linear class of face geometries as their model space. A face was represented as a linear combination of a neutral face (Fig. 20.1) and some number of face *metrics*, where a metric is a vector that linearly deforms

**Fig. 20.1** Neutral face



a face in certain way, such as to make the head wider, the nose bigger, and so on. To be more precise, let us denote the face geometry by a vector  $\mathcal{S} = (\mathbf{v}_1^T, \dots, \mathbf{v}_n^T)^T$ , where  $\mathbf{v}_i = (X_i, Y_i, Z_i)^T$  ( $i = 1, \dots, n$ ) are the vertices, and a metric by a vector  $\mathcal{M} = (\delta\mathbf{v}_1^T, \dots, \delta\mathbf{v}_n^T)^T$ , where  $\delta\mathbf{v}_i = (\delta X_i, \delta Y_i, \delta Z_i)^T$ . Given a neutral face  $\mathcal{S}^0 = (\mathbf{v}_1^{0T}, \dots, \mathbf{v}_n^{0T})^T$  and a set of  $m$  metrics  $\mathcal{M}^j = (\delta\mathbf{v}_1^{jT}, \dots, \delta\mathbf{v}_n^{jT})^T$ , the linear space of face geometries spanned by these metrics is

$$\mathcal{S} = \mathcal{S}^0 + \sum_{j=1}^m c_j \mathcal{M}^j \quad \text{subject to } c_j \in [l_j, u_j] \quad (20.2)$$

where  $c_j$  represents the metric coefficients, and  $l_j$  and  $u_j$  are the valid range of  $c_j$ .

The model fitting algorithm is similar to the approach by Fua and Miccio [13, 14], described earlier in this section. The advantage of using a linear class of face geometries is that it is guaranteed that every face in the space is a reasonable face, and, furthermore, it has fine-grain control because some metrics are global whereas others are only local. Even with a small number of 3D corner points that are noisy, it is still able to generate a reasonable face model. Figure 20.2 shows side-by-side comparisons of the original images with the reconstructed models for various people.

Note that in both approaches just described the model fitting is separated from the motion estimation. In other words, the resulting face model is not used to improve the motion estimation.

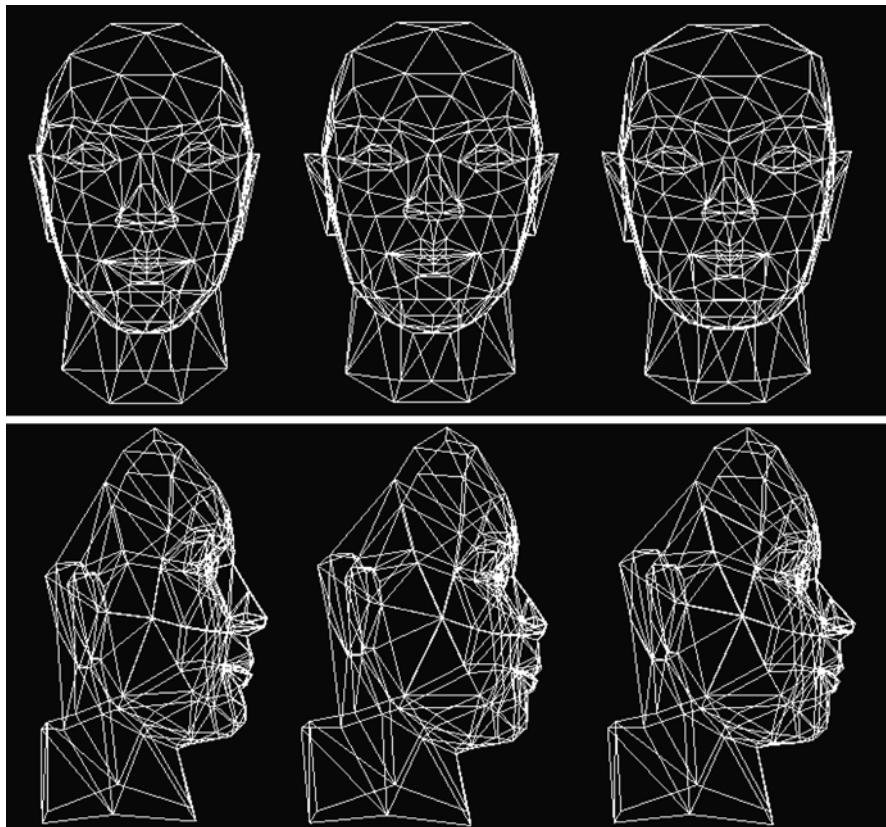
During motion estimation, the algorithm by Liu et al. [34] used only general physical properties about human faces. Even though Fua and Miccio [13, 14] used face model during motion estimation, they used it only as a regularizer constraint. The 3D model obtained with their model-driven bundle adjustment is in general



**Fig. 20.2** Side by side comparison of the original images with the reconstructed models of various people

inaccurate, and they have to throw away the model and use an additional step to recompute the 3D structure. The problem is that the camera motions are fixed on the second step. It may happen that the camera motions are not accurate owing to the inaccurate model at the first stage, so the structure computed at the second stage may not be optimal either. What one needs is to optimize camera motion and structure together.

Shan et al. [49] proposed an algorithm, called model-based bundle adjustment, that combines the motion estimation and model fitting into a single formulation. Their main idea was to directly use the model space as a search space. The model parameters (metric coefficients) become the unknowns in their bundle adjustment formulation. The variables for the 3D positions of the feature points, which are unknowns in the traditional bundle adjustment, are eliminated. Because the number of model parameters is in general much smaller than the isolated points, it results in a smaller search space and better posed optimization system.



**Fig. 20.3** Face mesh comparison. *Left*: traditional bundle adjustment; *Middle*: ground truth; *Right*: model-based bundle adjustment. (From Shan et al. [49], with permission)

Figure 20.3 shows the comparisons of the model-based bundle adjustment with the traditional bundle adjustment. On the top are the front views, and on the bottom are the side views. On each row, the one in the middle is the ground truth, on the left is the result from the traditional bundle adjustment, and on the right is the result from the model-based bundle adjustment. By looking closely, we can see that the result of the model-based bundle adjustment is much closer to the ground truth mesh. For example, on the bottom row, the nose on the left mesh (traditional bundle adjustment) is much taller than the nose in the middle (ground truth). The nose on the right mesh (model-based bundle adjustment) is similar to the one in the middle.

### 20.2.2 Face Modeling from Two Orthogonal Views

A number of researchers have proposed that we create face models from two orthogonal views [1, 8, 20]: one frontal view and one side view. The frontal view

provides the information relative to the horizontal and vertical axis, and the side view provides depth information. The user needs to manually mark a number of feature points on both images. The feature points are typically the points around the face features, including eyebrows, eyes, nose, and mouth. Because of occlusions, the number of feature points on the two views are in general different. The quality of the face model depends on the number of feature points the user provides. The more feature points, the better the model, but one needs to balance between the amount of manual work required from the user and the quality of the model.

Because the algorithm is so simple to implement and there is no robustness issue, this approach has been used in some commercially available systems [19]. Some systems provide a semiautomatic interface for marking the feature points to reduce the amount of the manual work. The disadvantage is that it is not convenient to obtain two orthogonal views, and it requires quite a number of manual interventions even with the semiautomatic interfaces.

### 20.2.3 Face Modeling from a Single Image

Blanz and Vetter [5] developed a system to create 3D face models from a single image. They used both a database of face geometries and a database of face textures. The geometry space is the linear combination of the example faces in the geometry database. The texture space is the linear combination of the example texture images in the image database. Given a face image, they search for the coefficients of the geometry space and the coefficients of the texture space so the synthesized image matches the input image. More details can be found in Chap. 10 and in their paper [5]. One limitation of their current system is that it can only handle the faces whose skin types are similar to the examples in the database. One could potentially expand the image database to cover more varieties of skin types, but there would be more parameters and it is not clear how it is going to affect the robustness of the system.

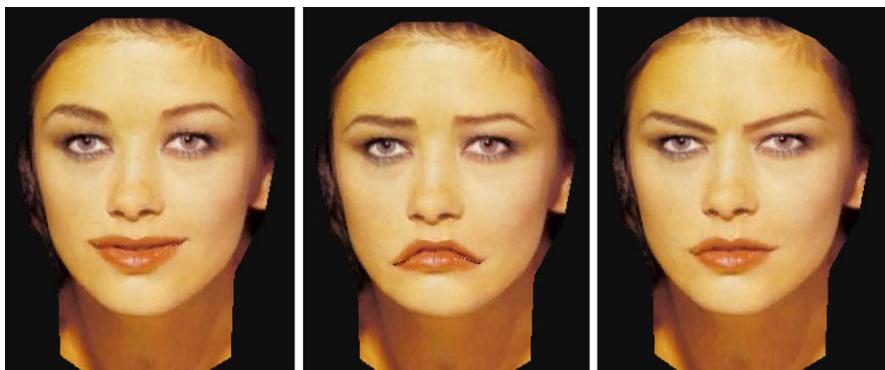
Liu [31] developed a fully automatic system to construct 3D face models from a single frontal image. They first used a face detection algorithm to find a face and then a feature alignment algorithm to find face features. By assuming an orthogonal projection, they fit a 3D face model by using the linear space of face geometries described in Sect. 20.2.1. Given that there are existing face detection and feature alignment systems [28, 62], implementing this system is simple. The main drawback of this system is that the depth of the reconstructed model is in general not accurate. For small head rotations, however, the model is recognizable. Figure 20.4 shows an example where the left is the input image and the right is the feature alignment result. Figure 20.5 shows the different views of the reconstructed 3D model. Figure 20.6 shows the results of making expressions for the reconstructed face model.



**Fig. 20.4** *Left:* input image. *Right:* the result from image alignment. (From Liu [31], with permission)



**Fig. 20.5** Views of the 3D model generated from the input image in Fig. 20.4. (From Liu [31], with permission)



**Fig. 20.6** Generating different expressions for the constructed face model. (From Liu [31], with permission)

## 20.3 Face Relighting

During the past several years, a lot of progress has been made on generating photo-realistic images of human faces under arbitrary lighting conditions [21, 26, 50, 53, 57, 64]. One class of method is inverse rendering [9, 10, 15, 36, 38, 63]. By capturing the lighting environment and recovering surface reflectance properties, one can generate photo-realistic rendering of objects including human faces under new lighting conditions. To recover the surface reflectance properties, one typically needs special setting and capturing equipment. Such systems are best suited for studio-like applications.

### 20.3.1 Face Relighting Using Ratio Images

Riklin-Raviv and Shashua [46] proposed a ratio-image technique to map one person's lighting condition to a different person. Given a face under two different lighting conditions, and another face under the first lighting condition, they used the color ratio (called the quotient image) to generate an image of the second face under the second lighting condition. For any given point on the face, let  $\rho$  denote its albedo, and  $\mathbf{n}$  its normal. Let  $E(\mathbf{n})$  and  $E'(\mathbf{n})$  be the irradiances under the two lighting conditions, respectively. Assuming a Lambertian reflectance model, the intensities of this point under the two lighting conditions are  $I = \rho E(\mathbf{n})$  and  $I' = \rho E'(\mathbf{n})$ . Given a different face, let  $\rho_1$  be its albedo. Then its intensities on the two lighting conditions are  $I_1 = \rho_1 E(\mathbf{n})$ , and  $I'_1 = \rho_1 E'(\mathbf{n})$ . Therefore, we have

$$\frac{I_1}{I} = \frac{I'_1}{I'}. \quad (20.3)$$

Thus,

$$I'_1 = I' \frac{I_1}{I}. \quad (20.4)$$

Equation (20.4) shows that one can obtain  $I'_1$  from  $I$ ,  $I'$ , and  $I_1$ . If we have one person's images under all possible lighting conditions and the second person's image under one of the lighting conditions, we can use (20.4) to generate the second person's images under all the other lighting conditions.

In many applications, we do not know in which lighting condition the second person's image is. Riklin-Raviv and Shashua [46] proposed that we use a database of images of different people under different lighting conditions. For any new person, if its albedo is “covered by” (formally called “rational span”, see Riklin-Raviv and Shashua [46] for details) the albedos of the people in the database, it is possible to figure out in which lighting condition the new image was.

### 20.3.2 Face Relighting from a Single Image

Researchers have developed face relighting techniques that do not require a database [21, 57, 59, 64]. Given a single image of a face, Wen et al. [59] first computed a special radiance environment map assuming known face geometry. For any point on the radiance environment map, its intensity is the irradiance at the normal direction multiplied by the average albedo of the face. In other words, the special radiance environment map is the irradiance map times a constant albedo. Zhang and Samaras [64] and Jiang et al. [21] proposed statistical approaches to recover the spherical harmonic basis images from the input image. A bootstrap step is required to obtain the statistical texture and shape information of human faces. To estimate the lighting, shape and albedo of a human face simultaneously from a single image, Wang et al. [57] used the 3D spherical harmonic morphable model (SHBMM), an integration of spherical harmonics into the morphable model framework. Thus, any face under arbitrary pose and illumination conditions can be represented simply by three low dimensional vectors: shape parameters, spherical harmonic basis parameters, and illumination coefficients. In this section, we describe the technique proposed by Wen et al. [59] in more detail.

Given a single image of a face, Wen et al. [59] computed the special radiance environment map using spherical harmonic basis functions [3, 45]. Accordingly, the irradiance can be approximated as a linear combination of nine spherical harmonic basis functions [3, 45].

$$E(\mathbf{n}) \approx \sum_{l \leq 2, -l \leq m \leq l} \hat{A}_l L_{lm} Y_{lm}(\mathbf{n}). \quad (20.5)$$

Wen et al. [59] also expanded the albedo function  $\rho(\mathbf{n})$  using spherical harmonics

$$\rho(\mathbf{n}) = \rho_{00} + \Psi(\mathbf{n}) \quad (20.6)$$

where  $\rho_{00}$  is the constant component, and  $\Psi(\mathbf{n})$  contains other higher order components.

From (20.5) and (20.6), we have

$$\rho(\mathbf{n}) E(\mathbf{n}) \approx \rho_{00} \sum_{l \leq 2, -l \leq m \leq l} \hat{A}_l L_{lm} Y_{lm}(\mathbf{n}) + \Psi(\mathbf{n}) \sum_{l \leq 2, -l \leq m \leq l} \hat{A}_l L_{lm} Y_{lm}(\mathbf{n}).$$

If we assume  $\Psi(\mathbf{n})$  does not have first four order ( $l = 1, 2, 3, 4$ ) components, the second term of the righthand side in (20.7) contains components with orders equal to or higher than 3 (see Wen et al. [59] for the explanation). Because of the orthogonality of the spherical harmonic basis, the nine coefficients of order  $l \leq 2$  estimated from  $\rho(\mathbf{n}) E(\mathbf{n})$  with a linear least-squares procedure are  $\rho_{00} \hat{A}_l L_{lm}$ , where ( $l \leq 2, -l \leq m \leq l$ ). Therefore, we obtain the radiance environment map with a reflectance coefficient equal to the average albedo of the surface.

Wen et al. [59] argued that human face skin approximately satisfies the above assumption, that is, it does not contain low frequency components other than the constant term.



**Fig. 20.7** Comparison of synthesized results and ground truth. The *top row* is the ground truth. The *bottom row* is the synthesized result, where the middle image is the input. (From Wen et al. [59], with permission)

By using a generic 3D face geometry, Wen et al. [59] set up the following system of equations:

$$I(\mathbf{n}) = \sum_{l \leq 2, -l \leq m \leq l} x_{lm} Y_{lm}(\mathbf{n}). \quad (20.7)$$

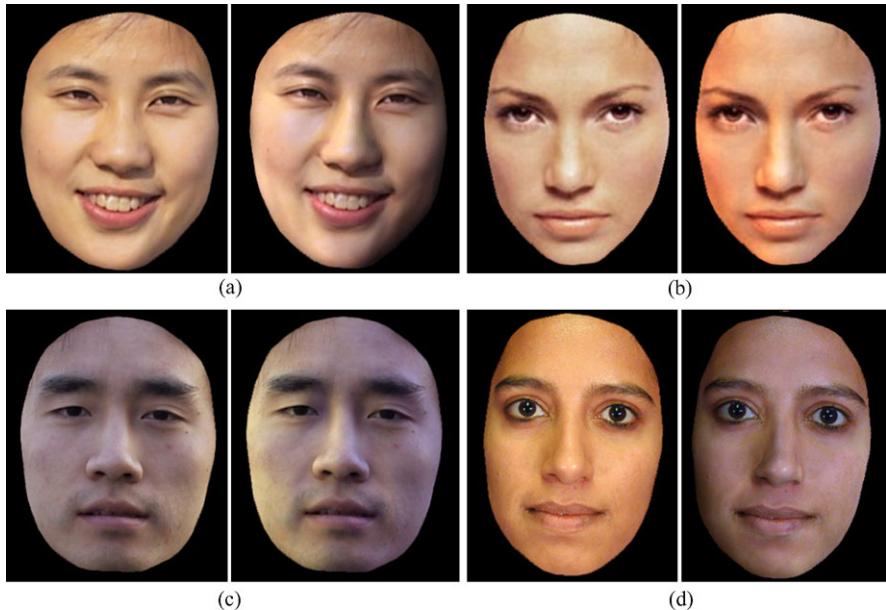
They used a linear least-squares procedure to solve the nine unknowns  $x_{lm}$ ,  $l \leq 2$ ,  $-l \leq m \leq l$ , thus obtaining the special radiance environment map.

One interesting application is that one can relight the face image when the environment rotates. For the purpose of explanation, let us imagine the face rotates while the environment is static. Given a point on the face, its intensity is  $I_f = \rho E(\mathbf{n})$ . The intensity of the corresponding point on the radiance environment map is  $I_s(\mathbf{n}) = \bar{\rho} E(\mathbf{n})$ , where  $\bar{\rho}$  is the average albedo of the face. After rotation, denote  $\mathbf{n}'$  to be the new normal. The new intensity on the face is  $I'_f = \rho E(\mathbf{n}')$ . The intensity on the radiance environment map corresponding to the  $\mathbf{n}'$  is  $I_s(\mathbf{n}') = \bar{\rho} E(\mathbf{n}')$ . Therefore,

$$I'_f = I_f \frac{I_s(\mathbf{n}')}{I_s(\mathbf{n})}. \quad (20.8)$$

The bottom row of Fig. 20.7 shows the relighting results. The input image is the one in the middle. The images at the top are the ground truth. We can see that the synthesized results match well with the ground truth images. There are some small differences mainly on the first and last images due to specular reflections. (According to Marschner et al. [37], human skin is almost Lambertian at small light incidence angles and has strong non-Lambertian scattering at higher angles.)

Another application is that one can modify the estimated spherical harmonic coefficients to generate radiance environment maps under the modified lighting con-



**Fig. 20.8** Lighting editing by modifying the spherical harmonics coefficients of the radiance environment map. The *left image* in each pair is the input image and the *right image* is the result after modifying the lighting. (From Wen et al. [59], with permission)

ditions. For each new radiance environment map, one can use the ratio-image technique (see (20.8)) to generate the face image under the new lighting condition. In this way, one can modify the lighting conditions of the face. In addition to lighting editing, this can also be used to generate training data with different lighting conditions for face detection or face recognition applications.

Figure 20.8 shows four examples of lighting editing by modifying the spherical harmonics coefficients. For each example, the left image is the input image, and the right image is the result after modifying the lighting. In example (a), lighting is changed to attach shadow to the person's left face. In example (b), the light on the person's right face is changed to be more reddish, and the light on her left face becomes slightly more bluish. In (c), the bright sunlight move from the person's left face to his right face. In (d), we attach shadow to the person's right face and change the light color as well.

### 20.3.3 Application to Face Recognition Under Varying Illumination

Qing et al. [44] used the face relighting technique as described in the previous section for face recognition under different lighting environments. For any given face

image under unknown illumination, they first applied the face relighting technique to generate a new image of the face under canonical illumination. Canonical illumination is the constant component of the spherical harmonics, which can be obtained by keeping only the constant coefficient ( $x_{00}$  in (20.7)) while setting the rest of the coefficients to zero. The ratio-image technique of (20.8) is used to generate the new image under canonical illumination.

Image matching is performed on the images under canonical illumination. Qing et al. [44] performed face recognition experiments with the PIE database [51]. They reported significant improvement of the recognition rate after using face relighting. The reader is referred to their article [44] for detailed experimental results.

## 20.4 Facial Expression Synthesis

In the past several years, facial expression synthesis has been an active research topic [7, 11, 24, 29, 35, 54, 56, 66]. Generally face expression synthesis techniques can be divided into three categories: physically based facial expression synthesis, morph-based facial expression synthesis, and expression mapping (also called performance-driven animation).

### 20.4.1 Physically Based Facial Expression Synthesis

One of the early physically based approaches is the work by Badler and Platt [2], who used a mass and spring model to simulate the skin. They introduced a set of muscles. Each muscle is attached to a number of vertices of the skin mesh. When the muscle contracts, it generates forces on the skin vertices, thereby deforming the skin mesh. A user generates facial expressions by controlling the muscle actions.

Waters [58] introduced two types of muscles: linear and sphincter. The lips and eye regions are better modeled by the sphincter muscles. To gain better control, they defined an influence zone for each muscle so the influence of a muscle diminishes as the vertices are farther away from the muscle attachment point.

Terzopoulos and Waters [55] extended Waters' model by introducing a three-layer facial tissue model. A fatty tissue layer is inserted between the muscle and the skin, providing more fine grain control over the skin deformation. This model was used by Lee et al. [25] to animate Cyberware scanned face meshes.

One problem with the physically based approaches is that it is difficult to generate natural looking facial expressions. There are many subtle skin movement, such as wrinkles and furrows, that are difficult to model with a mass-and-spring scheme.

### 20.4.2 Morph-Based Facial Expression Synthesis

Given a set of 2D or 3D expressions, one could blend these expressions to generate new expressions. This technique is called morphing or interpolation. This technique

was first reported in Parke's pioneer work [40]. Beier and Neely [4] developed a feature-based image morphing technique to blend 2D images of facial expressions. Bregler et al. [6] applied the morphing technique to mouth regions to generate lip-synch animations.

Pighin et al. [42] used the morphing technique on both the 3D meshes and texture images to generate 3D photorealistic facial expressions. They first used a multiview stereo technique to construct a set of 3D facial expression examples for a given person. Then they used the convex linear combination of the examples to generate new facial expressions. To gain local control, they allowed the user to specify an active region so the blending affects only the specified region. The advantage of this technique is that it generates 3D photorealistic facial expressions. The disadvantage is that the possible expressions this technique can generate is limited. The local control mechanism greatly enlarges the expression space, but it puts burdens on the user. The artifacts around the region boundaries may occur if the regions are not selected properly. Joshi et al. [22] developed a technique to automatically divide the face into subregions for local control. The region segmentation is based on the analysis of motion patterns for a set of example expressions.

#### ***20.4.3 Expression Mapping***

Expression mapping (also called performance-driven animation) has been a popular technique for generating realistic facial expressions. This technique applies to both 2D and 3D cases. Given an image of a person's neutral face and another image of the same person's face with an expression, the positions of the face features (e.g., eyes, eyebrows, mouths) on both images are located either manually or through some automatic method. The difference vector of the feature point positions is then added to a new face's feature positions to generate the new expression for that face through geometry-controlled image warping (we call it geometric warping) [4, 30, 61]. In the 3D case, the expressions are meshes, and the vertex positions are 3D vectors. Instead of image warping, one needs a mesh deformation procedure to deform the meshes based on the feature point motions [16].

Williams [60] developed a system to track the dots on a performer's face and map the motions to the target model. Litwinowicz and Williams [30] used this technique to animate images of cats and other creatures.

Because of its simplicity, the expression mapping technique has been widely used in practice. One great example is the FaceStation system developed by Eye-matic [19]. The system automatically tracks a person's facial features and maps his or her expression to the 3D model on the screen. It works in real time without any markers.

There has been much research done to improve the basic expression mapping technique. Pighin et al. [42] parameterized each person's expression space as a convex combination of a few basis expressions and proposed mapping one person's expression coefficients to those of another person. It requires that the two people

have the same number of basis expressions and that there is a correspondence between the two basis sets. This technique was extended by Pyun et al. [43]. Instead of using convex combination, Pyun et al. [43] proposed to the use of radial basis functions to parameterize the expression space.

Noh and Neumann [39] developed a technique to automatically find a correspondence between two face meshes based on a small number of user-specified correspondences. They also developed a new motion mapping technique. Instead of directly mapping the vertex difference, this technique adjusts both the direction and the magnitude of the motion vector based on the local geometries of the source and target model.

#### 20.4.3.1 Mapping Expression Details

Liu et al. [33] proposed a technique to map one person's facial expression details to a different person. Facial expression details are subtle changes in illumination and appearance due to skin deformations. The expression details are important visual cues, but they are difficult to model and synthesize. Given a person's neutral face image and an expression image, Liu et al. [33] observed that the illumination changes caused by the skin deformations can be extracted in a skin color independent manner using an expression ratio image (ERI). The ERI can then be applied to a different person's face image to generate the correct illumination changes caused by the skin deformation of that person's face.

Let  $I_a$  be person A's neutral face image, let  $I'_a$  be A's expression image. Given a point on the face, let  $\rho_a$  be its albedo, and let  $\mathbf{n}$  be its normal on the neutral face. Let  $\mathbf{n}'$  be the normal when the face makes the expression. By assuming Lambertian model, we have  $I_a = \rho_a E(\mathbf{n})$  and  $I'_a = \rho_a E(\mathbf{n}')$ . Taking the ratio, we have:

$$\frac{I'_a}{I_a} = \frac{E(\mathbf{n}')}{E(\mathbf{n})}. \quad (20.9)$$

Note that  $\frac{I'_a}{I_a}$  captures the illumination changes due to the changes in the surface normals; furthermore, it is independent of A's albedo.  $\frac{I'_a}{I_a}$  is called the expression ratio image. Let  $I_b$  be person B's neutral face image. Let  $\rho_b$  be its albedo. By assuming that B and A have similar surface normals on their corresponding points, we have  $I_b = \rho_b E(\mathbf{n})$ . Let  $I'_b$  be the image of B making the same expression as A; then  $I'_b = \rho_b E(\mathbf{n}')$ . Therefore,

$$\frac{I'_b}{I_b} = \frac{E(\mathbf{n}')}{E(\mathbf{n})} \quad (20.10)$$

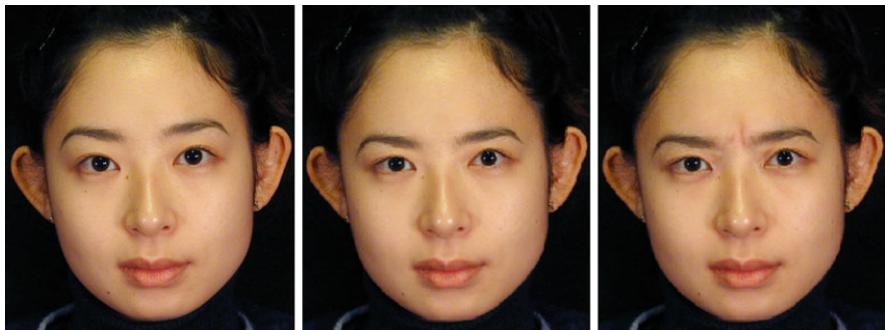
and so

$$I'_b = I_b \frac{I'_a}{I_a}. \quad (20.11)$$

Therefore, we can compute  $I'_b$  by multiplying  $I_b$  with the expression radio image.



**Fig. 20.9** Expression ratio image. *Left*: neutral face. *Middle*: expression face. *Right*: expression Ratio image. The ratios of the RGB components are converted to colors for display purpose. (From Liu et al. [33], with permission)



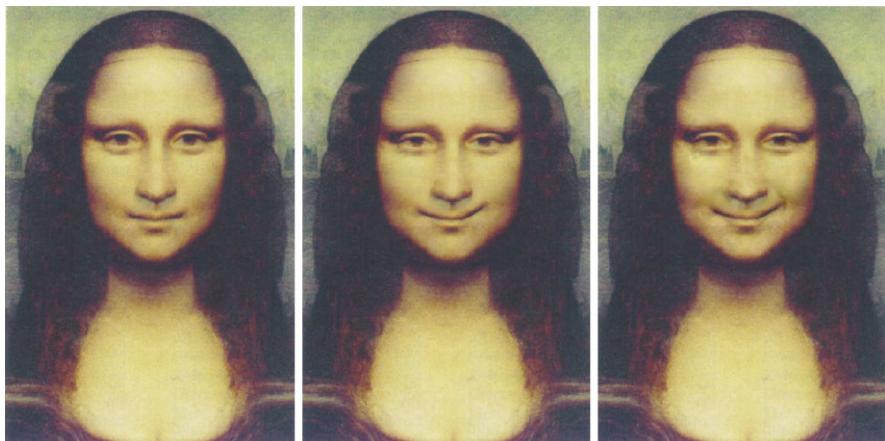
**Fig. 20.10** Mapping a thinking expression. *Left*: neutral face. *Middle*: result from geometric warping. *Right*: result from ERI. (From Liu et al. [33], with permission)

Figure 20.9 shows a male subject's thinking expression and the corresponding ERI. Figure 20.10 shows the result of mapping the thinking expression to a female subject. The image in the middle is the result of using traditional expression mapping. The image on the right is the result generated using the ERI technique. We can see that the wrinkles due to skin deformations between the eyebrows are mapped well to the female subject. The resulting expression is more convincing than the result from the traditional geometric warping. Figure 20.12 shows the result of mapping the smile expression (Fig. 20.11) to Mona Lisa. Figure 20.13 shows the result of mapping the smile expression to two statues.

#### 20.4.3.2 Geometry-Driven Expression Synthesis

One drawback of the ERI technique is that it requires the expression ratio image from the performer. Zhang et al. [65] proposed a technique that requires only the fea-

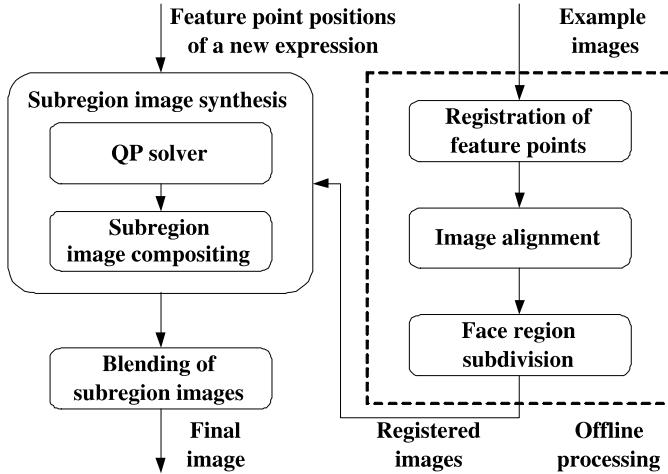
**Fig. 20.11** Smile expression used to map to other people's faces



**Fig. 20.12** Mapping a smile to Mona Lisa's face. *Left*: “neutral” face. *Middle*: result from geometric warping. *Right*: result from ERI. (From Liu et al. [33], with permission)



**Fig. 20.13** Mapping expressions to statues. **a** *Left*: original statue. *Right*: result from ERI. **b** *Left*: another statue. *Right*: result from ERI. (From Liu et al. [33], with permission)



**Fig. 20.14** Geometry-driven expression synthesis system. (From Zhang et al. [65], with permission)

ture point motions from the performer, as for traditional expression mapping. One first computes the desired feature point positions (geometry) for the target model, as for traditional expression mapping. Based on the desired feature point positions, the expression details for the target model are synthesized from examples.

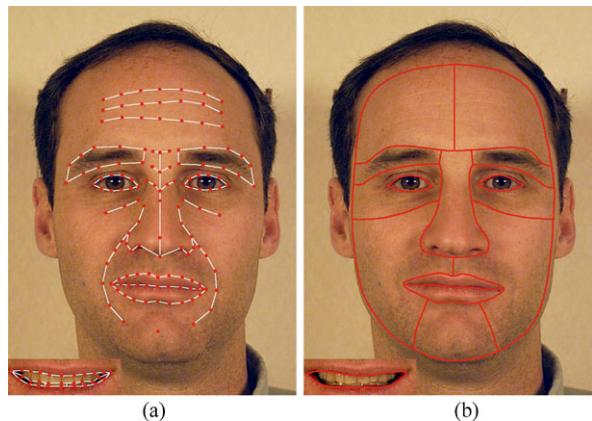
Let  $E_i = (G_i, I_i)$ ,  $i = 0, \dots, m$ , be the example expressions where  $G_i$  represents the geometry and  $I_i$  is the texture image (assuming that all the texture images  $I_i$  are pixel aligned). Let  $H(E_0, E_1, \dots, E_m)$  be the set of all possible convex combinations of these examples. Then

$$H(E_0, E_1, \dots, E_m) = \left\{ \left( \sum_{i=0}^m c_i G_i, \sum_{i=0}^m c_i I_i \right) \middle| \sum_{i=0}^m c_i = 1, c_i \geq 0, i = 0, \dots, m \right\}. \quad (20.12)$$

Note that each expression in the space  $H(E_0, E_1, \dots, E_m)$  has a geometric component  $G = \sum_{i=0}^m c_i G_i$  and a texture component  $I = \sum_{i=0}^m c_i I_i$ . Because the geometric component is much easier to obtain than the texture component, Zhang et al. [65] proposed using the geometric component to infer the texture component. Given the geometric component  $G$ , one can project  $G$  to the convex hull spanned by  $G_0, \dots, G_m$  and then use the resulting coefficients to composite the example images and obtain the desired texture image.

To increase the space of all possible expressions, they proposed subdividing the face into a number of subregions. For each subregion, they used the geometry associated with this subregion to compute the subregion texture image. The final expression is then obtained by blending these subregion images together. Figure 20.14 is an overview of their system. It consists of an offline processing unit and a run time unit. The example images are processed offline only once. At run time, the system takes as input the feature point positions of a new expression. For each sub-

**Fig. 20.15** **a** Feature points.  
**b** Face region subdivision.  
(From Zhang et al. [65], with permission)



region, they solve the quadratic programming problem of (20.12) using the interior point method. They then composite the example images in this subregion together to obtain the subregion image. Finally, they blend the subregion images together to produce the expression image.

Figure 20.15a shows the feature points they used by Zhang et al. [65]. Figure 20.15b shows the face region subdivision. From Fig. 20.15a, we can see that the number of feature points used for their synthesis system is large. The reason is that more feature points are better for the image alignment and for the quadratic programming solver. The problem is that some feature points, such as those on the forehead, are quite difficult to obtain from the performer, and they are person-dependent. Thus these feature points are not suited for expression mapping. To address this problem, they developed a motion propagation technique to infer feature point motions from a subset. Their basic idea was to learn how the rest of the feature points move from the examples. To have fine-grain control, they divided the face feature points into hierarchies and performed hierarchical principal component analysis on the example expressions.

There are three hierarchies. At hierarchy 0, they used a single feature point set that controls the global movement of the entire face. There are four feature point sets at hierarchy 1, each controlling the local movement of facial feature regions (left eye region, right eye region, nose region, mouth region). Each feature point set at hierarchy 2 controls details of the face regions, such as eyelid shape, lip line shape, and so on. There are 16 feature point sets at hierarchy 2. Some facial feature points belong to several sets at different hierarchies, and they are used as bridges between global and local movement of the face, so the vertex movements can be propagated from one hierarchy to another.

For each feature point set, Zhang et al. [65] computed the displacement of all the vertices belonging to this feature set for each example expression. They then performed principal component analysis on the vertex displacement vectors corresponding to the example expressions and generated a lower dimensional vector space. The hierarchical principal component analysis results are then used to propa-

gate vertex motions so that from the movement of a subset of feature points one can infer the most reasonable movement for the rest of the feature points.

Let  $v_1, v_2, \dots, v_n$  denote all the feature points on the face. Let  $\delta V$  denote the displacement vector of all the feature points. For any given  $\delta V$  and a feature point set  $F$  (the set of indexes of the feature points belonging to this feature point set), let  $\delta V(F)$  denote the subvector of those vertices that belong to  $F$ . Let  $\text{Proj}(\delta V, F)$  denote the projection of  $\delta V(F)$  into the subspace spanned by the principal components corresponding to  $F$ . In other words,  $\text{Proj}(\delta V, F)$  is the best approximation of  $\delta V(F)$  in the expression subspace. Given  $\delta V$  and  $\text{Proj}(\delta V, F)$ , let us say that  $\delta V$  is *updated* by  $\text{Proj}(\delta V, F)$  if for each vertex that belongs to  $F$  its displacement in  $\delta V$  has been replaced with its corresponding value in  $\text{Proj}(\delta V, F)$ .

The motion propagation algorithm takes as input the displacement vector for a subset of the feature points, say,  $\Delta v_{i_1}, \Delta v_{i_2}, \dots, \Delta v_{i_k}$ . Denote  $T = \{i_1, i_2, \dots, i_k\}$ . Below is a description of the motion propagation algorithm.

### **MotionPropagation**

Begin

    Set  $\delta V = 0$ .

    While (stop-criteria is not met) Do

        For each  $i_k \in T$ , set  $\delta V(i_k) = \Delta v_{i_k}$ .

    For all Feature point set  $F$ , set  $\text{hasBeenProcessed}(F)$  to be false.

        Find the feature point set  $F$  with the lowest hierarchy such that  $F \cap T \neq \emptyset$ .

        MotionPropagationFeaturePointSet( $F$ ).

    End

End

The function MotionPropagationFeaturePointSet is defined as follows:

### **MotionPropagationFeaturePointSet( $F^*$ )**

Begin

    Set  $h$  to be the hierarchy of  $F^*$ .

    If  $\text{hasBeenProcessed}(F^*)$  is true, return.

    Compute  $\text{Proj}(\delta V, F^*)$ .

    Update  $\delta V$  with  $\text{Proj}(\delta V, F^*)$ .

    Set  $\text{hasBeenProcessed}(F^*)$  to be true.

    For each feature set  $F$  belonging to hierarchy  $h - 1$  such that  $F \cap F^* \neq \emptyset$ .

        MotionPropagationFeaturePointSet( $F$ ).

    For each feature set  $F$  belonging to hierarchy  $h + 1$  such that  $F \cap F^* \neq \emptyset$ .

        MotionPropagationFeaturePointSet( $F$ ).

End

The algorithm initializes  $\delta V$  to a zero vector. At the first iteration, it sets  $\delta V(i_k)$  to be equal to the input displacement vector for vertex  $v_{i_k}$ . Then it finds the feature point set with the lowest hierarchy so it intersects with the input feature point set  $T$  and calls *MotionPropagationFeaturePointSet*. The function uses principal component analysis to infer the motions for the rest of the vertices in this feature point set. It then recursively calls *MotionPropagationFeaturePointSet* on other feature point



**Fig. 20.16** Example images of the male subject. (From Zhang et al. [65], with permission)

sets. At the end of the first iteration,  $\delta V$  contains the inferred displacement vectors for all the feature points. Note that for the vertex in  $T$  its inferred displacement vector may be different from the input displacement vector because of the principal component projection. At the second iteration,  $\delta V(i_k)$  is reset to the input displacement vector for all  $i_k \in T$ . The process repeats.

Figure 20.16 shows example images of a male subject, and Fig. 20.17 shows the results of mapping a female subject's expressions to this male subject.

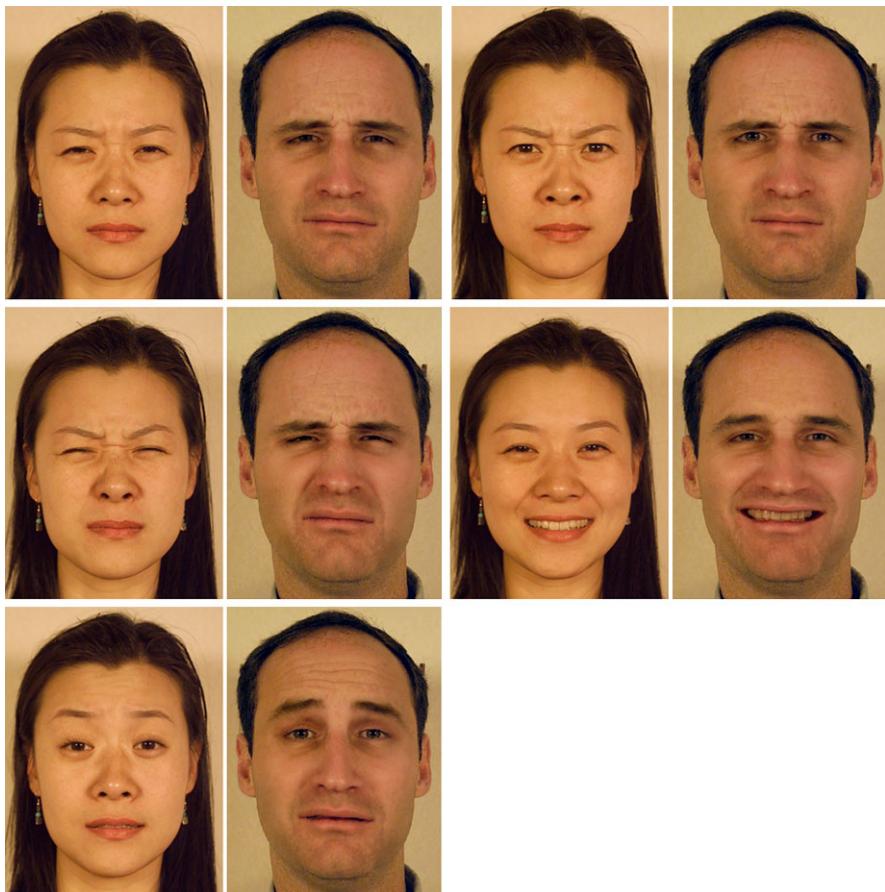
In addition to expression mapping, Zhang et al. [65] applied their techniques to expression editing. They developed an interactive expression editing system that allows a user to drag a face feature point, and the system interactively displays the resulting image with expression details. Figure 20.18 is a snapshot of their interface. The red dots are the feature points that the user can click and drag. Figure 20.19 shows some of expressions generated by the expression editing system.

## 20.5 Discussion

We have reviewed recent advances on face synthesis including face modeling, face relighting, and facial expression synthesis. There are many open problems that remain to be solved.

One problem is how to generate face models with fine geometric details. As discussed in Sect. 20.2, many 3D face modeling techniques use some type of model space to constrain the search, thereby improving the robustness. The resulting face models in general do not have the geometric details, such as creases and wrinkles. Geometric details are important visual cues for human perception. With geometric details, the models look more realistic; and for personalized face models, they look more recognizable to human users. Geometric details can potentially improve computer face recognition performance as well.

Another problem is how to handle non-Lambertian reflections. The reflection of human face skin is approximately specular when the angle between the view



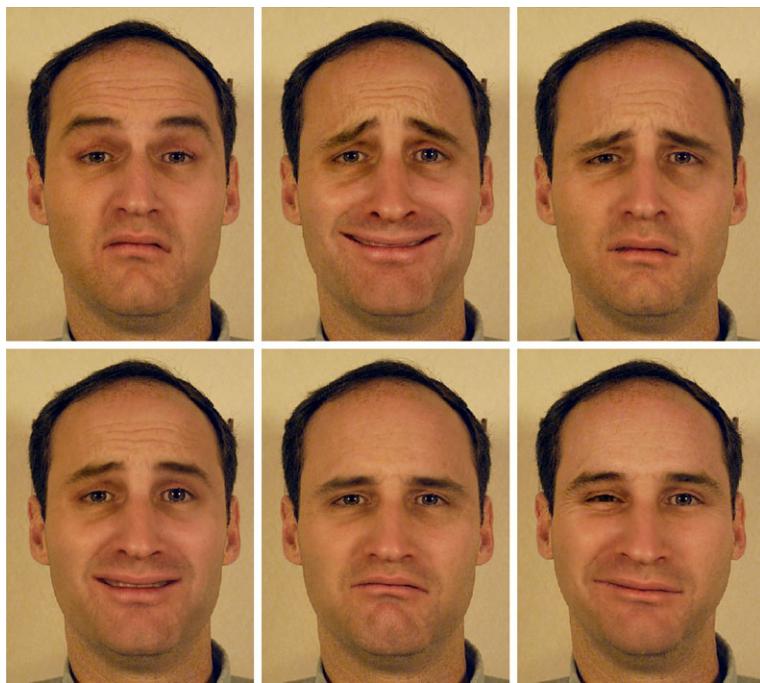
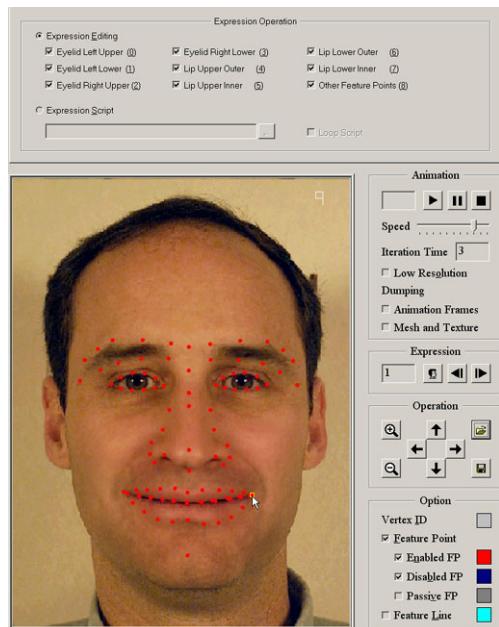
**Fig. 20.17** Results of the enhanced expression mapping. The expressions of the female subject are mapped to the male subject. (From Zhang et al. [65], with permission)

direction and lighting direction is close to  $90^\circ$ . Therefore, given any face image, it is likely that there are some points on the face whose reflection is not Lambertian. It is desirable to identify the non-Lambertian reflections and use different techniques for them during relighting.

How to handle facial expressions in face modeling and face relighting is another interesting problem. Can we reconstruct 3D face models from expression images? One would need a way to identify and undo the skin deformations caused by the expression. To apply face relighting techniques on expression face images, we would need to know the 3D geometry of the expression face to generate correct illumination for the areas with strong deformations.

One ultimate goal in face animation research is to be able to create face models that look and move just like a real human character. Not only do we need to synthe-

**Fig. 20.18** The expression editing interface. The red dots are the feature points which a user can click on and drag. (From Zhang et al. [65], with permission)



**Fig. 20.19** Expressions generated by the expression editing system. (From Zhang et al. [65], with permission)

size facial expression, we also need to synthesize the head gestures, eye gazes, hair, and the movements of lips, teeth, and tongue.

Face synthesis techniques can be potentially used for face detection and face recognition to handle different head poses, different lighting conditions, and different facial expressions. As we discussed earlier, some researchers have started applying some face synthesis techniques to face recognition [44, 48]. We believe that there are many more opportunities along this line, and that it is a direction worth exploring.

**Acknowledgements** We thank Ying-Li Tian for carefully reading our manuscripts and providing critical reviews. We also thank Zhengyou Zhang, Alex Acero, and Heung-Yeung Shum for their support.

