3D Face Modelling Under Unconstrained Pose & Illumination

by

David Bryan, BEngMgt

A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for
the degree of Master of Applied Science in Biomedical Engineering

Ottawa-Carleton Institute for Biomedical Engineering
Department of Systems and Computer Engineering
Carleton University
Ottawa, Ontario, Canada
January 2009

The undersigned recommend to the
Faculty of Graduate Studies and Research
acceptance of the thesis

3D Face Modelling Under Unconstrained Pose & Illumination

Submitted by David Bryan, BEngMgt
in partial fulfillment of the requirements for
the degree of Master of Applied Science in Biomedical Engineering

Thesis Supervisor
Dr. Andy Adler

Chair, Department of Systems and Computer Engineering
Dr. Victor Aitken

2009, Carleton University

Abstract

This thesis presents a system for three-dimensional modelling of the face to fit two-dimensional input images that are not constrained for pose or illumination. The system consists of a face-fitting algorithm that works in conjunction with a 3D morphable face model to synthesize a face whose projected image resembles as closely as possible the target image under analysis. The fitting algorithm optimizes across parameters that define face shape and texture, facial pose, and illumination conditions.

We show that this system is able to generate a face model with sufficient accuracy that the model is deemed to represent the same identity as the target according to a measure of face similarity. The face model can also be normalized for pose and illumination to provide more suitable input for an appearance-based face recognition algorithm. It was found, however, that lack of extracted skin detail limited success in this regard.

Acknowledgements

I would like to express my deepest gratitude to my supervisor Dr. Andy Adler, for the resources and helpful guidance he provided. I truly appreciate the opportunity I had to study with Dr. Adler.

Prior to pursuing graduate studies, as I investigated potential avenues for research related to image analysis tools for use in law enforcement or security applications, several people who work or have worked in law enforcement agreed to speak with me and were particularly helpful. I am indebted to Jennifer Strachan, Paul Gillespie, and Warren Bulmer who were all very generous with their time and readily shared their knowledge and experience. Their insight helped frame the motivating question I hoped to address in my research. Thank you.

Finally, I would like to thank my family and friends for their loving support and prayers. Most of all, I thank my wife Shirley for her patience, support, and encouragement over the years.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Human faces are complex three-dimensional objects characterized by distinguishing features such as eyes, mouth, and nose and by skin pigmentation. Nonetheless, humans are generally able to routinely and with minimal effort identify and recognize faces of others [42]. Perhaps inspired by this innate human ability, researchers from various disciplines have investigated and pursued machine-based automatic face recognition for over 40 years [73]. And while decades worth of research has afforded a certain level of maturity to the technology of automatic face recognition — manifested in numerous commercially available systems — there are numerous challenges that still exist.

1.1 The Biometric Context

The International Biometric Industry Association defines a biometric *characteristic* as a "measurable biological (anatomical and physiological) or behavioural characteristic that can be used for automated recognition" [1]. Similarly, a biometric *process* is an

automated method of recognizing individuals based on these measurable biological or behavioural characteristics.

In comparison with traditional identity management tools such as passwords, personal identification numbers (PINs), and smart cards, biometric technologies generally offer both enhanced security and user convenience [39]. As such, these technologies are increasingly being deployed in secure personal identification and verification systems at airports and other secure facilities, for computer and/or network access, in government documentation such as passports and drivers' licences, and in surveillance applications. Indeed, this increasing demand for reliable automated personal authentication has driven commercial implementations of physiological biometrics including face, fingerprint, iris, and retina and behavioural biometrics such as signature, keystroke, and gait, and continues to drive research in these areas as well as more obscure biometrics like body odour and ear shape [69].

The applicability of a particular biometric to a particular application depends on an evaluation of factors including perceived intrusiveness, distinctiveness (i.e., how accurately can different individual identities be resolved?), cost of deployment, and ease-of-use [29]. For instance, in 2001 the International Civil Aviation Organization (ICAO) evaluated a number of biometrics for potential use in machine-assisted identity confirmation upon presentation by aviation passengers of a machine-readable travel document (MRTD). Based on the evaluation criteria and a comprehensive set of constraints imposed by the ICAO, *face* was found to have the highest compatibility rating with their application [30].

Table 1.1: Typical applications of biometric face recognition.

Category	Application Scenario
Face identification	Drivers' licences
	Passports, immigration, national ID cards
	Entitlement program registration
Access control	Facility and vehicle access
	Computer and computer network access
	Online transaction access
Surveillance and security	Advanced video surveillance, CCTV control
	Secure flight boarding systems
Law enforcement	Suspect tracking and investigation
	Criminal face retrieval and recognition
Smart cards	Stored value security
	User authentication
Multimedia management	Face-based indexing and retrieval
	Face-based video segmentation and summarization
Human computer interaction (HCI)	Interactive gaming
	Face tracking for video-conferencing

1.2   Face as a Biometric

There are, in fact, many potential applications in which the use of face as a biometric identifier proves to be suitable. A fairly exhaustive list and description of applications is given in [28], with a condensed list provided here in table 1.1.

The various application scenarios outlined in table 1.1 can be classified broadly into the following three tasks [34, 48, 75]:

- **Verification.** Does the probe face image match the claimed identity? This involves a *one-to-one* comparison evaluated against some predetermined similarity score threshold. A typical application might involve the use of the face biometric in conjunction with or in lieu of a password for access to a secure facility or computer network.

- **Identification.** Whom does the probe face image represent? Identification

3

involves a *one-to-many* matching process in which the probe image is compared with a database of images with known identities, with the assumption that the query face image does represent a known identity in the database. Generally, the system will return the top match or the top k most likely matches. This task is modelled after law enforcement applications in which a suspect's image would be compared with a database of mugshots of known criminals [48].

- **Watch List.** Does the probe face image represent someone of interest? This involves a *one-to-few* comparison in which the system determines if the probe face represents an individual on a stored watch list and, if yes, returns the identity of that individual. Clearly, a typical application would be surveillance scanning of a crowd — perhaps at an airport — looking to identify anyone on a terrorist watch list and subsequently raise an alarm if a match is found. Another potential application might involve content-based indexing of an image and/or video database (described in [7]) in which the basis for search and retrieval is the face of a person of interest.

The verification task is different from the identification or watch list tasks in that it utilizes a claimed identity. A corollary to this is that in verification it can be assumed that the individual presenting his or her face to the recognition system is a cooperative subject, motivated to use the system in the proper manner and to achieve a successful result. In such cases, conditions can be imposed on image capture to improve system performance. Specifically, subjects can be required to assume a frontal face pose with neutral expression under controlled lighting conditions and at a specific distance to the image capture device. Recent independent evaluations of face recognition technology — the Face Recognition Grand Challenge (FRGC) [45,46]

and the Face Recognition Vendor Test (FRVT 2006) [49] — have shown that current state-of-the-art automated face recognition systems are capable of high performance under these controlled conditions. These same tests have shown, however, that recognition accuracy suffers significantly when conditions for facial image capture are not constrained.

1.3   Challenges to Face Recognition

Notwithstanding the expected differences in visual appearance between faces of different individuals, variations between face images can arise due to facial expression or several extrinsic factors such as pose or viewing angle, illumination, occlusion, and time delay between acquisition of the probe and gallery images that are being compared [24]. When a face recognition system is unable to constrain any or all of these factors, performance can be expected to suffer. Yet most potential surveillance and screening applications of face recognition technology involve subjects who are unaware that a biometric system is in place and make no effort to present themselves to the image capture device. In such situations, the system is unlikely to obtain a clear, favourably illuminated frontal view of the face with neutral expression.

While many techniques for the face recognition task and/or methods for preprocessing the facial images have been proposed, robust face recognition under unconstrained conditions remains difficult and elusive [34, 74, 75]. Among the challenges an unconstrained situation presents, pose and illumination are perhaps the most significant. In fact, one study concluded that variations between images of the same face due to differences in illumination and viewing angle generally exceed the image variation due to a change in face identity [4]. These difficulties were underscored

| (a) | (b) | (c) |
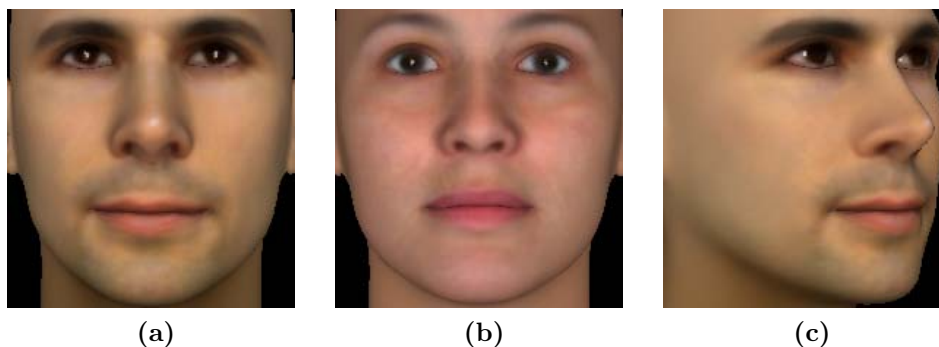
Figure 1.1: The pose problem illustrated. Under most measures of image similarity, images of different individuals at similar poses ((a) and (b)) would be deemed to be more similar than images of a single individual at different poses ((a) and (c)). [Images generated by FaceGen modeller, as described in section 2.3.]

in the evaluation report for the Face Recognition Vendor Test (FRVT 2002) [48] in which pose and illumination were identified as key future directions of research.

1.3.1 The Pose Problem

That unconstrained pose is a hindrance to effective, real-world face recognition is not unanticipated. After all, even humans show a decline in recognition of unfamiliar faces with increasing disparity in pose between learning and test faces [54]. Figure 1.1 illustrates the effect of pose on facial image similarity. Although images 1.1a and 1.1c represent the same individual from frontal and oblique viewpoints, respectively, while image 1.1b represents a different individual, most measures of similarity would declare images 1.1a and 1.1b to be more similar than 1.1a and 1.1c [62]. For instance, a straightforward pixel-wise measure of the L_2-difference (i.e., the sum of the squared difference between pixel values over all pixel locations) would imply the two frontal images of different people are more similar than the images of the same person from different viewpoints.

Dramatic changes in appearance of a given face upon changes in pose arise quite

Figure 1.2: The illumination problem illustrated. Under most measures of image similarity, images of different individuals under fixed illumination conditions (bottom row) would be deemed to be more similar than images of a single individual under varying illumination conditions (top row). [Images generated by FaceGen modeller, as described in section 2.3.]

simply from the three-dimensional structure of the face. As pose changes, the appearance of the face can change due to projective deformation which alters the spatial configuration of facial features and self-occlusion as parts of the face disappear from view [24].

1.3.2 The Illumination Problem

Even if the pose of a given face is fixed, illumination variation can significantly change its appearance. This is evident in figure 1.2 in which the top row contains four images of an individual with fixed pose but under varying illumination while the bottom row contains images of four different individuals under fixed pose and illumination. Again, a measure of the L_2-difference would judge any pair of images from the bottom row to be more similar than any pair of images from the top row [27].

Given the three-dimensional shape of the face and depending on the positions, distribution, and intensity of any light sources, the appearance of a face can change in terms of overall brightness as well as with the locations of any shadows or specular reflections. Cast shadows from a direct light source, in particular, can accentuate or diminish certain facial features [22, 24] as is clearly evident in the upper row of figure 1.2.

1.4   Thesis Objective and Outline

The objective of this thesis is to address the following question: Is it possible to accurately predict the appearance of an individual and subsequently generate a frontal and uniformly illuminated view of their face from an image that is unconstrained in pose and illumination?

The approach taken to address this question is an *analysis by synthesis* framework whereby a generative three-dimensional (3D) face model is used in conjunction with a face-fitting algorithm to synthesize a face model whose projected image resembles as closely as possible the target image being analyzed. This face model can then be normalized for pose and illumination such that an image of the subject can be re-rendered under conditions that would make the image more suitable for use in a face recognition application. While not developed as part of this thesis work, the 3D face model plays a key role and is described in chapter 2 along with some other pertinent background information. The face-fitting algorithm that was developed as part of this thesis work is described in chapters 3 and 4 which focus primarily on the solutions to the pose and illumination problems, respectively. In chapter 5 the system of face model and analysis algorithm is tested to determine how well it can address

8

the thesis objective by comparing novel views of individual faces — synthesized from images taken under sub-optimal pose and/or illumination conditions — with ground truth views of those same faces.

Chapter 2

Background

While face recognition systems are capable of high performance under controlled imaging conditions, variation in pose and illumination presents a significant challenge [34, 74, 75]. A number of methods have been proposed in recent years for pose and illumination invariant face recognition with arguably the most successful of these being a preprocessing approach based on a 3D morphable model of the face, which is described in detail in section 2.2. In section 2.3, a commercially available face modeller that is based on this 3D morphable model approach and which serves as the backbone for the face analysis conducted as part of this thesis is described. In section 2.4 the descriptors that will be used throughout this paper for rotation in three-dimensional space are established. Finally, in section 2.5 some standard face recognition system performance measures are described, with emphasis on those measures that will be used in the evaluation of our system which combines a face-fitting algorithm with a 3D face model.

2.1 Approaches to Face Recognition

Many approaches to face recognition have been proposed in recent years, with most being optimized to perform on frontal, uniformly illuminated views of the face. The reader is referred to [74] for a fairly recent critical survey of many of these methods.

There have been relatively few suggestions, meanwhile, of techniques to handle the joint problem of pose and illumination variation. In general, the goal is to make face recognition independent of the imaging conditions by distinguishing the subject's identity from the extrinsic imaging parameters. Ideally, it is desirable that any approach be able to generalize in all three factors involved, namely, subject identity, pose, and illumination [75]. That is, in considering the problem of unconstrained pose and illumination, the ideal face recognition system would be capable of identifying any person under any pose and any illumination conditions.

The 3D morphable model approach introduced by Blanz and Vetter [11] is considered the state-of-the-art approach for pose and illumination invariant face recognition [75], and tends to be the benchmark against which other proposed solutions measure themselves. In comparison with this model-based approach, appearance-based approaches extended to contend with variable pose and illumination tend not to perform as well. Certainly this is the case with two such techniques described in [21], namely, Fisher Light-Fields [23] and Bayesian Face Subregions coupled with an illumination preprocessing algorithm [22]. Somewhat more favourable results are reported for Illuminating Light Fields, another appearance-based light-field approach described in [75], but performance with this approach still falls short of that achieved with 3D morphable models. And, while the morphable model can generalize in pose, these appearance-based approaches consider only discrete sets of poses.

Were it possible, perhaps the most straightforward way to generalize across pose would be to have an image database of each subject at each possible pose position and to constrain the face recognition task to the detected pose position [9, 44]. A similar approach could be taken for illumination. The Tensorface approach [65, 66] builds on this idea with multi-linear image analysis whereby multiple images of the subject are used to build a statistical model to interpolate to novel poses and illumination conditions. It is argued in [75], however, that while identity and illumination are suitable for linear analysis, pose is not.

More recently, some researchers have proposed techniques for which they report performance in unconstrained face recognition that exceeds that of the morphable model approach. Two of these techniques are summarized as follows:

Tied factor analysis. Recently introduced as an approach for face recognition under variable pose but extensible to variable illumination, tied factor analysis is based on a generative model that creates a probabilistic mapping from an idealized "identity space" to models of individual facial features that change with pose [53]. This is purely a machine learning approach and does not generalize in pose but rather learns system parameters only for discrete poses.

Video-based modelling. In [70], the authors present a framework for integrating joint models for shape, illumination, and motion from *video sequences*. Given a probe sequence that is unconstrained in pose and illumination, the probe face is tracked to estimate its 3D motion (i.e., pose over time) while simultaneously an illumination model recovers illumination parameters as a function of time. The learned motion and illumination parameters are used to synthesize video sequences for each candidate in a gallery of 3D face models with the same

12

motion and illumination conditions as the probe. Gallery and probe similarities are evaluated, according to some distance measure, via comparison of the synthesized sequences and the probe sequence.

2.2   3D Morphable Models

Blanz and Vetter [11] introduced the 3D Morphable Model (3DMM) nearly a decade ago, building on the concept that faces form a linear object class and therefore, given a set of representative basis faces that span "face space", a new face image can be approximated as a linear combination of those representative faces set in shape correspondence with one another [8, 32, 67]. The morphable face model exploits the statistics of a large dataset of 3D face scans to encode shape and texture in terms of model parameters.

These model parameters describe the statistical variation in shape and texture that occurs with changes in identity. Other sources of variation that occur in facial images, meanwhile, can also be modelled but with independent sets of parameters [55]. Changes in facial pose, considered separately from shape, are rendered using a computer graphics simulation of perspective projection techniques [11]. And since the model is three-dimensional, surface normals can be directly computed and the interaction with light can be simulated using an illumination model [56].

2.2.1   Model Construction

A detailed description of the construction of a 3D morphable model from a dataset of 3D face scans can be found in [13]. Although no model was constructed for this thesis, the model generation process is briefly summarized here to lend insight to the

13

derivation and function of shape and texture model parameters.

To construct the model, a set of M 3D face scans are put into dense point-to-point correspondence with some reference face scan. This correspondence yields consistent labelling of all n vertices in each of the scans. (For the model described in [13], $M = 200$ and $n = 75,972$.) Shape and texture vectors for each subject $i \in 1, \ldots, M$ are then defined by

$$\mathbf{S}_i = (x_1, y_1, z_1, \ldots, x_n, y_n, z_n)^\mathsf{T}, \tag{2.1}$$

$$\mathbf{T}_i = (R_1, G_1, B_1, \ldots, R_n, G_n, B_n)^\mathsf{T}, \tag{2.2}$$

where (x_k, y_k, z_k) represents the position of the k-th vertex and (R_k, G_k, B_k) gives its colour.

Since the faces are in full correspondence, novel face shapes and textures can be created as convex linear combinations [11] of the M shape and M texture vectors as follows:

$$\mathbf{S} = \sum_{i=1}^{M} a_i \mathbf{S}_i, \quad \mathbf{T} = \sum_{i=1}^{M} b_i \mathbf{T}_i, \quad \sum_{i=1}^{M} a_i = \sum_{i=1}^{M} b_i = 1, \tag{2.3}$$

or more conveniently, novel shape and textures can be described as a linear combination of N_S shape and N_T principal components,

$$\mathbf{S} = \bar{\mathbf{S}} + \sum_{i=1}^{N_S} \alpha_i \mathbf{s}_i, \quad \mathbf{T} = \bar{\mathbf{T}} + \sum_{i=1}^{N_T} \beta_i \mathbf{t}_i. \tag{2.4}$$

$\bar{\mathbf{S}}$ and $\bar{\mathbf{T}}$ represent the average face shape and texture, respectively, and the principal components $\mathbf{s}_i$ and $\mathbf{t}_i$ are derived from the set of shapes and textures using Principal Component Analysis (see [31]).

The coefficients α_i and β_i are the shape and texture model parameters, respectively, and are theoretically capable of describing any face or, at least, any face that

falls within the "face space" implied by the range of face shapes and textures in the dataset of 3D scans. Images of the face model described by the parameters can subsequently be rendered according to any modelled pose and illumination parameters [55].

2.2.2 Fitting the 3D morphable model to 2D image

Image analysis by model fitting is essentially an inversion of the modelling process. Given a target image, the goal is to find the shape and texture coefficients α_i and β_i, along with parameters describing the face position, orientation, and illumination, such that an image of the model rendered according to these parameters matches as closely as possible the target image [12]. This process is illustrated in figure 2.1.

This is an ill-posed, non-linear problem that requires some optimization strategy [56] to solve. The particular strategy implemented for the purposes of this thesis will be described in section 3.5.

2.3 The FaceGen Face Modeller

The FaceGen Face Modeller is a standalone Windows-based parametric face modelling application produced by Singular Inversions Inc. Based on the morphable model framework, this product is targeted primarily to computer game developers and graphic artists for character creation. FaceGen enables 3D face creation from a frontal photograph, random face creation, and manipulation of face characteristics via shape and texture controls. A screen capture of the user interface is given in figure 2.2.

