Master's Thesis

3D Reconstruction of Human Faces from Reflectance Fields

Department of Electrical Engineering
Linköping University

Erik Johansson

Reg no: LiTH-ISY-EX-3444-2004
Linköping 2004

Examiner: **Ingemar Ragnemalm, Linköping University.**

Abstract

Human viewers are extremely sensitive to the appearance of peoples faces, which makes the rendering of realistic human faces a challenging problem. Techniques for doing this have continuously been invented and evolved since more than thirty years.

This thesis makes use of recent methods within the area of image based rendering, namely the acquisition of reflectance fields from human faces. The reflectance fields are used to synthesize and realistically render models of human faces.

A shape from shading technique, assuming that human skin adheres to the Phong model, has been used to estimate surface normals. Belief propagation in graphs has then been used to enforce integrability before reconstructing the surfaces. Finally, the additivity of light has been used to realistically render the models.

The resulting models closely resemble the subjects from which they were created, and can realistically be rendered from novel directions in any illumination environment.

Keywords: 3D reconstruction, image based rendering, light stage, reflectance fields, rendering faces, shape from shading

Contents

1. Introduction

1.1. Background

The rendering of realistic human faces, is and has been, a challenging problem within computer graphics for many years [11]. Faces have complex and individual shapes, skin has subtle and spatially varying reflectance properties and the face deforms in a complex way when moving. In addition, human viewers are very sensitive to the appearance of other peoples faces.

Paul Debevec visited the SIGRAD 2002 conference in Norrköping where he, among other things, presented his work on realistic image based rendering and lighting of non-animated faces [5]. The effectiveness of the method has been demonstrated in the film "The Matrix Reloaded", [3]. The relative simplicity of his method raised the curiosity of the author. By simple means, would it be possible to use this method to increase the realism of animated faces in computer graphics? In order to find out, it is necessary to have a model to animate. This thesis examines a method to construct a facial model from reflectance field data acquired from a human face.

1.2. Purpose

The purpose of this thesis is to construct a three dimensional model of the surface of a human face given a measurement of its reflectance field.

1.3. Document Overview

This document is divided into four parts. The first part, Chapter 2, introduces earlier work, on which this thesis depends. The second part, Chapter 3 and 4, contains the model assumption and details how measurements were made. The third part, Chapter 5, 6 and 7, describes the methods used to interpret reflectance field data to construct a facial model. Finally, the fourth part, Chapter 8, shows the results and discusses possible directions of future work.

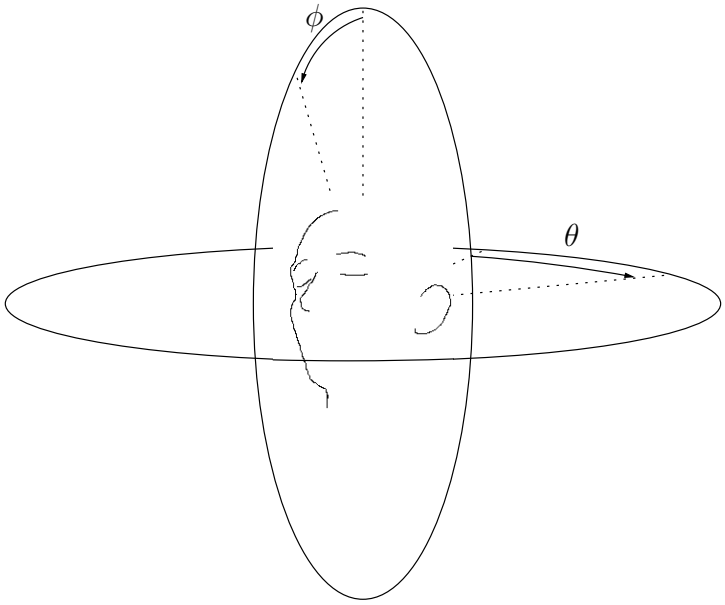

Figure 1.1. *The spherical coordinate system. This coordinate system is used to specify directions throughout the document. As is illustrated in the figure, the origin of the ϕ-coordinates is at the top of the sphere and the origin of the θ-coordinates is behind the subject.*

1.4. Definitions

The coordinate system used is described here together with a couple of expressions that may be new to the reader and that have not been defined elsewhere in the document.

This thesis makes use of a spherical coordinate system, Figure 1.1. The origin of the ϕ-coordinates is at the top of the sphere. The θ-coordinates origin is behind the subject.

Dynamic range Dynamic range is the difference in intensity between the lightest and the darkest part of a scene. Scenes lit by direct sunlight always have a very high dynamic range, since the intensity of the sunlight is many orders of magnitude greater than the intensity of, for example, light reflected from grass.

Light maps Light maps are images of incident light. They contain the same

information as a light probe, Section 2.2, resampled and projected onto a square two dimensional surface.

1.5. Notation

$R(\,)$	The complete six dimensional reflectance field function as defined in Section 2.1.1.
$R_x(\,)$	Four dimensional subset of the reflectance field function. The camera angle is implicit.
$R_{xy}(\,)$	Reflectance map function, which is a two dimensional subset of the reflectance field function, where both camera angle and point on the surface are implicit.
$g(x, y)$	Gradient map, a vector image, with one estimated surface normal per pixel.
ϕ	Refers to the angle of incident light, using the coordinate system described in Figure 1.1 (no subindex).
θ	Refers to the angle of incident light, using the coordinate system described in Figure 1.1 (no subindex).
$\bar{x}$	Unnormalized vector (bar).
$\hat{x}$	Normalized vector (hat).
CCD	Charge-Coupled Device, a technique commonly used in the light sensitive sensors on digital cameras.
SLR	Single Lens Reflex, a technique that allows a photographer to view a scene through the same lens as the film.

2. Related Work

This section gives an introduction to the methods used to capture reflectance field data from human faces and realistically relight the faces.

2.1. Introduction to Reflectance Fields

The traditional way of rendering faces is to texture-map a photograph of the face onto a geometric model. Faces rendered this way usually fail to appear realistic when changes are made to lighting, viewpoint or expression. The problem is that the reflectance properties of the face are complex, since skin reflects light both diffusely and specularly. Both of these reflection components are also spatially varying. A method that allows relatively accurate rendering of the complexities of an individual's face under arbitrary changes of lighting and viewing direction is described in [5]. That method is based on recorded imagery, where the central device is the light stage, Figure 2.1, which enables illumination of a subject from a dense sampling of directions of incident light. The effect of the illumination on the subject can be recorded using one or more stationary video cameras.

Due to the additive nature of light [4], Figure 2.2, it is possible to use illumination data to render the subjects under completely novel illumination conditions by simply computing linear combinations of the original images. Given measurements of how light from virtually all different directions affect a subject, we can assemble reflectance maps, Figure 2.3. Reflectance maps describe how light from different angles affect each point of the skin of a subject. We can then use the information contained in the reflectance maps to relight the subject under any illumination condition. The work flow can be seen in Figure 2.4. Rendering the subject in this way correctly reproduces all of the effects of diffuse and specular reflection, as well as interreflections between parts of the face.

2.1.1. Definition of the Reflectance Field

A reflectance field consists of reflectance maps, just like an image consists of pixels. This thesis only deals with non-local reflectance fields, where the

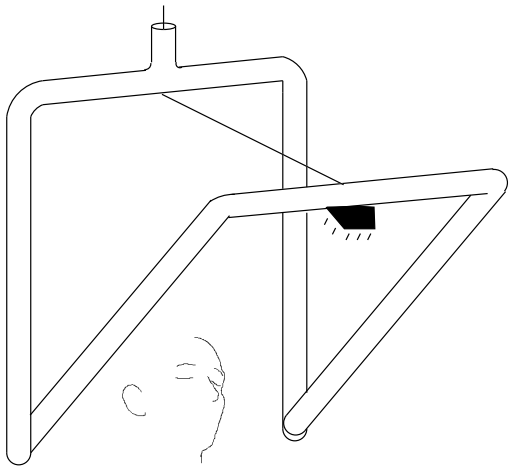

Figure 2.1. *The light stage concept. A light source is connected to two different bars, one that can spin around the ϕ-axis and one that can spin around the θ-axis. Using this setup it is possible to move the light source along a spherical trajectory.*

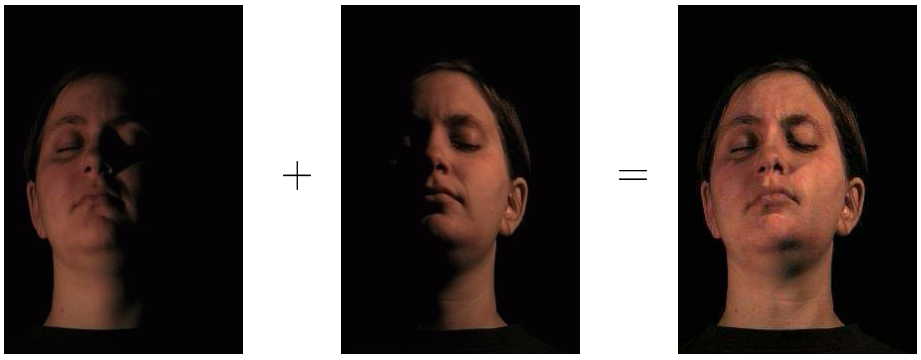

Figure 2.2. *Example illustrating the additivity of light. The subject, Anna, has been photographed lit from two different directions. A novel lighting condition can be created by adding the resulting images.*

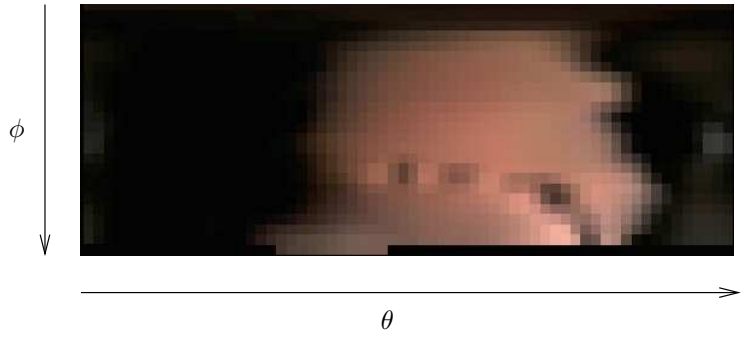

Figure 2.3. *Example of a reflectance map. A point on the facial skin has been lit from many different directions. Each pixel in the reflectance map displays the color achieved by illuminating that point from the corresponding direction.*