## References

1. Akimoto, T., Suenaga, Y., Wallace, R.S.: Automatic 3d facial models. *IEEE Comput. Graph. Appl.* **13**(5), 16–22 (1993)
2. Badler, N., Platt, S.: Animating facial expressions. In: Computer Graphics, pp. 245–252. Siggraph, August 1981
3. Basri, R., Jacobs, D.: Lambertian reflectance and linear subspaces. In: Proc. ICCV'01, pp. 383–390 (2001)
4. Beier, T., Neely, S.: Feature-based image metamorphosis. In: Computer Graphics, pp. 35–42. Siggraph, July 1992
5. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: Computer Graphics, Annual Conference Series, pp. 187–194. Siggraph, August 1999
6. Bregler, C., Covell, M., Slaney, M.: Video rewrite: Driving visual speech with audio. In: Computer Graphics, pp. 353–360. Siggraph, August 1997
7. Chuang, E., Bregler, C.: Mood swings: expressive speech animation. *ACM Trans. Graph.* **24**(2), 331–347 (2005)
8. Dariush, B., Kang, S.B., Waters, K.: Spatiotemporal analysis of face profiles: Detection, segmentation, and registration. In: Proc. of the 3rd International Conference on Automatic Face and Gesture Recognition, April 1998, pp. 248–253. IEEE, New York (1998)
9. Debevec, P.E.: Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In: Computer Graphics, Annual Conference Series, pp. 189–198. Siggraph, July 1998
10. Debevec, P.E., Hawkins, T., Tchou, C., Duiker, H.-P., Sarokin, W., Sagar, M.: Acquiring the reflectance field of a human face. In: Computer Graphics, Annual Conference Series, pp. 145–156. Siggraph, July 2000
11. Deng, Z., Neumann, U.: Data-Driven 3D Facial Animation. Springer, Berlin (2007)
12. Dimitrijevic, M., Ilic, S., Fua, P.: Accurate face models from uncalibrated and ill-lit video sequences. In: Computer Vision and Pattern Recognition, vol. II, pp. 1034–1041 (2004)
13. Fua, P., Miccio, C.: From regular images to animated heads: A least squares approach. In: Eurographics of Computer Vision, pp. 188–202 (1996)
14. Fua, P., Miccio, C.: Animated heads from ordinary images: A least-squares approach. *Comput. Vis. Image Underst.* **75**(3), 247–259 (1999)
15. Georghiades, A., Belhumeur, P., Kriegman, D.: Illumination-based image synthesis: Creating novel images of human faces under differing pose and lighting. In: IEEE Workshop on Multi-View Modeling and Analysis of Visual Scenes, pp. 47–54 (1999)
16. Guenter, B., Grimm, C., Wood, D., Malvar, H., Pighin, F.: Making faces. In: Computer Graphics, Annual Conference Series, pp. 55–66. Siggraph, July 1998

17. Horn, B.K.: Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A* **4**(4), 629–642 (1987)
18. <http://www.digimask.com>
19. <http://www.eyematic.com>
20. Ip, H.H.S., Yin, L.: Constructing a 3d individualized head model from two orthogonal views. *Vis. Comput.* **12**, 254–266 (1996)
21. Jiang, X., Kong, Y., Huang, J., Zhao, R., Zhang, Y.: Learning from real images to model lighting variations for face images. In: European Conference on Computer Vision (ECCV'2008), vol. IV, pp. 284–297 (2008)
22. Joshi, P., Tien, W.C., Desbrun, M., Pighin, F.: Learning controls for blend shape based realistic facial animation. In: Proc. Symposium on Computer Animation (SCA'03), pp. 187–192, July 2003
23. Kemelmacher, I., Basri, R.: Molding face shapes by example. In: European Conference on Computer Vision, pp. I:277–288 (2006)
24. Lau, M., Chai, J., Xu, Y.-Q., Shum, H.-Y.: Face poser: Interactive modeling of 3D facial expressions using facial priors. *ACM Trans. Graph.* **29**(1), 1–17 (2009)
25. Lee, Y., Terzopoulos, D., Waters, K.: Realistic modeling for facial animation. In: Computer Graphics, pp. 55–62. Siggraph, August 1995
26. Lee, J., Moghaddam, B., Pfister, H., Machiraju, R.: A bilinear illumination model for robust face recognition. In: International Conference on Computer Vision (ICCV'05), vol. II, pp. 1177–1184 (2005)
27. Lei, Z., Bai, Q., He, R., Li, S.: Face shape recovery from a single image using cca mapping between tensor spaces. In: Computer Vision and Pattern Recognition (2008)
28. Li, S.Z., Gu, L.: Real-time multi-view face detection, tracking, pose estimation, alignment, and recognition. In: IEEE Conf. on Computer Vision and Pattern Recognition Demo Summary (2001)
29. Li, H., Weise, T., Pauly, M.: Example-based facial rigging. In: ACM Transactions on Graphics. Siggraph, July 2010
30. Litwinowicz, P., Williams, L.: Animating images with drawings. In: Computer Graphics, pp. 235–242. Siggraph, August 1994
31. Liu, Z.: A fully automatic system to model faces from a single image. Microsoft Research Technical Report: MST-TR-2003-55 (2003)
32. Liu, Z., Zhang, Z.: Robust head motion computation by taking advantage of physical properties. In: IEEE Workshop on Human Motion (HUMO), pp. 73–77 (2000)
33. Liu, Z., Shan, Y., Zhang, Z.: Expressive expression mapping with ratio images. In: Computer Graphics, Annual Conference Series, pp. 271–276. Siggraph, August 2001
34. Liu, Z., Zhang, Z., Jacobs, C., Cohen, M.: Rapid modeling of animated faces from video. *J. Vis. Comput. Animat.* **12**(4), 227–240 (2001)
35. Ma, W.-C., Jones, A., Chiang, J.-Y., Hawkins, T., Frederiksen, S., Peers, P., Vukovic, M., Ouhyoung, M., Debevec, P.: Facial performance synthesis using deformation-driven polynomial displacement maps. In: SIGGRAPH Asia '08, pp. 1–10. ACM, New York (2008)
36. Marschner, S.R., Greenberg, D.P.: Inverse lighting for photography. In: IST/SID Fifth Color Imaging Conference, November 1997
37. Marschner, S.R., Westin, S., Lafontaine, E., Torance, K., Greenberg, D.: Image-based brdf measurement including human skin. In: Rendering Techniques (1999)
38. Marschner, S.R., Guenter, B., Raghupathy, S.: Modeling and rendering for realistic facial animation. In: Rendering Techniques, pp. 231–242. Springer, New York (2000)
39. Noh, J.J., Neumann, U.: Expression cloning. In: Computer Graphics, Annual Conference Series, pp. 277–288. Siggraph, August 2001
40. Parke, F.I.: Computer generated animation of faces. In: ACM National Conference, November 1972
41. Patel, A., Smith, W.: 3d morphable face models revisited. In: Computer Vision and Pattern Recognition (CVPR'09), pp. 1327–1334 (2009)

42. Pighin, F., Hecker, J., Lischinski, D., Szeliski, R., Salesin, D.H.: Synthesizing realistic facial expressions from photographs. In: Computer Graphics, Annual Conference Series, pp. 75–84. Siggraph, July 1998
43. Pyun, H., Kim, Y., Chae, W., Kang, H.W., Shin, S.Y.: An example-based approach for facial expression cloning. In: Proc. Symposium on Computer Animation (SCA'03), pp. 167–176, July 2003
44. Qing, L., Shan, S., Gao, W.: Face recognition with harmonic de-lighting. In: Asian Conference on Computer Vision (ACCV), January 2004
45. Ramamoorthi, R., Hanrahan, P.: An efficient representation for irradiance environment maps. In: Proc. SIGGRAPH 2001, pp. 497–500, August 2001
46. Riklin-Raviv, T., Shashua, A.: The quotient image: Class based re-rendering and recognition with varying illuminations. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 566–571, June 1999
47. Romdhani, S., Vetter, T.: Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In: Computer Vision and Pattern Recognition, vol. II, pp. 986–993 (2005)
48. Romdhani, S., Blanz, V., Vetter, T.: Face identification by fitting a 3d morphable model using linear shape and texture error functions. In: European Conference on Computer Vision (ECCV'2002), vol. IV, pp. 3–19 (2002)
49. Shan, Y., Liu, Z., Zhang, Z.: Modle-based bundle adjustment with application to face modeling. In: International Conference on Computer Vision (ICCV'01), vol. II, pp. 644–651 (2001)
50. Shim, H., Luo, J., Chen, T.: A subspace model-based approach to face relighting under unknown lighting and poses. *IEEE Trans. Image Process.* **17**(8), 1331–1341 (2008)
51. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination, and expression (PIE) database. In: Face and Gesture'02 (2002)
52. Smith, W., Hancock, E.: Recovering facial shape using a statistical model of surface normal direction. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(12), 1914–1930 (2006)
53. Stoschek, A.: Image-based re-rendering of faces for continuous pose and illumination directions. In: Computer Vision and Pattern Recognition (CVPR'00), pp. 582–587 (2000)
54. Sumner, R.W., Popović, J.: Deformation transfer for triangle meshes. In: Computer Graphics, Annual Conference Series, pp. 399–405. Siggraph (2004)
55. Terzopoulos, D., Waters, K.: Physically-based facial modeling and animation. *J. Vis. Comput. Animat.* **1**(4), 73–80 (1990)
56. Wang, Y., Huang, X., Lee, C.-S., Zhang, S., Li, Z., Samaras, D., Metaxas, D., Elgammal, A., Huang, P.: High resolution acquisition, learning and transfer of dynamic 3-d facial expressions. *Comput. Graph. Forum* **23**, 677–686 (2004)
57. Wang, Y., Zhang, L., Liu, Z., Hua, G., Wen, Z., Zhang, Z., Samaras, D.: Face re-lighting from a single image under arbitrary unknown lighting conditions. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(11), 1968–1984 (2009)
58. Waters, K.: A muscle model for animating three-dimensional facial expression. *Comput. Graph.* **22**(4), 17–24 (1987)
59. Wen, Z., Liu, Z., Huang, T.S.: Face relighting with radiance environment maps. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. II, pp. 158–165, June 2003
60. Williams, L.: Performance-driven facial animation. In: Computer Graphics, pp. 235–242. Siggraph, August 1990
61. Wolberg, G.: Digital Image Warping. IEEE Computer Society Press, Los Alamitos (1990)
62. Yan, S., Li, M., Zhang, H., Cheng, Q.: Ranking prior likelihood distributions for Bayesian shape localization framework. In: International Conference on Computer Vision (ICCV'03) (2003)
63. Yu, Y., Debevec, P.E., Malik, J., Hawkins, T.: Inverse global illumination: Recovering reflectance models of real scenes from photographs. In: Proc. SIGGRAPH 99, pp. 215–224, July 1999
64. Zhang, L., Samaras, D.: Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(3), 351–363 (2006)

65. Zhang, Q., Liu, Z., Guo, B., Shum, H.: Geometry-driven photorealistic facial expression synthesis. In: Proc. Symposium on Computer Animation (SCA'03), pp. 177–186, July 2003
66. Zhang, Q., Liu, Z., Guo, B., Terzopoulos, D., Shum, H.-Y.: Geometry-driven photorealistic facial expression synthesis. *IEEE Trans. Vis. Comput. Graph.* **12**(1), 48–60 (2006)
67. Zhou, S.K., Aggarwal, G., Chellappa, R., Jacobs, D.: Appearance characterization of linear Lambertian objects, generalized photometric stereo and illumination-invariant face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(2), 330–245 (2007)



**Part III**

**Performance Evaluation: Machines  
and Humans**



# Chapter 21

## Evaluation Methods in Face Recognition

P. Jonathon Phillips, Patrick Grother, and Ross Micheals

### 21.1 Introduction

Face recognition from still frontal images has made great strides over the last twenty years. Over this period, error rates have decreased by three orders of magnitude when recognizing frontal face in still images taken with consistent controlled illumination in an environment similar to a studio [5, 10, 15, 17–22]. Under these conditions, error rates below 1% at a false accept rate of 1 in 1000 were reported in the Face Recognition Vendor Test<sup>1</sup> (FRVT) 2006 and the Multiple Biometric Evaluation (MBE) 2010 [10, 21].

The heart of designing and conducting evaluations is the experimental protocol. The protocol states how an evaluation is to be conducted and how the results are to be computed. In this chapter, we concentrate on describing the FERET and FRVT 2002 protocols. The FRVT 2002 evaluation protocol is based on the FERET evaluation protocols. The FRVT 2002 protocol is designed for biometric evaluations in general, not just for evaluating face recognition algorithms. These two evaluation protocols served as a basis for the FRVT 2006 and MBE 2010 evaluations.

The FRVT 2002 protocol was designed to allow for computing a wide range of performance statistics. This includes the standard performance tasks of open-set and closed-set identification, and verification. It also allows for resampling techniques,

---

<sup>1</sup>Performance results in this chapter are labeled by the test participants. The identification of any commercial product or trade name does not imply endorsement or recommendation by the National Institute of Standards and Technology.

P.J. Phillips (✉) · P. Grother · R. Micheals

National Institute of Standards and Technology, Gaithersburg, MD 20899, USA

e-mail: [jonathon@nist.gov](mailto:jonathon@nist.gov)

P. Grother

e-mail: [pgrother@nist.gov](mailto:pgrother@nist.gov)

R. Micheals

e-mail: [rossm@nist.gov](mailto:rossm@nist.gov)

similarity score normalization, measuring the variability of performance statistics, and covariate analysis [1–4, 9, 11, 14].

## 21.2 Performance Measures

In face recognition and biometrics, performance is reported on three standard tasks: verification, open-set and closed-set identification. Each task has its own set of performance measures. All three tasks are closely related, with open-set identification being the general case.

A biometric system works by processing biometric samples. *Biometric samples* are recordings of a feature of a person that allows that person to be recognized. Examples of biometric samples are facial images and fingerprints. A biometric sample can consist of multiple recordings, for example, five images of a person acquired at the same time or a facial image and a fingerprint.

Computing performance requires three sets of images. The first is a *gallery*  $\mathcal{G}$ , which contains biometric samples of the people known to a system. The other two are *probe sets*. A *probe* is a biometric sample that is presented to the system for recognition, where recognition can be verification or identification. The first probe set is  $\mathcal{P}_G$  that contains biometric samples of people in a gallery (these samples are different from the samples in the gallery). The other probe set is  $\mathcal{P}_N$ , which contains biometric samples of people that are not in a gallery.

Closed-set identification is the classic performance measure used in the automatic face recognition community, where it is known as identification. In closed-set identification, the basic question asked is: whose face is this? This question is meaningful for closed-set identification, since the biometric sample in a probe is always someone in the gallery. The general case of closed-set identification is open-set identification.

In open-set identification, the person in the probe does not have to be somebody in the gallery. In open-set identification, the basic question asked is: do we know this face? In open-set identification, a system has to decide if the probe contains an image of a person in the gallery. If a system decides that a person is in the gallery, then the system has to report the identity of the person. When the gallery is small, open-set identification can be referred to as a watch list task. When the gallery is large, then open-set identification models mugshot book searching and the operation of large automatic fingerprint identification systems (AFIS as they are sometimes called). Open-set and closed-set identification are sometimes referred to as 1 to many matching or 1:N matching. Depending on the context and author, 1 to many matching or 1:N matching can refer to either open-set or closed-set identification.

In a verification task, a person presents a biometrics sample to a system and claims an identity. The system has to decide if the biometric sample belongs to the claimed identity. In verification, the basic question asked is: is this person who he claims to be? Verification is also called authentication or 1 to 1 matching.

### 21.2.1 Open-Set Identification

Open-set identification is the general case task, with verification and closed-set identification being special cases. In the open-set identification task, a system determines if a probe  $p_j$  corresponds to a person in a gallery  $\mathcal{G}$ . If the probe is determined to be in the gallery, then the algorithm identifies the person in the probe.

A gallery  $\mathcal{G}$  consists of a set of biometric samples  $\{g_1, \dots, g_{|\mathcal{G}|}\}$ , with one biometric sample per person. When a probe  $p_j$  is presented to a system, it is compared to the entire gallery. The comparison between a probe  $p_j$  and each gallery biometric sample  $g_i$  produces a similarity score  $s_{ij}$ . Larger similarity scores indicate that two biometric samples are more similar. (A distance measure between biometric samples can be converted to a similarity score by negating the distance measure.) A similarity score  $s_{ij}$  is a *match score* if  $g_i$  and  $p_j$  are biometric samples of the same person. A similarity score  $s_{ij}$  is a *nonmatch score* if  $g_i$  and  $p_j$  are biometric samples of the different people. If  $p_j$  is a biometric sample of a person in the gallery, then let  $g^*$  be its unique match in the gallery. The similarity score between  $p_j$  and  $g^*$  is denoted by  $s_{*j}$ . The function  $\text{id}()$  returns the identity of a biometric sample, with  $\text{id}(p_j) = \text{id}(g^*)$ . For identification, all similarity scores between a probe  $p_j$  and a gallery are examined and sorted. A probe  $p_j$  has rank  $n$  if  $s_{*j}$  is the  $n$ th largest similarity score. This is denoted by  $\text{rank}(p_j) = n$ . Rank 1 is sometimes called the top match.

Performance for open-set identification is characterized by two performance statistics: detection and identification rate, and false alarm rate. We will first look at the case where the identity of a probe is someone in the gallery; that is,  $p_j \in \mathcal{P}_{\mathcal{G}}$ . A probe is detected and identified if the probe is correctly identified and the correct match score is above an operating threshold  $\tau$ . These conditions formally correspond to:

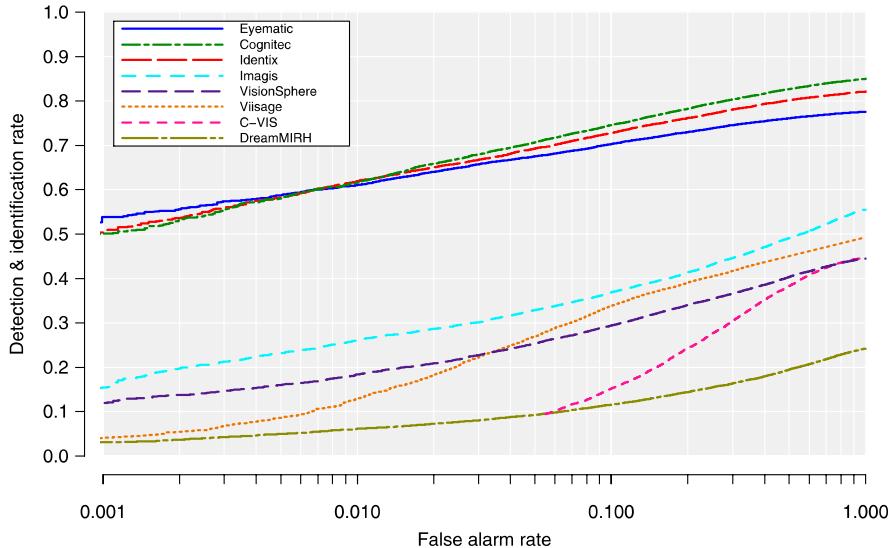
- $\text{rank}(p_j) = 1$  and
- $s_{*j} \geq \tau$  for the similarity match where  $\text{id}(p_j) = \text{id}(g^*)$ ,

for operating threshold  $\tau$ . The detection and identification rate is the fraction of probes in  $\mathcal{P}_{\mathcal{G}}$  that are correctly detected and identified. The detection and identification rate is a function of the operating threshold  $\tau$ . The detection and identification rate at threshold  $\tau$  is

$$P_{\text{DI}}(\tau, 1) = \frac{|\{p_j : p_j \in \mathcal{P}_{\mathcal{G}}, \text{rank}(p_j) = 1, \text{ and } s_{*j} \geq \tau\}|}{|\mathcal{P}_{\mathcal{G}}|}. \quad (21.1)$$

The second performance statistic is false alarm rate. The false alarm rate provides performance when a probe is not of someone in the gallery; that is,  $p_j \in \mathcal{P}_{\mathcal{N}}$ . This type of probe is also referred to as an imposter. A false alarm occurs when the top match score for an imposter is above the operating threshold. Formally, a false alarm occurs when

$$\max_i s_{ij} \geq \tau.$$



**Fig. 21.1** Open-set identification performance reported on an ROC. The gallery consisted of 800 individuals. Performance is for FRVT 2002 and is explained in Sect. 21.5.1

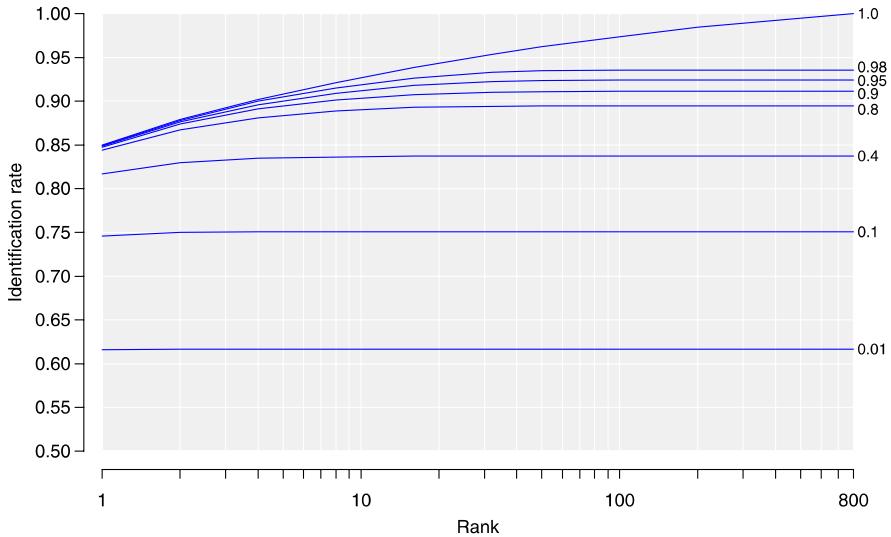
The false alarm rate is the fraction of probes in  $p_j \in \mathcal{P}_{\mathcal{N}}$  that are alarms. This is computed by

$$P_{FA}(\tau) = \frac{|\{p_j : p_j \in \mathcal{P}_{\mathcal{N}}, \max_i s_{ij} \geq \tau\}|}{|\mathcal{P}_{\mathcal{N}}|}. \quad (21.2)$$

The ideal system would have a detection and identification rate of 1.0 and a false alarm rate of 0.0. All people in the probe are detected and identified and there are no false alarms. However, in real-world systems there is a trade-off between the detection and identification rate, and the false alarm rate. By changing the operating threshold, the performance rates change. Increasing the operating threshold lowers both the false alarm rate and the detection and identification rate. Both these performance rates cannot be maximized simultaneously; there is a trade-off between them. This trade-off is shown on a receiver operating characteristic (ROC). An example of an ROC is shown in Fig. 21.1. The horizontal axis is the false alarm rate (scaled logarithmically). A logarithmic axis emphasizes low false alarm rates, which are the operating points of interest in applications. The vertical axis is the detection and identification rate. When reporting performance, the size of the gallery, and both probe sets need to be stated.

In the general open-set identification case, a system examines the top  $n$  matches between a probe and a gallery. A probe of a person in the gallery is detected and identified at rank  $n$  if the probe is of rank  $n$  or less and the correct match is above the operating threshold. These conditions formally correspond to:

- $\text{rank}(p_j) \leq n$  and
- $s_{*j} \geq \tau$  for the similarity match where  $\text{id}(p_j) = \text{id}(g^*)$ .



**Fig. 21.2** Open-set identification performance as a function of rank for eight false alarm rates. The false alarm rate for each curve is on the right side of the graph. The gallery size is 800. The *top curve* is for a false alarm rate of 1.0

The detection and identification rate at rank  $n$  is the fraction of probes in  $\mathcal{P}_G$  who are correctly detected and identified at rank  $n$ . The detection and identification rate at rank  $n$  and threshold  $\tau$  is

$$P_{DI}(\tau, n) = \frac{|\{p_j : p_j \in \mathcal{P}_G, \text{rank}(p_j) \leq n, \text{and } s_{*j} \geq \tau\}|}{|\mathcal{P}_G|}. \quad (21.3)$$

The computation of  $P_{FA}(\tau)$  at rank  $n$  is the same as in the rank 1 case.

The general open-set identification performance can be plotted along three axes: detection and identification rate, false alarm rate, and rank. The performance of a system is represented as a surface in this three-dimensional parameter space. Instead of plotting the complete open-set identification performance as a surface, performance is usually plotted as two dimensional slices. One example is Fig. 21.1 where rank is held constant at 1, and the trade-off between the detection and identification rate and false alarm rate is shown. Figure 21.2 presents another format for reporting open-set identification performance. The vertical axis is the detection and identification rate and the horizontal axis is rank on a logarithmic scale. Each curve is the performance of the same system at a different false alarm rate.

The method presented above for computing rank, assumes that all the similarity scores between a probe and a gallery are unique. Special care needs to be taken if there are multiple similarity scores with the same value, we will refer to these as tied similarity scores. There are three methods for handling tie scores: *optimistic*, *pessimistic*, and *average* rank. The optimistic rank is the number of similarity scores strictly greater than ( $>$ )  $s_{*j}$  plus one. In this case, we assign a probe the highest possible rank. The pessimistic rank is the number of similarity scores greater than

or equal to ( $\geq$ )  $s_{*j}$  plus one. In this case, we assign a probe the lowest possible rank. The average rank is the average of the optimistic and pessimistic ranks. In the FRVT 2002 protocol, the average rank was used. Resolving ties with the optimistic rank can lead to strange pathologies. For example, if a similarity matrix consisted of one value, then the identification rate reported with the optimistic rank would be 100%.

### 21.2.2 Verification

Verification or authentication follows the following operational model. In a typical verification task, a person presents his biometric sample to a system and claims to be a person in the system's gallery. The presented biometric sample is a probe. The system then compares the probe with the stored biometric sample of the person in the gallery. The comparison produces a similarity score. The system accepts the identity claim if the similarity score is greater than the system's operating threshold. The operational threshold is determined by the applications, and different applications will have different operational thresholds. Otherwise, the system rejects the claim.

There are two standard protocols for computing verification performance. The first is the round robin methods. In the round robin protocol, both of the probe set  $\mathcal{P}_g$  and  $\mathcal{P}_N$  are the same set and will be referred to as the probe set  $\mathcal{P}$ . All scores between gallery and probe set samples are computed. All match scores between the gallery and probe set are used to compute the verification rate, and all non-match scores are used to compute the false accept rate. Formal, for the round robin method, the verification rate is computed by

$$P_V(\tau) = \frac{|\{p_j : s_{ij} \geq \tau, \text{id}(g_i) = \text{id}(p_j)\}|}{|\mathcal{P}|}, \quad (21.4)$$

and the false accept rate is computed by

$$P_{FA}(\tau) = \frac{|\{s_{ij} : s_{ij} \geq \tau \text{ and } \text{id}(g_i) \neq \text{id}(p_j)\}|}{(|\mathcal{P}| - 1)|\mathcal{G}|}. \quad (21.5)$$

One complaint with the round robin protocol is that probes are used to generate both verification and false accept rates. There is a concern that this does not adequately model the situation where false identity claims are generated by people not in the gallery. The true imposter protocol addresses this concern. In the true imposter protocol, performance is computed from two probe sets,  $\mathcal{P}_g$  and  $\mathcal{P}_N$ . The verification rate is computed from the match scores between a gallery and  $\mathcal{P}_g$ . The number of match scores is the size of  $\mathcal{P}_g$ . The false alarm rate is computed from all non-match scores between the gallery and  $\mathcal{P}_N$ . These non-match scores are called true imposters because people in  $\mathcal{P}_N$  are not in the gallery. The number of non-match scores is  $|\mathcal{P}_N||\mathcal{G}|$ . Formal, for the true impostor method, the verification

rate is computed by

$$P_V(\tau) = \frac{|\{p_j : p_j \in \mathcal{P}_{\mathcal{G}}, s_{ij} \geq \tau, \text{id}(g_i) = \text{id}(p_j)\}|}{|\mathcal{P}_{\mathcal{G}}|}, \quad (21.6)$$

and the false accept rate is computed by

$$P_{FA}(\tau) = \frac{|\{s_{ij} : p_j \in \mathcal{P}_{\mathcal{N}}, s_{ij} \geq \tau\}|}{|\mathcal{P}_{\mathcal{N}}||\mathcal{G}|}. \quad (21.7)$$

### 21.2.3 Closed-Set Identification

Performance on the closed-set identification task is the classic performance statistic in face recognition. In closed-set identification, the question is not always “is the top match correct?,” but rather “is the correct answer in the top  $n$  matches?”

The first step in computing closed-set performance is to sort the similarity scores between  $p_j$  and gallery  $\mathcal{G}$ , and compute the rank( $p_j$ ). The identification rate for rank  $n$ ,  $P_I(n)$ , is the fraction of probes at rank  $n$  or lower. For rank  $n$ , let

$$C(n) = |\{p_j : \text{rank}(p_j) \leq n\}|, \quad (21.8)$$

be the cumulative count of the number of probes of rank  $n$  or less. The identification rate at rank  $n$  is

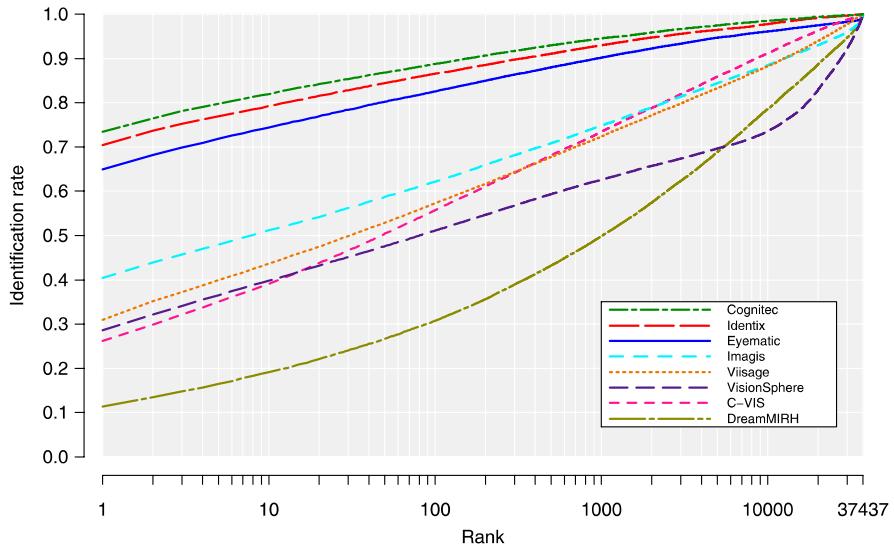
$$P_I(n) = \frac{|C(n)|}{|\mathcal{P}_{\mathcal{G}}|}. \quad (21.9)$$

The functions  $C(n)$  and  $P_I(n)$  are nondecreasing in  $n$ . The identification rate at rank 1,  $P_I(1)$ , is also called the correct identification rate, or top match rate.

Closed-set identification performance is reported on a cumulative match characteristic (CMC). A CMC plots  $P_I(n)$  as a function of rank  $n$ . Figure 21.3 shows a CMC. The horizontal axis is rank on a logarithmic scale and the vertical axis is  $P_I(n)$ .

Closed-set identification performance is most often summarized with rank one performance, the other points such as rank 5, 10, or 20 are commonly used. The strength and weakness of the CMC is its dependence on gallery size,  $|\mathcal{G}|$ . To show the effect of gallery size on performance, rank 1 performance versus gallery size is plotted. To remove the effect of gallery size, one can plot identification performance as a percentage of rank; that is, performance when the correct answer is in the top 10%.

Closed-set identification is a special case of open-set identification where the probe set  $\mathcal{P}_{\mathcal{N}}$  is empty and the operating threshold  $\tau = -\infty$ . An operating threshold of  $\tau = -\infty$  corresponds to a false alarm rate of 1.0. This means that  $s_{*j} \geq \tau$  for all match scores and all match scores are reported as alarms. Thus, for any  $n$ ,  $P_{DI}(-\infty, n) = P_I(n)$ . The curve in Fig. 21.2 with a false alarm rate of 1.0 (top curve) is the CMC for the closed-set version of this experiment. The CMC for an



**Fig. 21.3** Identification performance reported on a CMC. The gallery consisted of one image of 37437 individuals. The probe set consisted of two images for each of the 37437 individuals in the gallery. Performance is for FRVT 2002 and is explained in Sect. 21.5.1

open-set experiment is the closed-set identification performance computed in the open-set gallery  $\mathcal{G}$  and probe set  $\mathcal{P}_{\mathcal{G}}$ . In Fig. 21.2, it is interesting to note the difference between the CMC and the performance curve with a false alarm rate of 0.98. This shows that there are a reasonable number of match scores with a low similarity score.

#### 21.2.4 Normalization

The FRVT 2002 protocol introduced similarity score normalization procedures to biometric evaluations. Normalization is a post processing function  $f$  that adjusts similarity scores based on a specific probe. A normalization function is  $f : R^{|\mathcal{G}|} \rightarrow R^{|\mathcal{G}|}$ . The input to a normalization function is a vector  $\mathbf{s} = (s_{1j}, \dots, s_{|\mathcal{G}|j})$  of all similarity scores between a probe  $p_j$  and a gallery  $\mathcal{G}$ . The output is a vector  $\widehat{\mathbf{s}}$  of length  $|\mathcal{G}|$  which is a new set of normalized similarity scores  $\widehat{\mathbf{s}} = (\widehat{s}_{1j}, \dots, \widehat{s}_{|\mathcal{G}|j})$  between a probe  $p_j$  and a gallery  $\mathcal{G}$ . The normalization function attempts to adjust for variations among probes and to emphasize differences among the gallery signatures. The final performance scores are computed from the normalized similarity scores. An example of a normalization function is

$$\widehat{s}_{ij} = \frac{s_{ij} - \text{mean}(\mathbf{s})}{\text{sd}(\mathbf{s})},$$

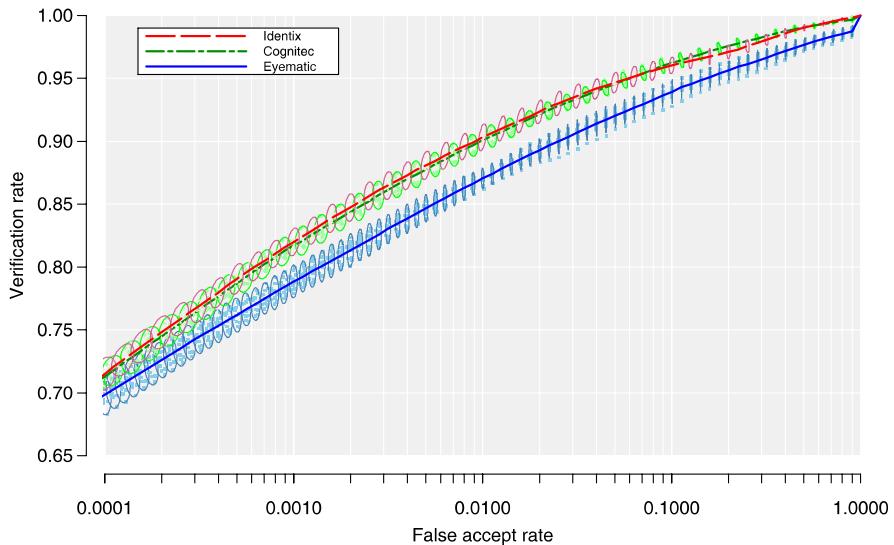
where  $\text{mean}(\mathbf{s})$  is the sample mean of the components of  $\mathbf{s}$  and  $\text{sd}(\mathbf{s})$  is the sample standard deviation of the components of  $\mathbf{s}$ . FRVT 2002 and the HumanID Gait Challenge problem demonstrated the effectiveness of normalization for verification [23].

If the gallery changes, then similarity scores need to be normalized again. This has implications for scoring techniques that require performance on multiple galleries. Traditionally, verification has been referred to as “1 to 1” matching. This is because, in verification, one probe is matched with one gallery signature. However, normalization requires that a probe be compared with a gallery set. When normalization is applied, is verification still “1 to 1” matching?

### 21.2.5 Variability

The variance of performance statistics in biometrics is an important but often overlooked subject in biometrics. We will look at variations in verification performance. The first is how performance varies with different galleries. This models the performance of a system that is installed at different locations. The second is how performance varies for different classes of probes. For example, what is the difference in performance for male and female probes? Each combination of the gallery and probe sets generates a different ROC. To study the variation, it is necessary to combine results over a set of ROCs. One method of combining results is to measure the variation of the verification rate for each false alarm rate. This models the situation where one can readjust the operating threshold for each gallery or probe set. For many applications, this is not feasible or desirable. However, this is an appropriate technique for combining ROCs from multiple systems because it is not possible to set uniform operating thresholds across different systems. For the same system, it is possible to set one operating threshold across all galleries and probe sets. Using this *base-operating threshold*, one computes the verification and false accept rate for each gallery and probe set. The resulting verification and false alarm rates will vary across different galleries and probe sets. This method for computing variance in performance models the situation in which the operating threshold is set once for an application. Setting the base-operating threshold can be based upon an overall desired performance level for the population that will use the system. In the FRVT 2002 protocol, the base-operating threshold is set based upon the system performance on an aggregate population. The base-operating threshold corresponds to a specific false accept rate on the aggregate population—this is referred to as the nominal false accept rate.

In most ROCs, verification performance is reported for a single large gallery. The results do not address the important question of how performance varies if the people in the gallery are different. This question was studied in FRVT 2002, and here we present the technique that was used. To measure variation due to gallery change, verification performance was computed for the twelve galleries, see Fig. 21.4. each of the twelve galleries consisted of different people. Each gallery consisted of 3000 people. The probe set contained two images of each person in the



**Fig. 21.4** Standard error ellipses for verification performance for Cognitec, Eyematic, and Identix. The standard error was computed for twelve galleries of size 3000. The *center line* is the ROC performance for the aggregate of all twelve galleries. The *ellipses* are two times the standard deviation at select performance points. Each ellipse is computed from the performance of the twelve small galleries at the selected performance points. Each point clustered around an ellipse corresponds to one of the twelve galleries. Performance is for FRVT 2002 and is explained in Sect. 21.5.1

gallery and 6000 true imposters (two images of 3000 individuals). The centerline is the aggregate performance for the twelve galleries. For selected operating points, performance was computed for the twelve small galleries and probe sets. For each of the twelve galleries, verification rates and false accept rates were computed. Thus, at each operating threshold, there are twelve pairs of verification and false accept rates. A standard error ellipse was computed for each set of verification and false accept rates.

Error ellipses in Fig. 21.4 are two times the standard deviation of the verification and false accept rates along the appropriate axes. An ellipse gives an estimate of the range in performance that could result if the people in the gallery are changed. If the large gallery were larger, it would be possible to compute performance for more small galleries of size 3000. The greater number of small galleries would increase the accuracy of the error ellipse. However, the size of the ellipses would not decrease as the number of small galleries increased. This is because the error ellipses are a function of the multiple small galleries, and composition of the small galleries reflects the natural variation in the population. The natural variation will always be present—more small galleries increase the accuracy of the estimated variation in the performance due to the natural composition of the population. In the HCInt, the ellipses are estimated from disjoint galleries and probe sets. This avoids many of the issues associated with resampling techniques. Resampling techniques require making assumptions about the distributional properties of the similarity scores. Typical

assumptions are that similarity scores are independent and identically distributed (i.i.d.). In interpreting the meaning of error ellipses, a number of subtle facts need to be noted. The error ellipses are not error bounds on the ROC. Rather, error ellipses are a measure of the variance in performance that occurs by changing the gallery. The standard error is an empirical estimate of the variation. They are not confidence intervals. Confidence intervals decrease in size as the number of samples increase. To estimate confidence intervals requires that one knows or can estimate the underlying distribution.

## 21.3 Evaluation Protocols

A set of design principles and its associated testing protocol describe how evaluations are designed and conducted. Design principles outline the core philosophy and guiding beliefs in designing an evaluation; the evaluation protocol provides the implementation details.

The defacto evaluation protocol standards in face recognition and biometrics are the FRVT 2002 and FERET evaluation protocols [17, 18]. The FRVT 2002 evaluation protocol is based on the September 1996 evaluation protocol. The FRVT 2002 protocol added general biometric samples, normalization of similarity scores, and an XML-based specification [12]. The XML-based specification is extensible to other biometrics and is being used for fingerprint recognition evaluation.

The design of FRVT 2002, along with the FERET evaluations and FRVT 2000, followed the precepts for biometrics evaluations articulated in Phillips et al. [16]. Succinctly stated, the precepts are:

1. Evaluations are designed and administered by groups that are independent of algorithm developers and vendors being tested.
2. Test data is sequestered and not seen by the participants prior to an evaluation.
3. The evaluation test design, protocol, and methodology are published.
4. Performance results are spread in a manner that allows for meaningful differences among the participants.

Points 1 and 2 ensure fairness in an evaluation. Point 1 provides assurance that the test is not designed to favor one participant over another. Independent evaluations help enforce points 2 and 4. In addition, point 2 ensures that systems are evaluated on their ability to generalize performance to new data sets, not the ability of the system to be tuned to a particular set of biometric samples. When judging and interpreting results, it is necessary to understand the conditions under which algorithms and systems are tested. These conditions are described in the evaluation test design, protocol and methodology. Tests are administered using an evaluation protocol that identifies the mechanics of the tests and the manner in which the tests will be scored. In face recognition, the protocol states the number and types of images of each person in the test, how the output from the algorithm is recorded, and how the performance results are reported. Publishing the evaluation protocol, as recommended in point 3, lets the readers of published results understand how the results were computed.

Point 4 addresses the *three bears* problem. Phillips et al. [17] first articulated the three bears problem in designing evaluations. The three bears problem sets guiding principles for designing an evaluation of the right level of difficulty. If all the scores for all algorithms are too high and within the same error margin, then one cannot distinguish among the algorithms tested. In addition, if the scores are too high in an evaluation, then that is an indication that the evaluation was in reality an exercise in ‘tuning’ algorithm parameters. If the scores are too low, then it is not possible to determine what problems have been solved. The goal in designing an evaluation is to have variation among the scores. There are two sorts of variation. The first type is variation among the experiments in an evaluation. Most evaluations consist of a set of experiments, where each experiment reports performance on different problems in face recognition. For example, experiments might look at changes in lighting or subject pose of a face. The second type of variation is among algorithms for each experiment. The variation in performance among the experiments lets one know which problems are currently sufficiently solved for consideration in field testing, which problems are research problems, and which problems are beyond the capabilities of the field. The variation among algorithm performance lets one know which techniques are best for a particular experiment. If all the scores for all algorithms across all experiments are virtually the same, then one cannot distinguish among the algorithms.

The key elements that ease adoption of points three and four can be incorporated into the evaluation protocol. For the FERET and FRVT evaluations, this was the FERET and FRVT 2002 evaluation protocol. This evaluation protocol was designed to assess the state of the art, advance the state of the art, and point to future directions of research. The ability to accomplish these three goals simultaneously was through a protocol whose framework allows for the computation of performance statistics for multiple galleries and probe sets. This allows for the FERET and FRVT 2002 evaluation protocol to solve the three bears problem by including galleries and probe sets of different difficulties into the evaluation. This produces a comprehensive set of performance statistics that assess the state of the art, progress in face recognition, and point to future directions of research. The use of an XML-based specification allows for this evaluation protocol to become a formal standard for biometric evaluation.

The solution to the three bears problem lies in the selection of images used in the evaluation. The characteristics and quality of the images are major factors in determining the difficulty of the problem being evaluated. For example, the location of the face in an image can affect problem difficulty. The problem is much easier if a face must be in the center of image compared to the case where a face can be located anywhere within the image. In FERET and FRVT 2002 data sets, variability was introduced by the size of the database, inclusion of images taken at different dates and both outdoor and indoor locations. This resulted in changes in lighting, scale, and background.

The testing protocol is based upon a set of design principles. The design principles directly relate the evaluation to the face recognition problem being evaluated. In particular, for FERET and FRVT 2000, the driving applications were searching large databases and access control. Stating the design principles allows one to

assess how appropriate the FERET tests and FRVT 2000 are for a particular face recognition algorithm. Also, design principles assist in determining if an evaluation methodology for testing algorithm(s) for a particular application is appropriate.

The FERET and FRVT 2002 evaluation protocols consists of two parts. The first is the rules for conducting an evaluation, and the second is the format of the results that allow for scoring. For FERET this was file format based and for FRVT 2002 the file format specifications are XML-based.

The input to an algorithm or system being evaluated is two sets of biometrics samples, target set  $\mathcal{T}$  and a query sets  $\mathcal{Q}$ . Galleries and probe sets are constructed from the target and query sets, respectively. The output from an algorithm is a similarity measure  $s_{ij}$  between all pairs of images  $t_i$  from the target set and  $q_j$  for the query sets. A similarity measure is a numerical measure of how similar two faces are. In FERET and FRVT 2002, a larger similarity scores implies greater similarity between two faces. Performance statistics are computed from the similarity measures. A complete set of similarity scores between all pairs of biometric samples from the target and query set is referred to as a similarity matrix. The first rule in the FERET and FRVT 2002 evaluation protocol is that a complete similarity matrix must be computed. This rule guarantees that performance statistics can be computed for all algorithms.

To be able to compute performance for multiple galleries and probe sets requires that multiple biometric samples of a person are placed in both the target and query sets. This leads to the second rule: Each biometrics sample in the target and query sets is considered to contain an unique sample. In practice, this rule is enforced by giving every sample in the target and query sets a unique random identifier.

The third rule is that training is completed prior to the start of an evaluation. This forces each algorithm to have a general representation for faces, not a representation tuned to a specific gallery. Also, if training were specific to a gallery, it would not be possible to construct multiple galleries and probe sets from a single run. An algorithm would have to be retrained and the evaluation rerun for each gallery. In the FRVT 2002 protocol, similarity score normalization is permitted. This allows for adjustments based on the samples in a gallery.

Using target and query sets allows us to compute performance for different categories of biometric samples. Using face recognition as an example, possible probe categories include (1) gallery and probe images taken on the same day, (2) duplicates taken within a week of the gallery image, and (3) duplicates where the time between the images is at least one year. This is illustrated in the following example. A target and query set consists of the same set of facial images. Eight images of each face are taken. Each face is taken both indoors and outdoors, with two different facial expressions on two different days. From these target and query sets, one can measure the effects of indoor versus outdoor illumination by constructing a gallery of indoor images and a probe set of outdoor images, both consisting of neutral expressions taken on the first day. Construction of similar galleries and probe sets would allow one to test the effects of temporal or expression changes. The effect of covariates such as age and sex of a person can also be measured. It is the ability to construct virtual galleries from the target set and virtual probe sets from the query set that allows the FERET and FRVT 2002 protocol to perform detailed analyses.

The FERET and FRVT 2002 evaluation protocol allows for the computation of performance statistics for verification, and open-set and closed-set identification tasks. The protocol is sufficiently flexible that one can estimate performance using subsampling and resampling techniques. For example, galleries of varying sizes are created to measure the effects of gallery size on performance. To estimate the variability of performance, multiple galleries are created.

Given the numerous theories and techniques that are applicable to face recognition, it is clear that evaluation and benchmarking of these algorithms is crucial. Evaluations and benchmarking allow for testing of theories and identification of the most promising approaches. The most important face recognition evaluations are the three FERET evaluations and the three Face Recognition Vendor Tests (FRVT). All six evaluations build on each other. The three FERET evaluations were administered in August 1994, March 1995, and September 1996. The three FRVT evaluations were administered in 2000, 2002, and 2006 (the FRVT 2006 is not covered in this review, please see Phillips et al. [21]). The MBE 2010 Still Face Track was administered between January and May 2010.

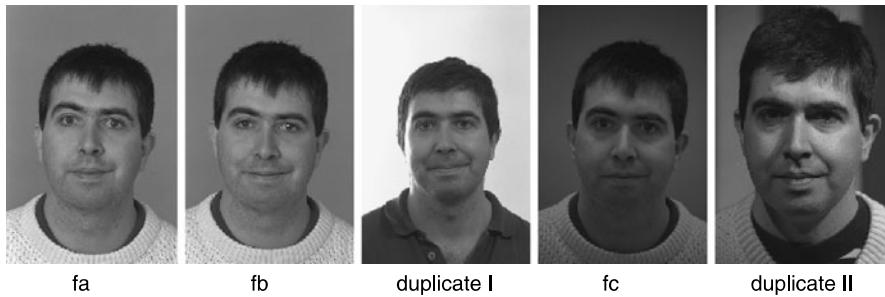
## 21.4 The FERET Evaluations

Until the FERET evaluations, there did not exist a common evaluation protocol that included a large data set and a standard evaluation method. This made it difficult to assess the status of face recognition technology, even though many existing systems reported almost perfect performance on small data sets.

The first FERET evaluation test was administered in August 1994 [15]. This evaluation established a baseline for face recognition algorithms, and was designed to measure performance of algorithms that could automatically locate, normalize, and identify faces. This evaluation consisted of three tests, each with a different gallery and probe set. (A gallery is a set of known individuals, while a probe is a set of unknown faces presented for recognition.) The first test measured identification performance from a gallery of 316 individuals with one image per person; the second was a false-alarm test; and the third measured the effects of pose changes on performance. The second FERET evaluation was administered in March 1995; it consisted of a single test that measured identification performance from a gallery of 817 individuals, and included 463 duplicates in the probe set [15]. (A duplicate is a probe for which the corresponding gallery image was taken on a different day; there were only 60 duplicates in the Aug94 evaluation.) The third and last evaluation (Sep96) was administered in September 1996 and March 1997.

### 21.4.1 Database

The FERET database was the first data set that was available to researchers. In terms of the number of people, it is the largest data set that is publicly available.



**Fig. 21.5** Images from the FERET dataset. The **fa** and **fb** were taken with the same lighting condition with different expressions. The **fc** image has a different lighting condition than the **fa** and **fb** images. The **duplicate I** image was taken within one year of the **fa** image and the **duplicate II** and **fa** image were taken at least one year apart

The images in the database were initially acquired with a 35-mm camera and then digitized.

The images were collected in 15 sessions between August 1993 and July 1996. Each session lasted one or two days, and the location and setup did not change during the session. Sets of 5 to 11 images of each individual were acquired under relatively unconstrained conditions. They included two frontal views; in the first of these (fa) a neutral facial expression was requested and in the second (fb) a different facial expression was requested (these requests were not always honored); see Fig. 21.5. For 200 individuals, a third frontal view was taken using a different camera and different lighting condition; this is referred to as the fc image. The remaining images were nonfrontal and included right and left profiles, right and left quarter profiles, and right and left half profiles. The FERET database consists of 1564 sets of images (1199 original sets and 365 duplicate sets)—a total of 14126 images. A development set of 503 sets of images were released to researchers; the remaining images were sequestered for independent evaluation. In late 2000 the entire FERET database was released along with the Sep96 evaluation protocols, evaluation scoring code, and baseline PCA algorithms.

### 21.4.2 Evaluation

For details of the three FERET evaluations, see [15, 17, 22]. The results of the most recent FERET evaluation (Sep96) will be briefly reviewed here. Because the entire FERET data set has been released, the Sep96 protocol provides a good benchmark for performance of new algorithms. For the Sep96 evaluation, there was a primary gallery consisting of one frontal image (fa) per person for 1196 individuals. This was the core gallery used to measure performance for the following four different probe sets.

- **fb** probes—Gallery and probe images of an individual taken on the same day with the same lighting (1195 probes).

- fc probes—Gallery and probe images of an individual taken on the same day with different lighting (194 probes).
- Dup I probes—Gallery and probe images of an individual taken on different days—duplicate images (722 probes).
- Dup II probes—Gallery and probe images of an individual taken over a year apart (the gallery consisted of 894 images; 234 probes).

The Sep96 evaluation tested the following ten algorithms:

- An algorithm from Excalibur Corporation (Carlsbad, CA) (Sept. 1996).
- Two algorithms from MIT Media Laboratory (Sept. 1996) [13, 25].
- Three Linear Discriminant Analysis based algorithms from Michigan State University [24]. (Sept. 1996) and the University of Maryland [8, 28] (Sept. 1996 and March 1997).
- A gray-scale projection algorithm from Rutgers University [26] (Sept. 1996).
- An Elastic Graph Matching algorithm from the University of Southern California [7, 27] (March 1997).
- A baseline PCA algorithm [14, 25].
- A baseline normalized correlation matching algorithm.

Performance was computed for both closed-set identification and verification. Three of the algorithms performed very well: Probabilistic Eigenface from MIT [13], Subspace LDA from UMD [28, 29], and Elastic Graph Matching from USC [7, 27].

A number of lessons were learned from the FERET evaluations. The first is that performance depends on the probe category and there is a difference between best and average algorithm performance.

Another lesson is that the scenario has an impact on performance. For identification, on the fb and duplicate probes, the USC scores were 94% and 59%, and the UMD scores were 96% and 47%.

### **21.4.3 Summary**

The availability of the FERET database and evaluation technology has had a significant impact on progress in the development of face recognition algorithms. The FERET data set facilitated the development of algorithms, and the FERET series of tests has allowed advances in algorithm development to be quantified. This is illustrated by, the performance improvements in the MIT algorithms between March 1995 and September 1996, and in the UMD algorithms between September 1996 and March 1997.

Another important contribution of the FERET evaluations is the identification of areas for future research. In general, the test results revealed three major problem areas: recognizing duplicates, recognizing people under illumination variations, and under pose variations.

## 21.5 The FRVT 2000

The Sep96 FERET evaluation measured performance on prototype laboratory systems. After March 1997, there was rapid advancement in the development of commercial face recognition systems. This advancement represented both a maturing of face recognition technology, and the development of the supporting system and infrastructure necessary to create commercial off-the-shelf (COTS) systems. By the beginning of 2000, COTS face recognition systems were readily available.

To assess the state of the art in COTS face recognition systems the Face Recognition Vendor Test (FRVT) 2000 was organized [5]. FRVT 2000 was a technology evaluation that used the Sep96 evaluation protocol, but was significantly more demanding than the Sep96 FERET evaluation.

Participation in FRVT 2000 was restricted to COTS systems, with companies from Australia, Germany, and the United States participating. The five companies evaluated were Banque-Tec International Pty. Ltd., C-VIS Computer Vision and Automation GmbH, Miros, Inc., Lau Technologies, and Visionics Corporation.

A greater variety of imagery was used in FRVT 2000 than in the FERET evaluations. FRVT 2000 reported results in eight general categories: compression, distance, expression, illumination, media, pose, resolution, and temporal. There was no common gallery across all eight categories; the sizes of the galleries and probe sets varied from category to category.

We briefly summarize the results of FRVT 2000. Full details can be found in Blackburn et al. [5], and include identification and verification performance statistics. The media experiments showed that changes in media do not adversely affect performance. Images of a person were taken simultaneously on conventional film and on digital media. The compression experiments showed that compression does not adversely affect performance. Probe images compressed up to 40:1 did not reduce recognition rates. The compression algorithm was JPEG.

FRVT 2000 also examined the effect of pose angle on performance. The results show that pose does not significantly affect performance up to  $\pm 25^\circ$ , but that performance is significantly affected when the pose angle reaches  $\pm 40^\circ$ .

In the illumination category, two key effects were investigated. The first was lighting change indoors. This was equivalent to the fc probes in FERET. For the best system in this category, the indoor change of lighting did not significantly affect performance. In a second experiment, recognition with an indoor gallery and an outdoor probe set was computed. Moving from indoor to outdoor lighting significantly affected performance, with the best system achieving an identification rate of only 0.55.

The temporal category is equivalent to the duplicate probes in FERET. To compare progress since FERET, dup I and dup II scores were reported. For FRVT 2000 the dup I identification rate was 0.63 compared with 0.58 for FERET. The corresponding rates for dup II were 0.64 for FRVT 2000 and 0.52 for FERET. These results show that there was algorithmic progress between the FERET and FRVT 2000 evaluations. FRVT 2000 showed that two common concerns, the effects of compression and recording media, do not affect performance. It also showed that

future areas of interest continue to be duplicates, pose variations, and illumination variations generated when comparing indoor images with outdoor images.