Shape and texture parameters are controlled by sliders whose range of appropriate values are governed by the distribution of underlying statistical appearance models.
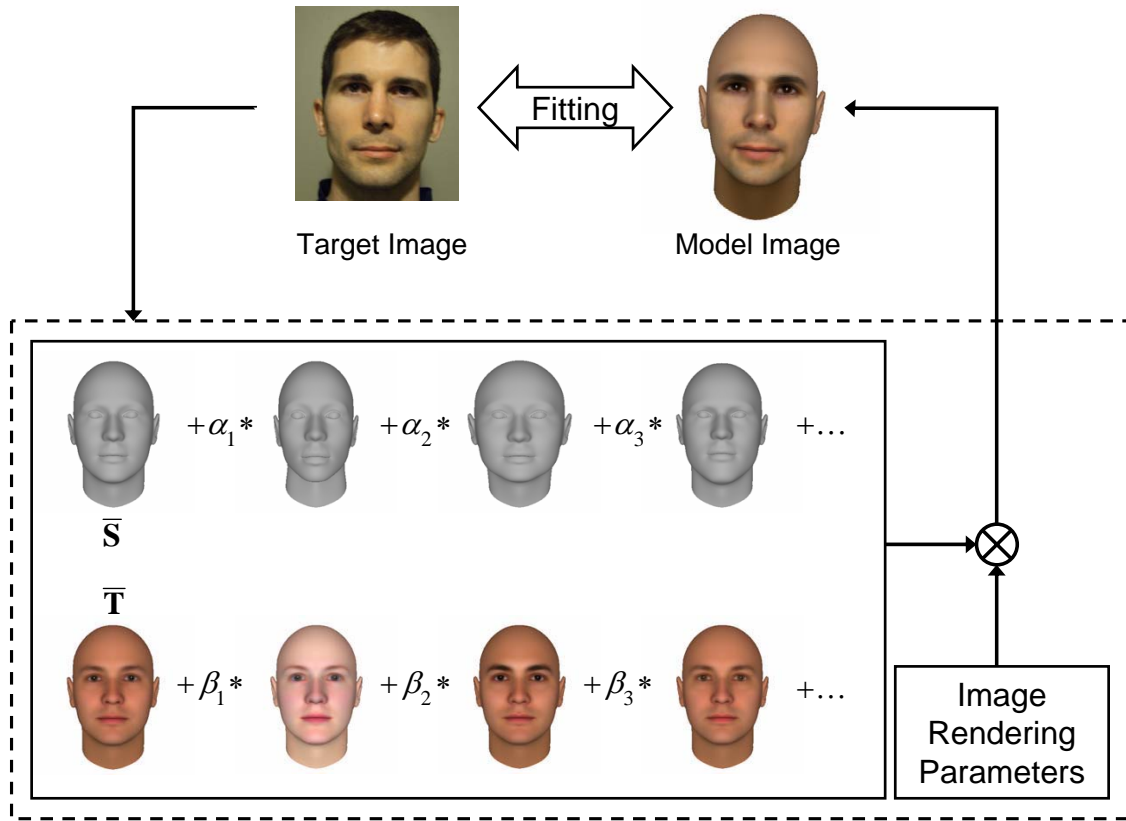
Figure 2.1: Fitting the 3D morphable model to a 2D image. The goal of the fitting process is to derive the shape and texture coefficients α_i and β_i that define a 3D face model that, when rendered according to the model imaging parameters, produces an image as similar as possible to the target. [Images generated by FaceGen modeller, as described in section 2.3.]

16
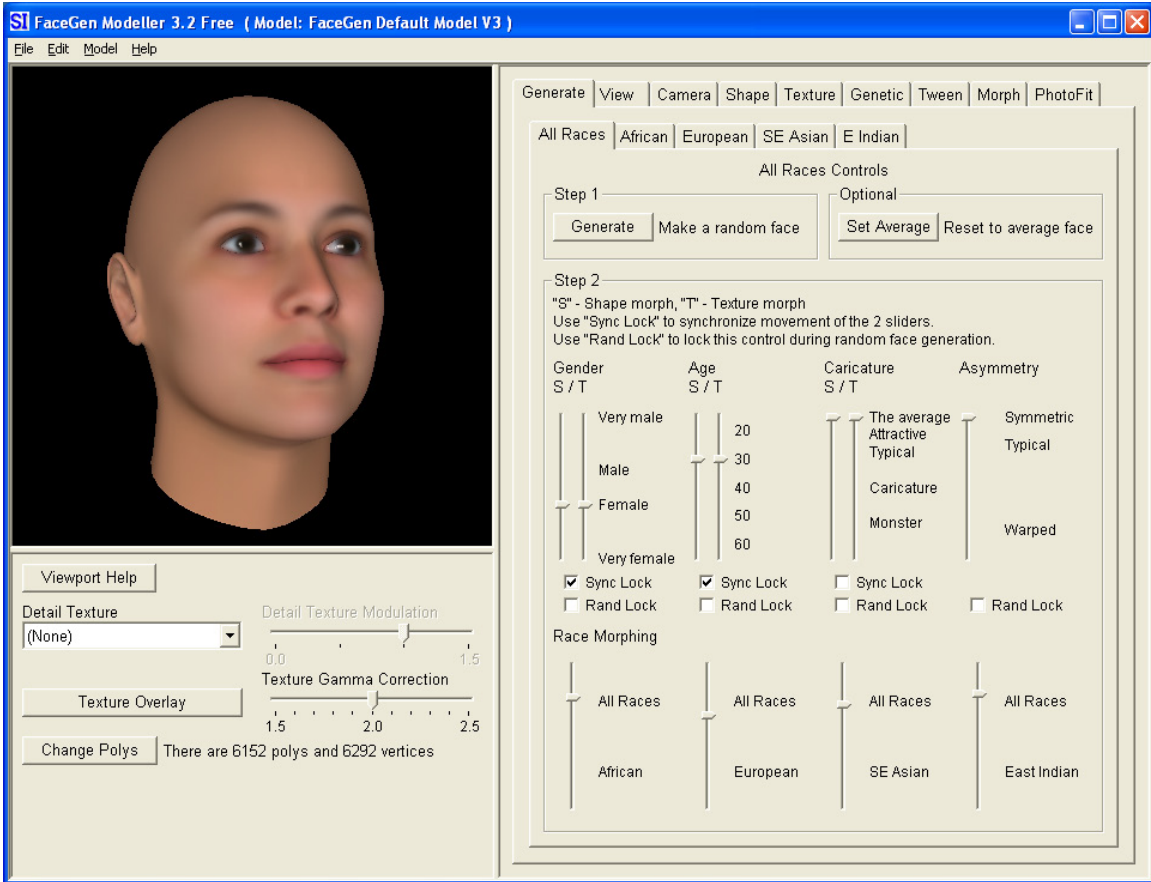
Figure 2.2: Screen capture of the FaceGen Face Modeller user interface.

In addition to shape and texture manipulation, the FaceGen Modeller allows head rotation, translation, and scaling through user mouse control with the mouse positioned over the face viewing window. The apparent brightness of ambient light and the position and brightness of up to three direct light sources can be manipulated. The modeller also allows morphing of the face shape consistent with various expressions and phonemes (i.e., distinct speech sounds), yet these controls were created by an animation specialist [61] and are not based on any statistical analysis of face shape deformation with facial expression.

Faces in the FaceGen environment are represented by a parameter vector of 130 signed floating point numbers [60]. The first 50 of these numbers describe the symmetric shape of the face, the next 30 the asymmetry in the shape, and the final 50 describe the texture (or colouring) [60]. One of these parameter vectors can be considered to be equivalent to the set of α_i, β_i shape and texture coefficients in the morphable model framework.

Singular Inversions offers a freely downloadable evaluation version of the FaceGen Modeller. This free version served as the morphable face model for the purposes of the research described in this thesis document, with the manipulation of shape, texture, and image rendering parameters being automated as described in the following. Usage of this software for evaluation purposes as part of this thesis research was verified to be compatible with the software license agreement.

2.3.1 Automation of Modeller Parameter Manipulation

Rather than using the slider controls in the FaceGen GUI to manipulate face shape and texture, these characteristics can be altered by direct manipulation of the 130 component parameter vector. Parameter vectors are stored in unique face coordinate

files whose contents can be read and the 130 floating point numbers that comprise the parameters extracted. Similarly, face coordinate files can be created by writing a 130 component parameter vector, with a previously extracted file header, to a file with the appropriate file extension.

Interaction with the GUI itself can be automated using the "Win32::GuiTest" Perl module [2]. This module provides a set of methods for manipulating window handles and controls and to simulate user input such as mouse clicks and movements, keyboard input, and window selection. A number of Perl scripts created for the automation of face model manipulation are included in appendix A.

2.3.2 The PhotoFit Utility

Within the FaceGen modeller is a utility called PhotoFit that creates 3D face models by model fitting to a *frontal* photograph, with the optional use of one or two associated profile photographs to improve the results. There are specific guidelines that photographs should meet in order to achieve good results [61]. These include (a) neutral facial expression, (b) mouth closed, (c) flash lighting used, and (d) the face region should be at least 512 x 512 pixels in size. PhotoFit requires that 11 facial feature points be manually located using an interactive tool on the frontal view and 9 points be located on the profile view(s) [60].

Thus PhotoFit does perform the model fitting that is the goal of this thesis work, but for constrained conditions only. This provides a convenient way of evaluating the performance of model fitting to unconstrained images when a well-constrained image of the same subject is also available. Considering the PhotoFit result from the constrained image to be a valid representation of the subject's 3D face shape and texture, direct comparison of this parameter file with the parameter file derived from

19
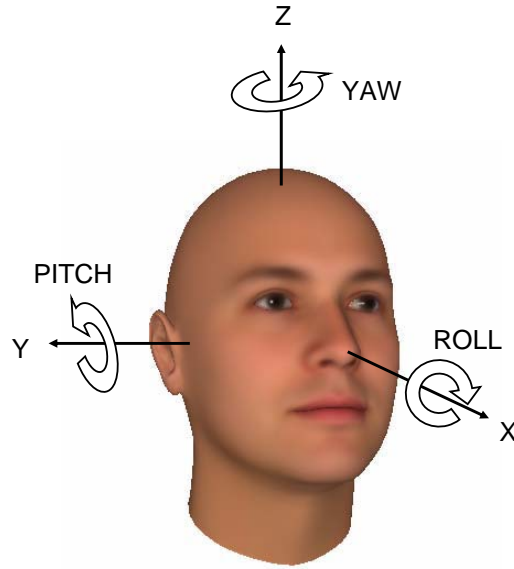
Figure 2.3: Classification of facial orientation in space by the rotational descriptors pitch, roll, and yaw.

model fitting to the unconstrained image provides a direct measure of the quality of the fit to the unconstrained image. This represents one of the modes of algorithm evaluation to be discussed in section 5.3.

2.4   Rotational Descriptors

To classify facial orientation in space, the three aeronautical rotational descriptors — pitch, roll, and yaw — are adopted. This is consistent with the emerging trend for describing dentofacial traits and orthodontic problems, as described in [3]. As shown in figure 2.3, *pitch* describes rotation in the anteroposterior plane (about the Y axis), *roll* describes rotation in the transverse plane (about the X axis), and *yaw* describes rotation about the vertical (Z) axis. In the sections to follow in this thesis, pitch, roll, and yaw will be respectively designated by the symbols θ, ϕ, and ψ.

2.5 Face Recognition Metrics

As introduced in section 1.2, face recognition applications can be classified in one of the following three representative tasks: verification, identification, and watch list. While all three are closely related, each of these tasks has its own performance metrics against which the performance of face recognition systems can be evaluated [47].

Generally, face recognition systems performing the verification task are evaluated based on the frequency with which two types of errors occur. *Type I errors* occur when a true match is incorrectly rejected. Conversely, *type II errors* occur when a non-match is incorrectly accepted [69]. A system's probability of type I error is commonly referred to as its False Reject Rate (FRR) while probability of type II error is known as False Accept Rate (FAR). The FAR and FRR are inversely related and depend on the system *match threshold*.

The identification task differs somewhat in that the face recognition system generally does not seek to determine whether the top match is correct, but rather if the correct identity falls within the top k matches [47]. This involves a ranking of the similarity scores between the face being tested and all the faces in the database against which it is being compared. The "rank-k" identification rate, $P_I(k)$, is defined as follows:

$$P_I(k) = \frac{C(k)}{N}, \tag{2.5}$$

where $C(k)$ is the cumulative count of the number of test images that are correctly identified within the top k or fewer most likely matches on the ranked list and N denotes the total number of test images considered in the evaluation.

Identification performance is often reported on a cumulative match characteristic (CMC) curve. A CMC plots the identification rate, $P_I(k)$, as a function of rank
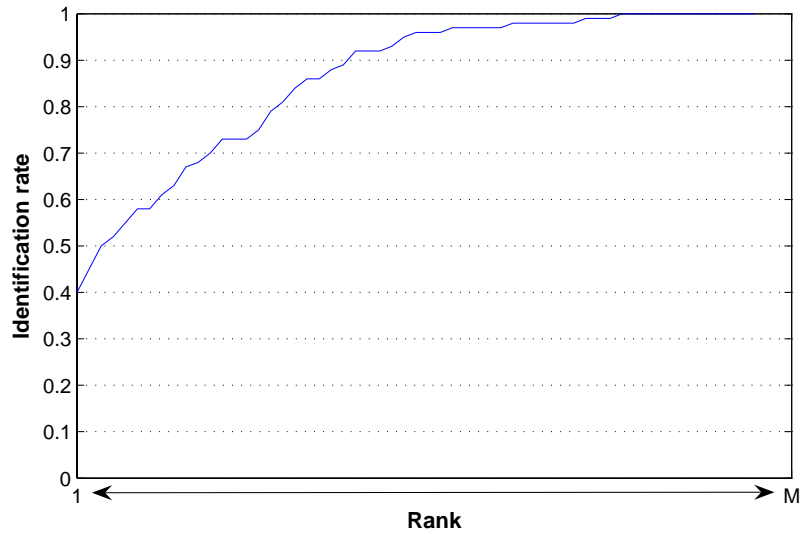
Figure 2.4: Example of a cumulative match characteristic (CMC) curve used to report identification performance.

k [47]. Figure 2.4 shows an artificially created CMC curve to illustrate two keys features. First, $P_I(k)$ is necessarily a non-decreasing function, and second, all algorithms achieve an identification rate of 1.0 as rank approaches M, where M represents the size of the gallery.

The system that was developed for this thesis and which will be described in the forthcoming chapters is evaluated in chapter 5 in terms of its identification performance.

Chapter 3

Image Analysis for Fitting 3D Face Model

The following two chapters describe an algorithm that was developed by the author for the generation of a controlled pose (i.e., frontal) view of a face under controlled illumination (i.e., frontal and of reasonable brightness) from an image that is unconstrained for pose and/or illumination. (In fact, any novel view can be generated.) This chapter considers the pose problem alone, and presents the face fitting algorithm and associated image processing operations. In this algorithm, the FaceGen 3D face model — described in chapter 2.3 — serves as the underlying model whose parameters are manipulated in order to improve its resemblance to the target face. Parameter manipulation, it will be shown, depends on the rules outlined in section 3.5 for estimating both the shape and texture (colour characteristics) of the target face. Successful shape and texture estimation in this scheme requires an appropriate pose estimation, presented in section 3.4, and operates on facial images that are processed as outlined in section 3.3. The required input to the algorithm is introduced

in section 3.1 and the initial pose estimation on the target image, including a pose correction step, is described in section 3.2.

3.1 Target Facial Feature Selection

The face fitting algorithm accepts as input a target facial image, which need not be controlled for pose, and the coordinates of three manually located facial features. The facial features selected depend on the pose of the target face. The preferred set of facial features are the outside corners of the eyes and the tip of the nose, as shown in figure 3.1b. Upon more extreme rotations in yaw, however, the outside corner of one of the eyes may become hidden. In these cases, the three facial features selected are the inside and outside corners of the visible eye and the tip of the nose, as shown in figures 3.1a and 3.1c. With pose estimation and alignment in three-dimensions based on these three feature points (as will be described in forthcoming sections), tests were conducted to determine the maximum range of yaw and pitch variation that would be acceptable as input. It was found that this scheme allows faces ranging in yaw between $\pm 79°$ and in pitch between $\pm 43°$ to be practically considered. Beyond this range, it becomes difficult or even impossible to reliably locate the required facial features.

3.2 Initial Pose Estimation

3.2.1 Correction for Roll

Manipulation of face pose in the FaceGen modeller comes via click and drag mouse movements with the mouse positioned over the face in the viewing window. While the

Figure 3.1: Target facial features used at different poses. [Original image source = IMM Face Database [40].]

modeller does allow for face rotation along three degrees of freedom (corresponding to rotations in yaw, pitch, and roll as previously defined in section 2.4), rotation in roll requires circular mouse movements [61] which could not be reliably automated. The preferred setting which offers improved reproducibility of face rotations limits face rotation to "pan" and "tilt" only, corresponding to rotations in yaw and pitch, respectively.

For this reason, it was decided to consider face *roll* — or, at least, the apparent effect of rotation in roll — to be a characteristic of the target image for which correction is required as a preprocessing step, with the extent of roll being calculated from the feature points. By rotating the target image such that the relative orientation of the facial feature points are consistent with that of the projected image of a FaceGen model in the absence of roll, the apparent effect of rotation in roll can be negated. While rotation in image space to correct for roll may also correspond to slight rotations in yaw and pitch (in three-dimensions) in addition to roll, we strive to correct for the apparent effect of roll in order to allow for the system to converge to a viable solution that is free from rotation in roll.

In the case where the feature points include the nose tip and outside corners of each eye (as in figure 3.1b), this correction is simply a matter of rotating the image to align the outside eye corners in terms of height in image space. (The assumption of facial symmetry with respect to the eyes which underlies this solution is, in general, a valid assumption based on anthropometric measurements of the face [17].)

Correcting for roll is those instances where the target face is rotated to a degree such that only the corners of a single eye along with the nose tip are selected as feature points, as in figures 3.1a and 3.1c, is more difficult. Upon locating and measuring the relative positions of the eye corners and nose tip across many different poses but in the absence of roll, though, it was found that the angle formed by the vector joining the nose tip to the outside eye corner could be expressed as a function of the relative locations of the feature points.

The *average* FaceGen face (that is, the face defined by average shape $\bar{\mathbf{S}}$ and average texture $\bar{\mathbf{T}}$) was rotated across the entire range of yaw and pitch values with each quantized into $1.8°$ increments. Upon imposing coloured markers at the locations of the nose tip and eye corners in the mean texture map underlying the FaceGen model, feature point locations could be reliably measured at each pose position. Defining $\mathbf{E}_o$ as the vector from the position of the nose tip to the outside eye corner and $\mathbf{E}_i$ as the vector joining the nose tip to the inside eye corner, a curve was fit via least squares regression to express the angle formed by the vector joining the nose tip to the outside eye corner $\angle\mathbf{E}_o$ as a function of ϵ, the angle between the eye vectors, where

$$\cos \epsilon = \frac{\langle \mathbf{E}_o, \mathbf{E}_i \rangle}{\|\mathbf{E}_o\| \cdot \|\mathbf{E}_i\|} \tag{3.1}$$

26

(with $\langle \mathbf{E}_o, \mathbf{E}_i \rangle$ designating the dot product), and the ratio of the vector norms,

$$r = \frac{\|\mathbf{E}_i\|}{\|\mathbf{E}_o\|}. \tag{3.2}$$

Specifically, following the procedure for curve fitting via least squares regression outlined in [16], the residual error $\mathbf{e}$ was minimized for the equation

$$\mathbf{y} = \mathbf{Zc} + \mathbf{e} \tag{3.3}$$

where the column vector $\mathbf{y}$ contained the measured values of the angle formed by the vector joining the nose tip to the outside eye corner $\angle\mathbf{E}_o$ at all poses, designated by indices 1 through n,

$$\mathbf{y} = [\angle\mathbf{E}_{o1} \ \angle\mathbf{E}_{o2} \ \ldots \ \angle\mathbf{E}_{on}]^\mathsf{T}, \tag{3.4}$$

where $\mathbf{Z}$ was a matrix containing the calculated values of the basis functions in ϵ and r that were heuristically determined to provide a good fit, columnized by function,

$$\mathbf{Z} = \begin{bmatrix} 1 & \cos\epsilon_1 & \sin\epsilon_1 & r_1 & r_1^2 \\ 1 & \cos\epsilon_2 & \sin\epsilon_2 & r_2 & r_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \cos\epsilon_n & \sin\epsilon_n & r_n & r_n^2 \end{bmatrix}, \tag{3.5}$$

and $\mathbf{c}$ was a column vector containing the unknown coefficients of the best-fit curve.