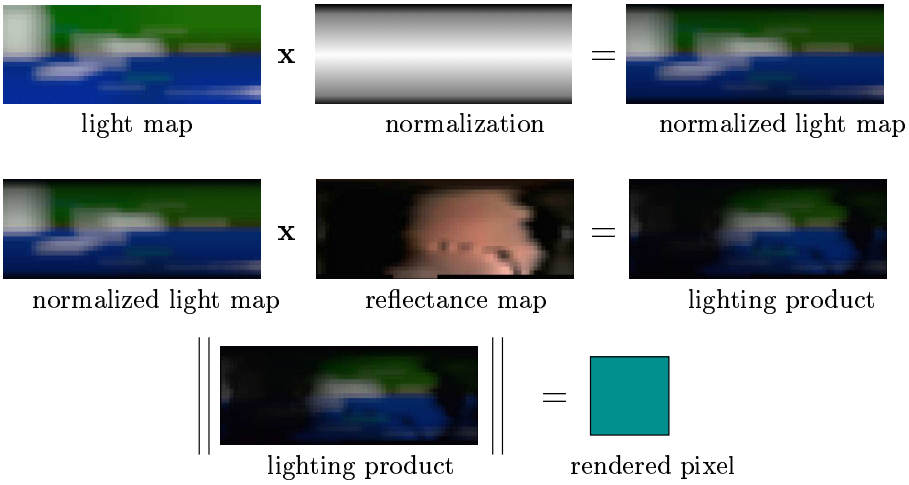

Figure 2.4. *Work flow when creating a new virtual illumination. First, the light map has to be normalized to compensate for the over-representation of samples from above and below the subject. The normalized light map is then multiplied pixel by pixel with the reflectance map. Finally, the result of the multiplication is summed into a single value which is the rendered pixel value. The process is repeated for the entire reflectance field in order to create an image of the subject under a novel lighting condition.*

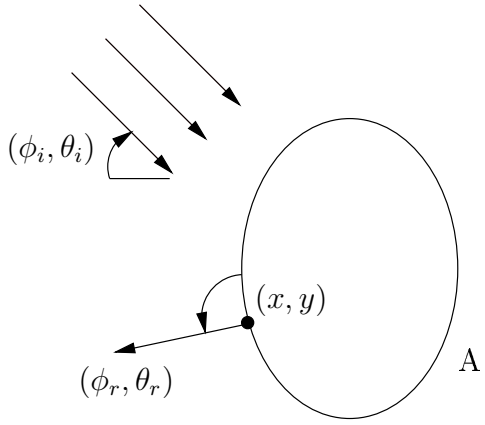

Figure 2.5. *Simple illustration of the reflectance field for the surface A. The reflectance field has the following parameters: The direction of incident illumination, (ϕ_i, θ_i), the point on the illuminated surface, (x, y) and the direction of the camera (ϕ_r, θ_r).*

incident illumination field originates far away from the measured surface A, so that all points on the surface can be considered to be lit by the same illumination field at any given time. This gives a six dimensional function $R(\phi_i, \theta_i, \phi_r, \theta_r, x, y)$ where (ϕ_i, θ_i) is the direction of the incident illumination, (ϕ_r, θ_r) is the direction of the viewpoint and (x, y) denotes the point on the surface A, see Figure 2.5. Since only one camera is used, the light stage will be used to sample a four dimensional subset of the reflectance function R from a specific viewing direction (ϕ_r, θ_r). As described in the notation, these four dimensional subsets will be referred to as R_x. Individual reflectance maps will be referred to as R_{xy}. The surface A is the model of the face.

2.2. Recovering Illumination Environments

Captured illumination environments from real scenes, light probes, will be used to render the results. In order to understand the necessary steps, the process will be briefly described here.

Almost the entire illumination environment in a particular point will be reflected by a mirroring sphere placed in that point, see Figure 2.6. By photographing the sphere from two directions, 90 degrees apart, the entire

Figure 2.6. *The mirroring sphere used to capture illumination environments. The reflection from the area behind the sphere is compressed near the edges of the sphere, and will be captured at a lower resolution than the area in front of the sphere.*

illumination environment can be measured with a reasonable resolution.

Scenes lit by direct sunlight or scenes with shiny materials and man made light sources often have such extreme differences in radiance values that they are impossible to capture without either under-exposing or saturating the film. One way to capture the full dynamic range in such a scene is to take a series of photographs with different exposures. However, there is a catch, photographic film, as well as the CCD-sensors in digital cameras, have non-linear response curves. The most significant nonlinearity is at the saturation point where any pixel with radiance above a certain level is mapped to the same maximum image value. A description of how to recover illumination environments using only a set of photographs with varying, known exposure can be found in [7]. Fortunately, there is free computer software available that will help us with the recovery of illumination environments [16].

2.3. Facial Modeling

There are many ways to model faces, for example, structured lighting, which uses a grid of light projected onto the surface to be modeled. There is also equipment, such as laser-triangulation scanners made by Cyberware, available for the purpose of modeling. However, these methods require additional measurements, that is, other measurements than the reflectance field. To build a model from a reflectance field we first have to find the surface normals, a shape from shading problem. Lots of work has been done within this area [18], but far from all methods are immediately applicable on the data contained in a reflectance field. Two methods that are of particular interest are [14, 17]. However, the work is not done simply because the surface normals have been determined. A surface that best fits those normals need to be constructed. Some of the work that has been done within the area of surface reconstruction is presented in [12, 13].

3. Model Assumption

3.1. Human Skin

The reflectance properties of real human skin are very complex. It is possible to identify a few of the components that make up the reflectance function of skin, but it is impossible to model all. For example, the color of the skin varies with the temperature and the mood of the person. Two components that can be identified are the specular reflection from the skin surface, and the diffuse reflection caused by subsurface scattering. The specular reflection behaves predictably, the diffuse reflection is however more difficult to model. The diffuse reflection is caused, among other things, by subsurface scattering in the epidermal and dermal layers of the skin. Lots of work in the area of examining human skin reflectance have been done, not only by the computer graphics community [15].

3.2. The Phong Model

The skin of the subjects is assumed to have a reflectance function that is a version of the phong reflectance model. An elaborate description of the phong model can be found in, for example, [2]. In a somewhat simplified form, the phong reflectance model can be written,

$$C = Ik + I\alpha\left(\hat{n}\cdot\hat{l}\right) + I\beta\left(\hat{v}\cdot\hat{r}\right)^{m} \tag{3.1}$$

The different vectors are illustrated in Figure 3.1. The scalar I represents the chromaticity of the light source, C represents the light emitted from the surface, in the direction of observation $\hat{v}$. There are three terms; the ambient reflection Ik, the Lambertian reflection $I\alpha\left(\hat{n}\cdot\hat{l}\right)$ and the specular reflection $I\beta\left(\hat{v}\cdot\hat{r}\right)^{m}$. Since no ambient light is present in the measurements, the first term can be discarded, $k = 0$. The reflection vector, $\hat{r}$ can be calculated as

$$\hat{r} \;=\; 2\left(\hat{l}\cdot\hat{n}\right)\hat{n} - \hat{l} \tag{3.2}$$

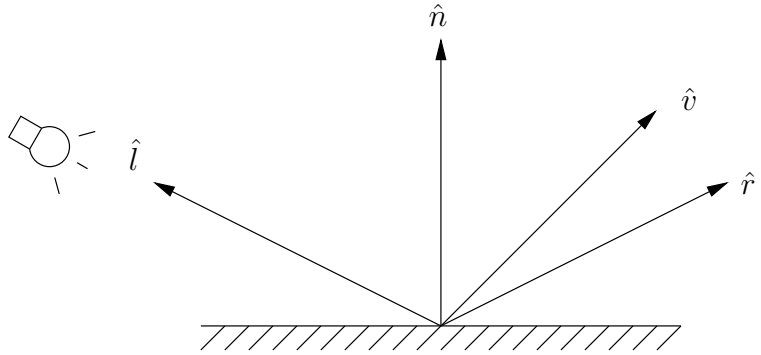

Figure 3.1. *Illustration of the relevant directional parameters of the phong model. Depicted in the figure are the following directions: The light source, $\hat{l}$, the surface normal, $\hat{n}$, the viewpoint, $\hat{v}$ and the direction of specularly reflected light, $\hat{r}$.*

Since we use color images, each constant has different values for each color, for example:

$$\alpha = \begin{pmatrix} \alpha_r \\ \alpha_g \\ \alpha_b \end{pmatrix} \tag{3.3}$$

Ideally, the measurements should be done with a white light source using color calibrated equipment, which means that $I = 1$ can be assumed for the reflectance fields. Furthermore, when looking at a particular reflectance map on the surface of a face, the surface normal, $\hat{n}$, and the specular exponent, m, are constant. Finally this leads us to the assumption that each value in the reflectance map is a function of the direction of incident light $\hat{l}$, and the camera angle $\hat{v}$,

$$R(\hat{l}, \hat{v}) = \alpha \left(\hat{n} \cdot \hat{l} \right) + \beta \left(\hat{v} \cdot \hat{r} \right)^m \tag{3.4}$$

4. Equipment and Measurements

This section describes equipment and methods. Section 4.1 – 4.3 details the acquisition of facial reflectance fields. Section 4.4 deals with illumination environments. The descriptions are intended to be detailed enough to enable a person, interested in using this method or equipment, to make new measurements.

4.1. The Light Stage

The apparatus described in this section may appear more complex than it actually is. It was built by the author, who has an academic background, in about a week.

4.1.1. Description

The light stage is a construction that makes it possible to move a light source in a spherical trajectory around a subject, see Figure 2.1. The idea is to place the subjects face in the center of the sphere described by the light source's path. The light source is then moved while the effect of the different directions of light is recorded by a video camera. This provides a densely sampled set of data describing how light from virtually all different directions contribute to the appearance of the subject.

Since we only want to acquire the effect of different directions of incident light, it is important that the subject does not move during the capture. Mathematically speaking, we only want to sample the value of the reflectance function $R_x(\phi, \theta, u, v)$ for different values of ϕ and θ, u and v must not change.

The light source is an ordinary halogen light with a nominal effect of 100W, commonly used on construction sites. The high intensity of the light emitted causes a number of problems. Most notable is the secondary reflection of light from walls and ceiling. To avoid this the light source is connected to a dimmer that allows adjustment of the intensity of the light source.

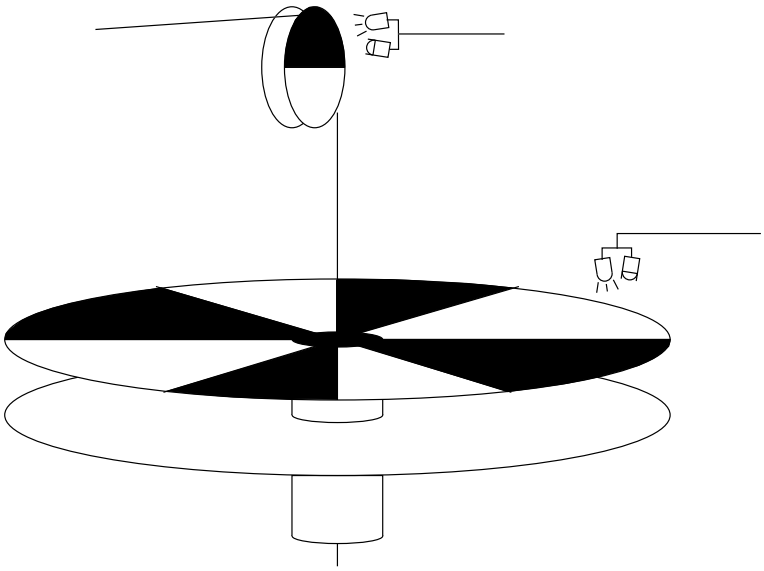

Figure 4.1. *The angular probe. The angular probe is located on top of the bars shown in Figure 2.1. The big horizontal disc rotates with the θ-bar, the smaller wheel rotates as the cord to the ϕ-bar is being pulled. Using infrared sensors, to keep track of reflectors on the rotating surfaces, it is possible to calculate the location of the light source.*

4.1.2. The Angular Probe

The light stage is propelled by ropes. One rope is wound around the θ-axis, the second rope is actually the power cord for the light source and is used to move along the ϕ-axis. When the ropes are pulled, the light source spins around. Since the ropes are manually pulled the exact trajectory of the light source is hard to predict as it varies from one session to the next. In order to keep track of the location of the angle of incident light, an angular probe that records the ϕ and θ angles is needed.