### 21.5.1 *The FRVT 2002*

The Face Recognition Vendor Test (FRVT) 2002 was a large-scale evaluation of automatic face recognition technology. The primary objective of FRVT 2002 was to provide performance measures for assessing the ability of automatic face recognition systems to meet real-world requirements. Ten participants were evaluated under the direct supervision of the FRVT 2002 organizers in July and August 2002. Ten companies participated in FRVT 2002: AcSys Biometrics Corp., Cognitec Systems GmbH, C-VIS Computer Vision and Automation GmbH, Dream Mirh Co., Ltd, Eyematics Interfaces Inc., Iconquest, Identix, Imagis Technologies Inc., Viisage Technology, VisionSphere Technologies Inc.

FRVT 2002 consisted of two parts: high computational intensity test (HCInt) and the medium computational intensity test (MCInt). The heart of the FRVT 2002 was the HCInt, which consisted of 121 589 operational images of 37 437 people. The images were provided from the U.S. Department of State's Mexican nonimmigrant Visa archive. From this data, real-world performance figures on a very large data set were computed. Performance statistics were computed for verification, closed-set identification, and open-set identification (watch list) tasks. Open-set identification performance is reported in Fig. 21.1, closed-set identification performance is reported in Fig. 21.3, verification performance with error ellipses is given in Fig. 21.2 for the HCInt (only eight of the ten companies took the HCInt portion of FRVT 2002). The MCInt measured performance on facial images from different categories. The categories included mugshot style images, still images taken outside, nonfrontal indoor images, and morphed nonfrontal images.

FRVT 2002 results show that normal changes in indoor lighting do not significantly affect performance of the top systems. Approximately the same performance results were obtained using two indoor data sets, with different lighting, in FRVT 2002. In both experiments, the best performer had a 90% verification rate at a false accept rate of 1%. On comparable experiments conducted two years earlier in FRVT 2000, the results of FRVT 2002 indicate there has been a 50% reduction in error rates. For the best face recognition systems, the recognition rate for faces captured outdoors, at a false accept rate of 1%, was only 50%. Thus, face recognition from outdoor imagery remains a research challenge area.

A very important question for real-world applications is the rate of decrease in performance as time increases between the acquisition of the database of image and new images presented to a system. FRVT 2002 found that for the top systems, performance degraded at approximately 5% points per year.

One open question in face recognition is: How does database and watch list size effect performance? Because of the large number of people and images in the FRVT 2002 data set, FRVT 2002 reported the first large-scale results on this question.

For the best system, the top-rank identification rate was 85% on a database of 800 people, 83% on a database of 1600, and 73% on a database of 37437. For every doubling of database size, performance decreases by two to three overall percentage points. More generally, identification performance decreases linearly in the logarithm of the database size.

Previous evaluations have reported face recognition performance as a function of imaging properties. For example, previous reports compared the differences in performance when using indoor versus outdoor images, or frontal versus non-frontal images. FRVT 2002, for the first time, examined the effects of demographics on performance. Two major effects were found. First, recognition rates for males were higher than females. For the top systems, identification rates for males were 6% to 9% points higher than that of females. For the best system, identification performance on males was 78% and for females was 79%. Second, recognition rates for older people were higher than younger people. For 18 to 22 year olds, the average identification rate for the top systems was 62%, and for 38 to 42 year olds was 74%. For every ten years increase in age, on average performance increases approximately 5% through age 63.

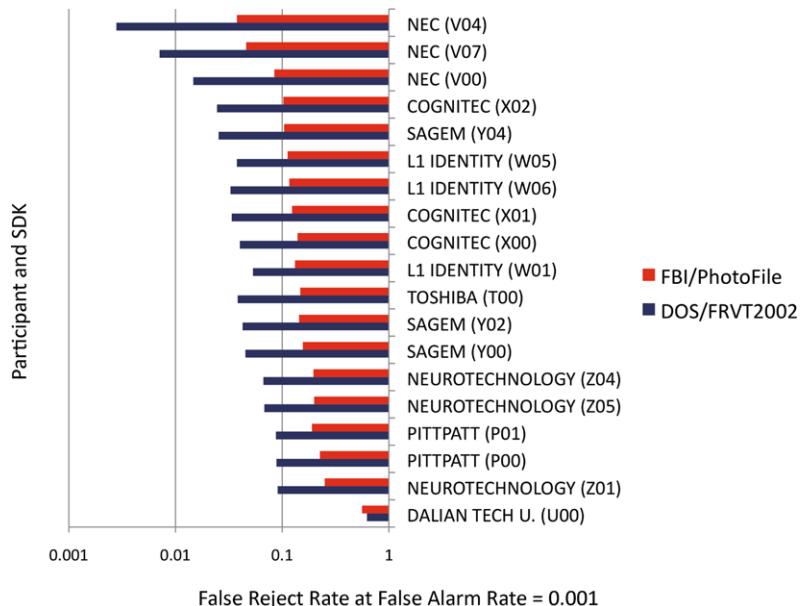
FRVT 2002 looked at two of these new techniques. The first was the three-dimensional morphable models technique of Blanz and Vetter [6]. Morphable models are a technique for improving recognition of non-frontal images. FRVT 2002 found that Blanz and Vetter's technique significantly increased recognition performance. The second technique is recognition from video sequences. Using FRVT 2002 data, recognition performance using video sequences was the same as the performance using still images.

In summary, the key lessons learned in FRVT 2002 were: (1) Given reasonable controlled indoor lighting, the current state of the art in face recognition is 90% verification at a 1% false accept rate. (2) Face recognition in outdoor images is a research problem. (3) The use of morphable models can significantly improve nonfrontal face recognition. (3) Identification performance decreases linearly in the logarithm of the size of the gallery. (4) In face recognition applications, accommodations should be made for demographic information since characteristics such as age and sex can significantly affect performance.

## 21.6 The MBE 2010 Still Face Track

The primary goals of the Multiple Biometric Evaluation (MBE) 2010 Still Face Track were to measure improvement in face recognition from frontal still faces since the FRVT 2006 and measure identification performance on extremely large datasets [10, 21]. The MBE 2010 Still Face Track had an open submission period from January through May 2010. Participants could submit multiple systems in an SDK format during this submission period.

Performance was measured on two primary datasets. The first dataset was the FRVT 2002 HCInt dataset. Performance was also reported for this dataset in the FRVT 2006, where it was known as the low-resolution dataset. Performance on this

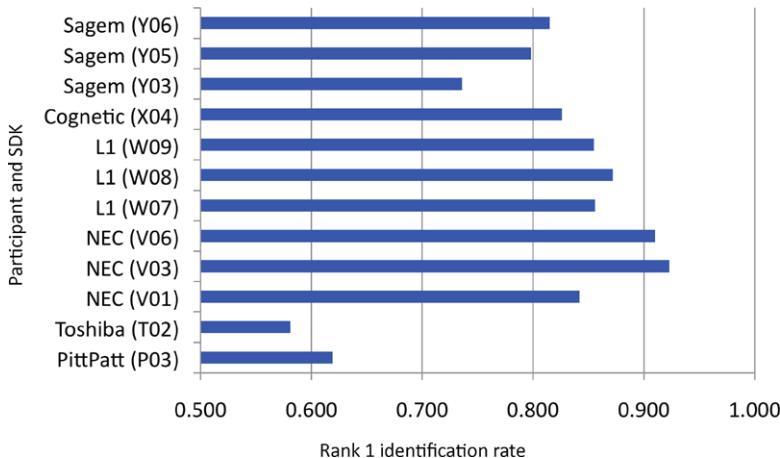


**Fig. 21.6** Verification performance on the FRVT 2002 HCInt and the FBI Photo File datasets. The FRR at a FAR = 0.001 is reported and the horizontal (FRR) axis is on a logarithmic scale. Results are reported by participant and SDK version

dataset allows for a direct measurement of improvement in algorithm performance from 2002 to 2010. The second consists of face images collected by various law enforcement agencies and transmitted to the Federal Bureau of Investigation (FBI) as part of various criminal records checks. This is known as the FBI Photo File dataset.

Verification results on the FRVT 2002 HCInt and the FBI Photo File datasets are reported in Fig. 21.6. On the HCInt data set the SDK V04 submitted by NEC achieved a false reject rate (FRR) = 0.003 at a false accept rate (FAR) = 0.001. On this dataset, the best performance in the FRVT 2002 was FRR = 0.2 at a FAR = 0.001 and in the FRVT 2006 a FRR = 0.03 at a FAR = 0.001 was achieved. The results in MBE 2010 show an improvement of almost two orders of magnitude between 2002 and 2010 on the same test set of images. The best performance on the FBI Photo File data was a FRR = 0.038 at a FAR = 0.001. On the FBI Photo File data set, a reasonable number of participants had FRR of around 0.10 or better at a FAR = 0.001.

The FBI Photo File dataset allowed for measuring identification rates from galleries in excess of 1 million faces. The protocol for the MBE 2010 Still Face was designed to allow for testing extremely large scale identification problems. Closed-set identification results are reported in Fig. 21.7. The SDK V03 submitted by NEC achieved a rank 1 identification rate = 0.93 on a gallery of 1.6 million faces. SDK's from four participants achieved a rank 1 identification rate of 0.80 or better on this dataset.



**Fig. 21.7** Closed-set identification performance on the FBI Photo File dataset with a gallery of 1.6 million faces. Rank 1 identification rate is report. Results are reported by participant and SDK version

The results from the MBE 2010 Still Face Track shows how automatic face recognition has matured since the start of the FERET program in 1993. The results from MBE 2010 show that automatic face recognition systems are ready for consideration for applications that performing recognition from controlled frontal face images.

## 21.7 Issues and Discussions

Periodic face recognition evaluations have advanced the field of face recognition from its infancy in the early 1990s to the deployment of systems. The periodic evaluations have developed standard evaluation protocols and methods for reporting performance scores. As the field of face recognition has advanced, there has been an concomitant advancement in evaluation techniques. This is leading to different styles of evaluations, each with its ability to answer different questions.

The advance in automatic face recognition from FERET to MBE 2010 has been documented by a series of evaluations. These evaluation show that recognition from frontal still images acquired under controlled conditions has matured. However, this does not mean that automatic face recognition is a solved problem. Face recognition from still and video taken in unconstrained conditions is a research challenge. In fact, the Multiple Biometric Grand Challenge (MBGC) was designed to address these problems [20].

The face recognition community, and biometrics in general, has developed a range of evaluations in terms of number of people and images. To provide a rough guide to evaluation size, we introduce the following nomenclature:

- Small: ~1000 signatures and ~330 individuals.

- Medium:  $\sim 10\,000$  signatures and  $\sim 3300$  individuals.
- Large:  $\sim 100\,000$  signatures and  $\sim 33\,000$  individuals.
- Very large:  $\sim 1\,000\,000$  signatures and  $\sim 330\,000$  individuals.
- Extremely large:  $\sim 10\,000\,000$  signatures and  $\sim 3\,300\,000$  individuals.

Each size has its own role and place. A larger evaluation is not inherently better, especially when cost is considered. Most evaluations have been small, but they have had a positive impact on the development and assessment of biometrics.

The FERET, FRVT 2000, and FRVT 2002 MCInt evaluations were small to medium evaluation, and were able to differentiate between large and small effects on performance. The FERET evaluations showed a big difference in performance between images taken on the same day and images takes on different days. This showed that the interesting problem for face recognition was images taken on different days. The FRVT 2002 MCInt results showed a large difference in performance between recognition of non-frontal images and non-frontal images that have been morphed. The MCInt results showed that morphable models improved performance for non-frontal images. Evaluation such as FERET and FRVT 2002 MCInt are good for making an assessment on (1) a specified set of experiments, and (2) where one is looking to distinguish between large and small effects.

The FRVT 2002 HCInt is a large evaluation and the MBE 2010 is an extremely large evaluation. The FRVT 2002 HCInt and the FBI Photo File datasets allowed for a more detailed analysis and was able to estimate the variance of performance statistics and measure the effects of covariates on performance. This analysis required not only a large numbers of images and people, but also an appropriate number of errors. If there had only been ten or hundred errors, we would not have been able to perform detailed covariate analysis. In designing very large and extremely large evaluations one needs to state the object of the evaluation and have an idea of the overall accuracy of the biometric being tested. For example, if a biometric has an identification rate of 0.9999 (error rate of one in 10 000), then an evaluation on a data set of 100 000 images would on average produce ten errors. To be able to perform a detailed analysis of performance, such as in the FRVT 2002 HCInt, would require a test set several orders of magnitude larger.

Evaluations of all sizes are needed and have their role in assessing performance of biometrics. Factors effecting the size and design of an evaluation include the evaluations goals and the overall accuracy of a biometric. The greater the accuracy of biometric, the larger the required size of an evaluation. The more detailed analysis needed, the larger the required size of an evaluation. At the other end of the scale, an evaluation with very specific and defined purposes maybe able to meets its goals with a small evaluation.

When research in automatic face recognition began, the primary goal was to develop recognition algorithms. With progress in face recognition, the goal of understanding the properties of face recognition algorithms has joined the goal of developing algorithms. Understanding the properties of face recognition is computational experiments.

## 21.8 Conclusions

Independent evaluations provide an assessment of the state-of-the-art, but do not provide an understanding of the fundamental properties of face recognition algorithms. The province of answering these types of questions is computational experiments. For example, FRVT 2002 showed that men are easier to recognize than women. However, FRVT 2002 was not designed to answer the more fundamental question of why men are easier to recognize than women. The computation experiments are being conducted. They will give greater understanding of face recognition and provide a strong scientific underpinning.

## References

1. Beveridge, J.R., She, K., Draper, B.A., Givens, G.H.: A nonparametric statistical comparison of principal component and linear discriminant subspaces for face recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 535–542 (2001)
2. Beveridge, J.R., Givens, G.H., Phillips, P.J., Draper, B.A., Lui, Y.M.: Focus on quality, predicting FRVT 2006 performance. In: Proceeding of the Eighth International Conference on Automatic Face and Gesture Recognition (2008)
3. Beveridge, J.R., Givens, G.H., Phillips, P.J., Draper, B.A.: Factors that influence algorithm performance in the Face Recognition Grand Challenge. *Comput. Vis. Image Underst.* **113**, 750–762 (2009)
4. Beveridge, J.R., Givens, G.H., Phillips, P.J., Draper, B.A., Bolme, D.S., Lui, Y.M.: FRVT 2006: Quo vadis face quality. *Image Vis. Comput.* **28**(5), 732–743 (2010)
5. Blackburn, D., Bone, M., Phillips, P.J.: Face recognition vendor test 2000. Technical report (2001). <http://www.frvt.org>
6. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: Proceedings, SIGGRAPH'99, pp. 187–194 (1999)
7. Okada, K., et al.: The Bochum/USC face recognition system. In: Wechsler, H., Phillips, P.J., Bruce, V., Fogelman Soulie, F., Huang, T.S. (eds.) *Face Recognition: From Theory to Applications*. Springer, Berlin (1998)
8. Etemad, K., Chellappa, R.: Discriminant analysis for recognition of human face images. *J. Opt. Soc. Am. A* **14**, 1724–1733 (1997)
9. Givens, G.H., Beveridge, J.R., Draper, B.A., Bolme, D.: A statistical assessment of subject factors in the pca recognition of human faces. In: CVPR 2003 Workshop on Statistical Analysis in Computer Vision Workshop (2003)
10. Grother, P.J., Quinn, G.W., Phillips, P.J.: MBE 2010: Report on the evaluation of 2D still-image face recognition algorithms. NISTIR 7709, National Institute of Standards and Technology (2010)
11. Micheals, R.J., Boult, T.: Efficient evaluation of classification and recognition systems. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 50–57 (2001)
12. Micheals, R.J., Grother, P., Phillips, P.J.: The NIST HumanID evaluation framework. In: Kittler, J., Nixon, M.S. (eds.) *Third Inter. Conf. on Audio- and Video-Based Biometric Person Authentication*. LNCS, vol. 2688, pp. 403–411. Springer, Berlin (2003)
13. Moghaddam, B., Nastar, C., Pentland, A.: Bayesian face recognition using deformable intensity surfaces. In: Proceedings Computer Vision and Pattern Recognition 96, pp. 638–645 (1996)
14. Moon, H., Phillips, P.J.: Computational and performance aspects of PCA-based face-recognition algorithms. *Perception* **30**, 303–321 (2001)

15. Phillips, P.J., Wechsler, H., Huang, J., Rauss, P.: The FERET database and evaluation procedure for face-recognition algorithms. *Image Vis. Comput.* **16**(5), 295–306 (1998)
16. Phillips, P.J., Martin, A., Wilson, C.L., Przybocki, M.: An introduction to evaluating biometric systems. *Computer* **33**, 56–63 (2000)
17. Phillips, P.J., Moon, H., Rizvi, S., Rauss, P.: The FERET evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 1090–1104 (2000)
18. Phillips, P., Grother, P., Micheals, R., Blackburn, D., Tabassi, E., Bone, J.: Face recognition vendor test 2002: Evaluation report. Technical Report NISTIR 6965, National Institute of Standards and Technology (2003). <http://www.frvt.org>
19. Phillips, P.J., Flynn, P.J., Scruggs, T., Bowyer, K.W., Chang, J., Hoffman, K., Marques, J., Min, J., Worek, W.: Overview of the face recognition grand challenge. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 947–954 (2005)
20. Phillips, P.J., Flynn, P.J., Beveridge, J.R., Scruggs, W.T., O'Toole, A.J., Bolme, D., Bowyer, K.W., Draper, B.A., Givens, G.H., Lui, Y.M., Sahibzada, H., Scallan, J.A. III, Weimer, S.: Overview of the Multiple Biometrics Grand Challenge. In: *Proceedings Third IAPR International Conference on Biometrics* (2009)
21. Phillips, P.J., Scruggs, W.T., O'Toole, A.J., Flynn, P.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: FRVT 2006 and ICE 2006 large-scale results. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(5), 831–846 (2010)
22. Rizvi, S., Phillips, P.J., Moon, H.: A verification protocol and statistical performance analysis for face recognition algorithms. In: *Computer Vision and Pattern Recognition 98*, pp. 833–838 (1998)
23. Sarkar, S., Phillips, P.J., Liu, Z., Robledo, I., Grother, P., Bowyer, K.W.: The HumanID gait challenge problem: Data sets, performance, and analysis. Technical report (2003). <http://www.gaitchallenge.org>
24. Swets, D., Weng, J.: Using discriminant eigenfeatures for image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **18**(8), 831–836 (1996)
25. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–86 (1991)
26. Wilder, J.: Face recognition using transform coding of gray scale projection projections and the neural tree network. In: Mammone, R.J. (ed.) *Artificial Neural Networks with Applications in Speech and Vision*, pp. 520–536. Chapman & Hall, London (1994)
27. Wiskott, L., Fellous, J.-M., Kruger, N., von der Malsburg, C.: Face recognition by elastic bunch graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.* **17**(7), 775–779 (1997)
28. Zhao, W., Chellappa, R., Krishnaswamy, A.: Discriminant analysis of principal components for face recognition. In: *3rd International Conference on Automatic Face and Gesture Recognition*, pp. 336–341 (1998)
29. Zhao, W., Krishnaswamy, A., Chellappa, R., Swets, D., Weng, J.: Discriminant analysis of principal components for face recognition. In: Wechsler, H., Phillips, P.J., Bruce, V., Fogelman Soulie, F., Huang, T.S. (eds.) *Face Recognition: From Theory to Applications*, pp. 73–85. Springer, Berlin, (1998)

# Chapter 22

## Dynamic Aspects of Face Processing in Humans

Heinrich H. Bülowhoff, Douglas W. Cunningham, and Christian Wallraven

### 22.1 Introduction

The human face is capable of a wide variety of facial expressions that manifest themselves as usually highly non-rigid deformations of the face. On the one hand, this presents the visual system with a problem: Recognizing someone requires determining what information in the face remains constant despite the various facial deformations. Extraction of such invariant features will allow me, for example, to identify my neighbor regardless of whether he or she is smiling or looking sad. On the other hand, the impressive repertoire of changes can also be seen as a positive: It provides considerable information. The particular way my neighbor smiles or looks sad might well be used for identification, similar to how Jack Nicholson's and Tom Cruise's smiles are very specific to them.

In addition to potentially providing information about who someone is, facial deformations can help us to infer something about a person's age, social status, general health, level of fatigue, and focus of attention. Likewise, changes in the facial surface play a central, albeit often ignored, role in communication. Facial deformations that serve this latter role are generally referred to as facial expressions.

A distinction should be drawn between the information that *is* present in a *specific image* and the information that *must be* present for that expression or person to be

---

H.H. Bülowhoff (✉) · D.W. Cunningham · C. Wallraven  
Max Planck Institute for Biological Cybernetics, Spemannstrasse 38, 72076 Tübingen, Germany  
e-mail: [heinrich.buelhoff@tuebingen.mpg.de](mailto:heinrich.buelhoff@tuebingen.mpg.de)

H.H. Bülowhoff · C. Wallraven  
Department of Brain and Cognitive Engineering, Korea University, Seoul, Korea

C. Wallraven  
e-mail: [wallraven@korea.ac.kr](mailto:wallraven@korea.ac.kr)

D.W. Cunningham  
Brandenburg Technical University, 03046 Cottbus, Germany  
e-mail: [douglas.cunningham@tu-cottbus.de](mailto:douglas.cunningham@tu-cottbus.de)

recognized. Trying to determine what information is perceptually necessary not only provides critical insights into how humans process faces, but can also yield clues for the design of automated facial recognition and synthesis systems.

Almost all research on the perception of faces—both for identity and expression—has tended to focus on the relatively stable aspects of faces. Some of this information is invariant to deformations of the facial surface, such as the color of or distance between the eyes. In other cases, the result of the deformation *is* the information, such as the shape of the mouth. In such cases, usually the maximum deformation (or peak expression) is examined. In other words, there is a pervasive emphasis on *static* facial information. To some degree, this focus on static information is driven by technology. It is difficult to systematically manipulate a photograph in order to provide the systematic and parameterized variations needed for perceptual experiments without making the photograph look unrealistic. It is considerably more difficult to perform the same manipulations on a *sequence* of images without introducing artifacts either in any given image or across images [29, 80].

In general, however, human faces are not static entities. Indeed, if we meet someone who never moved their face, we would most likely be rather uncomfortable. Some have gone so far as to argue that an individual (specifically, an android) that looks like a human but moves either incorrectly or not at all (i.e., has a “dead” face) will—as a result of the zombie-like appearance—lead to humans being repulsed by that individual [75]. This hypothesis is referred to as the “uncanny valley”. Regardless of whether a zombie—or zombie-like individual—will repulse humans or not, it is clear that the pattern of change over time is itself a great source of information for many visual processes [46] and can often be used to discern real from synthesized stimuli [106].

Fortunately, recent advances in technology, have allowed researchers to carefully and systematically alter video sequences without introducing noticeable artifacts and thus begin to examine the role of motion in face processing. Before one can determine what types of motion are used (i.e., uncover the dynamic features), one must determine if motion plays any role at all. It has been shown that facial motion can provide information about gender [15, 51] and age [14].

In this chapter, we will focus on the role of motion in identity (Sect. 22.2) and expression (Sect. 22.3) recognition in humans, and explain its developmental and neurophysiological aspects. We will make some inferences and conclusions based on results from literature.

## 22.2 Dynamic Information for Identity

Correctly identifying other people is critical to everyday survival and there has been a considerable amount of research on how such a task might be performed. The literature on how humans use faces to recognize people is quite extensive (for a review, see Chap. 26, this volume or [91]). While the great majority of this literature has focused on static information for identity, there has been an increasing interest in dynamic information (see, e.g., [81]).

Motion can be subdivided into rigid and nonrigid types. Rigid “facial” motion generally refers to the rotations and/or translations of the entire head (e.g., such as nodding or shaking the head). Nonrigid facial motion, in contrast, generally refers to the nonlinear deformations of the facial surface (e.g., lip motion, eyebrow motion). One of the first studies to examine motion presented a 10 second clip of an individual rotating in a chair through a full 360 degrees (rigid motion; [83]). This motion was chosen as it represents a simple change in relative viewpoint, and thus may help an observer to build up a more complete 3D representation of the individual. Accordingly, [83] found higher identity recognition performance in dynamic conditions than in static conditions.

Shortly thereafter, [24] presented contrasting results. They showed five frames of a person moving his/her head up and down and found *no* difference between static and dynamic conditions. The head motion here, they suggested, represents social communication (such as a nod of agreement). Thus the difference in results might be represent a contrast between viewpoint change and social signal. It is important to note that there are a number of other differences between the two studies, such as length of the stimuli, direction of motion (horizontal versus vertical rotations), and task. Subsequent studies have examined some of the differences to determine the source of the conflict.

Another way of saying that [83]’s videos were longer than [24]’s is to say that they contained many more images. This raises the possibility that not only is the difference in results due to the length of the stimuli, but perhaps [83]’s dynamic advantage itself is due merely to the number of images: The dynamic sequence has more images than the static image. Perhaps one of the many views shown in [83]’s had a facial view which was more optimal than the one used in the static condition or in [24]’s five image video. To explicitly test this hypothesis, [69] asked participants to identify a number of famous faces, which were presented in three different formats: as a nine-frame video sequence, a static  $3 \times 3$  array of the nine images in order, and a static  $3 \times 3$  array with the nine images in random order. Not only was there a dynamic advantage (despite the sequences being only 9 frames long), but performance in the two static conditions did not differ from one another. Thus, the reason why video sequences are recognized better is not simply that they have more snapshots. It is important to notice, however, that [69]’s sequences consisted primarily of nonrigid motion (talking, expressions) with little rigid motion.

Note that the 9 frames in [69]’s videos were presented sequentially in the dynamic condition and next to one another in the static conditions. Perhaps the mere presence of multiple images is sufficient, but they need to be presented one after another. That is, maybe the images need to be temporally separated or presented at the same spot on the monitor, and motion per se is not required. To test this, [67] and [83] presented a video where the images in the video were presented in a random order. Note that such sequences have motion, but this motion is jerky and erratic (i.e., the motion does not occur in nature). They found that identity was more accurately recognized in normal sequences than in random sequences. This suggests that it is not just the presence of time or motion that is important, but the specific, naturally occurring motion (either horizontal rotation or nonrigid motion) that provides the advantage. As a final, more stringent test that it is the *characteristic motion*

that is important, [67] showed that reversing the direction of motion (by playing the sequence backwards) decreases recognition performance, suggesting that the temporal direction of the motion trajectories is important. Likewise, they showed that changing the speed of the motion sequence (e.g., by playing parts or all of a video sequence too fast or too slow) decreased recognition performance.

The finding that a scrambled version of simple, uniform, horizontal head rotation was not better than a static photograph, while an intact rotation sequence yielded a significant recognition advantage [83] suggests that something other than characteristic motion might be important. Wallis and Bülthoff [104], for example, suggested that the temporal coherence of the stimuli provides some information. Specifically, they examined how people learned new faces. While the head of the person-to-be-learned was rotated (horizontally), the identity was changed. One unfamiliar face was shown when the head was at its left extreme position and a different (but still unfamiliar) face when the head was at the right extreme position. The intermediate positions were a morph between the two identities. The results clearly show that participants treated the different identities as if they were the same person (seen from different views). Moreover, the fusion of the two identities was only found for continuous head rotations; Scrambling the order in which the views were presented eliminated the effect. Thus, it seems that spatiotemporal continuity plays a role in learning identity.

In an attempt to determine the specific roles of rigid and nonrigid motion—*independent of static information*—[52] used motion capture recordings of a conversation to animate an average face. They artificially separated rigid from nonrigid motion and examined identity recognition and sex recognition. They found that both types of motion can be used for both tasks, but that rigid head motion was slightly more useful than nonrigid for identity recognition (while the reverse was true for sex recognition). They also showed that inverting the face and playing the sequence backwards reduced recognition, again pointing to some characteristic motion information.

Following this, [68] also compared the role of different forms of motion. In contrast to [52]'s use of an average face, they used degraded images of familiar individuals (individuals from the same working environment). They found a dynamic advantage for non-rigid motions such as expressions and speech, but not for rigid motions (head nodding). Interestingly, the dynamic advantage was stronger for “distinctive” facial motions. This finding has been extended by [71], who showed that recognition of familiar faces was better when the smile was “natural” rather than “artificial”. Based on these findings, [71] concluded that some familiar faces have characteristic motions that can help in identification via incorporating supplemental motion-based information about the face.

The role of nonrigid motion in the learning of new individuals was examined by [99]. While the novel faces were being learned, the beginning of a smile or a frown was presented (specifically, the first 18 frames). Using a sequential matching paradigm, they showed an advantage of motion when the test image was the same person with a different (static) expression as well as when the same person was seen from a different view (i.e., generalization across expression and viewpoint). The

effect of dynamic information in generalization was subsequently replicated by [84] using the motions from surprise and anger and a delayed visual search paradigm.

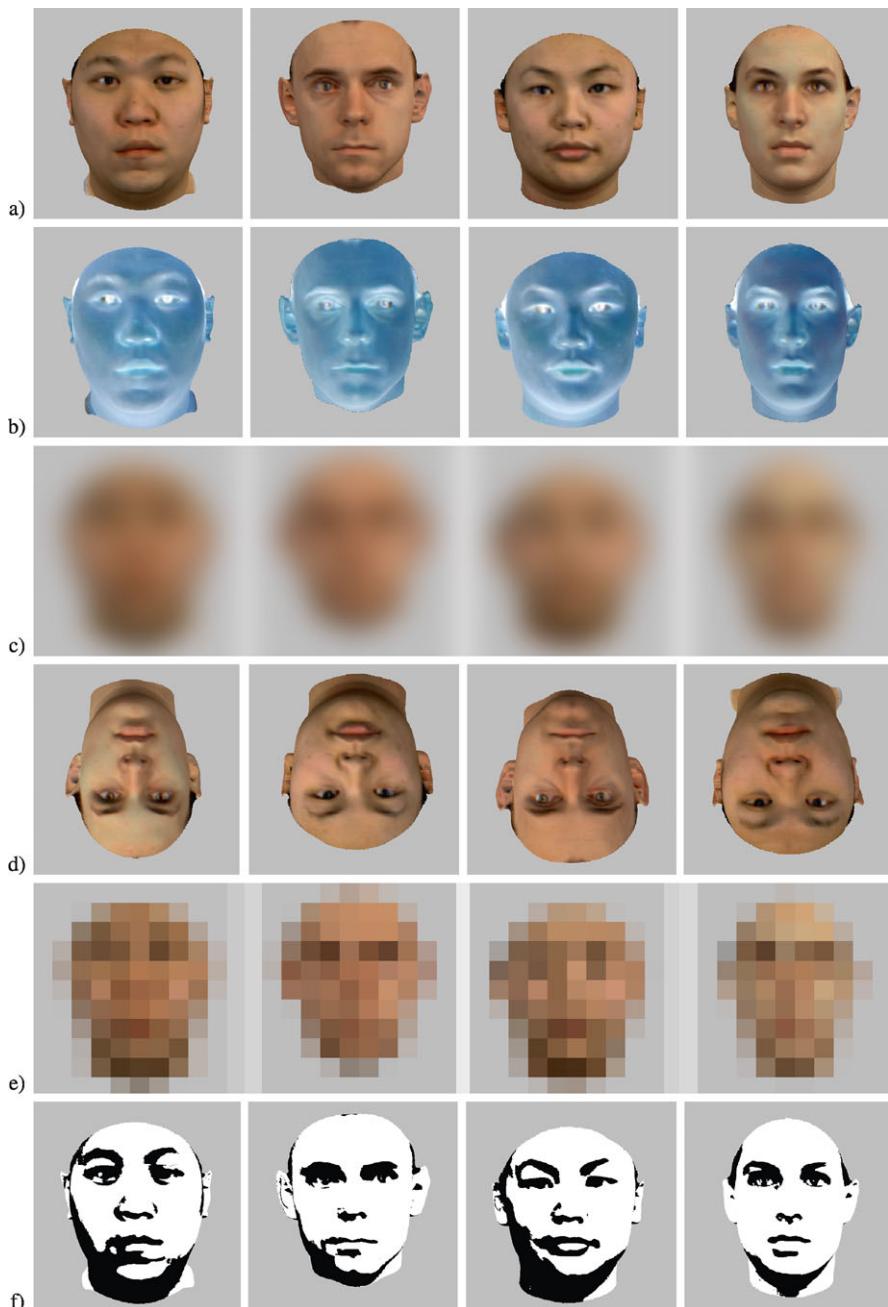
Many of the successful demonstrations of a dynamic advantage used degraded stimuli. For example, [65] presented photographic negatives of the faces (see Fig. 22.1b). Likewise, Lander and colleagues have impaired static information in a number of ways, including Gaussian blurring (see Fig. 22.1c), inverting (see Fig. 22.1d), pixelation (see Fig. 22.1e), and thresholding (reducing the image to a two-tone, black/white image; see Fig. 22.1f) [68–70]. As a result, it has been suggested that dynamic information only plays a role when static information is impaired and the person is familiar (see, e.g., [81]). The successful demonstration of a dynamic advantage in nondegraded stimuli of unfamiliar individuals by [52, 84, 99, 104] makes it unlikely that such an explanation captures the whole story.

As an alternate explanation, Thornton and colleagues [99] suggested that previous failures to find a dynamic effect (specifically that of [24]) may have been due to the task used (an old-new task). That is, it might be a memory effect, but not a face perception effect. Additional evidence for this comes from [88], who also used an old-new task to examine the role of different types of motion in learning new individuals. The videos were from a speech database, so contained rigid as well as nonrigid motion. In contrast to the other work on learning new individuals, they found no dynamic advantage, consistent with Thornton’s suggestion that the task may be problematic. There is, however, some difficulty with such an explanation: [68] found head nodding did not lead to a dynamic advantage, but they used a naming task (and not an old-new task) for familiar individuals. Interestingly, the type of rigid motion in [68]’s and [24]’s experiments was the same: vertical rotations. It is possible that this also plays a role.

In the first study to explicitly examine the interaction between static and dynamic identity information, [64] recorded several individuals performing various actions (such as chewing and smiling) and used those motions to animate unfamiliar faces. The motions and the faces were subsequently systematically combined. Participants learned, for example, two different faces each with its own motion. Subsequently, they were presented with a series of trials where the motion was always from one individual, but the face shape was a morph of the two. By systematically varying the degree of the morph, they were able to measure psychometric functions showing the independent contribution of shape and motion. They found clear evidence that the characteristic aspects of the motion influenced the learning and subsequent recognition of identity.

### 22.2.1 Developmental Aspects

As part of the large body of literature on the importance of motion and moving stimuli in general for the perceptual development of the infant, several studies have investigated how infants might benefit from the information inherent in moving faces. Looking at the performance of infants in the context of dynamic face recognition



**Fig. 22.1** Examples for several individuals of some image degradation procedures **a** the original photos; **b** Photographic negative version of those individuals; **c** Gaussian blur; **d** inversion; **e** pixelation; and **f** thresholding

is especially illuminating for two reasons: First, infants (especially in their first few years) have not yet become full experts in face processing. Indeed, as research has shown, face recognition takes an astonishingly long period to reach adult recognition levels [92]. This means, that infants have not yet reached “ceiling”, that is, they do not yet exhibit the excellent generalization capabilities and almost perfect recognition results usually found in adult observers. A second important reason for investigating infants is that during the first few years, information about facial identity and facial expressions comes exclusively from real, living, moving faces (for example, the gaze of infants is attracted to a movie of the mother’s face versus an abstract movie [55] at as early as 10 weeks of age), whereas for adults the processing of static faces (e.g., photographs) is a much more common activity.

One of the first studies to explicitly test whether infants are specifically sensitive to dynamic information in faces [96] is a follow-up study to [51] using the same stimuli. Since they tested young infants between 4 and 8 months of age, standard recognition paradigms could not be used. Instead, a variant on the preferential looking paradigm was employed. First, an average face told a joke. Whereas the shape of the average face was independent of the actor who told the joke, its deformation was driven by that particular actor’s motion. After having seen such a dynamic face, the infants were presented with the same average face telling a different joke. This time, however, there were two faces: one driven by the previous actor’s motion and another driven by another actor’s motion. The study found that infants tended to spend more time looking at the sequence with the previous actor’s motion indicating that, indeed, the motion signature of the actor was processed—and can be used by infants to disambiguate individuals.

Similarly, in [82] infants between 3 and 4 months of age were tested. In one experiment, infants were either presented with either a short clip of a face in motion (performing a facial expression) or of the last, static frame of that movie. After 30 seconds of exposure, infants then looked at two (static) faces differing in facial expression. The study found that infants could only tell the two faces apart using the dynamic familiarization stage. Only after the familiarization phase was extended to 90 seconds could infants use the static familiarization face for subsequent recognition.

In another related study [94], two groups of 7 and 10 year-old children were required to learn unfamiliar faces from either static or dynamic stimuli with a subsequent static or dynamic test phase. Interestingly, the study found that motion helped during familiarization but not during recognition, that is, it did not matter whether faces during testing were shown as pictures or as movies. On average—and in agreement with the general trend observed during development—older children performed better at the task.

Taken together, these studies show that motion plays perhaps the greatest role during the early stages of development, helping the perceptual system to recognize idiosyncratic movements quickly and enabling a better structural description of faces through the use of motion cues. During adulthood, however, the face processing system has reached an expert level, making it hard to identify the advantage of moving faces for recognition of identity.

### 22.2.2 Neurophysiological Aspects

Given that faces are learned in a dynamic context, one would expect the brain to have specific networks devoted to processing of spatiotemporal information about a face. Indeed, such a hypothesis would go hand in hand with the models proposed for face recognition [17, 20, 50] which posit a separate processing stream for “changeable aspects” of a face. Usually, such changeable aspects are contrasted with the invariant, static aspects such as the identity of a person. Given the discussion above of how motion can help recognition of identity in some cases, the idea of two *fully separate streams* seems unlikely in the light of these perceptual and developmental findings, however.

With the advent of fMRI as a widespread imaging technique, the question of how dynamic and static information about faces are dealt with in the brain has become the focus of a few recent studies. Again, however, the majority of studies have used non-natural, or highly abstract stimuli for testing any differences (cartoon faces, morphed expression-like stimuli, etc.). It seems clear from these earlier studies that, for example, the superior-temporal-sulcus (STS) in the dorsal pathway is active during perception of moving face stimuli (e.g., [2]). This region is also active during observation of biological motion and complex motion patterns such as optic flow in general [47].

Recently, two studies have directly contrasted activation in the brain during observation of static versus dynamic stimuli. In the first study [42], static and dynamic face stimuli were used in two different participant groups to localize areas involved in face perception. Such localizers are usually the first step in a fMRI study to identify candidate regions for closer inspection in the main part of the study. They found that using dynamic faces, face-sensitive regions could be much better identified than for static faces. Taking this one step further, a recent study [90] investigated response differences due to the dynamic information in the same group of participants. In addition to the expected activation in typical motion areas (Visual Area 5 and STS), they found that face regions in the ventral pathway that responded to static face stimuli were significantly more active for dynamic stimuli. This suggests that areas that are not traditionally associated with processing motion might already integrate dynamic information in the case of faces. At the very least, these results underline the fact that dynamic faces are the preferred stimulus for the brain.

### 22.2.3 Summary

In sum, it is clear that facial motion—both rigid and nonrigid—plays a role in the learning of new individuals and in the recognition of already learned individuals. This effect is strongest when the images are degraded. That is, the dynamic information helps to compensate for loss of static information. Since normal static images already contain a considerable amount of identity information, this is to be

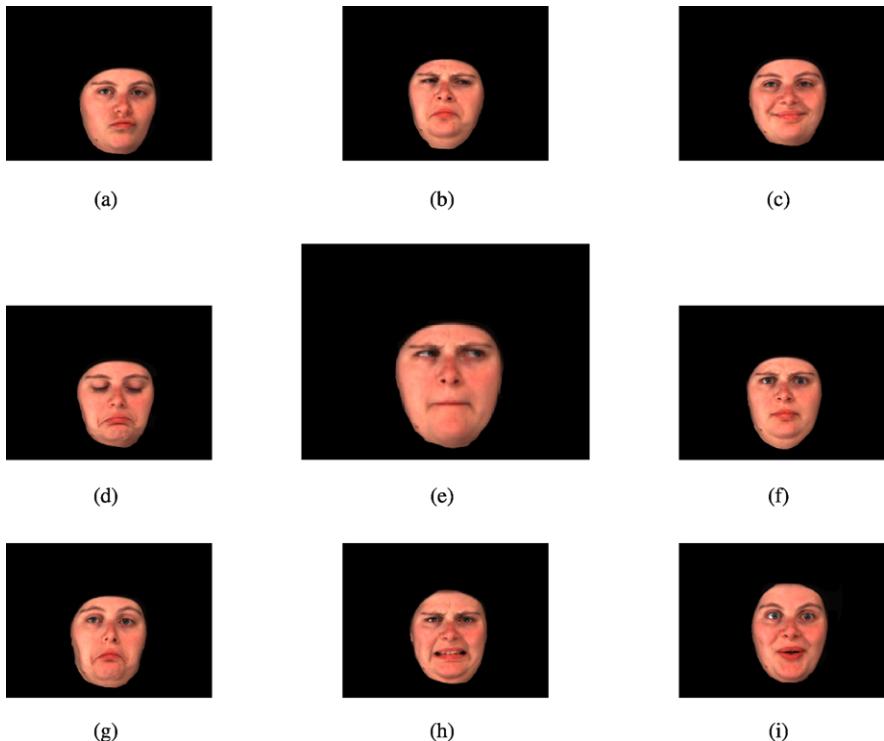
expected for many reasons. It is not surprising that for a phenomena which has multiple sources of information, the removal of one source (e.g., static information) allows the effect of other sources (e.g., dynamic information) to be more clearly seen. Likewise, one might well imagine that the recognition of identity could be near ceiling performance for many types of task. Regardless, it is clear that not only do simple motions (such as horizontal rotations) help us to identify individuals, but that complex, characteristic motions (such as a certain way of smiling) provide distinct information about specific individuals. Future studies will need to clarify exactly what types of facial motions we remember about a person and how these might help us in identification. In addition, it seems that the beneficial effect of motion information is much more pronounced during early development of the perceptual apparatus, as the developmental studies have shown a clearer motion advantage also already for unfamiliar faces. These results highlight the fact that human face perception undergoes a long process of optimization and fine-tuning to let us become experts in face processing.

## 22.3 Dynamic Information for Expressions

Although less studied than identity perception, facial expressions are no less important for everyday life. Compared to other species, humans have developed highly sophisticated communication systems for social interaction. One of the most important of these is based on facial expressions. More specifically, facial expressions are known to serve a wide variety of functions:

- **Meaning:** They can, of course, be used to independently express complex ideas and intentions. For example, someone can look happy, sad, confused, or disgusted (see Fig. 22.2) [8, 9, 16, 29, 34, 62].
- **Modifier:** They are also very useful in *modifying* the meaning of auditory communication [11, 19, 25, 31, 76]. For example, a spoken statement by itself conveys a different meaning than when it is accompanied by a look of boredom. Indeed, in situations where the meaning conveyed by the face differs from that in another communication channel, the face tends to be considered more important [21, 41, 73].
- **Emphasis:** They co-occur with vocal accentuation [32, 79]. This emphasis can be seen, to some degree, in the static snapshot shown in Fig. 22.3.
- **Control:** Listeners can provide a wealth of information to the speaker without ever saying a word (this is referred to as “back-channel” signals; [111]). For example, a properly timed nod of agreement can tell the speaker to continue speaking, while a look of confusion at the same junction of the conversation would indicate that the speaker should stop and try to explain the last point again [10, 12, 18, 22, 23, 56, 86, 102].

Starting at birth, humans are trained to process faces and facial expressions, resulting in a high degree of perceptual expertise for face perception and social communication. This highly trained degree of expertise makes facial expressions—both

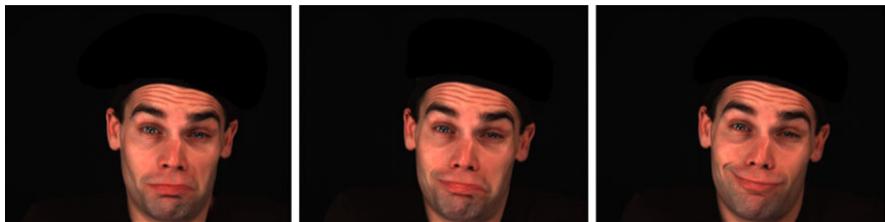
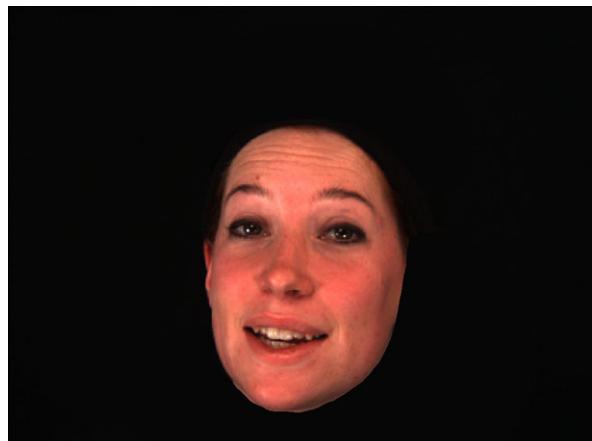


**Fig. 22.2** Static snapshots of several facial expressions. **a** Agreement; **b** Disagreement; **c** Happiness; **d** Sadness; **e** Thinking; **f** Confusion; **g** Cluelessness; **h** Disgust; **i** Surprise. Some of these expressions can be recognized even in a static snap shot. Other expressions, like agreement and disagreement, seem to rely more heavily on dynamic information

emotional and conversational—an extremely difficult topic to study: We can detect very small differences in both motion and meaning. Furthermore, the physical differences between an expression that is recognizable (or is seen as sincere) and one that is not can be very subtle (see, for example, Fig. 22.4). Moreover, there are a number of different ways that humans express any given meaning, and not all of the resulting expressions are easily recognized [27–29]. All these factors jointly make the recognition of facial expressions one of the most difficult tasks the human visual system can perform [7].

Facial expressions have been the topic of scientific examination since at least [30]’s and [33]’s seminal work. These studies, as well as the majority of studies that followed, examined the reaction of people to various facial poses using photographs. Obviously, different facial areas are important for the recognition of different emotions: The mouth is critical for a smile, and the eyes for surprise, etc. [9, 29, 48, 78, 85]. For example, [37] showed that a true smile of enjoyment has not only the characteristic mouth shape, but also specific wrinkles near the eyes whereas faked expressions of enjoyment, in contrast, contain just the mouth information. This also

**Fig. 22.3** A photograph of a facial expression that accompanied vocal emphasis. The facial expression that accompanies a vocal emphasis may be sufficient to recognize the emphatic nature of the statement, even in a static snapshot. This figure is taken from [29]



**Fig. 22.4** Three photographs showing that small physical changes can produce large perceptual changes. These three snapshots were taken from three separate recordings of a clueless expression. All three expressions were recorded within 5 minutes of each other. This figure is adapted from [29]

shows that different facial regions can contribute to the perception of sincerity as well as to the recognition of the underlying expression.

The differential role of facial areas in different expressions is reflected in the fact that most models of facial expressions are explicitly parts-based [36, 38, 39, 43, 44, 57, 72, 101]. For example, Massaro and colleagues have proposed a parts-based model of perception (the fuzzy logical model of perception, or FLMP) in which the features are independently processed and subsequently integrated [38]. In one study, they used computer generated static facial expressions where either (a) the mouth, (b) the eyebrow, or (c) both were parametrically varied. Participants were asked to say if the expression was happiness or anger. Ellison and Massaro found that both features affected the participants' judgments, and that the influence of one feature was more prominent when the other feature was neutral or ambiguous. Moreover, the FLMP captured patterns in the data better than either holistic models or a straight-forward additive model based on recognition rates of the individual features. There are, however, at least two models that integrate holistic (undecomposed whole face) information [58, 109].

Perhaps the most widely used method for *parameterizing* the high-dimensional space of facial expressions is the facial action coding system (or FACS, [36]), which segments the visible effects of facial muscle activity and rigid head motion into “action units”. Combinations of these action units can then be used to describe different expressions. It is important to note that FACS is a system for describing the elements of photographs (or series of photographs) of facial expressions. It is not a model of facial expression processing per se and makes no claims in and of itself about which elements go together to produce different expressions [89].

Regardless of being parts-based, holistic, or hybrid, nearly all models of expression perception focus exclusively on static information. One can, sometimes include some information about the change of the static features over time by looking at the features in every frame, but there is rarely any ability to describe information that is only available over time. One potential exception to this rule is FACS+ from [40]. They used an optic flow technique combined with a few domain-based constraints to estimate facial structure and motion, yielding an empirical model of facial motion and structure. It is unclear, however, how the components of this model relate to the features used by humans. It is nonetheless increasingly clear that there is dynamic information for expressions and that any model of expression processing must take it into account.

Some of the earliest hints at spatiotemporal expression information comes from [8, 9], who used Johansson point-light faces to examine the role of motion in expression recognition (for more on point-light stimuli, see [59]). Their displays consisted of low-light recordings of the face and neck of several actors and actresses performing either an expression (happy, sad, surprise, disgust, interest, fear, and anger) or a random facial motion. The visible parts were painted black and then covered with approximately 100 white spots and the eyes were closed during the recording sessions. Thus, in the final video sequences, all that were visible were the 100 white points, moving about the screen. Each participant saw a single display (to avoid any learning effect or comparison of displays) and was asked to describe what they saw (such a free description task also helped to prevent biasing the participants’ answers). The display was either single static snapshot or a full video recording of one expression. On average, the collection of points was recognized as being a face considerably more often in the dynamic conditions than in the static conditions (73% versus 22% of the time, respectively). That is, the procedure removed nearly all static information that the display was a face as well as any information for more specific facial properties (such as identity or the specific expression). In a second experiment, an additional set of recordings where the face was visible (that is, no makeup was used) was included and participants were asked to identify the expression using a forced choice task (note that in a forced-choice task, a limited set of response options is given and the participant must choose one of these options as the answer). Overall, the expressions were recognized more often in the nonmakeup condition than in the point-light condition (65% versus 33% correct responses, respectively). Critically, more expressions were recognized in the point-light condition than can be expected by pure guessing, suggesting that there is some temporal information for facial expressions. Additional work suggests that even in dynamic

expressions, different expressions rely on different facial regions, with most expressions relying primarily on a single region [9, 29, 80]. Note that although eye, eye-brow, mouth, and rigid head motion are jointly sufficient for accurate recognition of the conversational expressions, other regions do contain information about facial expressions [80].

During the 20 years following Bassili's work, little to no studies examined the perception of dynamic facial expressions. In one of the few exceptions, [37] demonstrated that deceptive expressions of enjoyment appear to have different temporal characteristics than spontaneous ones. At the start of a rebirth in interest in dynamic expressions, [34] conducted an innovative experiment to demonstrate human sensitivity to temporal information. They printed out each frame of a video sequence of an expression. These photographs were given to participants (in a scrambled order), who were asked to place the photographs in the correct order. Participants were remarkably accurate, with particularly strong sensitivity to the temporal characteristics in the early phases of an expression. Interestingly, participants performed better when asked to complete the task with extremely tight time constraints than when given unlimited time, from which Edwards concluded that conscious strategies are detrimental to this task.