Best-fit curves were derived independently for cases where the left eye corners would be identified as feature points (as in figure 3.1a) and for cases where the right eye corners woul be used (as in figure 3.1c). For the left eye, feature point positions were recorded at 1.8° increments in yaw between -79.2° and 0° and in pitch between

27

-43.2° and +43.2° for 2205 (45 x 49) sets of data points. Similarly for the right eye, 2205 sets of data points were used with yaw ranging between 0° and +79.2° and pitch between -43.2° and +43.2°. Plots of the calculated angle between the nose tip and the outside corner of the eye in question for recorded feature point positions are shown in figures 3.2a and 3.2b for the left eye and right eye, respectively. These plotted values are those forming the vector $\mathbf{y}$ defined in equation 3.4 for derivation of the best-fit curves.

The selection of basis functions to be used for curve fitting involved evaluation of 10 unique combinations of different functions in ϵ, r, and some other measurements derived from the feature point positions. The basis functions specified in equation 3.5 were deemed most suitable in terms of the combination of accuracy and simplicity. These functions yield best-fit curves of the form

$$\widehat{\angle \mathbf{E}_o} = c_1 + c_2 \cos \epsilon + c_3 \sin \epsilon + c_4 r + c_5 r^2 \tag{3.6}$$

which were found independently for the left eye and the right eye. For the *average* FaceGen face that was used in the analysis, the modelled data predicted by the best-fit curves exhibited strong correlation with the observed data, with coefficients of determination R^2 equal to 0.980 for both cases, left eye and right eye.

So, for a given target face image, the angle between the vectors joining the nose tip to the eye corners ϵ and the ratio of these vector norms r are calculated based on the positions of the manually located feature points. These values are used in equation 3.6 to yield an estimate of what would be the angle of the vector joining the nose tip to the outside eye corner for an average face *in the absence of any roll* with the same characteristic ϵ and r. This estimate for the average face is subsequently

28
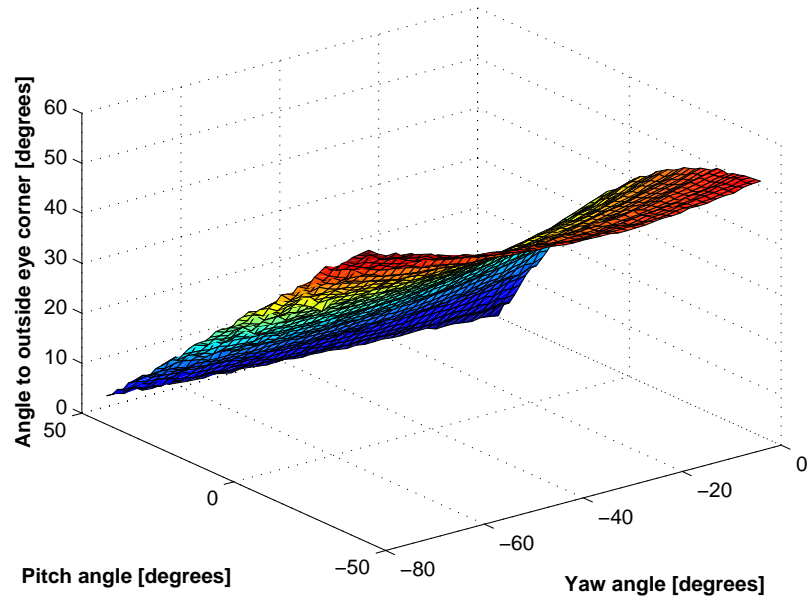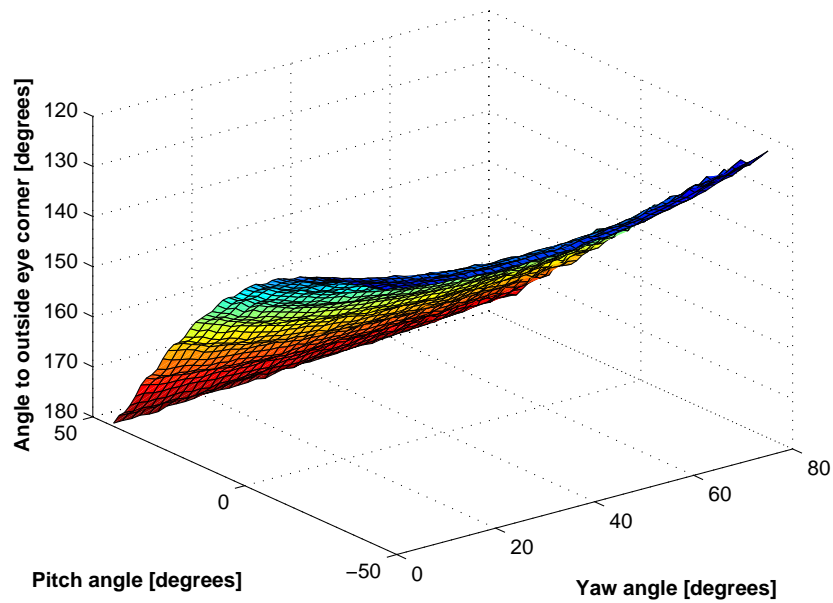
Figure 3.2: Measured angle between nose tip and outside eye corner of the average FaceGen face as a function of yaw and pitch for left eye (a) for right eye (b).

used for the target face. In order to correct for roll, the target image is rotated in image space by the difference between the measured outside eye angle $\angle\mathbf{E}_o$ and the modelled outside eye angle $\widehat{\angle\mathbf{E}_o}$. (Feature point positions are transformed into polar coordinates with the origin at the image centre, which is invariant under rotation, and the angular displacement is modified in accordance with the image rotation.) It is this pose corrected target image that is processed for comparison with the face model. Figure 3.3 shows a target image that exhibited a fairly substantial rotation in roll along with its roll-corrected equivalent. While in this instance roll correction was effective, it generally proved to be inconsistent as a preprocessing operation, as will be discussed in more detail in section 5.1.1.

3.2.2 Pose Estimation

Initial estimation of the pose (in yaw and pitch) of the roll-corrected target face image is likewise based on analysis of an average face. Using the feature point locations on the average face over the entire range of yaw (ψ) and pitch (θ), quantized into 1.8° increments, the (ψ,θ) pose estimate for a face in which both outside eye corners are used as feature points (as in figure 3.1b) is that which minimizes the cost function

$$\min_{\psi,\theta}[|\angle\mathbf{E}_{R,mod}|_{\psi,\theta} - \angle\mathbf{E}_{R,tgt}| + |\angle\mathbf{E}_{L,mod}|_{\psi,\theta} - \angle\mathbf{E}_{L,tgt}|] \qquad (3.7)$$

where the subscripts R and L designate right and left eyes, respectively, and *mod* and *tgt* designate the model and target face images, respectively. Similarly, the (ψ,θ) pose estimate for a face in which only one eye is used to specify feature points (as in

(a)
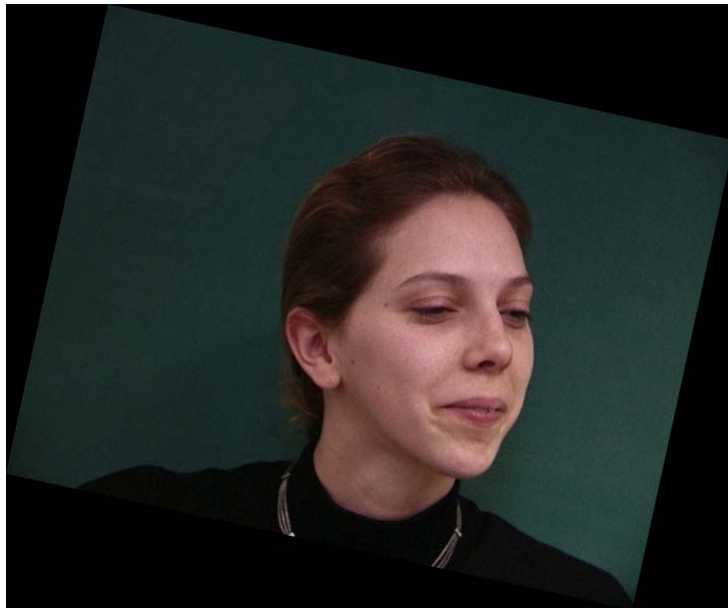
(b)

Figure 3.3: Example target image (a) and its roll-corrected equivalent (b). [Original image source = IMM Face Database [40].]

figures 3.1a and 3.1c) is that which minimizes the following cost function:

$$\min_{\psi,\theta}[|\angle\mathbf{E}_{o,mod}\big|_{\psi,\theta} - \angle\mathbf{E}_{o,tgt}| + |r_{mod}\big|_{\psi,\theta} - r_{tgt}|]. \tag{3.8}$$

Although subject to pose refinement as will be discussed in section 3.4, the FaceGen face model is rotated to this initial pose estimate for preliminary comparison with the target image.

3.3 Facial Image Processing

3.3.1 Pose Dependent Face Segmentation

The presence of outlying pixels in a facial image is a major source of fitting inaccuracies [56]. Outlying pixels are pixels inside the face area of an image whose value cannot be predicted by the face model without having a substantially negative impact on the model fitting to the rest of the face. Such pixels may result from the presence of glasses, facial hair, or some other occluding feature not normally modelled by the face model.

In order to mitigate the impact of any outlying pixels in the target image, and specifically the impact of hair which is not modelled by the FaceGen face modeller, automatic segmentation of the face area from the target image was incorporated into the algorithm. Pose dependent segmentation or crop parameters were found such that for any (ψ,θ) pose estimate, the target facial image would be cropped from just above the eyebrows to just below the chin in height and just bounding the face region in width. Once again, suitable crop parameters — upper-left corner set as a pose-dependent offset from the nose tip position along with rectangle width and

32

height — were found through analysis of the average FaceGen face under rotation. In consideration of the target face image, it was necessary to scale these crop parameters by an appropriate factor. This scale factor, s, is derived from the feature points as described in [68] for the least-squared fit of three pairs of corresponding points. This difference here is that rather than scaling the distance between all three points and the centroid of the triangle formed by these points, the nose tip is fixed as a point of reference for cropping and for image registration. So, the scale factor is calculated only from the vectors joining this fixed nose tip to the eye corners,

$$
s = \begin{cases} \frac{\langle \mathbf{E}_{R,mod}, \mathbf{E}_{R,tgt} \rangle + \langle \mathbf{E}_{L,mod}, \mathbf{E}_{L,tgt} \rangle}{\|\mathbf{E}_{R,mod}\|^2 + \|\mathbf{E}_{L,mod}\|^2} & \text{for frontal view,} \\[3mm] \frac{\langle \mathbf{E}_{o,mod}, \mathbf{E}_{o,tgt} \rangle + \langle \mathbf{E}_{i,mod}, \mathbf{E}_{i,tgt} \rangle}{\|\mathbf{E}_{o,mod}\|^2 + \|\mathbf{E}_{i,mod}\|^2} & \text{for side view.} \end{cases}
\tag{3.9}
$$

The use of this scale factor in cropping of the face region from the target image is illustrated in figure 3.4. A roll-corrected (and partially cropped for convenience of viewing) target image is presented in figure 3.4a along with the vectors that join the nose tip to the inside and outside eye corners, $\mathbf{E}_{i,tgt}$ and $\mathbf{E}_{o,tgt}$, respectively. The corresponding nose tip to eye corner vectors for the average FaceGen model face, $\mathbf{E}_{i,mod}$ and $\mathbf{E}_{o,mod}$, at the specific pose parameters estimated as outlined in section 3.2.2, are shown in figure 3.4b on the rotated FaceGen model image. The crop parameters (*x*- and *y-offsets* from the nose tip to the upper-left corner of a rectangle of *width* and *height*) have been predetermined for the average face at the estimated pose as shown in figure 3.4c. These crop parameters are scaled by the scale factor s as calculated in equation 3.9 to yield the cropped target image of figure 3.4d.
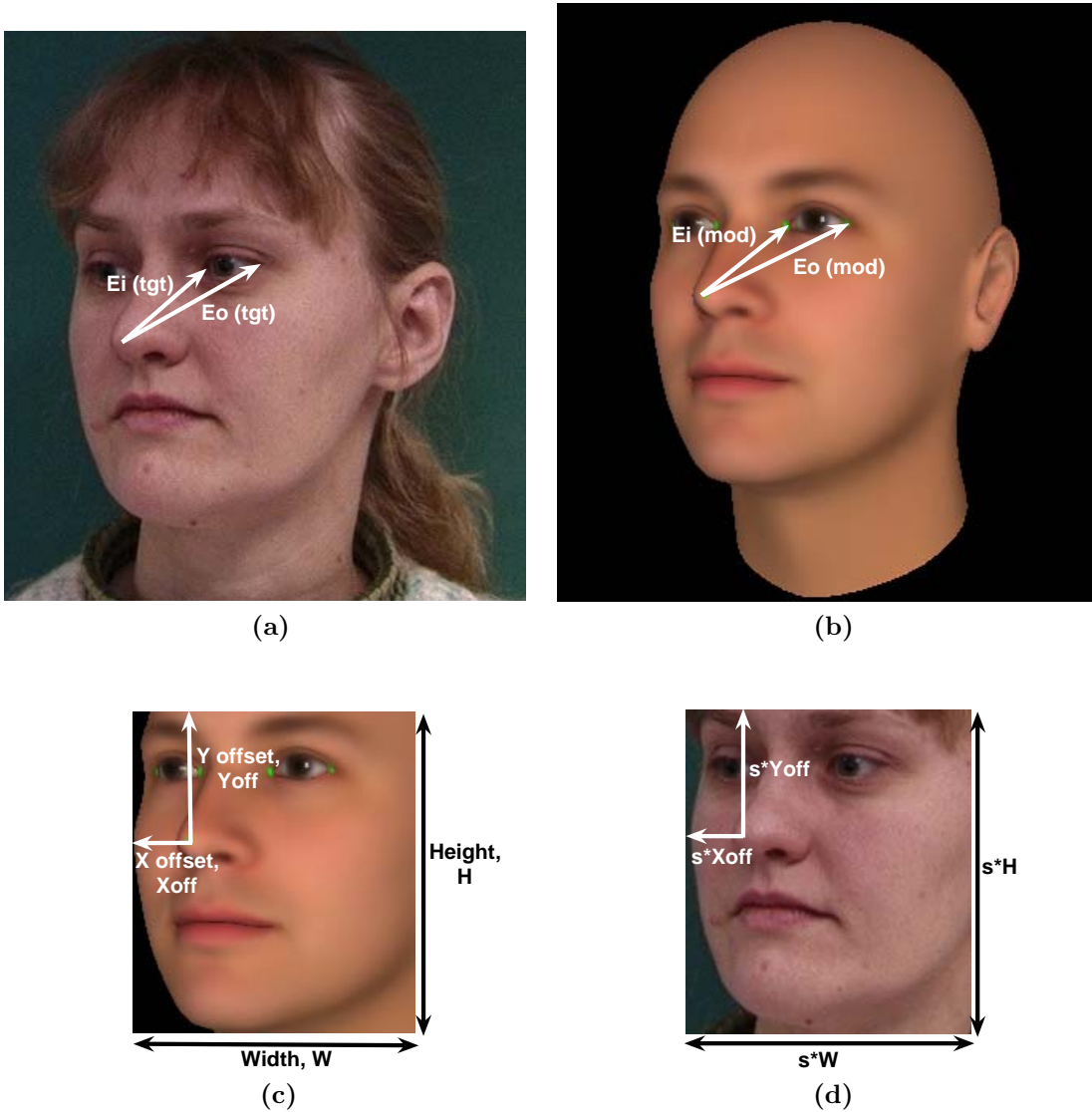
Figure 3.4: Use of scale factor for pose dependent face segmentation of target image. Nose tip to eye corner vectors $\mathbf{E}_{i,tgt}$ and $\mathbf{E}_{o,tgt}$ in target image (a) are used along with those for the average FaceGen face at the estimated pose (b) to calculate scale factor s. This value scales the predefined crop parameters for the average face, as shown in (c), to yield the crop parameters for the target face, as shown in (d).

3.3.2   Background Equalization

While consisting primarily of the face only, the segmented image does contain some background regions. In order to minimize the contribution of these background regions to any image comparisons, and thus minimize any errors the background contribution might introduce, it was decided to remove the background entirely by equalizing background pixel values in the target image with the (fixed value) background pixel values in the images from the FaceGen modeller. This requires effective segmentation of the face region from non-face regions in the target image. The approach taken to achieve this was colour-based skin segmentation.

The fact that it is possible to fairly robustly segment skin from its surroundings in an image is not entirely intuitive. After all, we often describe people's appearance based on their skin "colour". It would follow, then, that a colour corresponding to a given skin tone characteristic of a particular race would be unlikely to correspond to a skin tone characteristic of some other race. But this is not necessarily the case. Indeed, as noted in [63], effective object (in this case, skin) segmentation by colour is possible provided that consideration is given to selection of a suitable colour space exhibiting a reasonable object colour distribution across the selected colour space.

Skin Detection in YCbCr Colour Space

In order to test different colour spaces with respect to their suitability for colour-based skin segmentation, colour histograms of skin pixels were analyzed. Starting from a number of images taken under similar camera and lighting conditions, 20 by 30 pixel image swatches were taken from the skin regions of 19 different people. These 19 people provided a small representative sample of different skin tones characteristic of different races. The swatches were combined and the distribution of the resulting
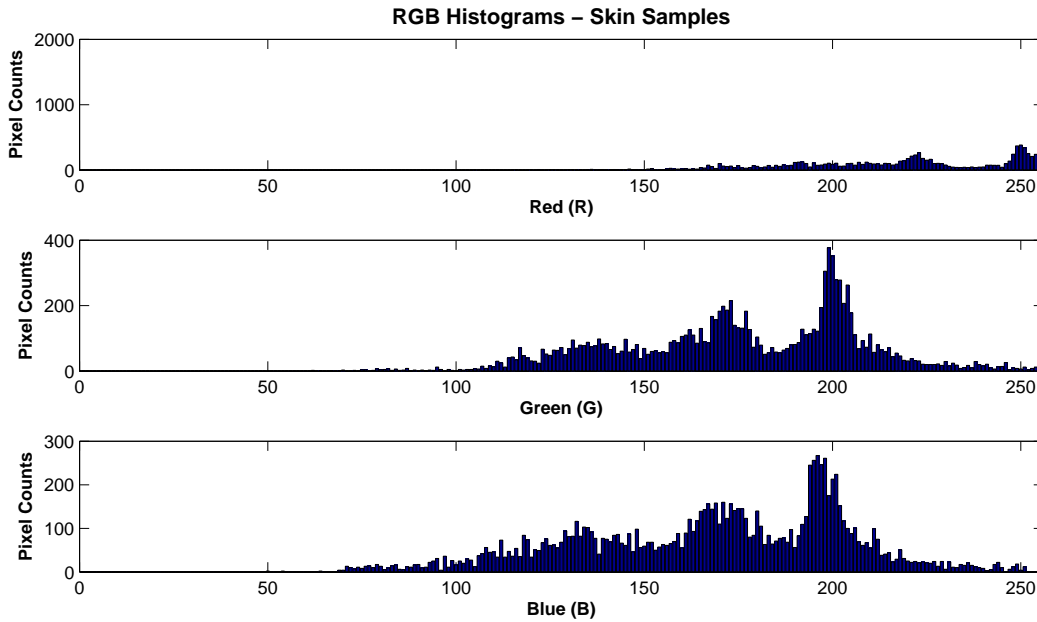
Figure 3.5: Distribution of skin pixels in RGB colour space.