The angular probe has two sensors, Figure 4.1. Each sensor consists of an infrared-diode and an infrared-sensor. Reflectors on the light stage will pass in front of these sensors as the light stage is propelled around the ϕ and θ axises. As data is being collected, it is sent by a serial cord to a computer. A detailed specification of the serial interface can be found in Appendix A.

Since the computer is not a real-time system, data could potentially be skewed by delays in input caused by the computer, for example, memory

swapping. To solve this, data is sent at a prespecified rate. By manually aligning the collected serial data with the video sequence at one single point, it is possible to determine the correct angle for each frame without having to use a real-time system.

Information about the position of the light source is made available by the edges of each reflector. Apart from when edges pass in front of a sensor, no information about the exact position is available, and the position has to be interpolated. The acceleration of the light source is assumed to contain discontinuities, which means that we want to interpolate the position of the light source with a C^1-continuous spline. Interpolation was done using a Catmull-Rom spline [8]:

$$\begin{aligned} p(t) \quad &= \quad \tfrac{1}{2}\left(p_{-1}\left(-t+2t^2-t^3\right)+p_0\left(2-5t^2+3t^3\right)+\right. \\ &\qquad \left.+p_1\left(t+4t^2-3t^3\right)+p_2\left(-t^2+t^3\right)\right) \qquad (4.1) \\ t \in [0,1] \end{aligned}$$

Endpoints are generally not a problem since measurement data is available both before and after the desired data set.

4.1.3. Discussion

The light stage was set up in a bomb shelter below the university building, which also serves as an archive. This had some adverse effects. The somewhat bizarre location made some of the subjects feel uneasy. Most notably, the height of the ceiling was limited. Secondary reflection from the ceiling became visible in the measured reflectance maps, Figure 4.2, and the radius of the sphere, that the light source moves along, had to be reduced to about 1 m. Ideally a radius of about 1.5 meters should be used which would allow the subjects to sit in a comfortable position, as well as increasing the distance between the subject and the light source.

Since the light from the light source is neither perfectly evenly distributed, nor perfectly white, it would be a good idea to calibrate it, thus making it possible to compensate the measurements for these errors. This could be done with the following procedure:

1. Place a flat white surface at the same distance from the light source as the face will be in the light stage. Make sure the surface is perpendicular to the direction of the light source.

2. Illuminate the surface with the light source.

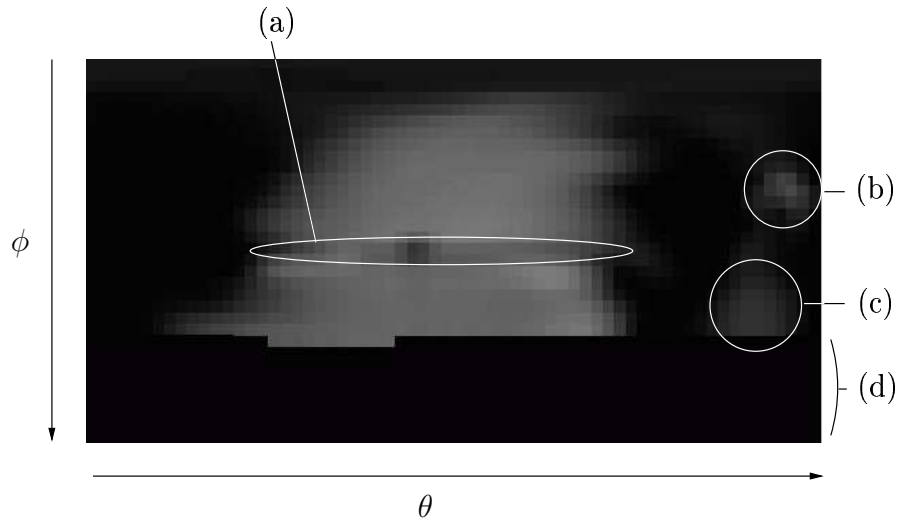

Figure 4.2. *Illustration of errors present in the reflectance maps. Four different kinds of errors can be seen in the reflectance map: a) Occlusion caused by parts of the light stage passing in front of the camera. b) Lens flare caused when the lens in the camera is hit by direct light. c) Secondary illumination from light that has bounced off the ceiling. d) Unmeasured area, because of restrictions on how far the ϕ-bar on the light stage can be lowered.*

3. Take a picture of the illuminated surface with a color calibrated camera.

4. The image of the surface can now be used to calculate the errors caused by imperfections in the light source.

This compensation was not done because of two reasons. First, the image of the flat white surface showed that the light source produces a very even illumination at the relevant distance. Second, the camera used for capturing the video sequence could not be easily color calibrated, making compensation for the color of the light source meaningless, see Section 4.2.2.

Other errors introduced into the reflectance maps are shown in Figure 4.2 and listed below, with numbering corresponding to the figure.

a) Occlusion because parts of the light stage pass between the subject and the camera.

b) Lens flare which occurs when light hits the cameras lens directly.

c) Secondary illumination from light that has bounced off the ceiling.

d) Unmeasured region because of the subjects body blocking the path of the light source.

4.2. The Network Camera

4.2.1. Description

The camera used to sample the subject's reflectance fields was an Axis 2130 network camera. It is basically the same thing as an ordinary web camera, with the exception of a built-in web-interface. For all measurements, the camera has been placed in front of, and slightly below the face, $\left(\phi_r = \frac{2}{3}\pi, \theta_r = \pi\right)$. The reason for not placing the camera straight in front of the face is that the occlusion, caused by the light stage, is significantly smaller when the camera is lowered. The effect is that the camera can "see below" the ends of the bar holding the light source.

When setting up the camera for a measurement, the aperture was determined by placing the light source in the position that gives the highest intensity of reflection and adjusting the aperture until saturation is no longer visible in the video stream. Since the camera takes images with a 4:3 ratio it was tilted ninety degrees to better fit the geometry of the face. This effect was compensated by rotating the resulting images prior to the reflectance field assembly.

4.2.2. Discussion

This camera has many interesting features, though none of them were particularly useful with the light stage. Instead, this camera has three major drawbacks:

1. The web-interface allows for adjustment of the cameras direction, focus, zoom and aperture. Unfortunately, no way of acquiring video sequences from this camera while using a prespecified white balance was found, which makes color calibration impossible. Additionally, the gamma correction applied by the camera is not specified, see Appendix B. This would be unacceptable in a production environment, but since the critical algorithms are based on intensity rather than color, it is still possible to use this camera to make a proof of concept. Fortunately, the automatic white balance did a reasonably good job so the

rendered results will still have visual appeal, though it wont be useful to compare the results with actual photographs, as done in [5].

2. The camera delivers its "video" as a stream of compressed jpeg images. Since the camera does jpeg compression, each image of the face, under a specific illumination, will contain compression artifacts. These artifacts are optimized to be as invisible as possible when viewed as an image. Unfortunately, the effect on the accuracy of the reflectance field values is most likely relatively larger. Because of this, the camera was set to compress the images as little as possible, while maintaining the highest possible frame rate.

3. The biggest drawback with this camera is that it doesn't provide a constant frame rate. Compression of, for example, images that contain a lot of black is faster than images with more complex content, hence the frame rate was faster when the illumination came from angles that leaves the face mostly black. This was very unfortunate, since the technique behind the angular probe, Section 4.1.2, assumes that the camera uses a constant frame rate. If the assumption of evenly spread images over time does not hold, the angles can not be determined by simply synchronizing the serial data with the video sequence. The solution to this problem was to manually synchronize the serial data with the video sequence at two points on the θ-axis, for every lap the light source spun. These specified points has then been used for synchronization with data from the angular probe. Constant frame rate has been assumed between each pair of synchronization points. This is by far the single largest cause of error in the reflectance field measurements.

4.3. Acquiring the Reflectance Fields

Reflectance fields from eight different persons has been acquired and used, see Figure 4.3. These people are personal friends of the author and others who were available and kind enough to participate in measurements. As far as possible, given the circumstances, persons with different physical attributes has been measured.

The actual acquisition of a reflectance field requires three participants. One person to be measured and two others to run the light stage, one that controls the ϕ-cord and one that controls the θ-cord. The following procedure (checklist) was used when measuring:

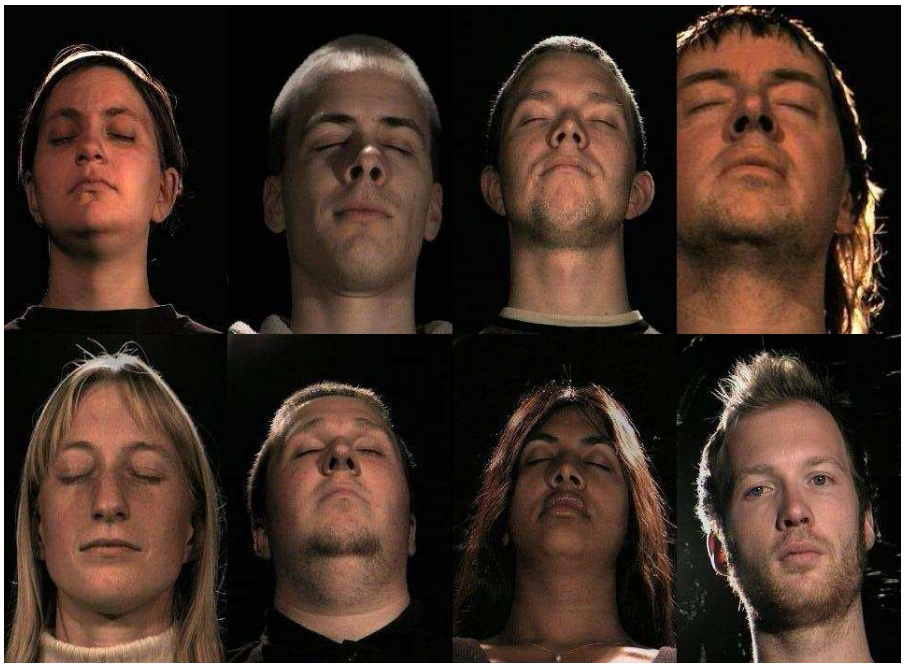

Figure 4.3. *The subjects measured. As far as possible, individuals with different skin tone, hair length and hair color have been selected.*

1. Place and connect the cameras to the computer. Connect the serial line from the angular probe on the light stage to the computer.
2. Wind up the cord needed to run the light stage.
3. Place the subject in the light stage.
4. Adjust the cameras to get the correct image, make sure that the exposure will not be changed during the acquisition.
5. Turn on the light source and the angular probe on the light stage.
6. Turn off the surrounding light.
7. Run the light stage.