Similar to [52]'s and [67]'s work with identity recognition, [62] examined the role of speed in the perception of expressions. Specifically, they manipulated the speed with which a neutral face transitioned into an emotional one. They found that happiness and surprise were better recognized from fast sequences, sadness better from slow sequences, and that anger was best recognized at medium speeds. Subsequently, [53], demonstrated that increasing the distance traveled by an area while holding the timing constant (which also alters the speed and acceleration of the part) can exaggerate the emotional content of sentence. This suggests that different expressions seem to have a characteristic speed or rate of change.

Consistent with the work on identity, most examinations of dynamic expressions used degraded stimuli [3, 26, 35, 49, 60, 77, 105, 107, 108]. It has been consistently shown that dynamic expressions are recognized better than static ones. It has even been shown that the recognition of dynamic expressions with degraded static information can be as good as if not better than static expressions that are not degraded. That is, dynamic information can compensate for the loss of static information [35, 60, 105]. For example, [105] systematically degraded the shape, texture, and motion of a series of computer animated facial expressions. They examined performance in a forced-choice task, and found that dynamic sequences produce higher recognition rates than the static sequences. Likewise, they found that degrading either shape or texture information in the static conditions decreased performance. Critically, all dynamic conditions showed equal performance. That is, the presence of dynamic information eliminated the negative effect of degrading static information (specifically, shape and texture). In a separate experiment, they showed that animations that had proper nonrigid motion but lacked rigid head motion were recognized much worse than expressions that had both rigid and nonrigid motion. Additionally, the absence of rigid head motion greatly increased reaction times (participants were much slower in performing with rigid head motion). Finally, a simple,

temporal linear morph resulted in a small but significant drop in performance from the full motion condition, indicating that not only is motion important, but that natural motion seems to be important.

Just because one can use dynamic information does not, however, mean that one normally uses it. Indeed, in all the studies that have shown a dynamic advantage for expressions, the dynamic and static stimuli generally differed along a number of dimensions (such as the number of images and facial poses). Thus, it is unclear whether the dynamic advantage is due to (spatio)temporal information or something simpler. To help determine whether the dynamic advantage is due to the simple presence of more images, [3] compared recognition performance for a static expression, a dynamic version of that expression, and a condition where a 200 ms Gaussian noise mask was interspersed between the frames of the dynamic sequence. The noise was intended to mask the motion information. The normal dynamic condition resulted in much better performance than either of the other two conditions. Performance in the static condition and the masked dynamic condition did not differ from each other. This latter result confirms that masking does eliminate the perception of motion, as expected. Unfortunately, such a mask is also known to inhibit the processing of static information (see, e.g., backwards masking: [5, 61, 97, 110] or change blindness: [13, 87, 93]). Moreover, the stimuli contained only the early phases of an expression (i.e., the first three to six frames). This means that the static condition was degraded in a particular fashion: it did not contain all of the static information that is present in the peak expression which is used in other expression studies. Thus, it is not clear that the dynamic advantage found in [3]'s experiment would generalize to real-world situations.

Recently, [26] presented a series of 5 experiments conclusively demonstrating the presence of spatiotemporal information for facial expressions. Most of the experiments are strict analogues to the series run by [67, 69, 83] for facial identity. In Experiment 1, [26] directly compared dynamic and static peak versions of nine conversational expressions (see Fig. 22.2), seven of which demonstrated a dynamic advantage (happy and thinking were roughly equivalent in the static and dynamic conditions). This shows that a dynamic advantage can be found for video recordings over peak static images using normal intensity, conversational and emotional expressions of real individuals. The second experiment examined several static explanations for the dynamic advantages. For example, it is possible that the frame chosen in the first experiment was sub-optimal. Perhaps another frame was better, and the presence of this single image in the dynamic condition is the cause of the improved results. Likewise, since the perception of faces is, at least partially, based on its component parts, it is possible that people pick and choose the best parts from different frames, and composite them into a joint static whole (some evidence for this comes from [4]'s work with identity perception). Thus, similar to [69], a shortened dynamic sequence (the last 16 frames) and two static arrays (scrambled and ordered) were compared to the full dynamic and static peak conditions. The full dynamic and 16 frame dynamic conditions, which did not differ from one another, both produced higher recognition rates than the three static conditions (interestingly, the two array conditions produced slightly better performance than the static peak).

These results, combined with those from the third experiment (which scrambled the order of the frames in the dynamic sequence) show that the mere presence of many images or even face-appropriate dynamic information is *not sufficient*. There is some specific information present in the normal temporal development of an expression. Likewise, playing the expressions backwards (Experiment 4) reduced performance. Finally, the fifth experiment demonstrated that performance increases with increases in the number of frames that are kept together (blockwise scrambling). The length of the temporal integration window was at least 100 ms. In sum, dynamic expressions are recognized more easily and more accurately than static expressions, this effect is fairly robust, and the effect cannot be explained by simple, static-based explanations.

### 22.3.1 *Developmental Aspects*

Dynamic information is an even more crucial factor for processing of facial expressions during the perceptual development than it is for face identity [103]. Interestingly, despite a few early studies, there have been only relatively few recent studies that directly highlight the difference between static and dynamic processing of facial expression from a developmental perspective.

In an early study, [103] found that 5 and 7 month-old infants were able to discriminate between dynamically presented happy and angry facial expressions when they were presented with a congruent vocal expression. Following up on this finding, [95] used dynamic happy and angry expressions that were either point-light stimuli or normal faces. The study showed that infants, indeed, preferred the congruent stimuli over the incongruent ones in both conditions, highlighting the fact that the dynamic information in the visual and acoustic domains were integrated—this was especially true, of course, for the point light stimuli which presented much reduced shape information.

One of the most well-known perceptual findings about people suffering from autism spectrum disorder (ASD) is that they seem to have problems in identifying facial expressions. Interestingly, most research on this had been done with photographs, again presenting the participants with “unrealistic” stimuli. In a study comparing ASD and normal children on their performance with static and dynamic presentations of emotional facial expressions, surprisingly few differences between the two groups were found [45]. Interestingly, the authors hypothesized that it was the presence of slow dynamic changes in the stimuli that they had used that gave the autistic group a better chance at processing the stimuli—a result that was recently confirmed [98]. In another study, two groups of ASD and normal children were tested with a more complex set of facial expressions that were presented either statically or dynamically [6]. Although individuals with ASD performed significantly worse than the control group, there was little difference between static and dynamic presentation of expressions, which was most likely due to low, overall recognition performance (see [26]).

Finally, a recent larger, cross-sectional study tested an age range from 4 to 18 years with facial animations portraying different emotions at varying intensity [74]. The study found that the performance of expression recognition increased with age and that it also increased with the intensity with which emotions were animated. Interestingly, performance increased faster for the girls than the boys in the study indicating a gender difference during development of emotional understanding.

In summary, whereas the importance of dynamic information—and with that also the use of dynamic, natural stimuli—for the investigation of facial expression processing has been recognized, more studies are needed to elucidate the development of dynamic expression processing.

### **22.3.2 *Neurophysiological Aspects***

There is some emerging evidence that the neural mechanisms responsible for the perception of expression are at least partially different for static and dynamic facial expressions [1, 54, 66]. These studies include reports from patients who are completely unable to recognize expressions from static pictures or static descriptions, but have normal recognition performance for both dynamic expressions and descriptions of dynamic expressions or actions. A recent neural model for the processing of facial expressions was presented in [1], which details the different areas that might be involved at different points in time. Whereas this model accounts for some dynamic aspects, an integrated model of how the human brain interprets the highly complex dynamic signals from facial expressions is still lacking.

In one of the first studies to use real-world video sequences (rather than, for example, expression morphs such as in [66]), participants observed happy and angry facial expressions in both static and dynamic versions [63]. The study used Positron Emission Tomography (PET) to chart the different neural networks that were involved in perception of the different stimuli. In good agreement with the fMRI studies mentioned earlier, they found a series of typical motion areas to be activated for the dynamic stimuli. Additionally, the found critical differences depending on the expression used. That is, different networks of areas were associated with perception of dynamic happy expressions than with the perception of dynamic angry expressions, and those networks were in turn different from the networks found during perception of static expressions. A recent study [100] extended the stimulus material to include many more actors (thereby avoiding potential habituation effects as in previous studies, which mostly used expressions of only one actor/actress), and contrasted static and dynamic version of neutral, happy, and disgusted expressions using fMRI. Again, the results for the dynamic stimuli indicated a much more widespread network of activation than for the static stimuli. Interestingly, this also included pre-motor areas that are thought to be the human equivalent of the mirror-neuron system and perhaps might be related to motor imagery, or (unconscious) imitation of the observed expression. In addition, the recognition results for dynamic stimuli were found to be better than those for static stimuli [3, 26].

Whereas all studies seem to converge on the fact that facial expression perception in the brain for dynamic stimuli is different from that of static stimuli, a point of criticism still is that stimuli (and also the most prevalent existing models of expression processing [1]) in all cases are based on a few examples from the universal, emotional expressions. It remains to be seen how conversational facial expressions and thus more general facial movements will fit into the overall picture.

## 22.4 Conclusions

It is clear that there is some form of characteristic facial information that is only available over time, and that it plays an important role in the recognition of identity, expression, speech, and gender. It is also clear that the addition of dynamic information improves the recognizability of expressions and identity, and can compensate for the loss of static information. Moreover, at least several different types of motion seem to exist, which play different roles, and a simple rigid/nonrigid dichotomy is neither sufficient nor appropriate to describe. Additional research is necessary to determine what the dynamic features for face processing are.

The sole reliance on static information in any attempt to understand how humans use the face and head to identify or communicate with other people will illuminate only a very limited and maybe even artificial part of the perceptual and cognitive mechanisms involved. Likewise, any system designed to describe or recognize people or facial expressions that does not explicitly allow for the description of dynamic information will never yield human-like performance.

Artificial systems that aim at communicating naturally and efficiently with humans need to be based on a truly spatiotemporal description of the face and communication. Such a description will not only open the door for computer vision systems in biometrics and human-computer interaction, but also enable a more targeted analysis and a potential road to successful therapy and training approaches for patients with both deficits in communication production (such as occurring after a stroke, for example) or in communication understanding (patients with Autism Spectrum Disorder, or Asperger's syndrome, for example).

**Acknowledgements** We gratefully acknowledge the support of the Max Planck Society and the WCU (World Class University) program through the National Research Foundation of Korea funded by the Ministry of Education, Science and Technology (R31-2008-000-10008-0). We are grateful to useful discussions with those who read earlier drafts of the manuscript.

## References

1. Adolphs, R., Tranel, D., Damasio, A.R.: Dissociable neural systems for recognizing emotions. *Brain Cogn.* **52**, 61–69 (2003)
2. Allison, T., Puce, A., McCarthy, G., et al.: Social perception from visual cues: role of the STS region. *Trends Cogn. Sci.* **4**(7), 267–278 (2000)

3. Ambadar, Z., Schooler, J.W., Cohn, J.: Deciphering the enigmatic face: The importance of facial dynamics in interpreting subtle facial expressions. *Psychol. Sci.* **16**, 403–410 (2005)
4. Anaki, D., Boyd, J., Moscovitch, M.: Temporal integration in face perception: Evidence of configural processing of temporally separated face parts. *J. Exp. Psychol. Hum. Percept. Perform.* **22**, 1–19 (2007)
5. Averbach, E., Coriell, A.S.: Short-term memory in vision. *Bell Syst. Tech. J.* **40**, 309–328 (1961)
6. Back, E., Ropar, D., Mitchell, P.: Do the eyes have it? Inferring mental states from animated faces in autism. *Child Dev.* **78**(2), 397–411 (2007)
7. Barton, J.J.S.: Disorders of face perception and recognition. *Neurol. Clin.* **21**, 521–548 (2003)
8. Bassili, J.: Facial motion in the perception of faces and of emotional expression. *J. Exp. Psychol.* **4**, 373–379 (1978)
9. Bassili, J.: Emotion recognition: The role of facial motion and the relative importance of upper and lower areas of the face. *J. Pers. Soc. Psychol.* **37**, 2049–2059 (1979)
10. Bavelas, J.B., Black, A., Lemery, C.R., Mullett, J.: I show how you feel—motor mimicry as a communicative act. *J. Pers. Soc. Psychol.* **59**, 322–329 (1986)
11. Bavelas, J.B., Chovil, N.: Visible acts of meaning—an integrated message model of language in face-to-face dialogue. *J. Lang. Soc. Psychol.* **19**, 163–194 (2000)
12. Bavelas, J.B., Coates, L., Johnson, T.: Listeners as co-narrators. *J. Pers. Soc. Psychol.* **79**, 941–952 (2000)
13. Becker, M., Pashler, H.: Volatile visual representations: Failing to detect changes in recently processed information. *Psychon. Bull. Rev.* **9**, 744–750 (2002)
14. Berry, D.: What can a moving face tell us? *J. Pers. Soc. Psychol.* **58**, 1004–1014 (1990)
15. Berry, D.: Child and adult sensitivity to gender information in patterns of facial motion. *Ecol. Psychol.* **3**, 348–366 (1991)
16. Bruce, V.: Recognising Faces. Lawrence Erlbaum, Hillsdale (1988)
17. Bruce, V., Young, A.: Understanding face recognition. *Br. J. Psychol.* **77**, 305–327 (1986)
18. Bull, P.: State of the art: Nonverbal communication. *The Psychologist* **14**, 644–647 (2001)
19. Bull, R.E., Connolly, G.: Body movement and emphasis in speech. *J. Nonverbal Behav.* **9**, 169–187 (1986)
20. Calder, A.J., Young, A.W.: Understanding the recognition of facial identity and facial expression. *Nat. Rev. Neurosci.* **6**, 641–651 (2005)
21. Carrera-Levillain, P., Fernandez-Dols, J.: Neutral faces in context: Their emotional meaning and their function. *J. Nonverbal Behav.* **18**, 281–299 (1994)
22. Cassell, J., Thorisson, K.R.: The power of a nod and a glance: Envelope vs. emotional feedback in animated conversational agents. *Appl. Artif. Intell.* **13**, 519–538 (1999)
23. Cassell, J., Bickmore, T., Cambell, L., Vilhjalmsson, H., Yan, H.: More than just a pretty face: conversational protocols and the affordances of embodiment. *Knowl.-Based Syst.* **14**, 22–64 (2001)
24. Christie, F., Bruce, V.: The role of dynamic information in the recognition of unfamiliar faces. *Mem. Cogn.* **26**, 780–790 (1998)
25. Condon, W.S., Ogston, W.D.: Sound film analysis of normal and pathological behaviour patterns. *J. Nerv. Ment. Dis.* **143**, 338–347 (1966)
26. Cunningham, D.W., Wallraven, C.: Temporal information for the recognition of conversational expressions. *J. Vis.* **9**, 1–17 (2009)
27. Cunningham, D.W., Breidt, M., Kleiner, M., Wallraven, C., Bühlhoff, H.: The inaccuracy and insincerity of real faces. In: Proceedings of Visualization, Imaging, and Image Processing (2003)
28. Cunningham, D.W., Breidt, M., Kleiner, M., Wallraven, C., Bühlhoff, H.H.: How believable are real faces?: Towards a perceptual basis for conversational animation. In: Computer Animation and Social Agents 2003, pp. 23–29 (2003)
29. Cunningham, D.W., Kleiner, M., Wallraven, C., Bühlhoff, H.H.: Manipulating video sequences to determine the components of conversational facial expressions. *ACM Trans. Appl. Percept.* **2**(3), 251–269 (2005)

30. Darwin, C.: *The Expression of the Emotions in Man and Animals*. John Murray, London (1872)
31. DeCarlo, D., Revilla, C., Stone, M.: Making discourse visible: Coding and animating conversational facial displays. In: *Proceedings of the Computer Animation 2002*, pp. 11–16 (2002)
32. Dohen, M., Loevenbruck, Cathiard, M.-A., Schwartz, J.-L.: Audiovisual perception of contrastive focus in French. In: *Proceedings of the AVSP'03 Conference*, pp. 245–250 (2003)
33. Duchenne, B.: *The Mechanism of Human Facial Expression or an Electro-Physiological Analysis of the Expression of the Emotions*. Cambridge University Press, New York (1862/1990)
34. Edwards, K.: The face of time: Temporal cues in facial expressions of emotion. *Psychol. Sci.* **9**, 270–276 (1998)
35. Ehrlich, S.M., Schiano, D.J., Sheridan, K.: Communicating facial affect: It's not the realism, it's the motion. In: *Proceedings of the ACM CHI 2000 Conference on Human Factors in Computing Systems*, pp. 252–253. ACM Press, New York (2000)
36. Ekman, P., Friesen, W.: *Facial Action Coding System*. Consulting Psychologists Press, Palo Alto (1978)
37. Ekman, P., Friesen, W.V.: Felt, false, and miserable smiles. *J. Nonverbal Behav.* **6**, 238–252 (1982)
38. Elison, J.W., Massaro, D.W.: Featural evaluation, integration, and judgment of facial affect. *J. Exp. Psychol. Hum. Percept. Perform.* **23**, 213–226 (1997)
39. Essa, I., Pentland, A.: A vision system for observing and extracting facial action parameters. In: *CVPR94*, pp. 76–83 (1994)
40. Essa, I., Pentland, A.: Coding, analysis, interpretation, and recognition of facial expressions (1997)
41. Fernandez-Dols, J., Wallbott, H., Sanchez, F.: Emotion category accessibility and the decoding of emotion from facial expression and context. *J. Nonverbal Behav.* **15**, 107–124 (1991)
42. Fox, C., Iaria, G., Barton, J.: Defining the face processing network: Optimization of the functional localizer in fMRI. *Hum. Brain Mapp.* **30**(5) (2009)
43. Frijda, N.H., Philipszoon, E.: Dimensions of recognition of emotion. *J. Abnorm. Soc. Psychol.* **66**, 45–51 (1963)
44. Frois-Wittmann, J.: The judgment of facial expression. *J. Exp. Psychol.* **13**, 113–151 (1930)
45. Gepner, B., Deruelle, C., Grynfelth, S.: Motion and emotion: A novel approach to the study of face processing by young autistic children. *J. Autism Dev. Disord.* **31**(1), 37–45 (2001)
46. Gibson, J.J.: *The Ecological Approach to Visual Perception*. Lawrence Erlbaum, Hillsdale (1979)
47. Giese, M., Poggio, T.: Neural mechanisms for the recognition of biological movements. *Nat. Rev., Neurosci.* **4**(3), 179–192 (2003)
48. Hanawalt, N.: The role of the upper and lower parts of the face as the basis for judging facial expressions: II. in posed expressions and “candid camera” pictures. *J. Gen. Psychol.* **31**, 23–36 (1944)
49. Harwood, N., Hall, L., Shinkfield, A.: Recognition of facial emotional expressions from moving and static displays by individuals with mental retardation. *Am. J. Ment. Retard.* **104**, 270–278 (1999)
50. Haxby, J.V., Gobbini, M.I., Furey, M., Ishai, A., Schouten, J., Pietrini, P.: Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science* **293**, 2425–2430 (2001)
51. Hill, H., Johnston, A.: Categorization and identity from the biological motion of faces. *Curr. Biol.* **11**, 880–885 (2001)
52. Hill, H., Johnston, A.: Categorizing sex and identity from the biological motion of faces. *Curr. Biol.* **11**, 880–885 (2001)
53. Hill, H., Troje, N.F., Johnston, A.: Range- and domain-specific exaggeration of facial speech. *J. Vis.* **5**, 793–807 (2005)
54. Humphreys, G., Donnelly, N., Riddoch, M.: Expression is computed separately from facial identity, and is computed separately for moving and static faces: Neuropsychological evidence. *Neuropsychologia* **31**, 173–181 (1993)

55. Hunnius, S., Geuze, R.: Gaze shifting in infancy: A longitudinal study using dynamic faces and abstract stimuli. *Infant Behav. Dev.* **27**(3), 397–416 (2004)
56. Isaacs, E., Tang, J.: What video can and can't do for collaboration: a case study. In: ACM Multimedia '93, pp. 496–503. ACM, New York (1993)
57. Izard, C.E.: The maximally discriminative facial movement coding system (max). Available from Instructional Resource Center, University of Delaware. Newark, DE (1979)
58. Izard, C.E., Dougherty, L.M., Hembree, E.A.: A system for identifying affect expressions by holistic judgments (1983)
59. Johansson, G.: Visual perception of biological motion and a model for its analysis. *Percept. Psychophys.* **14**, 201–211 (1973)
60. Kaetsyri, J., Klucharev, V., Frydrych, M., Sams, M.: Identification of synthetic and natural emotional facial expressions. In: International Conference on Audio-Visual Speech Processing (AVSP 2003), pp. 239–243 (2003)
61. Kahneman, D.: Method, findings, and theory in studies of visual masking. *Psychol. Bull.* **70**, 404–425 (1968)
62. Kamachi, M., Bruce, V., Mukaida, S., Gyoba, J., Yoshikawa, S., Akamatsu, S.: Dynamic properties influence the perception of facial expressions. *Perception* **30**, 875–887 (2001)
63. Kilts, C., Egan, G., Gideon, D., Ely, T., Hoffman, J.: Dissociable neural pathways are involved in the recognition of emotion in static and dynamic facial expressions. *NeuroImage* **18**(1), 156–168 (2003)
64. Knappmeyer, B., Thornton, I., Bühlhoff, H.: The use of facial motion and facial form during the processing of identity. *Vis. Res.* **43**(18), 1921–1936 (2003)
65. Knight, B., Johnson, A.: The role of movement in face recognition. *Vis. Cogn.* **4**, 265–273 (1997)
66. LaBar, K., Crupain, M.J., Voyvodic, J.T., McCarthy, G.: Dynamic perception of facial affect and identity in the human brain. *Cereb. Cortex* **13**, 1023–1033 (2003)
67. Lander, K., Bruce, V.: Recognizing famous faces: exploring the benefits of facial motion. *Ecol. Psychol.* **12**, 259–272 (2000)
68. Lander, K., Chuang, L.: Why are moving faces easier to recognize? *Vis. Cogn.* **12**, 429–442 (2005)
69. Lander, K., Christie, F., Bruce, V.: The role of movement in the recognition of famous faces. *Mem. Cogn.* **27**, 974–985 (1999)
70. Lander, K., Bruce, V., Hill, H.: Evaluating the effectiveness of pixelation and blurring on masking the identity of familiar faces. *Appl. Cogn. Psychol.* **15**, 101–116 (2001)
71. Lander, K., Chuang, L., Wickam, L.: Recognizing face identity from natural and morphed smiles. *Q. J. Exp. Psychol.* **59**, 801–808 (2006)
72. Leventhal, H., Sharp, E.: Facial expression as indicators of distress. In: Tomkins, S.S., Izard, C.E. (eds.) *Affect, Cognition and Personality: Empirical Studies*, pp. 296–318. Springer, New York (1965)
73. Mehrabian, A., Ferris, S.: Inference of attitudes from nonverbal communication in two channels. *J. Consult. Psychol.* **31**, 248–252 (1967)
74. Montirossi, R., Peverelli, M., Frigerio, E., Crespi, M., Borgatti, R.: The development of dynamic facial expression recognition at different intensities in 4- to 18-year-olds. *Soc. Dev.* **19**(1), 71–92 (2010)
75. Mori, M.: *Bukimi no tani*. Energy **7**, 33–35 (1970) [The uncanny valley, (K.F. Macdorman and T. Minato, transl.)]
76. Motley, M.T.: Facial affect and verbal context in conversation—facial expression as interjection. *Hum. Commun. Res.* **20**, 3–40 (1993)
77. Munhall, K.G., Jones, J.A., Callan, D.E., Kuratake, T., Vatikiotis-Bateson, E.: Visual prosody and speech intelligibility: head movement improves auditory speech perception. *Psychol. Sci.* **15**, 133–137 (2004)
78. Numenmaa, T.: The language of the face. In: Jyväskylä Studies in Education, Psychology, and Social Research, Jyväskylä, Finland (1964). Jyväskylän Yliopistoydystys
79. Nusseck, M., Cunningham, D.W., Ruiter, J.P.D., Wallraven, C.: Perception of emphasis intensity in audio-visual speech. *Speech Commun.* 11 under review

80. Nusseck, M., Cunningham, D.W., Wallraven, C., Bülthoff, H.H.: The contribution of different facial regions to the recognition of conversational expressions. *J. Vis.* **8**(8:1), 1–23 (2008)
81. O'Toole, A., Roak, D.: Memory for moving faces: The interplay of two recognition systems. In: Giese, M., Curio, C., Bülthoff, H. (eds.) *Dynamic Faces: Insights from Experiments and Computation*. MIT Press, Cambridge (2009)
82. Otsuka, Y., Konishi, Y., Kanazawa, S., Yamaguchi, M., Abdi, H., O'Toole, A.: Recognition of moving and static faces by young infants. *Child Dev.* **80**(4), 1259–1271 (2009)
83. Pike, G., Kemp, R., Towell, N., Phillips, K.: Recognizing moving faces: The relative contribution of motion and perspective view information. *Vis. Cogn.* **4**, 409–437 (1997)
84. Pilz, K.S., Thornton, I.M., Bülthoff, H.H.: A search advantage for faces learned in motion. *Exp. Brain Res.* **171**(4), 436–447 (2006)
85. Plutchik, R.: *The Emotions: Facts, Theories, and a New Model*. Random House, New York (1962)
86. Poggi, I., Pelachaud, C.: Performative facial expressions in animated faces. In: Cassell, J., Sullivan, J., Prevost, S., Churchill, E. (eds.) *Embodied Conversational Agents*, pp. 115–188. MIT Press, Cambridge (2000)
87. Rensink, R.A., O'Regan, J.K., Clark, J.J.: To see or not to see: the need for attention to perceive changes in scenes. *Psychol. Sci.* **8**, 368–373 (1997)
88. Roark, D.A., O'Toole, A.J., Abdi, H., Barrett, S.E.: Learning the moves: The effect of familiarity and facial motion on person recognition across large changes in viewing format. *Perception* **35**, 761–773 (2006)
89. Sayette, M.A., Cohn, J.F., Wertz, J.M., Perrott, M.A., Parrott, D.J.,: A psychometric evaluation of the facial action coding system for assessing spontaneous expression. *J. Nonverbal Behav.* **25**, 167–186 (2001)
90. Schultz, J., Pilz, K.: Natural facial motion enhances cortical responses to faces. *Exp. Brain Res.* **194**(3), 465–475 (2009)
91. Schwaninger, A., Wallraven, C., Cunningham, D.W., Chiller-Glaus, S.D.: Processing of facial identity and expression: A psychophysical, physiological and computational perspective. *Progress in Brain Research*, 325–348 (2006)
92. Schwarzer, G., Leder, H.: *The Development of Face Processing*. Hogrefe & Huber, Cambridge (2003)
93. Simons, D.J., Levin, D.T.: Failure to detect changes to people during a real-world interaction. *Psychon. Bull. Rev.* **5**, 644–649 (1998)
94. Skelton, F., Hay, D.: Do children utilize motion when recognizing faces? *Vis. Cogn.* **16**(4), 419–429 (2008)
95. Soken, N., Pick, A.: Intermodal perception of happy and angry expressive behaviors by seven-month-old infants. *Child Dev.* **63**(4), 787–795 (1992)
96. Spencer, J., O'Brien, J., Johnston, A., Hill, H.: Infants' discrimination of faces by using biological motion cues. *Perception* **35**(1), 79 (2006)
97. Stigler, R.: Chronophotische Studien ber den Umgebungskontrast (Effects of exposure duration and luminance on the contrast of the surround). *Pflügers Arch. Gesamte Physiol. Menschen Tiere* **134**, 365–435 (1910)
98. Tardif, C., Laine, F., Rodriguez, M., Gepner, B.: Slowing down presentation of facial movements and vocal sounds enhances facial expression recognition and induces facial–vocal imitation in children with autism. *J. Autism Dev. Disord.* **37**(8), 1469–1484 (2007)
99. Thornton, I., Kourtzi, Z.: A matching advantage for dynamic human faces. *Perception* **31**, 113–132 (2002)
100. Trautmann, S., Fehr, T., Herrmann, M.: Emotions in motion: Dynamic compared to static facial expressions of disgust and happiness reveal more widespread emotion-specific activations. *Brain Res.* **1284**, 100–115 (2009)
101. Tronick, E., Als, H., Brazelton, T.B.: Monadic phases: A structural descriptive analysis of infant-mother face-to-face interaction. *Merrill-Palmer Q. Behav. Dev.* **26**, 3–24 (1980)
102. Vertegaal, R.: Conversational awareness in multiparty vmc. In: *Extended Abstracts of CHI'97*, pp. 496–503. ACM, Atlanta (1997)

103. Walker-Andrews, A.: Infants' perception of expressive behaviors: Differentiation of multi-modal information. *Psychol. Bull.* **121**, 437–456 (1997)
104. Wallis, G., Bülthoff, H.H.: Effects of temporal association on recognition memory. *Proc. Natl. Acad. Sci. USA* **98**, 4800–4804 (2001)
105. Wallraven, C., Bülthoff, H.H., Fischer, J., Cunningham, D.W., Bartz, D.: Evaluation of real-world and computer-generated stylized facial expressions. *ACM Trans. Appl. Percept.* **4**, 1–24 (2007)
106. Wallraven, C., Breidt, M., Cunningham, D.W., Bülthoff, H.H.: Evaluating the perceptual realism of animated facial expressions. *ACM Trans. Appl. Percept.* **4**, 1–20 (2008)
107. Wehrle, T., Kaiser, S., Schmidt, S., Schere, K.R.: Studying the dynamics of emotional expressions using synthesized facial muscle movements. *J. Pers. Soc. Psychol.* **78**, 105–119 (2000)
108. Weyers, P., Mühlberger, A., Hefele, C., Pauli, P.: Electromyographic responses to static and dynamic avatar emotional facial expressions. *Psychophysiology* **43**, 450–453 (2006)
109. White, M.: Parts and wholes in expression recognition. *Cogn. Emot.* **14**, 39–60 (2000)
110. Williams, M., Breitmeyer, B., Lovegrove, W., Gutierrez, L.: Metacontrast with masks varying in spatial frequency and wavelength. *Vis. Res.* **31**, 2017–2023 (1991)
111. Yngve, V.H.: On getting a word in edgewise. In: Papers from the Sixth Regional Meeting of the Chicago Linguistic Society, pp. 567–578. Chicago Linguistic Society, Chicago (1970)

# Chapter 23

## Face Recognition by Humans and Machines

Alice J. O'Toole

### 23.1 Introduction

We recognize individual human faces dozens of times each day, with seemingly minimal effort. Our repertoire of known faces includes those of our family, friends, coworkers, acquaintances, and the many familiar faces we see in the news and entertainment media. At its best, human face recognition is highly robust to changes in viewing angle and illumination. It is also robust across the set of nonrigid face deformations that define emotional expressions and speech movements. Humans can even manage, in some cases, to recognize faces over decade-long lapses between encounters. Over time spans early in life, the head structure and facial features change markedly as children grow into adolescents and adults. Later in life, faces can age in ways that alter both the shape of the facial surface and the texture of the skin, as well as the color of the hair and eyebrows. By almost any current measure of the complexity of the computational vision problems solved to accomplish this task, the human face recognition system is impressive.

The description of human face recognition skills just offered is a *best case scenario* and one that is true only for the faces of people we know reasonably well (e.g., friends and family) or have seen many times in the popular media (e.g., Barack Obama, Angelina Jolie). For the faces of people we know from only a single or small number of encounters, human performance is not similarly robust. This is abundantly clear from the mundane day-to-day mistakes we make, and more critically, from the many well-documented cases of the fallibility of eyewitness identifications. In this latter case, a person may be seen initially under sub-optimal viewing conditions. A witness may then be asked days, weeks, or even months later to make an identification. Human recognition can be highly prone to error in these cases [9, 10].

---

A.J. O'Toole (✉)

School of Behavioral and Brain Sciences, The University of Texas at Dallas,  
800 W. Campbell Rd., Richardson, TX 75083-0688, USA  
e-mail: [otoole@utdallas.edu](mailto:otoole@utdallas.edu)

A key issue both for computer vision researchers and for psychologists is to understand how it is possible to create a representation of faces that achieves the kind of robust face recognition people show when they know someone well. Understanding the changes that occur in the quality of a face representation as a newly learned face becomes increasingly familiar over time may offer insight into the critical problems that still challenge even the best available computer-based face recognition systems. It may also help us to understand the conditions under which human face recognition is reliable. We will conclude, ultimately, that the best current face recognition algorithms are well on their way to achieving the recognition ability humans show for unfamiliar faces, but have a long way to go before they can compete with humans in the best case scenario.

In the first part of this chapter, I will describe the characteristics of human face processing, emphasizing psychological studies that illustrate the nature of human face representations. We begin with a brief overview of the multiple tasks humans do with faces, including identification, categorization, and expression perception. Next we will look at some characteristics of the human representation of faces that distinguish it from representations used in machine vision. In particular, we focus on the advantages of norm-based coding for identification tasks. This optimizes the feature dimensions used to code faces, but may have a cost in generalization to faces that are not described by the derived sets of features (e.g., faces of “other” races or ethnicities).

In the second part of the chapter, I will discuss a series of recent studies that compare human performance to the performance of state-of-the-art face recognition systems. One goal of these comparisons is to establish human benchmarks for the performance of face recognition algorithms. A second goal is to understand the strengths and weaknesses of humans and machines at the task of face recognition and to come up with strategies for optimally combining human and machine recognition decisions. A third aim of these studies is to understand qualitative similarities and differences in the pattern of errors made by humans and machines. We will argue that studying human face recognition in this way can help us to anticipate and mitigate the errors made by both human and computer-based systems.

## 23.2 What Humans Do with Faces

Perhaps the most remarkable aspect of the human face is the diversity of information it provides simultaneously to the human observer for solving different “tasks”. These tasks include recognition, identification, visually-based categorization (e.g., sex, race, age), and emotional/facial expression perception. The challenge for the human system is to extract and apply the information needed for the task at hand. As we will see, the coexistence of identity information (e.g., distinctive facial features) and social information (e.g., smiles, head turns, etc.) in the human face makes the problem of recognition even more challenging.

### 23.2.1 Recognition and Identification

Each human face is unique and, as such, provides information about the identity of its owner. Humans can keep track of hundreds (if not thousands) of individual faces. This far exceeds our ability to memorize individual exemplars from any other class of objects (e.g., How many individual suitcases can we remember?). For psychologists, recognition refers to the judgment that we have seen a particular face before. Identification assumes recognition, but with the added burden of being able to label the face with a name or context (e.g., the sales clerk at the grocery store). For humans, recognition can occur with a high degree of confidence and accuracy, even with no ability to supply a name or context. We have all had the experience of seeing someone, being certain you “know” them, but having no idea who they are or where you met them previously. The separability of recognition from identification success highlights the fact that, for humans, faces are coded perceptually, and that this perceptual code can be activated and remembered without reference to other semantic information like a name. This characteristic, of course, differs for machines in that the “recognition” of a face by computer necessarily implies the retrieval of a tag or label for the person.

Returning to the question of the visual information that is extracted and encoded for later recognition, it is clear that to identify a face we must locate information that makes the face *unique* or different from all other faces we have seen before and from all other unknown faces. Humans deal with this problem by coding faces relative to an average or *prototype* face. The prototype face is likely derived from the history or experience a person has with faces. In this sense, people with different experience profiles (e.g., with faces of different races) will have different prototype faces. We discuss the advantages and consequences of this relativistic coding strategy in Sect. 23.3.1. For now, we simply note this as an important characteristic of human face recognition that sets it apart from most approaches in computer vision.

### 23.2.2 Visually Based Categorization

In addition to our ability to recognize and identify faces, humans can also categorize faces along a number of dimensions referred to as *visually-derived semantic categories* [7], including sex, ethnicity/race, and age. By a broader definition, one can also include other visually-specified, albeit abstract, categories such as personality characteristics. For example, humans routinely make judgments about whether a face looks “generous” or “competent” or “extroverted”. Faces can be categorized quickly and effortlessly based on a host of social and personality dimensions. An intriguing aspect of the human tendency to make these types of judgments about faces is that the act of judging actually *increases* human accuracy at recognizing faces. Specifically, recognition is better when people are asked to make social judgments about a face as they are learning (e.g., “Is this person extroverted?”), as compared to when they make physical feature-based judgments (e.g., “Does this person have

a big nose?”) [4]. To date, judging faces for social and personality traits is still one of the best ways to improve human accuracy at face recognition.

Although the trait judgments we make about people from their faces do not predict a person’s social and personality characteristics *cite someone*, they nonetheless have real predictive power for other important decisions. Todorov and colleagues showed recently that judgments of “competence”, made only from a black and white picture of a face, predicted the outcome of U.S. Senate and House of Representatives elections at levels well above chance [33]. In that study, people were asked to answer the following questions about pairs of face images taken from actual election races. *Which person is the more competent?* The researchers eliminated data for any trials where the subject knew the candidates. Remarkably, naive subjects’ answer to this question predicted the outcome of 71.6% of United States Senate races and 66.8% of the House of Representative races.

This effect was replicated subsequently by researchers in Switzerland, who extended the finding to show that children’s judgments of faces predicted the outcomes of French parliamentary run-off elections in 2002 [2]. In that study, Swiss children between the ages of 5 and 13 years decided, “Which of these people do you want to be the captain of your boat?”. They found that the probability of predicting an election outcome based on the children’s choice of preferred captain was 0.71, similar to the predictive power of the adult judgment’s in the study by Todorov et al. [33]. The children’s judgments even predicted the margin of victory in the races.

Combined, these studies indicate a willingness to categorize faces along both physical and social dimensions. They further suggest the availability of categorical information in faces that humans perceive in a similar way as predictive of personal and social traits. More important, these perceptual judgments affect human face recognition accuracy and the decisions we make about people, from our simplest expectations of a person’s approachability to our trust in their ability to govern. Moreover, the information we extract from faces to make these categorical judgments is readily available even to children, who presumably know less about the social structure of the world.

### 23.2.3 Expression Processing

Facial expressions available on the human face provide others with information about our emotional state and social intent. Expressions of fear, happiness, sadness, disgust, surprise, and anger are universally interpretable as conveying information about the internally felt emotions of another person [13]. The universal nature of these expressions indicates that the information that specifies them is shared across all human faces. Because facial expressions are made by muscle movements that cause nonrigid deformations of the shape of facial features and configurations, in principle, they can complicate the problem of face recognition. For humans, there is surprisingly little data on the effect of expression change on face recognition accuracy, though anecdotal information suggests that for faces we know well, the effects

of expression are minimal. The effects for less familiar faces are not known, though it is likely that expression change may make recognition less accurate.

For familiar faces, one reason that expression change may have minimal effect on human abilities to recognize faces is that by most theoretical and neural accounts, the processing of facial expression by humans is at least partially independent from the processing of identity information in the face [7, 15]. Support for the neural side of this claim comes from long-standing observations in brain-damaged patients indicating that neural lesions can affect the identity processing system or the expression processing system *selectively*. Thus, there are documented cases of *prosopagnosia* (a complete inability to recognize people by their faces, with no other difficulties with object recognition) following brain damage, with spared ability to recognize facial expressions [37]. Concomitantly, there are documented cases of impaired expression recognition following brain damage with spared ability to recognize faces [1]. Although these observations have suggested the independence of the identity and expression-processing systems, the completeness of this separation has been questioned in recent years [11]. Notwithstanding, the implication is that the human system may process expression and identity in parallel systems with different goals. Moreover, current neural models allocate these two processes into separate systems that analyze facial motions including expression, gaze and facial speech, and those that analyze the static invariant structure of the face and its features for categorization and identification [15]. See Chap. 22 for more detail on moving faces.

### 23.3 Characteristics of Human Recognition

The characteristics of human face recognition have been studied by psychologists for decades. In this chapter, we focus on the characteristic of representations that may most differentiate human face recognition from machine recognition. Specifically, the use of relative rather than absolute face codes.

#### 23.3.1 Norm-Based Coding

There is strong evidence that humans code individual faces *relative* to a prototype or average face, rather than in absolute terms. This type of code directly captures how individual faces *differ* from the average face. Thus, we refer to the representation as “norm-based”. A norm-based code has several important advantages over absolute codes, including its neural efficiency and adaptability to the perceiver’s local environment of faces. The code also has some pitfalls which we consider subsequently. Critically, however, understanding the way humans represent faces provides us with a way of anticipating the types of errors we make.

Evidence for the human use of prototypes to encode faces comes from three types of findings: (a) the finding that typical/attractive faces are recognized less accurately

than distinctive or unusual faces [21]; (b) the effectiveness of caricatured faces for improving recognition (cf. [32]); and (c) recent effects on “perceptual adaptation” of faces [19, 35, 36]. All three of these findings can be understood in the context of a human representation of faces in a metaphorical *face subspace*, with an origin at the average face and with axes representing the features along which faces differ (i.e., shape, pigmentation, etc.). Each individual face can be thought of as a point in the face subspace, with feature values specified by the coordinates of their projections onto the axes that define the space. Faces close in the space appear similar and faces more distant from each other appear less similar.

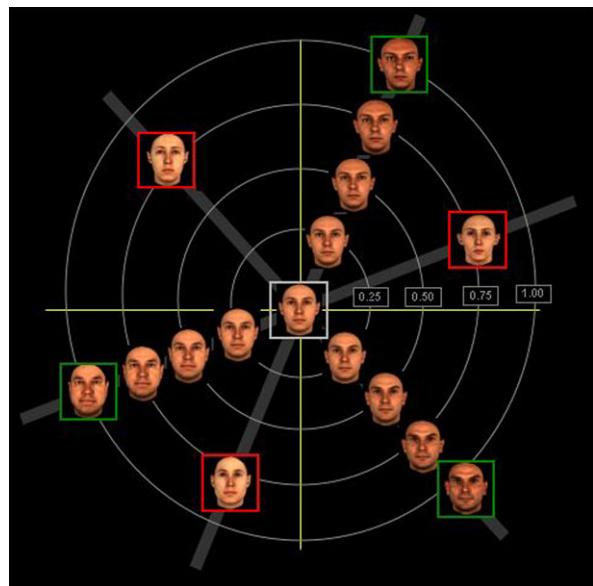
It is perhaps worth digressing briefly to present a more detailed *geography* of a face subspace. For purposes of illustration, we use the face subspace implemented in the three-dimensional morphing software developed by Blanz and Vetter [3]. That software creates a face subspace using three-dimensional laser scans from approximately 200 faces. These scans sample the surface and texture (reflectance) of the face with high resolution. The face subspace is created as follows. Faces are first put into correspondence with the average face. Individual faces are then coded in terms of their deformation from the average. Because the representation contains both surface and reflectance samples, each sample contains information about how the face differs in shape ( $\delta x$ ,  $\delta y$ ,  $\delta z$ ) and reflectance ( $\delta r$ ,  $\delta g$ ,  $\delta b$ ) from the corresponding sample point on the average face. Next, the axes of the space are calculated using a principal components analysis (PCA) of the face codes. This yields a high dimensional face subspace that can be used to describe a very large number of faces. The face subspace created in this morph software is, in some ways, a computational implementation of the norm-based face codes used by humans.

Figure 23.1 shows a simplified two-dimensional schematic of the resultant space, with a few sample faces. At the center is the average face. Individual faces are represented as directions through the (multidimensional space). Each of the original faces appears with a green box around it. Along the line between the original face and the average face is a morph line referred to as the *identity trajectory*, because the faces on this line retain the identity of the original. As the line approaches the average, the faces appear progressively less distinctive. These faces are referred to as *anti-caricatures*. Continuing along this line to the other-side of the mean at the opposite end of the line is the *anti-face*, which is a kind of “opposite” of the original face.

Returning to the question of how we see evidence for a norm-based face code in humans, let’s consider the typicality effect, caricature perception, and perceptual adaptation. We will make reference to the face subspace model just defined to help to clarify these effects.

**Typicality** It has been known for some time that faces rated by humans as “typical” are recognized less accurately than faces rated as “unusual”. The negative correlation between the typicality and “recognizability” of faces is one of the most robust findings in the face recognition literature (e.g., [21]). Moreover, much of the correlation between typicality and recognizability is due to a strong positive correlation between typicality and false alarm rate. In other words, highly typical faces are

**Fig. 23.1** A face subspace made using laser scans of human heads that are in correspondence with the average face [3]. The average face is at the center, individual trajectories in the space progress from the average to veridical faces (*green boxes*) with less distinct anti-caricatures in between. The anti-faces (*red boxes*) are physical opposites of the originals that lie on the other side of the average face. Faces in this space are represented in terms of their shape and reflectance deviations from the average face

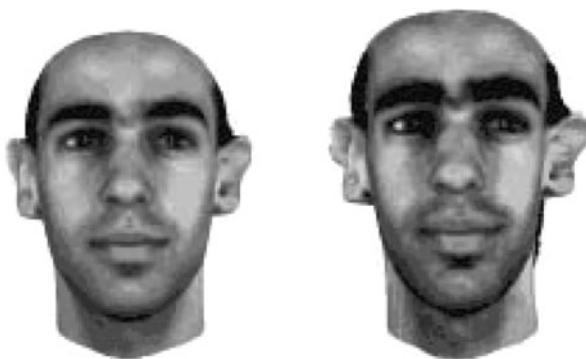


falsely recognized as “known” more frequently than less typical faces. In general, the face subspace model explains the typicality-recognizability relationship in terms of the density of faces closer to the average face. The assumption is that faces are distributed normally around the average, with typical faces close to the average and more distinctive faces farther away. Typical faces are difficult to remember because they are in the densest region of the space and are thus easily confused with other faces.

More theoretically, for human face recognition, the relationship between face typicality and recognizability points to a norm-based coding because typicality judgments inherently reference a standard face. Thus, when a human observer judges a face to be unusual or distinctive, nominally it might be because “the nose is too big”, or “the eyes are too close together”. It is clear, however, that implicit in these judgments is a reference to internalized knowledge about how long a nose *should be* or how close together eyes *should be*. As such, the typicality-recognizability finding has been interpreted as evidence that human observers store a representation of the average or *prototype* face, against which all other faces are compared [34]. This suggests that individual faces are represented, not in absolute terms, but in relative terms.

The finding is relevant for predicting face recognition success for human observers at the level of individual faces. Moreover, it suggests that some faces are more or less likely, on average, to be successfully recognized. In that sense, the human finding is related to work by Doddington who proposes that individual items in a biometric database can be categorized by their susceptibility to different kinds of recognition errors (e.g., false alarms) [12]. Doddington refers metaphorically to a categorization of individual items into types of “animals in a zoo” (e.g., goats are “difficult to match”, “wolves are good impersonators”, etc.). We will not explore

**Fig. 23.2** Original face (*left*) and caricature (*right*) made with both the shape and reflectance information in the face [3]



the connection between the computational and psychological theories in detail here. Instead, we use this as an example of how the performance of human and machine recognition systems can be analyzed and understood in related terms. Moreover, the major conclusion we draw here is that item characteristics, as well as viewing conditions (i.e., illumination, viewpoint, etc.), can pose special challenges to both human and machine-based face recognition. These item characteristics cannot be understood in absolute terms, but rather, they stem from the properties of the item relevant to a population of “distractor items”.

**Caricatures** The notion of a caricaturing simply extends the typicality effect beyond face variations that occur naturally, to a physical distortion of a face that moves it away from the average face in a particular direction. When an artist draws a caricature of a person, they do so by exaggerating the person’s distinctive or unusual features. Thus, a caricature of a person with a large nose, invariably portrays the individual with an even larger nose. People with eyes “too close together”, end up with eyes *much too close* together. Distinguishing features become further exaggerated and thereby more distinguishing and valuable for identifying a person. An example of a computer-generated caricature, based on the three-dimensional morphing software developed by Blanz and Vetter appears in Fig. 23.2. Note that in the context of the face subspace presented in Fig. 23.1, this caricatured face would be at a position along the identity trajectory, farther away from the average face than the original face.

From a psychological point of view, the paradox for human perception and memory is that caricatures are often a gross distortion of the human face, and yet we see them as a “good likeness”. Moreover, under some circumstances, we recognize people more accurately and efficiently from caricatures (for a review [32]). In the context of a norm-based face code and a face subspace, the paradox is easy to understand. A caricatured face remains on the identity trajectory, retaining its direction away from the average in the space, but is pulled farther away from other potentially similar distractor faces. It is therefore less likely to be confused with other faces.

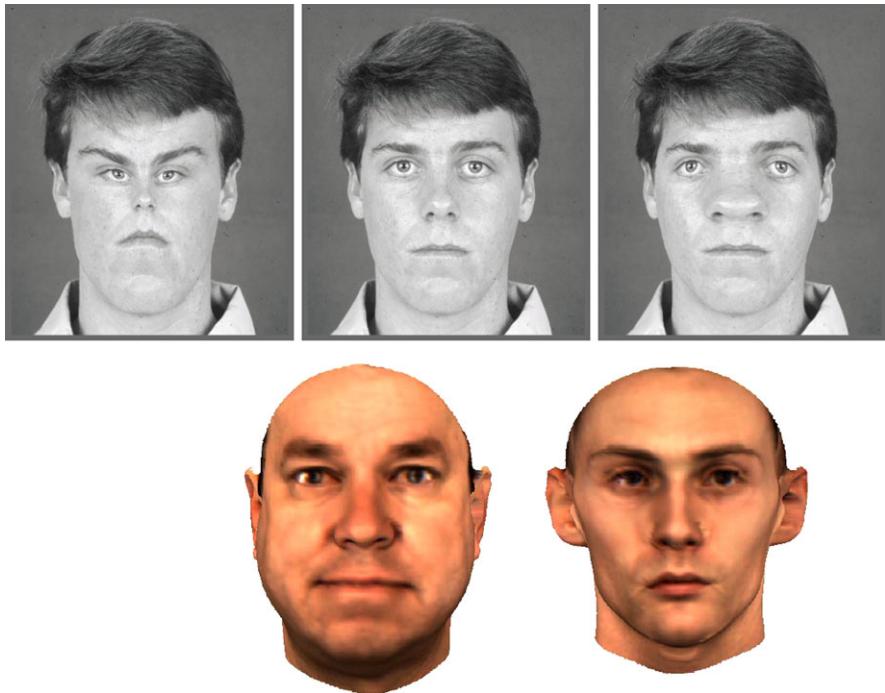
**Perceptual Adaptation** The third line of evidence for a norm-based coding of faces comes from data on perceptual after-effects and adaptation. Humans experience many simple visual after-effects, including effects involving color, orientation,

and motion. Let's take motion as an example. The Waterfall illusion is a motion direction after-effect first described in detail by Purkinje in 1820. It can be experienced by staring at a waterfall for about 60 seconds and then glancing away to a stationary object, such as a tree. Most people see a strong perception of the tree moving upward, counter to the direction of the waterfall motion. This perceptual illusion gives some basic insight into the nature of the neural code for motion. Long before neuroscientists were able to recode the activity in neurons, the illusion suggested that motion is coded in the human visual system with neural units that are selective to a particular axis of motion in a contrastive or opponent fashion. When these neurons are stimulated continuously with one motion direction, their neural firing rate slows or adapts to the continuous stimulus. This stable state sets a new norm against which increased velocity of motion, or reverse motion will be signally with greater potency. Thus, the stationary object contrasts strongly to the newly adjusted motion-direction norm. The code is efficient and adaptive to the local visual environment.

What does any of this have to do with faces? The short answer is that in the last decade, a host of analogous, albeit more complex, opponent-based after-effects have been demonstrated for faces. The first after-effect discovered operates on our perception of the configuration or shape of a face. It occurs after people stare at a face that has been distorted by unnaturally expanding or contracting the area around the eyes and nose (see Fig. 23.3, top row, left and right-most images). A subsequently presented normal face (Fig. 23.3, top row, middle image) then appears to be distorted in the opposite direction [35]. A related demonstration shows after-effects for perceiving the gender of a face. In this case, staring at male faces for a short period of time makes an androgynous face, defined usually as a 50% morph between a male and a female face, appear female, and vice versa [36].