11,400 pixels across various colour components were compared with a goal of identifying a colour space that would exhibit strong clustering of skin colours and thus enable segmentation based on application of simple thresholds.

For instance, figure 3.5 shows the distribution of skin pixels across the R, G, and B primaries of the RGB colour space that was the starting point for this analysis. Clearly, skin colour is fairly broadly distributed across each component making it difficult to characterize skin colour in this space. The problem is that each of R, G, and B represent not only colour but also brightness, and skin colour, as it turns out, can be effectively characterized by chromatic colours in the absence of any brightness component [71].

A number of other papers such as [71], [18], and [6] describe a normalization process applied to an RGB image representation in order to separate the chromatic components from a brightness component. Similar results are achieved upon conversion to a colour space described in [19] — the YCbCr colour space. YCbCr is a
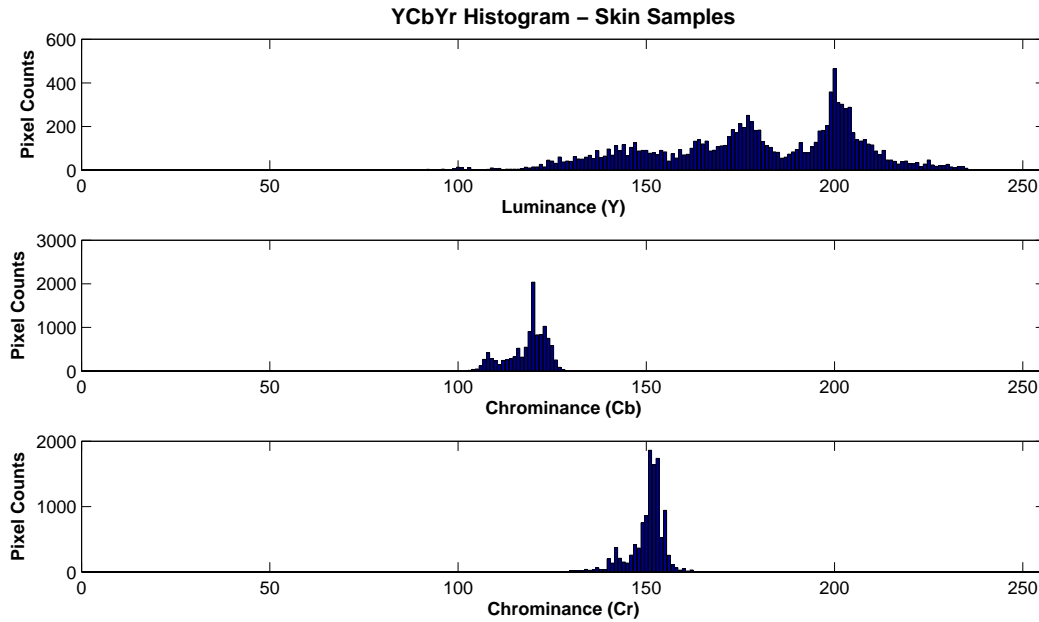
Figure 3.6: Distribution of skin pixels in YCbCr colour space.

luma-colour-difference space in which Y represents the luminance or brightness component and Cb and Cr represent two chrominance components. (Cb is the difference between blue and some reference value. Cr is the difference between red and some reference value [51].) Histograms of the pixel values of the skin samples in the YCbCr space are given in figure 3.6. Given the strong clustering of skin pixel values in the Cb and Cr components, this colour space was selected for the segmentation operation.

Skin Segmentation

With the segmented target image converted from its RGB representation to a YCbCr representation, any pixels having Cb and Cr chrominance values between the minimum and maximum thresholds implied by the histograms shown in figure 3.6 were considered likely to represent skin and were assigned a value of 1. Conversely, any pixels with chrominance values lying outside these limits were assigned a value of 0. The result, shown in figure 3.7b for the sample target image under consideration, is a

37
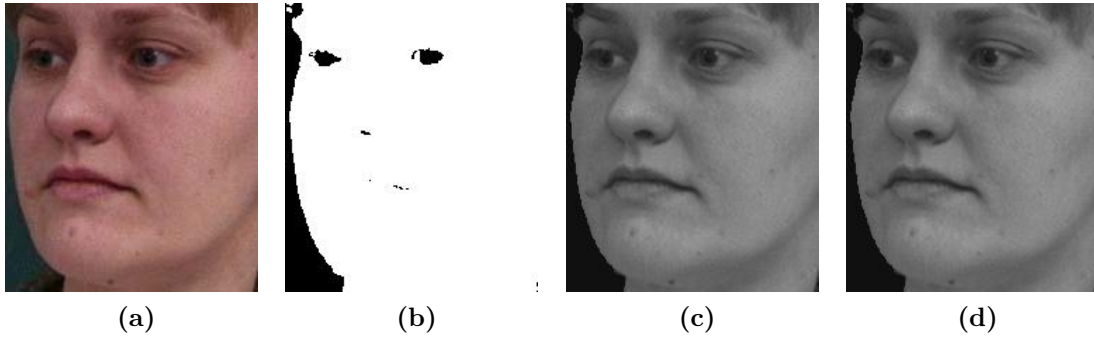
Figure 3.7: Pose-corrected and cropped target image (a); with detected skin regions (b); used for background equalization (c). The intensity image is contrast-stretched (d).

binary image in which the white regions correspond, at least roughly, to skin regions in the original image. Upon accepting as face only the connected white region that is largest in area — a valid assumption given that the target image has already been cropped to include predominately the target face — and filling all holes within this face region, all pixels not contained within this region were set to black, as shown in figure 3.7c, matching the background in the FaceGen images with which the target image is to be compared.

It should be noted that, to be thorough, the distribution of background pixels in the YCbCr colour space should also be analyzed. Since we are generalizing across all potential backgrounds, though, this has not been presented. In the event that the target image appears in front of a flesh-coloured background, this skin segmentation step and thus the background equalization operation will fail.

3.3.3 Contrast Enhancement

Figure 3.7d shows the results of the final target image processing operation. Contrast in the background equalized intensity image is enhanced via an intensity transformation function. The lower intensity bound is mapped to black (0) while the upper

bound is mapped to white (1) and intermediate values are mapped linearly in between. So, for an intensity image I having a minimum intensity level LOW and maximum intensity level $HIGH$, pixel intensity values at each pixel location (x, y) are transformed as follows:

$$T[I(x,y)] = \frac{I(x,y) - LOW}{HIGH - LOW}.$$ (3.10)

This contrast-stretching operation serves to enhance any differences between target and model images, thus giving greater resolution to the model parameter fitting procedure, to be discussed in section 3.5. The key benefit is that, given that the backgrounds in both target and model images are equalized (as per the operation described in 3.3.2) and are darker than any face, contrast-stretching necessarily "brightens" the face. This serves to increase pixel value differences between face and background pixels such that non-matching pixels in the target-to-model comparison are even further penalized in the model parameter fitting procedure. It was also found that contrast enhancement as a preprocessing step played a beneficial role upon introduction of unconstrained illumination conditions, to be discussed in chapter 4, by providing some normalization across variation in ambient lighting.

3.4   Pose Refinement

The (ψ, θ) pose estimate is based on the characteristic feature point locations of an *average* face. This provides a good initial estimate given our lack of knowledge of the 3D structure of the target face, but it is desirable to search in the vicinity of this initial estimate for an improved pose match. Prior to any estimation of shape

and texture parameters, then, the pose is refined using an approach similar to that described in [59].

Edges are extracted in both the target image and the image of the FaceGen model, rotated to the initial pose estimate. Several different edge detection approaches [38] were attempted with varying success. For the FaceGen model image, the Canny edge detector [15] was found to work particularly well. For target images, the success of any edge detection scheme depends largely on the imaging conditions and image characteristics. As such, when encountered with a target image from a new source, some experimentation will be required to identify an appropriate edge detection approach and appropriate associated parameters, if applicable. Typical detected edges for target and FaceGen model images are shown in figures 3.8a and 3.8b, respectively. For each pixel in both the target and model binary edge maps (EM), a distance transform (DT) — which assigns to each pixel the Euclidean distance to the nearest non-zero pixel (i.e., a detected edge) in the edge map — is computed. A cost function, $Dscore$, which measures the disparity between the two edge maps over all pixels (i, j) is computed as follows [38]:

$$Dscore = \frac{\sum_{(i,j)\in A_{EM,mod}} DT_{tgt}(i,j)}{N_{mod}} + \frac{\sum_{(i,j)\in A_{EM,tgt}} DT_{mod}(i,j)}{N_{tgt}} \qquad (3.11)$$

where $A_{EM} \triangleq \{(i,j) : EM(i,j) = 1\}$ and N is the cardinality of set A_{EM}, that is, the total number of non-zero pixels in the edge map EM.

$Dscore$ is computed at the initial pose estimate and at slight increases and decreases in both yaw and pitch. If a pose is found which provides a lower $Dscore$, it is set as the new pose estimate and a search for improvement in the new local vicinity is repeated. This continues until no improvement is found. The effectiveness of this

40
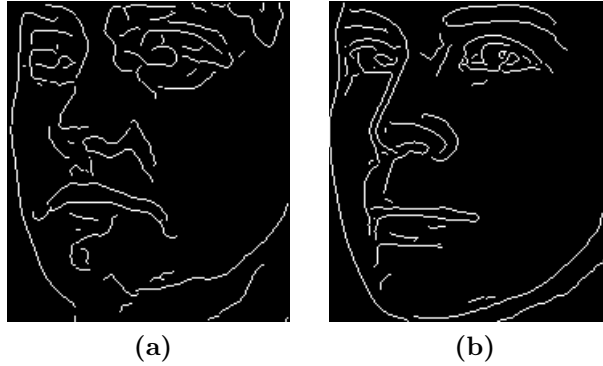
Figure 3.8: Binary edge maps for target (a) and pose estimated average FaceGen model (b) images.

pose refinement operation is discussed in section 5.1.2.

3.5 Face Image Fitting

Having an image of the FaceGen model that is at least close in pose to the target image, the problem to be addressed is as follows: what combination of FaceGen model parameters is best able to describe the target image? As outlined in section 2.2, the 3D morphable model allows any face to be generated as a linear combination of N_S shape and N_T texture principal components added to the average face shape $\bar{\mathbf{S}}$ and texture $\bar{\mathbf{T}}$ as per equation 2.4 which is reproduced here for convenience,

$$\mathbf{S} = \bar{\mathbf{S}} + \sum_{i=1}^{N_S} \alpha_i \mathbf{s}_i, \quad \mathbf{T} = \bar{\mathbf{T}} + \sum_{i=1}^{N_T} \beta_i \mathbf{t}_i. \tag{3.12}$$

The face-fitting algorithm that was developed for the purposes of this thesis follows the framework first introduced by Blanz and Vetter in [11]. The morphable model shape coefficients α_i and the texture coefficients β_i are optimized simultaneously with several rendering parameters — the pose parameters ψ and θ as well as illumination

parameters which are discussed in more detail in chapter 4 — so as to synthetically reconstruct the target image.

Fundamentally, the goal of the face-fitting algorithm is to minimize the sum of square differences over all colour channels and all pixels between the input image and the image rendered by the model [11–13]. A cost function for image differencing is defined as follows:

$$C_I = \sum_{x,y} \|\mathbf{I}_{tgt}(x,y) - \mathbf{I}_{mod}(x,y)\|^2, \tag{3.13}$$

where $\mathbf{I}_{tgt}(x,y)$ and $\mathbf{I}_{mod}(x,y)$ represent the pixel values over all colour channels for the target and model images, respectively, concatenated into a column vector. Rather than the using the RGB colour channels as described in [11,12] or a transformed RGB colour space as described in [13], here we make use of the YCbCr colour space since the conversion to this space was already carried out for skin segmentation in the background equalization step described in section 3.3.2, and this enabled use of the contrast enhanced Y or intensity component as per the operation described in section 3.3.3. Stretching the limits of the intensity component was found to provide greater resolution to the model fitting procedure, particularly under low ambient lighting conditions as will be addressed in the chapter that follows. A given target image would thus be expressed as

$$\mathbf{I}_{tgt}(x,y) = [I_Y(x,y), I_{Cb}(x,y), I_{Cr}(x,y)]^\mathsf{T}, \tag{3.14}$$

over all pixel locations (x,y) and with the $I_Y(x,y)$ component being contrast enhanced.

It has been shown that using only facial colour information (i.e. pixel values) to recover shape correspondence between target and model is not optimal as the cost

function to be minimized is highly non-convex and exhibits many local minima [55]. It has been demonstrated that using only the target image itself — and not any features thereof — to recover optimal shape and texture will result, in many cases, in convergence at a local minimum far from the global one, yielding an unrealistic face [58]. At the same time, image features have been shown to contribute to optimal parameter estimation. Most notably, the use of *edges* facilitates the finding of a more accurate estimate of the 3D shape [55, 58]. With target and model image edge maps already used for pose refinement prior to parameter estimation (section 3.4), here we define an edge cost function which is similar to equation 3.11 but considers only the distance from points in the target edge map to detected edges in the model image and not the converse,

$$C_E = \sum_{(i,j) \in A_{EM,tgt}} DT_{mod}(i,j).$$
(3.15)

Since the number of pixel locations representing detected edges in the model image, EM_{mod}, can be variable while the number of pixel locations representing detected edges in the target image, EM_{tgt}, is fixed, limiting the analysis to consider target-to-model edge distance (and not model-to-target) fixes the length of the vector formed by $DT_{mod}(i,j)$ where (i,j) are all pixels in $A_{EM,tgt}$. Image edges provide information about the 2D shape independent of the image texture, and hence the cost function used to fit the edge features provides a more direct constraint on shape and pose parameters [55].

A cost function incorporating prior probabilities for the shape and texture parameters is also used to penalize statistically unlikely face shape and texture parameters. According to the FaceGen Modeller documentation [60] the shape and texture coefficients (which correspond to α_i and β_i, respectively, in the morphable

model framework) are — to borrow the FaceGen terminology — principal component *modes* representing differences from the mean for that particular shape or texture parameter. These modes are considered to be statistically independent and normally distributed [60]. The model prior cost function, then, is defined as

$$C_P = \sum_{i=1}^{N_S} \frac{\alpha_i^2}{\sigma_{S,i}^2} + \sum_{i=1}^{N_T} \frac{\beta_i^2}{\sigma_{T,i}^2}, \tag{3.16}$$

where $\sigma_{S,i}$ gives the standard deviation for the normally distributed shape parameter i and $\sigma_{T,i}$ gives the standard deviation for the normally distributed texture parameter i. While not explicitly available in the model, these standard deviation values were estimated through analysis of the individual shape and texture coefficient distributions over 99 generated random faces, under the assumption that random face generation in the FaceGen modeller is based on the statistical distribution of faces over "face space", as per the documentation [60]. (Random face generation — based on selecting random coefficients for each mode according to a standard normal distribution [60] — is a function available in the FaceGen modeller.) Alternatively, parameter perturbation vectors can be derived for each of the unique face shape and texture slider controls in the modeller. Each slider control allows face characteristics to be varied between plus and minus 10 standard deviations of the distribution for that characteristic [61]. So by calculating a face parameter vector by multiplying a matrix containing the parameter perturbation vectors by a vector of slider control coefficients, the coefficient values can be used as $\sigma_{S,i}$ and $\sigma_{T,i}$ in the model prior cost function calculation. The difficulty with this approach is that the controls are not completely independent of each other, but rather adjusting one slider usually affects several others [61].

It is demonstrated in [58] that if the image features under comparison are independent and extracted by a deterministic algorithm, then the overall cost function is a linear combination of the cost functions of each feature taken separately. Our cost function, to be minimized over the shape parameters α_i, the texture parameters β_i, and the model image rendering parameters, is thus given by

$$C = w_I C_I + w_E C_E + w_P C_P \tag{3.17}$$

with the w's being weighting factors. Appropriate weights were heuristically determined through experimentation to provide satisfactory results. In particular, it should be noted that an excessively high weight on the model prior (w_P) will overly constrain the result to resemble the average face. At the same time, if w_P is too low, the face-fitting algorithm is more likely to produce a result that — although fitting the 2D target image — bears some statistically unlikely 3D shape characteristics.

The core of the face-fitting procedure for shape and texture parameter estimation is minimization of the cost function 3.17. Researchers have suggested the use of a number of different optimization algorithms for the solution of this problem, including *stochastic gradient descent* [11], *stochastic Newton optimization* [10, 13], and the *inverse compositional image alignment* algorithm [57]. Here, the more recently implemented approach of Romdhani *et al.* [55, 58] is adopted with the use of the Levenberg-Marquardt optimization algorithm.

3.5.1   Levenberg-Marquardt Method

Based on previous work by Levenberg [33], Marquardt [37] proposed an optimization approach that can be considered a hybrid between steepest descent and the Gauss-

Newton method [35]. This *Levenberg-Marquardt Method* works very well in practice and is popular for use in nonlinear least-squares routines given its ability to converge promptly from a wide range of initial guesses [52].

Pseudocode for the Levenberg-Marquardt algorithm is given in figure 3.9. The code describes a vector function f which maps a *parameter vector* $\mathbf{p} \in \mathbb{R}^m$ to an estimated *measurement vector* $\hat{\mathbf{x}} = f(\mathbf{p}), \hat{\mathbf{x}} \in \mathbb{R}^n$ where $n \geq m$ [35]. Starting with a measurement vector $\mathbf{x} \in \mathbb{R}^n$ and an initial parameter estimate $\mathbf{p}_0$, the algorithm yields an output parameter vector $\mathbf{p}^*$ that is the solution to the minimization of $\|\mathbf{x} - f(\mathbf{p})\|^2$. In our context, the vector $\mathbf{x}$ contains the target image pixel values concatenated with zeros of length N_{tgt} (the cardinality of set $A_{EM,tgt}$ containing all edge pixels in the target edge map) and zeros of length $N_S + N_T$. Meanwhile, parameter vectors $\mathbf{p}$ contain the N_S shape coefficients α_i, the N_T texture coefficients β_i, and the model image rendering parameter values. So, the vector $f(\mathbf{p})$ contains the image pixel values, minimum distances from target image edge pixels to model image edges, and model prior penalties for all shape and texture parameters for the given parameter set $\mathbf{p}$, as per all the elements of the cost function of equation 3.17.

In the code outlined in figure 3.9, $\mathbf{J}$ denotes the *Jacobian* matrix containing the first partial derivatives of the function components, where $\mathbf{J} \in \mathbb{R}^{n \times m}$ and

$$\mathbf{J}_{ij} = \frac{\partial f_i(\mathbf{p})}{\partial \mathbf{p}_j} \approx \frac{\Delta f_i(\mathbf{p})}{\Delta \mathbf{p}_j}. \tag{3.18}$$

That is, the Jacobian is populated with partial derivatives which are numerically calculated using small perturbations under the simplifying assumption that f is linear in the region of $\mathbf{p}$ for small changes in $\mathbf{p}$.

