Measuring and simulating haemodynamics due to geometric changes in facial expression

Timothy Scully

A dissertation submitted in partial fulfillment of the requirements for the degree of

Doctor of Engineering

of

University College London.

Department of Computer Science

University College London

March 11, 2020
I, Timothy Scully, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.
Abstract

The human brain has evolved to be very adept at recognising imperfections in human skin. In particular, observing someone’s facial skin appearance is important in recognising when someone is ill, or when finding a suitable mate. It is therefore a key goal of computer graphics research to produce highly realistic renderings of skin. However, the optical processes that give rise to skin appearance are complex and subtle. To address this, computer graphics research has incorporated more and more sophisticated models of skin reflectance. These models are generally based on static concentrations of skin chromophores; melanin and haemoglobin. However, haemoglobin concentrations are far from static, as blood flow is directly caused by both changes in facial expression and emotional state. In this thesis, we explore how blood flow changes as a consequence of changing facial expression with the aim of producing more accurate models of skin appearance.

To build an accurate model of blood flow, we base it on real-world measurements of blood concentrations over time. We describe, in detail, the steps required to obtain blood concentrations from photographs of a subject. These steps are then used to measure blood concentration maps for a series of expressions that define a wide gamut of human expression. From this, we define a blending algorithm that allows us to interpolate these maps to generate concentrations for other expressions. This technique, however, requires specialist equipment to capture the maps in the first place. We try to rectify this problem by investigating a direct link between changes in facial geometry and haemoglobin concentrations. This requires building a unique capture device that captures both simultaneously. Our analysis hints a direct linear connection between the two, paving the way for further investigation.
Abstract
Impact Statement

In Chapter 3, this thesis presents a comprehensive guide to extracting melanin and haemoglobin concentrations from skin images. It builds upon previous techniques, but provides experimental evaluation for camera selection, and for choice of inversion method in extracting the values. In addition, it demonstrates the applicability of skin scattering models that have been previously unused in the area of chromophore concentration extraction. We show their applicability by demonstrating a statistical correlation to previous skin models that have been shown experimentally to produce accurate results. These evaluations and guides would accelerate any further experiments that would be needed to be performed in academia related to blood concentration extraction.

In Chapters 4 and 5, we present a novel dynamic haemoglobin model that can be coupled to an animation rig. This has uses outside of academia, and can allow animators to produce more realistic skin rendering in both the film and the computer games industry.

In Chapters 6 and 7, we present details of how to construct a novel combined geometry and chromophore concentration scanner. Again, this has uses in academia to assist in further research, but can also be used in the media industries to capture and drive more realistic animation.

In Chapters 8 and 9, we demonstrate a connection between changes in geometry and haemoglobin concentration values. Although, there is limited evidence for this, it is enough to suggest a connection. This provides a solid ground for further experiments into this area.
Acknowledgements

Firstly, I would like to thank my supervisor Prof. Tim Weyrich for guiding me through the EngD process, offering me advice when I needed it and supporting me in forging my own research direction.

Secondly, I would like to thank my second supervisor, Prof. Simon Arridge, for reading and reviewing my thesis while I was writing it, ready for submission.

Thirdly, I would like to thank Procter & Gamble, and my industrial supervisor Dr. Paul Matts, for providing sponsorship for my EngD, along with the Royal Commission for the Exhibition of 1851 who awarded me a Fellowship that not only provided financial support but also provided encouragement during some difficult times.

Lastly, and not leastly, I would like to thank my family. For my Mum, who provided words of encouragement, but sadly passed away before I completed; for my Dad who provided academic insight and discussion, throughout the process; my partner, Liz, who supported me through the long nights and weekends, and when everything seemed tough; and my son, Jacob, for making me smile. I would like to dedicate this thesis to all of them.
Acknowledgements
# Contents

1 Introduction 27
   1.1 Contributions and organisation 29

2 Fundamentals & Related Work 33
   2.1 Facial expressions 33
   2.2 Blood Flow Modelling 34
      2.2.1 Mathematical Model 34
      2.2.2 Data-driven modelling 35
      2.2.3 Our Approach 35
   2.3 Light scattering in skin 36
      2.3.1 Radiometry 36
      2.3.2 Radiance 37
      2.3.3 Surface Reflectance 38
      2.3.4 Specular Reflection BRDF 39
      2.3.5 Diffuse Reflectance 40
      2.3.6 Modelling Translucent Materials 41
      2.3.7 Fresnel Reflectance 42
      2.3.8 BSSRDF 44
   2.4 Measuring skin chromophore concentrations 44
      2.4.1 Beer-Lambert law 45
      2.4.2 Kubelka-Munk Theory 46
      2.4.3 Diffusion Approximation 47
      2.4.4 The Diffusion Substitution 50
      2.4.5 Dipole Approximation 51
      2.4.6 Multipole Approximation 52
      2.4.7 Multilayered Multipole Approximation 53
      2.4.8 Extended Multipole Approximations 54
   2.5 Skin Structure 55

2.4.1 Beer-Lambert law 45
2.4.2 Kubelka-Munk Theory 46
2.4.3 Diffusion Approximation 47
2.4.4 The Diffusion Substitution 50
2.4.5 Dipole Approximation 51
2.4.6 Multipole Approximation 52
2.4.7 Multilayered Multipole Approximation 53
2.4.8 Extended Multipole Approximations 54
2.5 Skin Structure 55
2.5.1 Skin scatterers and Absorbers ........................................... 56

2.6 Geometry capture ............................................................ 57
   2.6.1 Single-shot facial capture ............................................. 57
   2.6.2 Structured Light ....................................................... 58
   2.6.3 Photometric Stereo .................................................... 59
   2.6.4 Multi-view stereo ...................................................... 60

2.7 Facial Performance Capture ............................................... 61

2.8 Differential Geometry ...................................................... 62
   2.8.1 Divergence ............................................................. 62
   2.8.2 Curvature .............................................................. 63
   2.8.3 Surface tangent field ................................................ 63

2.9 Discussion ........................................................................ 64

3 Haemoglobin Analysis & Initial Findings ................................. 65
   3.1 Physical Construction ..................................................... 66
   3.2 Software Pipeline ......................................................... 67
      3.2.1 Skin Lookup Table ................................................... 68
      3.2.2 Spectral camera transform ....................................... 69
      3.2.3 Spectral camera calibration ..................................... 71
      3.2.4 Implementation ....................................................... 73
      3.2.5 Uneven Illumination ................................................ 74
   3.3 Evaluation ..................................................................... 75
      3.3.1 Bit depth reduction .................................................. 76
      3.3.2 Compression .......................................................... 76
      3.3.3 Gamma curve correction ......................................... 78
      3.3.4 Image Noise ............................................................ 79
   3.4 Haemoglobin Analysis ..................................................... 80
      3.4.1 Inversion Method ...................................................... 80
      3.4.2 Evaluation ............................................................. 81
   3.5 Camera evaluation .......................................................... 83
      3.5.1 Bit depth reduction .................................................. 84
      3.5.2 Compression .......................................................... 84
      3.5.3 Gamma Curve ........................................................ 84
      3.5.4 Image Noise ............................................................ 86
   3.6 Accuracy ...................................................................... 86
   3.7 Initial Findings ............................................................... 88
   3.8 Summary ...................................................................... 91
# 8 Metric Computation

## 8.1 Observations

- 8.1.1 Curvature
- 8.1.2 Divergence

## 8.2 Image preparation

- 8.2.1 UV Mapping
- 8.2.2 View-dependent combination
- 8.2.3 Tracking and warping

## 8.3 Metric Computation

- 8.3.1 Scale Space Decomposition
- 8.3.2 Image Space Derivatives
- 8.3.3 Curvature computation
- 8.3.4 Divergence Computation
- 8.3.5 Divergence Image Space Computation

## 8.4 Summary

# 9 Analysis

## 9.1 Measurements

## 9.2 Observations

## 9.3 Model

## 9.4 Results and Discussion

# 10 Conclusions

# A Polarization

# B Camera Calibration Parameters

## B.1 Linear Intrinsic Parameters

## B.2 Non-linear Intrinsic Parameters

# C Thermal Noise Model

# Bibliography
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Full pipeline overview</td>
<td>29</td>
</tr>
<tr>
<td>2.1</td>
<td>Facial Expression classification</td>
<td>34</td>
</tr>
<tr>
<td>2.2</td>
<td>Radiance</td>
<td>37</td>
</tr>
<tr>
<td>2.3</td>
<td>BRDF vs BSSRDF</td>
<td>38</td>
</tr>
<tr>
<td>2.4</td>
<td>Polar coordinate plots of the Henyey-Greenstein scattering function for different values of the anisotropy factor, $g$</td>
<td>41</td>
</tr>
<tr>
<td>2.5</td>
<td>Fresnel Reflectance</td>
<td>43</td>
</tr>
<tr>
<td>2.6</td>
<td>Overview of the Monte Carlo simulation algorithm</td>
<td>48</td>
</tr>
<tr>
<td>2.7</td>
<td>Dipole and Multipole Approximation</td>
<td>51</td>
</tr>
<tr>
<td>2.8</td>
<td>Simple overview of the structure of skin</td>
<td>55</td>
</tr>
<tr>
<td>2.9</td>
<td>Absorption and Scattering Characteristics of Skin</td>
<td>56</td>
</tr>
<tr>
<td>2.10</td>
<td>Difference between a source (left) and a sink (right) in a vector field</td>
<td>62</td>
</tr>
<tr>
<td>3.1</td>
<td>Photograph of physical set-up and diagram showing the plane of polarisation</td>
<td>66</td>
</tr>
<tr>
<td>3.2</td>
<td>Haemoglobin Analysis - Overview</td>
<td>68</td>
</tr>
<tr>
<td>3.3</td>
<td>Bayer Matrix</td>
<td>70</td>
</tr>
<tr>
<td>3.4</td>
<td>RAW Image Processing pipeline</td>
<td>70</td>
</tr>
<tr>
<td>3.5</td>
<td>Common ColorCheckers</td>
<td>73</td>
</tr>
<tr>
<td>3.6</td>
<td>ColorChecker DC Illumination correction</td>
<td>74</td>
</tr>
<tr>
<td>3.7</td>
<td>Calibration data used to perform evaluation</td>
<td>75</td>
</tr>
<tr>
<td>3.8</td>
<td>Fit using 8-bit image of calibration target</td>
<td>76</td>
</tr>
<tr>
<td>3.9</td>
<td>Comparison of different image compression qualities</td>
<td>77</td>
</tr>
<tr>
<td>3.10</td>
<td>Comparison of different gamma curve reconstruction errors</td>
<td>77</td>
</tr>
<tr>
<td>3.11</td>
<td>Comparison of calibration fit under noise</td>
<td>79</td>
</tr>
<tr>
<td>3.12</td>
<td>Relative error of haemoglobin reconstruction using different interpolation and normalisation methods. Here, nearest/natural/bilinear represent the interpolation method, and spherical/cotton are the normalisation method.</td>
<td>82</td>
</tr>
</tbody>
</table>
3.13 Skin manifold in different spaces demonstrating metamerism. On the right we have highlighted the place where the metamerism occurs. ........................................ 82

3.14 Comparison of relative error of reconstructed haemoglobin between 8-bit quantisation (left) and 16-bit quantisation (right). .................................................. 84

3.15 Comparison of relative haemoglobin reconstruction error under different compression levels ................................................................. 85

3.16 Comparison of relative error of reconstructed haemoglobin due to error in the calibration of the gamma curve exponent ...................................................... 85

3.17 Comparison of relative error of reconstructed haemoglobin due to noise levels ................................................................. 86

3.18 Plot showing haemoglobin values obtained from Monte Carlo analysis of Kubelka-Munk theory lookup table ...................................................... 87

3.19 Facial haemoglobin distribution of a 33-year-old Caucasian female and of a 26-year-old Caucasian male. Dark pixels denote high concentration. On the right, we see the highly vascular nature of the blood perfusion ........................................ 88

3.20 Changes in haemoglobin concentration for subject when he changes his pose ................................................................. 89

3.21 Comparison of haemoglobin changes when a subject has a happy expression ................................................................. 89

3.22 Time multiplexed signals ................................................................. 90

3.23 Our time multiplexing scheme. Top: We sample the expressions at varying time offsets from the expression transition. Bottom: The images were taken in column-major order (as numbered), with a short phase shift between acquisition and expression changes. This gives a consistent 5-image temporal sequence of four expressions (neutral/smile/neutral/smile) when arranged in row-major order ................................................................. 91

4.1 Histogram Mapping between two CDFs .................................................. 94

4.2 A sample of different distributions in a radius around each point on the left, with fitted Gaussian curve. The color coded points, left, correspond to the distribution diagrams, right; namely forehead, nose, upper cheek, jaw and chin. (The histogram on the tip of the nose is not shown) .................................................. 95

4.3 Different maps used by our algorithm. The haemoglobin scale and bias maps shown here belong to the smile expression (Haemoglobin Scale shown at 90x scale for visualisation purposes) .................................................. 97

5.1 Rendered emotional states. The top row uses our blend shape method. The bottom row simply has the neutral haemoglobin map for comparison .................................................. 103

5.2 Example of blending poses: from left to right, the character transitions from the neutral pose to full anger (third image), to a combination of full anger and full surprise (fifth image). The last image adds the changes after physical exercise .................................................. 103
5.3 The neutral geometric pose of a face rendered with a neutral haemoglobin map (left), and predicted haemoglobin perfusion after exercise (middle) and after alcohol consumption (right). ......................................................... 103

5.4 Close-up of the neutral and the disgust poses. The changes in hemoglobin distribution are generated by our model. ................................................................. 104

6.1 Diagram showing final capture set-up ................................................................. 105

6.2 Comparison of reconstruction results for the Manta G-201C, Manta 504C, and Pike 505C. (From top to bottom) Original, Haemoglobin, Melanin, Diffuse Albedo Reconstruction, Shading (Irradiance), Reconstruction ......................................................... 107

6.3 Construction of a circular polariser as a composite of a linear polariser angled at 45°, and a quarter-wave retarder ................................................................. 109

6.4 Overall and spectral efficiency of the polariser as a function of rotation .................. 110

6.5 Spectral power distributions of a) halogen light source, b) white LED light source. ..... 111

6.6 Stability of the halogen spectrum over time ......................................................... 112

6.7 Noise levels over time as the temperature on the image sensor increases. We capture a series of frames of increasing temperature (left). However, we can see that the standard deviation stays constant over time (right). ......................................................... 113

7.1 Geometry Processing - Overview ........................................................................ 117

7.2 A sample image split into the three colour channels. The images show that the blue channel has a disproportionately high level of noise in comparison to the other channels as a result of the light source used. nb. Images are scaled such that the maximum value is 1 ....................................................................................... 118

7.3 Dot pattern provided by George Vogiatzis used for intrinsic calibration of images. . 119

7.4 Target used for extrinsic calibration, a) shows the original calibration target, b-e) shows the calibration target imaged simultaneously from all cameras. The images demonstrate the very low depth of field available for the aperture setting. ......................................................... 119

7.5 Mesh reconstruction process .............................................................................. 121

7.6 Low exposure areas (right) lead to unresolvable noise in the geometry (left) ........... 124

7.7 Mis-calibration of the extrinsic parameters can lead to ridges being formed along the boundary between two camera views. The ridge can clearly be seen on the left, and is highlighted in red (along with the triangulation) on the right. ................................. 124

8.1 Metric Computation - Overview ........................................................................ 127

8.2 Subject showing wrinkles changing curvature on facial expression. Color images (left), Haemoglobin Concentration (right) ....................................................................... 129
8.3 Subject showing curvature changes in cheek on facial expression. Color images (left), Haemoglobin Concentration (right) .......................................................... 130
8.4 Shrinking in skin shows reduction in blood concentration. Color images (left), Haemoglobin Concentration (right) .......................................................... 132
8.5 UV Mapping ......................................................................................... 133
8.7 The four images on the left represent our four images from different viewpoints of our subject. They first transformed into the same UV coordinate system. The final image shows the four images combined together. ........................................ 134
8.6 Manual alignment of the neutral expression such that the subject is facing forward .. 134
8.8 Tracking results over a capture sequence. The density of the tracked points is reduced for visualisation purposes ................................................................. 135
8.9 The results of the tracking are used to warped the images to align them with the initial frame. Original images (Top row), Warped images (Bottom row) . . . . . . . 136
8.10 Successive images showing the blurring of the geometry as a result of the frequency decomposition process. For ease of visualisation, the images show the normal map of the geometry, d scaled so that it ranges from 0 to 1. ........................................ 137
8.11 Rotation of the tangential basis around the normal can cause the apparent displacement of a fixed point on the surface .................................................. 140
8.12 Change in haemoglobin concentration (left) against change in the computed metrics: Divergence, Mean Curvature, Gaussian Curvature ((left to right)) with respect to the initial frame ................................................................. 142
9.1 Analysis - Overview ............................................................................. 145
9.2 We pick a set of points across the face (left, plotted a sample blood concentration map), showing the corresponding change in the divergence metric, blood concentration levels and resultant scatter plot and fit (right) ........................................ 146
9.3 Pearson correlation coefficient between the change in divergence per pixel and the changed in haemoglobin value ......................................................... 147
9.4 Pearson correlation coefficient between the change in Gaussian curvature per pixel and the changed in haemoglobin value ......................................................... 147
9.5 Absolute value PCC between divergence (left) and haemoglobin change (right). On the top a subject has demonstrated the disgust expression. On the bottom another subject is demonstrating the surprise expression. ................................. 148
9.6 Regressed linear relationship coefficients between surface divergence and haemoglobin changes. On the top a subject has demonstrated the disgust expression. On the bottom another subject is demonstrating the surprise expression. ......................................................... 149
9.7 The actual blood concentration maps vs the concentration maps generated by the model. On the right, we show the percentage error between the model and the measured blood concentration maps. ........................................ 151

9.8 The actual blood concentration maps vs the concentration maps generated by the model. On the right, we show the percentage error between the model and the measured blood concentration maps. ........................................ 152

9.9 Comparison of our blend model, and the geometric model. ........................................ 153
Glossary

3DMM  Three-dimensional morphable model.

BCSL  Boundary Coded Structured Light.

BRDF  Bidirectional Reflectance Distribution Function.

BSSRDF  Bidirectional Surface Scattering Reflectance Distribution Function.

CDF  Cumulative Distribution Function.

FACS  Facial Action Coding System.

ICIA  Inverse Compositional Image Alignment.

LLE  Locally Linear Embedding.

MVS  Multiview Stereo.

NCC  Normalized cross correlation.

NIR  Neutral Interface Reflector.

PCC  Pearson product-moment correlation coefficient.

PMVS  Patch-based multi-view stereo.

RTE  Radiative Transport Equation.

SVM  Support Vector Machine.

ULR  Unstructured Lumigraph Rendering.
Glossary
## Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_n$</td>
<td>2D Fourier transform of reflectance profile of layer $n$.</td>
</tr>
<tr>
<td>$R_n$</td>
<td>2D Fourier transform of transmission profile of layer $n$.</td>
</tr>
<tr>
<td>$R_{o}$</td>
<td>2D reflectance profile of layer $n$.</td>
</tr>
<tr>
<td>$T_{o}$</td>
<td>2D transmission profile of layer $n$.</td>
</tr>
<tr>
<td>$d_{r}$</td>
<td>Absolute distance from the negative dipole source.</td>
</tr>
<tr>
<td>$d_{e}$</td>
<td>Absolute distance from the positive dipole source.</td>
</tr>
<tr>
<td>$\mu_a$</td>
<td>Absorption coefficient, mm$^{-1}$.</td>
</tr>
<tr>
<td>$n_o$</td>
<td>Ambient refractive index.</td>
</tr>
<tr>
<td>$A$</td>
<td>Area.</td>
</tr>
<tr>
<td>$\mu_{em}$</td>
<td>Baseline absorption coefficient.</td>
</tr>
<tr>
<td>$D$</td>
<td>Beckmann geometric distribution.</td>
</tr>
<tr>
<td>$f$</td>
<td>Bi-directional Reflectance Distribution Function.</td>
</tr>
<tr>
<td>$\mu_{a}^{\text{deoxy}}$</td>
<td>Deoxy-haemoglobin absorption coefficient.</td>
</tr>
<tr>
<td>$dA'$</td>
<td>Differential detector area.</td>
</tr>
<tr>
<td>$dA$</td>
<td>Differential light area.</td>
</tr>
<tr>
<td>$dV$</td>
<td>Differential volume element.</td>
</tr>
<tr>
<td>$D$</td>
<td>Diffusion coefficient.</td>
</tr>
<tr>
<td>$\mu_{\text{eff}}$</td>
<td>Effective attenuation coefficient.</td>
</tr>
<tr>
<td>$\mu_{\text{tr}}$</td>
<td>Effective transport coefficient.</td>
</tr>
<tr>
<td>$\bar{J}$</td>
<td>Energy density.</td>
</tr>
<tr>
<td>$\mu_{em}^{\text{eum}}$</td>
<td>Eumelanin absorption coefficient.</td>
</tr>
<tr>
<td>$z_o$</td>
<td>Extrapolation distance for dipole boundary condition.</td>
</tr>
<tr>
<td>$M_n$</td>
<td>Facial mesh for frame $n$.</td>
</tr>
</tbody>
</table>
Symbols

$M_0$  Facial mesh for neutral expression.
$M_1$  First principal curvature.
$\Phi$  Flux.
$\gamma$ Fraction of oxygenated haemoglobin relative to deoxygenated.
$\theta_i$ Fresnel angle of incidence.
$\theta_t$ Fresnel angle of transmittance.
$F_R$ Fresnel reflectance coefficient.