Notably, there is also an opponent aftereffect between a face and its anti-face [19] (see Fig. 23.3, bottom row). Staring at the opposite of a face makes the perception of the face and its anti-caricatures easier. Again, this opposite is defined across many axes of variation. At its core, this is an expected property of a facial code that is centered at an average or prototype face and directly represents individual faces in terms of how they diverge from this average.

Combined the data on these after-effects suggests that human face representations are highly organized, and thoroughly interconnected. A face, and its individual characteristics and categorical dimensions (sex, race, age, competence, trustworthiness, etc.), continually reference locally similar and distantly contrastive faces in other parts of the face subspace. As noted at the outset, the advantage of this code is that it is highly efficient and adaptive in both the short term (i.e., perceptual adaptation) and long term senses. In the next section, we note the long term implications of the code and see an example of a weakness of over-optimizing representations.



**Fig. 23.3** Perceptual adaptation demonstrations that illustrate the norm-based nature of human face codes (*top row*). After staring at the unnaturally contracted face on the left for a short time (60 seconds), the normal face in the center appears unnaturally expanded (i.e., similar to the face on the right) and vice versa [35] (*bottom row*). A similar effect occurs for the face and anti-face pair. After staring at the anti-face, the identity of the original face is more efficiently perceived from anti-caricatures (i.e., less distinct versions of the original face) [19]

### 23.3.2 The “Other-Race Effect” for Faces

There is long-standing evidence from human face recognition studies that people recognize faces of their own-race more accurately than faces of other races [23]. Most explanations of the effect appeal to differences in the amount and quality of experience or contact we have with faces of our own race versus faces of other races (e.g., [8]). A simple version of this *contact hypothesis* explanation of the phenomenon, however, has been challenged by data that fail to show a consistent relationship between self-reported contact with people of other races and the magnitude of the other-race effect [20]. Self-reported contact is measured usually by asking someone how often they interact with people of another race. The magnitude of the other-race effect is measured as the difference in accuracy for faces of your own race and faces of another race. In sum, the relationship between these two variables is rather weak.

A broader look at the contact hypothesis, however, suggests that experience is indeed responsible for the other-race effect, with one important caveat. The expe-

rience must occur during childhood [17]. In particular, experience with other-race faces must occur as children are acquiring and tuning the feature sets they will use ultimately to optimize their representations of faces. This points to an important difference in learning during early infancy/childhood and learning later in life. Specifically, during early childhood, the brain is highly plastic in ways that do not continue into adulthood (e.g., physical growth and connectivity of neurons). Although clearly we continue to be able to “learn” later in life, the neural mechanisms by which this is accomplished are more stable and less malleable. In a more general context, this makes the process of learning to perceive and remember faces much like the process of learning language. It is well known that if children are exposed to a second language early in life (usually before the age of 4 or 5 years), they will acquire that language far more efficiently and with fewer negative indications (e.g., accents) than learning the language later in life [18]. The acquisition of human expertise for faces may proceed analogously.

The other-race effect suggests that human face codes develop from experience and that the types of features used may vary as a function of the face learning history of the individual. Although perceptual adaptation effects for faces are short-term and easily reversible, face learning in early life may have a more permanent effect on the quality of the representations we can create of other-race faces. We consider this question shortly in the context of the training methods used by current face recognition algorithms.

## 23.4 Human–Machine Comparisons

In the following, we consider human and machine-based face recognition systems in a common context. How accurate are machines relative to humans? Do machines and humans make the same kinds of errors? In what ways can the performance of humans inform algorithm developers about the challenges that they will need to overcome to enable machines to operate in natural environments? Can human strategies be helpful in mitigating obstacles to the application of face recognition technology in the real world?

Over the last few years, we have undertaken a number of direct comparisons between humans and automatic face recognition algorithms. As a source for these comparisons, we have made use of algorithms tested by U.S. Government-sponsored competitions of automated face recognition technology. These competitions have documented substantial gains in machine-based face recognition performance over the last decade (cf., [28, 29, 31]) (see also Chap. 21). Although these algorithms have been compared extensively to each other, there has been relatively little work aimed at benchmarking algorithm performance against humans. Because humans are currently the competition in most security applications, a full evaluation of human performance is required to know whether the use of an algorithm will improve security or put it at greater risk.

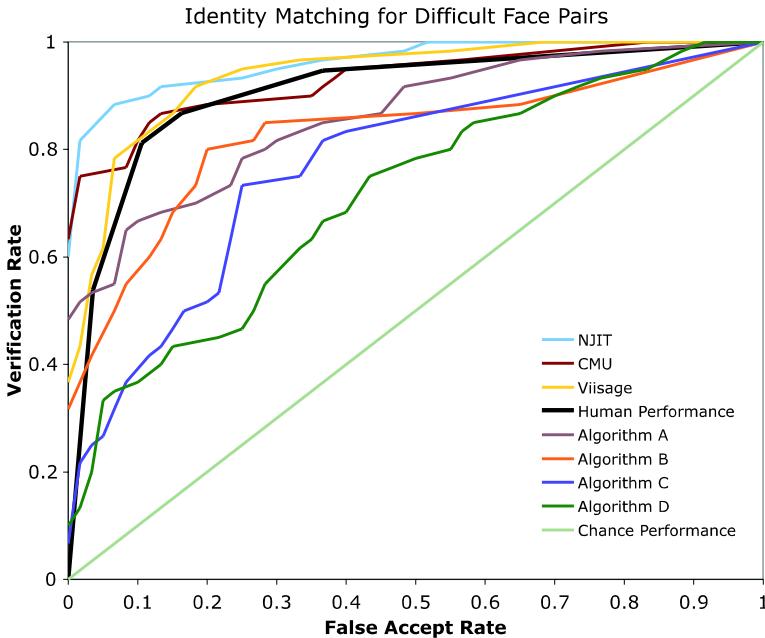
Before proceeding, it is perhaps worth giving a brief overview of the purpose and procedures used in the U.S. Government’s large scale tests, which have been organized by the National Institute of Standards and Technology (NIST) (for a more

detailed account, see Chap. 21). The stated purpose of these competitions is to spur the development of face recognition technologies to continually solve increasingly difficult recognition tasks. Face recognition algorithms developed in the late eighties and early nineties operated under highly controlled conditions, matching faces in frontal, neutral-expression images, taken close in time (i.e., without age or strong appearance differences) and under optimal illumination conditions. The first of the U.S. Government tests, the FERET evaluation (1991–1994), tested algorithms under these controlled conditions [28]. As the performance of algorithms improved on the controlled task, more challenging recognition tasks were attempted and tested in subsequent NIST evaluations [29, 31]. These conditions have included illumination variation between images, facial expression change, and differences in the appearance of a face that occur with time lapses of a year or more.

The advantage of the large scale evaluations is that they test many algorithms simultaneously using an identical protocol and stimulus set. Moreover, the tests make use of *very* large stimulus sets. Algorithm developers submit executable code to NIST, which is then used to test identity matching performance over a large set of *gallery* and *probe* images. The algorithms compute an  $n \times m$  similarity score matrix between all possible pairs of the  $n$  gallery and  $m$  probe images. The similarity score,  $s_{i,j}$  represents an algorithm's estimate of the likelihood that the  $i$ th gallery image and  $j$ th probe image are the same person. Performance is evaluated by NIST, using a standard receiver operating characteristic (ROC) curve that plots the verification (hit) rate against the false accept (false alarm) rate. This allows for a simple performance comparison among the algorithms. The data provided for algorithms in the NIST tests, operating under a variety of test scenarios, provide the source for the human comparisons we present.

## 23.5 Identification Accuracy Across Illumination

Matching the identity of people across two or more images is a common task for security personnel. It is also a common application for face recognition algorithms. Did this person apply for a driver's license under another name? Is this the person on a police watch list? For both humans and machines, the task is more difficult when the images to-be-compared vary in viewing conditions (pose, illumination) or in other extraneous ways (facial expression differences). One of the most studied of these viewing parameters is illumination. Many studies have demonstrated that recognizing people from images taken under different illumination conditions is challenging for both humans [5, 6, 16] and machines [14]. Although the problem has been studied both for humans and for machines, it is unclear how well face recognition algorithms perform *relative* to humans on this task. We carried out this comparison using data from the algorithms tested in the Face Recognition Grand Challenge (FRGC) [26]. In the FRGC, undertaken in 2005, the performance of face recognition algorithms was substantially worse when illumination varied between the target and query images than when both the target and query were taken under controlled illumination [31].



**Fig. 23.4** Human versus machine-based face recognition performance in the FRGC database [26] shows that the best algorithms are superior to humans. Humans performed better than four of the algorithms

To test humans using an analogous test, it was necessary to select a manageable sample of face pairs from the roughly 128 million pairs matched in the FRGC test. This was achieved by pre-screening face pairs by level of difficulty, using a PCA-based face recognition algorithm. The human test was implemented using a set of “difficult” face pairs. We measured performance by presenting pairs of faces side-by-side on a computer screen and asking human subjects to decide whether the two faces were the same person or different people. Humans also generated something like the similarity score computed by algorithms by rating their certainty that the images were of the same person (1: sure they are the same person; ... 5: sure they are different people). These confidence data can be used to create an ROC curve analogous to the ones that are used to describe algorithm performance [22].

The results of the human performance analysis appear in Fig. 23.4, along with the ROC curves for seven algorithms, tested with the identical set of difficult face pairs. The figure shows that three of the seven algorithms were more accurate than humans at this task. We have since repeated this kind of human-machine comparison using a new set of algorithms and two new stimulus sets from the FRVT 2006 [27]. The results again show human performance to be worse than the performance of the top algorithms.

In summary, we conclude that human and machine performance may be roughly comparable at a task of matching faces across changes in illumination, when these illumination changes are not too extreme.

## 23.6 Fusing Machine and Human Results

Do humans and machines make the same kinds of errors? We addressed this question using a *fusion* approach [25]. Fusion is a computational strategy for improving accuracy by combining judgments from multiple, but imperfect, sources. In general, fusion increases performance only when the information provided from the various sources is partially independent. Fusing algorithm and human judgments, therefore, can provide an indication of the extent to which human and machine recognition strategies diverge.

The fusion experiments we conducted again made use of the data collected in the FRGC experiment just described, in which algorithms matched identity across changes in illumination. The human part of the fusion was based on the similarity judgments generated by the human subjects in the previous study [26]. The fusion was approached in two steps. First, we combined the identity match estimates from the seven algorithms tested in the FRGC to determine whether differences among the algorithms could be leveraged to improve performance. In a second experiment, we fused the human-generated similarity estimates with the estimates from the seven algorithms, as a kind of eighth algorithm. To fuse, we used a statistical mapping procedure called partial least squares regression (PLS) that combined the face pair similarity estimates to predict the match status of the face pair (same or different person?). The model was tested with a cross validation procedure.

The results of the first fusion showed that the combination of the seven algorithms reduced the error rate in half, (0.06), over that of the best performing algorithm operating alone (0.12). When humans were included in the fusion, the error rate dropped to near perfect (0.003).

The two sets of fusion results suggests enough divergence in strategy among the algorithms and humans to successfully exploit these qualitative differences to improve the overall accuracy. Moreover, in combining human and machine judgments, there may be an optimal combination formula that can be discovered empirically using fusion techniques.

## 23.7 The “Other-Race Effect”

The human other-race effect for face recognition is likely due, at least in part, to the experience of individuals as they learn the complex task of recognizing faces. Many face recognition algorithms likewise employ training strategies that make use of one or more databases of images. These images can be used to select and optimize the feature sets that will be used by the algorithm to represent faces. The potential applications of automatic face recognition algorithms are now global. Moreover, in the FRVT 2006, for example, the algorithms submitted were developed in East Asia, Europe, and North America [31]. These regions vary immensely in their demographical compositions and presumably also in the demographical composition of the face databases easily available for training the algorithms. Given these circumstances, it is reasonable to ask whether state-of-the-art face recognition systems, like

humans, show an other-race effect. More specifically, does the geographical origin of an algorithm (i.e., where it was developed) affect its accuracy on different races of faces?

This question was addressed recently by comparing the performance of algorithms as a combined function of their geographic origin and the race of faces they recognized [30]. In this study, we used data from the FRVT 2006, which had participating algorithms from three East Asian countries (China, Japan and Korea) and three Western countries (France, Germany, and The United States). To compare performance on East Asian and Caucasian faces, we created an *East Asian algorithm* by fusing the similarity scores from the five algorithms submitted from East Asian countries and a *Western algorithm* by fusing the similarity scores from the eight algorithms submitted from Western countries. Identity match accuracy for the two algorithms was tested for all available East Asian ( $n = 205\,114$ ) and Caucasian ( $n = 3\,359\,404$ ) face pairs in the database. The findings showed a classic other-race effect, with the East Asian pairs matched more accurately by the East Asian algorithm and Caucasian face pairs matched more accurately by the Western algorithm.

In a second experiment, we conducted a head-to-head comparison between the algorithms and humans of East Asian and Caucasian descent. In this case, we tested with a more manageable number of face pairs, which were carefully matched by demographics other than face race. We found a standard other-race effect for the humans. We also found an other-race effect for the algorithms, but one with a different form than that found with all available pairs. The Western algorithm performed substantially worse on East Asian faces than on Caucasian faces. The East Asian also performed better on Caucasian faces, but by a far smaller margin. Phillips et al. (in press) suggested that one possible reason for the difference was the possibility that the East Asian algorithms, in preparing for the FRVT 2006, may have anticipated the demographic characteristics of the test. These algorithms may, therefore, have used both Caucasian and East Asian training faces. Notwithstanding, the relatively better performance on East Asian faces by the East Asian algorithm is consistent with an other-race effect.

Because the NIST tests work directly with executable code, without access to the algorithms and training procedures, it is impossible to establish an unambiguous cause for these results. It is nonetheless important to know that these biases may exist in current state-of-the-art systems and to anticipate, test for, and mitigate these types of problems before putting an algorithm into the field. Many application venues (e.g., major airports across the globe) will present unique challenges for stable, accurate, and fair recognition across ethnic and racial groups.

In summary, just as the human strategy of optimizing a representation for a particular demographic context has advantages and costs, so too must algorithms consider both the advantages and limitations of optimization strategies.

## 23.8 Conclusions

In this chapter, we have noted that an important characteristic of human face representations that these representations are centered on an average face. A unique indi-

vidual face is defined, therefore, by its differences from the average. The face subspace that results is richly interconnected and can be manipulated both by short term (by perceptual adaptation) and longer term (developmental) experience. A strategy for performing well in difficult face recognition tasks for humans may involve active and internal gain adjustments that can magnify the salience of some features while minimizing others.

As algorithms approach and begin to overtake humans on simple face recognition tasks, it is clear that they still operate in a monolithic way. The neural systems for face recognition are highly distributed and involve a number of distinct brain regions [15]. These regions are in part divided according to the type of task they carry out with faces (i.e., identification, the perception of non-rigid and rigid facial motions and expressions (see Chap. 22)). This flexibility of the system to multi-task may be the key to the expertise that humans show with the faces of people they know well [24]. This kind of flexible and robust recognition should be the aim of the next generation of algorithms. To achieve this goal, algorithms will need to understand not only the static structure of a face, but the ways in which faces move and change with age/appearance. They will also need to move beyond the sole concern of operating robustly over viewing conditions, and onto strategies that will allow them operate in real environments that will vary in demographic composition. If they succeed in all of these challenges, they will surpass even humans in even the best case scenario.

**Acknowledgements** This work was supported by funding to A.J. O'Toole from the Technical Support Working Group of the Department of Defense, USA.

## References

1. Adolphs, R., Tranel, D., Damsio, H., Damasio, A.R.: Impaired recognition of emotion in facial expressions following bilateral damage to the human amygdala. *Nature* **372**, 669–672 (1994)
2. Antonakis, J., Dalgas, O.: Predicting elections: Child's play! *Science* **323**(5918), 1183–1327 (2007)
3. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3d faces. In: *Proceedings, SIGGRAPH'99*, pp. 187–194 (1999)
4. Bower, G.H., Karlin, M.B.: Depth of processing pictures of faces and recognition memory. *J. Exp. Psychol.* **103**(4), 267–278 (1974)
5. Braje, W.: Illumination encoding in face recognition: effect of position shift. *J. Vis.* **3**, 161–170 (2003)
6. Braje, W., Kersten, D., Tarr, M.J., Troje, N.F.: Illumination effects in face recognition. *Psychobiology* **26**, 371–380 (1999)
7. Bruce, V., Young, A.W.: Understanding face recognition. *Br. J. Psychol.* **77**(3), 305–327 (1986)
8. Bryatt, G., Rhodes, G.: Recognition of own-race and other-race caricatures: Implications for models of face recognition. *Vis. Res.* **38**, 2455–2468 (1998)
9. Burton, A.M., Bruce, V., Hancock, P.J.B.: From pixels to people: A model of familiar face recognition. *Trends Cogn. Sci.* **23**, 1–31 (1999)
10. Burton, A.M., Wilson, S., Cowan, M., Bruce, V.: Face recognition in poor-quality video. *Psychol. Sci.* **10**, 243–248 (1999)

11. Calder, A., Young, A.: Understanding the recognition of facial identity and facial expression. *Nat. Rev., Neurosci.* **6**, 641–651 (2005)
12. Doddington, G., Ligget, W., Martin, A., Przybocki, M., Reynolds, D.: Sheeps, goats, lambs, and wolves: A statistical analysis of speaker performance in the NIST 1998 recognition evaluation. In: *Proceedings ICSLP '98*, 1998
13. Ekman, P., Friesen, W.V.: *Pictures of Facial Affect*. Consulting Psychologists Press, Palo Alto (1976)
14. Gross, R., Baker, S., Matthews, I., Kanade, T.: Face recognition across pose and illumination. In: Li, S.Z., Jain, A.K. (eds.) *Handbook of Face Recognition*, pp. 193–216. Springer, Berlin (2005)
15. Haxby, J.V., Hoffman, E.A., Gobbini, M.I.: The distributed human neural system for face perception. *Trends Cogn. Sci.* **20**(6), 223–233 (2000)
16. Hill, H., Bruce, V.: Effects of lighting on the perception of facial surface. *J. Exp. Psychol.* **22**, 986–1004 (1996)
17. Kelly, D.J., Quinn, P.C., Slater, A.M., Lee, K., Ge, L., Pascalis, O.: The other-race effect develops during infancy: Evidence of perceptual narrowing. *Psychol. Sci.* **18**, 1084–1089 (2007)
18. Kuhl, P.K., Williams, K.H., Lacerdo, F.: Linguistic experience alters phonetic perception in infants by 6 months of age. *Science* **225**, 606–608 (1992)
19. Leopold, D.A., O'Toole, A.J., Vetter, T., Blanz, V.: Prototype-referenced shape encoding revealed by high-level aftereffects. *Nat. Neurosci.* **4**, 89–94 (2001)
20. Levin, D.: Classifying faces by race: The structure of face categories. *J. Exp. Psychol. Learn. Mem. Cogn.* **22**, 1364–1382 (1996)
21. Light, L., Kayra-Stuart, F., Hollander, S.: Recognition memory for typical and unusual faces. *J. Exp. Psychol. Hum. Learn. Mem.* **5**, 212–228 (1979)
22. Macmillan, N.A., Creelman, C.D.: *Detection Theory: A User's Guide*. Cambridge University Press, Cambridge (1991)
23. Malpass, R.S., Kravitz, J.: Recognition for faces of own and other race faces. *J. Pers. Soc. Psychol.* **13**, 330–334 (1969)
24. O'Toole, A.J., Roark, D., Abdi, H.: Recognition of moving faces: A psychological and neural perspective. *Trends Cogn. Sci.* **6**, 261–266 (2002)
25. O'Toole, A.J., Abdi, H., Jiang, F., Phillips, P.J.: Fusing face recognition algorithms and humans. *IEEE Trans. Syst. Man Cybern.* **37**(5), 1149–1155 (2007)
26. O'Toole, A.J., Phillips, P.J., Jiang, F., Ayyad, J., Pénard, N., Abdi, H.: Face recognition algorithms surpass humans matching faces across changes in illumination. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**, 1642–1646 (2007)
27. O'Toole, A.J., Phillips, P.J., Narvekar, A.: Humans versus algorithms: Comparisons from the FRVT 2006. In: *Eighth International Conference on Automatic Face and Gesture Recognition*, 2008
28. Phillips, P.J., Moon, H., Rizvi, S., Rauss, P.: The FERET evaluation methodology for face-recognition algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**, 1090–1104 (2000)
29. Phillips, P.J., Grother, P.J., Micheals, R.J., Blackburn, D.M., Tabassi, E., Bone, J.M.: Face recognition vendor test 2002: Evaluation report. Technical Report NISTIR 6965, National Institute of Standards and Technology (2003). <http://www.frvt.org>
30. Phillips, P.J., Narvekar, A., Jiang, F., O'Toole, A.J.: An other-race effect for face recognition algorithms. *ACM Trans. Appl. Percept.* (2010)
31. Phillips, P.J., Scruggs, W.T., O'Toole, A.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: Frvt 2006 and ice 2006 large-scale experimental results. *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(5), 831–846 (2010)
32. Rhodes, G.: *Superportraits: Caricatures and Recognition*. The Psychology Press, Hove (1996)
33. Todorov, A., Mandisodza, A.N., Gren, A., Hall, C.C.: Inferences of competence from faces predict election outcomes. *Science* **308**, 1623–1626 (2005)
34. Valentine, T.: A unified account of the effects of distinctiveness, inversion, and race in face recognition. *Q. J. Exp. Psychol., A Hum. Exp. Psychol.* **43**, 161–204 (1991)

35. Webster, M.A., MacLin, O.H.: Figural after-effects in the perception of faces. *Psychon. Bull. Rev.* **6**, 647–653 (1999)
36. Webster, M.A., Kaping, D., Mizokami, Y., Duhamel, P.: Adaptation to natural facial categories. *Nature* **428**, 557–561 (2004)
37. Young, A.W., Newcombe, F., de Haan, E.H.F., Small, M., Hay, D.C.: Face perception after brain injury: Selective impairments affecting identity and expression. *Brain* **116**(4), 941–959 (1993)

## **Part IV**

# **Face Recognition Applications**



# Chapter 24

## Face Recognition Applications

Thomas Huang, Ziyou Xiong, and Zhenqiu Zhang

### 24.1 Introduction

One of the reasons face recognition has attracted so much research attention and sustained development over the past 30 years is its great potential in numerous government and commercial applications. In 1995, Chellappa et al. [5] listed a small number of applications of face recognition technology and described their advantages and disadvantages. However, they did not analyze any system deployed in real applications. Even the more recent review [38], where the set of potential applications has been grouped into five categories, did not conduct such an analysis. In 1997, at least 25 face recognition systems from 13 companies were available [3]. Since then, the numbers of face recognition systems and commercial enterprises have greatly increased owing to the emergence of many new application areas, further improvement of the face recognition technologies, and increased affordability of the systems. We have listed 10 of the representative commercial face recognition companies, their techniques for face detection, the face features they extract, and face similarity comparison methods in Table 24.1. These 10 companies are also the participants of the face recognition vendor test (FRVT 2002) [29] carried out independently by the U.S. government to evaluate state-of-the-art face recognition technology. Although some of these techniques are not publicly available for proprietary reasons, one can conclude that many others have been incorporated into commercial systems.

---

T. Huang (✉) · Z. Zhang  
University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA  
e-mail: [huang@ifp.uiuc.edu](mailto:huang@ifp.uiuc.edu)

Z. Zhang  
e-mail: [zzhang6@uiuc.edu](mailto:zzhang6@uiuc.edu)

Z. Xiong  
United Technologies Research Center, East Hartford, CT 06108, USA  
e-mail: [xiongz@utrc.utc.com](mailto:xiongz@utrc.utc.com)

**Table 24.1** Comparison of face recognition algorithms from 10 commercial systems in FRVT 2002. N/A: not available

| Company      | Method for face detection          | Face feature extraction method    | Matching method        |
|--------------|------------------------------------|-----------------------------------|------------------------|
| Acsys        | N/A                                | Biometric templates               | Template matching      |
| Cognitec     | N/A                                | Local discriminant analysis (LDA) | N/A                    |
| C-VIS        | Fuzzy face model and neural net    | N/A                               | Elastic net matching   |
| Dream Mirh   | N/A                                | N/A                               | N/A                    |
| Eyematic     | General face model                 | Gabor wavelet                     | Elastic graph matching |
| IConquest    | Fractal image comparison algorithm |                                   |                        |
| Identix      | N/A                                | Local feature analysis (LFA)      | Neural network         |
| Imagis       | Deformable face model              | Spectral analysis                 | N/A                    |
| Viisage      | N/A                                | Eigenface                         | Euclidean distance     |
| VisionSphere | N/A                                | Holistic feature code             | N/A                    |

As one of the most nonintrusive biometrics, face recognition technology is becoming ever closer to people's daily lives. Evidence of this is that in 2000 the International Civil Aviation Organization endorsed facial recognition as the most suitable biometrics for air travel [12]. To our knowledge, no review papers are available on the newly enlarged application scenarios since then [3, 5, 38]. We hope this chapter will be an extension of the previous studies. We review many face recognition applications that have already used face recognition technologies. This set of applications is a much larger super-set of that reviewed in [3]. We also review some other new scenarios that will potentially utilize face recognition technologies in the near future.

These scenarios are grouped into 10 categories, as shown in Table 24.2. Although we try to cover as many categories as possible, these 10 categories are neither exclusive nor exhaustive. For each category, some of the exemplar applications are also listed. The last category, called "Others," includes future applications and some current applications that we have not looked into. These 10 categories are reviewed from Sects. 24.3 to 24.11. In Sect. 24.12, some of the limitations of the face recognition technologies are reviewed. Concluding remarks are made in Sect. 24.13.

## 24.2 Face Identification

Face recognition systems identify people by their face images [6]. In contrast to traditional identification systems, face recognition systems establish the presence of an authorized person rather than just checking whether a valid identification (ID) or key

**Table 24.2** Application categories

| Category                         | Exemplar application scenarios                                                                                                                                                                                                                                                         |
|----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Face ID                          | Driver licenses, entitlement programs, immigration, national ID, passports, voter registration, welfare registration                                                                                                                                                                   |
| Access control                   | Border-crossing control, facility access, vehicle access, smart kiosk and ATM, computer access, computer program access, computer network access, online program access, online transactions access, long distance learning access, online examinations access, online database access |
| Security                         | Terrorist alert, secure flight boarding systems, stadium audience scanning, computer security, computer application security, database security, file encryption, intranet security, Internet security, medical records, secure trading terminals                                      |
| Surveillance                     | Advanced video surveillance, nuclear plant surveillance, park surveillance, neighborhood watch, power grid surveillance, CCTV control, portal control                                                                                                                                  |
| Smart cards                      | Stored value security, user authentication                                                                                                                                                                                                                                             |
| Law enforcement                  | Crime stopping and suspect alert, shoplifter recognition, suspect tracking and investigation, suspect background check, identifying cheats and casino undesirables, post-event analysis, welfare fraud, criminal face retrieval and recognition                                        |
| Face databases                   | Face indexing and retrieval, automatic face labeling, face classification                                                                                                                                                                                                              |
| Multimedia management            | Face-based search, face-based video segmentation and summarization, event detection                                                                                                                                                                                                    |
| Human computer interaction (HCI) | Interactive gaming, proactive computing                                                                                                                                                                                                                                                |
| Others                           | Antique photo verification, very low bit-rate image & video transmission, etc.                                                                                                                                                                                                         |

is being used or whether the user knows the secret personal identification numbers (PINs) or passwords. The security advantages of using biometrics to check identification are as follows. It eliminates the misuse of lost or stolen cards, and in certain applications it allows PINs to be replaced with biometric characteristics, which makes financial and computer access applications more convenient and secure. In addition, in situations where access control to buildings or rooms is automated, operators may also benefit from improved efficiency. Face recognition systems are already in use today, especially in small database applications such as those noted in Sect. 24.3. In the future, however, the targeted face ID applications will be large-scale applications such as e-commerce, student ID, digital driver licenses, or even national ID.

Large-scale applications still face a number of challenges. Some of the trial applications are listed below.

1. In 2000, FaceIt technology was used for the first time to eliminate duplicates in a nationwide voter registration system because there are cases where the same person was assigned more than one identification number [12]. The face recog-

nition system directly compares the face images of the voters and does not use ID numbers to differentiate one from the others. When the top two matched faces are extremely similar to the query face image, manual inspection is required to make sure they are indeed different persons so as to eliminate duplicates.

2. Viisage's faceFinder system [33] has been supplied to numerous state corrections authorities and driver license bureaus. This face recognition technology has also been used by the U.S. Department of State for the Diversity Visa Program, which selects approximately 50 000 individuals to be considered for a permanent U.S. visa from millions of applications submitted each year. Each application includes a facial image. The system compares the image of every applicant against the database to reduce the potential of the same face obtaining multiple entries in the lottery program. Once enrolled in the Viisage system, images can also be used during the diversity visa application process to help identify known individuals who pose specific security threats to the nation.

### 24.3 Access Control

In many of the access control applications, such as office access or computer login, the size of the group of people that need to be recognized is relatively small. The face pictures are also captured under constrained conditions, such as frontal faces and indoor illumination. Face recognition-based systems in these applications can achieve high accuracy without much cooperation from the users; for example, there is no need to touch an object by fingers or palms, no need to present an eye to a detector. When combined with other forms of authentication schemes such as fingerprint or iris recognition, face recognition systems can achieve high accuracy. Thus, the user satisfaction level is high. This area of application has attracted many commercial face recognition systems. The following are several examples.

- In 2000, IBM began to ship FaceIt [11] enabled screen saver with Ultraport camera for A, T, and X series Thinkpad notebook computers. Face recognition technology is used to monitor continuously who is in front of a computer terminal. It allows the user to leave the terminal without closing files and logging off. When the user leaves for a predetermined time, a screen saver covers the work and disables the keyboard and mouse. When the user returns and is recognized, the screen saver clears and the previous session appears as it was left. Any other user who tries to log in without authorization is denied.
- The University of Missouri-Rolla campus has chosen a face recognition system by Omron [28] to secure a nuclear reactor, which is a 200-kilowatt research facility that uses low-enriched uranium to train nuclear engineers. Visitors must pass through a staff-monitored lobby, a second door that is accessed with a key, and a third door that is secured with a keypad before getting to the face scanner, which regulates access to the reactor core.
- Another commercial access control system is called FaceGate [10]. Entering a building using FaceGate simply requires one to enter his entry code or a card and

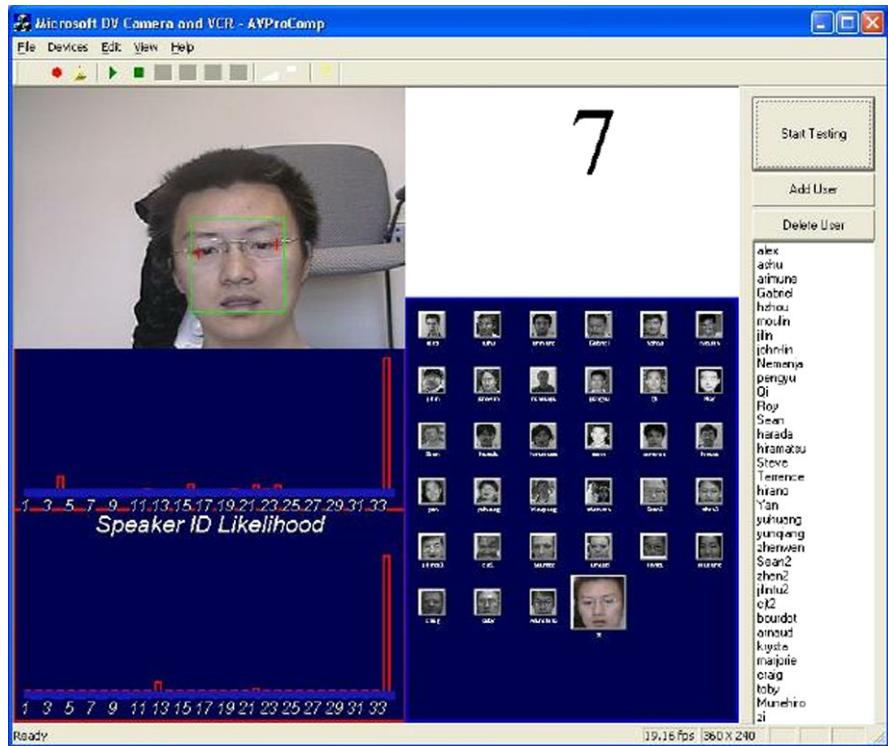
**Fig. 24.1** FaceGate access control system



face a camera on the door entry system. Figure 24.1 is a snapshot of the system. By applying a mathematical model to an image of a face, FaceGate generates a unique biometric “key.” Whenever one wishes to access a building, FaceGate verifies the person’s entry code or card, then compares his face with its stored “key.” It registers him as being authorized and allows him to enter the building. Access is denied to anyone whose face does not match.

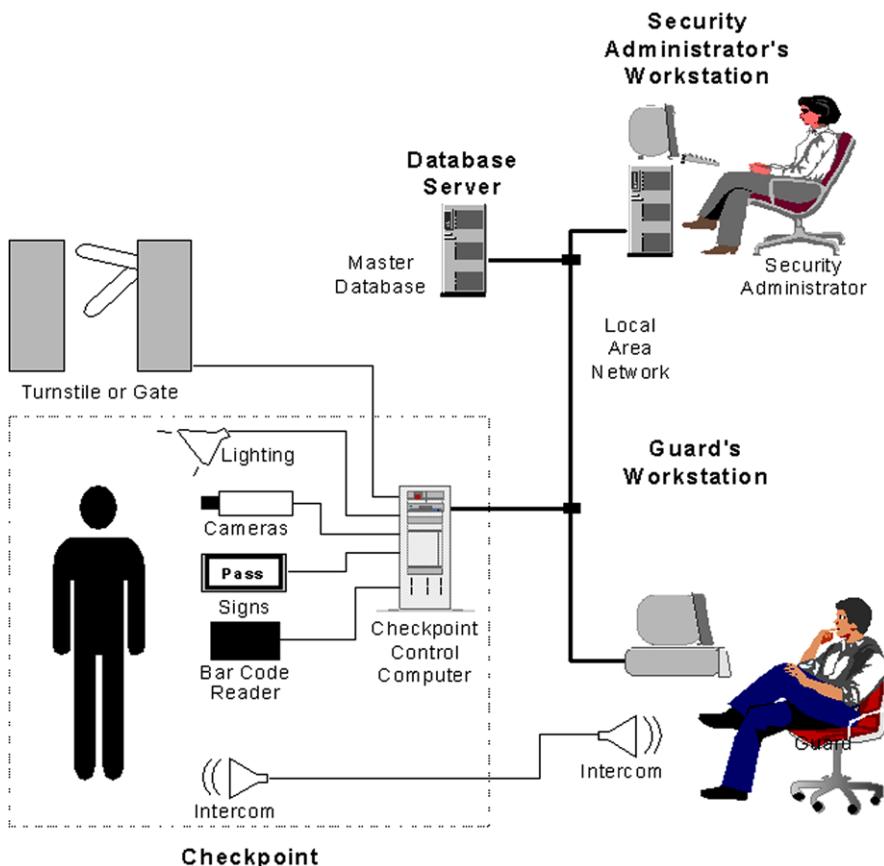
- The FaceKey standard biometric access control system combines face recognition and fingerprint recognition to provide a high level of security [13]. There is no need for cards, keys, passwords, or keypads. The combination of the two biometrics makes it possible to have security with a low error rate. The system can operate as a stand-alone, one-door system, or it can be networked to interconnect multiple doors at multiple sites, domestically or internationally.
- “FaceVACS-Entry” [6] adds facial recognition to conventional access control systems. At the access point, the face of each person is captured by a video camera, and the facial features are extracted and compared with the stored features. Only if they match is access permitted. For high security areas, a combination with card terminals is possible, so each card can be used only by its owner. Flexible communication interfaces enable easy integration into existing access control or time and attendance systems. Terminals with FaceVACS-Entry can be networked together, so after central enrollment the face data are distributed automatically to all the terminals. In addition, visual control by security personnel can be supported. All facial images collected at the terminals can be stored in a log for later visual inspection via a standard browser.

In addition to commercial access control systems, many systems are being developed in university research laboratories that are exploring new face recognition algorithms. We give two examples.



**Fig. 24.2** A computer access control system using both face and speaker recognition

- At the University of Illinois [37], face recognition and speaker identification systems have been integrated to produce high recognition accuracy for a computer login system. Figure 24.2 shows the system interface where the upper-left corner displays the real-time video captured by a digital camcorder. The upper-center displays text or digits for the user to read aloud for speaker identification. At the upper-right corner are three buttons titled “Start Testing,” “Add User,” “Delete User,” indicating three functionalities. Two bar charts in the lower-left corner display the face recognition and speaker identification likelihoods, respectively, for each user. In the lower-center, icon images of users that are currently in the database are shown in black and white and the recognized person has his image enlarged and shown in color. The lower-right of the screen displays all the names of the users currently in the database.
- The second system [2] uses a multilayer perceptron for access control based on face recognition. The robustness of neural network (NN) classifiers is studied with respect to the false acceptance and false rejection errors. A new thresholding approach for rejection of unauthorized persons is proposed. Ensembles of NN with different architectures were also studied.



**Fig. 24.3** An exemplar airport security system

## 24.4 Security

Today more than ever, security is a primary concern at airports and for airline personnel and passengers. Airport security systems that use face recognition technology have been implemented at many airports around the globe. Figure 24.3 diagrams a typical airport security system that employs face recognition technology. Although it is possible to control lighting conditions and face orientation in some security applications, (e.g., using a single pedestrian lane with controlled lighting), one of the greatest challenges for face recognition in public places is the large number of faces that need to be examined, resulting in a high false alarm rate. Overall, the performance of most of the recognition systems has not met the very low false rejects goal with low false alarm requirements. The user satisfaction level for this area of application is low.

Some of the exemplar systems at airports, stadiums, and for computer security are listed below.

1. During the 2008 Beijing Olympics games, a face recognition system developed by the Institute of Automation of the Chinese Academy of Sciences (CAS) was introduced into the entrance security checks for the Olympic opening and closing ceremony. According to CAS, it was the first time that such technology was adopted as security measures in the Olympic history.
2. In October, 2001, Fresno Yosemite International (FYI) airport in California deployed Viisage's face recognition technology for airport security purposes. The system is designed to alert FYI's airport public safety officers whenever an individual matching the appearance of a known terrorist suspect enters the airport's security checkpoint. Anyone recognized by the system would undergo further investigative processes by public safety officers.
3. At Sydney airport, Australian authorities are trying out a computerized face-recognition system called SmartFace by Visionics with cameras that have wide-angle lenses. The cameras sweep across the faces of all the arriving passengers and send the images to a computer, which matches the faces with pictures of wanted people stored in its memory. If the computer matches the face with that of a known person, the operator of the surveillance system receives a silent alarm and alerts the officers that the person should be questioned. The technology is also used at Iceland's Keflavik airport to seek out known terrorists.
4. At Oakland airport (San Jose, California), a face recognition system by Imagis Technologies of Vancouver, British Columbia, Canada is used in interrogation rooms behind the scenes to match suspects brought in for questioning to a database of wanted criminals' pictures.
5. Malaysia's 16 airports use a FaceIt-based security system to enhance passenger and baggage security. A lipstick-size camera at the baggage check-in desk captures a live video of the passengers and embeds the data on a smart-card chip. The chip is embedded on the boarding pass and on the luggage claim checks. The system ensures that only passengers who have checked their luggage can enter the departure lounge and board the aircraft, and that only the luggage from boarding passengers is loaded into the cargo area. During the boarding process, the system automatically checks a real-time image of a passenger's face against that on the boarding pass smart chip. No luggage is loaded unless there is a match.
6. Viisage's faceFINDER equipment and software were used to scan the stadium audience at the Super Bowl 2001 at the Raymond James Stadium in Tampa, Florida in search of criminals. Everyone entering the stadium was scanned by video cameras set up at the entrance turnstiles. These cameras were tied to a temporary law-enforcement command center that digitized their faces and compared them against photographic lists of known malefactors. The system is also used by South Wales Police in Australia to spot soccer hooligans who are banned from attending matches.
7. Computer security has also seen the application of face recognition technology. To prevent someone else from modifying files or transacting with others when the authorized individual leaves the computer terminal momentarily, users are continuously authenticated, ensuring that the individual in front of the computer screen or at a kiosk is the same authorized person who logged in.

## 24.5 Surveillance

Like security applications in public places, surveillance by face recognition systems has a low user satisfaction level, if not lower. Unconstrained lighting conditions, face orientations and other factors all make the deployment of face recognition systems for large scale surveillance a challenging task. The following are some examples of face-based surveillance.

1. In 1998 Visionics FaceIt technology was deployed for the first time to enhance town center surveillance in Newham Borough of London, which has 300 cameras linked to the closed circuit TV (CCTV) control room. The city council claims that the technology has helped to achieve a 34% drop in crime since its installation. Similar systems are in place in Birmingham, England. In 1999 Visionics was awarded a contract from National Institute of Justice to develop smart CCTV technology [12].
2. Tampa, Florida police use video cameras and face recognition software to scan the streets in search of sex offenders. FaceIt provided by Visionics quickly compares the face of a target against a database of people wanted on active warrants from the Hillsborough Sheriff's Office and a list of sex offenders maintained by the Florida Department of Law Enforcement. When the FaceIt system comes up with a close match, cops using it in a remote location can contact others on the street via radio and instruct them to do further checking.
3. Virginia Beach, Virginia is the second U.S. city to install the FaceIt system on its public streets to scan pedestrian's faces to compare with 2500 images of people with outstanding warrants, missing persons, and runaways.
4. In New York City, the National Park Service deployed a face recognition surveillance system for the security of the Statue of Liberty. The system, including two cameras mounted on tripods, at the ferry dock where visitors leave Manhattan for Liberty Island, takes pictures of visitors and compares them with a database of terror suspects. The cameras are focused on the line of tourists waiting to board the ferry, immediately before they pass through a bank of metal detectors.

## 24.6 Smart Cards

The Smart Card has an embedded microprocessor or memory chip that provides the processing power to serve many applications. Memory cards simply store data. A microprocessor card, on the other hand, can add, delete, and manipulate information in its memory on the card. A microprocessor card also has built-in security features. Contact-less smart cards contain a small antenna so the card reader detects the card from a distance. The Smart Card's portability and ability to be updated make it a technology well suited for securely connecting the virtual and physical worlds.

The application of face recognition technology in smart cards, in essence, is a combination of the two. This can be seen from the following two examples. Smart

cards store the mathematical characteristics of the faces during the enrollment stage. The characteristics are read out during the verification stage for comparison with the live capture of the person's face. If granted, the person can have his stored facial characteristics updated in the card's memory.

1. Maximus [26] coupled face recognition system with fingerprint technology to construct a smart card designed to help airline passengers quickly clear security. To get a smart card, one needs to submit to a background check and register his or her facial and fingerprint characteristics. Biometric readers, presumably set up in specially designated "fast lanes," then verify his or her identification.
2. The ZN-Face system [19], which combines face recognition and smart card technology, is used for protecting secure areas at Berlin airports. Potential threats posed by criminals who often succeed in entering high security areas by means of a suitable disguise (e.g., pilot uniforms) are ruled out effectively. The individual's face characteristics are stored on a smart card; ZN-Face compares and verifies the card information with the face readings at each access station.

Smart cards are used mainly in a face verification scenario. The accuracy of the similarity calculation between the face characteristics stored in the cards and the live-captured face depends on the elapsed time between the two images. With a timely update of the face characteristics, this elapsed time can be kept short. High user satisfaction level can be achieved for a small database of faces.

## 24.7 Law Enforcement

With a face recognition and retrieval program, investigators can find a suspect quickly. Face recognition technology empowers the law enforcement agencies with the ability to search and identify suspects quickly even with incomplete information of their identity, sometimes even with a sketch from a witness's recollection. Owing to the difficulty of obtaining good-quality face images of the criminals, the system performance is rather low. However, automatic face recognition is playing increasingly important role in assisting the police departments. Some examples in this category of applications are as follows:

1. A law enforcement system by Imegis provides the Huntington Beach, California's police officers and detectives with current arrest information and photographs, readily available by using laptops, Internet protocols, and secure wireless delivery and communication [18]. The Imegis system includes biometric facial recognition, and image and database management, giving officers invaluable investigative tools in their law enforcement and surveillance work. With this face recognition and retrieval program, investigators no longer have to spend hundreds of hours trying to identify a suspect. Now detectives can take a suspect composite and systematically search any digital database of booking images to identify possible suspects. Similarly, a suspect's image caught on a bank or convenience store surveillance video can be matched against a digital photo database

for possible identification. With a face ID interface on a county booking system, officers are able to utilize this face-recognition technology at the time of booking to immediately identify a criminal with multiple identities or outstanding warrants. Detectives can also use face ID to search for suspects in a database of registered sex offenders. It allows witnesses to identify specific features of various faces as a means to query a large database of images. This function enhances the crime resolution process by eliminating the need for witnesses to search large mug-shot books one image at a time.

2. When deployed in a casino environment, an intelligent surveillance and patron management system supported by Imagis's face recognition technology [17] allows casino operators to identify and exclude certain individuals from specific properties. Using a database of North American undesirable patrons or self-barred gamblers, casinos receive a highly effective security solution that can rapidly identify persons entering or playing in casinos. It not only can conduct face recognition searches from images captured through existing surveillance cameras against an internal database, a casino can also widen the identification search to the national database.

## 24.8 Face Databases

During the early 1990s, because of the emergence of large image databases, difficulties faced by the text-based image retrieval became more and more acute [32]. Content-based image retrieval tries to solve the difficulties faced by text-based image retrieval. Instead of being manually annotated by text-based keywords, images would be indexed by their own visual content, such as color and texture. Feature vector is the basis of content-based image retrieval, which captures image properties such as color and texture. However, these general features have their own limitations. Recently, researchers have tried to combine it with other image analysis technologies, such as face detection and recognition, to improve the retrieval accuracy. For example, web-based face recognition has been used in social-computer sites such as Facebook, Google's Picasa web album, and Microsoft's Windows Live Gallery. These applications use facial scanning and recognition algorithms to scan through a person's online photos and those public photos belonging to his/her friends in order to identify and suggest tags for the untagged people within them [9]. Although face recognition techniques have been mainly used to retrieve and index faces in face-only databases (e.g., searching mug-shot databases of criminals), recently these techniques have also been used for other databases containing both faces and nonfaces (e.g., personal photo albums).

The performance of these retrieval systems is still low because the size of face database is normally large and the face pictures are captured under unconstrained conditions.

### 24.8.1 Using Faces to Assist Content-Based Image Retrieval

A personal digital photo album has many images that have either human faces or no human faces. Deciding whether an image contains a face can be a preprocessing step to limit the range of search space for a given image query. FotoFile [20] is one of the systems that tries to support this functionality to make the management of personal photo albums easier. This system also blends human and automatic annotation methods. Fotofile offers a number of techniques that make it easier for a consumer to annotate the content manually and to fit the annotation task more naturally into the flow of activities that consumers find enjoyable. The use of automated feature extraction tools enables FotoFile to generate some of the annotation that would otherwise have to be manually entered. It also provides novel capabilities for content creation and organization.

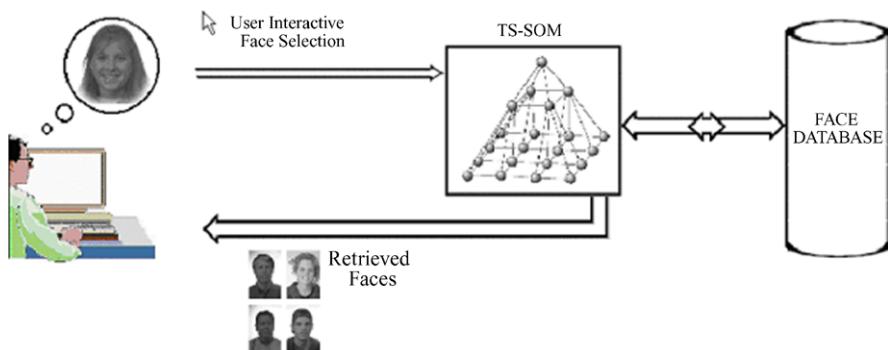
When presented with photos that contain faces of new people, the face recognition system attempts to match the identity of the face. The user either corrects or confirms the choice; the system then can match faces to their correct identities more accurately in subsequent photos. Once a face is matched to a name, that name is assigned as an annotation to all subsequently presented photos that contain faces that match the original. To handle the false positives and false negatives of the face recognition system, a user must confirm face matches before the annotations associated with these faces are validated.

### 24.8.2 Using Content-Based Image Retrieval Techniques to Search Faces

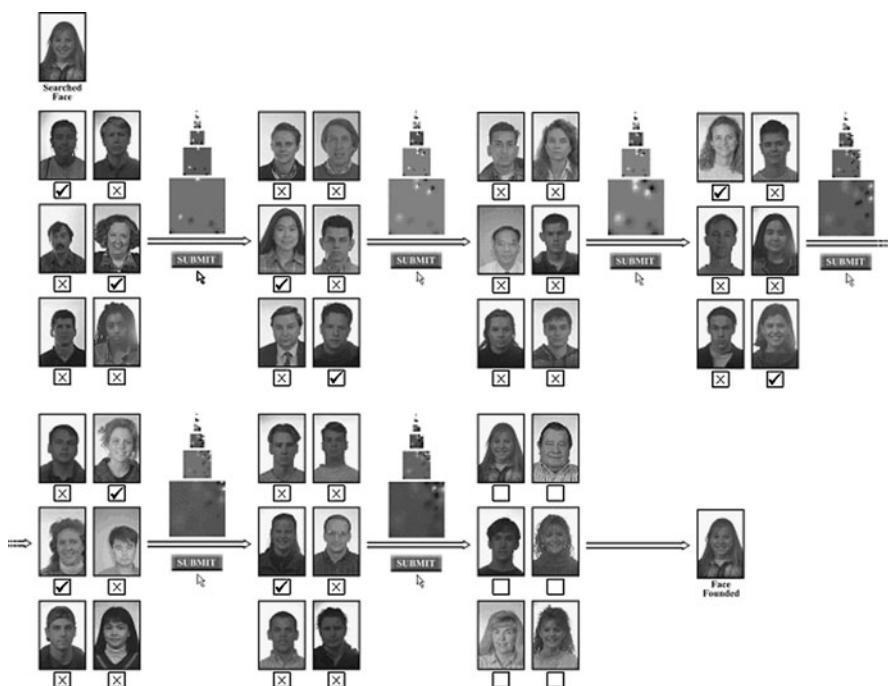
The content-based retrieval of faces has multiple applications that exploit existing face databases. One of the most important tasks is the problem of searching a face without its explicit image, only its remembrance. Navarrete and del Solar [27] used the so-called relevance feedback approach. Under this approach, previous human computer interactions are employed to refine subsequent queries, which iteratively approximate the wishes of the user. This idea is implemented using self-organizing maps. In particular, their system uses a tree-structured self-organizing map (TS-SOM) for auto-organizing the face images in the database. Similar face images are located in neighboring positions of the TS-SOM.