With the equipment used, about 3000-4000 samples have been acquired from one measurement.

4.4. The Illumination Environment Probe

The illumination environment probe, discussed in Section 2.2, is described here. The illumination environments are needed to create light maps, as seen in Figure 2.4. A shiny mirroring ball has been used to make probes for illumination environments, Figure 2.6. The ball was an 80 mm steel ball, intended for use in ball bearings. The camera was an EOS 1n SLR camera with a Tamron 24 – 135 mm lens. Fuji Sensia 100 positive film was used for imaging. Images were taken from directions 90 degrees apart and merged into high dynamic range images using HDRShop [16]. The high dynamic range images were then used to create the light probe. Images were scanned with an Epson Perfection 1660 Photo scanner.

4.4.1. Calibration of the Imaging Process

The color response curve depends on several different things; the film, the scanner, the settings on the scanner while scanning, etc. Nonlinearities are primarily induced by the film and the scanner or by the CCD, in the case of a digital camera. The color response curve compensates for all these things. For the actual calculation of the color response curve HDRShop [16] was used. The steps involved in the calibration process are outlined below:

1. Determine the shutter speeds used by the camera. The shutter speed of modern SLR cameras tend to differ a bit from the speed that the camera claims to use. Shutter speeds of e.g. $\frac{1}{64}$ s , $\frac{1}{128}$ s and $\frac{1}{256}$ s may be used instead of the speeds $\frac{1}{60}$ s, $\frac{1}{125}$ s and $\frac{1}{250}$ s that are commonly found on cameras. The actual shutter speed can be determined by recording the sound the camera makes when making an exposure. The recording can then, for example, be analyzed in Matlab, where the number of samples between the noise of the shutter opening and the shutter closing will enable us to calculate the actual shutter speed.

2. Several pictures of a scene are taken at different known exposures, using the film that is to be calibrated. The closer the exposures the better. Avoid modifying the aperture during the process, since that will not only affect the exposure, but also the depth of field. Preferably, the scene should contain bright and dark regions, as well as values in between.

3. Digitize the pictures using the particular scanner, and scanner settings, that are to be calibrated.

4. Feed the resulting images and shutter speeds into the Calibrate Camera Curve feature of HDRShop [16].

Obviously, if a digital camera is used, step three in the process outlined above is omitted.

4.4.2. Acquiring the Illumination Environments

Light probes have been made in a number of different locations. Both indoor and outdoor environments have been measured. Care has been taken to measure environments with soft light, as well as environments with strongly directional light. An artificial environment with light of three different colors has also been used. The method is described in Section 2.2. The steps involved are outlined below:

1. Place the mirroring sphere at the location where the light environment is to be measured.

2. With the use of a light meter, find the highest and lowest intensities of light present at the location.

3. Using the calibrated SLR camera, make exposures of the sphere covering all light from the highest to the lowest intensity. Use three steps of exposure difference between images.

4. Move the camera 90 degrees relative to the first position and make another set of exposures as in step 3. Since almost the entire environment is reflected in the ball, images from one direction could be enough. The reason for using images from two different directions, is to get rid of the reflection of the camera, and to avoid the poor resolution of light coming from behind the sphere.

5. Make at least one ordinary exposure of the scene to use as virtual background.

6. Use HDRShop [16] to merge the set of images from step 3 & 4 into two high dynamic range images of the mirroring sphere.

7. Use HDRShop [16] to merge the two high dynamic range images into one light probe.

5. Finding the Face

This section describes the method used for elimination of unwanted parts of the measured reflectance field. Specifically, that means eliminating data from the background and hair, in order to operate only on data that belongs to the actual skin surface of the face. This is done in two steps; thresholding followed by segmentation.

5.1. Description of the Problem

A reflectance field, as measured by the light stage, can be considered to consist of three different types of reflectance maps; background, skin and hair. The background is mostly black, only errors from flare contribute any values to the reflectance map. Skin adheres relatively well to the phong model and contribute values to the reflectance map from virtually all illumination angles that hit it. Hair does not adhere well to the phong model. It consists mostly of two rather narrow spikes, a specular spike and another spike caused by the light source shining through the hair from behind. If the intensity of all samples contributing to each reflectance map is summed, we get an image displaying the summed reflectance map intensities, Figure 5.1. As expected, it is evident that the summed reflectance map intensity for skin surfaces is notably higher than that for hair and background. This difference in summed intensity can be used to find the skin area.

Two different subjects are used to illustrate how this technique operates. A the first subject was a caucasian male with very short hair. His ears are visible and the absence of hair, blocking the light from behind, causes strong specular values to contribute a lot of intensity to the reflectance maps originating from the side of the face. The other test subject was a colored female with lots of hair visible. To make useful measurements of a subject with long hair, the hair will have to be tied back in a pony tail. The measurements of the female illustrated here was made with the hair hanging partly in front of the face. This is not good for rendering realistic images, but an interesting measurement for pushing the limits of this technique.

Figure 5.1. *Images of summed reflectance map intensities. When the intensities of all samples in a subjects different reflectance maps are summed into corresponding pixel values, and the dynamic range of the result is scaled to a displayable format, this is the result. The female subject's hair was not tied back during measurement in order to test the algorithm. Normally, a measurement where the hair covers the face would be useless.*

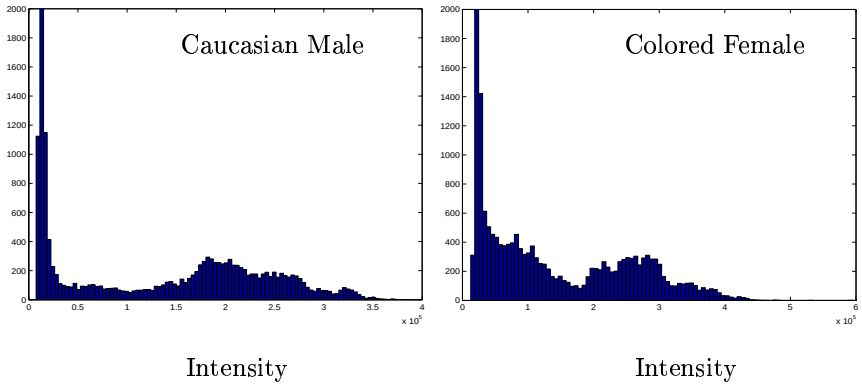

Figure 5.2. Histograms of the images shown in Figure 5.1.

5.2. Calculating the Threshold Value

The correct threshold value can be determined by studying the histogram of the summed reflectance map intensities. A histogram function with 100 buckets was used, the resulting histograms was then smoothed by convolution with an 11 samples long averaging kernel:

$$\frac{1}{11}[1\,1\,1\,1\,1\,1\,1\,1\,1\,1\,1] \tag{5.1}$$

The size and shape of the kernel are chosen as they are because it was the first approach tried when smoothing the histograms, and it proved to work very well. It seems likely that any low-pass kernel would work.

The resulting curves can be seen in Figure 5.2 and 5.3. Looking at the curves we note two distinct peaks, the leftmost high and narrow peak, originates from the background, the rightmost, a more distributed peak, originates from the skin. The effect of the hair lies in between the peaks. Fortunately, there is a distance between skin and hair which is indicated by the local minimum to the left of the skin peak. That local minimum makes a good thresholding value. The steps of the algorithm are outlined in Algorithm 1.

5.3. Thresholding and Segmentation

Once we have the thresholding value we can threshold the image. Simply thresholding the image based on intensity values would eliminate areas such

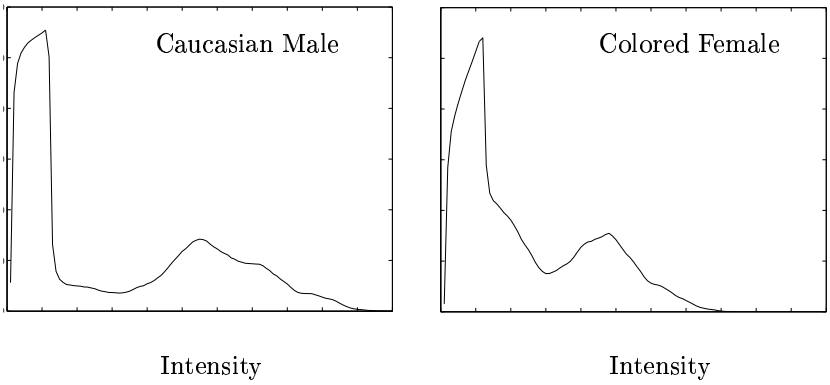

Figure 5.3. *Smoothed histograms. The graphs are obtained by convolving the histograms from the images in Figure 5.1 with an averaging kernel. Two distinct peaks can be seen; one high and narrow to the right, made up of low intensity contributions from the background, and one lower, more distributed, on the left, made up of brighter pixels from the skin surface.*

Algorithm 1 Algorithm for finding the threshold value to use when separating the skin from hair and background.

```
1. sum reflectance map intensities
2. make histogram with 100 buckets
3. convolute histogram with 11 sample averaging kernel
4. find local maxima in the resulting curve
5. find the first local minimum below the second highest maximum
```

Algorithm 2 Algorithm to connectively threshold an image.

```
1. scan the reflectance field from upper left to lower right
2.   if intensity value < threshold
3.     if connected to edge or marked neighbor
4.       mark reflectance map as background
5. scan the reflectance field from lower right to upper left
6.   if intensity value < threshold
7.     if connected to edge or marked neighbor
8.       mark reflectance map as background
```

as eyebrows and nostrils, which we want to keep. Some more knowledge of the problem has to be used to get rid of the background and hair without loosing areas that are surrounded by skin. The skin is connected, and so is the area that we want to get rid of. Each reflectance map originating from hair or background is connected to either the edge of the reflectance field or another reflectance map belonging to hair or background. Because of this, segmentation is done in two passes over the image. The first pass from upper left to lower right, the second from lower right to upper left. For each reflectance map we check whether its summed intensity is above or below the threshold. If it is below, we check if this reflectance map is four-connectively connected to either the edge of the reflectance field or another reflectance map which has been marked as background. For pseudo code, see Algorithm 2.

Since the background is four-connected, the remaining area is eight-connected. There is also a small possibility that "islands" exist that are not connected to the face at all. For modeling purposes one single four-connected area is desired, hence the image is segmented (four-connectively) and only the largest segment is kept. The resulting area can be seen in Figure 5.4. As can be seen in the figure, the eyebrows of the female subject have been classed as hair. This may seem incorrect at a first glance. However, in this case they are connected to the rest of the hair, which means that they will be removed. It can also be noted that the sides of the neck has been removed. It is due to the large amount of hair shadowing this area from most directions of light. As has been mentioned earlier, this measurement is not good for anything other than testing the method. This method works for all the subjects measured.