$K$  Gaussian curvature.
$G$  Geometric self-shadowing effect.
$\mathcal{G}$  Geometry image.

$h$  Haemoglobin concentration.
$\mathcal{H}$  Haemoglobin concentration image.
$C_h$  Haemoglobin volume fraction.
$z$  Height above the surface.
$z_v$ Height of the negative dipole source above the surface.
$z_r$ Height of the positive dipole source above the surface.
$\int_{\Omega_i}$ Hemi-spherical integration.

$\Phi_i$ Incoming flux.
$\vec{\omega}_i$ Incoming light vector.
$E$ Irradiance.
$I$ Irradiance image.

$H$  Mean curvature.
$l$  Mean free path length.
$\mu_n$ Mean of dark noise, or shot noise.
$\mu_p$ Mean of photon noise.
$m$  Melanin concentration.
$\mathcal{M}$ Melanin concentration image.
$\mu_{\text{mie}}$ Mie scattering coefficient.

$\Phi_r$ Outgoing flux.
$\vec{\omega}_r$ Outgoing light vector.
$\mu_{\text{oxy}}$ Oxy-haemoglobin absorption coefficient.
Symbols

$\mu_{a}^{em}$  Pheomelanin absorption coefficient.
\( \bar{r} \)  Position.

\( r \)  Radial distance from a point.
\( L \)  Radiance.
\( P \)  Radiant power.

$\mu_{s}^{mie}$  Rayleigh scattering coefficient.
\( n_{d} \)  Refractive index of incidence material.
\( \eta \)  Relative refractive index.

\( \alpha' \)  Scattering albedo.
\( g \)  Scattering anisotropy factor.
\( \mu_{s} \)  Scattering coefficient, mm$^{-1}$.
\( p \)  Scattering phase function.
\( M_{2} \)  Second principal curvature.

\( \hat{\omega} \)  Solid angle.
\( \sigma_{p} \)  Standard deviation of photon noise.
\( \sigma_{t} \)  Standard deviation of shot noise.
\( \bar{r}_{t} \)  Surface position of reflected light due to sub-surface scattering.
\( \rho \)  Surface reflectivity.
\( m \)  Surface roughness parameter.

\( t \)  Time.
\( d \)  Tissue slab thickness.

\( \hat{n} \)  Unit vector surface normal.

\( \bar{\hat{h}} \)  Vector halfway between the incoming direction, \( \hat{\omega}_{i} \) and the outgoing direction, \( \hat{\omega}_{r} \).
\( \hat{\omega}_{s} \)  Vector of ideal specular reflectance.
Chapter 1

Introduction

The human brain has evolved to be very adept at recognising imperfections in human skin. In particular, observing someone’s facial skin appearance is useful for many reasons such as recognising when someone is ill, or making a decision in choosing a suitable mate. In addition, perfusion effects such as blushing can indicate a person’s emotional state. As a consequence of this adeptness, renderings of human skin have to be very realistic. Deviation from realism can quickly lead down the uncanny valley [SN07], and be disturbing for others. Jung and Wagner [JW10] have also provided a statistical analysis on effectively people can recognize emotion when colouration changes are added. They found that recognition rates were high, when dynamic skin changes were added to facial expressions. As a result, a large amount of research over the past two decades has been dedicated to reproducing perfection in skin appearance rendering.

The appearance of skin, and of any object, is the brain’s interpretation of the reflected light from that object. When light interacts with an object it is absorbed and scattered by a number of different optical processes. For some objects, this process happens at the surface and is straightforward to simulate. However, skin is composed of semi-translucent layers where light can enter its structure and interact with components inside. This complicated interaction means that the realism depends on simulating both the static and dynamic components of both surface and structure.

In this thesis, we concentrate on studying, in particular, the effect on appearance of dynamic blood perfusion. Changes caused by emotion such as blushing are difficult to study, so we concentrate on changes caused by facial expression. As the muscles move, blood concentration levels are affected leading to a change in appearance. As the blood concentration level rises, the skin colour will become redder in complexion, and vice-versa when falling. In studying this topic, we aim to test several hypotheses:

- Adding natural perfusion changes alongside classical facial animation greatly improves realism.

- It is possible to develop an algorithm that allows us to mimic blood perfusion changes as a blend between perfusion maps of different facial expressions.
• There exists a relationship between geometric change in facial expression and haemoglobin changes in skin.

As discussed, to reproduce realistic skin appearance we must model how light interacts with the skin’s structure. By accurately simulating these interactions we can determine the proportion of incident light reflected by a skin patch. If we simulate the reflectance on a per-wavelength basis, we can also determine its colour leading to skin appearance. However, the skin’s structure is complex and therefore is difficult to simulate accurately. It is the task of ongoing computer graphics research, and this thesis, to improve this accuracy.

For most inorganic opaque objects, light interactions occur at the surface and can be split into two distinct types [Pho75]. The first type is specular reflectance, which is characterised by light reflecting off an object in a mirror-like fashion. The second type is diffuse reflectance which appears as though light is being reflected in all directions by a material’s surface. It gives an object a chalky appearance, and was used as an early approximation to skin appearance [Nic65]. Both types of reflectance can be combined to realistically render a wide variety of different object appearances.

However, skin is more complex and its appearance cannot be reproduced by surface reflectance alone. This is due to its multi-layered semi-transparent structure which allows light to pass into it. Light that enters the skin’s structure is scattered and absorbed by internal molecules before eventually bouncing out and travelling towards an observer [Tuc00]. These sub-surface interactions mean that a skin appearance is greatly affected by skin’s internal structure giving us the ability to see veins and tattoos under the surface of our skin. It is by modelling these scattering and absorption phenomena that we move away from the chalky appearance of opaque objects and improve greatly the realism of skin reproductions [KB04b].

The causes of the two phenomena, scattering and absorption, are dominated by two different structural elements in skin [Tuc00]. The former is collagen fibrils which strengthen the skin’s structure, whereas the latter is primarily caused by two different chromophores: melanin and haemoglobin. Melanin is used to protect cell nuclei from harmful UV rays. Haemoglobin is used to bind oxygen for transportation to muscles and is present in our blood cells. It is the distribution of these structural elements that modulate the colour and appearance of skin.

For simulation purposes the distribution of the elements is defined on a per-layer basis. A typical skin model may consist of three separate layers (although more are possible) namely the epidermis, dermis and hypodermis. Melanin and haemoglobin are primarily contained in the epidermis and dermis respectively; although haemoglobin has some concentration in both layers. Donner et al. [DWd+08] proposed that skin can be accurately be modelled by encoding the concentration of scatterers in two-dimensional maps. These distribution maps become inputs to the skin appearance simulation.

---

1This can be observed as highlights on the surface of the skin.
2Although this is a simplification of the complexity of the interactions.
1.1 Contributions and organisation

Using the chromophore distribution maps we can simulate the appearance of a skin patch. For each layer we use the distributions to simulate both the scattering and absorption phenomena. We aggregate them together to compute per-layer reflectance, and then convolve these to find overall skin reflectance. The simulations are based on research from the medical imaging community that model the absorption and scattering effects. Originally, Monte Carlo models were used to produce highly accurate simulations of skin. However, these models have been replaced with computationally simpler steady-state approximations to the physical reality. These models provide high-enough accuracy for computer graphics rendering purposes.

Up until now, the computer graphics community has assumed that the input distribution maps were static over time. However, simple phenomenological observations show that although melanin is relatively static [PHCY02], haemoglobin in blood is very dynamic [Hud85]. Some graphics research has provided qualitative assessments of skin colour changes [YW07, YW05, YW04, KMT94]. However, as humans are so adept at analysing skin colour these do not provide sufficient accuracy. It is this aspect, in particular the effect of haemoglobin changes, that we try to improve in this thesis.

### 1.1 Contributions and organisation

In this thesis, we aim to study the relationship between changing facial expression and perfusion changes. To enable this we primarily need two tools; a way to measure accurate facial geometry, and another to measure haemoglobin concentrations. Once we have accurate facial geometry we compute metrics over time that show how it is changing. Finally, we use regression techniques to derive a relationship.

In Chapter 2 we provide an overview of current research relating to appearance modelling and geometry capture. The former is used to extract haemoglobin and melanin concentrations from photographs of a subject, and the latter is used to capture the geometry required as inputs for analysis. We also provide an overview of topics required as a base for understanding the thesis that are non-standard in the field of computer graphics.
In Chapter 1, we describe the initial design of a haemoglobin measurement setup. We construct a device that is able to capture images of a subject and extract haemoglobin concentrations using an efficient lookup table approach. We generate a set of spectral skin reflectance profiles for a range of haemoglobin and melanin values. We then use a calibration process to find a transformation from this spectral space to our camera’s RGB space. This forms a table which we can use to lookup values in an image to find haemoglobin concentrations. To ensure the best possible results we test a variety of different cameras for efficacy. Finally, we empirically analyse a set of initial haemoglobin captures to describe a set of observations. These lead to set of findings that inspire the dynamic blood flow models that we develop.

In Chapter 2, we describe a colour appearance model that allows us to generate a haemoglobin map for a wide gamut of facial expression. For each subject we capture a set of basis expressions that span the psychologically motivated set of human expressions. We then describe an algorithm that takes these basis maps and blends them to generate haemoglobin distributions for a novel expression. We then provide an optimisation to this algorithm so that it can be used in an animation pipeline.

In Chapter 3, we describe how to apply the colour appearance model in an animation pipeline. We demonstrate how it can be applied to a blend-shape model rig.

In Chapter 4, we describe the construction of a novel capture device to form a generalised colour appearance model. The device captures simultaneously blood concentration and facial geometry; where the haemoglobin capture is taken from the initial work in Chapter 1 and the geometry capture is based on a multi-view camera setup. The analysis of this data in the later chapters forms the final haemoglobin model.

In Chapter 5 and Chapter 6, we describe metrics that allow us to measure changes in geometry of facial expressions. Firstly, we describe how to process the images from a multi-view camera setup to compute a point cloud. We then mesh the point cloud using Poisson reconstruction, and refine it using commonly used algorithms. Finally we take this geometry and compute metrics on its surface to describe how the expression geometry is changing.

In Chapter 7, we analyse the captured data and discover a linear relationship between a skin temporal divergence metric, and haemoglobin changes. This finding confirms the hypothesis that such a relationship exists. However, questions are discovered over the origin and nature of the linear coefficients.

The initial contributions are presented in the following publication:


Personal contributions to this paper involved construction and calibration of capture device, performing capture experiments, creating chromophore reconstruction pipeline, involvement in initial
analysis and observations, optimization of blending model, and creation of blended haemoglobin maps for rendering.
Chapter 2

Fundamentals & Related Work

In this chapter we provide the background knowledge, and previous research in dynamic appearance modelling. In Section 2.1, we explore the theoretical background behind classifying facial expression and its origin in psychological literature. In Section 2.2, we explore literature in blood flow modelling and models that provide data-driven appearance changes for different facial expressions. This thesis concentrates on data-derived and data-driven models, and so in the latter sections we describe how to measure the data required for the models. We use models of skin reflectance to generate a lookup table that allows us to reverse engineer blood concentrations from photographs. Typically, the research comes from the Medical Imaging community and therefore we go into greater depth to familiarise the reader with foreign concepts. In Section 2.3, we give background knowledge of basic light scattering theory. This leads to a discussion of current skin chromophore measuring techniques outlined in Section 2.4. In Section 2.5, we give a detailed overview of skin structure to motivate the choice of skin model. In the last two sections we describe research and background measuring the geometry of facial expressions. In Section 2.6, we discuss alternatives to capture mesh representations of a facial expression. In Section 2.8, we explore the basics of differential geometry to allow us in later chapters to describe how the facial expression deforms over time.

2.1 Facial expressions

Charles Darwin [Dar74] began studies on the connection between emotion and expression, providing evidence that certain expressions are consistent across different subjects. He hypothesised that emotions were biological in nature and therefore independent of specific cultures.

Ekman [EFO+71] followed on from this work and stated that although emotions were universal their expressions may be different across cultures. Ekman reduced the set of all expressions to a combination of six fundamental expressions: Anger, Fear, Disgust, Surprise, Joy and Sadness (see Figure 2.1a).

Later, Ekman et al. [EF78] expanded on the six basic emotions by adding contempt and producing a set of facial movements known as the Facial Action Coding System (FACS). The FACS details the muscle movements required to enact the seven basic emotions. It is commonly used in animation and
is a further breakdown of the emotions into their component parts. It consists of a set of 44 facial muscle movements. However, the use of FACS for scientific study requires the use of a trained actor, that is, actors who are able to pull each of the muscle movements individually. This requirement means that it is prohibitive for our purposes and so we opt for a different expression encoding technique.

Plutchik and Kellerman [PK86] later created the concept of a wheel of emotion, which describes a more complete set of expressions that are a blend of a set of different fundamental expressions. In this wheel, they used the original six emotions of Ekman, but expanded on them adding Trust and Anticipation. They set out the formulation that all emotions lay on a wheel that was bounded by these expressions (see Figure 2.1b). Each spoke lies opposite its polar opposite emotion and all emotions lie within the wheel’s gamut.

2.2 Blood Flow Modelling

There are several approaches to blood flow modelling that we could adopt; we give a brief overview of research here:

2.2.1 Mathematical Model

The first approach would be to derive a mathematical model of blood flow changes. The model would provide a mathematical derivation of the dynamics of blood flow in the vascular system, and how they are related to muscle movement.

A fully detailed model of blood flow dynamics coupled with muscle movement would become very complex. Blood flow modelling has been studied in great depth from modelling individual blood vessels [TD04] to entire vascular networks [MACS02]. Coupled with this, the muscles’ requirement for oxygen is controlled by a complicated system of pressure regulation [Urs98] and
vessel dilation [ABMP05]. This would then have to be combined with a model of the effects of soft-tissue deformation on the shape and size of the blood vessels [MLH09].

Although approximations could be made to reduce the complexity of such a model, the usefulness for our purposes would be minimal. Animation requires a sufficient level of realism to be believable, but at the same time provides a model with reasonable performance and minimal parameters. Any increase in parameters drastically increases the time it takes artists and software engineers to understand and obtain good results from an algorithm.

### 2.2.2 Data-driven modelling

The second approach would be to use a database of blood maps for different facial expressions. As the character is animated, the blood maps would be combined to create a realistic blood map for a novel expression.

Many techniques are based on storing tables of colouration for different expressions. Jung et al. [JK06] store a 3D texture of colour changes, where each slice represents a different emotional colouration ranging from the most pale expression to most blood-filled expression. Novel expressions are coloured by moving through these slices. However, the colouration maps are based purely on artist impression and not based on physical measurement. A similar technique is used by both Lin and Cheng [LC12] and Park et al. [PK08] who associate a table of RGB values with each emotion. They then provide a set of interpolation equations that describe how to generate colouration of novel expressions. However, the colouration is based on associated metaphorical connections and perceived colour/emotion correlations.

In another data-driven approach, Yamada and Watanabe [YW04, YW05, YW07] describe a connection between blood flow and hue/saturation changes for a range of expressions. They state that temperature changes in skin are directly correlated with blood flow changes. By measuring temperature and hue/saturation changes at five regions across the face they show a correlation between appearance and blood flow. They then generate a statistical model from their samples, and apply this model to an average face generated from the same samples. They test the performance of the algorithm and found that adding colour to a facial expression generated a much richer emotional response. Their approach generates novel skin appearance from observed values, and moves away from artistic expression. However, changes in hue are applied on a region basis, and do not reproduce fine scale changes.

### 2.2.3 Our Approach

In our initial approach, we use a database of blood maps associated with each emotion. We capture blood concentration maps for each emotion using a custom built measurement device. These maps are extended to blood maps for novel expressions using a blending algorithm. The approach creates more realistic results than using artistic impressions, and can be directly incorporated into animation pipelines. However, it also requires the creation of the measurement device to produce the initial data.
To circumvent the need to create a custom capture device, in our second approach, we create a generalised model of blood flow changes. We hypothesised there is a correlation between changes in the geometry of the facial expression and underlying blood flow. We capture both geometry and blood concentrations and use machine learning to fit a model linking both. This model can be used on any new facial animation with only the requirement of creating a base blood-flow map.

For both approaches to blood-flow modelling, we are required to measure two different metrics simultaneously; a measure of our subject’s facial expression, and a measure of the subject’s blood concentration maps. In the next sections we explore how this is achieved in current research for later use in the thesis.

2.3 Light scattering in skin

For this thesis we require an accurate model of light scattering in skin, and therefore its appearance, for two purposes. First, we use skin appearance models to obtain blood concentration maps from photographs of our subjects. Different concentrations of compounds in skin produce different light scattering properties. We can measure the concentrations by measuring the light scattering nature of a sample of skin. This has numerous uses in the medical imaging community and we examine the large body of research that has originated there.

Secondly, we require a model of light scattering in skin to render the results of our model. The computer graphics community has taken the light scattering models created in the medical imaging community and adapted them for rendering purposes. We therefore follow on from our study of the light scattering models with an overview of how they can be used for rendering of realistic results.

2.3.1 Radiometry

To be able to define the models that relate changes in incident light to changes in the light reflected from an object we must be able to quantify radiation. In the optical frequency range, the measurement of electromagnetic radiation is known as Radiometry. The amount of incident radiation on a patch is known as irradiance, and the amount of light reflected from a point is known as radiance. By first defining these two terms we can define a general model for the reflectance properties of an object.

A typical light source will radiate continuous waves of different wavelengths in all directions. Each of these waves can be seen in its duality as a stream of photons, each possessing an energy inversely proportional to its wavelength. If we have a detector upon which these photons are incident we can measure the rate, or incident photons per second. By multiplying the rate at which photons are incident by their energy we can compute the amount of energy transferred per second. This energy per second is known as radiant power and is measured in Watts (Joules per second).

An isotropic point light source will radiate power in all directions in a perfect sphere. As you move away from the light source, the surface area of the sphere will get larger in proportion to the square of the distance from the light source. At ever increasing distances, the radiant power will be
2.3. Light scattering in skin

spread over an ever increasing surface area. A detector of fixed area, $A$ at some distance, $r$ will therefore detect more photons closer to the light source than if it were placed further away. The power per surface area is denoted radiant exitance and it has units of Watts per square metre.

Additionally, a detector will detect more light facing the light source than if it were turned away. After rotation, the true area exposed to the radiation is a function of the angle between the incoming light direction and the normal of the detector’s face. This function is the cosine of the angle between them and represents the area of the detector projected onto the direction of the travelling light. This law is known as Lambert’s cosine law.

It quickly becomes inconvenient to define the radiation from a light source in terms of the detector’s surface area. It is more convenient to define it in terms of some fixed division of the unit sphere’s surface. This division is known as a solid angle, and it is the counterpart of an angle for a circle. We can then compute the solid angle that the object’s projected surface subtends and multiply this by the power per solid angle. The area subtended by one solid angle of a sphere of radius $r$ is $r^2$, and their units are steradians. This means that there are $4\pi$ steradians in a sphere.

Similarly, there are terms for radiation that is incident upon a surface. The counterpart of radiant exitance is known as irradiance. Given some radiant power, $P$, incident upon a surface with an area, $dA$, the irradiance upon it is the radiant power divided by the total surface area.

### 2.3.2 Radiance

Radiance allows us to consider the amount of power for light emitters that are not point sources. These emitters will have finite surface area and can be light sources themselves or even reflective surfaces. Radiance then defines the quantity of radiant power coming from the emitter that is incident upon a detector.

In a similar way to the effect on the detector, we must again apply Lambert’s law to an area light source. As we turn the light source away from the detector, we reduce the amount of radiation it emits in its direction. This reduction is the same cosine of the angle between the viewing angle and normal of the light source’s area.

The amount of flux radiating from an area element of the light source $dA$ which is then incident


upon an area \( dA \) of the detector at angles \( \theta \) and \( \theta' \) respectively to an axis gives the formulation:

\[
\Delta \Phi = \frac{L \cdot dA \cos \theta \cdot dA' \cos \theta'}{r^2},
\]

(2.1)

where \( r \) is the distance between them.

We can define a solid angle originating from a differential area \( dA_1 \), and subtending the area of the detector \( dA_2 \) as:

\[
d\vec{\omega} = \frac{dA' \cos \theta'}{r^2},
\]

(2.2)

this follows from the definition of a solid angle \( \vec{\omega} \).

Substituting Eq. (2.2) into Eq. (2.1) and rearranging gives us the radiance from a differential patch of a light source, \( dA \), we define radiance as:

\[
L = \frac{d\Phi}{d\vec{\omega} \cdot dA \cdot \cos \theta},
\]

(2.3)

The Eq. (2.3) completes the formal definition of radiance. It states that radiance is the amount of flux radiating from a projected area of a light source in a direction defined by solid angle \( \vec{\omega} \). It is this fundamental quantity that encapsulates both Lambert’s law and directionality of radiant flux.

If we have a definition of radiance from a light source, we can recover the total radiant flux \( \Phi \) from it. We simply integrate the radiance over the positive hemisphere \( \Omega_+ \), and over the area of the light source:

\[
\Phi = \int_A \int_{\Omega_+} d\Phi \cos \theta \cdot d\vec{\omega} \cdot dA
\]

(2.4)

where \( \vec{\omega} \) is an outgoing light vector.