The distinguishing feature of the Levenberg-Marquardt algorithm is the *damping*

Set: $k := 0, \nu := 2, \mathbf{p} := \mathbf{p}_0$

Given: tol_1, tol_2 (tolerances); k_{max} (maximum iterations); τ

Algorithm:

$\mathbf{A} := \mathbf{J}^\mathsf{T}\mathbf{J};$ $\epsilon_\mathbf{p} := \mathbf{x} - f(\mathbf{p});$ $\mathbf{g} := \mathbf{J}^\mathsf{T}\epsilon_\mathbf{p};$

$\mu := \tau * \max_{i=1,\ldots,m}(A_{ii});$

while $(\|\mathbf{g}\|_\infty \geq tol_1$ & $(k < k_{max})$

    $k := k + 1;$

    $\boxed{\text{Solve } (\mathbf{A} + \mu\mathbf{I})\delta_\mathbf{p} = \mathbf{g};}$

    if $(\|\delta_\mathbf{p}\| \leq tol_2\|\mathbf{p}\|)$

        break

    else

        $\mathbf{p}_{new} := \mathbf{p} + \delta_\mathbf{p};$

        $\rho := \frac{\|\epsilon_\mathbf{p}\|^2 - \|\mathbf{x} - f(\mathbf{p}_{new})\|^2}{\delta_\mathbf{p}^\mathsf{T}(\mu\delta_\mathbf{p} + \mathbf{g})};$

        if $\rho > 0$ [step improves solution]

            $\mathbf{p} = \mathbf{p}_{new};$

            $\mathbf{A} := \mathbf{J}^\mathsf{T}\mathbf{J};$

            $\epsilon_\mathbf{p} := \mathbf{x} - f(\mathbf{p});$

            $\mathbf{g} := \mathbf{J}^\mathsf{T}\epsilon_\mathbf{p};$

            $\mu := \mu * \max(\frac{1}{3}, 1 - (2\rho - 1)^3); \nu = 2;$

        else

            $\mu := \mu * \nu; \nu := 2 * \nu;$

        endif

    endif

endwhile

$\mathbf{p}^* := \mathbf{p} ;$

Figure 3.9: Levenberg-Marquardt algorithm pseudocode, adapted from [36] and [35].

term denoted by μ in figure 3.9. If μ is large, the algorithm behaves like the method of steepest descent, updating with relatively small steps in the steepest descent direction. Conversely, as μ becomes smaller, the algorithm performs more like the Gauss-Newton method and exhibits fast convergence [35]. At each iteration, if the updated parameter vector results in a reduction in error and thus the gain ratio (ρ in figure 3.9) is positive, the update is accepted and the process repeats with a decreased damping term μ. Otherwise, the damping term is increased with the aim of both getting closer to the descent direction *and* reducing the step size [36]. The equations are subsequently resolved and the process iterates until an updated parameter vector that does decrease the error is found or any of the stopping conditions are met [35].

Chapter 4

Handling Unconstrained Illumination

Given the innate ability of humans to recognize faces and the importance of this task in our daily lives, it is perhaps not surprising that the neural processing of faces has been widely studied in recent years [41, 42]. Various studies have shown that, when confronted with unfamiliar faces, even humans find the task of recognizing faces over changes in illumination to be difficult [14, 26]. Computer-based face recognition systems struggle similarly with the task of recognizing faces when illumination conditions are not controlled. Although, when it comes to matching pairs of face images taken under different illumination conditions, state-of-the-art face recognition algorithms compete favourably with humans and the best of these even surpass human matching performance [43], illumination variation continues to evoke a significant deterioration in algorithm performance. Even with improvements over recent years, this fact was made clear in the recent Face Recognition Grand Challenge (FRGC) [46] upon comparison of matching performance for illumination-controlled and -uncontrolled face

image pairs. FRGC results, reported in [45] (and summarized in [43]), show that algorithms participating in the experiment testing recognition from a single frontal image taken under *controlled* illumination (experiment 1 described in [46]) achieved a median verification rate of 0.91 at a false acceptance rate of 0.001. In contrast, those algorithms participating in the experiment testing recognition from a single frontal image taken under *uncontrolled* illumination (experiment 4 in [46]) achieved a median verification rate of only 0.42 at the same false acceptance rate.

This chapter builds on the algorithm for 3D modelling of a face under unconstrained pose that was described in chapter 3 by introducing unconstrained illumination as another variable to be considered. In section 4.1 the 3D morphable model approach to modelling illumination, and specifically the approach taken by the FaceGen face modeller, is briefly described. The illumination parameters that were introduced to the algorithm as variables in the optimization problem are described in section 4.2, while the scheme developed for initial estimation of these parameters is described in section 4.3. Finally, in section 4.4, the face modelling system developed for this thesis is considered in its entirety as a tool for synthesizing a novel view of a face — most usefully a frontally illuminated frontal pose — from an image that is unconstrained for both pose and illumination.

4.1   3DMM Approach to Modelling Illumination

The 3D morphable model underlies the face-fitting algorithm, so while a model was not developed for this thesis, it is worthwhile mentioning how it accounts for variation in illumination. The elegance of the 3D model is that, like the parameters governing the orientation of the face, parameters governing illumination conditions are free

variables subject to optimization [11]. The first implementations of the morphable model approach [11, 12] included only the intensities and colours of ambient and parallel light as optimization variables while other illumination parameters such as the direction of that parallel light remained fixed to values estimated by the user. More recent publications [10, 13, 55] describe implementations in which the direction of the parallel light (i.e., the position of a single directed light source relative to the face) is also included as an optimization variable.

In the morphable model framework, illumination is not restricted to Lambertian reflection, but also takes into account specular reflections and cast shadows, which strongly influence the apparent texture of the face [13]. In fact, most implementations describe the use of the Phong illumination model [50] to approximate the diffuse and specular reflection of the face surface. It has also been proposed that more photo-realistic images could be achieved over a wider range of illumination conditions, including multiple sources of illumination, by integrating spherical harmonics into the morphable model framework [72], thus introducing spherical harmonic basis parameters as another set of optimization variables.

4.1.1 Specifics of the FaceGen Modeller Approach

The FaceGen modeller similarly accounts for specular reflections and cast shadows. Diffuse shading can be applied in one of three modes: Phong, Gouraud, or Flat based on the respective illumination models. The default setting, and the one used in image rendering for this thesis, is the Gouraud illumination model [20]. In the FaceGen environment, Gouraud shading gives the face a sufficiently smoothly curved appearance while allowing the model to render faces faster and with fewer artefacts than the Phong shading mode [61].

Settings in the modeller control various illumination options, which are described in [61]. The ambient brightness component illuminates all surfaces equally, regardless of the orientation of the face. Up to three directed light sources can also be enabled, with each having adjustable brightness and incoming direction. These light sources can be considered to be capable of rotation about the face, and the light direction is thus specified by the position of the light in azimuth and elevation. And unlike the generic morphable model approach, these lights are strictly white.

4.2   Illumination Parameters

The face image fitting algorithm described in section 3.5 optimizes the face model shape coefficients (α_i) and texture coefficients (β_i) along with several model image rendering parameters. So far, only the rendering parameters for pose in yaw (ψ) and pitch (θ) have been described. We now introduce a set of illumination parameters that are also subject to optimization.

The algorithm also accounts for the brightness of ambient light, L_{amb}, and the brightness of a single directed light, L_{dir}. This directed light can rotate freely about the face with its angular position relative to the origin (i.e., directly in front of what would be a frontal face view) being specified in azimuth, λ_{az}, and elevation, λ_{el}. These illumination components are represented in figure 4.1 in images of a non-frontal face from the front, top, and side vantage points.

While these illumination parameters are subject to optimization in the face-fitting algorithm, such that the illumination conditions of the target image are estimated along with the orientation and the 3D shape and texture coefficients of the target face, initial estimates of these illumination conditions are required as a starting point [11].
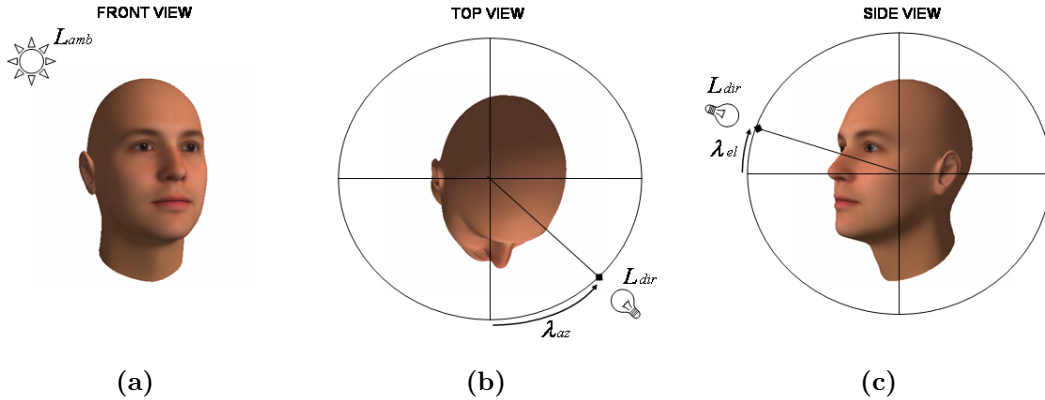
Figure 4.1: Front view of a face at a non-frontal pose (a) along with top (b) and side (c) projections showing the illumination parameters L_{amb}, L_{dir}, λ_{az}, and λ_{el}.

4.3   Illumination Parameter Estimation

It was found that for all images tested (as per the results to be given in chapter 5), the algorithm was able to converge to an illumination solution starting from "average" initial estimate values for both the ambient and directed light brightness settings, L_{amb} and L_{dir}. This average setting was represented by the FaceGen modeller default with 60% of overall brightness coming from the single directed light and 40% coming from ambient illumination [61]. Even for target images exhibiting rather poor illumination, the search area for illumination brightness was sufficiently small that the algorithm was able to reach a good estimate starting from these average initial values.

Even so, the contrast-stretching operation described in section 3.3.3 was found to mitigate to an extent the effect of any poor estimate in the ambient brightness L_{amb}. The top row of figure 4.2 shows a series of intensity images of faces at a fixed pose and with fixed direct lighting under variable ambient light brightness. The bottom row of this figure gives the contrast-stretched versions of these same faces. Visibly, the faces in the bottom row appear to be more similar than those in the top row,
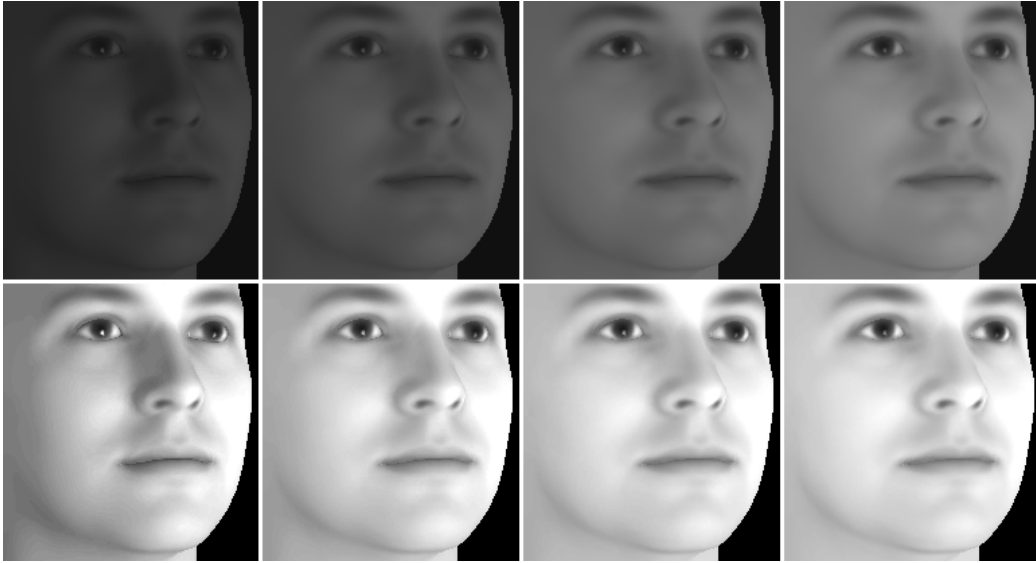
Figure 4.2: Illustration of the reduction in image differences resulting from variation in ambient brightness by the contrast enhancement operation of section 3.3.3. The top row contains images of faces at a fixed pose and with fixed direct lighting under variable ambient light brightness. The bottom row contains the contrast-stretched equivalents. [Images generated by FaceGen modeller, as described in section 2.3.]

and indeed, the mean L_1 pixel difference between the rightmost and leftmost images of each row dropped from 67.1 to 30.6 (with pixel values in the range [0, 255]) upon contrast enhancement. This operation therefore gave improved robustness to the initial estimation of the position of the directed light source, λ_{az} and λ_{el}, in that an invalid initial estimation of ambient brightness would introduce less error to the estimation of light source position than it otherwise would.

To estimate the unknown position of the directed light source for a given face image, a set of training images representing a range of illumination conditions over the range of poses considered by the algorithm was created using the FaceGen modeller. The approach here is similar to one described in [25] in which the authors sought to construct a lighting model (for a face at a fixed frontal pose) to express any arbitrary lighting conditions as a sum of point light sources. Here though, we attempt to esti-
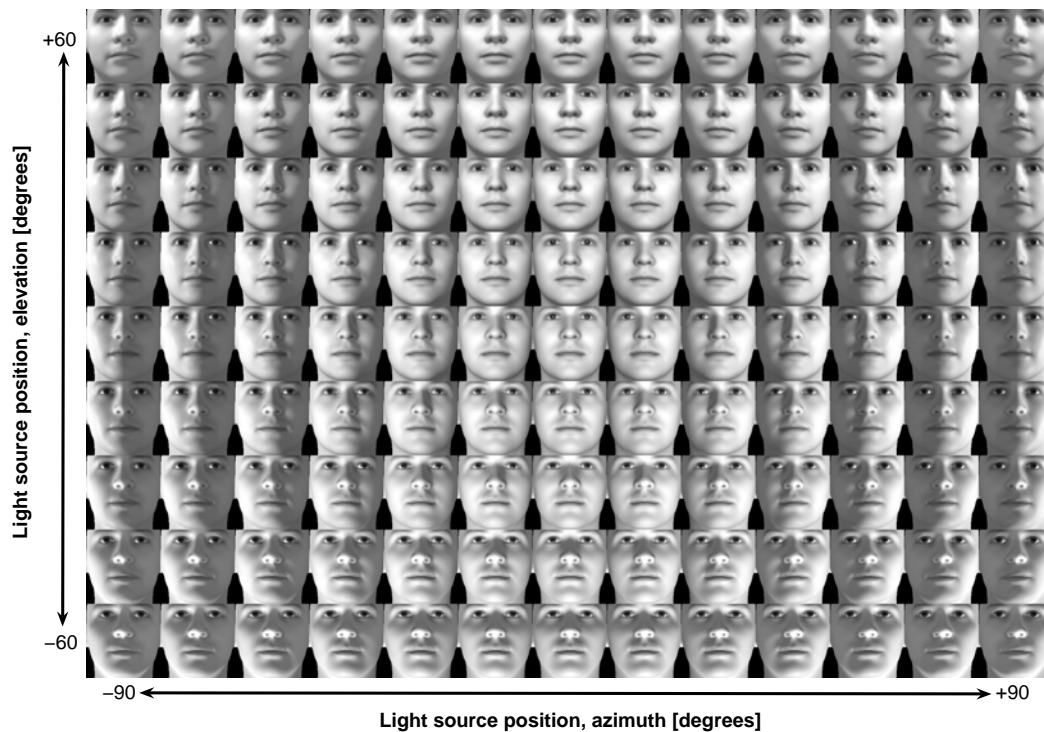
Figure 4.3: Set of intensity images for average face oriented at 0° in yaw and 4.8° in pitch showing variation in light source position in 15° increments from -90° to +90° in azimuth and -60° to +60° in elevation. [Images generated by FaceGen modeller, as described in section 2.3.]

mate the position of a single light source illuminating a face *at a known pose* based on characteristic images of an average face *at that pose* under a range of illumination directions. Once again making use of the morphable model framework and its ability to synthesize different imaging conditions, the average FaceGen face was rotated across the entire range of yaw and pitch values with each quantized into 1.8° increments. At each pose position, a set of images of the face under varying directional illumination was created. Light source position in this set ranged in 15° increments from -90° to +90° in azimuth and -60° to +60° in elevation. A grid of these training images for a characteristic pose (yaw = 0°; pitch = 4.8°) is given in figure 4.3.

For a given target image, the image processing operations of sections 3.2 and 3.3 yield a contrast enhanced intensity image that has been rotated to correct for rotation in roll and cropped to the face region based on an initial pose estimate in yaw and pitch. This processed target image is then compared with the subset of illumination training images representing the conditions for a face *at that pose estimate* in the following manner to derive the estimated position of a direct light source: Considering only those pixels at locations identified as representing skin according to the colour-based segmentation operation described in section 3.3.2, the mean intensity value over all skin pixels is calculated. At the same time, mean intensity values are calculated in 16 subregions of the skin area defined by a 4x4 grid centred at the nose tip and bounded by the limits of the detected skin region. This grid has two columns of equal width to the left of the nose tip to the left-most boundary of the skin region and two columns of equal width to the right of the nose tip to the right-most boundary. Similarly, the grid has two equally spaced rows above the nose tip to the upper boundary of the skin region and two equally spaced rows below the nose tip to the lower boundary. These 16 mean intensity values are normalized by the overall mean intensity value to give a 4x4 array of relative mean subregion intensities. Figure 4.4b shows a typical processed target image and figure 4.4c shows the corresponding mean intensity values for each subregion imposed on the face area for visualization.

The same relative mean intensity calculations were carried out for the illumination training images described previously. In order to derive an estimate of light source position, the 4x4 array of relative mean subregion intensity values for the target image is compared with the set of arrays representing the range of illumination conditions at the estimated target face pose, with the light source position estimate being the $(\lambda_{az}, \lambda_{el})$ combination at which the target and training arrays most closely match
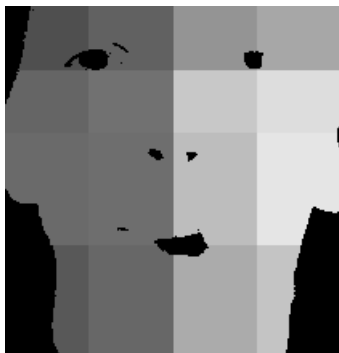
Figure 4.4: Original target image of a face under directed illumination (a); and roll-corrected, cropped, contrast-enhanced intensity version (b); with the corresponding mean intensity values over a 4x4 grid of skin subregions (c); and the average FaceGen face illuminated according to the estimated light source position (d). [Source of image (a) = IMM Face Database [40].]

each other. With the 4x4 arrays columnized into 16x1 vectors, the distance measure is defined as the norm of the difference between the two vectors. That is,

$$D_\lambda = \|G_{tgt} - G_{mod}\|, \tag{4.1}$$

where G_{tgt} and G_{mod} represent the length 16 vector of relative mean subregion intensity values for the target and model images, respectively. The $(\lambda_{az}, \lambda_{el})$ position estimate is that at which D_λ is a minimum over the quantized values of λ_{az} (-90° to +90° in 15° increments) and λ_{el} (-60° to +60° in 15° increments) for which training images were generated at the estimated pose of the target face. In order to verify whether the estimate was sensitive to the choice of distance measure, other distance measures — namely, the sum of absolute values of the difference vector and the cosine of the angle between the vectors — were tested on 5 faces and yielded the very same light source position estimates as D_λ in each case.

For instance, the target face of figure 4.4b has an estimated pose of 0° yaw and 4.8° pitch. (The illumination training images for this pose estimate are those shown in figure 4.3.) The D_λ distance measure values across the range of azimuth and elevation estimates at the pose estimate $(\psi, \theta) = (0°, 4.8°)$ are plotted in figure 4.5. The minimum D_λ value in figure 4.5 specifies the light source position estimate $(\lambda_{az}, \lambda_{el})$ = (60°, -15°). The average FaceGen face illuminated according to this initial light source position estimate is shown in figure 4.4d.