To know the location of the requested face in the map, the user is asked to select face images he considers to be similar to the requested one from a given set of face images. The system then shows the new images, which have neighboring positions, with respect to the ones selected by the user. The user and the retrieval system iterate until the interaction process converges (i.e., the requested face image is found). This retrieval system in shown in Fig. 24.4, and a real example of the interactive face-retrieval process is shown in Fig. 24.5.

Eickeler and Birlinghoven [8] explored the face database retrieval capabilities of a face recognition system based on the hidden Markov model. This method is able



**Fig. 24.4** Interface of the SOM system



**Fig. 24.5** Face retrieval using the SOM system

to work on a large database. Experiments carried out on a database of 25 000 face images show that this method is suitable for retrieval on a large face database. Martinez [25] presented a different approach to indexing face images. This approach is based on identifying frontal faces and allows reasonable variability in facial expressions, illumination conditions, and occlusions caused by eyewear or items of clothing such as scarves. The face recognition system of this approach is also based on the hidden Markov model [8].

### 24.8.3 Photo Tagging

Face recognition has been used to tag faces found in photographs. Apple's iPhoto and Google's Picasa software allows a user to search and/or tag his/her personal collection of photos based on a tagged photo. Moreover, companies such as face.com specializing in face recognition techniques have introduced software to search and/or tag larger online photo repositories on social networking sites such as Facebook and Twitter. For example, one application of face.com's software is for finding lost photos of a user and his/her friends on Facebook sites [9].

## 24.9 Multimedia Management

Human faces are frequently seen in news, sports, films, home video, and other multimedia content. Indexing this multimedia content by face detection, face tracking, face recognition, and face change detection is important to generate segments of coherent video content for video browsing, skimming and, summarization. Together with speech recognition, natural language processing, and other image understanding techniques, face processing is a powerful tool for automatic indexing, retrieval, and access to the ever-growing digital multimedia content.

One difficulty of directly using face recognition in multimedia applications is that usually the gallery set is not available. The identity of the person whose face has been detected must be obtained through the multimedia content itself. Houghton [15] developed a "face-naming" method. His method finds names in Web pages associated with the broadcast television stations using three text processing methods and names in images using the optical character recognition (OCR) technique. The names are then linked to the faces detected in the images. The face detector is the FaceIt [11]. In this way, a gallery set is created. Queried about an image without a name, the "face-naming" system compares the faces in the image with the gallery and returns the identity of the face.

Ma and Zhang [24] developed an interactive user interface to let the user annotate a set of video segments that the face recognizer concludes to be belonging to the same nonenrolled person. They have used the face detection algorithm [31] to detect faces to help to extract key frames for indexing and browsing home video. Chan et al. [4] used face recognition techniques to browse video databases to find shots of particular people.

One integrated multimedia management system is the "Infomedia" project at Carnegie Mellon University [34]. This project aims to create an information digital video library to enhance learning for people of all ages. Thousands of hours of video content is indexed and archived for search and retrieval by users via desktop computers through computer networks. One of its indexing schemes is the face detection developed by Rowley et al. [31]. The detected human faces and text are used as a basis for significance during the creation of video segments. A small number of face images can be extracted to represent the entire segment of video containing an

individual for video summarization purposes. It supports queries such as “find video with talking heads” supported by face detection, “find interviews by Tim Russert” supported by face detection and video text recognition, and so on.

Another system is a multilingual, multimodal digital video library system, called iVIEW, developed at the Chinese University of Hong Kong [23]. Its face recognition scheme is similar to the one in Houghton’s article [15]. Faces detected are cross-referenced with the names detected by OCR on on-screen words. iVIEW is designed on Web-based architecture with flexible client server interaction. It supports access to both English and Chinese video contents. It also supports access via wired and wireless networks.

Wang and Chang [35] developed a system for real-time detection, tracking, and summarization of human faces in the video compressed domain at Columbia University. Their face detection component uses the MPEG compressed data to detect face objects and refine the results by tracking the movement of faces. The summaries of people appearance in the spatial and temporal dimensions help users to understand the interaction among people.

Because the orientation of faces or lighting conditions in most of the multimedia content is seldom controlled, face recognition accuracy is relatively low.

## 24.10 Human Computer Interaction

To achieve efficient and user-friendly human computer interaction, human body parts (e.g., the face) could be considered as a natural input “device”. This has motivated research on tracking, analyzing, and recognizing human body movements.

### 24.10.1 Face Tracking

Although the goal of such interfaces is to recognize and understand human body movements, the first step to achieve this goal is to reliably localize and track such human body parts as the face and the hand. Skin color offers a strong cue for efficient localization and tracking of human body parts in video sequences for vision-based human computer interaction. Color-based target localization could be achieved by analyzing segmented skin color regions. Although some work has been done on adaptive color models, this problem still needs further study. Wu and Huang [36] presented their investigation of color-based image segmentation and nonstationary color-based target tracking by studying two representations for color distributions. Based on the so-called D-EM algorithm, they implemented a nonstationary color tracking system. Figure 24.6 shows an example of face localization and tracking in a typical laboratory environment.



**Fig. 24.6** Tracking results based on color model

#### 24.10.2 Emotion Recognition

It is argued that for the computer to be able to interact with humans it must have the communication skills of humans, and one of these skills is the ability to understand the emotional state of the person. The most expressive way humans display emotions is through facial expressions. Cohen et al. [7] reported on several advances they have made in building a system for classifying facial expressions from continuous video input. They used Bayesian network classifiers for classifying expressions from video. Figure 24.7 shows four examples of real-time expression recognition. The labels show the recognized emotion of the user.

#### 24.10.3 Face Synthesis and Animation

A realistic three dimensional head model is one of the key factors in natural human computer interaction. A graphics-based human model provides an effective solution for information display, especially in collaborative environments. Examples include 3D model-based very low bit-rate video coding for visual telecommunication, audio/visual speech recognition, and talking head representation of computer agents. In noisy environments, the synthetic talking face can help users understand the associated speech, and it helps people react more positively during interactive sessions. It has been shown that a virtual sales agent inspires confidence in customers in the case of e-commerce, and a synthetic talking face enables students to learn better in computer-aided education [14].

Hong et al. [14] have successfully designed a system, called iFACE, that provides functionalities for face modeling and animation. The 3D geometry of a face is



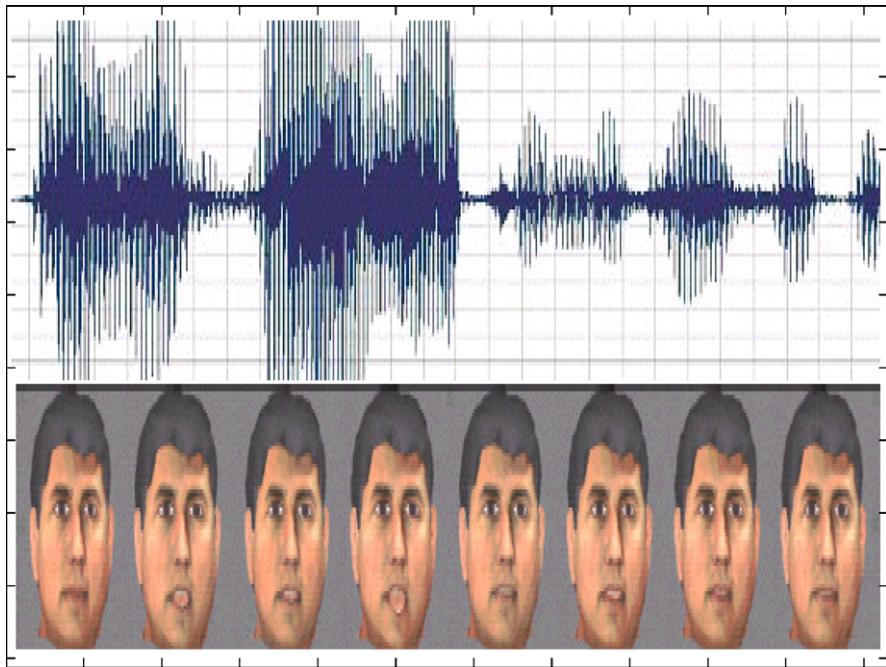
Fig. 24.7 Emotion recognition results

modeled by a triangular mesh. A few control points are defined on the face mesh. By dragging the control points, the user can construct different facial shapes. Two kinds of media, text stream and speech stream, can be used to drive the face animation. A display of the speech-driven talking head is shown in Fig. 24.8.

## 24.11 Other Applications

Many of the application scenarios in this section require close collaboration between face recognition systems and domain experts. The face recognition systems assist the domain experts.

- **Antique photo verification.** It is of great value for historians, biographers, and antique collectors to verify whether an antique photo of a person is genuine, given a true photo taken when that person is much older. The age difference and sometimes the low quality of the antique photo pose a great challenge for the face recognition systems.
- **Face images transmission.** Li et al. [21] coded the face images with a compact parameterized model for low bandwidth communication applications, such as videophone and teleconferencing. Instead of sending face images or video,



**Fig. 24.8** Speech driven face animation

they send robust feature representation of the faces to the other end of the channel so that by fitting a generic face model to the face feature representation a good reconstruction of the original face images can be achieved. Similarly, Lyons et al. [22] developed an algorithm that can automatically extract a face from an image, modify it, characterize it in terms of high-level properties, and apply it to the creation of a personalized avatar in an online Japanese sumo game application. The algorithm has potential applications in educational systems (virtual museums or classrooms) and in entertainment technology (e.g., interactive movies, multiple user role-playing communities).

- Chellappa et al. [5] listed several application scenarios that involve close collaboration between the face recognition system and the user or image domain expert. The interaction between the algorithms and known results in psychophysics and neuroscience studies is needed in these applications. We summarize these applications below; for detailed information see Chellappa et al. [5].
1. “Expert Identification”: An expert confirms that the face in the given image corresponds to the person in question. Typically, in this application a list of similar looking faces is generated using a face identification algorithm. The expert then performs a careful analysis of the listed faces.
  2. “Witness Face Reconstruction”: The witness is asked to compose a picture of a culprit using a library of features such as noses, eyes, lips, and so on. The “sketch” by the user is compared with all the images in the database to find the

closest matches. The witness can refine the “sketch” based on these matches. A face recognition algorithm can recompute the closest matches in the hope of finding the real culprit.

3. “Electronic Lineup”: A witness identifies a face from a set of face images that include some false candidates. This set of images can be the results from the “Witness Face Reconstruction” application by the face recognition algorithm.
4. “Reconstruction of Face from Remains” and “Computerized Aging”: Available face images are transformed to what a face could have been or what the face will be after some time.

## 24.12 Limitations of Current Face Recognition Systems

Although face recognition technology has great potential in the applications reviewed above, currently the scope of the application is still quite limited. There are at least two challenges that need to be addressed to deploy them in large-scale applications.

1. Face recognition technology is still not robust enough, especially in unconstrained environments, and recognition accuracy is still not acceptable, especially for large-scale applications. Lighting changes, pose changes, and time difference between the probe image and the gallery image(s) further degrade the performance. These factors have been evaluated in FRVT 2002 using some of the best commercial systems [29]. For example, in a verification test with reasonably controlled indoor lighting, when the gallery consisted of 37 437 individuals with one image per person and the probe set consisted of 74 854 probes with two images per person, the best three systems, on average, achieved a verification rate of 90% at a false alarm rate of 1%, 80% at a false alarm rate of 0.1%, and 70% at a false alarm rate of 0.01%. Although good progress has been made to increase the verification rate from 80% to 99% at a false alarm rate of 0.1%, as reported in FRVT 2006 [30], this level of accuracy may be (or may not be) suitable for an access control system with a small database of hundreds of people but not for a security system at airports where the number of passengers is much larger. When evaluating the performance with respect to pose change, with a database of 87 individuals the best system can achieve an identification rate of only 42% for faces with  $\pm 45^\circ$  left or right pose differences and 53% with  $\pm 45^\circ$  up or down pose differences. The elapsed time between the database and test images degrades performance at a rate of 5% per year of difference. Lighting changes between probe images obtained outdoors and gallery images taken indoors degrades the best systems, from a verification rate of 90% to around 60% at a false accept rate of 1%. The test results in FRVT 2002 can partly explain why several systems installed at airports and other public places have not received positive feedback based on their poor performance. One example is that the crowd surveillance system tested by Tampa, Florida police reported 14 instances of a possible criminal match in a 4-day session, but they were all false alarms. The Tampa police department has abandoned the system.

2. The deployment of face recognition-based surveillance systems has raised concerns of possible privacy violation. For example, the American Civil Liberties Union (ACLU) opposes the use of face recognition software at airports due to ineffectiveness and privacy concern [1]. In addition to listing several factors affecting the face recognition accuracy, such as change of hairstyle, weight gain, or loss, eye glasses or disguise, the ACLU opposes face recognition because “facial recognition technology carries the danger that its use will evolve into a widespread tool for spying on citizens as they move about in public places.”

## 24.13 Conclusions

We reviewed many face recognition systems in various application scenarios. We also pointed out the limitations of the current face recognition technology. The technology has evolved from laboratory research to many small-, medium- or, large-scale commercial deployments. At present, it is most promising for small- or medium-scale applications, such as office access control and computer log in; it still faces great technical challenges for large-scale deployments such as airport security and general surveillance. With more research collaborations worldwide between universities and industrial researchers, the technology will become more reliable and robust.

Another direction for improving recognition accuracy lies in a combination of multiple biometrics and security methods. It can work with other biometrics such as voice-based speaker identification, fingerprint recognition, and iris scan in many applications. For security purpose at airports, face recognition systems can also work together with X-ray luggage scanners, metal detectors, and chemical trace detectors at security checkpoints.

This chapter concludes with the following description of how face recognition could be used in our daily lives in the near future, although some of them are already in place.

If we drive to work, a face recognizer installed in the car will decide whether to authorize our usage of the vehicle before starting the engine. If we choose to take a bus or subway to work, our prepaid boarding pass will be verified by a face recognizer comparing the photo on the pass and live captured pictures of our faces. At the entrance of the office building, we go through a face recognition based access control system that compares our face images with those in its database. We sit down in front of the office computer, a face recognizer in it runs its face recognition algorithm before we log on. When we go to a secure area in the office building, the security check is carried out by another face recognizer. On a business trip, when we use the smart ATM, we are subject to a face recognizer of the bank system. At the airport, our boarding pass and passport or identity card are screened by the airport’s face recognizer for passenger security purpose. When we go to a retail store, restaurant, or a movie theater, the cameras equipped with cash registers would be aimed at our faces to compare our pictures with those in a customer database to identify us, if not, we could complete the purchase by using a PIN (personal identification

number). After the cash register had calculated the total sale, the face recognition system would verify us and the total amount of the sales would be deducted from our bank accounts [16]. When we go back home, a face recognition based home security system makes sure we are living in the house before we open the door.

**Acknowledgements** We sincerely thank Dr. Ying Wu, Northwestern University, Dr. Lawrence Chen, Kodak Research Lab, Dr. Javier Ruiz-del-Solar, Universidad de Chile, Chile, and Dr. Julian L. Center, Jr., Lau Technologies for providing some of the pictures and their permission to use them in the paper. We also want to thank Dr. Anil K. Jain, Michigan State University for giving us helpful suggestions on improving the manuscript.

## References

1. ACLU. <http://archive.aclu.org/features/f110101a.html>
2. Bryliuk, D., Starovoitov, V.: Access control by face recognition using neural networks and negative examples. In: Proceedings of the 2nd International Conference on Artificial Intelligence, pp. 428–436 (2002)
3. Bunney, C.: Survey: face recognition systems. *Biom. Technol. Today* 8–12 (1997)
4. Chan, Y., Lin, S.-H., Kung, S.: Video indexing and retrieval. In: Sheu, B.J., Ismail, M., Wang, M.Y., Tsai, R.H. (eds.) *Multimedia Technology for Applications*. Wiley, New York (1998)
5. Chellappa, R., Wilson, C., Sirohey, S.: Human and machine recognition of faces: a survey. *Proc. IEEE* **83**(5), 704–740 (1995)
6. Cognitec. <http://www.cognitec-systems.de/index.html>
7. Cohen, I., Sebe, N., Cozman, F.G., Cirelo, M.C., Huang, T.: Learning Bayesian network classifiers for facial expression recognition using both labeled and unlabeled data. In: Proceedings of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR 2003) (2003)
8. Eickeler, S., Birlinghoven, S.: Face database retrieval using pseudo 2D hidden Markov models. In: Proceeding of IEEE Int. Conf. on Face and Gestures (FG 2002) (2002)
9. Facebook facial-recognition tagger goes live. <http://blogs.wsj.com/digits/2009/11/11/facebook-facial-recognition-tagger-goes-live>
10. FaceGate. <http://www.premierelect.co.uk/faceaccess.html>
11. FaceIt. <http://www.identix.com>
12. FaceIt-Hist. [http://www.identix.com/company/comp\\_history.html](http://www.identix.com/company/comp_history.html)
13. FaceKey. <http://www.facekey.com>
14. Hong, P., Wen, Z., Huang, T.: IFace: a 3D synthetic talking face. *Int. J. Image Graph.* **1**(1), 19–26 (2001)
15. Houghton, R.: Named faces: putting names to faces. *IEEE Intell. Syst.* **14**(5), 45–50 (1999)
16. [http://en.wikipedia.org/wiki/Facial\\_recognition\\_system](http://en.wikipedia.org/wiki/Facial_recognition_system)
17. Imagis. <http://www.imagistechnologies.com>
18. Imagis-Beach. [http://cipherwar.com/news/01/imagis\\_big\\_brother.htm](http://cipherwar.com/news/01/imagis_big_brother.htm)
19. Konen, W., Schulze-Krüger, E.: ZN-face: a system for access control using automated face recognition. In: Proceedings of the International Workshop on Automatic Face and Gesture Recognition, pp. 18–23 (1995)
20. Kudhinsky, A., Pering, C., Creech, M.L., Freeze, D., Serra, B., Gvvezdka, J.: FotoFile: a consumer multimedia organization and retrieval system. In: Proceedings of CHI'99, pp. 496–503 (1999)
21. Li, H., Roivainen, P., Forchheimer, R.: 3D motion estimation in model-based facial image coding. *IEEE Trans. Pattern Anal. Mach. Intell.* **15**(6), 545–555 (1993)
22. Lyons, M., Plante, A., Jehan, S., Inoue, S., Akamatsu, S.: Avatar creation using automatic face recognition. In: Proceedings, ACM Multimedia 98, pp. 427–434 (1998)

23. Lyu, M.R., Yau, E., Sze, S.: A multilingual, multimodal digital video library system. In: ACM/IEEE Joint Conference on Digital Libraries, JCDL 2002, Proceedings, pp. 145–153 (2002)
24. Ma, W.-Y., Zhang, H.: An indexing and browsing system for home video. In: Proc. of 10th European Signal Processing Conference (2000)
25. Martinez, A.: Face image retrieval using HMMs. In: Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries (1999)
26. Maximus. <http://www.maximus.com/corporate/pages/smартcardsvs>
27. Navarrete, P., del Solar, J.: Interactive face retrieval using self-organizing maps. In: Proceedings, 2002 Int. Joint Conf. on Neural Networks: IJCNN2002 (2002)
28. Omron. <http://www.omron.com>
29. Phillips, P., Grother, P., Michaels, R., Blackburn, D., Tabassi, E., Bone, M.: Face recognition vendor test 2002: evaluation report. <http://www.frvt.org>
30. Phillips, P., Scruggs, W., O'Tools, A., Lynn, P., Bowyer, K., Schott, C., Sharpe, M.: FRVT 2006 and ICE 2006 large-scale results. <http://www.frvt.org>
31. Rowley, H., Baluja, S., Kanade, T.: Neural network-based face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**, 22–38 (1998)
32. Rui, Y., Huang, T.S., Chang, S.-F.: Image retrieval: current techniques, promising directions and open issues. *J. Vis. Commun. Image Represent.* **10**(4), 39–62 (1999)
33. Viisage. <http://www.viisage.com>
34. Wactlar, H., Smith, T.K.M., Stevens, S.: Intelligence access to digital video: informedia project. *Computer* **29**(5), 46–52 (1996)
35. Wang, H., Chang, S.-F.: A highly efficient system for automatic face region detection in mpeg video sequences. *IEEE Trans. Circuits Syst. Video Technol.* **7**(4), 615–628 (1997). Special Issue on Multimedia Systems and Technologies
36. Wu, Y., Huang, T.: Nonstationary color tracking for vision-based human computer interaction. *IEEE Trans. Neural Netw.* **13**(4) (2002)
37. Xiong, Z., Chen, Y., Wang, R., Huang, T.: Improved information maximization based face and facial feature detection from real-time video and application in a multi-modal person identification system. In: Proceedings of the Fourth International Conference on Multimodal Interfaces (ICMI'2002), pp. 511–516 (2002)
38. Zhao, W., Chellappa, R., Rosenfelda, A., Phillips, J.: Face recognition: a literature survey. Technical Report, CS-TR4167R, University of Maryland (2000)

# Chapter 25

## Large Scale Database Search

Michael Brauckmann and Christoph Busch

This chapter focuses on large scale search systems for the biometric and face recognition, in particular, on issues related to scalability, system throughput, biometric accuracy, and database sanitization.

### 25.1 Introduction

The accuracy of core face recognition algorithms has been increasing continuously over the past three decades. The series of face recognition vendor tests conducted by the U.S. National Institute of Standards and Technology (NIST), indicated a reduction by factor of two over a development period of four years in the late ninetieth. The first NIST-testing resulted for a fixed false match error rate of 0.001 in a false nonmatch-error-rate of FNMR = 0.79 as documented in the FERET 1993 report. Errors were reduced from FNMR = 0.54 in the FERET 1997 test down to 0.20 in FRVT 2002 test. Since then the biometric performance of commercially available systems improved even by factor of ten every four years. At the same false accept level, a FNMR = 0.026 was observed in FRVT 2006 and a FNMR = 0.003 in the MBE 2010 test report [1]. However, the likelihood of a false match of a subject to any of the enrolled references in a large database increases with the size of the database and thus the false positive identification rate (FPIR) is linear dependent of the size of the enrollment data base [6]. Thus, specifically for large scale applica-

---

M. Brauckmann (✉)  
L-1 Identity Solutions AG, Bochum, Germany  
e-mail: [MBrauckmann@l1id.com](mailto:MBrauckmann@l1id.com)

C. Busch  
Hochschule Darmstadt/Fraunhofer IGD, Darmstadt, Germany  
e-mail: [christoph.busch@igd.fraunhofer.de](mailto:christoph.busch@igd.fraunhofer.de)

tions high accuracy is required to perform successful and reliable searches in the enrollment database, which can easily exceed the million entry boundary in practice. One of the largest enrollment databases today is the U.S. Visit database with 110 million enrollment records that were collected after six years of operation until the year 2010 [1].

Accuracy is often considered as the most important property at the algorithmic level. However at the system or application level dealing with large databases, operators do require the consideration of aspects other than the accuracy of the underlying algorithm. Design principles comprise integrability, flexibility, scalability, and durability.

Integrability is important to allow integration services to deploy systems in existing environments. A key factor for large scale system is the adherence to information technology standards with regards to communication protocols and platforms. The system needs flexibility to allow choices on hardware platforms, operating systems, algorithms, and business applications. Further the design principle scalability is relevant, which allows the extension of the system when databases grow or throughput requirements, i.e. the number of searches to be performed in a certain time period, change. Durability encompasses measures for fault tolerance, recovery scenarios, backup and replication, redundancy and no single point of failure.

The core functions required in a biometric system are enrollment, update, deletion, and search of biometric data in a enrollment database. Execution times, hardware requirements, scalability, integrability, flexibility, fail-safety and many other aspects gain importance as they become key factors of the system when the data sets get large.

Use cases can have different underlying business processes such as forensic searches with adjudication of a list of candidate reference record for a data subject or civil ID systems where alarms are raised in case of potential ID fraud. The application requirements do impact the system processes and protocols and need to be reflected by the system behavior. Requirements can change the importance of a single factor dramatically for example, it makes a big difference whether a process requires a system to answer within a second or within an hour, or whether 10 searches per hour or 1000 searches per hour need to be performed.

This chapter focuses on the biometric and face recognition related aspects of large scale search systems, focusing on issues related to scalability, system throughput, biometric accuracy, and database sanitization.

## 25.2 Design Objectives and Cost Criteria

The design objective for a large scale system is a multi-dimensional optimization task. In general an optimal system should perform a

- certain **number of searches**
- in a certain **time period**
- on a database of a certain **size**

- on a certain **hardware architecture**
- at a certain **accuracy level**

The business process defines, which of these criteria are variables and which ones are fixed parameters. In addition, variables are commonly be bound to a certain range and suppliers need to commit themselves that deployed systems do not exceed those limits.

*Example 25.1* In the case of a passport application system, a single application transaction might be coupled with a duplicate enrollment check (DEC) to detect potential applications under different identities that are linked to the same biometric characteristic. For the de-duplication confirmation, the required number of searches in a certain time period is bound by a lower limit since a state has a certain number of applicants for passports per day. Operating at a lower throughput would create a backlog increasing over time. Also, the lower bound for the size of the database is given by the number of passport holders. Solving the problem at the boundaries leaves the option to trade accuracy for hardware for example, by using a faster, but less accurate algorithm on the same hardware.

*Example 25.2* In the case of a forensic search system, accuracy is paramount. The search speed or throughput, that is, number of searches per time period is not critical since as long as the number of cases per day is significantly lower than the capacity of the system and thus piling up of cases is avoided. Within these constraints, it is most important to solve the case, but solving more cases in a certain period is of value. The size of the database is known, obviously. This constellation allows to accept a reduced throughput and consequently a longer system reaction is acceptable, if at the same point in time a better accuracy can be expected.

Beyond the criteria listed here an essential further aspect for the design is the direct communication with the data subject or indirect communications. For indirect communication, the data subject has provided his biometric data in a capture session and biometric searches are conducted off-line and reported back to the individual eventually. However, in direct communication such as for instance second level background checks conducted at a border the data subject is waiting in the control process until the result of the biometric search allows any further step in the control process. Aside from the pragmatic constraints to avoid any piling of transactions the total duration of a single transaction is essential. If the single transaction exceeds a given time limit the Biometric system would be considered inconvenient and incompatible with given border control regulations.

In order to calculate solutions that lie within the bounds of the variables and the specific parameters, it is necessary to describe the system in a model that predicts the variable interrelations. Thus, variables and parameters specified above serve as the input for an optimization process. Optimization in general is related to a cost function to be minimized. The definition of such a cost function is not straightforward since some decision criteria are not known, for example, a decision maker decides

to save money on hardware. In practice, cost in real currency plays an important role and multiple variations need to be created and provided during the procurement process.

### 25.3 Scalability

A single processor has an upper limit on the search speed and the memory capacity. Thus, the number of biometric templates that can be stored in the computer's memory is limited by an upper bound while the acceptable response time of the system for a single transaction defined by the application criteria again constitutes an upper bound limit. A scalable system provides methods that allow to fulfill the operational task within the agreed criteria even when demands grow.

Scalability is categorized in **vertical scaling** and **horizontal scaling**. Vertical scaling, also known as **scale-up** takes place on a single **node** of a system and increases resources such as processing power to enable a larger number of processes and memory capacity to store a larger amount of biometric templates. Horizontal scaling, also known as **scale out** adds more nodes to the system. Horizontal scaling provides a way to maintain a throughput requirement that is larger than the largest possible throughput on a single node, for example, if a node has a search speed of one search per second, but a search speed of two searches per second is required then two nodes will be necessary.

As a consequence, a scalable system produces distributed results across processes or processors. In consequence, multiple individual number or decisions need to be merged to a single final result. In a practical implementation of an identification application, the candidate list with ranked references need to be merged while respecting comparison scores when computing the overall rank position. Furthermore, modules and components need to support parallelism and interleaved access to a unique or segmented enrollment database.

### 25.4 System Throughput and Biometric Accuracy

When considering the throughput of a single node, there are strategies for a given hardware platform to optimize accuracy and speed as well as the capacity of the database. For an *accurate* system, we expect accuracy metrics that is, on the algorithm level the false nonmatch rate (FNMR) and false match rate (FMR) as well as on the system level failure to acquire rate (FTA) to be *low*; thus for a system a low false accept rate (FAR) and a low false reject rate (FRR) depends on algorithm accuracy according to the relationship [6].

$$\text{FAR} = \text{FMR} \cdot (1 - \text{FTA}) \quad (25.1)$$

and

$$\text{FRR} = \text{FTA} + \text{FNMR} \cdot (1 - \text{FTA}). \quad (25.2)$$

For the embedded algorithms we assume the following to hold:

- a more accurate algorithm is computationally more expensive than a less accurate algorithm, that is, high accuracy implies slow transaction speed
- a more accurate algorithm compares biometric probes to biometric templates, which are larger in size than the template of a less accurate algorithm, that is, high accuracy implies high memory consumption

To cope with contradicting objectives of high accuracy at low memory consumption with high search speed, various concepts can be followed such as for instance the **multi-stage comparison** or **binning** of the enrollment database. The concept of multi-stage comparison is elaborated in Sect. 25.4.1 while binning on three dimensional databases is considered in Sect. 25.4.6.

### 25.4.1 Multi-stage Comparison

Multi-stage comparison analyzes the database in a first stage with a very fast algorithm, which consumes a small amount of the main memory while it operates at a moderate accuracy. It is expected that this algorithm ranks a mate in the upper area of the ranks (e.g., ranks one to 10 000), but not necessarily at the top. The next stage consists of an algorithm which is more accurate than the previous one and thus slower. This algorithm will examine only the upper area of the ranks which will take much less time than examining the whole database. The results of the first stage will provide indices of mid-sized templates and while comparison is conducted in the second stage on longer feature vectors and optimal accuracy there is a good chance to find the mate therein. It is expected that this algorithm ranks a mate even higher since it is more accurate. Again, the upper ranks of this result (e.g., ranks one to 1000) serve as the input for the next algorithm operating, which is even more accurate, but again slower. This process is reiterated until comparison is conducted on the full-sized feature vector in the final stage and the final candidate is reached.

Applying the steps above with all templates in the computers main memory is a very resource consuming strategy. While it is extremely slow to load the whole database from a hard disk drive this is affordable for a relatively small number of biometric templates. Thus, once the number of biometric templates to analyze is reduced, loading from disk can be considered for memory economy.

### 25.4.2 Meaningful Scores

The set of scores returned from a search should ideally have a meaning that is useful in the process. Scores may reflect the false acceptance risk enabling the operator to tell the likelihood of an accidental match on a certain database size. The false acceptance risk associated with a score is in general determined empirically on sample data. Alternatively, the database size may be incorporated.

This allows setting thresholds to accomplish a tolerable false alarm rate.

*Example 25.3* Let scores be related to the false acceptance risk. For convenience a mapping that represents  $-\log_{10}(\text{false acceptance risk})$  is used such that e.g. a score of 3 corresponds to a false acceptance risk of  $0.001 = 10^{-3}$ . This means it is expected to have one false positive match in 1000 comparisons on average. Assuming an enrollment database of a size of 1 000 000 entries a threshold of 6 would create one false positive match per search since each search comprises 1 000 000 comparisons in the database. To get one false positive match in every 10 searches the threshold needs to be set to 7, to get one in every 100 searches the threshold should be 8 etc.

Alternatively, the database size may be incorporated in the score by means of  $-\log_{10}(\text{false acceptance risk}) - \log_{10}(\text{database size})$ .

### 25.4.3 False Positive Identification in Large Scale Systems

For many large scale identification applications we are concerned to avoid a potential false positive alarm, while we do not bother to which biometric reference the probe falsely matched. Thus we can estimate the false positive rate for an open-set systems, based on the FMR [6].

$$\text{FPIR} = (1 - \text{FTA}) \cdot (1 - (1 - \text{FMR})^N) \quad (25.3)$$

where FPIR is the False-Positive-Identification-Rate. For a small FMR we can substitute in (25.3)

$$(1 - \text{FMR})^N \approx 1 - N \cdot \text{FMR} \quad (25.4)$$

and thus under the assumption of  $\text{FTA} = 0$  we derive

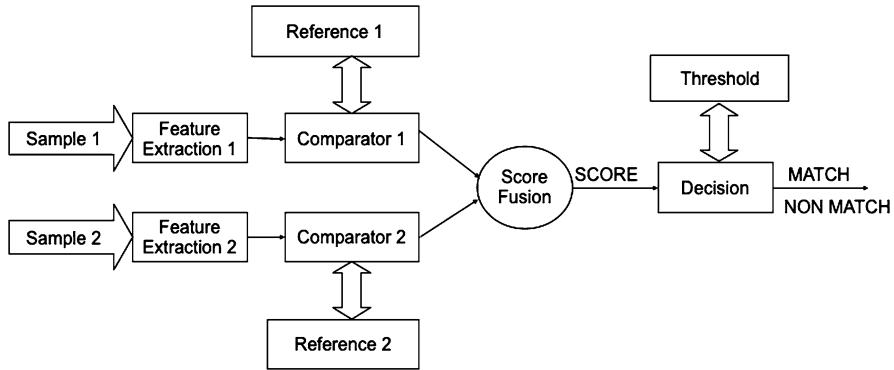
$$\text{FPIR} = (1 - 0) \cdot (1 - (1 - N \cdot \text{FMR})), \quad (25.5)$$

$$\text{FPIR} = N \cdot \text{FMR}. \quad (25.6)$$

As we can see the likelihood for a false positive identification is linearly dependent with the size of the database. However, such estimates cannot take account of correlations in the comparisons involving the same data subject, and consequently can be quite inaccurate.

### 25.4.4 Fusion

The availability of multiple algorithms in multi-stage comparison scenarios allows for improved accuracy by means of fusion. The concept of multi-biometric fusion is divided into the categories feature level fusion, score level fusion, and decision level



**Fig. 25.1** Score level fusion in a multi-biometric system, where multiple information channels (e.g., multiple algorithms or multiple modalities) are fused

fusion. For details on fusion strategies, the reader is referred to the ISO Technical Report on Multi-Modal and Other Multi-Biometric Fusion [2]. In the design of large scale systems, score level fusion as depicted in Fig. 25.1 is oftentimes preferred.

This is because the concept allows adding new algorithms at the system level without deeper knowledge of the feature structure while feature level fusion would require such knowledge. The decision level has the disadvantage that it makes sorting difficult if not impossible as a result would be a decision between hit and non hit. The downside of the feature-level fusion strategy is that a comprehensive score normalization is a precondition, in order to adjust to a standardized metric (i.e., similarity scores versus dissimilarity scores) and to a harmonized distribution of the scores. While a large variety of score normalization methods has been proposed, a simple Z-score normalization can be sufficient for a simple application, which computes a normalized score according to

$$S^* = \frac{S - S_{\text{Mean}}}{S_{\text{SD}}} \quad (25.7)$$

where  $S$  is the score rendered by a single algorithm,  $S_{\text{Mean}}$  is the mean of both the impostor and genuine score distribution for this algorithm and  $S_{\text{SD}}$  the respective standard deviation. The underlying assumption of equal standard distribution that is symmetric about the mean might not be met in the reality but is a good approximation. It is relevant to invest efforts to measure those a priori numbers for the target population. Subsequent fusion of scores in a specific comparison can weight the accuracy of the individual algorithms  $i$  according to

$$S = \sum_{i=1}^M w_i S_i^* \quad (25.8)$$

where for algorithm  $i$  the weight  $w_i$  is inversely proportional to the measured biometric performance metrics for the algorithm  $i$  and thus determines to which extend a contribution to the joint decision of all  $M$  algorithms should be made.

It is noteworthy that all the calculations above describe the behavior of a large ensemble on average. Z-score normalization is a popular candidate whenever the available scores reflect the whole distribution. Unfortunately, this assumption is violated in the case of multi-pass systems. In the higher stages of such systems only the tail of the distribution, represented in the top ranks, is available.

Thus, it is worthwhile to take a deeper glance at the statistics of top ranks. A search returns a list of items ordered by their score. This allows for a refined modeling by means of the order statistic. Order statistic determines the distribution of random samples drawn from a known distribution and placed in ascending order. Minimum and maximum are special cases in the order statistic. Formally, a sample of  $N$  unsorted variables  $X_1, \dots, X_N$  is sorted such that  $X_{(1)} < X_{(2)} < \dots < X_{(N)}$ , that is, the index denotes the respective rank after sorting. In the sequel, we will derive the probability density function<sup>1</sup> and the corresponding cumulative distribution function<sup>2</sup> for the sorted variables at a certain rank based on the PDF and CDF of the unsorted variables. Let  $f(x)$  and  $F(x)$  denote the PDF and the CDF<sup>3</sup> of the unsorted variables while  $f_{X(r)}(x)$  and  $F_{X(r)}(x)$  denote the PDF and the CDF of the sorted variables at rank  $r$ .

If the probability density function  $f(x)$  and the distribution function  $F(x)$  are known, then the probability density function  $f_{X(r)}(x)$  is given by

$$f_{X(r)}(x) = \frac{N!}{(r-1)!(N-r)!} [F(x)]^{r-1} [1 - F(x)]^{N-r} f(x). \quad (25.9)$$

It is convenient to use the multinomial coefficient which is an extension of the binomial coefficient. It describes the number of ways to order elements for two or more categories and is defined by

$$\binom{n}{k_1, k_2, \dots, k_m} = \frac{n!}{k_1! k_2! \dots k_m!} \quad (25.10)$$

with  $k_1, k_2, \dots, k_m$  being the numbers of elements in the categories  $1, 2, \dots, m$  and where  $n$  is the total number of elements.

*Example 25.4* (Multinomial coefficient) We want to know how many ways we can arrange 2 blue, 3 red, and 10 green balls in. The multinomial coefficient for three categories is defined by  $\binom{a+b+c}{a,b,c} = \frac{(a+b+c)!}{a!b!c!}$  and tells us  $\frac{15!}{2!3!10!} = 30\,030$  ways.

With this definition, we can rewrite (25.9) by

$$f_{X(r)}(x) = \binom{N}{r-1, 1, N-r} [F(x)]^{r-1} [1 - F(x)]^{N-r} f(x). \quad (25.11)$$

<sup>1</sup>Probability Density Function—PDF.

<sup>2</sup>Cumulative Distribution Function—CDF.

<sup>3</sup>We omit the index  $X$  in  $f_X(x)$  and  $F_X(x)$  for brevity.

In Sect. 25.4.4.1, we will take advantage from the fact that we want scores to be related to the false acceptance risk. The false acceptance risk is a uniform distribution with a probability density function at a constant value of 1 in the interval  $[0, 1]$ . Thus, (25.9) simplifies to

$$f_{X(r)}(x) = \binom{N}{r-1, 1, N-r} x^{r-1} (1-x)^{N-r} \quad x \in [0, 1]. \quad (25.12)$$

So far we can tell the distribution of scores at any rank given that the underlying assumptions hold. It is of particular interest to analyze conditional distributions.

*Example 25.5* (Order statistics) We may ask: what is the probability of exceeding a score of  $x$  at rank  $j$  given that we saw a score of  $y$  at rank  $k$ ?

To solve the question in Example 25.5, we need the conditional probability. In general, the conditional probability density function of  $X$  given the occurrence of the value  $y_o$  of  $Y$  can be calculated with the joint probability density  $f_{X,Y}(x, y)$  and the marginal density  $f_Y(y)$  and is given by

$$f_X(x | Y = y_o) = \frac{f_{X,Y}(x, y_o)}{f_Y(y_o)}. \quad (25.13)$$

For ranks  $j < k$ , we are interested in the joint density  $f_{X(j), X(k)}(x, y)$ . Expressed with a multinomial coefficient this leads to

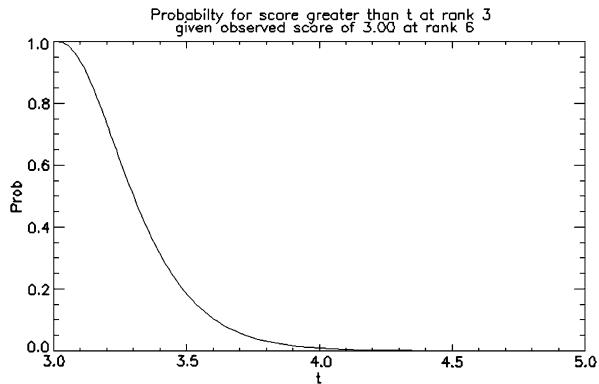
$$\begin{aligned} f_{X(j), X(k)}(x, y) \\ = \binom{N}{j-1, 1, k-j-1, 1, N-k} F(x)^{j-1} f(x) (F(y) - F(x))^{k-j-1} \\ \times f(y) (1 - F(y))^{N-k}. \end{aligned}$$

Here we have 5 categories, the  $j-1$  elements smaller than rank  $j$ , one element at rank  $j$  and one at rank  $k$ ,  $k-j-1$  elements between rank  $j$  and rank  $k$  and  $N-k$  elements above rank  $k$ . We find

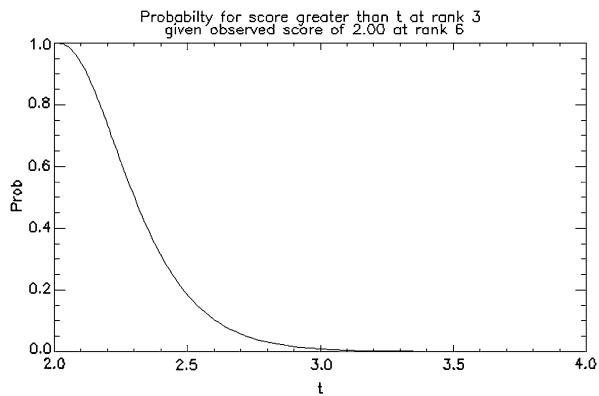
$$\begin{aligned} f_{X(j)}(x | X(k) = y_o) &= \frac{f_{X(j), X(k)}(x, y_o)}{f_{X(k)}(y_o)} \\ &= \binom{k-1}{j-1, 1, k-j-1} \frac{F(x)^{j-1} f(x) (F(y_o) - F(x))^{k-j-1}}{F(y_o)^{k-1}}. \end{aligned} \quad (25.14)$$

Interestingly, we don't have the gallery size  $N$  any longer in term above. This is encoded in the observation of the rank  $k$ .

**Fig. 25.2** Probability of observing a  $-\log\text{FAR}$ -mapped score greater than  $t$  at rank 3 given that we observed a score of 3 at rank 6



**Fig. 25.3** Probability of observing a  $-\log\text{FAR}$ -mapped score greater than  $t$  at rank 3 given that we observed a score of 2 at rank 6



#### 25.4.4.1 Uniform Distribution

The uniform distribution is of particular interest since the  $-\log\text{FAR}$ -mapped scores denoted by  $s_L$  have a correspondence to this distribution, assuming that the mapping generalizes well.  $t = 10^{-s_L}$  is uniformly distributed for impostor scores and relates to the empirical risk of observing an impostor score larger than  $s_L$ . For the uniform distribution,  $f(x) = 1$ ,  $0 \leq x \leq 1$  and  $F(x) = x$ ,  $0 \leq x \leq 1$  holds.

$$f_{X_{(j)}, X_{(k)}}(x, y) = \binom{n}{j-1, 1, k-j-1, 1, N-k} x^{j-1} (y-x)^{k-j-1} (1-y)^{n-k}.$$

The conditional probability density function in this case becomes

$$f_{X_{(j)}}(x | X_{(k)} = y_o) = \binom{k-1}{j-1, 1, k-j-1} \frac{x^{j-1} (y_o - x)^{k-j-1}}{y_o^{k-1}}.$$

We can see from Figs. 25.2 and 25.3 that the shape of the curve does not depend on the observed score for uniform distributions. The math above allows to calculate the likelihood that an observed score originates from an impostor.

*Example 25.6* Figure 25.2 tells us that seeing a score of 4.0 or higher at rank 3 given that we saw an impostor score of 3.0 at rank 6 is a fairly unlikely statistical event, that is, statistically this happens in less than 5% of all cases. Thus, we can be confident in the assumption that this score belongs to a client.

Care has to be taken in cases where the underlying assumptions for the approach are violated. Such a case may arise for databases with statistical distributions quite different from the statistical distributions of the database used to calculate  $-\log\text{FAR}$  mapping.

#### 25.4.5 Filtering and Demographic Information

While previous considerations employed biometric information extracted from the captured biometric sample there is oftentimes additional non-biometric data available such as the age of the subject, height, gender, etc. In many cases, such as for the subject height the information can be exploited at no extra costs as it can be contained in the reference data (in a respective field in ISO 19794-5 [4]) and is a side-information from the capturing process, in which the capture device has been adjusted to the eyelevel. Sorting out subjects based on demographic knowledge can tremendously reduce the number of biometric comparisons required in a search. The ratio of the number of biometric comparisons made after filtering by means of demographic data and the whole population size in the enrollment database is called penetration rate. Employing demographic information has various advantages:

- The search can be accelerated as a fewer number of comparisons are required.
- Accidentally high scoring impostors may be filtered out.
- Chances of clients to end up in the top  $N$  ranks is a function of the database size. Reducing the database size increases this chance. Moreover, in multi-stage comparison the first stage is typically the weakest stage in terms of accuracy. While the more accurate higher stages might have the discriminative power to rank a mate high the first stage may fail to propagate it to the higher stages. Like above, this depends on the database size. Thus, the chances for propagating a mate to the next stage are raised by demographic filtering.

#### 25.4.6 Binning

The objective of binning reduces the search space and thus directly the number of references that are compared to a biometric probe. The concept is equivalent to



**Fig. 25.4** Anthropometric measurements according to Alphonse Bertillon. (Image source: [http://c1.ac-images.myspacecdn.com/images02/138/l\\_70d895b111834c3c9e68e89cc861055c.jpg](http://c1.ac-images.myspacecdn.com/images02/138/l_70d895b111834c3c9e68e89cc861055c.jpg))

the approach described in Sect. 25.4.5 while for **binning** intrinsic information that is contained in the biometric sample is exploited. A prominent case for such an approach has been in use in Automatic Fingerprint Identification Systems (AFIS) that pre-segmented the enrollment database in several bins according to the Henry pattern (i.e., left loop, right loop, whorl, arch). Likewise the pattern is determined for the probe sample and the database search is conducted only on the pre-selected bin that is labeled with the corresponding Henry pattern.

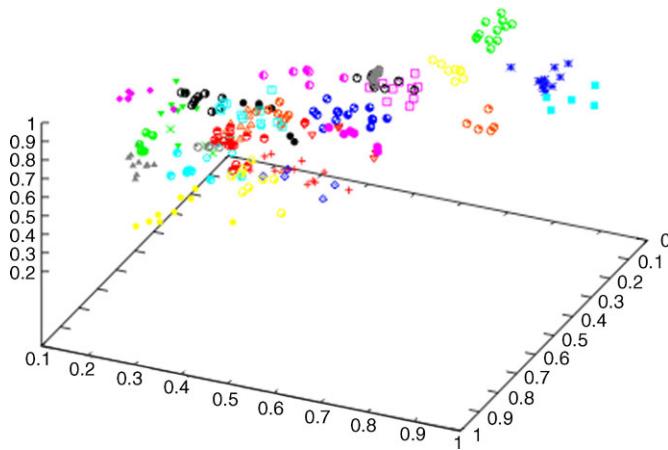
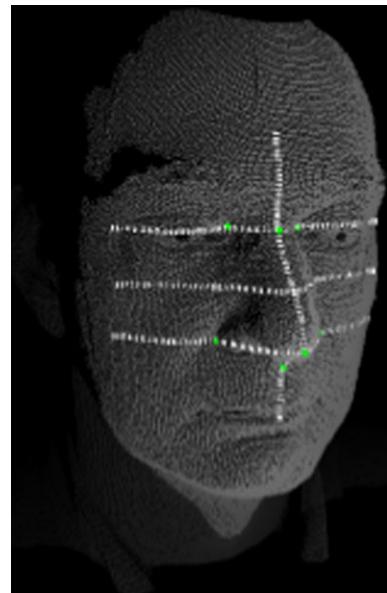
Applying binning to a face recognition system is not as straight-forward possible as one might wish. By experience we can tell that the human face recognition system and its biological neural network can reliably distinguish between male and female no matter whether the full three-dimensional face information or just a two-dimensional image supports that decision. On the contrary artificial neural networks and other pattern classifiers struggle with exactly the same task.

However, under the assumption that for the face-recognition system a multi-modal information with 2D textured information and 3D shape information is available such binning strategies based on intrinsic biometric data can be accomplished. A 3D face capturing device maintains at a known scale the given metric of the subject in the sampled model space. Measurements between distinct landmarks in the captured 3D-model can be conducted and will provide various robust base-lines. Thus, a preselection of the search space is achieved based on analysis of fixed base-lines in the captured model. With this approach, large scale systems can implement what Alphonse Bertillon has applied in forensic applications more than 150 years ago with his anthropometric measurements (see Fig. 25.4).

As illustrated in Fig. 25.5, landmarks that provide the respective base-lines can be extracted quite reliable from the model, when the symmetric property of the face is assumed [7].

These base-lines and other anatomical Bertillion features such as curvatures on the central profile can effectively partition the enrollment database. While this will not improve the biometric performance the approach can significantly reduce the

**Fig. 25.5** Automatically detected landmarks in 3D-model (from Mracek [7])



**Fig. 25.6** Distribution of selected anatomical Bertillion features in the feature space (from Mracek [7])

transaction time, as only a preselected bin is investigated, in which anatomical features of all references correspond to the biometric probe.

This application has a significant additional challenge over the above AFIS application that needs to be addressed. While the AFIS approach is based on discrete enumerated properties (loop, whorl, arch), the analysis of the 3D-shape results in values in a continuous scale (i.e., a distance metric or a curvature metric). In consequence, the derived bins correspond to a quantization of those measured values and

misclassifications (binning errors) are likely to happen at decision boundaries. This needs to be addressed with suitable decision strategies such as fuzzy logic. A further compensating strategy is to exploit the anatomical Bertillion features merely to sort the reference in a sequence that mate are likely to be visited first, while nonmate will only be considered if time allows. In this respect, the binning approach can effectively be combined with the multi-stage comparison as described in Sect. 25.4.5.

Binning in its strictest sense is a classification task executed prior to the comparison. Binning errors can be mitigated by adding neighbored bins. It has to be weighed during the design process whether or not this is a desirable strategy. The most influencing factor is the dimension of the feature space. The higher the dimension, the larger the number of neighbors that need to be taken into account. Employing an appropriate distance (or similarity) between the features is sometimes an attractive alternative, since it eliminates the mentioned boundary problems. The decision whether binning should be favored over the distance (or similarity) method depends on

- comparison speed
- memory requirements

Omitting items during a search is certainly more effective than comparing each item with the probe, but for very fast comparators the difference may not be dramatic. Binning allows to reserve a single partition of the memory per bin. This ion turn can lead to very effective searches. However, if the spread of database items over various memory partitions is compensated by the gain in accuracy from a cost perspective the later may be preferred.

## 25.5 Database Sanitization

The information stored in databases has oftentimes a number of incorrect entries not known to the owner of the database. Such cases can arise from operator-typos occurring when meta information is entered during the enrollment process, subjects with similar names that get assigned wrongly and many other reasons. In general, there are two types of error:

1. Two different individuals share the same unique identifier
2. Two unique identifiers point to the same individual

Both errors are reflected in the respective biometric scores. The first type leads to very *low* genuine scores, when mated samples are analyzed, while the second type leads to very *high* imposter scores, when nonmated samples are analyzed. Thus, while biometric systems are usually designed to detect fraud (i.e., duplicate enrollment attempts) the very same technology provides a powerful method to perform data sanitization. An efficient support for a semi-automatic consistency check of a large scale database is given, if identifier labels for the minimum imposter scores and the maximum imposter scores are filtered and can be validated visually. This process of database sanitization becomes more relevant as the size of the database grows.