2.3.3 Surface Reflectance

\[\text{(a) BRDF reflectance enters and originates from the same point, } \vec{r}, \text{ on the surface}\]

\[\text{(b) BSSRDF encapsulates the scattering properties of turbid materials by allowing the reflected light to emerge from a different point, } \vec{r'}, \text{ than the incident point, } \vec{r}\]

Figure 2.3: BRDF vs BSSRDF
The simplest type of reflectance occurs at the surface of an object. Light incident from incoming direction $\vec{ω}_i$ is reflected in outgoing direction $\vec{ω}_r$. In surface interactions, the point of incidence is the same as the point of reflectance $\vec{r}$. To describe the surface reflectance properties of an object, we need to define a function that relates how a change in the incoming irradiance $E$ changes the outgoing radiance $L$ (see Figure 2.3). This function is known as a Bidirectional Reflectance Distribution Function (BRDF) [Nic65].

The definition of a BRDF, $f$, at a point on a surface $\vec{r}$, is the differential change in $L$ in direction $\vec{ω}_r$, given a differential change in irradiance $E$ on the same point from direction $\vec{ω}_i$:

$$f_r(\vec{r},\vec{ω}_i,\vec{ω}_r) = \frac{dL(\vec{r},\vec{ω}_r)}{dE(\vec{r},\vec{ω}_i)}.$$

(2.5)

Here we use the directional form of irradiance, which requires Lambert’s law. This states that the change in irradiance, $E$, per solid angle is the projected incoming radiance $L_i$ from direction $\vec{ω}_i$:

$$dE(\vec{ω}_i) = L_i(\vec{ω}_i)(\vec{ω}_i \cdot \hat{n}) \, d\vec{ω}_i.$$

(2.6)

Using the definition of the BRDF, we can define the diffuse reflectivity of a surface, or its albedo $\rho$, representing the overall reflectivity of the surface. The albedo of a surface is the ratio of the differential change in the amount of outgoing flux $d\Phi_r$ to a differential change in the amount of incoming flux $d\Phi_i$. It is given as [NRH77]:

$$\rho(\vec{r}) = \frac{d\Phi_r(\vec{r})}{d\Phi_i(\vec{r})} = \frac{\int_{\Omega_r} f_r(\vec{r},\vec{ω}_r,\vec{ω}_i)L_i(\vec{r},\vec{ω}_r)(\vec{ω}_i \cdot \hat{n}) \, d\vec{ω}_r \, d\vec{ω}_i}{\int_{\Omega_i} L_i(\vec{r},\vec{ω}_i)(\vec{ω}_i \cdot \hat{n}) \, d\vec{ω}_i},$$

(2.7)

where $\int_{\Omega_r}$ represents integrating over the positive hemisphere of directions.

The BRDF is the general form of a reflectance equation and several models have been created that simulate both specular and diffuse reflectance of different surfaces. In the following subsections we describe some of these models.

2.3.4 Specular Reflection BRDF

Specular reflectance is a mirror-like reflectance from the surface of a typically smooth object. For an ideal specular reflector, such as a mirror, the direction of reflectance $\vec{ω}_r$ for an incoming irradiance direction $\vec{ω}_i$ is dependent on the surface normal $\hat{n}$ and is given by:

$$\vec{ω}_r = 2(\vec{ω}_i \cdot \hat{n})\hat{n} - \vec{ω}_i.$$

(2.8)

This law states that the normal bisects the incoming and reflection directions, or that the angle of reflection is equal to the angle of incidence.

For a mirror, the BRDF becomes infinite when the outgoing direction coincides with the law of
reflection. Using the definition of the ideal reflectance direction $\tilde{\omega}_s$, we get the following \textbf{BRDF} for an incoming direction $\tilde{\omega}_i$:

$$f_r(\tilde{r}, \tilde{\omega}_r, \tilde{\omega}_r) = \delta(\tilde{\omega}_r - \tilde{\omega}_s). \quad (2.9)$$

However, this \textbf{BRDF} only applies to mirrors, and does not describe other objects that display different types of specular reflectance. For most other objects light does not reflect specularly in a single direction, but forms a lobe around the direction of mirror-like reflectance.

Of particular interest to skin rendering is the Cook-Torrance model [CT82] which has been shown to approximate well the specular reflectance of skin [WMP+06]. It models the surface of an object as a distribution of tiny micro-facets that each exhibit perfect specular reflectance. The effect of the micro-facets is to reduce the overall reflectance as a consequence of the self-shadowing effect the micro-facets have on each other, $G$, and proportional to the Fresnel reflectance $F_R$ (see Section 2.3.7):

$$f_r(\tilde{r}, \tilde{\omega}_r, \tilde{\omega}_r) = D(\tilde{\omega}_r, \tilde{\omega}_i, \hat{n}) F_R(\tilde{r}, \tilde{\omega}_r, \hat{n}) G(\tilde{\omega}_i, \tilde{\omega}_r, \hat{n}), \quad (2.10)$$

where the self-shadowing effect is defined as:

$$G(\tilde{\omega}_i, \tilde{\omega}_r, \hat{n}) = \min\left(1, \frac{2(\hat{h} \cdot \hat{n})(\tilde{\omega}_r \cdot \hat{n})}{(\tilde{\omega}_r \cdot \hat{h})}, \frac{2(\hat{h} \cdot \hat{n})(\tilde{\omega}_r \cdot \hat{n})}{(\tilde{\omega}_r \cdot \hat{h})} \right), \quad (2.11)$$

where $\hat{h}$ is the halfway vector between the incoming direction $\tilde{\omega}_i$ and the outgoing direction $\tilde{\omega}_r$. The term $D$ is the Beckmann distribution [BS87] and represents the distribution of micro-facet normals parameterised by the roughness parameter $m$. It is defined as:

$$D(\tilde{\omega}_r, \tilde{\omega}_i, m) = \frac{1}{\pi m^2 (\hat{n} \cdot \hat{h})^2} e^{\left(\frac{\hat{h} \cdot \hat{n} - 1}{m^2(\hat{n} \cdot \hat{h})^2}\right)}, \quad (2.12)$$

### 2.3.5 Diffuse Reflectance

The Lambertian model described by Lambert [Lam60] in 1760 describes an ideal diffuse reflector. Light incident upon a point on the surface of a Lambertian reflector has an equal chance of being reflected in any outgoing direction forming a hemisphere around the incident point. This means that the \textbf{BRDF} is constant for all incoming and outgoing directions, but can change only by position on the surface. Using our definition of a \textbf{BRDF} (see Eq. 2.5), we can state that the irradiance $E$ is directly proportional to the radiance $L$:

$$f_r(\tilde{r}, \tilde{\omega}_r, \tilde{\omega}_r) = \frac{dL(\tilde{r}, \tilde{\omega}_r)}{dE(\tilde{r}, \tilde{\omega}_r)}.$$

$$\therefore dL(\tilde{r}, \tilde{\omega}_r) = f_r(\tilde{r}, \tilde{\omega}_r, \tilde{\omega}_r) dE(\tilde{r}, \tilde{\omega}_r),$$

$$\int_{\Omega_r} dL(\tilde{r}, \tilde{\omega}_r) = \int_{\Omega_r} f_r(\tilde{r}, \tilde{\omega}_r, \tilde{\omega}_r) dE(\tilde{r}, \tilde{\omega}_r), \quad (2.13)$$
if the BRDF is constant over all outgoing and using Eq. 2.7 we can show it has a relationship to the diffuse reflectance of:

\[ f_{r,d}(\vec{r}) = \frac{\rho_{r}(\vec{r})}{\pi}. \]  

(2.14)

### 2.3.6 Modelling Translucent Materials

Skin is a semi-translucent multilayered heterogeneous material and as such cannot be modelled by simple surface reflectance. Light passes into the layers of semi-translucent material where it interacts with chromophores embedded in its structure. As a consequence of these interactions, light may re-emerge from the surface on which it was incident. The light that re-emerges is the reflected light and determines the appearance of the material. Skin can be approximately described a turbid material, a material similar to a transparent gel with suspended particulates [ZD06].

The interactions fall into two distinct categories; absorption and scattering. Absorption occurs when a photon is incident upon an atom, and its energy is absorbed. Typically this energy is transferred into the atom’s internal thermal, or vibrational energy. The absorption coefficient \( \mu_a \) is a measure of the average number of absorption events a photon will encounter per unit length, and has units \( \text{mm}^{-1} \).

Scattering occurs when a photon is first absorbed by an atom and then re-emitted in a random direction. Typically, we only deal with elastic scattering where the energy of the emitted photon is the same as the absorbed photon. This means that we can model the interaction as a change in direction of the photon’s path [WEBM06]. The scattering coefficient \( \mu_s \) is a measure of the average number of scattering events a photon will encounter per unit length, and has units \( \text{mm}^{-1} \).

The direction in which the photon is re-emitted is modelled by a phase function. There are several different phase functions; Isotropic, Rayleigh, and Henyey-Greenstein [Lor12]. Of these, the most commonly used is the Henyey-Greenstein function [HG41] which was originally designed to simulate scattering of light in the galaxy. However, it has been shown to be accurate experimentally for skin tissue (see Cheong et al. [CPW90]). It is defined as:

\[ p(\theta) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2g \cos \theta)^{3/2}}, \]  

(2.15)

where \( \theta \) is the angle between the incident direction of the photon and the scattering direction.
For a single layered semi-translucent material the amount of scattering and absorption in a material controls two distinct aspects of its appearance. The dominant effect of absorption is to control the overall spectral distribution of reflected light, i.e. which wavelengths of light are absorbed. The dominant effect of scattering is to control the amount of diffusion the light undergoes within its surface i.e. how much the light scatters radially in the structure. This stems from two simplifying assumptions made about the material; first, the layer is assumed to occupy a half-space; second, the material is assumed to be homogeneous throughout. With these two assumptions the problem becomes two-dimensional and we obtain a per-wavelength scattering profile and a per wavelength reflectance value. The reflectance value controls the overall percentage of a wavelength of light being reflected, and the scattering profile describes the diffusion/colour bleeding into areas surrounding the point of incidence of a light.

For a multilayered material the solution becomes more complicated. This complication stems from the fact that scattered light can bounce between layers, which have different absorption properties. So the scattering properties have a greater effect on the overall reflectance distribution. However, we can still separate reflectance into a radially dependent diffusion profile, and an overall spectral reflectance distribution.

This distinction between the scattering profile, and the spectral reflectance distribution will be an important assumption in our later work. Initial models of translucent materials rely on the reduction of the problem to two dimensions, which we explore first. We then explore subsequent models that allow sub-surface scattering and the ability of light to exit a material at a different point to which it entered.

The term $g$ is known as the anisotropy factor and determines the directional bias of the phase function. For an anisotropy factor $g$ of -1 the scattering is back-scattering; for a $g$ of 0 the scattering is isotropic; for a $g$ of 1 the phase function is forward-scattering (see Figure 2.4). The anisotropy factor is equivalent to the mean cosine of scattering angle:

$$g \equiv \langle \cos \theta \rangle = 2\pi \int_0^\pi p(\theta)\cos\theta\sin\theta\,d\theta.$$  \hspace{1cm} (2.16)

2.3.7 Fresnel Reflectance

Before light enters the boundary of the translucent material to undergo scattering and absorption interactions, it undergoes Fresnel reflectance. This type of reflectance occurs when there is a difference in the refractive index at the boundary. Under these conditions a proportion of the light is refracted into the material, and some is reflected out as specular reflectance. Only the light that is refracted in will undergo scattering and absorption within the structure.
2.3. Light scattering in skin

Figure 2.5: Fresnel Reflectance

For unpolarised light the proportion reflected at the boundary is calculated as follows:

\[
F_R = \frac{1}{2} \left( \frac{n_o \cos \theta_i - n_d \cos \theta_t}{\sin \theta_i} \right)^2 + \left( \frac{n_o \cos \theta_i + n_d \cos \theta_t}{\sin \theta_i} \right)^2,
\]

(2.17)

where \(n_o\) is the refractive index of the material where the light source originates; \(n_d\) is the refractive index of the material into which the light is entering; \(\theta_t\) is the angle with the boundary of the two materials that the light is transmitted at; \(\theta_i\) is the angle of incidence that the light makes with the boundary.

To compute the angle of transmittance and incidence we simply apply Snell’s law:

\[
\sin \theta_i = \frac{n_d}{n_o}.
\]

(2.18)

As an alternative, Schlick [Sch94] has proposed an approximation to the reflection coefficient:

\[
F_R(\vec{h}, \vec{\omega}_r) = F_R^0 + (1 - F_R^0)(1 - (\vec{h} \cdot \vec{\omega}_r))^5,
\]

(2.19)

\[
F_R^0 = \left( \frac{n_o - n_d}{n_o + n_d} \right)^2,
\]

(2.20)

where \(\vec{h}\) is the halfway vector which bisects the outgoing vector, \(\vec{\omega}_r\) and the incoming vector \(\vec{\omega}_i\).

In addition, it is useful in later discussion to use the total Fresnel diffuse reflectance. This is a integral over the outgoing hemisphere of the Fresnel reflectance. It is denoted \(F_{dt}\), and although it has a full solution [WW12], it is commonly approximated by the following [EHR73]:

\[
F_{dt} \approx \begin{cases} 
-0.4399 + \frac{0.7099}{\eta} - \frac{0.3319}{\eta^2} + \frac{0.0636}{\eta^3}, & \eta < 1, \\
-1.4399 + \frac{0.7099}{\eta} + 0.6681 + 0.0636\eta, & \eta.
\end{cases}
\]

(2.21)

where \(\eta\) is the relative ratio of refractive indices \(\frac{n_d}{n_o}\).
2.3.8 BSSRDF

A BRDF only describes reflectance that happens at the surface of an object, and does not extend to all material types. For some, light incident on the surface will pass through it, and be reflected internally before exiting back to an observer. This separation of incidence and exitance point on the surface is not captured by a BRDF. Therefore the Bidirectional Surface Scattering Reflectance Distribution Function (BSSRDF) was introduced to extend the BRDF to incorporate this difference.

The BSSRDF

\[ S(\vec{r}_r, \vec{\omega}_i, \vec{\omega}_r) = \frac{dL(\vec{r}_r, \vec{\omega}_i, t)}{d\Phi(\vec{r}_r, \vec{\omega}_r, t)} \]  (2.22)

separates the exit \( \vec{r}_r \) and entry point \( \vec{r}_r \) of incident and reflected light but defines the same ratio as Eq. 2.5.

It has the visual effect of lateral colour bleeding within skin, which has been successfully employed in computer graphics. The descriptions of internal reflectance that the BSSRDF represents are discussed in the next sections.

2.4 Measuring skin chromophore concentrations

The ability of measuring the haemoglobin concentration in skin is the result of decades of research in the Medical Imaging community. The ability to simulate the appearance of skin of known haemoglobin concentration is the forward problem with respect to finding haemoglobin concentration given an optical measurement of skin. Whereas this research has been used in the Computer Graphics community to increase the realism of rendered skin, measuring haemoglobin concentrations have not been utilised primarily as a result of a lack of necessity. We describe the method of measuring haemoglobin concentration in Chapter 3 in great detail as an introduction to the Graphics community. In the following sub-sections, we give an overview of the models that are used to simulate skin. However, we first present a synopsis of the history of Diffuse Optical Imaging as a segue into a more in depth look from the perspective of Computer Graphics, and as a note to the creators of the original experimental and theoretical work. A more in-depth analysis and review of the sub-discipline can be found in Durduran et al. [DCBY10].

Cutler’s research [Cut31] on the transillumination of breast tissue is one of the first examples of using diffuse imaging for diagnostic purposes. A bright light is placed behind the breast such that it shines through to the front. Where there is a solid tumour in the breast, its optical absorption properties mean that it will absorb the light, and appear as a silhouette to an observer. Although a simple experiment, this principle forms the basis of optical imaging. Light entering a sample of tissue is scattered and absorbed. The light that emerges from the tissue is measured and reveals information about the structure and chromophore concentration of the tissue.

Early measurement of haemoglobin concentration centred on measuring in vitro by measuring the transmitted and remitted light from a cuvette of suspended solution. Early measurements were
2.4. Measuring skin chromophore concentrations

based on Beer-Lambert law \[\text{Lor12}\], and its two-flux enhancement Kubelka-Munk theory \[\text{Kub48}\]. Later, they were based on theory of scattering of light by particles \[\text{VDH57}\]. Improvements in the understanding of skin tissue as a diffuse scatterer of light \[\text{LZ68}, \text{Coh69}, \text{ZP70}, \text{CL71}\] led to mixed results in improving the accuracy of measurements over Kubelka-Munk theory \[\text{HFL75}\]. The research leads to the dipole solution, which solves boundary conditions by using two virtual light sources; one negative, one positive (Section 2.4.5). However, all methods required the measurement of both transmitted and remitted light meaning they could not be applied in vivo.

Eason et al. \[\text{EVNT78}\] present the first technique for obtaining the absorption and scattering coefficients of the medium solely from back-scattered light. They present models for three different types of localised light source, and are tested against in vitro and also preliminary in vivo measurements. They improve upon the dipole solution by introducing an early form of the multipole expansion.

Further work expanded on this idea by leveraging the ability to measure the scattering and absorption coefficients to calculate chromophore concentrations and measure the oximetry of blood in vivo \[\text{THN+88}, \text{CD88}\].

Jobsis \[\text{Job77}\] showed that using near-infrared wavelengths allowed imaging of brain tissue and oxygenation levels. This eventually led to measurement of structural properties of the tissue in addition to the chromophore concentrations using several experimental \[\text{HAD97}\] and theoretical techniques \[\text{AH97}\]. These techniques typically involved measuring time-evolved solutions to the diffusion equations, as the scattering acts to blur the recoverable detail from tissue. This was exemplified by studies by Yodh and Chance \[\text{YC95}\].

Whereas measurements of the optical properties of tissues through the skull require probes to be directly placed on the subject, Cotton and Claridge \[\text{CCH97}\] present a method for measuring the optical properties of skin properties. This led to the development of the SIAScope \[\text{CMCH01}\] which allowed direct measurement of melanin and haemoglobin concentrations directly from colour images.

In the next subsections we present a theoretical overview of the modelling of skin appearance. This provides an overview of the models that we use in this thesis. We frame it as a combination of the purpose of measuring chromophore concentration, and secondly as advancements in Computer Graphics in rendering more realistic images.

2.4.1 Beer-Lambert law

In the early 1700s, the Beer-Lambert law was introduced. The law states that the amount of light absorbed as it passes through a semi-translucent material is directly proportional to the chromophore concentration within the material. Initial experiments were based on measuring the amount of transmitted light as it passed through a cuvette containing a chromophore dissolved in water.

The Beer-Lambert relation states the relationship between the incident intensity of light \(I\), and
the transmitted light $T$ through a turbid fluid or material:

$$\frac{I}{T} = e^{-\varepsilon cr},$$

(2.23)

where $\varepsilon$ represents the extinction coefficient of the chromophores in the material, $c$ represents their concentration, and $r$ represents the distance travelled by the light.

This relationship was used to measure the concentration of haemoglobin. However, the approach has several limitations. Firstly, the measurement can only be made on transmitted light and so thereby only applies to measurements in vitro. Secondly, as a measurement of only transmitted light and not remittance is made, then only absorption is taken into account. This restricts the use of the Beer-Lambert law to only very highly absorbing chromophores.

### 2.4.2 Kubelka-Munk Theory

Kubelka-Munk theory extends the Beer-Lambert law by using a two-flux model. The model requires both the measurement of the remitted and the transmitted flux through a material. In contrast to the Beer-Lambert law this allows the inclusion of scattering materials and can also be used for multilayered materials.

The Kubelka-Munk states the following relationship between the remitted light $R$, and transmitted light $T$ for a turbid material. It states the differential relationship as follows:

$$dT = -(2\mu_a + 2\mu_s)T \, dx + 2\mu_a R \, dx,$$

$$dR = +(2\mu_a + 2\mu_s)R \, dx + 2\mu_s T \, dx,$$

where $\mu_a$ is the absorption coefficient and $\mu_s$ is the scattering coefficient. It simply states that the loss in the transmitted light is due to scattering and absorption and is offset by backscattered light from the remitted light. A complementary relationship for the remitted light also holds.

The solution found by Egan and Hilgeman [EH79] (see for further details) is:

$$R(\beta, K, d) = \frac{(1 - \beta^2)(e^{Kd} - e^{-Kd})}{(1 + \beta)^2 e^{Kd} - (1 - \beta)^2 e^{-Kd}},$$

$$T(\beta, K, d) = \frac{4\beta}{(1 + \beta)^2 e^{Kd} - (1 - \beta)^2 e^{-Kd}},$$

where

$$K = \sqrt{2\mu_a(2\mu_a + 4\mu_s)},$$

$$\beta = \sqrt{\frac{2\mu_a}{2\mu_a + 4\mu_s}}.$$

We can use this formulation to compute the total reflected light $R$, and the total transmitted light.
2.4. Measuring skin chromophore concentrations

$T$ from the measured absorption and scattering coefficients $\mu_a$ and $\mu_s$ of our chromophores.

Both the Kubelka-Munk and Beer-Lambert law assume that the skin reflectance is a BRDF. However, for skin the use of Kubelka-Munk theory was successfully applied by Cotton [CCH97] to predict its colour. This is due to the fact that a BRDF can approximate a BSSRDF if we integrate over the outgoing hemisphere and assume that the scattering radius tends to be very small. It has been separately verified by Matts et al. [MDM07] that there is a correlation between the readings produced by the Cotton model, and direct observations from biopsied tissue.