Unfortunately, the precise position of the light source illuminating figure 4.4b is unknown, so the quality of this initial light source position estimate cannot be definitively confirmed. However, the original image is described as having "a spot light added at the person's left side" [40], and judging from the shadow pattern in
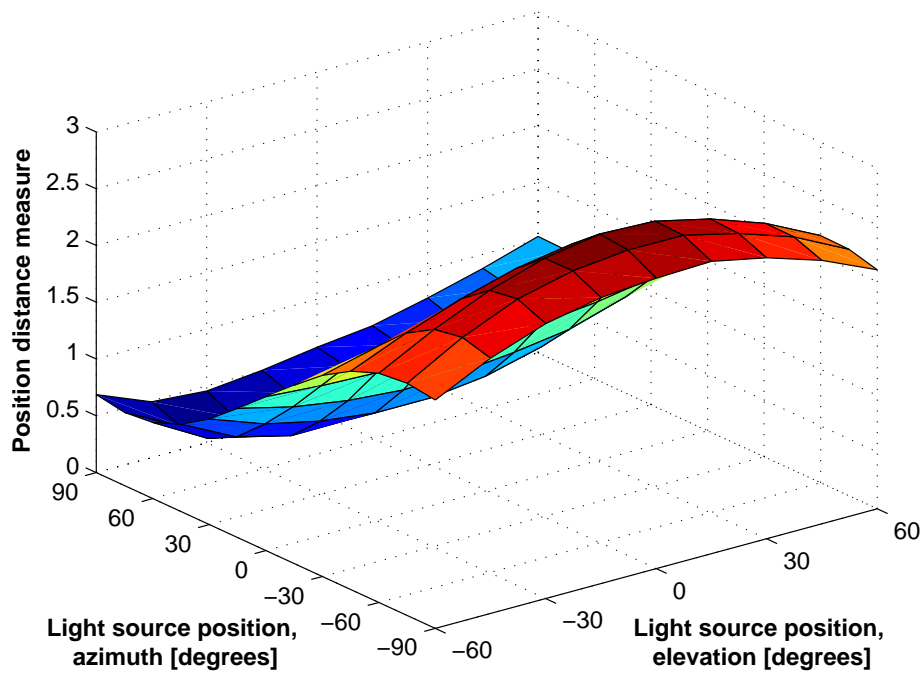
Figure 4.5: Plot of D_λ values against light source positions ranging in azimuth and elevation. Plot minimum defines the initial light source position estimate.

the background behind the subject in figure 4.4a, one could infer that this light is positioned at somewhat less than 90° in azimuth to the subject. Further, the original target image was rotated to correct for roll (as per section 3.2.1) to the order of -5.5°. This image rotation has the effect of shifting the apparent elevation of the light source by the same amount. So, our estimate of $(\lambda_{az}, \lambda_{el}) = (60°, -15°)$ is reasonable and is certainly sufficient as a starting point for the face-fitting optimization algorithm. The optimization algorithm, as it turns out, converged on a light source position of $(\lambda_{az}, \lambda_{el}) = (68.1°, -4.0°)$ in this case.

4.3.1 Illumination Refinement

Following the pattern of the pose refinement step of section 3.4, we search in the local area of the initial illumination condition estimation for improvements to the estimate. Illumination estimates are compared based on the L_2 image difference between sub-sampled target and model images. More accurately estimating the illumination parameters prior to attempting full-scale optimization contributes to greater overall algorithm efficiency. Otherwise, since illumination effects can cause such significant image differences, if the initial illumination parameter estimate is weak the first few iterations of the optimization algorithm — each of which requires population of the Jacobian of equation 3.18 with partial derivatives of the cost function with respect to each model parameter — are dominated by changes in illumination conditions only.

However, for a target face exhibiting a darker skin tone, considering illumination parameters exclusively at this stage could lead to an estimation skew in the direction of a darker light source and darker ambient brightness. As such, we also incorporate the first texture principal component as a parameter subject to optimization for illumination refinement. This principal component governs skin shade. It should be

noted that the presence of facial hair also has the tendency to introduce a similar skew, but since modelling the darker texture associated with facial hair is more complicated than modelling shade alone, this is left as a confounding factor at this stage.

4.4 Using the Model to Synthesize Novel Views

The algorithm developed for this thesis can be summarized as follows: Given a target facial image, shape and texture parameters of an underlying 3D morphable face model (in this case, the FaceGen face modeller) are optimized along with a set of image rendering parameters such that they produce an image that resembles as closely as possible the target image. The image rendering parameters describe the pose of the model face and the conditions under which it is illuminated. Starting with an average model face, the algorithm first estimates the pose and illumination conditions of the target image and applies the corresponding rendering parameters to the model. At each iteration of the optimization process, the algorithm generates a 3D face model based on the current model parameters, renders an image of that model, compares the rendered image with the target, and updates the model parameters according to a cost function that takes into account the residual pixel difference, goodness-of-fit between detected edges, and the likelihood of the modelled face based on a statistical prior.

Once the algorithm has derived the best-fit face model under the best-fit rendering parameters, synthesizing novel views is straightforward. The 3D face model defined by the optimal set of shape and texture parameters can simply be re-rendered according to any pose and illumination parameters. Perhaps most usefully, and as will be done for evaluating algorithm results in the chapter that follows, the face model can be

rotated to a frontal pose and illuminated from the front. The re-rendered image, it is hoped, should be more suitable for use in face recognition applications.

The entire algorithm is broken down by operation in the block diagram given in figure 4.6.
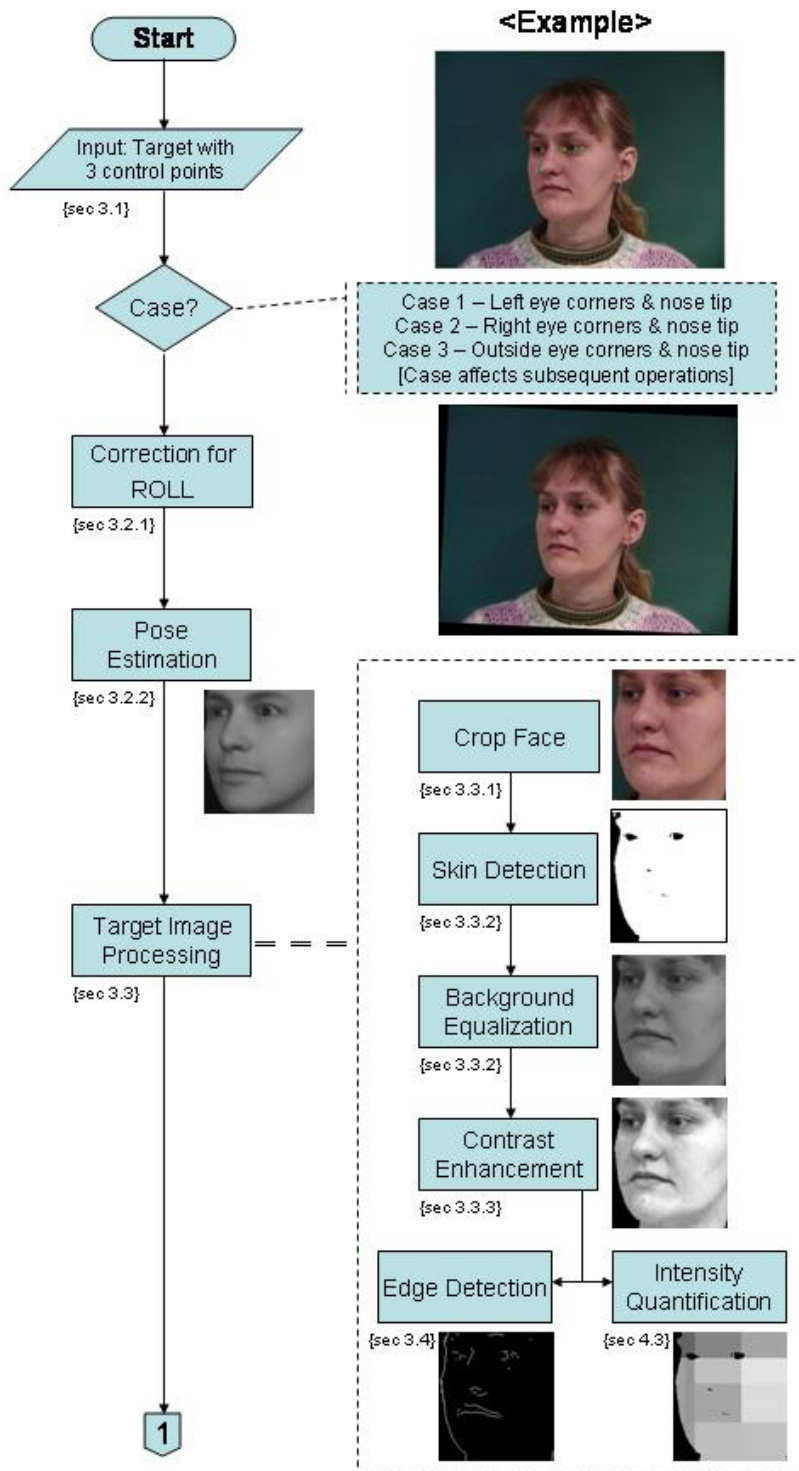
Figure 4.6: Block diagram of the 3D model to face image fitting algorithm.

63
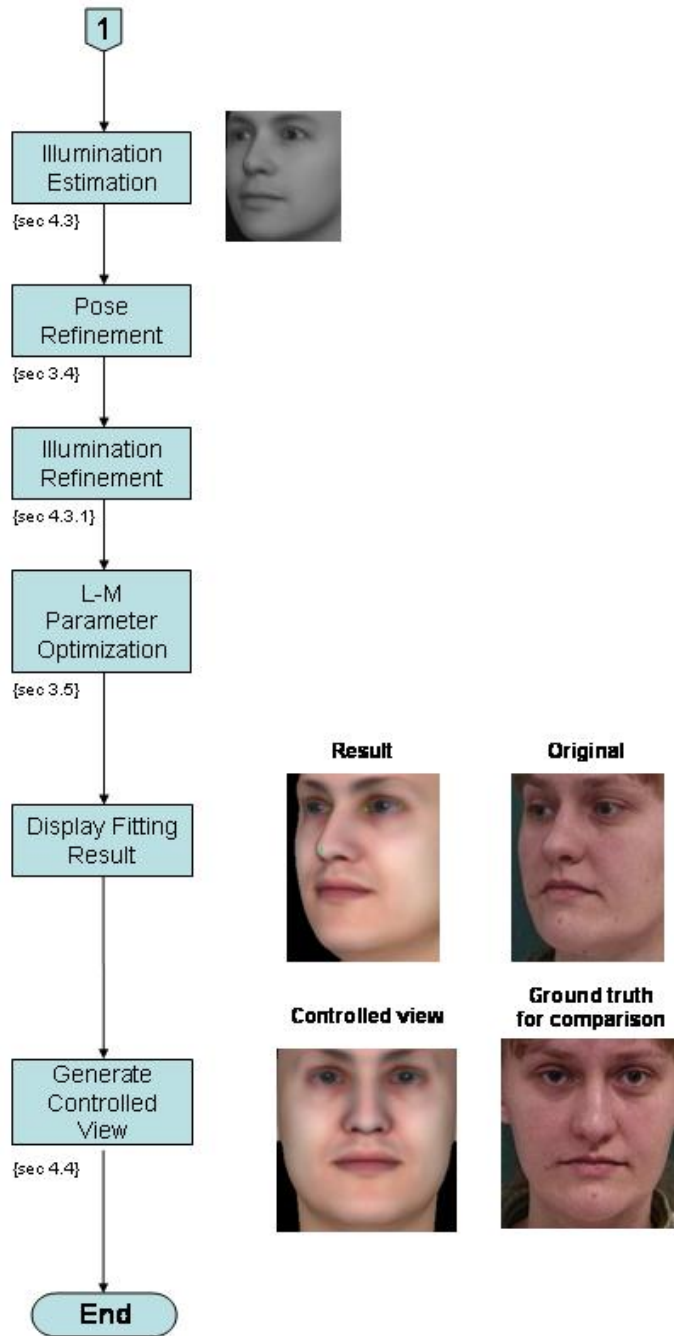
Figure 4.6: Block diagram of the 3D model to face image fitting algorithm (continued).

Chapter 5

Experiments and Discussion

The face-fitting algorithm was tested using images from the IMM face database [40, 64]. This database contains images of 40 subjects, but only those with colour images were tested, leaving 37 subjects. All of the subjects are Caucasian with 7 being female and 30 male. There are six images of each subject in the IMM database, the following four of which were used in our evaluation:

- Full frontal view with neutral expression and diffuse lighting.

- Left oblique view with neutral expression and diffuse lighting.

- Right oblique view with neutral expression and diffuse lighting.

- Full frontal view with neutral expression and side spot light added.

The full frontal view with neutral expression and diffuse lighting was considered to represent *ground truth* and served as the baseline for comparison with the images exhibiting non-frontal pose or illumination. The remaining three images of each subject were processed by the face-fitting algorithm to recover the 3D shape and

texture of the subject's face while subsequently recovering the pose and illumination conditions.

The 3D face models produced by the face-fitting algorithm were evaluated according to the identification task. That is, we assume that a *probe* image of an unknown person is presented to the system for comparison with a *gallery* of people of known identity, under the assumption that the probe image represents an individual who is in the gallery. In section 5.3, identification rates are presented both for identification directly from model parameters and for identification from pose and illumination normalized images using an appearance-based face recognition algorithm. Before presenting these results, we discuss in section 5.1 the effectiveness of certain previously described image preprocessing operations as contributors to algorithm performance.

5.1 Effectiveness of Preprocessing Operations

Before presenting the results of the face-fitting algorithm in terms of its ability to positively affect face recognition performance, the performance of certain preprocessing operations that were introduced in chapters 3 and 4 will be assessed.

5.1.1 Correction for Roll

For near-frontal target face images, the roll correction step described in section 3.2.1 worked flawlessly, as might be expected since two symmetric face features (outside eye corners) could be selected as feature points. For target faces at an oblique angle, on the other hand, the roll correction operation was inadequate.

Out of the 74 oblique angle faces from the IMM database [40,64] that were tested, between 15 and 20 of the roll-corrected target images did not properly account for roll

and were thus unsuitable as input to the face-fitting algorithm. Cases of over-rotation and under-rotation were approximately balanced. For the other target images, automatic roll correction provided good but not exact results. Since rotation in roll could not be reliably automated in the FaceGen environment and consequently roll angle ϕ could not be considered as a variable subject to optimization, it was decided to allow for additional roll correction measures.

Inaccuracy in roll correction and, indeed, in estimation of the face's orientation in yaw and pitch as well, stems from the use of only three feature points. The choice to use only three manually located feature points to the algorithm developed here was driven by a desire to minimize the amount of user input and effort required, and three points was considered to be the minimum required to bring target and model faces into reasonable correspondence. (By fixing the nose tip in image space and scaling according to the nose tip to eye vectors, as in equation 3.9, the use of only three feature points worked sufficiently well, at least, to allow the algorithm to proceed and fit the model to the image.) In contrast, the original morphable model system described in [13] required up to seven facial feature points in front and oblique views and up to eight in profile views to be manually identified using an interactive tool. The greater the number of facial feature points used, of course, the greater the expected quality of the fitting results [10]. Since, as testing progressed, it became clear that the roll correction calculation described in section 3.2.1 did work well in some instances, it was decided not to manually select additional feature points for all. Rather an interactive step was introduced whereby an image showing the *proposed* roll-corrected target was generated and the user could choose to scale the roll-correction angle, if necessary, to produce a more suitable result. All results presented here can be assumed to have been processed as necessary with this additional consideration.

5.1.2 Initial Pose & Illumination Estimation

Both the pose estimation scheme described in section 3.2.2 and the illumination estimation scheme of section 4.3 generally worked sufficiently well to provide good starting points. In about 4 out of every 5 non-frontal target faces tested, the initial pose estimate was likely good enough to act as a starting point to the optimization algorithm even without pose refinement as described in section 3.4. Pose refinement proved to be worthwhile, however, in the other cases for which the generalization to *average* face characteristics was unsuitable and thus the initial pose estimate was poor. Still, in 100% of the test images considered, the system converged to a reasonable pose solution. (Again, pose is unknown in the test images so the quality of the pose fit cannot be quantified.) Thus, the combination of initial pose estimation and pose refinement provided to the optimization algorithm a starting pose condition that robustly allowed for convergence to a reasonable solution.

Likewise, initial estimation of the illumination conditions was generally satisfactory. Since the illumination estimate was based on the pose estimate, illumination estimation suffered in those cases where the initial pose estimate was poor. Refinement of the illumination estimate prior to optimization (section 4.3.1) proved to be particularly worthwhile in such instances. Again, in all cases, the combination of initial illumination parameter estimation and illumination parameter refinement provided to the optimization algorithm starting illumination conditions from which the algorithm could converge to a reasonable solution.

5.2 Sample Results

Figure 5.1 contains examples of face reconstructions that were deemed to be 'good' in that the fitted model was a rank-1 match to the target image according to the face recognition measurement to be introduced in section 5.3.1. The left-most column of this figure contains the roll-corrected, cropped target images and the next column shows the image rendered by the fitted face model. This face model was used to generate a novel view which can be compared with the ground truth image from the right-most column.

Figure 5.2, meanwhile, provides examples of poor face-fitting results. In the first row, the original target image shows a subject with eyes partially closed. As the FaceGen face modeller was not configured to account for this, the algorithm converged on a solution in which the model exhibited very pale eyes. The second row was also a poor fit due to characteristics of the target image. In this case, the target image is rotated slightly in roll (in spite of the additional roll correction measure just discussed in section 5.1.1) which seems to have prevented convergence to a better solution. Also, the subject was not looking straight ahead, resulting again in a solution with pale eyes as the irises were fit partially to eye whites. The presence of facial hair seems to have negatively affected the fitting result of the third row. The mouth is shifted downwards into the beard area — perhaps impelled by edge matching to detected edges in the hair — and, since the subject's beard is fair in colour, the lips in the model became very pale.

Target	Fitted	Novel	Ground
Image	Model	View	Truth

Figure 5.1: Examples where the face-fitting algorithm generated a good fit to the target image, based on face recognition score. The target image is shown with the associated model image that resulted from the face-fitting process. The fitted model was re-rendered with normalized pose and illumination to create a novel view which can be compared with the ground truth image of the subject.