Figure 5.4. *The estimated skin surface of the faces. The algorithm has removed reflectance maps belonging to the background and the hair, pixels corresponding to those reflectance maps have been set to black. Worth noticing is that the eyebrows of the female subject have been removed, which is not strange when considering that they were connected to the rest of the hair, see Figure 5.1.*

6. Analyzing the Reflectance Maps

This chapter describes various operations performed on the reflectance maps, in order to analyze the information they contain. The information obtained will later be used for reconstruction of the skin surface, and to render the results.

6.1. Estimation of Surface Normals

The first step in the reconstruction of the facial models is to estimate the surface normal for all points on the surface. Each surface normal will be estimated from one reflectance map, independent of surrounding reflectance maps. For simplicity, the reflectance function of the skin is assumed to be purely Lambertian, which means that the intensity i, of each sample in the reflectance map is a function of the direction of incident light $\hat{l}$, and the surface normal $\hat{n}$:

$$i = \alpha\left(\hat{l} \cdot \hat{n}\right) \tag{6.1}$$

As described in [17], the reflectance map provides a set of samples with known values of i and $\hat{l}$, which means that finding $\alpha \cdot \hat{n}$ is a least mean square problem:

$$\begin{aligned} \begin{pmatrix} i_1 \\ \vdots \\ i_n \end{pmatrix} &= \alpha \begin{pmatrix} \hat{l_1}^\top \\ \vdots \\ \hat{l_n}^\top \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} \\ I &= \alpha L \hat{n} \\ \left(L^\top L\right)^{-1} L^\top I &= \alpha \cdot \hat{n} \end{aligned} \tag{6.2}$$

We must remove samples that originate from directions that does not illuminate the surface corresponding to the current reflectance map. This is done by thresholding values that have zero, or very small, intensity. Specular reflection and errors are also present, which means that some contributions to the intensity will be incorrect. After an initial estimate of the surface normal, we can eliminate samples that deviate too much from the initial

Algorithm 3 Algorithm to estimate a surface normal from a reflectance map.

```
1. threshold to get rid of "black" samples
2. solve the LMS equation
3. while deviating samples exist
4.    remove deviating samples
5.    solve new LMS equation
```

estimate and reestimate the surface normal. This can be iterated until no more samples deviate and we have estimated a surface normal from the samples that best fit the Lambertian model. For pseudo code, see Algorithm 3.

After doing this for each reflectance map we have a gradient map $g(x, y)$, which is a vector image with one estimated surface normal per pixel.

6.2. Algorithm for Separation of Specular and Lambertian Components

In order to realistically render models from novel viewing directions, the specular and the diffuse components of the reflectance function has to be separated. This algorithm was developed in order to try to separate the Lambertian and specular components assuming that the reflectance maps adhere reasonably well to the phong model. The algorithm assumes that the surface normal $\hat{n}$, is known or estimated in an earlier step, see Section 6.1. The algorithm has the following steps:

1. Remove any samples in the reflectance map where the direction of incident light $\hat{v}$, reveals that it cannot illuminate this surface. $\hat{v} \cdot \hat{n} < 0$

2. Remove any samples where the intensity i, is close to zero. These samples may result from occlusion or self shadowing and should not be considered.

3. Create a vector

$$\bar{I} = \begin{pmatrix} i_1 \\ \vdots \\ i_n \end{pmatrix} \tag{6.3}$$

with the intensities of the remaining samples, and a vector

$$\bar{L} = \begin{pmatrix} \hat{l_1} \cdot \hat{n} \\ \vdots \\ \hat{l_n} \cdot \hat{n} \end{pmatrix} \tag{6.4}$$

with the Lambertian contribution of each sample.

4. Assume a value of the specular exponent, for example, $m = 14$.

5. Create a vector

$$\bar{S} = \begin{pmatrix} (\hat{r_1} \cdot \hat{v})^m \\ \vdots \\ (\hat{r_n} \cdot \hat{v})^m \end{pmatrix} \tag{6.5}$$

with the specular contribution of each sample.

6. Ideally, the following relationship should be true:

$$\bar{I} = \alpha\bar{L} + \beta\bar{S} = \begin{pmatrix} \bar{L} & \bar{S} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \tag{6.6}$$

The coefficients α and β are calculated by solving the least square equation:

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \left(\begin{pmatrix} \bar{L} & \bar{S} \end{pmatrix}^\top \begin{pmatrix} \bar{L} & \bar{S} \end{pmatrix} \right)^{-1} \begin{pmatrix} \bar{L} & \bar{S} \end{pmatrix}^\top \bar{I} \tag{6.7}$$

7. A measure of how well the estimated model fits the reflectance map can be calculated as:

$$e = \left\| \bar{I} - \begin{pmatrix} \bar{L} & \bar{S} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \right\|^2 \tag{6.8}$$

8. Go back to step 3, try surrounding values of m and keep going in the direction that gives the lowest error e, until a local minimum is found.

Empirical studies show that the search criteria in step 8 reaches a local minimum for the majority of all reflectance maps.

This algorithm does not work very well for the measurements used, the coefficient β for the specular component is computed to be too high, while

the exponent m is too low for many reflectance maps. There are two possible reasons for this. First, the incident directions of light $\hat{v}$, are relatively inaccurate, which smears out the specular component. The other reason is that the skin does not adhere very well to the phong model. For example, there are effects from subsurface scattering, and interreflections from different parts of the face, that are not accounted for.

6.3. Calculating the Appearance under Novel Illumination

Since no animation of the model is done, rendering the result is a simple matter of creating a texture map from the reflectance field, which is then mapped onto the model. Calculation of each pixel on the texture map is done as outlined in Figure 2.4 with a few exceptions. Since the light stage used cannot measure the reflectance for light coming from angles lower than $\phi \approx \frac{2}{3}\pi$, the additions of light from that area will not be included. Areas of the reflectance map with incorrect values, due to occlusion, will give incorrect contributions, so these will also be removed. Since all the undesired pixels are black or very dark, all samples whose intensity is zero or close to zero are removed. Contributions to the rendered pixel caused by flare will not be removed, nor will contributions caused by secondary reflection from the ceiling. Since we are used to seeing images affected by flare, the inclusion of flare may actually be perceived as additional realism.

When rendering the face from a novel viewpoint, the specular reflection of the skin will be incorrectly rendered. This could be compensated, by using the specular and Lambertian components of the reflectance map, to reconstruct the specular component in the correct place, before performing the calculation described above. A description of how this can be done, using a more elaborate skin model, is outlined in the second part of [5]. No such compensation was used here, the results look realistic enough, for the purpose of rendering the model, without performing this compensation.

6.3.1. Resampling the Reflectance Maps

Each reflectance map consists of a set of samples of the reflectance from various directions, as described in Section 2.1.1. These directions are not uniformly distributed, which is required in order to be able to use them together with a light map, as described in Section 2.1. The resampling of a reflectance map is straight-forward, see Algorithm 4.

Algorithm 4 Algorithm to resample a reflectance map.

```
1. kernel_radius = distance between new samples
2. for each new sample
3.   for each old sample
4.     if (old vector) · (new vector) < kernel_radius
5.       add weighted reflectance to new sample
```

The distance between the new, uniformly distributed, samples is not constant, samples are further apart around "the equator", that is, when $\phi = \frac{\pi}{2}$. For the results rendered in Chapter 8, a cone shaped kernel was used. The kernel size was calculated as the largest distance between any two neighboring samples.

7. Reconstruction of the Skin Surface

Reconstruction of the skin surface is done in three steps. First, the surface normals of the skin are translated into gradient estimates. Second, the estimated gradients are adjusted to minimize errors. Finally, the surface is created by integration.

7.1. Interpreting the Surface Normals

The observed surface normals are represented by pixels in a gradient image $g(x, y)$. Each gradient has two properties of interest, the slope in the $\hat{x}$ direction λx, and the slope in the $\hat{y}$ direction λy, see Figure 7.1.

$$\begin{cases} \lambda x &= \frac{\delta g(x,y)}{\delta x} \\ \lambda y &= \frac{\delta g(x,y)}{\delta y} \end{cases} \tag{7.1}$$

The derivatives are calculated in a left handed coordinate system where $\hat{z}$ – "upward" is the direction of the camera. The values of the derivatives are

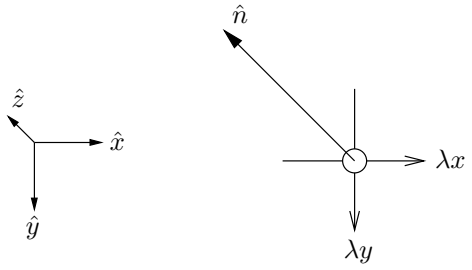

Figure 7.1. *One gradient from the gradient image $g(x, y)$, showing the coordinate system and the directions of the derivatives λx and λy.*

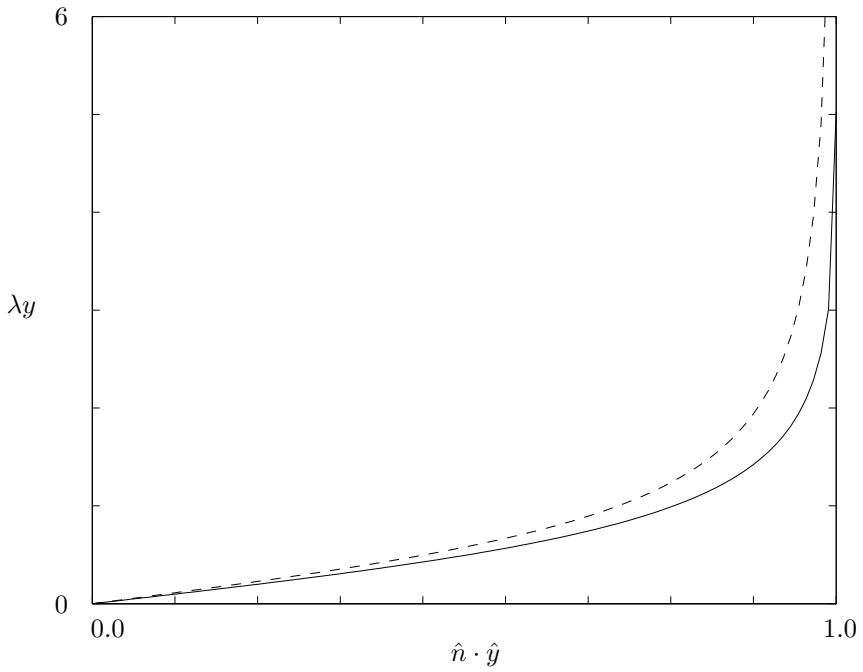

Figure 7.2. *Curves for the functions described in Equation 7.2 (dashed) and Equation 7.4 (solid). The dashed curve grows towards infinity, where as the solid curve has its maximum well below 6.*

calculated from the surface normal $\hat{n}$, in the following way

$$\begin{cases} \lambda x &= \tan\left(\arcsin\left(\hat{n} \cdot \hat{x}\right)\right) \\ \lambda y &= \tan\left(\arcsin\left(\hat{n} \cdot \hat{y}\right)\right) \end{cases} \tag{7.2}$$

Since each pixel in the gradient image represents a fixed position (x, y), there will be problems with points where the estimated surface normal is equal to either the $\hat{x}$ or $\hat{y}$ direction. Such estimates may be perfectly valid, for example, on the side of the nose, but will still cause problems when the model is integrated. The corresponding derivative will become infinite, which can be seen as a property of the tangens function, see Figure 7.2. The effect on the model building process is devastating, see Figure 7.3.