The above formulation assumes that the scattering material is only a single layer. To extend it to multiple layers, Kubelka [Kub48] propose the use of a geometric series. Transmittance through two layers is the combination of the transmittance through the first layer, and then through the second. However, this does not take into account inter-layer reflection. Light transmitted through the first, can be reflected back to the first, and then reflected back to the second and transmitted through. In fact, there are a possibly infinite number of reflections that can occur between the layers in this way:

$$T_{12} = T_1 \cdot T_2 + T_1 \cdot R_2 \cdot R_1 \cdot T_2 + T_1 \cdot R_2 \cdot R_1 \cdot R_2 \cdot R_1 \cdot T_2 + \ldots,$$

(2.24)

where $T_n$ represents the two dimensional transmission profile of layer $n$, and $R_n$ represents its reflectance. This can be reduced to a geometric series:

$$T_{12} = T_1 \cdot T_2 \cdot (1 + R_2 \cdot R_1 + R_2 \cdot R_1 \cdot R_2 \cdot R_1 + \ldots),$$

(2.25)

which using the geometric series formula becomes:

$$T_{12} = \frac{T_1 T_2}{1 - R_2 R_1}.$$  

(2.26)

Similarly, for the combined reflectance of the layers we get:

$$R_{12} = R_{12} + \frac{T_1 R_2 T_1}{1 - R_2 R_1},$$

(2.27)

2.4.3 Diffusion Approximation

The diffusion approximation addresses the problem of moving from a BRDF to a BSSRDF description of scattering. This means that the surface of the skin has a spatially varying reflectance, and this is found by simulating volumetric sub-surface scattering.

The most fundamental approach to simulating this is the Monte Carlo method. In this type of simulation photon packets representing a single of many photons are beamed into a digital representation of the layers of skin. The photons undergo probabilistic events that describe when they are scattered and absorbed. A brief overview of the algorithm is shown in Figure 2.6, and a full description is outside the scope of this thesis, but full details can be found in Wang et al. [WJZ95].
Launch photon
(dimensionless $s = 0$)

Set new $s$ if $s = 0$
($s = -\ln(\xi)$)

Find distance to boundary $d_b$

Hit boundary ?
($d_b\mu_t \leq s$)

Move $s/u_t$

$S = 0$

Absorb

Scatter

Photon dead?

Weight small?

Survive roulette?

Last photon?

End

Hit boundary ?
($d_b\mu_t \leq s$)

Move $d_b$

$S = S - d_b\mu_t$

Transmit / reflect

Figure 2.6: Overview of the Monte Carlo simulation algorithm
The Monte Carlo method is very computationally expensive to compute (see Figure 2.6), and therefore becomes impractical to use. Although the recent introduction of the GPGPU has led to much more efficient results [FB09, JGS+10], a solution has been found to reduce the problem to a photon diffusion process in skin. The diffusion approximation solves the so-called Radiative Transport Equation (RTE) that describes how the electromagnetic radiation (light) travels through a scattering and absorbing material.

In the next section, we present a brief overview of the RTE and its approximate solution (for full details see Ishimaru [Ish78] and Wang [WW12]).

Boltzmann Equation / Radiative Transfer Equation

If we conceive of the structure of skin as a combination of differential volume elements. We can define that the differential changes in energy per volume element \( d\rho \) is the differential radiance divided by the speed at which it is travelling:

\[
\frac{\partial L(\vec{r}, \vec{\omega}_r, t)}{\partial t} = \frac{\partial L(\vec{r}, \vec{\omega}_r, t)}{c} \frac{dV d\vec{\omega}}{c} \frac{d\vec{\omega}}{c} = (2.28)
\]

where \( \vec{\omega}_r \) is the direction of propagation of the light, \( d\vec{\omega} \) is the solid angle of radiance, \( \vec{r} \) is time, and \( \vec{r} \) is our radial vector of position. This equation in contrast to previous discussion has a relation to time, as it is dynamic. To remove the time dependence of the equation, the model is taken in a limit to its state-steady, when the radiance has become constant over time.

The differential change in energy per volume element can be defined as a balance between energy flowing out of the volume element and energy flowing in:

\[
dP = -dP_{\text{div}} - dP_{\text{ext}} + dP_{\text{sca}} + dP_{\text{src}},
\]

(2.29)

where the terms represent the gain and loss of energy in a differential volume element as below:

- \( dP_{\text{div}} \) - As the beam enters the material it diverges away from the initial collimated state, causing energy loss to surrounding volume elements.

- \( dP_{\text{ext}} \) - This represents the energy lost due to scattering and absorption events within the volume element.

- \( dP_{\text{sca}} \) - Energy scattered from surrounding volume elements into the volume element.

- \( dP_{\text{src}} \) - Energy that enters the volume due to the light source.

By combining the mathematical definition of the constituent parts and substituting into Eq. (2.29) we arrive at the following RTE:

\[
\frac{\partial L(\vec{r}, \vec{\omega}_r, t)}{\partial t} = -\nabla L(\vec{r}, \vec{\omega}_r, t) \cdot \vec{\omega}_r - \mu_t L(\vec{r}, \vec{\omega}_r, t) + \mu_t \int_{4\pi} L(\vec{r}, \vec{\omega}_r, t)p(\vec{\omega}_r \cdot \vec{\omega}_r) d\vec{\omega} + S(\vec{r}, \vec{\omega}_r, t). \]

(2.30)
where $S(\vec{r}, \vec{\omega}, t)$ represents the function representing a source of light.

Here, we make the assumption that the phase function, $p$, is rotationally invariant and so we can equate $p(\vec{\omega}_i, \vec{\omega}_r)$ and $p(\vec{\omega}_i \cdot \vec{\omega}_r)$. Although this is not always the case in general scattering theory, is it the case for the common skin phase functions, such as the Henyey-Greenstein function as described in Section 2.3.6. Primarily, this is due to spherical scatterer on average.

### 2.4.4 The Diffusion Substitution

To solve the time-invariant RTE, an approximation to the radiance is substituted in. The radiance is decomposed into a set of spherical harmonic functions, and further assumed to be nearly isotropic from a highly scattering medium. Therefore only the first two of these harmonics are used, giving radiance the following equation:

$$
L(\vec{r}, \vec{\omega}_r, t) = \frac{1}{4\pi} \Phi(\vec{r}, t) + \frac{3}{4\pi} \vec{J}(\vec{r}, r) \cdot \vec{\omega}_r.
$$

(2.31)

The term $\Phi(\vec{r})$ represents an isotropic spherical expansion and is modified by the energy flow vector $\vec{J}$ which has direction in the greatest change of flow away from the point $\vec{r}$.

By combining the spherical expansion Eq. 2.31 and the Eq. 2.30 and solving we obtain two equations, one scalar differential:

$$
\frac{\partial \Phi(\vec{r}, t)}{c \partial t} + \mu_a \Phi(\vec{r}, t) + \nabla \cdot \vec{J}(\vec{r}, r) = S(\vec{r}, t),
$$

(2.32)

and one vector differential:

$$
\frac{\partial L(\vec{r}, \vec{\omega}_r)}{c \partial t} = (\mu_a + \mu_s') \vec{J}(\vec{r}, r) + \frac{1}{3} \frac{\nabla \Phi(\vec{r}, t)}{= 0}.
$$

(2.33)

Under the assumption that the change in energy density between a photon’s position over a small amount of time is very low; Eq. 2.33 reveals Fick’s Law:

$$
\vec{J}(\vec{r}, r) = -D \Phi(\vec{r}, t),
$$

(2.34)

where the diffusion coefficient $D$ is:

$$
D = \frac{1}{3(\mu_a + \mu_s')}. 
$$

(2.35)

Combining Eq. 2.34 and substituting into the spherical harmonic equation Eq. 2.31 and finally solving for time-independence we obtain:

$$
\Phi(\vec{r}) - \nabla \cdot \left( \frac{1}{\mu_{eff}} \nabla \Phi(\vec{r}) \right) = \frac{S(\vec{r})}{\mu_a}. 
$$

(2.36)
2.4. Measuring skin chromophore concentrations

2.4.5 Dipole Approximation

The diffusion approximation describes how the radiance of light in skin diffuses as it approaches a steady state. For a non-infinite medium, we must take into account boundary conditions. Jensen et al. [JMLH01] introduced to the graphics community the formulation to provide a solution to light diffusion in a semi-infinite slab. For a situation where the incoming medium has different refractive index to the reflecting medium we have a balance of two fluxes.

Fresnel reflectance (see Section 2.3.7) at the boundary affects the amount of light that enters the semi-infinite slab. As this causes Fresnel reflection away from the semi-infinite slab the total upward ambient radiance is non-zero. So for mismatched boundary conditions we simply set the boundary condition at the boundary to balance the downwards radiance and the upwards Fresnel reflected radiance, giving:

\[
\Phi(\vec{r}) - 2AD\frac{\partial \Phi(\vec{r})}{\partial z} = 0,
\]

(2.39)
where:
\[
A = \frac{1 + F_{dr}}{1 - F_{dr}},
\]  
(2.40)

and \(D\) is the diffusion coefficient. The diffusion equation above is a boundary condition, and gives us the solution that the fluence rate at \(z = -2AD\) is zero. Here, the height above the surface is denoted \(z\). The value \(-2AD\) is called the extrapolated boundary, and \(F_{dr}\) is presented in Section 2.3.7. To solve this a dipole is used, which is a combination of a negative and a positive light source that represents the flux, and satisfies the boundary condition that the fluence must cancel at the point \(z = -2AD\).

The sum of the two dipole sources gives the final equation for diffuse reflectance. Here, the position in space has been replaced by the surface distance \(r\) from the position of dipole sources:
\[
R_d(r) = \alpha' \frac{4}{\pi} \left[ \frac{z_r(1 + \mu_{s} d_r) e^{-\mu_{tr} d_r}}{d_r^3} - \frac{z_v(1 + \mu_{s} d_v) e^{-\mu_{tr} d_v}}{d_v^3} \right],
\]  
(2.41)

where \(\alpha'\) represents the scattering albedo \(\mu_{s}/\mu_{t}\), and \(\mu_{tr}\) is the effective transport coefficient (\(\sqrt{3\mu_{a}\mu_{t}}\)). The symbols \(z_r\) and \(z_v\) represent the height of the positive and negative sources above and below the surface of the slab. The distances \(d_r\) and \(d_v\) represent the absolute distance of the sources from point \(\vec{r}\).

The final BSSRDF including two Fresnel terms, one for the incident radiance entry to the slab, and one for the radiance’s exit from the slab is then presented as:
\[
S_i(\vec{r}, \omega_i, \vec{r}_r, \omega_r) = \frac{1}{\pi} F_T(\eta, \omega_i) R_d(\vec{r} - \vec{r}_r) F_T(\eta, \omega_r),
\]  
(2.42)

In addition to the above diffuse BSSRDF Jensen et al. include a single scattering term to take into account photons that do not undergo multiple scattering events. The model suffers from the fact that it only includes a single semi-infinite layer, and the structure of skin has a more complex multilayered nature. It also assumes that the chromophore distribution is homogeneous across the slab.

### 2.4.6 Multipole Approximation

In 2005 Donner and Jensen [DJ05], used the multipole approximation to increase the accuracy of skin appearance modelling. The multipole approximation was designed to deal with thin slabs of tissue, thus alleviating one of the problems of the dipole model. It can be achieved by solving boundary conditions at both the bottom and the top of a thin slab. The boundary conditions are stated in a similar way to Eq. 2.38.

However the two boundary conditions can now only be satisfied simultaneously when there are an infinite number of matching dipoles, hence the name multipole. Given a different refractive index
2.4. Measuring skin chromophore concentrations

at the top and bottom of the thin slab we have the following fluence equations to satisfy:

\[ \Phi(\vec{r}) - 2A(0)D \frac{\partial \Phi(\vec{r})}{\partial z} = 0, \]

\[ \Phi(\vec{r}) - 2A(d)D \frac{\partial \Phi(\vec{r})}{\partial z} = 0, \]

here \(d\) is the thickness of the slab of tissue, and \(A(0)\) and \(A(d)\) are the equivalent of Eq. 2.40 at the top and bottom of the slab.

This gives an infinite set of dipole sources, and therefore an infinite set of extrapolation distances:

\[ z_{r,i} = 2i(d + z_b(0) + z_b(d)) + l, \]

\[ z_{v,i} = 2i(d + z_b(0) + z_b(d))l - 2z_b(0), \]

where \(z_b\) represents the extrapolated distance, and \(l\) is the mean free path length.

The reflectance profile as the summation of the now \(2n + 1\) dipole, which becomes exact in the limit of \(n \to \infty\):

\[ R(r) = \sum_{i=-n}^{n} \frac{\alpha'}{4\pi} \left[ z_{r,i} \frac{(1 + \mu_t d_t) e^{-\mu_d d_t, i}}{d_t^3} - z_{v,i} \frac{(1 + \mu_t d_v) e^{-\mu_d d_v, i}}{d_v^3} \right]. \]

We can now compute transmittance, as the slab is no longer semi-infinite leading to a similar equation:

\[ T(r) = \sum_{i=-n}^{n} \frac{\alpha'}{4\pi} \left[ (d - z_{r,i}) \frac{(1 + \mu_t d_t) e^{-\mu_d d_t, i}}{d_t^3} - (d - z_{v,i}) \frac{(1 + \mu_t d_v) e^{-\mu_d d_v, i}}{d_v^3} \right]. \]

This is effectively the reflectance of the bottom boundary, i.e., the equation is flipped on its head. This also means that the computation of \(d_v\) and \(d_r\) change \([WY15]\), giving \(d_r = \sqrt{r^2 + (d - z_r)^2}\) and \(d_v = \sqrt{v^2 + (d - z_v)^2}\).

2.4.7 Multilayered Multipole Approximation

Donner and Jensen \([DJ05]\) also introduce multiple layers by simulating each layer separately. For each layer in the multilayered structure, we compute a two-dimensional transmittance and reflectance profile from the above thin-slab equations.

In a similar way to the multilayered combination in Section 2.4.2, the combination of layers produces an infinite set of operations. For two-dimensional profiles this is now a convolution:

\[ T_{12} = T_1 * T_2 + T_1 * R_2 * R_1 * T_2 + T_1 * R_2 * R_1 * R_2 * R_1 * T_2 + \ldots \]

Using Fourier transforms this set of convolutions again becomes an infinite geometric series,
giving for the transmittance of two layers:

\[ R_{12} = \frac{R_1 R_2}{1 - T_2 T_1} \]  \hfill (2.46)

where \( R_n \) represent the Fourier transform of the transmission profile, and \( T_n \) the reflectance profile of layer \( n \).

The reflectance becomes:

\[ T_{12} = T_{12} + \frac{R_1 T_2 R_1}{1 - T_2 T_1}. \]  \hfill (2.47)

To retrieve the reflectance and transmittance profiles, the inverse Fourier transform is applied to the result.

Donner et al. \[ DWd^08 \] use the multipole with spatially varying skin parameters. This heterogeneity is captured during the convolution of two layers’ respective transmission or reflectance profiles. They also introduce an infinitesimally thin absorption layer in between the two layers, this allows for further control of the heterogeneity. For the two layers this produces a profile of transmittance which is no longer radially symmetric, and therefore cannot be resolved into orthogonal blur kernels. Donner et al. however provide a fast GPU method based on the Gaussian decomposition work of d’Eon \[ dLE07 \].

Donner et al. \[ DWd^08 \] introduces the simulation of heterogeneity and multiple layers which solves some of the problems with the dipole diffusion model. It has reasonable computation time, with several approximations allowing real-time rendering with low quality degradation (see rendering section below). However, a new problem is introduced, whereby a very thin layer will be too thin to contain a dipole source term. It is at this point that the model breaks down, as the boundary conditions at the top and bottom surfaces can no longer be satisfied.

### 2.4.8 Extended Multipole Approximations

d’Eon \[ dI11 \] attempts to address several problems associated with the multipole method. The multipole suffers when the layer is very thin, such that the dipole sources lie within the boundary of the tissue. d’Eon solves for a very thin layer by quantising time, and applying stepping of the diffusion approximation only in very small time steps. This way the process can be stopped before it reaches the boundary of the layer, by exploiting the time-dependent equations instead.

Habel et al. \[ HJ13 \] demonstrate the same advantages as the d’Eon method but more efficiently. They achieve this counter-intuitively by using a hybrid Monte Carlo method for simulating sub-surface scattering. However, it is only more efficient if the material is not multilayered, and thus is not an improvement for skin.
2.5 Skin Structure

As inputs to the appearance models, we must be able to accurately describe the structure of skin and the distribution of absorbers and scatterers. Skin is composed of three layers (in order from top to bottom, Figure 2.8), the epidermis, the dermis, and the hypodermis. The epidermis is formed of keratinised horny cells that start their life at the lower layers of the epidermis, and are slowly squashed and hardened as they reach the top surface. By the time that the cells reach the top surface, named the stratum corneum, they have died and formed small slivers of keratinised cell membrane. It is at this point in the skin structure that the melanocytes lie, which contain melanin, one of the dominant chromophores of skin. The chromophores melanin and haemoglobin have a dominant effect on the appearance of skin [AP81].

The dermis is an elastic sub-structure that supports the skin, the elasticity is provided by collagen, which is a long chain polymer molecule. It can be divided into two sub-layers: the papillary dermis which lies just below the epidermis, and contains thin blood vessels; and below, the recticular dermis which contains thicker blood vessels, and thicker collagen fibers to give extra support to the upper layers.

Lying between the dermis and the epidermis is the epidermal-dermal barrier, which consists of a set of villous protrusions. They protrude from the dermis into the epidermis, and contain loops of blood vessels which perfuse blood into the epidermis base layer (also called the stratum basale).

The complexity of the skin’s structure means that previous appearance models have used a wide range of different number of layers. A very good discussion of the spectral properties can be found in Donner and Jensen [DJ06]. These can range from a single layer [JMLH01] to several layers [MM02, KB04a]. However, two-layer models have shown to be sufficient for accurate results [TOS+03, TG79, VGJSS89, SSAS98, CMCH01]. In addition, the increase of layers in the model means that the number of parameters increase. When we measure the chromophore concentrations later in the thesis, the number of derivable parameters is restricted to two.

In this thesis we use the two-layer model of skin, consisting of namely epidermis and dermis. For our purposes we consider that the hypodermis is a perfectly white reflector of light [DT03]. It is however, highly unlikely that light will penetrate this far into skin without having been reflected back towards its surface or absorbed. It is this regard that it is a widely accepted practice that the
hypodermis can be ignored \cite{MM02}. We therefore model the dermis as a semi-infinite slab, and we fix the epidermal thickness to 0.25mm \cite{SMPW03}. In addition, we assume that the refractive index is 1.4 throughout the tissue, and 1 in the ambient medium.

### 2.5.1 Skin scatterers and Absorbers

Jacques \cite{Jac98} has measured the both the scattering and absorption spectra and where possible has provide a closed form function that sufficiently replicates the measured data. Melanin can be split into two types of melanin; eumelanin and pheomelanin.

#### Epidermal absorption

The primary chromophore in the epidermis is melanin. There are two types of melanin present in skin; the first, eumelanin, is a brown-black reflector, the second is pheomelanin and is a yellowish-red absorber. The absorption spectra are approximated by simple power laws. For eumelanin $\mu_a^{em}$ we have:

$$\mu_a^{em}(\lambda) = 6.6 \times 10^{10} \times \lambda^{-3.33} \text{mm}^{-1}.$$  

Absorption due to pheomelanin $\mu_a^{pm}$ is defined as:

$$\mu_a^{pm}(\lambda) = 2.9 \times 10^{14} \times \lambda^{-4.75} \text{mm}^{-1}.$$  

There is also a baseline absorption coefficient $\mu_a^{\text{baseline}}$ which covers all non-dominant absorption in skin:

$$\mu_a^{\text{baseline}}(\lambda) = 0.0244 + 8.53 e^{-(\lambda-154)/66.2} \text{mm}^{-1}.$$
We can then define the total spectral absorption of the epidermis as:

$$\mu_{\text{epi}}(\lambda) = C_m(\beta_m \mu_{\text{em}}(\lambda) + (1 - \beta_m) \mu_{\text{pm}}) + (1 - C_m) \mu_{\text{baseline}}.$$  

**Dermis absorption**

The dominant chromophore in the dermis is haemoglobin. Haemoglobin comes in the two forms oxyhaemoglobin and de-oxyhaemoglobin. Jacques [Jac98] provides tables of the absorption coefficients which we use and denote $\mu_{\text{oxy}}$ and $\mu_{\text{deoxy}}$. In a similar way to the epidermis we define the total absorption coefficient for the dermis:

$$\mu_{\text{derm}}(\lambda) = C_h(\gamma \mu_{\text{oxy}}(\lambda) + (1 - \gamma) \mu_{\text{deoxy}}) + (1 - C_h) \mu_{\text{baseline}}.$$  

where $\gamma$ is the fraction of oxy-haemoglobin to the total concentration of haemoglobin, and $C_h$ is total haemoglobin volume fraction.

**Scattering**

The collagen fibres in the dermis are the major scattering elements, and their reduced scattering coefficient can be approximated as a combination of Mie and Rayleigh scattering, respectively:

$$\mu'_{\text{s}} = \mu_{\text{mie}} + \mu_{\text{rayleigh}} = 2 \times 10^4 \lambda^{-1.5} + 2 \times 10^{11} \lambda^{-4} \text{mm}^{-1}.$$  

Donner and Jensen state that the total scatter coefficient is sufficient for the epidermis, but must be scaled by 50% for the dermis.

### 2.6 Geometry capture

Sandbach et al. [SZPY12] have produced a comprehensive survey of facial performance capture and expression techniques. They divide geometry capture into four types of techniques; single image reconstruction, structured light, photometric stereo, and multi-view stereo.

#### 2.6.1 Single-shot facial capture

Single-shot facial capture is the process of extracting facial geometry from a single image without utilizing any stereo reconstruction. Invariably this means that the geometry is based on a parametrised notion of the shape of the face. The parameters of the shape are then optimised such that the distance between a rendering of the shape and the photograph is minimised. Parameters must be solved for shape and texture; pose, scale and position; camera properties and illumination conditions. This results in a large number of parameters, meaning the minimisation problem is difficult to solve.