70

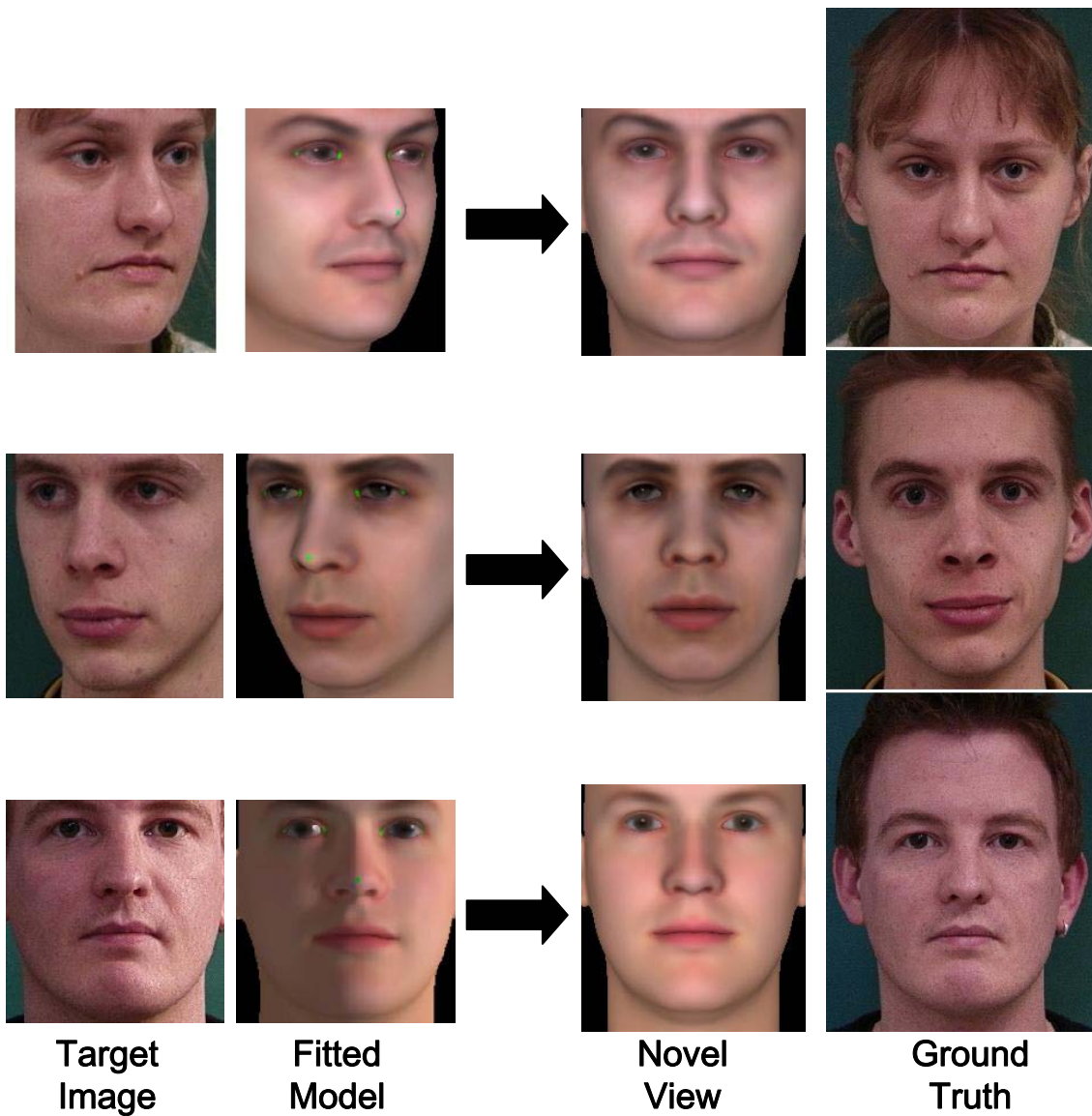| Target Image | Fitted Model | Novel View | Ground Truth |

Figure 5.2: Examples where the face-fitting algorithm generated a poor fit to the target image, based on face recognition score. The target image is shown with the associated model image that resulted from the face-fitting process. The fitted model was re-rendered with normalized pose and illumination to create a novel view which can be compared with the ground truth image of the subject.
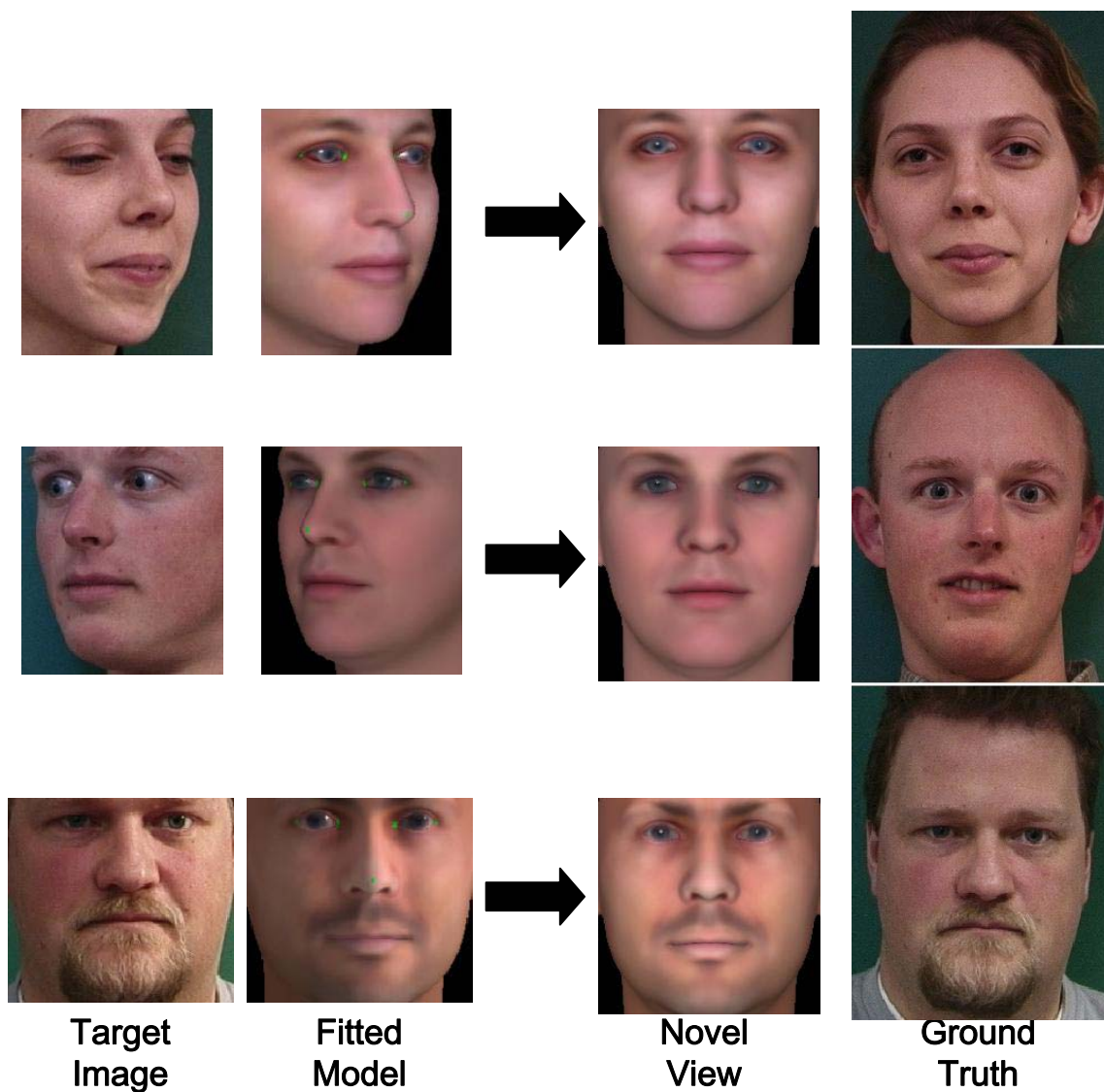
5.3 Evaluation of Identification Performance

In the 3DMM framework, face recognition can be performed in two distinct ways. The first method involves direct comparison of the parameters that define the shape and texture of the face model [12]. (The gallery in this case could even be parameters sets derived straight from 3D face scans. Alternatively, parameters sets could be entirely derived from model fitting to 2D images.) In the second approach, the modelled face can be re-rendered at a frontal pose and under favourable illumination such that any appearance-based recognition system can be used [10].

Both methods were used to evaluate the performance of our system and its ability to improve face recognition from unconstrained images. In both cases, the gallery set contained all 40 subjects of the IMM database [40]. As previously mentioned, images from only 37 subjects were tested. Images of the other 3 were grayscale only and thus unsuitable as input to our algorithm.

5.3.1 Identification using Model Parameters

The fitted face model is defined by a set of shape and texture coefficients, α_i and β_i, that remain constant over any changes in pose or illumination. In theory, model fitting to any two images of the same subject — regardless of pose or illumination — should yield the same parameter set. For identification from model parameters, we define a parameter vector,

$$\mathbf{p} = [\alpha_1, \ldots, \alpha_{N_S}, \ldots, \beta_1, \ldots, \beta_{N_T}]. \tag{5.1}$$

Our probe, then, is the vector $\mathbf{p}_p$ resulting from the face-fitting algorithm.

To derive a gallery vector, $\mathbf{p}_g$, we make use of the FaceGen PhotoFit utility (section 2.3.2). Running PhotoFit on each of the full frontal ground truth images, we assume that the output shape and texture coefficients represent each subject's 'true' face. The gallery vectors can then be compared with any given probe vector and rank identification determined. We adopt as match score MS a distance measure equivalent to the cosine of the angle between probe and gallery vectors:

$$MS = \frac{\langle \mathbf{p}_g, \mathbf{p}_p \rangle}{\|\mathbf{p}_g\| \cdot \|\mathbf{p}_p\|}. \tag{5.2}$$

The higher the match score, the more similar the parameter vectors, and the more likely it is that the vectors represent the same individual.

As outlined in section 2.5 for the identification task, match score values for each gallery image are ranked relative to a given probe image. Ideally, the gallery image representing the same subject identity as the probe should appear first in this ranked list. Practically, we consider the rate of identification against rank in the ordered list as a measure of algorithm performance.

Cumulative match characteristic (CMC) curves illustrating algorithm performance are given in figure 5.3. Probe images in which pose was the primary source of variation (74 images total) are considered separately from those in which illumination was the primary source of variation (37 images total), to facilitate comparison with the output from a commercial appearance-based face recognition algorithm. This commercial algorithm, which is about 5 years old, was tested using unprocessed gallery and probe images.

Figure 5.3a shows that our algorithm slightly outperforms the commercial algorithm in the task of identifying people from images that are unconstrained in pose.
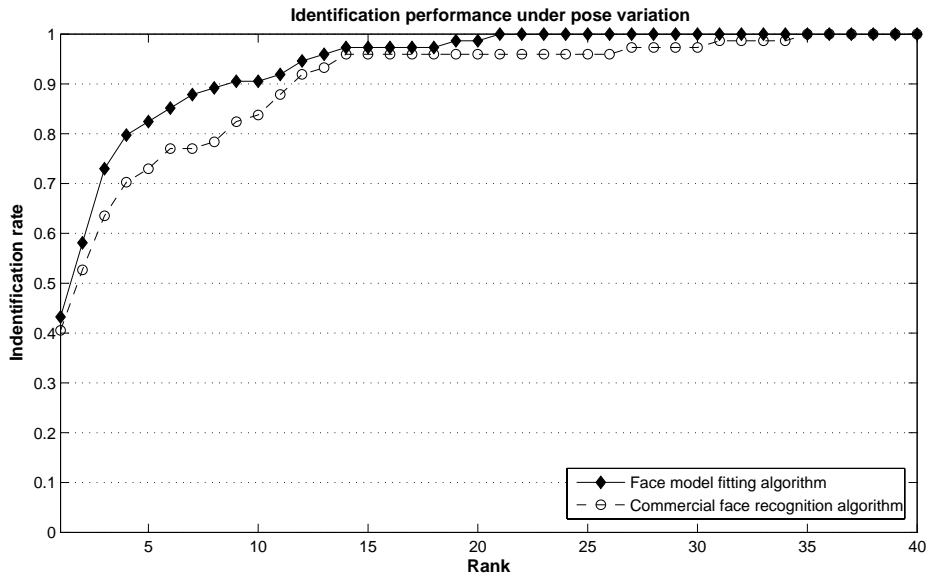
(a)
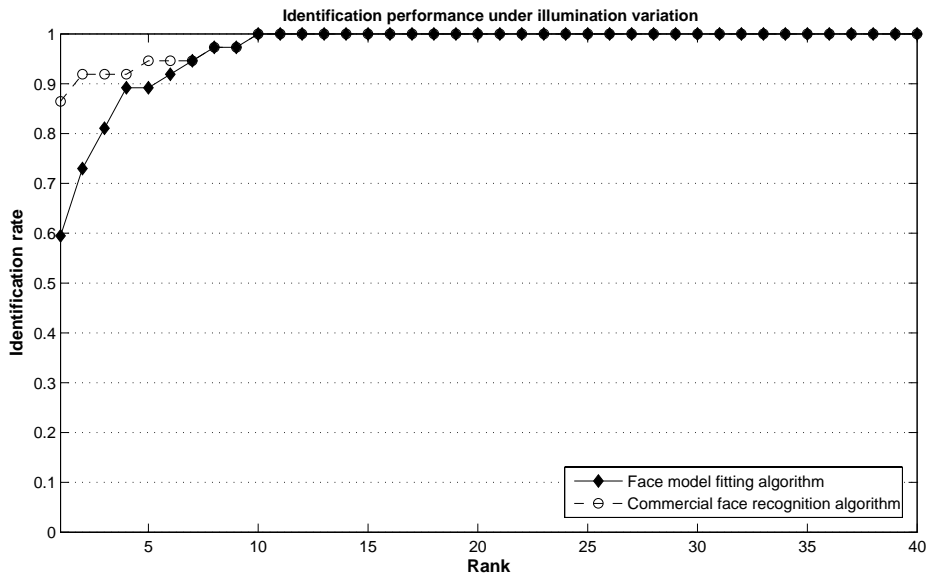
(b)

Figure 5.3: Cumulative match characteristic curves showing identification performance (with match score from model parameters) of the face model fitting algorithm in comparison with a commercial face recognition algorithm. In (a), results are given for 74 probe images varying from the gallery image conditions primarily in pose, while (b) gives results for 37 probe images varying primarily in illumination. In both cases, probes were compared against a gallery of 40 individuals.

On the other hand, the commercial algorithm outperforms our morphable model approach in recognizing faces under varying illumination, as is evident from figure 5.3b.

5.3.2   Identification using Normalized Images

Having derived a face model whose shape and texture is invariant under changing pose and illumination, we can normalize to a frontal pose and to illumination conditions that are conducive to successful face recognition (as outlined in section 4.4) and use the newly rendered face image as input to a face recognition algorithm rather than using the original unconstrained target face image.

Upon initial face recognition testing using this approach with normalized images of whole head FaceGen models, an interesting result was noticed. For nearly every probe image, the same gallery subject ranked as the most likely identity according to the commercial face recognition algorithm used for testing. In a previous study that involved this particular algorithm [5], Adler surmised that hair style bore some significance in the algorithm's fundamental similarity measure. This does indeed seem to be the case since, as shown in figure 5.4, the subject most highly identified with many normalized model images bore a strikingly similar hair style — or lack thereof.

To correct for this, all FaceGen images to be tested in the face recognition algorithm were cropped to remove the top of the head. Recognition results with this modified set of pose and illumination normalized images were poor to say the least. Over the whole set of test images (pose and illumination variation), the mean identification rank was 18.5 — only slightly better than what one could expect by using random face images as probes. Again, there were 3 or 4 gallery subjects who ranked highly in identification score for many different model images. Visual analysis showed these subjects to be those with 'smoother' than average skin texture, that is, those

Figure 5.4: Hair style influences the similarity measure of the commercial face recognition algorithm used in testing. Most full head FaceGen model images ranked as highly probable identity matches with the gallery subject in the centre above.

with fewer wrinkles, freckles, and the like.

This would seem to indicate a strong contribution from skin detail to the commercial algorithm's similarity metric since the FaceGen model faces being presented as probe images were all devoid of the fine skin detail that cannot be directly simulated by the statistical appearance models that govern the β_i texture coefficients [61]. To test this, some of the poorer performing model face images were re-rendered with skin detail texture overlaid on the texture map. (Adding and customizing detail texture to the model is a feature of the FaceGen modeller.) Also, given the apparent contribution of hair to the face recognition algorithm's similarity scores, hair was added using standard hair models available in FaceGen. The new face images with detailed texture and with hair were retested to investigate if an improvement in recognition would be realized.

Example cases showing the original model image and the corresponding modified version are given in figure 5.5. Testing on the image pair to the left showed an improvement in identification from rank-34 to rank-3 while the pair on the right

Figure 5.5: Adding hair and skin detail texture to the model image resulted in a significant improvement in face recognition for the face pair on the left but marginal improvement only for the pair on the right.

showed only marginal improvement from rank-39 to rank-35. Clearly, skin detail is an important factor to the particular face recognition algorithm used in testing. For a given probe image, even adding skin detail and hair that is not derived from the original target image can lend to an improvement in matching performance, but this is not necessarily the case. So, for identification using normalized images processed by an appearance-based face recognition algorithm — at least, for the algorithm on which testing was performed — we need some way to extract the fine detail of the probe subject's skin and impose this detail on the recovered face model. Our algorithm is not currently capable of this function, and is thus unsuitable as a preprocessing step for face recognition by the commercial algorithm.

Chapter 6

Conclusion

The ability of face recognition systems to accurately recognize faces is degraded when images are not constrained for factors such as facial pose, illumination, facial expression, and occlusions. In this thesis the problem of unconstrained pose and illumination was considered with the objective being to address the following question: Is it possible to accurately predict the appearance of an individual and subsequently generate a frontal and uniformly illuminated view of their face from an image that is unconstrained in pose and illumination?

6.1   Summary

The approach chosen to try to answer this question was that of a generative 3D morphable face model used in conjunction with a face-fitting algorithm, following the technique introduced by Blanz and Vetter [11, 12]. This system synthesizes a face model whose projected image resembles as closely as possible a given target image under analysis. We have shown that, in comparison with a commercial facial

78

recognition algorithm, this system increases the likelihood of accurately predicting an individual's identity from an image that is unconstrained primarily in pose, but offers no such improvement for images exhibiting frontal pose but unconstrained illumination. While our system does allow for the generation of frontal and uniformly illuminated views of a subject's face from an image that is unconstrained in pose and illumination, the accuracy of these reconstructions was not confirmed. Using a commercial face recognition algorithm, identification performance from normalized images was not good and is suspected to have been limited by lack of extracted skin detail.

6.2   Key Contributions

The key contributions of this thesis made during the investigation and implementation of the 3D morphable model approach as a solution to the problem of unconstrained target image pose and illumination are as follows:

- An algorithm for face modelling to match images of faces under variable pose and illumination has been developed. This algorithm manipulates the parameters of a commercial 3D face modeller known as FaceGen. The ability of this algorithm to successfully generate a face model whose identity matches that of the target face image has been demonstrated. The efficacy of the FaceGen face modeller as the underlying model in the 3D morphable model approach has thus been proven.

- It has been shown that the morphable model approach can work when initiated with as few as three facial feature points located. Improved results would be

expected, though, with the use of more feature points as is the case in other implementations.

- In comparison with a commercial face recognition algorithm — albeit one that is approximately five years old — our algorithm showed slight improvement for face identification under variable pose, but performed more poorly when testing frontal face views under variable illumination. These results came when using a similarity score based on model parameters rather than using pose and illumination normalized images as input to the same face recognition algorithm.

- Skin detail texture was found to factor significantly in similarity score measurement for the particular commercial face recognition algorithm under comparison.

- Exploiting the characteristics of the "average" face under rotation, schemes for estimating pose from the relative position of three facial feature points and for estimating the position of a direct light source illuminating a face at a given pose have been developed. Although based on characteristics of an average face only, these estimates were found to be reasonable across a range of different faces. It was also also found that the angle between the nose tip and outside eye corner of the average face across rotations in yaw and pitch could be expressed as a function of the relative position of the nose tip and both corners of the same eye, yet this relationship did not hold well across non-average faces.

- Using the "Win32::GuiTest" Perl module [2], scripts were created for the automation of FaceGen face model manipulation, simulating user input such as mouse clicks and movements, keyboard input, and window selection. This work

is potentially transferable in that similar scripts could be developed to automate interaction with other GUI applications.

6.3 Directions for Future Work

Some suggestions for future work to build on this thesis research are as follows:

- **Automatic facial feature detection.** Implementation of an automatic facial feature detection algorithm would enable the use of more feature points with less user input. Improved fitting results would be expected with the locating of more feature points.

- **Skin detail texture extraction.** Statistical texture models are unable to account for the detail in skin texture. Once the face model has been fit to the target image, some mechanism is needed to extract the detail texture from the skin surface. This detail texture could then be mapped to the entire face surface (using assumptions of texture symmetry, if necessary) to render an improved likeness of the target which would be suitable as input to any appearance-based face recognition algorithm.

- **Modelling from multiple images.** The algorithm could be extended to optimize shape and texture parameters under simultaneous fitting to multiple images of the same target.

Appendix A

GuiTest Perl Scripts

Listed in this appendix are various Perl scripts that enabled automation of face model manipulation in the FaceGen environment.