## 25.6 Conclusions

The design of large scale systems requires a series of considerations beyond the ones covered by the algorithmic design. Template size, comparison speed, and response time of the system can be very important factors depending on the use case of the system. Apart from speedup of the algorithms, there are ways to accelerate the system by means of filtering and binning. Furthermore, in case of fusion the overall accuracy of the system can outperform the accuracy of each single algorithm in the system. All these aspects have an impact on the complexity and the cost of a large scale system. Moreover, an essential factor for maintenance costs is adherence to standards. More specifically, compliance of system components with standardized BioAPI interfaces [5] is required that will lead to independence of suppliers of sensors and other individual components. Compliance with standards supports operation reliability in cases, where product lines are no longer supported and maintenance is discontinued. For those cases, the storage of biometric references in standardized formats such as the formats developed by the International Organization for Standardization (ISO) is essential. The reader will find more details on this in the standards ISO/IEC 19794 [3] and [4]. A best practice example of a large scale system being compliant to standards is the Indian e-ID system that was established in 2010. This system relies not only on standardized sensor component but also on standardized interfaces to three independently biometric engines operated in the background.

**Acknowledgements** The authors thank Brian K. Martin, Joseph Atick, Stan Li and Anil Jain for their feedback on this chapter and S. Mracek for contributing the Figs. 25.5 and 25.6.

## References

1. Grother, P., Quinn, G.W., Phillips, P.J.: Multiple-biometric evaluation (mbe)—report on the evaluation of 2d still-image face recognition algorithms. NIST Interagency Report 7709, National Institute of Standards and Technology, June 2010
2. ISO/IEC JTC1 SC37 Biometrics. International Standards ISO/IEC TR 24722, Multimodal and Other Multibiometric Fusion. International Organization for Standardization (2007)
3. ISO/IEC JTC1 SC37 Biometrics. ISO/IEC 19794-1:2011 Information Technology—Biometric Data Interchange Formats—Part 1: Framework. International Organization for Standardization (2011)
4. ISO/IEC JTC1 SC37 Biometrics. ISO/IEC 19794-5:2011. Information Technology—Biometric Data Interchange Formats—Part 5: Face Image Data. International Organization for Standardization (2011)
5. ISO/IEC TC JTC1 SC37 Biometrics. ISO/IEC 19784-1:2006. Information Technology—Biometric Application Programming Interface—Part 1: BioAPI Specification. International Organization for Standardization and International Electrotechnical Committee (2006)
6. ISO/IEC TC JTC1 SC37 Biometrics. ISO/IEC 19795-1:2006. Information Technology—Biometric Performance Testing and Reporting—Part 1: Principles and Framework. International Organization for Standardization and International Electrotechnical Committee, March 2006
7. Mracek, S.: Biometric recognition of 3d faces. Master thesis, Brno University of Technology, May 2010



# Chapter 26

## Face Recognition in Forensic Science

Nicole A. Spaun

### 26.1 Introduction

The use of facial recognition in the field of forensic science presents a challenging set of issues. Forensic science is the use of scientific principles and methods to answer questions of interest to a legal system. Forensic science differs from the field of security; in security applications the goal is to prevent incidents from occurring, while in forensic cases typically an incident has already occurred.

Unlike security or portal scenarios where the administrators have control over the scene and the setup of cameras, in forensics the evidence and surveillance generated is completely uncontrolled by the user of the facial recognition system. Unconstrained lighting conditions, face orientation, and other factors all make the deployment of face recognition systems for surveillance a difficult task [7]. For example, surveillance cameras in places of business are generally pointed at specific locations to spot theft by criminals or employees. These locations include entry/exit doors where the opening of the door may allow the contrast of the camera to be overwhelmed or above an employee's head, where the angle will be steep and the camera is more likely to observe the top of the subject's head than the front of their face. Such conditions lead to the inability to enroll facial images or the worsening of system accuracy rates. Low system accuracy can be disastrous in legal matters. Thus, many forensic organizations have yet to embrace facial recognition as fully as users in the field of security.

---

N.A. Spaun (✉)

Forensic Audio, Video and Image Analysis Unit, Federal Bureau of Investigation, Quantico, VA, USA

e-mail: [Nicole.Spaun@us.army.mil](mailto:Nicole.Spaun@us.army.mil)

*Present address:*

N.A. Spaun

United States Army Europe Headquarters, Heidelberg, Germany

N.A. Spaun

USAREUR, CMR 420, Box 2872, APO AE 09036, USA

In this chapter, we will first explain the current means of comparing faces used by forensic science laboratories. It is a nonautomated process performed by forensic examiners and has been referred to as facial “photographic comparison” [15] or forensic facial identification. Next, we will outline the innovative ways in which facial recognition systems are being used by the forensic community. Lastly, we will discuss the growing future of facial biometrics in the legal system and the increasing (not decreasing) need for human examiners to perform facial identification in combination with the automated facial recognition systems.

## 26.2 Characteristics of Forensic Facial Recognition

Forensic facial recognition and facial identification are distinct, separate processes. In the past facial recognition in the forensic context referred to the process of using eye-witnesses to identify a suspect from either a physical or photo line-up. In today’s terminology, facial recognition is the use of an automated system to determine matches to a probe image from a gallery of images, a one-to-many search, or to verify the identity of an individual, a one-to-one check. The one-to-many process can propose suspects to be investigated or can generate candidate lists to be shown to eyewitnesses. This differs from forensic facial identification, which is a manual process where an expert performs a photographic comparison focusing on the face of an individual. According to the Scientific Working Group on Imaging Technology, photographic comparison is an assessment of the correspondence between features in images and/or known objects for the purpose of rendering an expert opinion regarding identification or elimination [15]. Within these manual comparisons, the features of the head will be analyzed and compared both morphologically and spatially. We will focus on forensic facial identification in this section and the use of automated facial recognition in the following section.

Forensic photographic comparison has a long history; documentation shows it has been in use within the US legal system since at least 1970 [4]. Specialized photographic comparisons have assisted different forensic fields: fingerprint comparisons, tire tread and tool mark analysis, footwear impressions, ballistics, etc.

The specific analysis of faces in images has been performed by forensic examiners with various backgrounds, such as image analysts, photographers, forensic artists, and forensic anthropologists. This diversity of backgrounds is due to the nature of facial identification, where one is assessing a highly three-dimensional aging anatomical feature, a face, in an image that is generally subject to varying photographic conditions including lighting and angles of view. While the backgrounds of persons performing facial identification may vary, the common approach is the application of scientific principles in the course of a visual comparison.

One way of articulating the scientific method for use in photographic comparisons was derived by R.A. Huber for questioned documents [8] and later referred to as “ACE-V”: Analyze, Compare, Evaluate and Verify. When assessing a face, a forensic examiner can use this method to rigorously document facial characteristics, compare them, and form an opinion that can be verified by a similarly trained examiner.

The goal of facial identification is to determine if the questioned individual is the same individual as the known to the exclusion of all others. If so, this is called an Individualization or Identification in the forensics community. Within the biometrics community such a one-to-one comparison is called a “verification”; however in forensic science the term verification generally specifically refers to a peer review or other post-examination evaluation process. Because there are differences in terminology between the forensic and biometric communities, when using a word that has different meanings we will introduce both terms and then continue with the forensic usage for consistency. In contrast to an identification, the examination may lead to the conclusion of Exclusion or Elimination of the known individual as being the questioned individual. If an individualization or elimination cannot be made, it is reported that no definitive conclusion is possible, while listing noted similarities and/or dissimilarities.

A typical facial comparison begins with at least two images depicting individuals for identification. In forensic science, the subject of interest is commonly referred to as the “questioned individual”; in biometrics the image of the subject to be analyzed is called the “probe”. Likewise, in forensic science the suspect depicted in the image is generally called the “known individual”; in biometrics the images of the suspects or potential matches are referred to as the ‘gallery’. It is common for known images in forensic cases to be controlled images, such as those from a driver’s license, passport, previous arrest photograph (a.k.a. mugshot), or other official sources. Most questioned images are typically uncontrolled, obtained from surveillance images or video. The difficulty of the comparison is compounded when both the questioned and known images are uncontrolled. As an example, the case of “the ruthless care giver”, a subject accused of felony embezzlement who later pled guilty, features uncontrolled known and questioned images.

Figure 26.1 depicts four known images submitted to the Federal Bureau of Investigation (FBI) for a facial identification examination. These personal photographs demonstrate several challenging elements for facial identification and recognition: the position (tilt, rotation, and pitch) of the head differs in each image, background/scene is busy and differs in each image, facial expressions differ, illumination varies, image resolution is low, and the face is obstructed by eyewear and other objects. It is worth noting, however, that the ability to resolve details on these images is considerably better than for typical surveillance video obtained in most cases. The submitted questioned images of a suspect were also highly uncontrolled and arrived as both photographs and digital video; the three best images were selected for further examination (see Fig. 26.2). There is at least 10 years in time between when the pictures were taken, possibly more. Matching uncontrolled images such as these is challenging for facial identification but virtually impossible with current facial recognition technology; this highlights the importance of having human examiners involved in the process of forensic facial comparison.

In the analysis stage of the examination, the morphology and texture of the face is reviewed. One observes and notes the characteristics of the questioned face and then for the known face(s). The traits that are used within facial comparisons, like most other forensic examinations, fall into two categories: class and individual characteristics. Class characteristics are those that place an individual within a class or



Known Image 1 - K1

K2

K3

K4

**Fig. 26.1** Four known images of the subject. Images were submitted as printed photographs. Notice the variation in perspective, lighting, expression, pose, and background scene of these uncontrolled images



Questioned Image 1 - Q1

Q2

Q3

**Fig. 26.2** Three questioned images of the suspect taken under uncontrolled conditions. Image Q1 is derived from digital video; Images Q2 and Q3 were submitted as printed photographs

group [17]. These general characteristics include hair color, overall facial shape, presence of facial hair, shape of the nose, presence of freckles, etc. Individual characteristics are those that are unique to the individual and/or allow for a person to be individualized [17]. These specific characteristics include number and location of facial minutiae, such as moles and blemishes, as well as scars, tattoos, chipped teeth, lip creases, wrinkles, etc. The mere presence of freckles is a class characteristic whereas, if the image is detailed enough for one to observe them, the specific number, pattern, and relative location of freckles can be an individualizing characteristic. Many individuals develop wrinkles around the eyes therefore the presence of crow's feet would be a class characteristic, however matching patterns, lengths, and locations of individual wrinkles may be unique. In order to bring out such details within the questioned and known images, the examiner may deem it beneficial to enhance all or part of the image. For example, simple contrast adjustments may bring out details in skin texture such as freckles, blemishes, and scars.

The procedure of the comparison can be qualitative or quantitative, using relative or absolute dimensions. In a morphological comparison, the location and size of facial features is measured relatively, not absolutely. If the perspectives of the questioned and known images are similar and the position of the head is similar, the image depicting the known individual can be scaled to that of the questioned individual by using the interpupillary distance or other consistent features within the image.

An overlay of the scaled known image and the questioned image can then be made in order to determine if the relative alignment of other facial features is consistent. This overlay of images is also referred to as the superimposition method and can be performed with video editing or image processing equipment [19]. A variation of the overlay approach is a photogrammetric one: a side-by-side of the images is prepared and two sets of 3 or more parallel lines are drawn through facial features, such as the jawline, pupils, nasal bridge, on both images and compared by position [4].

For both a photogrammetric and overlay approach, the images must be of the same perspective, but the key difference is that an overlay allows one to view the length and width simultaneously although viewing the lines in the photogrammetric approach leaves more to human perception as one looks across both images. Superimposition can appear to be doctoring the evidence if not properly explained because scaling implies changing the images to effect alignment, but the method is sound. Consider that if you scale an image of Abraham Lincoln to the same eye corner-to-corner distance as that of George Washington, that scaling will not force the length of the face or shape of the jaw to match up, and rightly so because they are different individuals. Just as scaling two images of Abraham Lincoln to the same interpupillary distance will demonstrate the similar locations of facial marks and the consistent sizes of facial features because the images do depict the same individual. Therefore a superimposition can provide extremely beneficial information to determine if features appear to be the same and if the relative locations and dimensions relate.

With an overlay, the examiner can “blink” back and forth between questioned and known imagery to assist in the comparison by identifying similarities and dissimilarities. In this type of comparison, facial landmarks, standard reference marks generally defined by the underlying structure of the skull [6, 10], are used in the main as guides and are not typically measured.

## 26.3 Anthropometric Method

On the contrary, the anthropometric method of facial comparison relies on measurements between facial landmarks [10, 19]. The challenge with the anthropometric method, and others that are absolute measurement driven, is two-fold: they are severely affected by image perspective and are therefore fairly inaccurate in surveillance situations, where the position of the head and camera-to-subject distances are



Known Image 4 - K4

Questioned Image 1 - Q1

**Fig. 26.3** To make this side-by-side chart, the K4 image was scaled by the distance from the ear to the nose of the Q1 image. Of all the submitted images, these two were most similar in pose and perspective, although there is clearly a difference in the rotation of the head. An overlay of the images was attempted but proved to be unhelpful

uncontrolled [13], and also it is difficult to consistently locate the landmarks in different images by different examiners [5]. Therefore, the FBI's forensic examiners do not use an anthropometric method for facial identification and instead use a fusion of the morphological and superimposition technique. Figure 26.3 depicts two images, one questioned and one known, that are of a similar perspective. However, in this case, the images were scaled by the nose to ear distance to assist in preparing a visual aid, a side-by-side chart, instead of a direct overlay as the superimposition would be affected by the difference in position of the head. Figure 26.4 depicts additional side-by-side charts that display greater variation in perspective.

Within the comparison examination, the number and significance of corresponding features must be evaluated [17]. If there are dissimilarities, the examiner works to understand the nature of the dissimilarity. The mere presence of a dissimilarity is not necessarily a cause for exclusion as many dissimilarities can be readily explained by differences in pose, illumination, expression, and time. It is also important to note the significance of any dissimilarity. For example, a large variation in ear length between individuals in questioned and known images is more significant than a possible difference in skin-tone as the latter can potentially be explained by make-up, sun exposure, or differences in photographic conditions.

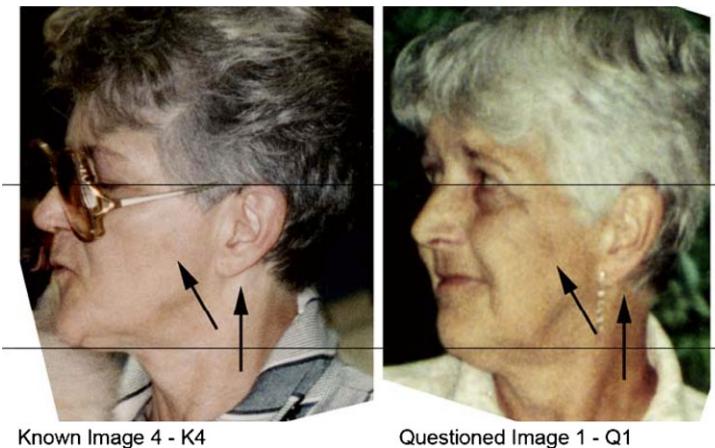
Another consideration when weighing the significance of dissimilarities is to consider the effects of time: wrinkles, transience of facial markings, and changes in weight. If a dissimilarity can be logically explained, such as the disappearance of a pimple or increase in eye wrinkles, then it can be weighted accordingly as a less substantial difference. In our example shown, similarities are noted in overall class characteristics to include shape of the head and nose and the presence and location



**Fig. 26.4** Two side-by-side charts depicting image K4 for comparison to images Q2 and Q3. No superimposition of the images was attempted due to the significant differences in rotation and tilt of the head between images

of wrinkles. Figure 26.5 is the chart from Fig. 26.3 with arrows added to identify specific individualizing features: a blemish on the left cheek and the ear pattern.

One distinct advantage that humans have over today's automated facial recognition programs is that we regard the ear as part of the face and can use it in our analysis. Automated systems for ears are being developed, generally separately from facial recognition programs, and the combination of face and ear analysis is considered multi-modal biometrics. The ear is important because researchers have noted that they have not found any ears that are alike in all parts and that the ear is also stable throughout adulthood [9, 18]. The ear itself contains approximately 16 different features that can be assessed and compared. Figure 26.6 focuses on the left ears depicted in the questioned and known images. The similarity in ear features is



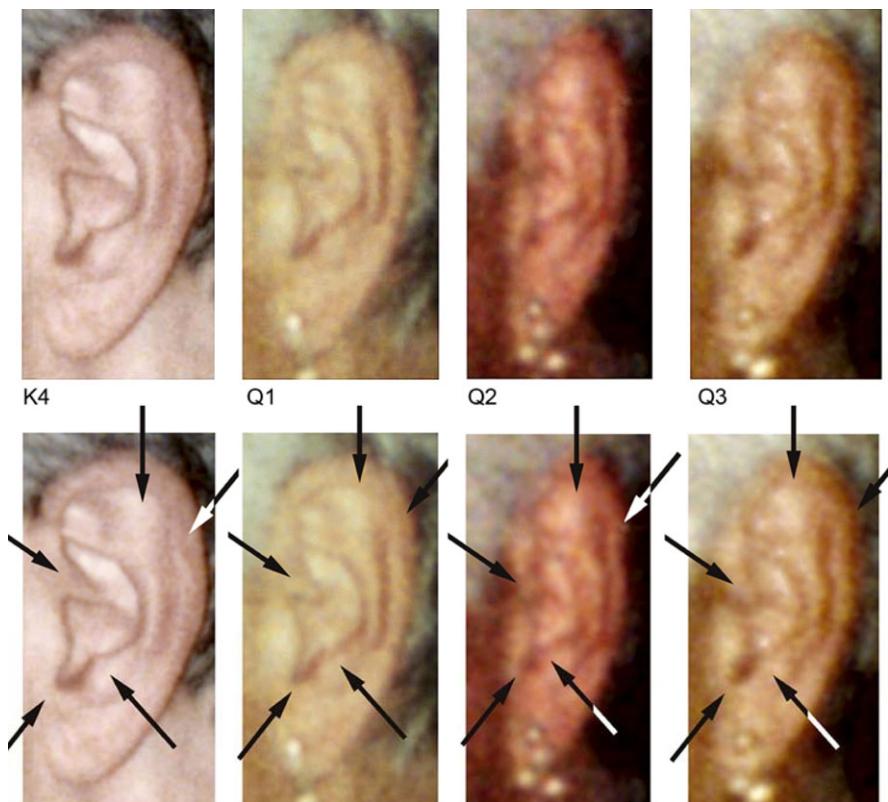
**Fig. 26.5** Arrows have been added to the K4 and Q1 images depicted in Fig. 26.3 to indicate the most individualizing similar characteristics: a blemish on the left cheek and the pattern of the left ear

striking, to include the pattern of the crux of the helix and triangular fossa and the projection of the tragus and anti-tragus.

Based on the number and significance of the similarities in the case depicted in the figures, the known individual was identified as the questioned individual by the FBI examiner. The result was peer-reviewed by a similarly trained FBI examiner as a verification of the case results. If the examiners had a difference of opinion, the results would be discussed and then arbitrated if further disagreement existed. In this instance, the examiners readily reached the same conclusion. A similar analysis by the Oakland County Sheriff's Office in state of Michigan [1] also reached the same conclusion. The suspect was arrested in 2005 and pled guilty to 6 out of 10 charges. The facial identification examinations in this case were a critical component of the forensic investigation.

## 26.4 Use of Facial Recognition in Forensics

Facial recognition technology is being embraced by, and for, law enforcement world-wide. It is being used in novel ways and is pushing the limits of the technology. It is those limits of the technology, mainly the accuracy rates, which are holding back the usage of facial recognition in the legal system. This section will explore several uses of automated facial recognition systems in forensics.



**Fig. 26.6** *Top.* Enlargements of the left ear depicted in the fourth known image, K4, and the three questioned images, Q1–Q3. *Bottom.* Arrows added to indicate similarities in the pattern of the crux of the helix, Darwin's point, triangular fossa into the antihelix, tragus, and intertragus notch

#### 26.4.1 Department of Motor Vehicles

When most people think of their state's department of motor vehicles (DMV) they think of long lines and unflattering photographs. However, when it comes to facial recognition, those photographs are showing their worth. Facial biometrics is a natural fit in a DMV environment because the photographs are controlled, taken under consistent circumstances. Many states are now on the forefront of fraud and identity-theft detection by using automated facial recognition systems. West Virginia was the first state to use an automated facial recognition system in the late 1990s; by 2009, more than 30 states were using facial recognition systems in their driver's licensing procedures.

Most DMV fraud occurs when people use different spellings of their names, use aliases, and show false documents. Previously these cases of fraud would have involved numerous hours of investigation by state DMVs and police departments.

The investigation time has been decreased by facial image screening measures that start when a person tries to apply for a license.

Presently in many states in the US, an individual's photograph is taken for the driver's license and later fed into a facial recognition system. If there is supposed to be an existing license photo on file and the submitted photograph is not matched to it that could be a sign of identity theft where either the first or second photograph is of an imposter. Likewise, if there is a facial match found, then the biographical information is checked to determine whether this is the same individual applying for a license renewal or if there is deception in the biographical information and the person is seeking to obtain an illegal license bearing fake information or a stolen identity.

As an example, the North Carolina Division of Motor Vehicles has used facial recognition in their licensing procedures since 2005. Their Viisage (now part of L-1 Identity Solutions) system has successfully detected fraud, such as one man with nine license pictures and social security numbers, a woman with 11 different identities, and a 15-year old who had posed twice to get fake IDs to get into bars [3]. Additionally, the successes from the DMV's use of facial recognition have been publicized, providing a deterrent to other individuals contemplating fraud. The use of biometric systems has changed the way DMVs operate. In Indiana and several other states, individuals are now asked to assume a neutral expression, instead of a smile, to better control variability between images and enhance the matching accuracy of their automated systems [3]; the state of Virginia even adopted such a policy in 2009 in anticipation of having a facial recognition system in use in the future. The Oregon, Nevada, Wisconsin and other DMVs no longer issue same day license cards in order to allow their investigators time to review the biometric data before sending the license to the customer.

In a revolutionary shift, a new contract for the North Carolina DMV with the MorphoTrak-Safran Group (formerly Sagem Morpho) outlines plans to begin capturing facial images in 3-D for better facial matching. Because a driving license agency generally has the largest repository of images of state residents, DMV facial databases have always been an asset to police looking for photos of their suspects. Now a DMV's biometric system is also proving to be a boon. Nevada police were able to arrest a fugitive with an outstanding felony warrant for sexual assault after his identity fraud was detected with their facial recognition system. In North Carolina, the DMV has welcomed the FBI into their facilities to use their automated system to check for the FBI's wanted individuals. The NC DMV has generated successes for both the North Carolina law enforcement and the FBI, demonstrating the benefit of cross-agency cooperation in forensic facial recognition.

### ***26.4.2 Police Agencies***

The archetypal application of automated facial recognition in forensics is to enable law enforcement to take images obtained from surveillance or Closed Circuit

Television (CCTV) and query a database as a means of identifying the subjects depicted [20]. Using automated biometrics for the development of suspects is nothing new to forensics: fingerprint and DNA results are frequently used to narrow down a suspect pool or identify unknown subjects. However the unconstrained lighting, perspective of the face, obstructions to the face (e.g., hats, glasses), generally poor resolution, and other factors affecting common CCTV imagery severely affects current facial recognition technology. This leads to accuracy rates well below 90%; while that may sound fair for a small database, most law enforcement agencies will be sieving databases of at least several hundred thousand people, if not millions.

The time spent reviewing the incorrectly selected images can be a drain on police resources and the false reject rate could be an even larger concern. Yet the use of facial biometrics to develop a list of leads for police to then investigate is a more manageable task for today's technology. Many smaller police agencies are taking up the challenge and pioneering facial recognition systems for development of suspects in their investigations. We will next outline police use cases by exploring as examples of two police agencies that have been using facial recognition biometrics for several years and a third who are rolling out a ground-breaking new system.

In Washington State, the Pierce County Sheriff's Department has been using automated facial recognition since 2008. The county, which includes the city of Tacoma, was part of a pilot study by Sagem Morpho Inc (now known as Morpho) to use their MorphoFace Investigate (MFI) software in a forensic setting. Crimes like Automated Teller Machine (ATM) robbery, or identity theft using an ATM, have always been a challenge for biometrics because of the sheer volume of innocent fingerprints on the machine. Within the first six months of using facial biometric technology, Pierce County had already matched photos from ATM machines that were robbed with a suspect who was in their database; the suspect was charged with 11 crimes and pleaded guilty. Their database is primarily made up of booking photographs from people previously entered into the county's penal system. During the booking process, the identity of people was previously validated solely by their fingerprints. The photographs taken are now fed into the MFI software for a secondary validation; repeat offenders offer a chance to test the system by matching the new booking photograph to their previous arrest photos. The output of the system is reviewed by a human examiner, who makes the final determination. As a forensic service, officials began using the software to revisit cold cases by selecting good surveillance images and comparing them against the database. This technique is now also in use for active cases when suitable images are available.

Besides using forensic facial recognition at the station, it can be used by law enforcement in the field as well. In Florida, the Pinellas County Sheriff's Office pioneered a system that deputies can operate from their patrol cars. In place since 2002, the facial recognition system developed by Viisage (now L-1 Identity Solutions) has lead to over 500 arrests and assisted in identifying countless other individuals. At the station, the system resembles that used by Pierce County: booking photographs are taken and submitted to the system to verify the identity of the arrestee. Later, if the person is not charged with a crime, their photograph is expunged from the system. The database contains over 8.3 million arrest photos, including records from

several other sheriff's departments throughout Florida, the state Department of Corrections, and access to driver's license photographs from several Florida counties. It is not uncommon for other law enforcement agencies to send their own questioned images to the Pinellas County Sheriff's Office for facial recognition searches.

The novel aspect of the Pinellas County Sheriff's Office system is its mobile application. Installed in more than 170 patrol cars, the mobile system includes a standard digital camera. When a deputy encounters an individual who either does not have a driver's license or ID, or the deputy has reason to doubt the validity of the document, the deputy requests to take a photograph of the individual. An obvious advantage of this system is that the public are generally more willing to allow themselves to be photographed than fingerprinted. The image is uploaded into the automated system at the in-car terminal. Within seconds, the software provides a gallery of images and biographical data as potential matches to the law enforcement officer. The deputy then decides if the person resembles a photograph in the gallery. If the match is to the identity verbally provided by the individual, the deputy can now be confident about the identity even without the individual having their driver's license present. If the match is to an individual with an outstanding warrant, the deputy can bring the person to the station for further identity verification by fingerprint.

The success of their program using off-the-shelf cameras has garnered additional attention and funding from agencies to include the Departments of Defense, Justice, and Homeland Security for Pinellas County Sheriff's Office to expand their mobile facial recognition system. Mobile facial recognition applications for law enforcement are increasing in usage as they are ideal for both officers in the field and detectives at the station.

One such break-through system in Massachusetts is using an application for the iPhone. The police in the city of Brockton are using a system developed by BI2 called MORIS: Mobile Offender Recognition and Identification System. While the technology is not new, the application of it over the iPhone is recently developed. The multi-modal wireless application is using face now with iris and fingerprint under development. The iPhone is used to take a photograph of an individual and then wirelessly upload it into a secure network where it is analyzed by facial recognition software. If a match is made, the officer's phone then receives the additional images and biographic information about the individual. The Massachusetts Sheriff's Association has plans to use the technology in thirty-two police departments and sheriff's offices throughout the state. As with the mobile system used in Pinellas County Florida, the speed and ease-of-use of this technology are likely to entice other law enforcement agencies to adopt similar systems.

## 26.5 Future Perspectives

The automation of facial analysis in forensic science is as inevitable as it was for fingerprint analysis. However, while the idea of a lights-out facial recognition system is appealing, the reality is that there are too many external variables, such as lighting

conditions, and internal variables, such as aging, to allow the use of an automated facial recognition system as the final evaluator for identifying faces in the immediate future. Just as latent print examiners use an automated system to develop or narrow a suspect list and then perform the manual analysis to make the final conclusion, humans will need to be in the facial analysis process as well [16]. Therefore, the near future of facial comparisons involves the fusion between automated systems performing facial recognition and humans verifying the results through facial identification.

Studies have shown that the combination of algorithms and humans yields accuracy rates that surpass those for either solely algorithm or human facial evaluations [14]. In addition, most algorithms in use today rely on measurements between facial landmarks and dimensions of facial features; human methods are more textual, focusing on facial minutiae such as blemishes and wrinkles. Thus, using a fusion of human examiners and algorithms provides a more diverse approach overall until a lights out system can be created.

Because the goal of forensic examinations is successful crime-solving and prosecutions, facial identification must maintain standards that continue its acceptance within the judicial system. A hindrance to both forensic facial identification and automated facial recognition is the paucity of robust statistics for the size and spacing of facial features/landmarks and the frequency of occurrence of facial minutiae. Examiners provide opinion conclusions at this time without quantitative support, other than being 100% positive of their conclusion that it is, or is not, the same individual.

According to Evison and Vorder Bruegge [4], what is lacking is a quantitative means of establishing a match between two facial images, and in the event of a match, there is no process by which to estimate the frequency of any given face shape in the general population. A statistical foundation would allow the examiner to give DNA-like results.

A qualitative conclusion could be supported by a statistical deduction, such as only a given percentage of the population has both a certain interpupillary distance and a blemish on their left cheek, therefore limiting the possible number of people that the questioned image may depict. Furthermore, currently a percentage match score presented by automated facial recognition is a factor of the system's algorithm. If it means anything at all, it is a measure of certainty in the match by the system; this is the equivalent of an eyewitness saying they are 75% sure that the image depicts the person they observed at the crime scene. It would be a great benefit in forensic usage if the score presented was rooted in measurable physical characteristics instead, such that a result of 91 would actually mean that the person is within the 91st percentile of the population who have certain facial statistics and therefore only 9% of the population could be the depicted person. While not definitive, it would be a significant improvement over the current result scores presented by automated systems.

Inevitably, as facial recognition algorithms improve, the number of applications that need human evaluation should decrease in the future. By incorporating the methods that human examiners use into algorithms, Jain and Park [11] have shown that algorithms to detect and compare facial minutiae can be used in tandem with

standard facial recognition systems to improve the overall accuracy rates of the automated system. In other approaches, system designers are experimenting with 3-D facial recognition to improve accuracy. The critical difference between 3-D facial technology in security and forensic applications is that one can obtain both questioned and known imagery in 3-D for access control or document checks but in a forensic instance there is little chance of capturing 3-D questioned images. Therefore in a forensic capacity, the possibility of generating 3-D models from 2-D images is more promising. Multiple known images of varying perspective are taken and can be fused to produce a 3-D model of a face that can be positioned to match the perspective of the face in an uncontrolled questioned image, such as a frame of CCTV video [2, 12].

Facial recognition algorithms combined with face finding software also provides a powerful tool for future forensics. The push for advanced software that can find, track, and extract “faces in the wild” (e.g., from video or the Internet) comes from the commercial sector as much as the government sector. For example, the Picasa photo-album program by search engine giant Google includes facial recognition technology to sort and label personal photos by the faces they contain. Similarly, law enforcement agencies are interested in using the same type of programs in an array of applications.

The FBI is interested in locating faces of potential suspects during computer forensic examinations of seized computers and mobile phones; such technology will greatly benefit gang related and organized crime or terrorism cases where the network of individuals is as important as the initial subject. The National Institute of Justice is assisting the National Center for Missing and Exploited Children (NCMEC) in using face finding software to search the Internet for missing or abducted children in a program similar to the ChildBase system used by the National Criminal Intelligence Service in the United Kingdom to identify child pornography on the Internet. In London, where there is one CCTV camera per 14 people, the Metropolitan Police Service is looking to use such programs to sort through the ubiquitous video to track criminals in both post-event, forensic situations and real-time scenarios.

There is also interest in using facial recognition technology to update more traditional police procedures. Rather than manually developing photo arrays to present to eye witnesses for review, police could use forensic facial recognition software to create arrays of individuals who are more similar in appearance to reduce bias and increase accuracy. Facial recognition technology is already being used to improve the performance of facial composite software used by forensic artists to develop images from eyewitness accounts, yet future uses could entail using automated facial one-to-many matching of sketches and composites against law enforcement databases to develop suspects [21].

## 26.6 Conclusions

It is clear that forensic science will benefit from improvements in facial recognition technology and increased usage thereof. Of course, the ultimate goal for facial

recognition in forensic science is for the advances it will bring in policing and security to lead to less forensics being needed due to fewer criminals in our neighborhoods. Until then, the combination of automated facial recognition to develop leads and forensic examiners performing facial identification will be a great step up from existing purely manual processes.

## References

1. Bailey, B.A.M.: When to call a forensic artist. Evidence Technology Magazine Online. Technical note (2009). <http://www.evidencemagazine.com/index.php/?option=comcontent&task=view&id=164>
2. Bowyer, K.W., Chang, K., Flynn, P.: A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition. *Comput. Vis. Image Underst.* **101**(1), 1–15 (2006)
3. Browder, C.: DMV technology cracks down on fraud. WRAL.com. News article (2008). <http://www.wral.com/news/local/story/4158631/>
4. Evison, M., Vorder Bruegge, R.W.: The Magna Database: A database of three-dimensional facial images for research in human identification and recognition. *Forensic Sci. Commun.* **10**(2) (2008)
5. Evison, M.P., Vorder Bruegge, R.W.: Computer-Aided Forensic Facial Comparison. CRC Press, Boca Raton (2010)
6. Farkas, L.G., Munro, I.R. (eds.): Anthropometric Facial Proportions in Medicine. Charles C Thomas, Springfield (1987)
7. Huang, T., Xiong, Z., Zhang, Z.: Face recognition applications. In: Li, S.Z., Jain, A.K. (eds.) *Handbook of Face Recognition*. Springer, New York (2005)
8. Huber, R.A.: Expert witnesses. *Crim. Law Q.* **2**, 276–296 (1959)
9. Iannarelli, A.V.: The Iannarelli System of Ear Identification. Foundation Press, Brooklyn (1964)
10. Iscan, M.Y., Helmer, R.P. (eds.): *Forensic Analysis of the Skull*. Wiley-Liss, New York (1993)
11. Jain, A.K., Park, U.: Facial marks: Soft biometric for face recognition. In: *Proceedings IEEE International Conference on Image Processing*, pp. 37–40 (2009)
12. Jingu, H., Savvides, M.: In between 3D active appearance models and 3D morphable models. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshop*, pp. 20–26 (2009)
13. Kleinberg, K., Pharm, B., Vanezis, P., Burton, A.M.: Failure of anthropometry as a facial identification technique using high-quality photographs. *J. Forensic Sci.* **4**, 779–783 (2007)
14. O'Toole, A.J., Abdi, H., Jiang, F., Phillips, P.J.: Fusing face-verification algorithms and humans. *IEEE Trans. Syst. Man Cybern. B* **37**(5) (2007)
15. Scientific Working Group on Imaging Technologies, Best practices for forensic image analysis. *Forensic Sci. Commun.* (2005). <http://www.fbi.gov/hq/lab/fsc/backissu/oct2005/standards/200510standards01.htm>
16. Spaun, N.A.: Facial comparison by subject matter experts: Their role in biometrics and their training. In: Tistarelli, M., Nixon, M.S. (eds.) *Advances in Biometrics*. LNCS, vol. 5558, pp. 161–168. Springer, Berlin (2009)
17. Tuthill, H., George, G.: Individualization: Principles and Procedures in Criminalistics, 2nd edn. Lightning Powder, Jacksonville (2002)
18. van der Lugt, C.: Earprint Identification. Elsevier, Amsterdam (2001)
19. Vanezis, P., Brierley, C.: Facial image comparison of crime suspects using video superimposition. *Sci. Justice* **36**, 27–34 (1996)

20. Vorder Bruegge, R.W., Musheno, T.: Some cautions regarding the application of biometric analysis and computer-aided facial recognition in law enforcement. In: Proceedings of the American Defense Preparedness Association's 12th Annual Joint Government-Industry Security Technology Symposium and Exhibition, p. 8 (1996)
21. Zhang, Y., McCullough, C., Sullins, J.R., Ross, C.R.: Human and computer evaluations of face sketches with implications for forensic investigations. In: Proceedings of 2nd International Conference on Biometrics: Theory, Applications, and Systems (BTAS), pp. 475–485 (2008)

# Chapter 27

## Privacy Protection and Face Recognition

Andrew W. Senior and Sharathchandra Pankanti

### 27.1 Introduction

Digital imagery—from personal cameras, cellphones, surveillance cameras and television—is now ubiquitous, being used by governments, corporations and individuals to bring undreamed-of new capabilities for entertainment and government or commercial services. This pervasiveness, together with the new technologies for analyzing and exploiting such images, lead us to ask what are the risks to privacy thus created or exacerbated, and what protections are, or could be put, in place to protect individuals’ privacy. Visual privacy has been of concern since the invention of photography, but the issues are becoming critical as digital imagery is used more widely. At the same time, image processing, computer vision and cryptography techniques are, for the first time, able to deliver technological solutions to some visual privacy problems.

In this chapter, we describe the privacy issues surrounding the proliferation of digital imagery, particularly of faces, in surveillance video, online photo-sharing, medical records and online navigable street imagery. We highlight the growing capacity for computer systems to process, recognize and index face images and outline some of the techniques that have been used to protect privacy while supporting ongoing innovation and growth in the applications of digital imagery.

We first examine what is meant by privacy, and visual privacy in particular, focusing on the privacy concerns surrounding facial images. In Sect. 27.2, we examine some of the factors that determine visual privacy, and in Sect. 27.3 we summarize particular domains in which visual privacy is important. Section 27.4 describes tech-

---

A.W. Senior (✉)  
Google Research, New York, NY 10011, USA  
e-mail: [andrewsenior@google.com](mailto:andrewsenior@google.com)

S. Pankanti  
IBM Research, Yorktown Heights, NY 10598, USA  
e-mail: [sharat@us.ibm.com](mailto:sharat@us.ibm.com)

nologies for protecting privacy in images and Sect. 27.5 presents three systems that have been developed for applying privacy enhancing technologies to face images.

### 27.1.1 What is Privacy?

The problem of protecting privacy is ill-posed in the sense that privacy means different things to different people [1], and attitudes to its protection vary from the belief that this is a right and obligation, to an assumption that anyone demanding privacy must have something to hide [11]. Just as it is difficult to define privacy, it is difficult to determine when privacy has been intruded upon. Equally, many people are happy to trade in their intangible privacy for only small incentives [53], whereas others guard their privacy jealously, so it is hard to determine the value of privacy and privacy intrusions. There is a continuum of privacy intrusion, and our comfort point on that continuum can easily be displaced, by a small incentive or a bout of media hype. A range of factors come into play and our personal tolerances all vary with those factors, from less-than-flattering photos on the Internet to images that form part of our medical record.

In many applications where there are privacy concerns, it is hard to point to examples where there is material effect on a person “who has done nothing wrong”, yet the feeling of disquiet remains perhaps because everyone has done something “wrong”, whether in a personal or legal sense. The area where the public is perhaps most concerned over privacy is video surveillance, with fears aroused by authoritarian governments, and science fiction like *1984* or *Minority Report*. Few people wish a society where all its laws (speeding, parking, jaywalking...) are enforced absolutely rigidly, never mind arbitrarily, by an omniscient state. There is always the possibility that a government to which we give such powers may begin to move towards authoritarianism and apply them towards ends that we do not endorse.

Danielson [16] views the ethics of video surveillance as “a continuously modifiable practice of social practice and agreement”. What is considered acceptable or intrusive in video privacy is a result of cultural attitudes (Danielson contrasts attitudes in the UK and Canada) but also technological capability. A report of the US General Accounting Office [54] quotes the 10th Circuit Court of Appeals decision to uphold the use of surveillance cameras on a public street without a warrant on grounds that “activity a person knowingly exposes to the public is not a subject of Fourth Amendment protection, and thus, is not constitutionally protected from observation.” However technology (with capabilities such as high zooms, automatic control, relentless monitoring, night vision and long term analysis) enables surveillance systems to record and analyze much more than we might naturally believe we are “exposing to the public”. It has been argued that the “chilling” effect of video surveillance is an infringement of US first amendment rights.

Brin, in “The Transparent Society” [10], argues that at some level privacy cannot be preserved and suggests that in the face of inevitable ubiquitous surveillance, our only choice is whether to leave this surveillance in the hands of the authorities

or democratize access to the surveillance mechanisms and use these same tools to “watch the watchers” and so protect the populace against abuses of the tremendous power that the surveillance apparatus affords.

### ***27.1.2 Visual Privacy vs. General Data Privacy***

In many legal systems, visual privacy falls under the legislation dealing with general data privacy and thence data protection. In the European Union, for instance, this is covered by EU directive 95/46/EC which is enacted by member states in their own legislation and came into force in March 2000. In the United Kingdom, with perhaps the densest video surveillance, the relevant legislation is the 1998 Data Protection Act (DPA) which outlines the principles of data protection, saying that data must be:

- Fairly and lawfully processed.
- Processed for limited purposes.
- Adequate, relevant and not excessive.
- Accurate.
- Not kept longer than necessary.
- Processed in accordance with the data subject’s rights.
- Secure.
- Not transferred to countries without adequate protection.

The act requires all CCTV systems to be registered with the Information Commissioner, extending the 1984 Data Protection act that only required registration of CCTV systems that involved “Automatic Processing” of the data. It further gives specific requirements on proper procedure in a CCTV system in order to protect privacy:

Users of CCTV systems must prevent unauthorized access to CCTV control rooms/areas; all visitors must be authorized and recorded in the visitors log and have signed the confidentiality proforma. Operators/staff must be trained in equipment use and tape management. They should also be fully aware of the Codes of Practice and Procedures for the system. The observation of the data by a third party is to be prevented for example, no unauthorized staff must see the CCTV monitors.

It has been estimated [33] that 80% of CCTV systems in London’s business district are not compliant with the DPA.

The act also guarantees the individual’s right of access to information held about them, which extends to access to CCTV recordings of the individual, with protections on the privacy of other individuals who may have been recorded at the same time.<sup>1</sup>

---

<sup>1</sup>“The DPA supports the right of the individual to a copy of any personal data held about them. Therefore data controllers are obliged to provide a copy of the tape if the individual can prove that they are identifiable on the tape, and they provide enough detail to locate the image (e.g., 1 hour

The European Convention on Human Rights guarantees the individual's right to privacy<sup>2</sup> and further constrains the use of video surveillance, most explicitly constraining its use by public authorities. The Swiss Federal Data Protection Commissioner has published these guidelines: [57]

When private individuals use video cameras, for example to protect individuals or prevent material damage, this is subject to the federal law of 19th June 1992 on data protection (DPL; SR 235.1) when the images filmed show identified or identifiable individuals. This applies irrespective of whether the images are stored or not. The processing of the images—such as acquisition, release, immediate or subsequent viewing or archiving—must comply with the general principles of data protection.

A big difference between ordinary data privacy and image privacy is the amorphous nature of images, and the difficulty in processing them automatically to extract useful information. A video clip can convey negligible amounts of information or may contain very detailed and specific information (about times, a person's appearance, actions). Privacy is hard to define, even for explicit textual information such as name, address and social security number fields in a database, knowledge of which can be used for identity theft, fraud and the mining of copious information about the individual from other databases. It becomes much harder to assess the privacy-intrusion that might result from the unstructured but potentially very rich information that could be harvested from surveillance video. A simple video of a person passing in front of a surveillance camera by itself affords little power over the individual, except in a few rare circumstances (such as proving or invalidating an alibi).

There are already strong restrictions on the use of microphones for surveillance because of the presumption of privacy of conversations, but video has been less restricted because there is an expectation of being observed when entering a public space. The UK DPA exempts from controls data where, "The information contained in the personal data has been made public as a result of steps deliberately taken by the data subject." While the act of walking along, the street could be construed as deliberate steps to make ones visual appearance public, we have seen that the DPA does provide privacy safeguards for CCTV.

Until recently, the unmanageability of images has limited their potential for abuse. Few photographs were online, and those that were only manually labeled, and mostly of celebrities, for whom privacy is handled somewhat differently. It takes time to review surveillance video to find "interesting" excerpts, and the storage requirements have added to privacy reasons to ensure that recordings are retained for only short periods of time. Long term storage, and detailed analysis have been

---

before/after the time they believe they were captured by CCTV, their location and what identifiable features to look for). They must submit an appropriate application to the Data Controller and pay a £10 fee. However, the request can be refused if there are additional data/images on the tape relating to a third party. These additional images must be blurred or pixelated out, if shown to a third party. A good example would be a car accident where one party is attempting to claim against another. The data controller is obliged to say no to a civil request to view the tape, as consideration must be given to the other party. A request by the police is a different matter though."

<sup>2</sup>See <http://www.crimereduction.gov.uk/cctv13.htm>.

reserved for situations with strong economic or forensic motivation. However, the advent of sophisticated computer algorithms to automate the extraction of data from images and video, means that imagery is becoming as easy to mine and interrelate as a queryable, machine-readable database.

## 27.2 Factors in Visual Privacy

The goal of privacy protection is to prevent access to information that intrudes on an individual's privacy, but specifying exactly what information is sensitive is difficult. For the purposes of this chapter, we limit ourselves to considering images of faces, though certainly other visual information (e.g., documents, the presence of an object, other biometric identifiers) can compromise privacy in certain circumstances. Identification of individuals is the major threat to privacy in images, and facial appearance is the most commonly captured and easily recognized biometric in images. In Sect. 27.3, we review specific domains where face images are handled. Here, we consider a number of factors (listed in Table 27.1) that play a role in the privacy-intrusiveness of automatic video surveillance, the most complex of these domains. Most systems must be designed to operate under multiple combinations of these factors, requiring multiple levels of privacy protection.

The location of the camera is certainly a principal factor. In high security surveillance environments, no privacy protection may be necessary, but to some, in the home no level of video obfuscation may be considered acceptable. The person with access to the information also determines the level of privacy-intrusiveness, as shown by [30]. A person from any of the categories of Table 27.1 may be familiar with an individual observed by the system, increasing the risk of information being viewed as sensitive, but an unfamiliar person is still subject to voyeurism and prejudiced treatment. In each category the availability of each type of data must be limited as far as possible, consistent with the person's need to access information. The person seen by the camera also plays a role, being observed with some kind of informed consent (e.g., an employee); with active consent, or denial of such, perhaps expressed through the carrying of a privacy token (Sect. 27.4.4); passively as a member of the public; or indeed as an intruder.

In preventing privacy breaches from a surveillance system, we must review the information that can be leaked, the access points to that information within the system, and the availability to different groups of people. Raw video contains much privacy-intrusive information, but much effort is required to get to that information. A key frame may convey much less information, but if well-chosen presents information succinctly. An index with powerful search facilities can easily direct a user to a particular clip of video. The power to intrude on privacy is greatly enhanced if the system has the capability to identify individuals (Sect. 27.2.1). While in principle, all the information stored in a networked digital system is vulnerable to hacking, such breaches are defended against and their effects minimized, by conventional information and physical security, for instance strict access controls should be in

**Table 27.1** Factors affecting privacy protection in a video surveillance system

| Scenario                        | Observer                                 | Familiarity                               | Role of subject                      |
|---------------------------------|------------------------------------------|-------------------------------------------|--------------------------------------|
| High security                   | Law enforcement                          | Familiar                                  | Member of general public             |
| Low security<br>e.g., workplace | System managers<br>System operators      | Unfamiliar                                | Employee<br>(Non-)consenting subject |
| Public space                    | Authorized accessors<br>Public           |                                           | Wearer of privacy tag<br>Intruder    |
| Private space                   | Hackers<br>Person observed               |                                           |                                      |
| Effort                          | Data type                                | Tools                                     |                                      |
| Passive                         | Raw video/image                          | Summary                                   |                                      |
| Opportunistic                   | Redacted video/image                     | Video review                              |                                      |
| Deliberate                      | Extracted metadata                       | Freeze-frame                              |                                      |
| Sophisticated.                  | Anonymized data<br>Linked to an identity | Search<br>Biometric ID<br>Weak identifier |                                      |

place to limit access to privacy-sensitive information, this information can be always encrypted when stored or transmitted, and there may be audit trails to record who accessed what data under what circumstances.

### 27.2.1 Absolute and Relative Identification

A major distinction that we have drawn for privacy in surveillance systems [51], that significantly correlates with how likely they are to intrude on privacy, is the level of anonymity they afford. We distinguish three types of system: *Anonymous*, *Relative ID*, and *Absolute ID*:

- **Anonymous** A traditional CCTV system without computer augmentation is anonymous—it knows nothing about the individuals who are recorded onto the tape or presented on the monitors. While open to abuse by individuals watching the video, it does not facilitate that abuse in a systematic way.
- **Absolute ID** These systems have some method of identifying the individuals observed (usually face recognition but also such identifiers as a badge swipe correlated with the video) and associating them with a personal record in a database. Such systems require some kind of enrollment process [7] to register the person in the database and link the personal information (such as name, social security number) with the identifying characteristic (face image or badge number), though the enrollment can happen without the knowledge or consent of the subject.

- **Relative ID** These systems can recognize people they have seen before, but have no enrollment step. Such systems can be used to collect statistics about people's comings and goings, but do not know any individual information. A relative ID system may use weaker methods of identification (such as clothing colors) to collect short term statistics as people pass from one camera to another, but be unable to recognize people over periods of time longer than a day, or use face recognition without any external label.

Clearly, anonymity protects the individual's privacy. An absolute ID system might, for instance be made to "Give a report on the movements of Joe Bloggs at the end of each day". A relative ID system with a "strong identifier" can easily be converted retrospectively into an Absolute ID with a manual enrollment, and as the availability of labeled data on the web increases, it is becoming easier to partially automate that enrollment. Extracting Relative or Absolute ID from an Anonymous system would require storing and reprocessing the data.

## 27.3 Explosion of Digital Imagery

In this section, we review some of the domains in which the privacy of face images is important.

### 27.3.1 Video Surveillance

CCTV deployment is undoubtedly expanding rapidly. In 2003, McCahill and Norris [33] estimated that there were more than 4 million CCTV cameras in operation in the UK. At the time, most such CCTV systems were rarely monitored and of poor quality, installed largely as a deterrent. Automatic processing of surveillance video, however, is bringing a new era of CCTV with constant monitoring, recording and indexing of all video signals.

Many groups around the world [6, 8, 26, 29, 34, 55] are developing software tools to automate and facilitate the task of "watching" and understanding surveillance videos. These systems also have the potential for gathering much richer information about the people being observed, as well as beginning to make judgments about their actions and behaviors, as well as aggregating this data across days, or even lifetimes. It is these systems that magnify the potential for video surveillance, taking it from an expensive, labor-intensive operation with patchy coverage and poor recall, to an efficient, automated system that observes everything in front of any of its cameras, and allows all that data to be reviewed instantly and mined in new ways: tracking a particular person throughout the day; showing what happens at a particular time of day over a long period; looking for people or vehicles who return to a location, or reappear at related locations. This brings great power in the prevention and solution of crimes.

Some CCTV systems have already publicly deployed face recognition software which has the potential for identifying, and thus tracking, people as effectively as cars are recognized today (for instance, the London Congestion Charging scheme [13]). Currently, face recognition technology is limited to operating on relatively small databases or under good conditions with compliant subjects [42]. Further algorithms bring the potential to automatically track individuals across multiple cameras, with tireless uninterrupted monitoring, across visible and non-visible wavelengths. Such computer systems may in future be able to process many thousands of video streams—whether from cameras installed for this purpose by a single body, public webcams [56] or preinstalled private CCTV systems [38]—resulting in blanket, *omnivident* surveillance networks.