Blanz and Vetter [BV99] [BV03] use a space of captured shape and texture vectors, termed Three-dimensional morphable model (3DMM). A correspondence is created between a set of scanned faces and a reference face. From this set of vectors they compute a set of orthogonal vectors that
define combined texture and shape space. Finally, using a cost function they find the optimal set of
coordinates in the space to find the closest match to the image on intensity values. Later, the method
is extended to use features points [RV05, MLPM03].

Romdhani and Vetter [RV03] improve on the accuracy of the fitting process by introducing the
notion of Inverse Compositional Image Alignment (ICIA) to the process. Here the cost function, in
addition, minimizes the distance under the inverse projection from the original image to the original
model.

Similar techniques have been developed using a variety of different optimisation strategies such
as Support Vector Machine (SVM) [BKK+08] or different priors [PS09]. The same technique is
demonstrated using video [Bra01], transferring expression changes between subjects [SHK11] and
improved accuracy using multiple views [FPS08]. However, all the methods are based on the same
method of obtaining coordinates in the texture and shape space.

Wang and Lai [WL11] improve upon this by first finding the coordinate but then perform a
further warping of the geometry to improve the match. This warping is performed by using Locally
Linear Embedding (LLE) [RS00].

In general, techniques that utilize single shot facial capture are not sufficient for our purposes.
Firstly, they do not extract key facial details such as wrinkles, dimples etc. Secondly, their accuracy
relies heavily on the quality of the facial geometry database used which on its own takes time and
effort to compile. They do however perform well for facial recognition tasks and for identifying a
particular expression, rather than extracting accurate geometry.

2.6.2 Structured Light

Beumier and Acheroy [BA99] introduce the use of structured light to facial capture. In general,
structured light is the use of a projected fixed light pattern onto a surface to measure its geometry.
The structured pattern is warped by the geometry it is projected onto. By measuring this warping the
geometry can be inversely inferred.

Work by Chang et al. [CVTV05] used structured light to build up a facial expression database.
This database is used in a similar way to Single Shot capture. Here, the database describes a space of
different human expressions. They use this to drive a facial expression analysis tool that allows them
to identify and edit facial expressions from 2D video. The capture technique is based on a Boundary
Coded Structured Light (BCSL) [SCV02] scheme which uses a series of coloured lines projected
onto the subject to perform structured light reconstruction based on similar techniques [Gen96].
The colored pattern of lines encodes the position of the point in the projection. Typically the RGB
channels are used as the selection of colors, allowing the texture of the object to be reconstructed over
time. Vieira et al. [VVSC05] show that the technique can be used to capture real-time geometry and
texture, and Tsalakani et al. [TFMS05] apply it to facial recognition.

Huang et al. [HHJC99, HZC03] increase the accuracy of the capture technique by using a set of
colored sinusoidal patterns instead of defined lines in their pattern. This is based on previous research by Wust and Capson [WC91]. This is then used by others such as Zhang et Yau [ZY06], Hall-Holt and Rusinkiewicz [HHR01] and Weise et al. [WLVG07].

Structured-light techniques produce high-quality geometry capture, at increasingly high capture rates. Using these techniques would allow us to capture the required facial expression geometry. However, the texture is computed by integrating color over several frames. Although, the texture does not change rapidly enough to produce visible differences for animation, it would affect our blood flow concentration measurements.

### 2.6.3 Photometric Stereo

Woodham et al. [Woo80] were the first to introduce photometric stereo. The technique works by estimating surface normals from several images under different illumination conditions. This follows from the fact that the reflectance of a surface is dependent on the normal direction relative to the light source and observer. Hence, if we know the position of light source and camera we can estimate the normal direction. Once the normal field is estimated across the surface of the subject, the geometry field can be reconstructed using integration techniques [ACR05, ARC06]. Prior to the work of Woodham et al., a special case of photometric stereo using a single image was studied by Horn [Hor89, HB86].

Debevec et al. [DHT+00] use a light stage to capture the reflectance field of the subject’s face. This is accomplished by capturing images illuminated from different directions. A rotating arm is used to position the light around the subject to create the different lighting conditions. Then the reflectance field is algorithmically separated into its specular and diffuse components. A Lambertian model is then fitted to the diffuse component to derive the surface normals. Lensch et al. [LKG+03] improve upon this reflectance field acquisition by reducing the number of captures required to reconstruct it. They do this by making assumptions about the homogeneity of the spatially varying nature of an object’s reflectance. This is again similar to the work of Zickler et al. [ZERB05] who trade the resolution of the spatial variance for angular resolution. All these techniques rely on the subject being in a fixed position, for human subjects this relies on the head being held in position.

Debevec et al. [DWT+02] enhance their light stage so that rather than lighting the subject with a rotating arm a series of lights attached to a dome are used. Weyrich et al. [WMP+06] utilize a similar light dome to measure subsurface scattering parameters as well as facial geometry and overall skin reflectance. They measure the parameters for a large number of subjects assuming a single homogeneous layer of sub-surface scattering.

Chen et al. [CGS06] use specularity to acquire the mesostructure of several objects such as oranges and human skin. They detect specularity by thresholding pixels in the image by their intensity. Similarly, the object is captured using lighting from multiple directions. However, one problem associated with reconstruction from specular highlights is the narrow angle in which reflectance
occurs. This means that many captures have to be performed to avoid aliasing the normal field.

In contrast, computing normals from a subsurface scattering object can also be inaccurate. The subsurface scattering effect tends to blur the diffuse reflectance of the surface. Therefore the accuracy of any computed normals will be affected. Ma et al. [MHP07] use polarisation to separate specular and diffuse reflectance. They use normals computed from the specular reflectance to reconstruct high-frequency detail from the surface, and standard structured light techniques to reconstruct low-frequency geometry. A key technique that they introduce is the use of spherical gradients, where the normal can be measured directly from the photograph. The ability to compute the normal field directly removed the need for any fitting of BRDF models, although this is constrained.

The presence of subsurface scattering can also affect the accuracy of structured light reconstruction, where the pattern can become blurred at the edges. Gupta et al. [GAVN13] describe this error, and propose a solution using a set of complementary patterns. In addition, their technique deals with the errors created by interreflections over a rough surface.

The work of Weyrich et al. [WMP06] requires a lot of captures to fit a facial reflectance model. In addition, a hand held probe is used to obtain the most accurate skin parameters, however they do provide a sample for the full range of skin types. Ghosh et al. [GHP08] increase the complexity of the skin model that is fitted including a range of different scattering effects; specular, deep, shallow and single. Following on from this Ghosh et al. [GCP10] measure the Stokes reflectance field of their subjects. From this they compute the spatially varying index of refraction and specular roughness parameters.

### 2.6.4 Multi-view stereo

A stereo reconstruction system combines information from two camera views for reconstructing geometry. Points are determined in three-dimensional space by using triangulation of corresponding points in the two views. A multi-view stereo system combines multiple camera views to obtain a more accurate reconstruction by using a consensus of matching points.

A wide range of research has been done into multi-view stereo reconstruction [SCD06], and especially facial geometry reconstruction [SZPY12]. Its wide use is a testament to the applicability and accuracy of the results. The original work was based on reconstruction using epipolar constraints, and culminated in the COLMAP software [SZFP16], which has been demonstrated to give the best results [SSG17].

Recently, more work has concentrated on the robotics sector, where multiple techniques have been developed to allow multiple moving objects to be simultaneously reconstructed in a scene [ZJB11] [KKJ11] [WKM15]. These work by identifying objects, segmenting them from the background, and reconstructing using the segmented foreground pixels. For non-rigid objects, like facial reconstruction, non-rigid structure from motion (NRSFm) techniques have been developed [JDBDA18]. They normally require very dense point clouds, but more recent work has moved
2.7 Facial Performance Capture

As an alternative to mesh reconstruction, we could use a facial performance capture system. In such a system, facial movements are captured separate from facial geometry. The movements could then be transferred to a generic facial rig as an input to a dynamic blood flow model.

One of the original techniques to facial performance capture is to use a parameterised facial model and fit it to the observed data. Li and Cheng [LRF93] use a parameterised model called Candide that encodes all the expressions in the FACS system [EF78]. Essa et al. [EBDP96] use a similar system of mesh fitting to the observed data, however they use a facial mesh adapted to the subject [MP94, PMS94]. They couple the fitting of the mesh with a facial muscle model that allows them to extract physical parameters related to the facial expression. DeCarlo and Metaxas [DM96] combine deformable face models with optical techniques to provide an improved symbiosis. Pighin et al. [PSS99] fit a model that is a linear combination of facial expression models. Blanz et al. [BBPV03] extend this work by computing a set of basis 3D shapes and textures from a database of facial expressions that describe a space of possible facial expressions to fit.

Other techniques can be split into two types: Active and Passive. Active techniques involve either marking the face with paint, structured lighting, or polarisation. Passive capture avoids the use of any markings or specially constructed lighting.

As examples of active capture, Williams [Wil90] introduces the concept of using markers on the face to drive a three-dimensional facial animating mesh. Changes in expression are a result of manually mapping the markers in the photograph to points on the mesh. A similar technique is used by Guenter et al. [GGW+98].

Bickel et al. [BBA+07] use marker tracking to deform a mesh captured before the facial performance begins. They augment a lower-dimensional real-time performance capture with a temporally slower but higher resolution one. Using a technique similar to shape from shading, the higher-dimensional images are used to imprint wrinkles on to the mesh increasing realism.

Furukawa et al. [FP09] increase the accuracy of motion captured data by estimating and imposing constraints on the deformability of the mesh. The constraints are used to regularize the tracking of markers to increase the accuracy. They also demonstrate their regularisation on their own markerless motion capture system [FP10b].

Although the use of facial performance capture is useful for animation techniques, it is unlikely that a generic rig would provide the geometric resolution for accurate modelling. However, tracking techniques will be useful later for aligning different frames of a capture for temporal measurements.
2.8 Differential Geometry

In this section, we give a brief overview of differential geometry. In Chapter 8, we use this to compute metrics that describe how facial geometry changes over time. We then relate this to derive a model of haemoglobin changes from it. We track changes in the divergence of the skin, and how the curvature of the skin changes.

2.8.1 Divergence

Figure 2.10: Difference between a source (left) and a sink (right) in a vector field

The divergence of a vector field describes how the vector field flows into and out of points in the field. It is a scalar field where each scalar value describes whether or not that point is a source or a sink (see Figure 2.10). Divergence acts on a vector field \( F \), which is our three dimensional Euclidean space \( \mathbb{R}^3 \), where each point, \( x \), in the space has an associated vector. In the standard way, we define a point in the subspace using a set of coefficients \( X_i \), and associated basis vectors \( x_i \):

\[
x = \sum_{i=1}^{3} X_i x_i.
\]  
(2.48)

In a similar way we define the vector \( F(x) \) for every point \( x \) in our space using the set of coefficients \( F_i(x) \):

\[
F(x) = \sum_{i=1}^{3} F_i(x) x_i.
\]  
(2.49)

Finally, divergence is defined as the dot product between the derivative operator and the vector field:

\[
\nabla \cdot F(x) = \sum_{i=1}^{3} \frac{\partial F_i(x)}{\partial x_i}.
\]  
(2.50)

This is the sum of a small component-wise differential change of the field wrt. its respective basis vector.
2.8. Differential Geometry

2.8.2 Curvature

The curvature of a line in two dimensions is inversely proportional to the radius, $r$, of its osculating circle (closest fitting tangent circle):

$$M = \frac{1}{R}.$$  \hspace{1cm} (2.51)

To describe the curvature of a surface in three dimensions, we compute the curvature of two two-dimensional slices through the surface. The slices are formed by intersecting the surface with a plane that contains the surface normal, and represent the minimal and maximal curvatures, which are known as principal curvatures, $M_1$ and $M_2$. Euler has shown that the maximal and minimal curve profiles are orthogonal to each other.

We also compute two further curvature metrics, describing the type of surface curvature. The first metric is Gaussian curvature $K$, for a positive Gaussian curvature, the surface is dome shaped; for a negative, the surface forms a saddle; and for a zero Gaussian curvature, the surface is parabolic or flat. It is computed by taking the product of the two principal curvatures:

$$K = M_1 M_2.$$  \hspace{1cm} (2.52)

The second metric is mean curvature $H$ computed as the mean of the two principal curvatures. If the mean curvature is zero, the surface is locally minimum at that point:

$$H = \frac{1}{2} (M_1 + M_2).$$  \hspace{1cm} (2.53)

To compute the curvature of a surface we use the work of Mokhtarian et al. \cite{MKY01}. This requires a parameterisation $(u,v)$ of the surface.

2.8.3 Surface tangent field

The capture system produces a single two-dimensional manifold described by a set of interconnected triangles. To compute metrics over the surface of the geometry we must define a tangential space.

To form the field, at each point on the surface we define a set of tangential basis vectors. This is composed of the tangent plane and the normal to the surface, $\vec{n}$. The two vectors that define the tangent plane are known as the tangent and bitangent vectors denoted $\vec{t}$ and $\vec{b}$ respectively.

We can use the orthonormality of the three basis vectors to compute them from the geometry. Given the normal vector $\vec{n}$ at point $x$ on the surface, we can compute the tangent and bitangent. We first compute the tangent vector as the cross product between the normal vector and the $\vec{x}$ basis vector of our $\mathbb{R}^3$ space:

$$\vec{t} = \vec{n} \times \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}.$$  \hspace{1cm} (2.54)
We then compute the bitangent vector as the cross product between the tangent vector $\mathbf{t}$ and the normal $\mathbf{n}$:

$$\mathbf{b} = \mathbf{t} \times \mathbf{n}. \tag{2.55}$$

These vectors form a two dimensional tangent plane around the point $x$ which we denote $\mathbf{s}$:

$$\mathbf{s} = t + b. \tag{2.56}$$

Finally, we end up with the matrix transformation $\mathbf{M}$ that is a combination of the basis vectors above:

$$\mathbf{M} = \begin{pmatrix} \mathbf{t} & \mathbf{b} & \mathbf{n} \end{pmatrix}, \tag{2.57}$$

which allows a vector to be transformed into the local space.

### 2.9 Discussion

In this chapter, we have described the previous research into a wide range of topics used in this thesis. The central theme in the thesis regards the connection between changing facial expression and blood concentration in skin. As a result, we first explore how to describe facial expressions. We then look at previous research into blood flow modelling. From this, we conclude that our approach to modelling should be based on haemoglobin measurements of different expressions to provide a basis for our model. To obtain accurate measurements, we require an accurate model of skin reflectance to match against and reverse engineer the concentrations. We therefore provide a brief overview of skin scattering theory, as a basis for a further discussion about skin modelling and therefore chromophore measurement. Later in our thesis, we try to determine a relationship between differential geometry and blood concentration changes. We therefore discuss the techniques to capture geometry, and a brief overview of differential geometry mathematics.

In the next chapter, we describe how we bring together previous work to measure chromophore concentrations. We describe the theoretical background, and practical steps that we had to take allowing us to perform initial measurements of several subjects. We analyse the accuracy of these measurements, and use them to provide some observations about blood flow in general.
Chapter 3

Haemoglobin Analysis & Initial Findings

In this chapter, we describe the method used to obtain chromophore concentration measurements from a subject. The measurements will later be used to drive our dynamic appearance models. The models require that we have measurements of the subject’s entire expression at once. This requirement means that we have to use a non-contact measuring device.

As we require non-contact measurements, we leverage the work of Cotton and Claridge and their SIAScope [CCH97], which recovers the haemoglobin concentrations from a photographs of a subject. We combine this with research on camera calibration [FHH98] to allow us to apply the procedure to any combination of camera and light source that we require. In addition, we also explore the effect that camera parameters (such as compression, noise and bit-depth) have on the quality of both the calibration and haemoglobin concentrations that we retrieve from the process.

From the previous chapter, we know that the colour of skin is a combination of two distinct effects. The first is scattering which is caused by elements of the skin structure such as collagen fibres. The second is absorption caused by chromophores embedded in the skin’s structure. The two dominant skin chromophores are melanin and haemoglobin, which we embed in a multi-layered skin model that simulates its optical properties.

This simulation allows us to produce a spectral lookup table that spans a range of different melanin and haemoglobin concentrations. We can then transform the spectral reflectance profiles into the camera’s RGB colour space. This transformation is obtained through a calibration process; first, a colour calibration target of known spectral reflectance is photographed; second, the imaged pixel values are then used to create and solve a linear system that embodies the transformation between spectral and RGB space. However, this calibration process is prone to error sources, due to the inherent properties and features of a particular camera. So, we investigate a wide range of possible sources and study their effect on said process. This allows us to make an informed decision about the best camera to use. After the steps are complete, we have a lookup table that maps from RGB values to melanin and haemoglobin concentrations.

Once we have this lookup table, we can start to map pixels in the image to chromophore

---

1 A brief history of the techniques used in measuring haemoglobin is given in Section 2.4.3
concentration values. First, we remove specularity using cross-polarisation methods, and then normalise the image to remove effects of shading. The particular choice of normalisation procedure can affect the accuracy of the results, and so we investigate effects of different algorithms on the quality of results. Once the lookup table is normalised it forms a scattered set of data points in a three-dimensional RGB space. Therefore, the final step is to use scattered data interpolation techniques to transform pixel values in our also normalised image into melanin and haemoglobin concentrations. Again, we analyse the effect of our camera parameters on the interpolation process and present our findings.

In the last section of the chapter, we discuss some initial findings that are produced from some preliminary experiments. These experiments involve processing some expressions of a few subjects, and computing the haemoglobin concentrations. We study the images and identify a set of properties that form the basis of the dynamic models that we create.

## 3.1 Physical Construction

(a) Photograph of the physical set-up.  
(b) The axis of the polarisation filter must be aligned with the plane of polarisation.

![Figure 3.1: Photograph of physical set-up and diagram showing the plane of polarisation.](image)

As noted earlier, we base our physical construction on the design of Cotton and Claridge’s SIAScope [CCH97]. The set-up consists of two Portaflash 336VM flashes, and a Fuji Finepix S2 Pro camera, where the two flashes are connected directly to the camera for synchronicity. To facilitate the time slicing experiments described later, and for ease of operation camera control is automated by connecting it to a computer. Once an image is taken, it is transferred automatically through a firewire cable connected to the machine.

Of the two types of reflectance that comprise the appearance of skin our physical set-up must only image the diffuse reflectance, and not specular reflectance. The diffuse reflectance is a product of our skin’s internal composition and is therefore correlated with the skin’s haemoglobin concentration. The specular reflectance only occurs at the surface of our skin and as a consequence doesn’t contain...
any information about the underlying skin composition. It is therefore imperative to remove specular reflection to be able to obtain accurate measurements of the underlying skin tissue.

To filter out surface reflectance, we take advantage of the properties of polarised light. Linear polarisation is preserved by specular reflected light, and not by diffusely reflected light (see Appendix A). We can take advantage of this by polarising the light radiating from our flashes. We then know that any specularly reflected light is of the same polarisation as the original source. By then placing a perpendicularly polarising filter over the lens the specular light is filtered out. This process is known as cross-polarisation, and has been used in previous research [Möl96, MHP+07, NFB97]. However, the polarisation of linearly light is not rotationally symmetric, and therefore to maximise the amount of filtered light we placed both flashes and camera in the same physical plane. A deviation from this plane rotates the polarisation basis of the light sources with respect to that of the camera, and reduces their effectiveness.

Figure 3.1a shows the physical construction of the device. It shows three apertures with the camera in the centre, and the flashes either side. The cardboard frame is constructed in such a way that light can only exit and enter through the apertures, which are covered with the polarising filters. Figure 3.1b shows an idealised view of the polarisation, to demonstrate the concept. The polarisation of the light is defined by two basis vectors: s- and p-polarisation, along with the third vector, the direction of light travel. As the lenses and aperture are in a plane defined by one polarisation vector, and the direction of light travel, the new direction of specularly reflected light is also in this plane. This means that the orthogonality of the radiant light’s, and the reflected light’s polarisation bases is preserved. If it weren’t, the effectiveness of the polarisation filters would be reduced.

Figure 3.1a also shows black felt on the walls, and below the camera device. This is used to ensure only polarised light illuminates our subject. If the wall were left uncovered the light from the flashes could radiate the wall, depolarise, and reflect back on our subject. The black felt prevents this by absorbing all light incident upon it. The felt under the device stops any possible light leakage from the flashes. In addition, the camera lens is wrapped in felt so that it exists in a completely light controlled environment.

3.2 Software Pipeline

In this section, we describe the methodology to calculate per-pixel haemoglobin values from a linear RGB digital image. We construct an RGB appearance lookup table in our camera’s colour space, which is parameterised by the two dominant chromophores; melanin and haemoglobin. We use this to solve the inverse problem of finding the best fit melanin and haemoglobin value for each per pixel RGB value in our image.

To generate our RGB lookup table, we first use a skin model to compute a lookup table of spectral reflectance profiles. This lookup table spans a gamut of common haemoglobin and melanin concentrations. To simulate how these reflectance profiles would appear when illuminated, we
multiply point-wise by the spectral power distribution of our light source. Finally, we integrate the spectral power using per-channel curves representing the camera’s sensitivity to different wavelengths of light. To find the spectral power distribution of the light source, and the sensitivity curves we perform a calibration process described in Section 3.2.2.