To avoid the problem of infinite derivatives, it is desirable to "flatten" the tangens curve, in order to limit the gradient values for points where the estimated surface normal is close to either $\hat{x}$ or $\hat{y}$. One way of doing that is to add an offset angle to the result of the arcsin function. However,

Figure 7.3. *A model built without compensation for extreme values of local derivatives.*

that would have a large impact on smaller angles which is undesirable. The solution is to use a gradually increasing offset.

$$\tan\left(\arcsin\left(\hat{n}\cdot\hat{x}\right)-\frac{\frac{10}{180}\pi\cdot\arcsin\left(\hat{n}\cdot\hat{x}\right)}{\frac{\pi}{2}}\right)=\tan\left(\frac{8}{9}\arcsin\left(\hat{n}\cdot\hat{x}\right)\right) \quad (7.3)$$

The derivatives are calculated as

$$\begin{cases} \lambda x &= \tan\left(\frac{8}{9}\arcsin\left(\hat{n}\cdot\hat{x}\right)\right) \\ \lambda y &= \tan\left(\frac{8}{9}\arcsin\left(\hat{n}\cdot\hat{y}\right)\right) \end{cases} \quad (7.4)$$

7.2. A Note on Integrability

Because of noise and errors in the estimated surface normals, a property that is true for continuous surfaces is no longer valid. This property will in this document, somewhat sloppily, be referred to as the *integrability* of the surface. When integrating along a line integral on the surface, with the same point used as starting point and end point, the value at the starting point should be the same as that at the end point. Consider the simple case displayed in Figure 7.4, it is the smallest possible path that can be integrated along on the gradient image, and is referred to as an *elementary*

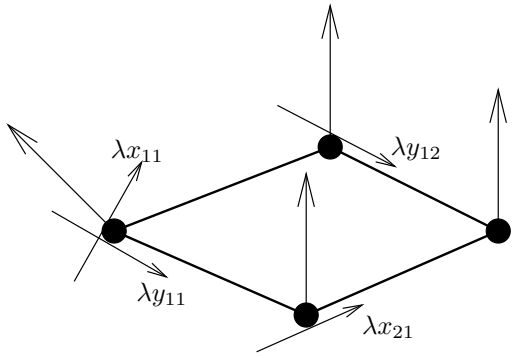

Figure 7.4. *Example of a non-integrable elementary loop. If integrating along a line-integral following the square, the end value will not be the same as the starting value. A violation of a property referred to as integrability in this document.*

loop. It is trivial to realize that, with the estimated surface normals, the property of integrability will not always be valid. A clockwise walk around the loop yields

$$\lambda x_{11} + \lambda y_{12} - \lambda x_{21} - \lambda y_{11} = 0.5 + 0 - 0 - 0 \neq 0 \tag{7.5}$$

The result is that the calculated height of any given point depends on the route that was taken to reach that point. Any noise and error in the surface normals passed along the way, will be accumulated, as can be seen in Figure 7.5.

7.3. Enforcing Integrability

A proposed method for enforcing integrability of the surface is described in [12]. The method described in that document is rather intuitive, it makes use of the sum product algorithm [10], to pass information from local gradient estimates to its neighbors. Unfortunately, it is a poorly written document with references to non-existing figures and at least one obvious mathematical error. When the algorithm was implemented as described, it did not converge into an integrable solution. Although it does not converge, the algorithm is useful to reduce the errors caused by violations of the integrability constraint.

Each pixel on the gradient image is translated into one function node and two gradient nodes with connected observable nodes. They are connected

Figure 7.5. *A model built without enforcing the integrability of the estimated surface normals. Errors have accumulated, causing the left eye to bulge out, and the sides of the neck to point in the wrong direction.*

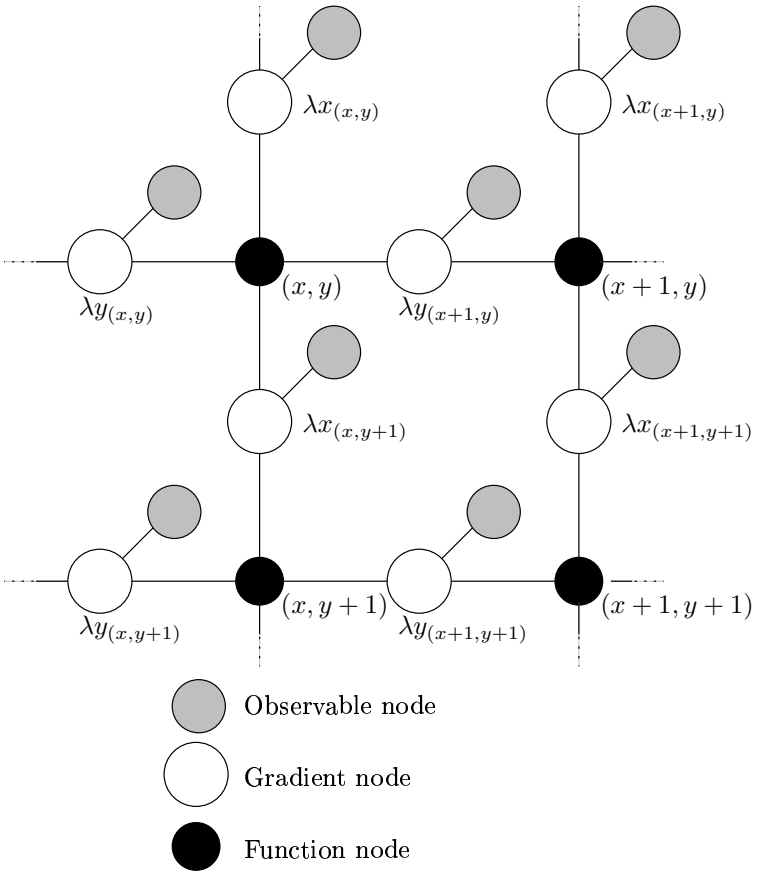

Figure 7.6. *Structure of the graph, on which the algorithm for enforcement of the integrability constraint operates.*

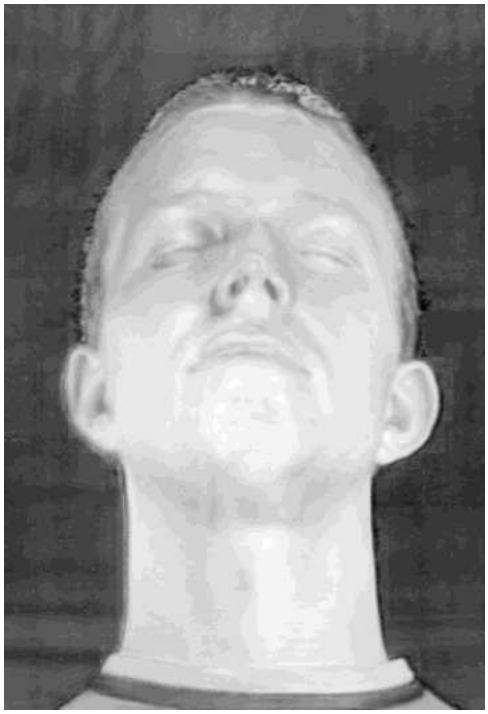

Figure 7.7. *Image showing which parts of a subjects face that best fits the phong model. Brighter regions indicate a better fit.*

according to Figure 7.6. The idea is to try to enforce the integrability of each elementary loop, by passing messages between the nodes. Messages v, consist of conditional probability density functions, in this case the normal distribution function, $v = \mathcal{N}(\lambda, \sigma^2)$. The integrability constraint of an elementary loop is fulfilled, if the property described in Equation 7.5 is true.

Observable nodes never receive any messages. They are containers for the observed values of the derivatives, λx and λy, and the associated variances. The variances are calculated as the relative difference between the estimated phong model and the actual values in the corresponding reflectance map. This means that for a particular model, those derivatives that are estimated from reflectance maps that fit the phong model well, will have small variances. Figure 7.7 shows what parts of the face that best fits the phong model for one of the subjects.

Gradient nodes receive messages from their observable nodes and from the function nodes to which they are connected. Messages from one function

node is multiplied with the local observation, and sent to the other function node. Multiplication is defined as the weighted summation of two normal distribution functions:

$$\begin{array}{rcl} v & = & v_1 \cdot v_2 \\ \mathcal{N}(\lambda, \sigma^2) & = & \mathcal{N}(\lambda_1, \sigma_1^2) \cdot \mathcal{N}(\lambda_2, \sigma_2^2) \\ \lambda & = & \left(\frac{\lambda_1}{\sigma_1^2} + \frac{\lambda_2}{\sigma_2^2}\right) \cdot \frac{1}{\sigma_1^{-2}+\sigma_2^{-2}} \\ \sigma^2 & = & \frac{1}{\sigma_1^{-2}+\sigma_2^{-2}} \end{array} \tag{7.6}$$

The greater the variance of a message is, the smaller the effect of that message will be.

Function nodes receive messages from the corresponding gradient nodes. New messages are then calculated and sent to each gradient node. The messages are calculated using Equation 7.5. Given the messages from three of the nodes, for example, $vx_{(x,y)}$, $vx_{(x,y+1)}$ and $vy_{(x,y)}$, the message to the fourth node $vy_{(x+1,y)}$ is calculated in the following way

$$\begin{array}{rcl} vy_{(x+1,y)} & = & -vx_{(x,y)} + vy_{(x,y)} + vx_{(x,y+1)} \\ \mathcal{N}\left(\lambda y_{(x+1,y)}, (\sigma y)^2_{(x+1,y)}\right) & = & -\mathcal{N}\left(\lambda x_{(x,y)}, (\sigma x)^2_{(x,y)}\right) + \\ & & +\mathcal{N}\left(\lambda y_{(x,y)}, (\sigma y)^2_{(x,y)}\right) + \\ & & +\mathcal{N}\left(\lambda x_{(x,y+1)}, (\sigma x)^2_{(x,y+1)}\right) \\ \lambda y_{(x+1,y)} & = & -\lambda x_{(x,y)} + \lambda y_{(x,y)} + \lambda x_{(x,y+1)} \\ (\sigma y)^2_{(x+1,y)} & = & (\sigma x)^2_{(x,y)} + (\sigma y)^2_{(x,y)} + (\sigma x)^2_{(x,y+1)} \end{array} \tag{7.7}$$

That is, the message sent to the gradient node, is the gradient estimate which would fulfill the integrability constraint, for the considered elementary loop.

The algorithm has five phases:

1. All gradient nodes receive a single message from their observable node.
2. The gradient nodes send their messages to all the function nodes they are connected to, Figure 7.8a.
3. The function nodes compute and send a message to each gradient node they are connected to, Figure 7.8b.
4. The gradient nodes multiply the message from one function node with the local observation and send the message to the other function node, Figure 7.8c.
5. Each gradient node multiplies the final messages with the local observation to obtain the new estimate, Figure 7.8d.