A.1   Face Rotation

Rotates the face by `xmove` in the horizontal (rotation in yaw) and `ymove` pixels in the vertical (rotation in pitch) from the default face pose.

```
use Win32::GuiTest qw( FindWindowLike SetFocus MouseMoveAbsPix
  SetForegroundWindow SendLButtonDown SendLButtonUp
  TabCtrl_SetCurSel TabCtrl_SetCurFocus PushChildButton
  GetChildWindows GetWindowRect SetActiveWindow );

my $xmove = $ARGV[0];
my $ymove = $ARGV[1];

my @whnds = FindWindowLike( 0, "FaceGen Modeller 3.2 Free" );
die "Could not find FaceGen\n" if not @whnds;

my @chnds = GetChildWindows($whnds[0]);

SetForegroundWindow($whnds[0]);

TabCtrl_SetCurFocus($chnds[18], 2 );
TabCtrl_SetCurSel($chnds[18], 2 ); # Selects "Camera" tab.

PushChildButton($whnds[0],1384);
#Pushes "Reset Camera" button under "Camera" tab.
## PushChildById($whnds[0],1384,4); # Also works.
sleep 1;

my @fgwin = FindWindowLike( $whnds[0], undef, "FanWin3dClass" );
```

```
( $lx, $ty, $rx, $by ) = GetWindowRect( $fgwin[0] );

$xoff = -0.5*$xmove + 202;
$yoff = -0.5*$ymove + 202;

my $xctr = $lx + $xoff;
my $yctr = $ty + $yoff;

SetActiveWindow( $fgwin[0] );
SetFocus( $fgwin[0] );
MouseMoveAbsPix($xctr,$yctr);
SendLButtonDown;
MouseMoveAbsPix($xctr+$xmove,$yctr+$ymove);
SendLButtonUp;
```

A.2   Set Direct Light Source Parameters

Sets the light source to a position of `azimval` in azimuth and `elevval` in elevation and to a brightness of `brigval`.

```
use Win32:: GuiTest qw( FindWindowLike SendKeys MouseClick
  SetForegroundWindow WMSetText TabCtrl_SetCurSel
  TabCtrl_SetCurFocus GetChildWindows GetTabItems );

my $azimval = $ARGV[0];
my $elevval = $ARGV[1];
my $brigval = $ARGV[2];

my @whnds = FindWindowLike( 0, "FaceGen" );
die "Could not find FaceGen\n" if not @whnds;

my @chnds = GetChildWindows($whnds[0]);
my @labels = GetTabItems( $chnds[18] );

SetForegroundWindow($whnds[0]);

TabCtrl_SetCurFocus($chnds[18], 1 );
TabCtrl_SetCurSel($chnds[18], 1 ); # Selects "View" tab.

my $azim_ctrl_id = 1343; # Azimuth edit window, 53F Hex
my $elev_ctrl_id = 1345; # Elevation edit window, 541 Hex
my $brig_ctrl_id = 1350; # Brightness edit window, 546 Hex

my @azim = FindWindowLike($whnds[0],undef,"^Edit",$azim_ctrl_id);
my @elev = FindWindowLike($whnds[0],undef,"^Edit",$elev_ctrl_id);
my @brig = FindWindowLike($whnds[0],undef,"^Edit",$brig_ctrl_id);
```

```
MouseClick('Enable');
MouseClick('Enable');
SendKeys("TAB");
WMSetText($azim[0], $azimval);
SendKeys("TAB");
WMSetText($elev[0], $elevval);
SendKeys("TAB TAB");
WMSetText($brig[0], $brigval);
MouseClick('Enable');
MouseClick('Enable');
```

A.3   Set Ambient Light Brightness

Sets the ambient light brightness to a value of `ambival`.

```
use Win32::GuiTest qw( FindWindowLike SendKeys MouseClick
  SetForegroundWindow WMSetText TabCtrl_SetCurSel
  TabCtrl_SetCurFocus GetChildWindows GetTabItems );

my $ambival = $ARGV[0];

my @whnds = FindWindowLike( 0, "FaceGen" );
die "Could not find FaceGen\n" if not @whnds;

my @chnds = GetChildWindows($whnds[0]);
my @labels = GetTabItems( $chnds[18] );

SetForegroundWindow($whnds[0]);

TabCtrl_SetCurFocus($chnds[18], 1 );
TabCtrl_SetCurSel($chnds[18], 1 ); # Selects "View" tab.

my $ambi_ctrl_id = 1329; # Ambient Brightness edit window, 531 Hex
my @ambi = FindWindowLike($whnds[0],undef,"^Edit",$ambi_ctrl_id);

MouseClick('Enable');
MouseClick('Enable');
sleep 1;
SendKeys("+TAB +TAB");
WMSetText($ambi[0], $ambival);
MouseClick('Enable');
MouseClick('Enable');
```

A.4   Save Face Image

Saves an image of current face model with the filename `name`. (Similar scripts are used to open and save face models, as noted in the script comments.)

```perl
use Win32::GuiTest qw(FindWindowLike SetForegroundWindow SendKeys
  GetMenu GetSubMenu MenuSelect);

my $name = $ARGV[0];

my @windows = FindWindowLike(0, "FaceGen Modeller 3.2 Free");
die "Could not find FaceGen\n" if not @windows;
SetForegroundWindow($windows[0]);

$menu_id = GetMenu($windows[0]);
$submenu_id = GetSubMenu($menu_id,0); # Pulls "File" sub-menu.

MenuSelect("Save &Image", $windows[0], $submenu_id);
# Similarly to open and save a face model -
# MenuSelect("&Open", $windows[0], $submenu_id);
# MenuSelect("&Save", $windows[0], $submenu_id);

SendKeys("$name~"); # In format and to location of previous save.
```

Bibliography

[1] The International Biometric Industry Association. `http://www.ibia.org/ biometrics/glossary.php`, accessed Nov 2008.

[2] Win32::GuiTest project home page. `http://sourceforge.net/projects/ winguitest/`, accessed May 2008.

[3] J.L. Ackerman, W.R. Proffit, D.M. Sarver, M.B. Ackerman, and M.R. Kean. Pitch, roll, and yaw: Describing the spatial orientation of dentofacial traits. *American Journal of Orthodontics and Dentofacial Orthopedics*, 131(3):305–310, Mar 2007.

[4] Y. Adini, Y. Moses, and S. Ullman. Face recognition: The problem of compensating for changes in illumination direction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):721–732, Jul 1997.

[5] A. Adler. Sample images can be independently restored from face recognition templates. In *Proceedings IEEE CCECE 2003, Canadian Conference on Electrical and Computer Engineering*, volume 2, pages 1163–1166, May 2003.

[6] S.A. Al-Shehri. A simple and novel method for skin detection and face locating and tracking. In *Proceedings of the Asia-Pacific Conference on Computer-Human Interaction*, pages 1–8, 2004.

[7] E. Ardizzo, M. La Cascia, V. Di Gesú, and C. Valenti. Content-based indexing of image and video databases by global and shape features. In *Proceedings, 13th International Conference on Pattern Recognition*, volume 3, pages 140–144, Aug 1996.

[8] D. Beymer and T. Poggio. Image representations for visual learning. *Science*, 272(5270):1905–1909, Jun 1996.

[9] D.J. Beymer. Face recognition under varying pose. In *Proceedings, CVPR 1994 — Conference on Computer Vision and Pattern Recognition*, pages 756–761. IEEE Computer Society, Jun 1994.

[10] V. Blanz, P. Grother, P.J. Phillips, and T. Vetter. Face recognition based on frontal views generated from non-frontal images. In *Proceedings, CVPR 2005 — Conference on Computer Vision and Pattern Recognition*, volume 2, pages 454–461. IEEE Computer Society, Jun 2005.

[11] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Proceedings, 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 99)*, pages 187–194, Aug 1999.

[12] V. Blanz and T. Vetter. Face identification across different poses and illuminations with a 3D morphable model. In *Proceedings, International Conference on Automatic Face and Gesture Recognition*, pages 192–197. IEEE, May 2002.

[13] V. Blanz and T. Vetter. Face recognition based on fitting a 3D morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1063–1074, Sep 2003.

[14] W.J. Braje, D. Kersten, M.J. Tarr, and N.F. Troje. Illumination effects in face recognition. *Psychobiology*, 26(4):371–380, 1999.

[15] J.F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:679–698, Nov 1986.

[16] S.C. Chapra. *Applied Numerical Methods with Matlab for Engineers and Scientists*. McGraw-Hill Higher Education, 2nd edition, 2008.

[17] L.G. Farkas. *Anthropometry of the Head and Face*. Raven Press, New York, 2nd edition, 1994.

[18] P. Gejgus and M. Sperka. Face tracking in color video sequences. In *Spring Conference on Computer Graphics*, pages 245–249, 2003.

[19] R.C. Gonzalez, R.E. Woods, and S.L. Eddins. *Digital Image Processing Using Matlab*. Pearson Prentice Hall, Upper Saddle River, NJ, 2004.

[20] H. Gouraud. Continuous shading of curved surfaces. *IEEE Transactions on Computers*, C-20(6):623–629, Jun 1971.

[21] R. Gross, S. Baker, I. Matthews, and T. Kanade. Face recognition across pose and illumination. In S.Z. Li and A.K. Jain, editors, *Handbook of Face Recognition*, chapter 9. Springer, New York, 2005.

[22] R. Gross and V. Brajovic. An image preprocessing algorithm for illumination invariant face recognition. In *Proceedings, 4th International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 375–380, Guildford, UK, Jun 2003.

[23] R. Gross, I. Matthews, and S. Baker. Fisher light-fields for face recognition across pose and illumination. In *Proceedings, DAGM 2002 — 24th Symposium on Pattern Recognition*, pages 481–489, Zurich, Switzerland, Sep 2002.

[24] R. Gross, J. Shi, and J. Cohn. Quo vadis face recognition? Technical report CMU-RI-TR-01-17, Robotics Institute, Carnegie Mellon University, Jun 2001.

[25] P.L. Hallinan, G.G. Gordon, A.L. Yuille, P. Giblin, and D. Mumford. *Two- and Three-Dimensional Patterns of the Face*. A K Peters, Natick, MA, 1999.

[26] H. Hill and V. Bruce. Effects of lighting on the perception on facial surface. *Journal of Experimental Psychology: Human Perception & Performance*, 22(4):986–1004, 1996.

[27] J. Ho and D. Kriegman. On the effect of illumination and face recognition. In W. Zhao and R. Chellappa, editors, *Face Processing: Advanced Modeling and Methods*, chapter 11. Elsevier Academic Press, Burlington, MA, 2006.

[28] T. Huang, Z. Xiong, and Z. Zhang. Face recognition applications. In S.Z. Li and A.K. Jain, editors, *Handbook of Face Recognition*, chapter 16. Springer, New York, 2005.

[29] International Biometric Group. *Which is the Best Biometric Technology?* http://www.biometricgroup.com/reports/public/reports/best_biometric.html, accessed Nov 2008.

[30] International Civil Aviation Organization, Montreal, Canada. *Machine Readable Travel Documents Selection of a Globally Interoperable Biometric for Machine-Assisted Identity Confirmation with MRTDs*, 2001. ICAO TAG MRTD/NTWG Technical Report.

[31] I.T. Joliffe. *Principal Component Analysis*. Springer series in Statistics. Springer-Verlag, New York, 2nd edition, 2002.

[32] A. Lanitis, C.J. Taylor, and T.F. Cootes. Automatic interpretation and coding of face images using flexible models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):743–756, Jul 1997.

[33] K. Levenberg. A method for the solution of certain problems is least squares. *Quarterly of Applied Mathematics*, 2(2):164–168, Jul 1944.

[34] S.Z. Li and A.K. Jain. Introduction. In S.Z. Li and A.K. Jain, editors, *Handbook of Face Recognition*, chapter 1. Springer, New York, 2005.

[35] M.L.A. Lourakis and A.A. Argyros. Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment? In *Proceedings, Tenth IEEE International Conference on Computer Vision*, volume 2, pages 1526–1531, Oct 2005.

[36] K. Madsen, H.B. Nielsen, and O. Tingleff. Methods for non-linear least squares problems. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Lyngby, Denmark, Apr 2004.

[37] D.W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431–441, 1963.

[38] P.A. Mlsna and J.J. Rodríguez. Gradient and Laplachian edge detection. In A. Bovik, editor, *Handbook of Image & Video Processing*, chapter 4.13. Elsevier Academic Press, Burlington, MA, 2005.

[39] National Science and Technology Council Subcommittee on Biometrics, Washington, DC. *The National Biometrics Challenge*, Aug 2006. `http://www.biometrics.gov/Documents/biochallengedoc.pdf`, accessed Nov 2008.

[40] M.M. Nordstrøm, M. Larsen, J. Sierakowski, and M.B. Stegmann. The IMM face database - an annotated dataset of 240 face images. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, Lyngby, Denmark, May 2004.

[41] A. O'Toole, F. Jiang, D. Roark, and H. Abdi. Predicting human performance for face recognition. In W. Zhao and R. Chellappa, editors, *Face Processing: Advanced Modeling and Methods*, chapter 9. Elsevier Academic Press, Burlington, MA, 2006.

[42] A.J. O'Toole. Psychological and neural perspectives on human face recognition. In S.Z. Li and A.K. Jain, editors, *Handbook of Face Recognition*, chapter 15. Springer, New York, 2005.

[43] A.J. O'Toole, P.J. Phillips, J. Fang, J. Ayyad, N. Penard, and H. Abdi. Face recognition algorithms surpass humans matching faces over changes in illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1642–1646, Sep 2007.

[44] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *Proceedings, CVPR 1994 — Conference on Computer Vision and Pattern Recognition*, pages 84–91. IEEE Computer Society, Jun 1994.

[45] P.J. Phillips, P.J. Flynn, T. Scruggs, K.W. Bowyer, and W. Worek. Preliminary Face Recognition Grand Challenge results. In *Proceedings, FGR 2006—International Conference on Automatic Face and Gesture Recognition*, pages 15–24. IEEE Computer Society, Apr 2006.

[46] P.J. Phillips, P.J. Flynn, W.T. Scruggs, K.W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek. Overview of the Face Recognition Grand Challenge. In *Proceedings, CVPR 2005—Conference on Computer Vision and Pattern Recognition*, volume 1, pages 947–954. IEEE Computer Society, Jun 2005.

[47] P.J. Phillips, P. Grother, and R. Michaels. Evaluation methods in face recognition. In S.Z. Li and A.K. Jain, editors, *Handbook of Face Recognition*, chapter 14. Springer, New York, 2005.

[48] P.J. Phillips, P. Grother, R.J. Michaels, D.M. Blackburn, E. Tabassi, and M. Bone. *Face Recognition Vendor Test 2002: Evaluation Report*, Mar 2003. http://www.frvt.org/FRVT2002/documents.htm, accessed Oct 2008.

[49] P.J. Phillips, T. Scruggs, A.J. O'Toole, P.J. Flynn, K.W. Bowyer, C.L. Schott, and M. Sharpe. FRVT 2006 and ICE 2006 large-scale results. Technical Report NISTIR 7408, National Institute and Standards and Technology, Mar 2007.

[50] B.T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.

[51] C.A. Poynton. *A Technical Introduction to Digital Video*. Wiley, New York, 1996.

[52] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 2nd edition, 1992.

[53] S.J.D. Prince, J. Warrell, J.H. Elder, and F.M. Felisberti. Tied factor analysis for face recognition across large pose differences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):970–984, June 2008.

[54] D.A. Roark, A.J. O'Toole, H. Abdi, and S.E. Barrett. Learning the moves: The effect of familiarity and facial motion on person recognition across large changes in viewing format. *Perception*, 35(6):761–773, 2006.

[55] S. Romdhani, J. Ho, T. Vetter, and D.J. Kriegman. Face recognition using 3-D models: Pose and illumination. *Proceedings of the IEEE*, 94(11):1977–1999, Nov 2006.

[56] S. Romdhani, J-S. Pierrard, and T. Vetter. 3D morphable face model, a unified approach for analysis and synthesis of images. In W. Zhao and R. Chellappa, editors, *Face Processing: Advanced Modeling and Methods*, chapter 4. Elsevier Academic Press, Burlington, MA, 2006.

[57] S. Romdhani and T. Vetter. Efficient, robust and accurate fitting of a 3D morphable model. In *Proceedings, Ninth IEEE International Conference on Computer Vision*, volume 1, pages 59–66, Oct 2003.

[58] S. Romdhani and T. Vetter. Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *Proceedings, CVPR 2005 — Conference on Computer Vision and Pattern Recognition*, volume 2, pages 986–993. IEEE Computer Society, Jun 2005.

[59] A. Roy-Chowdhury and R. Chellappa. 3D face modeling from monocular video sequences. In W. Zhao and R. Chellappa, editors, *Face Processing: Advanced Modeling and Methods*, chapter 6. Elsevier Academic Press, Burlington, MA, 2006.

[60] Singular Inversions Inc., Vancouver, BC. *FaceGen 3.1 Main SDK Documentation*, 2007.

[61] Singular Inversions Inc., Vancouver, BC. *FaceGen Modeller User Manual*, 2008. `http://www.facegen.com/modeller_help.htm`, accessed Dec 2008.

[62] P. Sinha, Benjamin J. Balas, Y. Ostrovsky, and R. Russell. Face recognition by humans. In W. Zhao and R. Chellappa, editors, *Face Processing: Advanced Modeling and Methods*, chapter 8. Elsevier Academic Press, Burlington, MA, 2006.

[63] G. Stamou, M. Krinidis, E. Loutas, N. Nikolaidis, and I. Pitas. 2D and 3D motion tracking in digital video. In A. Bovik, editor, *Handbook of Image & Video Processing*, chapter 4.11. Elsevier Academic Press, Burlington, MA, 2005.

[64] M.B. Stegmann, B.K. Ersbøll, and R. Larsen. FAME – a flexible appearance modelling environment. *IEEE Transactions on Medical Imaging*, 22(10):1319–1331, 2003.

[65] M.A.O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *Proceedings ECCV 2002, 7th European Conference on Computer Vision*, volume 2350 of *Lecture Notes in Computer Science*, pages 447–460, Copenhagen, Denmark, May 2002.

[66] M.A.O. Vasilescu and D. Terzopoulos. Multilinear image analysis for facial recognition. In *Proceedings, 16th International Conference on Pattern Recognition*, volume 2, pages 511–514, Quebec City, Canada, 2002.

[67] T. Vetter and T. Poggio. Linear object classes and image synthesis from a single example image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):733–742, Jul 1997.

[68] D.M. Weinstein. The analytic 3-D transform for the least-squared fit of three pairs of corresponding points. Department of Computer Science Technical Report UUCS-98-005, University of Utah, Mar 1998.

[69] J.D. Woodward, Jr., N.M. Orlans, and P.T. Higgins. *Biometrics*. McGraw-Hill/Osborne, Berkeley, CA, 2003.

[70] Y. Xu, A. Roy-Chowdhury, and K. Patel. Integrating illumination, motion and shape models for robust face recognition in video. *EURASIP Journal on Advances in Signal Processing: Advanced Signal Processing and Pattern Recognition Methods for Biometrics*, Volume 2008, 2008. Article 469698, 13 pages.

[71] J. Yang and A. Waibel. A real-time face tracker. In *Proceedings, Workshop on Applications in Computer Vision*, pages 142–147, 1996.

[72] L. Zhang, S. Wang, and D. Samaras. Face synthesis and recognition from a single image under arbitrary unknown lighting using a spherical harmonic basis morphable model. In *Proceedings, CVPR 2005 — Conference on Computer Vision and Pattern Recognition*, volume 2, pages 209–216. IEEE Computer Society, Jun 2005.

[73] W. Zhao and R. Chellappa. A guided tour of face processing. In W. Zhao and R. Chellappa, editors, *Face Processing: Advanced Modeling and Methods*, chapter 1. Elsevier Academic Press, Burlington, MA, 2006.

[74] W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, 2003.

[75] S.K. Zhou, R. Chellappa, and W. Zhao. *Unconstrained Face Recognition*. Springer, New York, 2006.