Yu et al. [63], in work supported by the US Department of Justice, describe one potential future direction for higher-level learning based on face recognition. They show how automatically captured location tracks and face images from fixed and steerable cameras can be used to learn graphs of social networks in groups of people, particularly targeted at identifying gangs and their leaders in prisons.

### 27.3.1.1 Camera-Based Sensors

While surveillance has driven the widespread deployment of cameras, low cost sensors and more sophisticated algorithms are enabling many other applications that involve the installation of cameras that will see people, but in which it is not the images themselves that are of interest, but rather the data extracted from them. These range from today's traffic cameras and cameras that anticipate drownings in swimming pools [44] to “human aware” buildings that adjust heating, lighting [32], elevators and telephones according to the locations and activities of people, as well as controlling physical access and assisting with speech recognition by lip-reading [45]. Many future devices and systems will have cameras installed because they are a low-cost sensor that “sees the world as humans see it”. While the purpose of these sensors is often merely to detect a single piece of information, such as the number of people at a check-out line [60], the same hardware could equally be used for surveillance and face recognition. It is impossible for the subjects of the observation to know what is happening to the data once it has left the sensor, so without suitable oversight these devices are a potential and perceived privacy intrusion.

### 27.3.1.2 Ambient Video Connections

Some of the earliest work on image-based privacy relates to the use of video for ambient awareness in media spaces, particularly video for awareness of co-workers in a remote location. Here, a worker may choose to be shown in a constant video feed to provide a sense of copresence. However, in times when there is no explicit face-to-face conversation the worker may wish to reveal only general information, such as presence or general location without revealing specific details that would be visible in a full-resolution video. Such a privacy protection system that uses model-based face obscuration is described in Sect. 27.5.2.

### 27.3.2 *Medical Images*

Medical images are also proliferating, with the advances in medical science and the lowering cost of imaging devices. Much attention has been paid to the electronic patient record and its privacy implications. The ability to copy and transmit sensitive patient records in electronic form as well as access them remotely, together with the increasing richness of the records has led to stricter controls on medical record privacy, such as the HIPAA [58] regulations in the USA. These regulate medical records as a whole, but photographs of patients that show their faces are of specific concern here. Face images may be an important component of a patient record for such areas as oral and maxillofacial surgery, dentistry, dermatology and plastic and reconstructive surgery. It is important to protect the patient from exposure of the data both through unauthorized access and use for teaching or research material. It is essential in the latter case to remove identifying information while preserving the usefulness and accuracy for the intended purpose. De-identifying faces (Sect. 27.5.2) is an important technique here.

### 27.3.3 *Online Photograph Sharing*

Photo-sharing has recently become one of the most popular activities on the Internet, featuring in social networking sites like Facebook, and special photo storage and sharing sites like Flickr or photobucket.com. Billions of photographs are stored by such services.<sup>3</sup> As traffic has grown, the affordances for labeling have become more sophisticated. Text tagging has evolved to labeled bounding boxes and to the automatic face recognition found in Picasa and Windows Live Photo Gallery. Now the task of labeling a photo album has been made much easier by software which allows the user to name a person in one picture and then propagate that label to other similar photos of the same person. These new labels can be confirmed or corrected and the face model is improved accordingly, so a large photo collection can be iteratively labeled with relatively little manual intervention.

Companies such as PolarRose are seeking to apply these techniques to social network sites, and companies such as Google have developed face recognition technologies to label photographs and videos [48] of celebrities on the web. As recognition technology improves and the quantity of labeled data increases, it seems that it is only a matter of time before all photos of you on the Internet can be tagged as such and searchable.

Google Goggles which allows visual search using images captured with a smart phone has the potential to carry out face recognition, but privacy concerns have prevented it from being made available [39], according to a spokesman. “We do have the relevant facial recognition technology at our disposal. . . . But we haven’t implemented this on Google Goggles because we want to consider the privacy implications and how this feature might be added responsibly.” [46].

---

<sup>3</sup>More than 3 billion photos a day are uploaded to Facebook [20].

### 27.3.4 Street View

Online services such as Google’s Street View, Bing Streetside, MapJack, and Everscape present systematically captured street-level imagery on an unprecedented scale, allowing users to explore distant places through intuitive user interfaces in their computer browser. The extent of their coverage, the high image quality and the easy access have aroused concern over the effect of the imagery on privacy. Individuals are concerned about the possibility of their presence in a particular location being publicly visible and about their property being easily examinable without their knowledge and scouted by burglars. In Japan, privacy concerns led to Street View imagery being recaptured with the car-mounted cameras lowered by 40 cm so that the service would not present imagery taken over people’s garden walls, and there has been considerable opposition to the service on privacy grounds in Switzerland and Germany [2]. Mechanisms are provided for individuals to request that particular images are not made public, but the ubiquity of faces and license plates in the imagery, and the general unease that these elicit, required an automated solution to attempt to automatically obscure all the faces and license plates. The automatic system that Google deployed to blur faces and license plates is described in Sect. 27.5.1. Flores and Belongie [21] have shown preliminary work using multiple views and inpainting to remove isolated pedestrians images from Street View images.

### 27.3.5 Institutional Databases

Increasingly in recent years, governments and corporations have sought to harness Information Technology to improve efficiency in their provision of services, to prevent fraud and to ensure the security of citizens. Such developments have involved collecting more information and making that information more readily available to searching and through links between databases. Silos of information, collected for an authorized process are readily accepted for the benefits they bring, but the public becomes more uneasy as such databases succumb to “function creep”, being used for purposes not originally intended, especially when several such databases are linked together to enable searches across multiple domains. Plans for Australian identity cards were rejected because of just such fears [17] and there was a significant backlash when retired Admiral John Poindexter conceived the “Total Information Awareness” (TIA) project [43] which aimed to gather and mine large quantities of data, of all kinds, and use these to detect and track criminals and terrorists. The Orwellian potential for such a project raised an outcry that resulted in the project being renamed the *Terrorist Information Awareness* project, an epithet calculated to stifle objection in post-September 11th America.

Naturally, faces are an important part of such electronic databases allowing the verification of identity for such purposes as border control and driver licensing, but registered faces provide a link between definitive, exploitable identification information such as name, address, social security number, bank accounts, immigration

status, criminal record and medical history and the mass of images of individuals that is building up from other channels like surveillance and photo-sharing.<sup>4</sup> Many authors, from Bentham [5] to the present have expressed concern about the potential for state oppression by the exercise of extensive monitoring and the projection that such monitoring is pervasive if unknowable.

The widespread use of electronic records and their portability has led to numerous cases of records being leaked or lost, and their potential value for identity theft has made them a target for theft and hacking, from within as well as outside the controlling institution. This inadvertent exposure is a major reason for strong automatic privacy protection controls such as encryption, tight access control and image redaction even in databases where normal use would not lead to privacy intrusion.

## 27.4 Technology for Enabling Privacy

In recent years, a number of technological solutions have been proposed for the general problem of privacy protection in images and video, and for face privacy protection in particular. In this section, we review the principal methods being developed: intervention, redaction, and provably secret processing, together with a discussion of privacy policies and tokens for claiming or relinquishing privacy protection.

### 27.4.1 Intervention

Patel et al. [41] have proposed a system that prevents unauthorized photography by detecting cameras using their retro-reflective properties. In their detection system, a bright infra-red source is located near a camera. If the lens of another camera is pointed toward the detector, a strong retro-reflection is seen in the image, which can easily be detected automatically. When a camera is detected, a light is flashed towards it using a digital projector, spoiling any images that it may record. This unusual approach, dubbed an “anti-paparazzi” device, exploits computer vision to create a privacy-protection solution where no control can be exerted over the use of the images once recorded. As well as privacy protection, the system is envisaged for copyright protection, for instance to prevent recording of new release films in cinemas.

### 27.4.2 Visual Privacy by Redaction

Most recent work on visual privacy protection has focused on systems that modify, or redact, visual images to mask out privacy sensitive information. Such systems

---

<sup>4</sup>Consider the case of the British fraudster John Darwin who faked his own death but was identified in a photograph on a real estate web site after subsequently buying property [61].

typically use computer vision technology to determine privacy sensitive regions of the image, for instance tracking moving people in a video [51], or detecting faces [22] in still or moving images. Such regions of interest are then changed in some way to prevent subsequent viewers or algorithms from extracting the privacy sensitive information. Obscuration methods that are commonly used include blurring, masking, pixellating [27], scrambling [18], or permuting pixels [12]. Recent work has investigated the limitations of some of these, for instance Gross et al. [25] show that simple pixellation and blurring may not be strong enough to defeat face recognition systems. They train a *parrot* [37] recognizer on gallery images with the same distortion as the probe and obtain markedly higher recognition rates than using a system trained on clean images. Neustaedter et al. [35] have also found global blurring and other obscuration techniques to be unable to supply simultaneously both sufficient privacy and adequate information for always-on home video conferencing. Koshimizu et al. [31] have explored the acceptability of different obscuration and rerendering techniques for video surveillance.

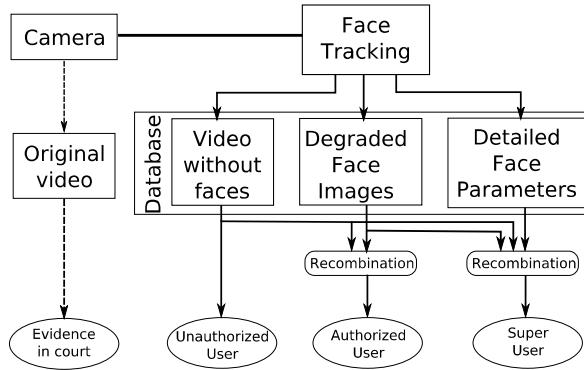
Stronger masking with greater changes to the image may have the limitation of reducing the usability of the video for its intended purpose, but rerendering [51] may alleviate this by showing computer generated images to convey important information hidden by the redaction process. One example of this would be to obscure a person's face in an image with a computer generated face—hiding the identity yet preserving the gaze direction and expression. Two extensions of this using face modeling are described in Sect. 27.5.2.

One important aspect of redaction systems is reversibility. It may be desirable for some purposes to permanently destroy the privacy-intrusive information, but for others it may be desirable or necessary, perhaps for evidential reasons, to be able to reconstruct the original video.

When redacted information is to be presented to the user, one fundamental question is what redaction is necessary and when the redaction is to occur. In some scenarios, the system may need to guarantee that redaction happens at the earliest stage, and that unredacted data is never accessible. For such a scenario, we proposed the PrivacyCam [52], a camera with on-board redaction that behaves as a normal video camera but outputs video with privacy-sensitive areas redacted. Only one level of redaction at a time is possible when such a system is a drop-in replacement for an analogue camera. However for a general system, it may be necessary to present both redacted and unredacted data to end users according to the task and their rights, and to allow different types and extents of redaction according to the circumstances.

In a distributed surveillance system, there are three principal locations through which the data must pass: the video processor, database and browser (or end-user application), at each of which the redaction could take place:

**Browser:** Here the unredacted data is delivered to the client and client software carries out the redaction and presents the redacted information to the user. This scenario means that redacted data does not need to be stored and transmitted but metadata for redaction does need to be transferred with the raw data. Since the browser is the part of a system most exposed to attack, transmitting the unredacted data there is not secure.



**Fig. 27.1** Double redaction: Video is decomposed into information streams which protect privacy and are recombined when needed by a sufficiently authorized user. Information is not duplicated and sensitive information is only transmitted when authorized. Optionally an unmodified (but encrypted) copy of the video may need to be securely stored to meet requirements for evidence

**Content management:** The content management system can redact the information when requested for viewing, which will minimize storage requirements and allow complete flexibility, but involve additional processing (with the same keyframe perhaps being redacted multiple times), latency and imposes image modification requirements on the database system. If the unredacted video is stored, unauthorized access can reveal the privacy-intrusive data.

**Video analytics:** The video analytics system has access to the richest information about the video activity and content, and thus can have the finest control over the redaction, but committing at encoding time allows for no post-hoc flexibility. In the other two scenarios, for instance, a set of people and objects could be chosen and obscured on-the-fly. Sending redacted and raw frames to the database imposes bandwidth and storage requirements.

**Double redaction:** Perhaps the most flexible and secure method is *double redaction* [50], in which privacy protection is applied at the earliest possible stage (ideally at the camera), and privacy-protected data flows through the system by default. Separate encrypted streams containing the private data can be transmitted in parallel to the content management system and to authorized end users, allowing the inversion of the privacy protection in controlled circumstances. The operating point of the detection system can even be changed continuously at display time according to a user's rights, to obscure all possible detections, or only those above a certain confidence. Figure 27.1 shows an example of such a double redaction scheme, with two levels of redaction.

Several authors [28, 40] have adopted such a double redaction system and have explored options for embedding or hiding additional data streams in the redacted video data, for instance Zhang et al. [64] store the information as a watermark, Carillo et al. [12] use a reversible cryptographic pixel permutation process to obscure information in a manner that can be reversed, given the right key, and that is robust to compression and transcoding of video. Li et al. transform sensitive data using the

Discrete Wavelet Transform, preserving only low frequency information and hide the encrypted high-frequency information in the JPEG image.

### 27.4.3 *Cryptographically Secure Processing*

A recent development in visual privacy protection is in the development of cryptographically secure methods for processing images. These methods establish protocols by which two parties can collaborate to process images without risk of privacy intrusion. In particular, if one party owns images and another party has an image processing algorithm, algorithms such as “Blind Vision” [3] allow certain algorithmic operations to be carried out by the second party on the first party’s data without the data itself or the algorithm being made available to the other party. Such systems have been applied to the problems of face detection and recognition, as will be discussed in Sect. 27.5.3.

### 27.4.4 *Privacy Policies and Tokens*

An important aspect of privacy protection systems is the policies for determining what data needs to be obscured for which users. As we have seen in Sect. 27.2, privacy systems may need to operate in different modes according to different factors including the roles, authorization and relationship of the observer and the observed. Determining privacy policies is a complex area, made more so when detection of the privacy sensitive information is not reliable.

Brassil [9] and Wickramasuriya et al. [62] explore the use of devices (detected through separate sensors) that can be used to claim privacy protection in public cameras, and Schiff et al. [49] use visual cues (hats or jackets) to designate individuals whose faces are to be redacted, or preserved from redaction.

## 27.5 Systems for Face Privacy Protection

In this section, we describe three approaches that have been specifically designed for privacy protection of face images.

### 27.5.1 *Google Street View*

As mentioned in Sect. 27.3.4, Google’s Street View and similar sites present particular privacy challenges with their vast coverage of street-level public imagery. Frome et al. [22] describe the system that they developed to address this problem.

They highlight both the scale of the problem and the challenging nature of their “in the wild” data. They use a standard sliding-window face detector that classifies each square region of each image as face or non-face. The detector is applied with two operating points, and the results are combined with a number of other features (including face color and a 3D position estimate) using a neural network to determine a final classification of the region as face or non-face. All face detections are blurred in the final imagery that is served in Google Maps. A similar system is used to detect and blur license plates.

They describe the criteria used for choosing the redaction method, that it should be: (1) irreversible; (2) visually acceptable for both true faces and false positives; (3) makes it clear to the public that redaction has taken place, a requirement that precludes the use of rerendering techniques from the next section. To meet these requirements, the authors choose to redact the faces with Gaussian blurring and the addition of noise.

### 27.5.2 De-identifying Face Images

Coutaz et al. [15], described a system for preserving privacy in the CoMedi media space which gives remote participants a sense of co-presence. The system offered shadowing and resolution lowering redaction methods [27] for privacy protection but also used eigenspace filtering for face redaction. In this technique an eigenface [59] representation is constructed using a training set of face images, and faces detected in the mediaspace video are projected into that eigenspace before rerendering. This effectively constrains the rendered face to conform to variations seen in the training set, and obscures other kinds of appearance differences. While this can protect privacy in some ways, such as hiding socially incorrect gestures or expressions, it is also shown to have limitations. The choice of the correct model and the corresponding training set is crucial. Using a mismatched model may unintentionally change the identity, pose, or expression of the face.

In several papers, Sweeney and collaborators [23–25, 36] have described a series of algorithms for de-identifying faces that extend this eigenface approach, tackling the problem of identity hiding. They use the Active Appearance Model [14], face representation which normalizes for pose and facial expression. Their algorithms create a weighted average of faces from different people, such that the resulting face is not identifiable. This deidentification is termed  $k$ -same in that it results in a face whose chance of being correctly identified is no more than  $\frac{1}{k}$ . In their more recent work [25], they use a multifactor decomposition in their face representations that reduces blending artifacts and allows the facial expression to be preserved while hiding the face identity. They also consider the application of this in a medical video database, showing patients’ responses to pain, in which facial expression, not identity, is important.

### 27.5.3 *Blind Face Recognition*

As described in Sect. 27.4.3, a new field of research is cryptographically provable privacy preserving signal processing, or “Blind vision”. Recent work has applied this to face detection [4] and recognition algorithms. Erkin et al. [19] describe a secure implementation of an eigenface face recognition algorithm [59]. Their system performs the operations of projecting a face image onto the eigenvectors of “face subspace” and calculating the distances to each of the enrolled faces, without the querying party, Alice, having to reveal the query image, nor the owner of the face recognizer, Bob, having to reveal the enrolled faces. Such a secure multiparty computation can be very laborious and time consuming, with a single recognition taking 10–20 s, though speed-ups have been proposed [47].

## 27.6 Delivering Visual Privacy

The technological tools of the previous section can help to prevent privacy intrusion from image-based applications, but as we have seen they form only part of a privacy solution, along with information security and privacy policies. To ensure that the privacy benefits are delivered effectively, two further factors must be considered—ensuring that systems are used where appropriate and operate effectively when installed.

### 27.6.1 *Operating Point*

Video information processing systems are error prone. Perfect performance can not be guaranteed, even under fairly benign operating conditions, and systems make two types of errors when determining image regions for redaction: missed detection (of an event or object) and false alarm (triggering when the event or object is not present). We can trade these errors off against one another, choosing an *operating point* with high sensitivity that has few missed detections, but many false alarms, or one with low sensitivity that has few false alarms, but more often fails to detect real events when they occur.

The problems of imperfect image processing can be minimized by selecting the appropriate system operating point. The *costs* of missed detection and false alarm can be quite different, as seen in Sect. 27.5.1, where not blurring a face reveals private information and blurring a non-face degrades the quality of the information provided. In a surveillance system, the operating point for privacy protection may be chosen differently than for general object detection for indexing. Given the sensitive nature of the information, it is likely that a single missed detection may reveal personal information over extended periods of time. For example, failing to detect, and thus obscure, a face in a single frame of video could allow identity information

to be displayed and thus compromise the anonymity of days of aggregated track information associated with the supposedly anonymous individual. On the other hand, an occasional false alarm and unnecessary redaction may have a limited impact on the effectiveness of the installation. The operating point can be part of the access-control structure—greater authorization allows the reduction of the false alarm rate at a higher risk of compromising privacy. Additional measures such as limiting access to freeze-frame or data export functions can also reduce the risks associated with occasional failures in the system. For some applications it will be some time before algorithms are accurate enough to deliver an operating point that gives useful privacy benefits without degrading the usefulness of the data provided.

Even with perfect detection, anonymity cannot be guaranteed. While face recognition is the most salient identifier in video, a number of other biometrics such as face, gait or ear shape; and weak identifiers (height, pace length, skin color, clothing color) can still be preserved after face redaction. Contextual information alone may be enough to uniquely identify a person even when all identifying characteristics are obscured in the video. Obscuring biometrics and weak identifiers will nevertheless reduce the potential for privacy intrusion. These privacy-protection algorithms, even when operating imperfectly, will serve the purpose of making it harder, if not impossible, to run automatic algorithms to extract privacy-intrusive information, and making abuses by human operators more difficult or costly.

### ***27.6.2 Will Privacy Technology Be Used?***

The techniques described in this chapter could be considered as optional additions to systems that display images—that will cost more and risk impinging on the usefulness of the systems, while the privacy protection benefits may accrue to stakeholders other than the service provider or the primary users. We must then ask why providers of image-based services will choose to bear the extra burden of implementing privacy protection technologies, even when the technologies are fast and accurate enough to be practically deployed. Clearly in many cases companies will choose to implement them as being the “right thing” to do, out of concern for protecting privacy, and for guarding their good name. Others may be pressured by the public, shareholders or customers to apply such technologies, or be asked to do so by privacy ombudsmen. Finally explicit legislation may be implemented to require such technologies, though creating manageable legislation for the nebulous area of privacy is extremely difficult. Existing legislation in some jurisdictions may already require the deployment of these techniques in domains such as surveillance as soon as they become feasible and commercially available.

Even when privacy protection methods are mandated, compliance and enforcement are still open to question, particularly in private systems such as medical images and surveillance. McCahill and Norris [33] estimated that nearly 80% of CCTV systems in London’s business space did not comply with current data protection legislation, which specifies privacy protection controls such as preventing unauthorized

people from viewing CCTV monitors. Legislating public access to surveillance systems as proposed by Brin [10] is one solution, but that still begs the question—are there are additional video feeds that are not available for public scrutiny? A potential solution that we have proposed [52] is certification and registration of systems, along the lines of the TRUSTe system that evolved for Internet privacy. Vendors of video systems might invite certification of their privacy-protection system by some independent body. (In the US, the Federal Trade Commission Act<sup>5</sup> has the power to enforce companies' privacy policies.) For purpose-built devices with a dedicated camera sensor (like PrivacyCam, Sect. 27.4.2), this would suffice. Individual surveillance installations could also be certified for compliance with installation and operating procedures, with a certification of the privacy protection offered by the surveillance site prominently displayed on the equipment and CCTV advisory notices. Such notices might include a site (or even camera) identification number and the URL or SMS number of the surveillance privacy registrar where the site can be looked up to confirm the certification of the surveillance system. Consumer complaints would invoke investigations by the registrar, and conscientious companies could invite voluntary inspections.

## 27.7 Conclusions

As cameras and networking have become cheaper and ubiquitous, there has been an explosion in the dissemination of images, for new and traditional applications from photo-sharing to surveillance and medical imaging. With this explosion, there has been a corresponding increase in the potential for privacy-intrusive uses of those images. Thus far, controls on such privacy intrusions have been very limited. We have examined how images in different domains can contain sensitive information, particularly images of faces that allow individuals to be identified. We have described ways in which that information can be obscured by redaction, based on computer vision techniques to identify regions of interest, and image processing techniques to carry out the redaction in a secure, possibly invertible, manner. Finally, we have described three particular systems that have been used to apply privacy preserving techniques to face images and explored ways in which such privacy protection techniques can be deployed and might become more widespread.

## References

1. Acquisti, A.: Privacy and security of personal information: Economic incentives and technological solutions. In: Camp, J., Lewis, R. (eds.) *The Economics of Information Security*. Kluwer, Dordrecht (2004)
2. Associated Press. Swiss official demands shutdown of Google Street View. *New York Times* (2009)

---

<sup>5</sup><http://www.ftc.gov/privacy>.

3. Avidan, S., Butman, M.: Blind vision. In: European Conference on Computer Vision, vol. 3953, pp. 1–13 (2006)
4. Avidan, S., Butman, M.: Efficient methods for privacy preserving face detection. In: NIPS, pp. 57–64 (2006)
5. Bentham, J.: Panopticon Letters. London (1787). <http://cartome.org/panopticon2.htm>
6. Black, J., Ellis, T.: Multi camera image tracking. In: International Workshop on Performance Evaluation of Tracking and Surveillance (2001)
7. Bolle, R.M., Connell, J.H., Pankanti, S., Ratha, N.K., Senior, A.W.: Guide to Biometrics: Selection and Use. Springer, New York (2003)
8. Boult, T., Micheals, R.J., Gao, X., Eckmann, M.: Into the woods: Visual surveillance of non-cooperative and camouflaged targets in complex outdoor settings. Proc. IEEE **89**(10), 1382–1402 (2001)
9. Brassil, J.: Using mobile communications to assert privacy from video surveillance. In: 19th IEEE International Parallel and Distributed Processing Symposium, p. 290a (2005)
10. Brin, D.: The Transparent Society: Will Technology Force Us to Choose Between Privacy and Freedom. Perseus, Cambridge (1999)
11. Caloyannides, M.: Society cannot function without privacy. IEEE Secur. Priv. Mag., May/June 2003
12. Carrillo, P., Kalva, H., Magliveras, S.: Compression independent reversible encryption for privacy in video surveillance. EURASIP J. Inf. Secur.
13. Congestion charging: Enforcement technology. BBC LDN (2003). <http://www.bbc.co.uk/london/congestion/technology.shtml>
14. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 492–7 (2001)
15. Coutaz, J., Bérard, F., Carraux, E., Astier, W., Crowley, J.L.: CoMedi: Using computer vision to support awareness and privacy in mediaspaces. In: CHI, pp. 13–14. ACM Press, New York (1999)
16. Danielson, P.: Video surveillance for the rest of us: Proliferation, privacy, and ethics education. In: International Symposium on Technology and Society, 6–8 June 2002, pp. 162–167 (2002)
17. Davies, S.: The loose cannon: An overview of campaigns of opposition to national identity card proposals. In: e-ID: Securing the Mobility of Citizens and Commerce in a Greater Europe. Unisys, February 2004
18. Dufaux, F., Ebrahimi, T.: Scrambling for video surveillance with privacy. In: Proceedings of Computer Vision and Pattern Recognition, June 2006, p. 160 (2006)
19. Erkin, Z., Franz, M., Guajardo, J., Katzenbeisser, S., Lagendijk, I., Toft, T.: Privacy-preserving face recognition. In: Privacy Enhancing Technologies Symposium (2009)
20. Facebook press room, 21 April 2010
21. Flores, A., Belongie, S.: Removing pedestrians from Google Street View images. In: International Workshop on Mobile Vision, June 2010
22. Frome, A., Cheung, G., Abdulkader, A., Zennaro, M., Wu, B., Bissacco, A., Adam, H., Neven, H., Vincent, L.: Large-scale privacy protection in Google Street View. In: Proceedings of Computer Vision and Pattern Recognition (2009)
23. Gross, R., Airoldi, E., Malin, B., Sweeney, L.: Integrating utility into face de-identification. In: Workshop on Privacy Enhancing Technologies. CMU (2005)
24. Gross, R., Sweeney, L., de la Torre, F., Baker, S.: Model-based face de-identification. In: Workshop on Privacy Research in Vision. IEEE, New York (2006)
25. Gross, R., Sweeney, L., Cohn, J., de la Torre, F., Baker, S.: Face de-identification. In: Senior, A.W. (ed.) Protecting Privacy in Video Surveillance. Springer, Berlin (2009)
26. Hampapur, A., Brown, L., Connell, J., Ekin, A., Lu, M., Merkl, H., Pankanti, S., Senior, A., Tian, Y.L.: Multi-scale tracking for smart video surveillance. IEEE Trans. Signal Process. (2005)
27. Hudson, S., Smith, I.: Techniques for addressing fundamental privacy and distribution trade-offs in awareness support systems. In: CSCW, pp. 248–257 (1996)
28. Ito, I., Kiya, H.: One-time key based phase scrambling for phase-only correlation between visually protected images. EURASIP J. Inf. Secur.

29. Khan, S., Shah, M.: Tracking people in presence of occlusion. In: Asian Conference on Computer Vision (2000)
30. Koshimizu, T., Toriyama, T., Babaguchi, N.: Factors on the sense of privacy in video surveillance. In: Proceedings of the 3rd ACM Workshop on Continuous Archival and Retrieval of Personal Experiences, pp. 35–44. ACM, New York (2006)
31. Koshimizu, T., Umata, I., Toriyama, T., Babaguchi, N.: Psychological study for designing privacy protected video surveillance system: PriSurv. In: Senior, A.W. (ed.) Protecting Privacy in Video Surveillance. Springer, Berlin (2009)
32. Lipton, A.J., Clark, J.I.W., Thompson, B., Myers, G., Zhang, Z., Titus, S., Venetianer, P.: The intelligent vision sensor: Turning video into information. In: Advanced Video and Signal-based Surveillance. IEEE, New York (2007)
33. McCahill, M., Norris, C.: CCTV. Perpetuity Press, Leicester (2003)
34. McKenna, S., Jabri, J.S., Duran, Z., Wechsler, H.: Tracking interacting people. In: International Conference on Face and Gesture Recognition, March 2000, pp. 348–53 (2000)
35. Neustaedter, C., Greenberg, S., Boyle, M.: Blur filtration fails to preserve privacy for home-based video conferencing. ACM Trans. Comput. Hum. Interact. (2006)
36. Newton, E., Sweeney, L., Malin, B.: Preserving privacy by de-identifying facial images. Technical Report CMU-CS-03-119, Carnegie Mellon University, School of Computer Science, Pittsburgh (2003)
37. Newton, E., Sweeney, L., Malin, B.: Preserving privacy by de-identifying facial images. IEEE Trans. Knowl. Data Eng. 2(17), 232–243 (2005)
38. New York city police department releases draft of public security privacy guidelines for public comment. NYPD Press Release, February 2009
39. Palmer, M.: Google debates face recognition technology. Financial Times, 19 May 2010
40. Paruchuri, J.K., Cheung, S.S., Hail, M.W.: Video data hiding for managing privacy information in surveillance systems. EURASIP J. Inf. Secur.
41. Patel, S.N., Summet, J.W., Truong, K.N.: Blindspot: Creating capture-resistant spaces. In: Senior, A.W. (ed.) Protecting Privacy in Video Surveillance, pp. 185–201. Springer, Berlin (2009)
42. Phillips, P.J., Scruggs, W.T., O’Toole, A.J., Flynn, P.J., Bowyer, K.W., Schott, C.L., Sharpe, M.: FRVT 2006 and ICE 2006 large-scale results. Technical Report NISTIR 7408, NIST, Gaithersburg, MD 20899, March 2006
43. Poindexter, J.: Overview of the information awareness office, August 2002. <http://www.fas.org/irp/agency/dod/poindexter.html>
44. Poseidon. <http://www.poseidon-tech.com/>
45. Potamianos, G., Neti, C., Gravier, G., Garg, A., Senior, A.W.: Recent advances in the automatic recognition of audiovisual speech. Proc. IEEE (2003)
46. Privacy fears force search giant to block facial recognition application on Google goggles. The Daily Mail Online, December 2009
47. Sadeghi, A.R., Schneider, T., Wehrenberg, I.: Efficient privacy-preserving face recognition. In: 12th International Conference on Information Security and Cryptology (2010)
48. Sargin, M.E., Aradhye, H., Moreno, P., Zhao, M.: Audiovisual celebrity recognition in unconstrained web videos. In: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, April 2009
49. Schiff, J., Meingast, M., Mulligan, D.K., Sastry, S., Goldberg, K.: Respectful cameras: Detecting visual markers in real-time to address privacy concerns. In: Senior, A.W. (ed.) Protecting Privacy in Video Surveillance. Springer, Berlin (2009)
50. Senior, A.W.: Privacy protection in a video surveillance system. In: Senior, A.W. (ed.) Protecting Privacy in Video Surveillance. Springer, Berlin (2009)
51. Senior, A.W., Pankanti, S., Hampapur, A., Brown, L., Tian, Y.-L., Ekin, A.: Blinkering surveillance: Enabling video privacy through computer vision. Technical Report RC22886, IBM T.J. Watson Research Center, NY 10598, August 2003
52. Senior, A.W., Pankanti, S., Hampapur, A., Brown, L., Tian, Y.-L., Ekin, A.: Enabling video privacy through computer vision. IEEE Secur. Priv. 3(5), 50–57 (2004)

53. Spiekermann, S., Grossklags, J., Berendt, B.: E-Privacy in 2nd Generation E-Commerce: Privacy Preferences Versus Actual Behavior, pp. 38–47. ACM Press, New York (2001)
54. Stana, R.: Video surveillance. Technical Report GAO-03-748, United States General Accounting Office, June 2003
55. Stauffer, C., Grimson, W.E.L.: Learning patterns of activity using real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(8), 747–757 (2000)
56. Sweeney, L., Gross, R.: Mining images in publicly-available cameras for homeland security. In: AAAI Spring Symposium on AI technologies for Homeland Security (2005)
57. Swiss Federal Data Protection Commissioner. Leaflet on video surveillance by private individuals. 3003 Bern, January 2003
58. The health insurance portability and accountability act (HIPAA) privacy and security rules (1996)
59. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cogn. Neurosci.* **3**(1), 71–86 (1991)
60. Venetianer, P., Zhang, Z., Scanlon, A., Hu, Y., Lipton, A.: Video verification of point of sale transactions. In: AVSS (2007)
61. Weaver, M.: Canoe mystery man arrested for fraud. *The Guardian*, December 2007
62. Wickramasuriya, J., Alhazzazi, M., Datt, M., Mehrotra, S., Venkatasubramanian, N.: Privacy-protecting video surveillance. In: SPIE International Symposium on Electronic Imaging (2005)
63. Yu, T., Lim, S.-N., Patwardhan, K., Krahnstoever, N.: Monitoring, recognizing and discovering social networks. In: Proceedings of Computer Vision and Pattern Recognition (2009)
64. Zhang, W., Cheung, S.-C., Chen, M.: Hiding privacy information in video surveillance system. In: Proc. International Conference on Image Processing (2005)



# Index

## Symbols

- 3D face capture, 364
- 3D face recognition, 431
- 3D face rendering, 149
- 3D Landmark detection, 437
- 3D model fitting, 257
- 3D morphable model, *see* Morphable model
- 3D reconstruction, 153, 364
- 3D representation, 138
- 3D scan, 138
- 3D+2D enrollment, 443
- 3D-3D face recognition, 435
- 3D-3D face recognition for partial scans, 436
- 3D-aided 2D face recognition, 443
- 3D-aided profile recognition, 440

## A

- AAM, 125, 306, 311, 370, 463, 685
  - For tracking, 478
  - Multi-resolution, 371
  - Tensor-based, 307
- Absolute ID, 676
- Access control, 620
- Action units, 465, 470
- Active appearance model, *see* AAM
- Active shape model, *see* ASM
- Active vision, 358, 360
- AdaBoost, 279, 280
  - Learning algorithm, 288
  - Optimal threshold of weak classifiers, 285
  - Strong classifier, 280
  - Weak classifier, 280
- AFEA, *see* Automatic facial expression analysis
- Aging modeling, 252
- Aging pattern, 258
- Aging simulation, 252

- Aligning shapes, 110
- Ambient video, 678
- Analysis by synthesis, 138, 149, 150
- Annotated Face Model (AFM), 429
- Anonymous surveillance, 676
- Anthropometric, 258
- Anthropometric method, 659
- Aperture stop, 356
- Appearance feature, 504
- Appearance models, 109, 118
- Appearance-based, 10, 278
  - Tracking, 473, 478
- Appearance-based representation, 140
- ASM, 92, 121, 306, 311
- AsymBoost, 290
- Asymmetric learning, 289
- Atmospheric distortion, 356
- Atmospheric transmission loss, 365
- Automatic enrollment, 369
- Automatic facial expression analysis, 488

## B

- Baseline, 364
- Basis functions, 20
  - KLT, 21
- Bayes rule, 152
- Bayesian face subregions, 210
- Benchmarking, 564
- Bidirectional relighting, 447
- Bilateral Total Variation (BTV), 375
- Binary tree, 313
- Bing streetside, 680
- Binning, 643
- Biometric samples, 552
- Biometric surveillance system, 366
- Blind vision, 684
- Blurring, 682

Bone-based animation, 469  
 Bones, 469  
 Bregman divergence, 71, 72

**C**

Camera calibration, 366, 367  
 Camera models, 471  
 Camera network, 338  
 Candide, 469, 474, 478  
 Caricatures, 604  
 Cascade of classifiers, 290  
 Certification, 688  
 Challenges, 7  
 Chi-square distance, 90  
 Closed-set identification, 552, 557  
 Cognitec FaceVACS, 369  
 Color  
     In video sequences, 238  
 Color camera, 226  
 Color cue  
     For face detection, 239  
     For face recognition, 244  
 Color spaces, 234  
     For skin, 234  
     Overlapping, 234  
 Compliance, 687  
 Compositional AAM, 131  
 Congestion charging, 678  
 Convex combination, 534, 535  
 Cooperation  
     Subject, 353, 355  
 Cooperative scenario, 3  
 Copyright protection, 681  
 Correspondence, 138  
 Cranial size, 258  
 Cryptography, 683

**D**

Database  
     Facial expression, 496  
     FERET, 36, 564  
     Long range, 359  
     MBGC, 337  
 De-identifying faces, 685  
 Deformable model fitting, 439  
 Deliberate expression, 494  
 Demographic information, 649  
 Department of motor vehicles, 663  
 Depth of field, 356  
 Developmental aspects, 589  
 Dimension  
     Intrinsic, 20

Dimensionality, 20  
 Discriminative Locality Alignment (DLA), 53, 63, 64

Distant Human Identification (DHID), 360  
 Distributed surveillance, 682  
 Double redaction, 683  
 Dynamic aspect, 575  
 Dynamic information, 576

**E**

Edgelet, 364  
 Eigen light-fields  
     Classification, 206  
     Definition, 202  
     Estimation algorithm, 203  
     Face recognition across pose, 204  
 Eigenface, 19, 140, 465, 685, 686  
 Eigenfaces, 24, 465, 479  
     Bayesian, 27  
     Probabilistic, 24  
 Eigenfeatures, 45  
 EKF, *see* Extended Kalman filter  
 EM-algorithm, 147  
 Emotional-based expressions, 489  
 Evaluation, 564  
 Evasive subject, 355  
 Exposure time, 356  
 Expression, 600  
 Expression details, 535, 538  
 Expression editing, 541, 543  
 Expression intensity, 493  
 Expression mapping, 533, 534, 538  
 Expression model, 120  
 Expression ratio image (ERI), 535, 536  
 Expression recognition, 506  
 Expressions, 575, 583  
 Extended Kalman filter, 463, 474, 476  
 Extrapersonal variations, 27  
 Eye candidate, 308  
 Eye localization, 307  
 Eye-pair, 311  
 Eyeglasses, 394

**F**

F-number, 355  
 F-stop, 356  
 Face acquisition, 488, 499  
 Face aging, 395  
 Face alignment, 370  
 Face cataloger, 361  
 Face categorization, 599  
 Face databases, 627  
 Face description using LBP, 87  
 Face detection, 4, 5, 277, 499

- Merge, 295  
Nonfrontal faces, 280, 292  
Face detector, 685  
Face Identification, 618  
Face image  
    Broad band, 401  
    Multispectral, 401  
    Narrow bands, 401  
Face image variations, 139  
Face matching, 4  
Face model compactness, 371  
Face modeling, 464, 522  
    from a single image, 522, 527  
    from an image sequence, 522  
    from two orthogonal views, 522, 526  
Face normalization, 4  
Face recognition, 5, 197  
    Across illumination, 199  
    Across pose, 197  
    Across pose and illumination, 201  
    Multiview, 197  
    under varying illumination, 532  
Face recognition processing, 3  
Face recognition system, 1  
Face recognition vendor test, *see* FRVT02  
Face recognition with spherical harmonic representations, 184  
Face relighting, 522, 529  
    from a single image, 530  
Face subregions, 210  
Face subspace, 20, 140, 141, 144, 147, 465  
Face tracking, 323  
Facebook, 679  
FaceVACS, 253  
Facial action coding system, 465  
Facial aging, 251  
Facial animation parameters, 466  
Facial definition parameters, 466, 468  
Facial expression analysis, 487  
Facial expression recognition, 488  
Facial expression synthesis, 522, 533  
Facial feature extraction, 502  
Facial feature representation, 502  
Facial landmark, 305  
Facial Landmark Model (FLM), 437  
Facial motion, 582  
Facial representation, 488  
FACS, *see* Facial action coding system  
FACS action units, 489  
False alarm rate, 553  
FAP, *see* Facial animation parameters  
FDP, *see* Facial definition parameters  
Feature extraction, 4  
Feature selection, 284, 389  
Feature-based tracking, 474  
Featureless, *see* Appearance-based  
Federal Bureau of Investigation (FBI), 657  
FERET, 90, 156, 551, 561, 564  
FG-NET, 254  
Field of view, 355  
Fisher light-fields  
    Definition, 214  
    Face recognition across pose and illumination, 214  
Fisherfaces, 26  
    Kernel, 36  
Fisher's Linear Discriminant Analysis (FLDA), 52–55, 57, 59, 60, 66, 71, 72, 74  
Fitting algorithm, 138, 150  
    Initialization, 151  
Fitting score, 158  
Flexibility, 640  
Flickr, 679  
FloatBoost, 289  
    Learning algorithm, 289  
fMRI, 582  
Focal length, 149, 355  
Focus-of-attention, 358  
Forensic examiners, 656  
Forensic photographic comparison, 656  
Forensic science, 655  
Forward feature selection, 288  
Fourier transform profilometry, 365  
Foveated imaging, 358, 361  
FRVT, 160, 551, 554, 558, 561, 567, 617  
Funk–Hecke theorem, 175  
Fusion of machine and human recognition, 610, 667
- G**  
Gait, 360  
    Fusion with face, 365  
Generative model, 138  
Generative Topographic Mapping (GTM), 53, 62, 64  
Geometric feature, 503  
Geometric Mean Subspace Selection (GMSS), 55–59  
Geometric warping, 534  
Geometry-driven expression synthesis, 536  
Google, 679  
    Goggles, 679  
    Street view, 680  
Grassmann manifold, 57, 335  
Gray-scale invariance, 84  
Ground truth, 496

**H**

- Haar-like features, 281  
 Harmonic Mean Subspace Selection (HMSS), 56–59  
 Harmonic reflectances, 180  
 Head orientation, 494  
 Head pose estimation, 500  
 Head rotations, 292  
 Hessian Eigenmaps (HLLE), 52, 53, 60, 62  
 Hierarchical principal component analysis, 539  
 HIPAA, 679  
 Histogram, 85, 86, 88  
 History of research, 1  
 Holography, 365  
 Horizontal scaling, 642  
 Human computer interaction (HCI), 631  
 Human face processing, 598  
 Human ID, 359  
 Human performance, 597  
     Characteristics of, 601  
 Human-machine comparisons, 607  
 HumanID gait challenge problem, 559

**I**

- ICA, 29  
 Identification, 2, 675  
     Across pose, 156  
     Across pose and illumination, 156  
     Confidence, 158  
 Identification rate, 553  
 Identity variation, 120  
 Identix faceIt, 369  
 Illumination, 608  
 Illumination change, 383  
 Illumination cone, 174  
 Illumination invariant, 386, 388  
 Illumination modeling, 149  
 Illumination normalization, 217  
 Image space, 19  
 Independent component analysis, 29  
 Integrability, 640  
 Interior point method, 539  
 Intervention, 681  
 Intrapersonal variations, 27  
 Inverse rendering, 529  
 Inverse-compositional AAM, 306  
 ISOMAP, 52, 53, 60, 62  
 Iterative closest point, 523

**K**

- Kalman filter, 463, 476  
 Kalman filter tracking, 366, 368  
 Karhunen–Loëve Transform, 21

- Kinetic Traveling Salesman Problem (KTSP), 363

**KLT**, 21**L**

- Lambertian, 170–174, 187, 200, 204, 325  
 Lambertian model, 386, 387  
 Laplacian Eigenmaps (LE), 52, 60, 62  
 Laser, 365  
 Laser radar, 365  
 Law enforcement, 626  
 LBP, 84  
 LBP-TOP, 86  
 LDA, 10, 26  
 Learning, 10  
 Legislation, 687  
 Light-field, 202  
 Linear asymmetric classifier, 290  
     Cascade learning goal, 290  
 Linear class of face geometries, 523  
 Linear discriminant analysis, 26  
 Linear Gaussian model, 146  
 Linear interpolation, 258  
 Local binary patterns, 84  
 Local tangent space alignment (LTSA), 53, 60, 62  
 Locality Preserving Projections (LPP), 53, 62, 66, 67, 74, 75  
 Locally Linear Embedding (LLE), 52, 60, 62  
 LRHM database, 359

**M**

- Manifold, 5  
 Manifold Elastic Net (MEN), 64–69  
 Manifolds  
     Nonlinear, 7  
 Max-Min Distance Analysis (MMDA), 52, 54, 57–59  
 Maximum a posteriori estimation, 151  
 MBE, *see* Multiple Biometric Evaluation  
 MBGC, *see* Multiple Biometric Grand Challenge  
 Mean shape, 259  
 Mechanical vibration, 356  
 Media space, 678, 685  
 Medical images, 679  
 Metric, 523  
 Model space, 523  
 Model-based bundle adjustment, 525  
 Model-based tracking, 472  
 Model-driven bundle adjustment, 522  
 Monte Carlo method, 329  
 Morph targets, 468  
 Morph-based facial expression synthesis, 533

- Morphable model, 138, 170, 255, 320  
Construction of, 141  
Regularized, 146  
Segmented, 147
- Morphology, 657
- Motion blur, 356
- Motion propagation, 539
- Motion-based tracking, 472
- MPEG-4 Facial Animation, 463, 466
- Mugshot, 657
- Multi-stage comparison, 643
- Multi-view videos, 339
- Multimedia management, 630
- Multiple Biometric Evaluation, 551, 569
- Multiple Biometric Grand Challenge, 337, 359, 360, 571  
NIST, 359
- Multispectral face database, 409
- Multispectral imaging, 402  
Band selection, 402, 408  
Capture systems, 403  
Feature extraction, 408  
Fixed-filter systems, 404  
SpectraCube, 405  
Turnable filter systems, 404
- Multispectral power distributions, 412
- N**
- Narrow field of view, 358
- Near infrared face image  
Matching, 391  
Modeling, 387
- Near infrared imaging, 384  
Active lighting, 384  
Frontal lighting, 384  
Non-intrusive, 384  
Outdoor, 396  
System, 384
- Neurophysiological aspects, 582, 590
- NFOV, *see* Narrow field of view
- NIR, *see* Near infrared
- NIR face recognition, 383
- Non-additive combinations, 493
- Non-cooperative scenario, 3
- Normalized color coordinates (NCC), 235
- O**
- Object centered representation, 140
- Omnivident, 678
- Open-set identification, 552, 553
- Operating point, 686
- Optical axis, 149
- Orthogonal projection, 256
- Orthographic projection, 470
- P**
- Pan-tilt-zoom camera, 358, 366  
Calibration, 367
- Paparazzi, 681
- Parametric face model, 523
- Parrot recognizer, 682
- Partial Registration, 439
- PCA, 10, 21, 51, 53, 60, 64, 66, 67, 71, 74, 111, 172, 312, 465  
Kernel-PCA, 35  
Nonlinear, 33
- Penetration rate, 649
- Perceptual adaptation, 604
- Performance evaluation, 551
- Performance measures, 552
- Performance statistics  
Variance of, 559
- Permutation
- Pixel, 682
- Person detection, 366
- Person tracking, 366
- Perspective projection, 149, 471  
Weak, 149
- Phong model, 149
- Photometric stereo, 189
- Physically-based facial expression synthesis, 533
- Picasa, 679
- PIE (Pose, Illumination and Expression), 153, 156, 357
- Pittsburgh pattern recognition, 360
- Pixelation, 682
- Plenoptic function, 202
- PolarRose, 679
- Police agencies, 664
- Pose correction, 254
- Pose normalization, 160
- Principal component analysis, *see* PCA
- Principal components, 255
- Principal components analysis, 144
- Principal components analysis  
probabilistic, 146
- Principal curves, 33
- Privacy, 671  
Data, 673  
Visual, 673
- Privacy concern, 672
- Privacy policies, 684
- Privacy protection, 671
- Privacy registrar, 688

- Privacy threat, 675  
 Privacy tokens, 684  
 PrivacyCam, 682, 688  
 Procrustes analysis, 110  
 Profile extraction, 441  
 Projection models, 470
- Q**  
 Quadratic programming, 539  
 Quality of model matching, 126  
 Quotient image, 529
- R**  
 Radial basis function, 258  
 Radiance environment map, 530  
 Random forest, 311  
     Classifier, 313  
     Embedded ASM, 311  
 RANSAC, 481  
 Ratio image, 529, 532, 533  
 Re-rendering across pose, 204  
 Redaction, 681  
 Region of interest (ROI), 443  
 Reinforcement learning, 361  
 Relative ID, 677  
 Relighting from a single image, 530  
 Rerendering, 682  
 Resource allocation, 363  
 Reversibility of redaction, 682  
 Rigid transformation, 149  
 Robots, 378  
 Rotation matrix, 257
- S**  
 Scalability, 640  
 Scale out, 642  
 Scale-up, 642  
 Scrambling, 682  
 Security, 623  
 Sequential importance sampling, 329  
 SfM, *see* Structure from motion  
 Shape AAM, 131  
 Shape models, 109  
 Shape pattern, 258  
 Shape reconstruction, 189  
 Simultaneous inverse compositional (SIC), 370  
 Singular value decomposition, 22  
 Skin color, 223  
     Canonical image, 228  
     Difficulty in using, 223  
     In face detection, 224, 225  
     Mathematical models, 236  
     Non-canonical image, 231  
     White balancing, 227
- Skin reflectance model, 445  
 Skinning, 469  
 Smart cards, 625  
 Social network, 362  
 Spatio-temporal patterns, 326  
 Spatiotemporal LBP, 86  
 Specular reflectance, 187  
 Spherical harmonic basis, 530  
 Spherical harmonic basis image, 530  
 Spherical harmonic morphable model, 522, 530  
 Spherical harmonic representations, 174  
 Spontaneous expression, 494  
 Stereo baseline, 364  
 Still-to-video matching, 326  
 Stochastic Newton optimization (SNO), 152  
 Strong classifiers, 287  
 Structure from motion, 463, 474, 476  
 Subspace, 4, *see also* Face subspace, 320  
 Subsurface scattering, 171  
 Super-resolution  
     Facial, 363, 375  
     Facial side-view, 365  
     General, 375  
     Registration, 375  
     Regularization, 375  
 Surveillance, 3, 625  
 Surveillance perspective, 357  
 SVD, *see* Singular value decomposition  
 Synthetic face image, 149
- T**  
 Task visibility interval, 363  
 Technologies, 11  
 Tensorfaces, 31  
 Texture mapping, 469, 479  
 Texture models, 115  
 Texture pattern, 259  
 Texture primitives, 85  
 Texture representation, 118  
 Transparent society, 672  
 Throughput, 642  
 Tikhonov regularization, 256  
 Total information awareness (TIA), 680  
 Tracking-then-recognition, 326  
 Transfer Subspace Learning (TSL), 71  
 Transitions, 492  
 TRUSTe, 688  
 Typicality, 602
- U**  
 University of Texas at Dallas, 359  
 Unsharp masking, 363

## Utility of video

- Appearance modeling, 325
- Ensemble matching, 325
- Frame-based fusion, 324

UTK-LRHM database, 359, 363

**V**

- Vectorization, 205
- Verification, 2, 552, 556
- Vertical scaling, 642
- Video privacy, 672
- Video sequence, 323
- Video-based, 323
- Video-to-video matching, 334
- Virtual environment simulator, 363

## Virtual view, 160

- Visual privacy, 671, 675
- Volume LBP, VLBP, 86

Voyeurism, 675

**W**

- Watchlist identification, 3
- Weak classifier, 284
  - Decision tree, 284
  - Stump, 284
- Weak perspective projection, 471
- WFOV, *see* Wide Field Of View
- Wide Field Of View, 358
- Windows live photo gallery, 679