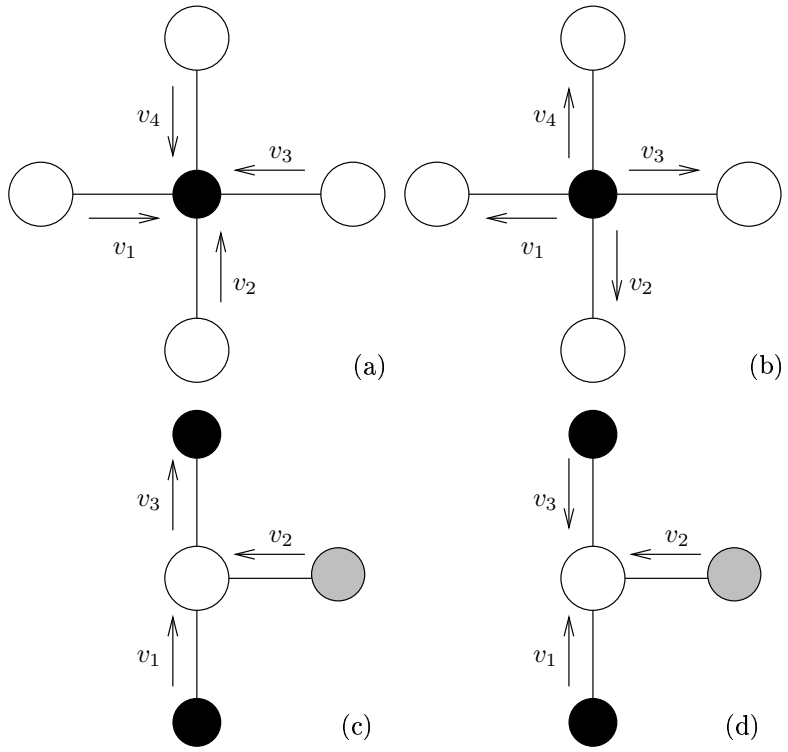

Figure 7.8. *Different phases of the integrability enforcement algorithm. (a) The gradient nodes send their messages to all the function nodes they are connected to. (b) The function nodes compute and send a message to each gradient node they are connected to. (c) The gradient nodes multiply the message from one function node with the local observation and send the resulting message to the other function node. (d) Each gradient node multiplies the final messages with the local observation to obtain the new estimate.*

A measure of how well the algorithm is doing, can be calculated by summing the result of Equation 7.5, for all the elementary loops in the graph. Phase three and four are iterated until no further improvements of the integrability constraint is achieved.

Nodes along the edges of the surface, where the enforcement of integrability is not possible, simply return any messages sent to them, after multiplication with the local observation.

7.4. Integrating the Surface

If the integrability constraint of the gradient map $g(x, y)$ was satisfied, it would be a simple matter to compute the relative heights $h(x, y)$ on the surface, by integrating from an arbitrary starting point (x_0, y_0).

$$h(x, y) = \int_{x_0}^{x} \int_{y_0}^{y} g(u, v)\, \delta u \delta v \tag{7.8}$$

Since the integrability constraint is not guaranteed to be satisfied, the starting point and integration path should be selected in a way that will minimize the effect of the accumulation of errors. The starting point is selected as the center of the segment that describes the skin surface. The height of each point on the surface is then calculated relative to the first point, according to a breadth first algorithm. An example of a skin surface, reconstructed in this way, can be seen in Figure 7.9. The profile of the face is quite nicely reconstructed, as are the mouth, forehead, cheeks and chin.

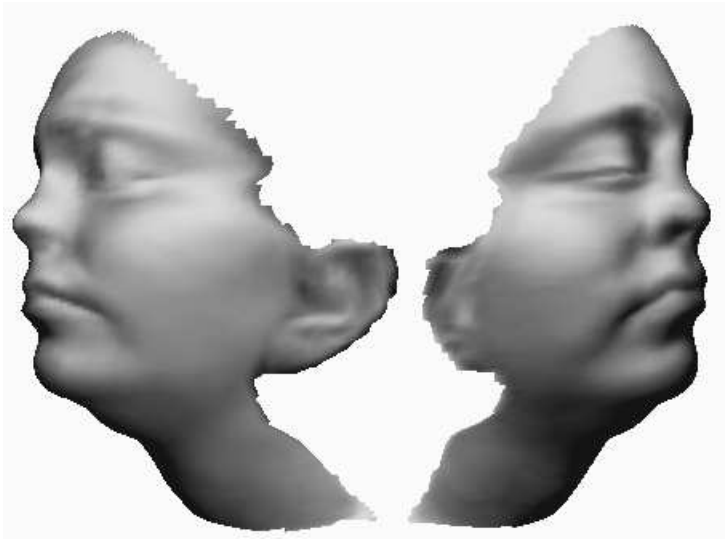

Figure 7.9. *Model rendered without texture shown from two different angles. The left half of the face (in the right part of the picture) is reconstructed somewhat better than the other half. This may be because the face was turned a little bit to the right during the measurement, so that the camera had a better view of the left half. This hints at the quality improvements that could be achieved using multiple cameras.*

8. Discussion and Future Work

As stated in Section 1.2, the objective of this thesis was to construct a three dimensional model of the surface of a human face, given a measurement of its reflectance field. Looking at the models presented in the following section, it feels safe to conclude that this objective has been fulfilled.

8.1. Results

This section consists mainly of a number of figures that show the results. First up, we take a look at a few different models. In Figure 8.1, the profile of a model is compared with the profile of the subject. It is evident that the model building process has done a good job with the chin and mouth. The nose, however, is a bit less pointy on the model than on the subject. A model of another subject is shown in Figure 8.2. It is built from the same measurement that was used to illustrate the face finding algorithm, described in Chapter 5. Despite the rather poor quality of the measurement, due to the large amount of hair shading the face, it was possible to reconstruct a rather realistic model. Finally, Figure 8.3 shows a limitation of the method. The method used has problems with discontinuities in the surface. Hence, the front of the neck has been reconstructed in the wrong place. This, as well as the apparent flattening of the face, can be helped somewhat by changing Equation 7.4 to allow higher gradient values. Unfortunately, the possible improvements comes at a price, see Figure 7.3.

The most powerful feature of the collected reflectance fields is the possibility to, with a high degree of realism, render the subjects under novel lighting conditions. Since we also have made models of the subject, this can be done from novel viewpoints. The relationship between the light map in use, and the resulting look of the subject, is illustrated by Figure 8.4 and 8.5. Figure 8.6, 8.7 and 8.8 show the effect of different illumination environments, captured at the locations used as background for each image. The same model is used in each image to help show the effect of the illumination, rather than differences between models.

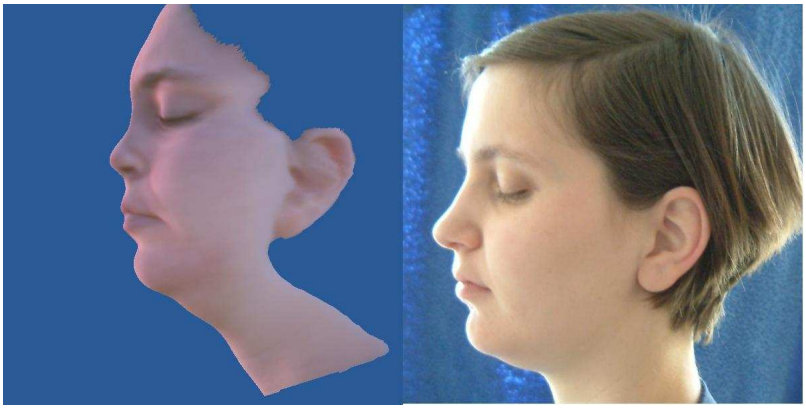

Figure 8.1. *Comparison of profiles between the model and the subject. The model is not lit by the same illumination environment as the subject.*

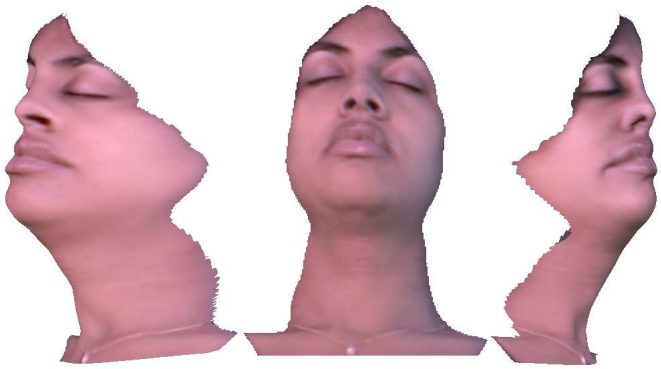

Figure 8.2. *Model of the female subject that was used to illustrate the face finding algorithm in Chapter 5. The eyebrows that were removed by that algorithm were restored manually. Despite large areas of the face being in the shadow of the hair during measurement, a reasonably good model of the face could be reconstructed.*

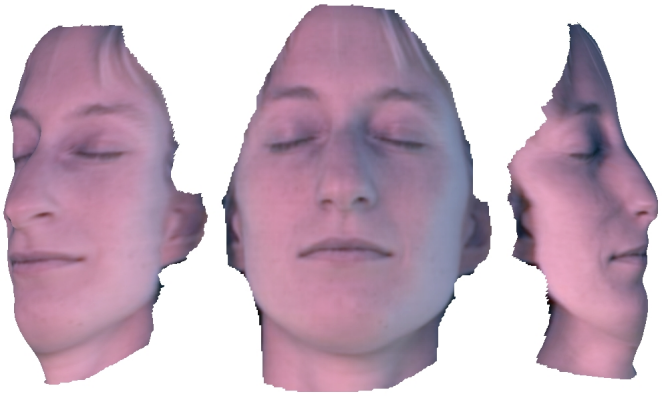

Figure 8.3. *Model built from a measurement containing a discontinuity. The skin below the chin has not been visible to the camera, hence the chin and the front of the neck has been reconstructed incorrectly. The model is also too "flat", to some extent due to errors in the data collected by the angular probe.*

Figure 8.4. *Anna lit with an artificial illumination environment, the light map used can bee seen in Figure 8.5.*

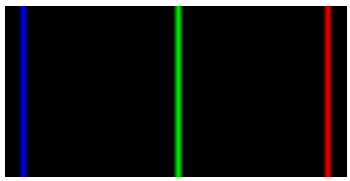

Figure 8.5. *The artificial light map used to render the image of Anna in Figure 8.4. The light map can be interpreted as a green light source in front of the subject, and red and blue light sources from either side behind the subject.*