3.2.1 Skin Lookup Table

Our lookup table maps known chromophore concentrations to a spectral reflectance distribution. To build a mathematical model of skin reflectance, we assume the Dichromatic Reflectance Assumption, where the reflectance of skin is a linear combination of specular and diffuse reflectance. We further assume that the specular reflectance is uniform in colour across our subject’s expression, although we remove the need for this assumption later. The diffuse reflectance is modulated by the chromophore concentrations, and is computed using models explored in Chapter 2. Combining these forms our model of skin reflectance and allows us to create a more formal definition.

We now provide a definition of the reflectance distribution $S(m,h,\lambda)$ of infinitesimally small skin patch, with a melanin concentration $m$ and haemoglobin concentration $h$. It is defined as a linear combination of its specular reflectance distribution $P(\lambda)$, and diffuse reflectance distributions $K(m,h,\lambda)$; both normalised such that their area is one. They are then scaled by respective coefficients $d, s$. The first coefficient represents the amount of diffuse shading of the surface as the normal to the patch turns away from the light source and the second represents the amount of specular reflectance. Under this definition, we define the total reflectance as:

$$S(m,h,\lambda) = dK(m,h,\lambda) + sP(\lambda).$$  \hfill (3.1)

---

2Governed by Lambert’s Law (see Section 2.3.5)
This definition provides us with the per-wavelength ratio of radiance to irradiance for our patch. We then point-wise multiply this by the spectral power distribution of our light source to find the spectral power distribution of reflected light. With the spectral power distribution of our light source \( L(\lambda) \) the definition becomes:

\[
S(m, h, \lambda) \circ L(\lambda) = dK(m, h, \lambda) \circ L(\lambda) + sP(\lambda) \circ L(\lambda).
\] (3.2)

Our physical set-up removes specularity using cross polarisation. Using this, we can eliminate our specular reflectance distribution:

\[
S(m, h, \lambda) \circ L(\lambda) = dK(m, h, \lambda) \circ L(\lambda).
\] (3.3)

To compute our skin lookup table we use a skin model (see Chapter 2) to compute diffuse spectral reflectance profiles for a range of melanin and haemoglobin concentrations. For each skin model we assume that skin is composed of a multilayered half-space reducing the problem to two-dimensions over the surface of the skin. That is, we remove the effect of the depth dimension, as we can assume all light is either reflected out from the surface, or absorbed internally. For some models, such as the Kubelka-Munk theory this assumption is already part of the model, as it is a simple two-flux model. However, for BSSRDF models we use a common technique of integrating over the outgoing hemisphere to reduce it to two-dimensions. We compute each wavelength in turn by varying the wavelength dependant scattering and absorption coefficients, under the assumption of elastic scattering.

Once we have our skin lookup table in spectral space we need to transform it into RGB space. In the next subsections we discuss how to define the transformation and how to calibrate for it.

### 3.2.2 Spectral camera transform

In this section, we transform the spectral reflectance lookup table in the previous section into our camera’s RGB colour space. For a patch of our subject’s skin, light from an external source illuminates it with a given spectral power distribution. The reflected light then enters the camera’s lens and is incident upon the imaging sensor where it is converted into an electrical signal. The sensor is composed of multiple photosites, representing pixels. When the reflected flux is incident upon the sensor the conversion to electrical current is described by the quantum efficiency curve. The pixel brightness that we see in the image is directly proportional to this current.

However, if all photosites responded in exactly the same way to light as described, we would only obtain a monochrome image. To obtain a colour image, colour filters are placed over each of the photosites. The filters only allow a select broadband spectrum of wavelengths to pass through. In all there are three different types of filter allowing approximately the red, green and blue parts of the spectrum through. As each photosite
can only have one colour filter the filters are arranged in a Bayer matrix (see Figure 3.3).

The arrangement of the colour filters allows missing R, G, and B values for each pixel to be interpolated from surrounding pixels. Although, the peak sensitivity of the green cones in the eye are muted by their lower proliferation in the eye [WC83] the rods in the eye are more sensitive to green wavelengths and so they appear brighter [Hec87]. After interpolation, we obtain a full three-channel colour image in respond to reflected light.

![Bayer Matrix](image)

**Figure 3.3: Bayer Matrix**

We can now provide a mathematical definition of the camera transform. Light of given distribution $L(\lambda)$ is reflected from an object of reflectance distribution $R(\lambda)$. This light is incident upon a photosite with a filter that allows light to pass with the distribution $F(\lambda)$. The photosite converts incident flux into current with quantum efficient curve $Q$ and integrate the current over all photon energies. The integrated current is proportional to the pixel’s brightness $P$:

$$P \propto \int Q(\lambda) (R(\lambda) L(\lambda) F(\lambda)) \, dA,$$

where $dA$ is the area of the photosite.

We further simplify this definition by using some assumptions about our sensor. Typically, the quantum efficiency of the photosites will be constant for the visible wavelengths of light, and given good manufacturing, the per pixel area will also be constant. This means they can both be absorbed into the proportionality giving us:

$$P \propto \int R(\lambda) L(\lambda) F(\lambda) \, dA. \tag{3.5}$$

For practical purposes, we can approximate the continuous integral above as a discrete sum over the visible wavelengths of light:

$$P \propto \sum_{\lambda=400nm}^{\lambda=700nm} R(\lambda) L(\lambda) F(\lambda) \Delta \lambda. \tag{3.6}$$

Finally, the total current is converted into an $n$-bit image using an analogue-to-digital converter. This conversion quantises the available pixel values, and also clamps pixel values to maximum brightness.

---

3The greater number of green filters allows for more accurate measurements of green colours, due to the dominance of green in human vision.
3.2.3 Spectral camera calibration

Our description of the transform from the object’s reflectance properties to RGB values requires that we know two things: the spectral power distribution \( L(\lambda) \), and the RGB filter curves \( F(\lambda) \). These curves can be obtained from manufacturer’s data, and/or measured precisely using spectral instrumentation. However, we can obtain both simultaneously, that is the product \( L(\lambda)F(\lambda) \) using a calibration target using the method we describe in this section. The method has the advantage that we can also easily and accurately account for changes in the spectral power distribution of the light over different acquisitions. It is based on the work of Finlayson and Hubel [FHH98]. They introduce a method for extracting a camera’s sensitivity from a calibration target. As we use small localised area lights for our capture, we add correction for the variability of light across the target.

The method consists of solving the Eq. 3.6 for a set of patches with known reflectance. We take an image of a calibration target with a series of patches of known reflectance, giving us a set of RGB observations per patch. We can then combine all of these values into a system of linear equations whose solution is a combined filter and light curve for each of the R, G, and B channels. We can also absorb the global constant of proportionality into the combined light curve, allowing us to create a system of linear equations, and normalise for this at a later point.

The system of linear equations is simply the matrix form of Eq. 3.6. We form a pixel matrix \( P_{RGB} \) of size \( 3 \times n \) containing our observations of the RGB values of our patches. We define a \( 3 \times l \) matrix \( F \) of our combined filter and light source values per \( l \) wavelengths and per the three channels. Finally, we define our per patch reflectance values as an \( l \times n \) matrix \( R \). Using these three matrices we can define our system of equations as:

\[
P_{RGB} = F_{RGB,\lambda}R_{\lambda}\Delta\lambda.
\]  

(3.7)

In Finlayson and Hubel, they state that a direct least-squares minimisation without regularisation quickly becomes degenerate. So, we use their method to perform a constrained optimisation with smoothness regularisation techniques to allow us to find a non-degenerate solution.

We use the following of their constraints to regularise the optimisation:

1. Each value in the solution must be positive, as both negative filter values and negative light spectral power distributions are physically impossible.

2. Both the light source spectral power distribution and photosite filters generally have smooth spectral variation. This means that we minimise the second order derivative of our solution around the local neighbourhood of every point.

3. At some minimum and maximum wavelength the colour filters cut out all light incident upon the photosites. This means we apply clamping to zero at both ends of the solution.
They also state a constraint of unimodality on the spectra. However, this is an idealised view of camera reflectance spectra. In Mauer and Wueller [MW09], we can see even from a relatively small set of results there is dual modality in many camera’s spectral sensitivity curves; notably Canon EOS 450D, and Arriflex D-21. In addition, recall our method obtains a combined light source, and spectral camera calibration. Therefore we must also take into account any light sources that have dual modality, for example LED light sources (see Figure 6.5). In the interest of generality, we therefore remove this constraint, as this will allow to explore different cameras and light sources later in the thesis.

The first constraint can be applied by using a least-squares solver that only allows positive solutions. Several contrained optimisers and non-negative optimisers are provided by MATLAB, and we pick from them. Further details of the algorithms used can be found in the documentation.

The second constraint states that the curve be smooth. Mathematically, the smoothness of the curve increases as its second derivative approaches zero. This can be encapsulated in matrix form, by convolving the curve with a discrete second-order derivative operator, and solving for equality with zero:

\[
F_{RGB,\alpha}S_{\lambda} = \begin{pmatrix}
F^1_R & \cdots & F^l_R \\
F^1_G & \cdots & F^l_G \\
F^1_B & \cdots & F^l_B
\end{pmatrix}
\begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
-2 & 1 & 0 & \cdots & 0 \\
1 & -2 & 1 & \cdots & 0 \\
0 & 1 & -2 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & 1 \\
0 & 0 & 0 & \cdots & -2 \\
0 & 0 & 0 & \cdots & 1
\end{pmatrix}
\begin{pmatrix}
\alpha \\
0 \\
0 \\
0 \\
\alpha
\end{pmatrix}
= \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0
\end{pmatrix}
\]

(3.8)

The third constraint clamps the minimum and maximum wavelengths to zero, where the range of wavelengths are typically the visible range. In matrix form, this becomes:

\[
F_{RGB,\alpha}M_{\lambda} = \begin{pmatrix}
F^1_R & \cdots & F^l_R \\
F^1_G & \cdots & F^l_G \\
F^1_B & \cdots & F^l_B
\end{pmatrix}
\begin{pmatrix}
\alpha \\
0 \\
0 \\
\cdots \\
\alpha
\end{pmatrix}
= \begin{pmatrix}
0 & 0 \\
0 & 0 \\
0 & 0
\end{pmatrix}
\]

(3.9)

where the parameter, \(\alpha\), controls the strength of this constraints.

Note, these equations can be combined into a single matrix equation:

\[
\begin{bmatrix}
P_{RGB} & 0 & 0
\end{bmatrix}
= F_{RGB,\alpha} \begin{pmatrix} R_{\lambda} & S_{\lambda} & M_{\lambda} \end{pmatrix} A_{\lambda}.
\]

(3.10)
where \( \mathbf{0} \) is a zero matrix as described in the above second and third constraints.

However, for our particular evaluation calibration we cannot use a direct linear least-squares solver. As shown in Figure 3.7a, not all channels obey the third constraint, where minimum and maximum values of the spectra are zero. We can see that the red spectra has a positive value at the maximum wavelength, and so fails the constraint. However, we still want to apply clamping to the red minimum, and the ends of the other colour channel spectra. To solve all channels simultaneously, we therefore need to make a non-linear system of equations. An alternative, would be to be the split the matrix equation, and run each channel separately. The non-linear solver requires a function to minimize; in our case this is simply the difference between the calculated right-hand side of Eq. 3.10 and the measured left-hand side. To avoid clamping the maximum wavelength of the red channel, we artificially set the result of the clamping constraint result to zero.

### 3.2.4 Implementation

*Figure 3.5: Common ColorCheckers*

In choosing a suitable calibration target the larger the number of patches it has, the higher calibration sampling frequency we can obtain. X-Rite (formerly MacBeth) produce a number of different targets with a different numbers of patches. They are used primarily for photographers to control the colour space of an image, and have tightly controlled reflectance spectra. The standard ColorCheckers, Figure 3.5, give us a choice of 24 patches, 140 patches, and 240 patches. However, the degrees of freedom are reduced by the number of patches having the same spectrum. Taking into account the ColorChecker with 240 patches we can obtain a maximum resolution of 6nm, and 9nm for 140 patches after taking into account linear dependency of some patches. This linear dependency can be observed empirically, and by design, from the reference spectra used in solving the matrix equation, Eq. 3.10. These patches are the grey, white and black patches that have an uniform reflectance spectra.

The left hand side of Eq. 3.6 requires that we collect the observed RGB values of our patches. We use a simple system where a user clicks a set of points corresponding to the left hand side, right hand side, centre of the top and centre of the bottom rows. We then use these points to interpolate the centres of all the other patches by calculating the projective transformation of the ColorChecker in the image. Once we have the centres of all the patches we use them to seed a flood fill operation that creates a mask for each one of the patches. Within each mask we take the average of the RGB pixel
values which helps to reduce noise in the measurement. These average values form the observations in the linear system.

### 3.2.5 Uneven Illumination

![Figure 3.6: ColorChecker DC Illumination correction](image)

The linear system of equations requires that the light’s spectral power distribution is constant across all patches. However, the calibration target can be illuminated by a source that varies both in strength and colour across the target, for example, a spotlight. A clear difference in the uniformity of light can be seen in Figure 3.6 after the differences have been extracted and corrected for. Colour variations are less likely than luminance differences, as in general lights emit photons of different wavelengths in all directions homogeneously. However, luminance variations are common and we must correct this before we can solve the system of equations.

To correction for the variation, we divide each patch’s RGB observation by the strength of illumination on that patch. We compute the per-patch strength by interpolating between white patches around the border and at the centre of the calibration target. These white patches are Neutral Interface Reflector (NIR) of varying reflectance, and so the length of the RGB vector will be in direct proportion to the irradiance at that patch.

To interpolate between patches we use a spring-based interpolation method provided by John D’Errico [D’E04]. The method operates on a matrix of numbers containing two types of value: unknown and known. In our case, the known values are the measurements from the white patches, and the unknown values everything else. For every unknown value, $V_u$, we compute a set of horizontal and vertical neighbours. We then use these to construct a series of linear equations to solve. For neighbours with known values, we equate the two:

$$V_u = V_k;$$

for neighbours with unknown value, $V'_k$, we minimize the difference:

$$V_k - V'_k = 0.$$  

Solving this using least-squares methods lead to unknown values that are minimally distant to their neighbours, while approximating their known neighbours as closely as possible.
3.3. Evaluation

Once we have compensated for any variation in illumination strength we can solve for our spectral calibration Eq. (3.10). The solution of the linear system will only be proportional to the true calibration data for our camera and light source. This is due to the fact that in the dividing by the illumination strength we are normalising the strength to one; meaning that the resulting curve will not be radiometrically calibrated. However, we do not require this for our haemoglobin analysis as it will be automatically resolved when we remove shading in the next section.

3.3 Evaluation

(a) Synthetic ColorChecker SG

(b) Calibration fit under ideal conditions

Figure 3.7: Calibration data used to perform evaluation

To test the accuracy of the calibration procedure we test it against a simulated calibration target. We use measured data of a camera and light source combination and combine this with known spectra of a ColorChecker SG calibration target (see Figure 3.7a). We use a combination of data from an AVT Guppy camera, and a halogen light source. We then apply our calibration procedure to the image and compare the results to the known procedure.

We show the results of the calibration procedure in Figure 3.7b. We show the separate curves for the red, green and blue channels. To provide a fair comparison we scale the calibration results so the peak of the true green spectrum matches that of the same point on the fitted spectrum. On the actual spectrum this peak is at the 540nm point. In comparison to the combined light source and camera spectra, we see that the shape and relative magnitude of the fitted spectra is a good fit for the original spectra.

We wish to also evaluate the calibration procedure under varying imperfect conditions, informing us which cameras we can use when constructing a capture set-up. We wish to know whether or not a camera is suitable not in the calibration procedure but for later parts in the pipeline for haemoglobin analysis. In the remainder of this section we evaluate the effect different image properties have on the calibration accuracy, and subsequent haemoglobin analysis error. We investigate the effect that the properties bit depth reduction, compression, gamma curve correction and noise have on the calibration procedure.
3.3.1 Bit depth reduction

![Figure 3.8: Fit using 8-bit image of calibration target](image)

In choosing a camera for performing haemoglobin analysis there are various bit depths that cameras can capture images at. The higher the bit depth, the lower the quantisation error caused by it. We therefore need to determine the effect of this quantisation on the calibration fit.

To test the effect of changing bit depth on the calibration accuracy we perform a simple test. In the ideal condition fit we use a bit depth of 16-bit encoded in a PNG file. We reduce the bit depth to 8-bit, the standard bit depth for a JPEG image, but for a controlled experiment keep to PNG file format to preserve linearity and no compression. We then perform the calibration procedure and show the results in Figure 3.8.

The results show that the accuracy is not greatly affected by the reduction in bit depth. However, it does show that the higher the bit depth of the camera the better the calibration accuracy.

3.3.2 Compression

Many commercial cameras use compression as part of their pipeline to reduce storage requirements. In general, these algorithms are designed to work with the human perceptual system and so do not appear to create visual artefacts. However, they can create problems when processed by computer. Many cameras build compression into their hardware pipeline which makes it impossible to circumvent. As a result, we need to know the effect of compression on the accuracy of our calibration.

To understand the effect, we perform the same calibration procedure after applying compression to the image of our ColorChecker. We pick JPEG compression, which we apply using the ImageMagick toolkit, which converts the image from 16-bit PNG file format to 8-bit JPEG. We vary the JPEG compression quality parameter from 50 to 100, where 0 is the most compressed to 100 which is least compressed. The reduction in bit depth is an unfortunate consequence of the limitations of the JPEG format which does not allow 16-bit. However, as shown in the previous subsection the bit depth reduction does not have a significant effect on the overall calibration accuracy.

Figure 3.9 shows the results of the experiments. We can see from the figure that even a small
3.3. Evaluation

Figure 3.9: Comparison of different image compression qualities

Figure 3.10: Comparison of different gamma curve reconstruction errors
amount of compression has an effect on the accuracy of calibration. This is especially true for the red channel reconstruction where the spike is over-estimated even when the compression quality is at its highest. As the quality of the compression is reduced, then the reconstruction gets progressively worse in both the blue and green channels as well.

### 3.3.3 Gamma curve correction

Gamma correction approximates the non-linearity of a CRT monitor output, but also approximates the non-linear response of the human visual system. In general, humans are much more sensitive to changes in dark tones than they are to light. The gamma function is a power function that acts on a pixel value, and has the form:

$$
g(x, y) = i(x, y)^\gamma$$

(3.11)

where $g(x, y)$ is the gamma-corrected pixel value, $i(x, y)$ is the original pixel values, and $\gamma$ is the exponent which is typically 2.2 [AMCS96]. Some cameras will automatically apply gamma which cannot be adjusted, and so we must study its effect on the calibration process, and can vary for different cameras and displays.

In our calibration procedure, we remove our gamma correction before we solve the system in Section 3.2.2. However, we may not know the gamma exponent and may need to measure it. Debevec and Malik [DM08] describe a system of finding the gamma curve, and non-linearity of the DAC, by imaging a scene at multiple exposures. At increasing exposures, a pixel’s intensity is multiplied by a factor defined by the difference in the F-stop values [AT12]. If a gamma curve is applied, then these ratios will be changed, and knowing the F-stop values will allow us to identify this curve. Debevec and Malik supply an algorithm to perform the fitting of the curve using a least squares optimisation process. However, in some cases the camera’s F-stops values are not known and therefore the algorithm cannot be applied. Alternatively, we can use observations of the ColorChecker directly to compute the gamma curve. At the centre of the ColorCheckers SG and DC there are a series of gray/white patches of known relative reflectance. Once the gamma curve is applied the relationship between these patches changes; knowing the original relationship once again allows us to calculate the gamma curve. However, there are a small number of patch observations to fit to, and so the results can be inaccurate.

Both identified methods may not be accurate in measuring the gamma curve, and so we must examine the effect any inaccuracy may cause on calibration. To test this we apply gamma correction to our ColorChecker image, but then remove it using a slightly different value. We then perform our calibration steps and compare the results to the ideal curves. We use a value of 2.2 as a base exponent for our initial correction, and then apply a perturbation to it of between -0.1 and 0.1.

Figure 3.10 shows the effect of a gamma correction measurement error on the accuracy of a camera calibration. We can see that even with a significant measurement error, that the effect is not

*The exposure is changed by varying the shutter time or opening the aperture.*
large. In general, the red channel is affected more than the blue and green channels.

### 3.3.4 Image Noise

![Graphs showing image noise comparison](image)

Figure 3.11: Comparison of calibration fit under noise

Image noise is an unavoidable side-effect in digital cameras, and so we must simulate its effect on our calibration procedure. To be able to examine the effect, we must first create a model that mimics digital noise in a camera. Modelling noise perfectly is very complex and beyond the scope of this thesis so we use a simpler, but reasonable approximation.

The model is a combination of two sources of noise: photon noise and dark voltage noise. The first type of noise, photon noise, occurs in the sensor due to the uneven distribution of photons that are incident upon it. Even when the flux is completely uniform across the sensor, the distribution of collected photons will be uneven. The second type of noise, dark voltage noise, is accumulated charge on the sensor where there is no incident light, it is also known as thermally dependant noise as it changes with ambient temperature. [FCW12]

We define the additive image noise model using the following equation [Sch03]:

$$n(x, y) = N(\mu_p, \sigma_p)\sqrt{i(x, y)} + N(\mu_t, \sigma_t),$$

where $i(x, y)$ is the original signal, $\mu_p$ and $\sigma_p$ are the mean and standard deviation of the photon noise, $\mu_t$ and $\sigma_t$ are the mean and standard deviation of the dark or thermally dependant noise. The noise distributions are zero mean-distributed and so $\mu_p$ and $\mu_t$ are both zero. For the test we vary the standard deviation of the thermally dependant noise from 0.01 to 0.045, and keep the standard deviation of the photon shot noise distribution to 0.01.

The noise model affects the image at the sensor level, and so we must simulate applying it at this level. To get the sensor image of the ColorChecker, we compute how the monochrome sensor image
would appear according to an RGGB Bayer matrix (see Section 3.2.2). We then apply the camera noise to monochrome sensor image before then de-Bayering the image to compute the noisy colour image. Once we have the noisy colour image, we proceed with the calibration procedure and compare with the ideal results.

We can see from Figure 3.11 that up to a standard deviation of 0.015 the fit is accurate. However, after this the fit becomes less and less accurate. This means that the camera can not exhibit overall noise of this level to be useful for our purposes.

3.4 Haemoglobin Analysis

We can now describe how to solve the inverse problem of mapping from the captured image of our subject to melanin/haemoglobin concentration. In previous subsections, we have described how to create an RGB skin lookup table particular to our capture device. This lookup table provides a mapping from a patch of skin with a given haemoglobin/melanin concentrations to its appearance in our device’s RGB space. We can use the inverse of this map to transform our image’s RGB pixel values to haemoglobin/melanin concentrations. We now describe how we perform this inversion.

Eq. 3.3 describes the RGB values generated in the skin lookup table, as well as the observed values in our image. In the case of our skin lookup table, the coefficient $d$, is constant throughout table, and governed by the constant of proportionality from our calibration (see Section 3.2.1). This means that both the generated skin lookup table and the observed values in our image will match in colour but not in intensity. As a consequence, to perform the inversion using the lookup table we must first normalise both the table and the image to remove differences in intensity.

3.4.1 Inversion Method

To solve the inverse problem we now have two steps to perform. The first step is to normalise both the skin lookup table and the observed image. The second step is to perform a scattered data interpolation in the normalised space to transform the observations into haemoglobin/melanin concentrations. In the rest of the section, we investigate the best methods to perform both the normalisation and interpolation. It should be noted that the normalisation to remove the surface shading reduces the degrees of freedom of the equation by one. This only leaves just enough degrees of freedom to solve for haemoglobin and melanin.

In general, the normalisation method eliminates the diffuse coefficient, $d$, from the observed vector equation in RGB space:

$$\vec{o} = d \vec{k}.$$ (3.13)

However, there are several alternatives of how to perform this normalisation.

Stating this in component form this becomes (where $\vec{o}_R, \vec{o}_G, \vec{o}_B$ and $\vec{k}_R, \vec{k}_G, \vec{k}_B$ represent the R,
We propose a second method for normalisation of the RGB vector. In this method we divide the observed RGB vector by the red component projecting into a unit sphere centred at the origin:

\[
\begin{pmatrix}
\hat{o}_R \\
\hat{o}_G \\
\hat{o}_B
\end{pmatrix} = d \begin{pmatrix}
\hat{k}_R \\
\hat{k}_G \\
\hat{k}_B
\end{pmatrix}.
\] (3.14)

The normalisation step is then applied to both image and lookup table to transform them both into a normalised space. When in this space we perform interpolation to map to melanin/haemoglobin space. For this we test a number of different methods to subsample the manifold. In the original Cotton and Claridge paper they use bilinear interpolation. We also test nearest neighbour and natural neighbour interpolation [Sib81].

Colour space normalisation

The first method used by Cotton and Claridge [CP03] divides the observed RGB vector by the red component projecting into BG/R colour space. This is due to the fact that reflectance from skin is highest within the red wavelengths, and therefore has the highest signal-to-noise ratio. After applying this we obtain:

\[
\begin{pmatrix}
\hat{o}_B / \hat{o}_R \\
\hat{o}_G / \hat{o}_R
\end{pmatrix} = \frac{d \cdot \hat{k}_B}{d \cdot \hat{k}_R} \begin{pmatrix}
\hat{k}_R \\
\hat{k}_G \\
\hat{k}_B
\end{pmatrix} = \begin{pmatrix}
\hat{k}_B / \hat{k}_R \\
\hat{k}_G / \hat{k}_R
\end{pmatrix} \in [0,1]^2.
\] (3.15)

Spherical projection normalisation

We propose a second method for normalisation of the RGB vector. In this method we divide the vector by its length. This length is a multiple of the diffuse coefficient, and so it is removed from the vector by division. This is equivalent to projecting in RGB onto a unit sphere centred at the origin:

\[
\frac{1}{\|\hat{o}\|} \begin{pmatrix}
\hat{o}_R \\
\hat{o}_G \\
\hat{o}_B
\end{pmatrix} = d \begin{pmatrix}
\hat{k}_R \\
\hat{k}_G \\
\hat{k}_B
\end{pmatrix} = \frac{1}{\|k\|} \begin{pmatrix}
\hat{k}_R \\
\hat{k}_G \\
\hat{k}_B
\end{pmatrix} \in S^2.
\] (3.16)

3.4.2 Evaluation

To test the alternative normalisation methods, we first generate a spectral skin reflectance profile using the Kubelka-Munk model (see Section 2.5.2) that is used by Cotton and Claridge. We generate a skin lookup table that is 256px × 256px for a range of typical melanin and haemoglobin values. We then generate a test image that uses the same skin model by supersampling melanin and haemoglobin space such that the image becomes 1024px × 1024px in size. This simulates matching pixels in an image that do not exactly match those in the skin table. We also quantise both the skin lookup table, and the test image to simulate an image taken by a camera. For a camera and light calibration we use a combination of a halogen light source and the spectral sensitivity of an AVT Guppy camera.

The relative error of the reconstructed haemoglobin values to the actual haemoglobin values for
Figure 3.12: Relative error of haemoglobin reconstruction using different interpolation and normalisation methods. Here, nearest/natural/bilinear represent the interpolation method, and spherical/cotton are the normalisation method.

Figure 3.13: Skin manifold in different spaces demonstrating metamerism. On the right we have highlighted the place where the metamerism occurs.

Each method is shown in Figure 3.12. The vertical axis is haemoglobin value, and the horizontal axis is the melanin value. The figures are blue where relative error is low, and red where relative error is high, corresponding to the axis at the side of the figure. We omit the combination of bilinear, and spherical projection, as the bilinear method requires monotonically increasing dependant variables, which spherical projection no longer provides. We can see that all normalisation methods perform roughly equivalently, with the combination of spherical projection and nearest neighbour interpolation showing slightly better results. As a result, we use this method throughout the thesis.

It is interesting to note that all normalisation methods demonstrate the same form of their inaccuracy. This region can be seen near the top of each of the subfigures. This inaccuracy is due to an effect known as metamerism. This occurs when projecting from a higher dimensional space to a lower dimensional one; in the case of our normalisation, we are projecting from three to two dimensions. Where the projection is non-injective, i.e. unique values in three dimensions map to
the same value in two, we describe the values as metamers. If metamers exists, then interpolation between scattered points becomes inaccurate as two distinct points in melanin and haemoglobin space become indistinguishable.

If we again look at the Figure 3.12 we see two distinct regions that can be identified. The first is where the melanin concentration increases, and the RGB reflectance becomes darker. These values are very close to zero, and so after normalisation project to the same point creating a metamer. This is also symptomatic of the fact that dark colours contain very little information and so there is very little to separate them even in RGB space. The second is a funnel shaped region in the top left of the figures. This inaccuracy is produced as a consequence of the shape of the skin colour manifold in RGB space. Figure 3.13b shows the skin colour manifold in RGB space, with the axes representing the different colour channels. The axes of Figure 3.13a correspond to the axes of Figure 3.12, and the colours correspond to those in Figure 3.13b. From this we can see which parts of the manifold correspond to those in Figure 3.12. Finally, Figure 3.13c shows an example manifold after normalisation, in this case dividing by the R channel. From these three figures, we can see where the metamerism occurs. The highlighted area in Figure 3.13c shows an area where the manifold folds over onto itself. The folding over means that after projection metamerism occurs.

Cotton and Claridge discuss this in their paper and state that the skin manifold does not exhibit this trait in LMS space. This space corresponds to the Long, Medium and Short wavelength sensitive cones in the human eye. However, we can clearly see that the problem occurs in our particular device’s RGB colour space. This can be rectified by a particular combination of camera and light source that avoids this problem. However, we opted to replicate their set-up, as we know it has been directly tested against biopsied tissue [MDM07]. This introduced the problem into our pipeline. However, later techniques and analysis contained in this thesis later allow us to generalise beyond their particular choice of equipment. It should be noted, though, that this problem exists and should be a factor if different equipment is to be used. In addition, our chosen normalisation method improves the accuracy of the funnel region, and high melanin concentrations are unsuitable for our experiments as they mask underlying haemoglobin measurements.

### 3.5 Camera evaluation

In the same vein as the evaluation of the camera parameters in Section 3.3, we now analyse the effect of camera properties on the quality of the obtained haemoglobin values. We perform these experiments using the Monte Carlo skin model, which is the model that we use in the latter part of this thesis. Again, we use the AVT Guppy sensitivity curves combined with the spectral power distribution of a halogen lamp to transform into RGB space.

In all tests, we use the skin lookup as a test image. We apply one of our test effects to it, and then try to reconstruct the original haemoglobin and melanin concentrations using the interpolation process. We then compute relative error between the reconstructed value and the original haemoglobin or
melanin value.

### 3.5.1 Bit depth reduction

![Comparison of relative error of reconstructed haemoglobin between 8-bit quantisation (left) and 16-bit quantisation (right).](image)

To test bit depth reduction accurately, we first reduce the bit depth of the test image, but also of the lookup table. In reducing the bit depth of the lookup table, we are transforming the lookup table to match the colour space of the camera which will also be quantised. Therefore natural neighbour lookup on exact matches will avoid any quantisation errors after both the normalisation and interpolation step.

Figure 3.14 shows the results of reducing the bit depth of the image. We can see that reducing the bit depth of the image to 8-bits reduces the accuracy significantly. This is probably due to metamerism and aliasing that is caused by the reduction. However, we can see that quantisation of 16-bits leads to negligible error, with noticeable error only in part where the reflectance of the skin is very low.

### 3.5.2 Compression

The compression test proceeds in the same way as we used for testing the calibration procedure, see Figure 3.3.2. The compression algorithm tested is for that of JPEG images, which also entails a reduction in the bit depth to 8-bit. The image format is common in most modern cameras and therefore is a sensible choice. As a consequence, we must directly compare to the results after reduction to 8-bits.

The results are shown in Figure 3.15 where the effect of the compression can be clearly seen from right to left. The figures show relative error where red is high error and blue is low relative error corresponding to the associated colourbar. An artefact of JPEG compression is to create blocks of colour, that are not apparent to a human observer but become very apparent when processed by machine. It can be seen that where these blocks exist, the reconstruction accuracy reduces dramatically.

### 3.5.3 Gamma Curve

Again the gamma curve test proceeds in a similar fashion to that for the spectral camera calibration, see Section 3.3.3. As a control, we do not perform any bit depth reduction on the image, or the lookup
3.5 Camera evaluation

Figure 3.15: Comparison of relative haemoglobin reconstruction error under different compression levels

Figure 3.16: Comparison of relative error of reconstructed haemoglobin due to error in the calibration of the gamme curve exponent
table. The results in Figure 3.16 show that even a small miscalibration in the exponent of the gamma coefficient leads to high source of error in the reconstruction.

### 3.5.4 Image Noise

![Image Noise](image-6.png)

Figure 3.17: Comparison of relative error of reconstructed haemoglobin due to noise levels

We use the same noise model that we used in the previous calibration section. In this case we keep the dark noise standard deviation at 0.0003, and allow the photon noise to vary between 0 and 0.01 in 8 steps. For comparison values we compute the signal-to-noise ratio of the sample image using the standard formulation of dividing the mean of the Bayer matrix signal, by the standard deviation of the noise image. We compute the base 10 logarithm of the result to convert to decibel units.

According to the ISO standard, for an image a SNR of 32 dB is an image of excellent quality and of 25 dB is of acceptable quality. We can see that even in the case where the signal-to-noise signifies an excellent quality image the accuracy of the reconstruction is still low.

### 3.6 Accuracy

As discussed, the skin reflectance spectra, which serves as the basis of the haemoglobin analysis method, is computed using a virtual model of skin structure. In Section 2.4, we describe an overview of different models that can be used. From these models, we use the Kubelka-Munk theory, and the Monte Carlo model.

For our initial findings, and later dynamic appearance model, we use the Kubelka-Munk theory. This model has been verified experimentally to match chromophore measurements from biopsied tissue [MDM07]. However, it contains a number of simplifying assumptions [Cho14] that create inaccuracies when simulating skin reflectance.

The Monte Carlo model simulates skin reflectance on the photon level. It is therefore, given the accuracy of the skin optical properties, the most accurate optical simulation that can be performed. We therefore use it for the later chapters to ensure we have the highest possible accuracy of skin reflectance modelling.
However, even though the accuracy is greater theoretically, we cannot guarantee that it is in practice. We know from Matts et. al. that analysis using the Kubelka-Munk model is correlated with chromophore concentrations in biopsied tissue. To show that the Monte-Carlo method works in a similar way, we perform an experiment. We use the RGB skin lookup map generated using the Kubelka-Munk theory, and analyse it using the Monte-Carlo method. We then compute the correlation between haemoglobin values of both models to show that there is a relationship.

Figure 3.18: Plot showing haemoglobin values obtained from Monte Carlo analysis of Kubelka-Munk theory lookup table.

Figure 3.18 shows haemoglobin values with the Kubelka-Munk theory against those obtained using the Monte Carlo model in the analysis. It can be clearly seen, and the correlation co-efficient is calculated as 0.97, which indicates a very strong correlation. There are two features to note about the graph; first, there is a non-linear relationship between haemoglobin values in the Kubelka-Munk model, and those obtained from the Monte Carlo analysis; second, there is a clear non-injective relationship between the two values. In the former case, the haemoglobin values for each model are in different units; Kubelka-Munk uses Molar concentration, and the Monte Carlo model uses volume fraction; these two quantities are related non-linearly. In the latter case, the non-injective fanning effect is caused by the fact we are projecting away from the melanin concentration dimension; any haemoglobin values with different melanin values in Kubelka-Munk should therefore map to the same haemoglobin value in the Monte Carlo model. An intuition about this can be found be studying
the surfaces analysed in Section 3.4.1.

3.7 Initial Findings

Empirical analysis of the haemoglobin maps lead to a number of initial observations. These findings motivate the initial design of our model to describe the changes in haemoglobin concentration. In this section we demonstrate how these findings manifest themselves.

1. **Visible blood is mainly contained within the ascending and descending capillaries originating from the relatively superficial sub-papillary plexus**

   ![Figure 3.19: Facial haemoglobin distribution of a 33-year-old Caucasian female and of a 26-year-old Caucasian male. Dark pixels denote high concentration. On the right, we see the highly vascular nature of the blood perfusion](image)

   As described in Section 2.5, blood is concentrated within multiple layers of the skin’s structure. However, it only becomes visible when concentrated in the ascending and descending capillaries extending from the sub-papillary plexus. This is due, in principal, to the fact that visible light cannot penetrate the deeper optically turbid layers AP81. This results in a highly vascular appearance to blood perfusion. On the right of Figure 3.19 we see vascular nature in and around the nose. It is clear that there is a pattern of capillaries and veins.

2. **The mechanical deformation of facial expressions may lead not only to drainage of blood in compressed regions, but also to a perfusion increase in other regions.**

   In Figure 3.20 we can observe both the effects of draining and perfusion in separate areas. In particular if we look at the forehead of the subject, we observe both drainage and perfusion simultaneously. At points where the wrinkle forms a valley blood is pushed away, and the concentration reduces. Conversely, where there is a peak the blood is pushed together, and the concentration increases.

3. **While the qualitative changes connected with a single expression correspond well across subjects, there are large differences in each individual’s spatial pattern of perfusion.**

   Figure 3.21 shows that between subjects there are common qualitative changes. In both images, due to the change in the facial expression the cheeks have an increase in haemoglobin concentration. However, we can see that overall the pattern of haemoglobin concentration is different between subjects.
3.7. Initial Findings

(a) Subject in neutral pose  (b) Subject in surprise pose

Figure 3.20: Changes in haemoglobin concentration for subject when he changes his pose

Figure 3.21: Comparison of haemoglobin changes when a subject has a happy expression.

4. Conversely, this subject-specific pattern appears to be very static and only subject to decrease or increase of local blood concentration
Figure 3.22: Time multiplexed signals

5. Global effects cannot simply be understood as a scalar rise across the face.

Figure 3.22 (top row) shows the temporal perfusion variations as a result of high exercise level (after descending and climbing 9 flights of a staircase; subsequent measurements for one minute) and under alcohol consumption (500 ml beer, spread over one hour; regular measurements during this period). These measurements were taken at the five points shown in Figure 3.22a. It shows a distinctly different perfusion increase at different points on the face.

To study facial perfusion at a time scale less than the physical limits of the camera we employ a time slicing technique. Using propriety software we timed the total capture time to be 5.5s. This included the time for camera triggering and retrieving the image from the camera.

The subject was asked to alternate between the neutral and smile expressions in time with a periodic audio signal. The camera was then setup to take an image using a separate periodic signal that was out of phase with the audible signal. The out-of-phase signals can then be re-ordered and used to reconstruct a signal with a resolution of the difference between the two times. This was performed at periodic differences of 0.5s and 0.25s.

The captures were 22 and 44 captures (each capture consisting of both a smile and a neutral expression) in total for the 0.5s and 0.25s respectively. This gives us a total duration of our multiplexed signal of 22s, from a total capture time of just over 4 minutes.

The reconstructed signals shown in Figure 3.22 demonstrate that the haemoglobin values are
3.8. Summary

In this chapter we have described both the method and physical construction required to capture haemoglobin concentrations of our subject’s expression. The method is based on analysing the diffuse colour of our subject’s skin. To this end our construction consists of replicating the original SIA SCOPE design, which incorporates polarisation to remove specular reflection, and use black felt to minimise spillage of unpolarised light.

The original SIA SCOPE paper does not describe the calibration method required to transform consistent across the same facial expressions i.e. repeating the same facial expression does not gradually increase the amount of blood flow over time. It also demonstrates that any transitional phase between expressions is less than 0.25s. This leads to the final observation that:

6. Dynamic effects in the transition of expression-induced perfusion changes are of lesser visual importance and may hence be ignored in a practical model.

Figure 3.23: Our time multiplexing scheme. Top: We sample the expressions at varying time offsets from the expression transition. Bottom: The images were taken in column-major order (as numbered), with a short phase shift between acquisition and expression changes. This gives a consistent 5-image temporal sequence of four expressions (neutral/smile/neutral/smile) when arranged in row-major order.
from the spectral skin reflectance models to a camera’s RGB space. We incorporate the calibration method from Finlayson and Hubel which uses a MacBeth ColorChecker to derive a combined camera spectral sensitivity and light source spectral power distribution. Due to the non-uniform nature of the light sources being used we incorporate a novel method for adjusting for the variation of light intensity over the ColorChecker.

To test the accuracy of the calibration method, we ran a series of experiments by virtual modulating a set of common camera properties. These properties include the image bit depth, its compression, gamma curve calibration and noise. These properties vary across consumer cameras that could be used for haemoglobin/melanin analysis. Our tests show that even though bit depth reduction does not have an effect on the calibration process; compression, gamma curve correction and noise do. It is therefore shown that the calibration is most effective on a camera capable of capturing RAW linear images.

Following the calibration step we can generate an RGB skin lookup table that allows us to solve an inverse problem and derive haemoglobin and melanin concentrations from a photograph of our subject. The solving of this inverse problem requires two steps; normalisation and interpolation. Cotton and Claridge use a particular method of normalising into BG/R colour space, and interpolating using bilinear interpolation. We observed that all methods lead to metamerism given a particular light source and camera combination. However, we explore several methods of normalisation and interpolation to try to mitigate this. Although we found that nearest neighbour performed the best in and around the metameric area; it produced artefacts in other areas of the haemoglobin/melanin space. We therefore pick a compromise of a spherical normalisation and natural neighbour interpolation.

To further investigate the effect of camera selection, we tested the accuracy of haemoglobin reconstruction under different camera parameters. These parameters are the same as those used to test the limits of the calibration. Here we strengthen our conclusion that any capture device must be capable of capturing linear RAW images. In addition, we find that bit depth reduction has a significant effect on the accuracy of reconstructed haemoglobin.

Finally, using our completed setup, we capture some exploratory images of haemoglobin concentrations of subjects in different poses. From these poses we derive a set of fundamental assumptions that will be used to derive our dynamic model. In the next chapter, we describe this model.
Chapter 4

Color Appearance Model

In this chapter, we develop a practical model that allows us to compute haemoglobin concentrations based on dynamic changes in facial expression. From our initial findings, in the previous chapter, we know that any changes to blood concentration levels are contained within the upper vascular network of the skin (see initial findings 1 and 4 from the previous chapter Section 3.7). This means that effects below the surface of the skin do not produce visible appearance changes. Using this we can reduce the model to one that modulates a neutral haemoglobin map to produce one associated with a specific expression.

In Section 2.1 we explored the Ekman and Plutchik theories of expression. In Plutchik’s work, they define a wheel of emotion where all facial expressions lie within the wheel formed by a set of fundamental expressions circumjacent to it. All expressions within the wheel are then defined as a blend of these fundamental expressions. In this thesis, we use the original work of Ekman with a set of six expressions, but use the wheel theory to justify a blending algorithm. The six emotional expressions, lie around the edge of the wheel, and a neutral expression sits at the centre. We disregard the additional emotions of Plutchik, Trust and Anticipation, as these have no obvious analogue facial expressions. Extreme emotions like anger or fear trigger very strong vascular dilation or constriction, resulting in significant blushing of pallor. These effects are difficult to measure experimentally, however we can easily simulate them by scaling the global haemoglobin with respect to the neutral pose. In our examples the anger and fear poses were scaled by 127% and 81% respectively.