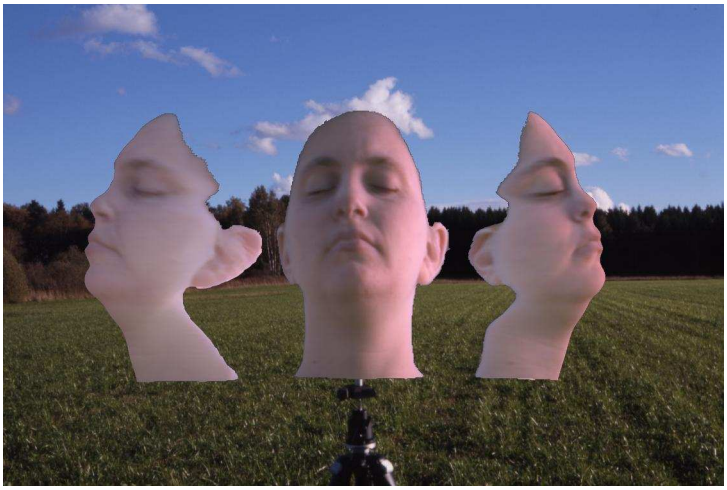

Figure 8.6. *Model of Anna lit by an illumination environment with direct sunlight, captured on an open field.*

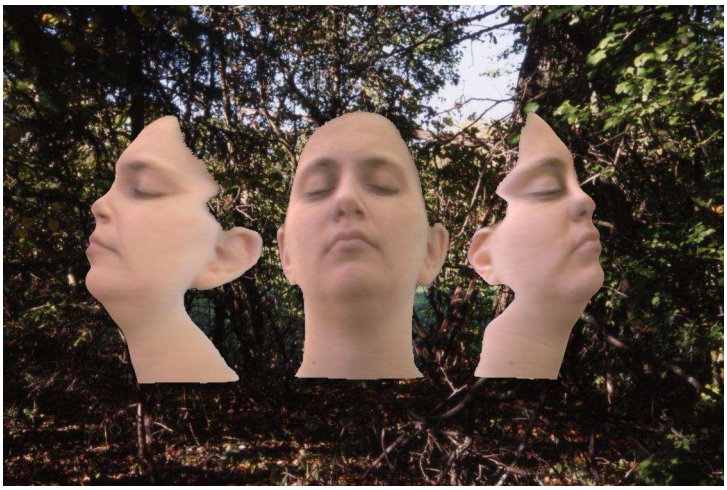

Figure 8.7. *Model of Anna lit by an illumination environment captured in a forest. The sunlight is filtered through green leaves, which causes the slightly green taint on the model.*

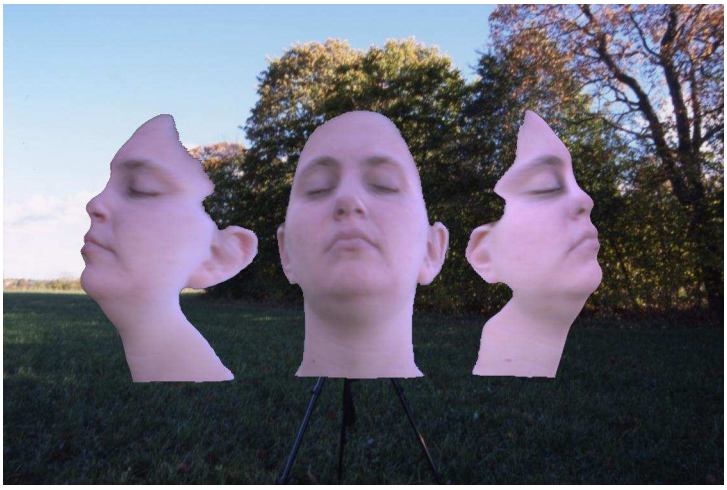

Figure 8.8. *Model of Anna lit by an illumination environment with indirect sky light. The direct sunlight is blocked, which causes the model to be lit by the clear blue sky.*

8.2. Possible Improvements

- Color correction could be employed when making the reflectance fields. This is needed to make truly photo realistic renderings. The steps needed that were not done are; using color corrected cameras, calibrating the color of the light source, calibrating the intensity of the light source to achieve even illumination over the entire subject.
- A model better suited to describe the skin surface, for example the model employed in [5], could be used. Consequently, the reflectance maps could be compensated for novel viewpoints in a more realistic way.
- More cameras can be used when performing measurements with the light stage, the extra data would help improve both the model building process, as well as the rendering of the results.

8.3. Future Work

The model could be animated, for example, using [9] or [1]. Using some of the improvements mentioned above and techniques to make the model follow a video sequence, it would even be possible to examine whether the information collected in a measurement can be used to relight a face in a video sequence. It would make for an interesting comparison with the method described in [6].

Video sequences that could be used to animate the model were shot of some of the subjects. These recordings were made under known lighting, that is, the light environment was measured with the light probe prior to filming. The resulting sequences were not used in this project but are, however, available for later use.

Bibliography

[1] Jörgen Ahlberg. *Model-based Coding.* PhD dissertation, Department of Electrical Engineering, Linköping University, 2002.

[2] Edward Angel. *Interactive computer graphics.* Addison-Wesley, 2003.

[3] George Borshukov and J. P. Lewis. Realistic human face rendering for "The Matrix Reloaded". In *Proceedings of the SIGGRAPH 2003 conference on Sketches & applications.* ACM Press, 2003.

[4] I. W. Busbridge. *The Mathematics of Radiative Transfer.* Cambridge University Press, Bristol, UK, 1960.

[5] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 145–156. ACM Press/Addison-Wesley Publishing Co., 2000.

[6] Paul Debevec, Andreas Wenger, Chris Tchou, Andrew Gardner, Jamie Waese, and Tim Hawkins. A lighting reproduction approach to live-action compositing. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 547–556. ACM Press, 2002.

[7] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 369–378. ACM Press/Addison-Wesley Publishing Co., 1997.

[8] E.Catmull and R.Rom. A Class of Local Interpolating Splines. In *Computer Aided Geometric Design.* Academic Press, 1974.

[9] Marco Fratarcangeli. MPEG-4 Compliant Anatomy-Based Modeling for Fast Facial Expression Synthesis. Master's thesis, Department of Electrical Engineering, Linköping University, 2004.

[10] F. R. Kschischang, B. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. Inform. Theory*, 47(2):498–519, 2001.

[11] Frederick I. Parke. Computer generated animation of faces. In *Proceedings of the ACM annual conference*, pages 451–457. ACM Press, 1972.

[12] Nemanja Petrovic, Ira Cohen, Brendan J. Frey, Ralf Koetter, and Thomas S. Huang. Enforcing Integrability for Surface Reconstruction Algorithms Using Belief Propagation in Graphical Models. Published on website `citeseer.nj.nec.com/476395.html`.

[13] A. Robles-Kelly and E.R. Hancock. A graph-spectral method for surface height recovery from needle-maps. *Proceedings of the 2001 IEEE Computer Society Conference*, 1:141–148, 2001.

[14] Hideo Saito, Y. Somiya, and S. Ozawa. Shape Reconstruction of Skin Surface from Shading Images Using Simulated Annealing. In *ACCV95*, volume III, pages 348 – 352, December 1995.

[15] Essenpreis M Cope M Simpson CR, Kohl M. Near infrared optical properties of ex-vivo human skin and subcutaneous tissues measured using the Monte Carlo inversion technique. *Phys Med Biol*, 43:2465–2478, 1998.

[16] Chris Tchou and Paul Debevec. HDR shop. Available at `www.debevec.org/HDRShop/`, 2001.

[17] T.Yamada, Hideo Saito, and S.Ozawa. 3D reconstruction of Skin Surface from Image Sequence. In *IAPR Workshop on Machine Vision Applications (MVA98)*, November 1998.

[18] Ruo Zhang, Ping-Sing Tsai, James Edwin Cryer, and Mubarak Shah. Shape from Shading: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(8):690–706, 1999.

A. Description of the Serial Interface to the Light Stage

The angular probe on the light stage consists of a single PIC-processor and some supporting components, such as a MAX232 serial driver and two photo transistors. The PIC has built in A/D converters to which the photo transistors are connected. The PIC sends data on the serial line with a speed of 460 bytes/second, each byte is divided into two components, Figure A.1. The first bit is called the "source bit" and tells which of the sensors that has measured the value in the seven "level bits". The value contained in the level bits is the voltage measured from the corresponding sensor's photo transistor. The sensors are polled alternatingly, so every second transmitted byte has the source bit set.

The sensors are interchangeable, so it may not make sense to specify which sensor has the source bit set and which has not, but with the setup used for this work, the sensor keeping track of the θ-axis had the source bit set. Worth noting is also the fact that the ϕ-angle cannot be linearly interpolated from the number of times the reflectors pass in front of the sensor, see Figure A.2. For each full turn of the wheel, 14 centimeters of the ϕ-cord passes through. Knowing the radius of the sphere that the light source travels along r, the ϕ-angle can be calculated as:

$$\phi = 2 \arcsin\left(\frac{l}{2r}\right) \tag{A.1}$$

where l is the length of the ϕ-cord.

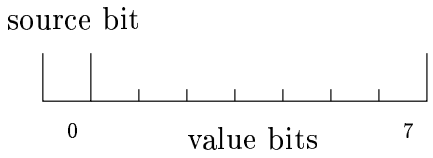

Figure A.1. *Description of the byte syntax used by the angular probe on the light stage.*

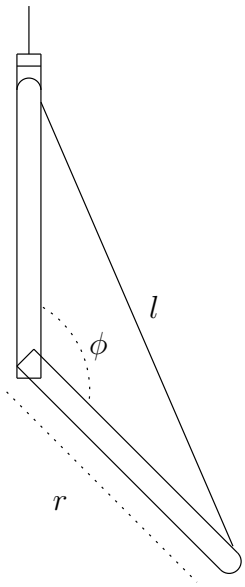

Figure A.2. *Description of the entities involved in the calculation of the ϕ-angle.*

B. Gamma Correction

This appendix on gamma correction is included because the distinction between RGB colorspace and gamma corrected RGB, referred to as R'G'B', is frequently overlooked. Nowadays, most images are stored in R'G'B' space, and treating them as RGB leads to incorrect results.

What is Gamma Correction?

Gamma correction is needed because of the nature of cathode ray tubes (CRT) monitors. If you have some sort of real live scene and turn it into a computer image by measuring the amount of light coming from each point of the scene, then you have created a linear image. This is a good thing because you can manipulate the image as if the values in the image file were light, that is, adding and multiplying will work just like real light. Now if you take the image file and turn each pixel value into a voltage and feed it into a CRT, you find that the CRT does not give you an amount of light proportional to the voltage. The amount of light coming from the phosphor in the screen depends on the the voltage something like this:

$$Transmitted\,light \quad = \quad Voltage^{crt\,gamma} \tag{B.1}$$

So if you just dump a nice linear image out to a CRT, the image will look much too dark. To fix this you have to gamma correct the image.

How is it Calculated?

To display an image correctly, you first need to do the opposite of what the CRT will do to the image, so that things cancel out.

$$\begin{aligned} R' &= R_{max}\left(\frac{R}{R_{max}}\right)^{\frac{1}{\gamma_r}} \\ G' &= G_{max}\left(\frac{G}{G_{max}}\right)^{\frac{1}{\gamma_g}} \\ B' &= B_{max}\left(\frac{B}{B_{max}}\right)^{\frac{1}{\gamma_b}} \end{aligned} \tag{B.2}$$

Typically, $R_{max} = G_{max} = B_{max} = 255$.