For each one of the fundamental expressions, we capture a haemoglobin map of our subject. To find a haemoglobin map for a novel expression, we first describe its position in the wheel. This position determines the proportion of blending of each of the fundamental expressions required to form the novel haemoglobin map. The initial findings from the previous chapter show us that the change in haemoglobin concentrations can be modelled by simply modulating the neutral haemoglobin map. This is because the shape of the underlying vascular network does not change, only the level of blood concentration at each point in the network. As a result the preservation of fine detail in the

1 There is a debate over which expressions complete the definition of the wheel.
neutral haemoglobin map is essential. Matusik et al. [MZD05] have shown local histogram matching provides accurate blending between textures which preserve sharp details in the images. We therefore opt to use local histogram matching as the basis of our blending technique.

In the next section, we describe our novel blending algorithm, and some simple optimisations. We then go on to describe some practical considerations when using the technique with a standard rendering pipeline.

4.1 Blending algorithm

As described above, local histogram matching forms the basis of our blending algorithm. This technique maps a pixel value in a source image to a value in the destination using the statistics in the neighbourhood around that pixel. More specifically, we generate histograms from the values in corresponding neighbourhoods in both the source and destination images. We then identify the percentile of the source pixel in the source distribution. The destination value then becomes the pixel value from the destination distribution with the same percentile. The mapping of percentiles is performed by using the local Cumulative Distribution Function (CDF) around the pixel.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure4_1.png}
\caption{Histogram Mapping between two CDFs}
\end{figure}

To perform histogram matching between points in our haemoglobin map, we map between their CDFs as shown in Figure 4.1. For corresponding neighbourhoods, we compute CDFs for the source and destination image, $F_S$ and $F_D$. We then compute a new destination pixel value, $p_D$, using:

$$p_D = F_D^{-1}(F_S(p_S)).$$

(4.1)

However, the wheel of expressions is bounded by our five facial expression. We therefore need to expand the histogram technique to blend between multiple haemoglobin maps. Matusik et al. demonstrate how to do this by forming a new inverse CDF $F_N^{-1}$, as a weighted sum of the CDFs for the $j$ expressions, $F_j^{-1}$. The weightings are determined by the position of the expression in wheel of expressions:

$$F_N^{-1}(p) = \sum_j F_j^{-1}(p).$$

(4.2)
4.1. Blending algorithm

Figure 4.2: A sample of different distributions in a radius around each point on the left, with fitted Gaussian curve. The color coded points, left, correspond to the distribution diagrams, right; namely forehead, nose, upper cheek, jaw and chin. (The histogram on the tip of the nose is not shown)

4.1.1 Gaussian Approximation

The histogram technique quickly becomes inefficient as we must store and invert a CDF for every point in a haemoglobin map. By observing the histogram at key points on the face, Figure 4.2 (the forehead, the nose, the upper cheek, the jaw, and the chin), we found that for most surface points the histogram of blood concentrations was approximately Gaussian, (see Figure 4.2). It is this observation that allows us to simplify the histogram matching technique. Rather than computing a CDF per pixel, we can approximate the distribution of haemoglobin values in a local neighbourhood as a Gaussian. Where the distribution is not an exact fit for a Gaussian, the dynamic blood concentration values will have an increased or decreased contrast between local levels where the standard deviation is over-estimated or under-estimated respectively. Where the true mean of the distribution is inaccurately represented, it will lead to a higher or lower overall blood concentration locally, as the local mean is shifted up and down. We can then use the equations for the Gaussian CDF and its inverse to perform the histogram matching.

For a Gaussian of mean \( \mu \), and standard deviation \( \sigma \), we have CDF and inverse CDF:

\[
\Phi(x, \mu, \sigma) = \frac{1}{2} \left[ 1 + \text{erf} \left( \frac{x - \mu}{\sigma \sqrt{2}} \right) \right],
\]

\[
\Phi^{-1}(x, \mu, \sigma) = \sqrt{2}\sigma \left( \text{erf}^{-1}(2x - 1) + \mu \right), x \in (0, 1),
\]
where \( \text{erf}(x) \) is the Error Function and with its inverse is defined as:

\[
\text{erf}(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^{x} e^{-t^2} \, dt,
\]

(4.5)

\[
\text{erf}^{-1}(x) = \sum_{k=0}^{\infty} \frac{c_k}{2^k + 1} \left( \frac{\sqrt{\pi}}{2} \right)^{2k+1}, x \in (-1, 1)
\]

(4.6)

To complete the model, we could simply store the pre-computed values for the \( \text{erf}(x) \) and use values of standard deviation and mean computed for each pixel in our image. However, in Section 4.1.3, we describe a function to reduce the blending algorithm to a scale and bias function. In the next section, we first describe how to compute the mean and standard deviation of the Gaussian distributions at each point in the image.

### 4.1.2 Computing standard deviation

To fit our Gaussian distribution, we need to compute the standard deviation and mean of the haemoglobin values in the neighbourhood surrounding each pixel in our image. Given the set of neighbourhood pixels \( N \) surrounding a central pixel \( p \), we can compute the mean \( \langle H \rangle \) of their corresponding haemoglobin values using the following summation:

\[
\langle H \rangle = \frac{1}{|N|} \sum_{n \in N} H(n),
\]

(4.7)

and for the standard deviation \( \sigma(H) \) as follows:

\[
\sigma(p) = \sqrt{\langle H^2 \rangle - \langle H \rangle^2},
\]

(4.8)

where \( H^2 \) represents the square haemoglobin image.

The computation of the standard derivation in Eq. (4.8) requires the use of a mean operator on both the haemoglobin texture and its square. We can directly compute the mean using Eq. (4.7), or equivalently we can convolve the image with an averaging kernel. This comprises a two-dimensional matrix of values with a one for every pixel contained within the neighbourhood, and zero everywhere else. The ones are then divided by the size of the neighbourhood, so that after convolution the result is an average rather than a summation. We use a disc shaped neighbourhood for the kernel as this is, intuitively, more conducive to natural perfusion. The convolution kernel can now be used to compute the mean and standard deviation:

\[
\mu(p) = (H * \kappa)(p),
\]

(4.9)

\[
\sigma(p) = \sqrt{(H^2 * \kappa)(p) - \mu(p)^2},
\]

(4.10)

where \( \kappa \) is the kernel. To increase the performance, it can be shown that using Discrete Fast Fourier
Transformations can accelerate a 2D convolution significantly [Pod07].

4.1.3 Optimisation

Once we have computed our per-pixel Gaussian parameters, we perform histogram matching between the source and destination as in Eq. 4.1. However, as we have fitted Gaussians we can perform a further optimisation to the calculation. Assuming that both CDFs are continuous and monotonic we can reduce the blending algorithm to a simple scale and bias operation. More formally, given our neutral (source) and destination expression respective Gaussian CDFs, \( \Phi_n \), and \( \Phi \), we substitute them into Eq. 4.1 to get:

\[
H(p) = \Phi^{-1}(p; \mu, \sigma) \left[ \Phi_n(p; \mu_n, \sigma_n) \right].
\] (4.11)

Using the definitions in Eq. 4.3 and Eq. 4.4 we can reduce it to:

\[
H(p) = \mu(p) + \frac{\sigma(p)}{\sigma_n(p)} (H_n(p) - \mu_n(p)),
\] (4.12)

4.2 Model Parameters

![Image showing different maps used by the algorithm](image)

**Figure 4.3:** Different maps used by our algorithm. The haemoglobin scale and bias maps shown here belong to the smile expression (Haemoglobin Scale shown at 90x scale for visualisation purposes).

Using Eq. 4.12 we extend the blending function to encompass our entire wheel of expression, rather from neutral to a single expression. We denote the set of haemoglobin maps, one for each expression \( H_i \forall i = 1 \ldots k \), and the neutral expression haemoglobin map \( H_n \). From this we define a mean-free base concentration:

\[
H_0 = H_n - \mu_n,
\] (4.13)

and a set of scales and biases:

\[
\text{bias}_0 = \mu_n, \quad \text{bias}_i = \mu_i, \quad i > 0, \\
\text{scale}_0 = 1, \quad \text{scale}_i = \frac{\sigma_i}{\mu_n},
\] (4.14)

where the argument "\((p)" has been removed for readability.
Using these quantities we can transform from the neutral haemoglobin map to any blend of the fundamental expression maps, $H_i$:

$$H = \sum_{i=1}^{k} w_i \text{bias}_i + H_0 \sum_{i=1}^{k} w_i \text{scale}_i$$

(4.15)

where $w_i$ and the relative weights sum to one. We can leverage this representation for storage efficiency, as the low-frequency nature of the biases and scale can be exploited and we explore this in the next section. Following existing terminology in geometric animation, we refer to the set of $w_i$ as parameters of an appearance rig.

To capture suitable haemoglobin maps for the model, we must follow a series of processing steps. First, we capture the haemoglobin maps of the subject using the setup described in Chapter 3. We then manually warp each of the haemoglobin maps to align them with the photograph of the neutral expression. Due to the complex nature of expression changes, it is impossible to achieve pixel-accurate alignment of the maps. However, as we are using a radial neighbourhood around each pixel to compute local statistics, we can still achieve good results.

The non-local support of the histograms $^2$ implicitly leads to a separation of low-frequency and high-frequency changes (as opposed to frequency-based separation of global and local changes $^{[GS99]}$) resulting in two distinct advantages. Firstly, we can apply relative changes captured from one subject to another subject’s neutral map, or even a hand-painted map. Secondly, as the scale and bias textures store low-frequency information (see Figure 4.3), we can downsample them to reduce their memory footprint and not lose significant amounts of information.

### 4.3 Summary

In this chapter, we presented a dynamic haemoglobin colour appearance model using the assumptions discovered in the previous chapter. The wheel of expression, taken from psychological literature, is circumscribed by a series of fundamental expressions, with a neutral expression at its centre. The assumptions in the previous chapter lead us to the conclusion that the dynamic haemoglobin model can be reduced to a modulation of this central neutral expression.

We first present a model that allows us to calculate the amount of modulation by blending between the neutral expression and a destination expression. This blending algorithm uses local histogram matching, where we transfer information about neighbourhood statistics between corresponding pixels in both images. We then extend this blending to be able to blend between multiple expressions simultaneously. The blending weights are determined by our novel expression’s position in the wheel.

To make the model suitable for real-time rendering algorithms, we make a couple of adjustments. We first observe that in the local neighbourhood around each point the distribution of haemoglobin

$^2$Here non-local support means the histogram statistics are based on neighbouring pixels around which the histogram is centred
values is approximately Gaussian. By leveraging the similarity of Gaussians we reduce the transformation to a simple scale and bias. This allows us to pre-compute a set of scale and bias textures, that can be easily loaded into memory and used efficiently.

In the next chapter, we demonstrate how the model can be used in an animation pipeline linked to a real-time animation rig.
Chapter 4. Color Appearance Model
Chapter 5

Real-Time Colour Animation Rig

In the previous chapter, we described a dynamic haemoglobin concentration model based on a histogram matching blending algorithm. In this chapter, we describe how the model can be driven by a geometry rig for use in animation pipeline.

There are a number of different geometric animation approaches that either globally or locally control geometry [War05]. Of the most common approaches, such as blend shapes, bones, or a combination; most depend on a use of local or global weights to drive the model. The linear model from the previous chapter can be easily adapted to these existing techniques. Although there are no intrinsic limitations to any of the approaches, our method is particularly appropriate for global blend shapes [DCFN06] used commonly in production environments [RCB05].

5.1 Implementation

The real-time facial colour animation rig consists of four components (in execution order):

5.1.1 Facial shape interpolation
To compute the facial shape interpolation from a set of $k$ expressions, we blend between meshes that were created by using a barycentric coordinate system. Simply, a weighted sum of all the pose vertex positions $V_i$ with the sum of the weights normalised to 1:

$$V = \sum_{i=0}^{k} \frac{w_i}{\sum_{i=0}^{k} w_i} V_i.$$ (5.1)

5.1.2 Haemoglobin change computation
Following from this, we compute the haemoglobin change texture using our scale and bias textures from the previous chapter:

$$H = \sum_{i=0}^{k} w_i \text{bias}_i + H_0 \sum_{i=0}^{k} w_i \text{scale}_i.$$ (5.2)

The blend shape animation is computed as efficiently as possible by using graphics hardware capabilities and streamed blend shapes using DirectX® 10. This avoids the limitation that only
four blend shapes can be packed into per-vertex attributes simultaneously. A set of transformed vertices is stored in a buffer, allowing an unlimited number of blend shapes to be used using multiple passes [Lor07]. Wrinkles are rendered using Jimenez et al. [JE01], which is based on masking wrinkle zones and efficiently adding the influence of multiple normals combined from different zones using partial derivative normal maps.

5.1.3 Base skin colour computation
A skin colour map is precomputed using the methods described in Chapter 3. For the simulation of subsurface scattering, we use the method by Jimenez et al. [JS09, JSB10]. We perform the sum-of-Gaussians texture-space diffusion [dLE07] as a post-process in screen space, by modifying the width of the convolution kernel according to the depth gradient information. This scales well with geometric detail, but still retains high visual fidelity. Lateral colour bleeding as a result of subsurface scattering is explicitly captured as part of the acquisition process, so we only use the post-scattering texture [dLE07] to avoid blurring the albedo twice. The screen-space approximation depends on pre-scattering blur, so we separate the diffuse illumination, albedo, and specular reflectance in different render targets. We then selectively apply subsurface scattering only to the diffuse component, and finally composite the components together into a final image.

The parameter maps obtained from the acquisition process are blurred due to subsurface scattering in the skin (see Chapter 4). This acts as a further optimisation to the rendering process; where previously the melanin and haemoglobin values in the map controlled the subsurface blurring effect [DDw08], here we only need to use them to look up a skin colour in our precomputed lookup table.

5.1.4 Simulating skin subsurface scattering
During rendering, we perform a standard subsurface scattering simulation using parameters for colourless skin. The result is modulated using the colour from the lookup table using the melanin and haemoglobin values at each point. Finally, we use Kelemen and Szirmay-Kalos [KSK01] for our specular model with spatially-varying parameters for roughness and specular intensity [WMP06].

5.2 Results and Conclusion
Figure 5.1 demonstrates the results of the real-time skin appearance rig. The images use the haemoglobin maps of a 26-year old male subject. The figure shows six emotional states described by Ekman [EFO71], and the wide variation in skin colour for each expression can be clearly seen. Only some of the colour change is caused by deformation, and some is caused by capillary dilation.

Figure 5.3 shows two additional physiological states, exercise and alcohol consumption, applied to a neutral pose. The results appear subtle, but still increase realism and convey emotions, and still free animators from tweaking of skin parameters. Figure 5.4 shows a close-up of neutral and disgust poses which shows the difference as a result of our model. The approach is therefore general
5.2. Results and Conclusion

With Our Model

Without Our Model

Anger Disgust Fear Sadness Smile Surprise

Figure 5.1: Rendered emotional states. The top row uses our blend shape method. The bottom row simply has the neutral haemoglobin map for comparison.

Figure 5.2: Example of blending poses: from left to right, the character transitions from the neutral pose to full anger (third image), to a combination of full anger and full surprise (fifth image). The last image adds the changes after physical exercise.

enough that any expression, emotion or state may be added, allowing for combinations of poses (see Figure 5.2).

The results demonstrate an efficient, real-time skin appearance rig with dynamic skin colour changes. Previous chapters have described how to capture haemoglobin maps, and a model that can drive dynamic haemoglobin changes for novel expressions. However, the model still requires capturing haemoglobin textures for each one of the expressions, and therefore still has an overhead in creating the model. Beginning in the next chapter, and for the rest of the thesis, we explore a

(a) Neutral  (b) Exercise  (c) Alcohol

Figure 5.3: The neutral geometric pose of a face rendered with a neutral haemoglobin map (left), and predicted haemoglobin perfusion after exercise (middle) and after alcohol consumption (right).
Figure 5.4: Close-up of the neutral and the disgust poses. The changes in hemoglobin distribution are generated by our model.

generic model for simulating haemoglobin changes that relies purely on metrics describing geometric changes in facial expression.
Chapter 6

Physical Construction

In the previous chapter, we proposed a model for animating the effects of a changing facial expression on the skin’s blood concentration. This model utilizes blood concentration maps captured by analysing static photographs of a subject with different facial expressions. These expressions are defined in psychology literature as a basis for the gamut of all facial expressions. Utilizing this set of basis expressions allows the animator to create novel expressions by using them as blend shapes, and simultaneously in our data-driven model. Novel expressions are created by using a histogram matching technique to interpolate between the haemoglobin maps for each expression. This data-driven approach requires the artist to capture haemoglobin concentration maps for each
of the basis expressions. However, a professional blend-shape rig require a large amount of blend shapes, which make acquisition very difficult. By studying the correlation between geometric changes and haemoglobin changes the hope is to eliminate the need for such costly captures, and addition extending the applicability of dynamic haemoglobin changes to non blend-shape rigs.

In the next two chapters, we expand upon our work discussed so far to avoid the need to build specialist equipment. In our extended model we posit that there exists a functional relationship between the mean and standard deviation statistics, and a geometric measure of the change in facial expression. This measure is computed from a three-dimensional facial mesh representing the expression. In this way an artist can animate a mesh and the required changes in mean and standard deviation statistics are automatically computed.

To compile these data sets required to build a general model, we must first construct a hardware and software pipeline that can capture both geometry and blood concentration maps simultaneously. This capture device will record video footage of changing facial expressions and perform multi-view stereo and haemoglobin analysis on a per-frame basis. In this chapter, we describe the design decision required to physically construct the capture device. In the next chapter, we describe the software pipeline required to produce three-dimensional meshes. Our haemoglobin reconstruction pipeline remains the same as described in previous chapters.

The physical construction of our capture set-up therefore has to marry up two different aims. First the physical set-up must be capable of removing specular highlights for capturing haemoglobin concentration. At the same time we must reconstruct the 3D geometry of subject. The resulting physical set-up along with the reconstruction pipeline is unique in its design and is a major contribution of this thesis.

A diagram of the full physical construction is shown in Figure 6.1. It is composed of several distinct parts; multi-view cameras, polarisation filter, custom made capture hardware and software, halogen light sources, curtains and aluminium frame. The multi-view cameras allow us to reconstruct the geometry of the subject’s facial expression. To allow sufficient illumination suitable for haemoglobin reconstruction, we choose high-powered halogen lamps. These lamps’ smooth spectrum allows an accurate calibration as described in Chapter 3. Similarly, as shown in that chapter, we remove specularly highlights that mask underlying haemoglobin concentrations, by the use of different polarisation filters over the light and cameras. To stop any unpolarised light from reaching the cameras we make custom light blocking curtains. These curtains are hung from an aluminium frame that is suspended from the ceiling.

6.1 Camera configuration

When constructing a stereo reconstruction set-up there are two decision that we have to make. The first decision is the configuration of the cameras to allow complete and accurate capture of the subject’s expression. The second decision is the camera specification and the lens specification.
Figure 6.2: Comparison of reconstruction results for the Manta G-201C, Manta 504C, and Pike 505C. (From top to bottom) Original, Haemoglobin, Melanin, Diffuse Albedo Reconstruction, Shading (Irradiance), Reconstruction.
For geometry capture we use a multi-view stereo capture configuration. This gives us several advantages over binocular stereo. For our purposes we need to collect geometric information describing the face in its entirety. This includes reconstructing geometry from both the sides and the front of the face. For a binocular stereo system, it is very difficult, if not impossible, to set-up two cameras such that their viewpoints cover enough of the face but still share enough image space to be able to find matching points in both images.

From the experiments presented in Section 3.3 we know that the camera must have a certain set of properties, specifically; linear gamma curve, uncompressed RAW output, and low sensor noise. We sampled a number of such cameras to evaluate a selection of lenses and camera resolutions. The three cameras tested were: the Manta F-201C which is a two megapixel camera capable of a resolution of 1624 by 1234; the Manta 504C capable of 2452 by 2056; the Pike 505C (which shares a sensor with the more expensive Prosilica GT2750) capable of 2448 by 2050.

All three cameras are capable of capturing linear RAW images with data transferred directly from the sensor. For the higher-resolution cameras the pixel size is much smaller than for the two megapixel cameras which can increase noise. A comparison of the reconstruction results for all three cameras can be seen in Figure 6.2. We can see that the results are comparable, and the increase in noise has little effect on the reconstruction accuracy. We therefore go for higher resolution images and compensate for noise.

With our experiments we have also shown that a higher bit depth is preferable. Both higher-resolution cameras have a bit depth of 14-bit. This is at the high end of what is possible with cameras of this specification and was deemed sufficient for our purposes.

Finally, to be able to create a dynamic haemoglobin perfusion model that is a function of changing geometry, we must be able to capture high frame rate video footage for tracking. Given the throughput of data required for high-resolution and high bit depth, we have found that the maximal frame rate obtainable from the selection of cameras is 15 fps. Although not ideal this again should be sufficient for our purposes. To be able to transfer the large quantities of data, both high resolution cameras transfer the data over Gigabit Ethernet.

### 6.2 Capture Software and Hardware

The capture software and hardware have to have sufficient capacity to record all four cameras simultaneously. The cameras of choice are the Prosilica GT2750C, which are capable of capturing a resolution of 2448 by 2050 at a bit depth of 14 bits at a rate of 15fps. The data is transferred via Ethernet to the machine through a four port switch. For a single camera, this equates to a data transfer rate of:

\[
\text{width} \times \text{height} \times \text{bit depth} \times \text{fps} = 2448 \times 2050 \times 14 \times 15 = 131.73 \text{MBps}.
\]
For four cameras, this becomes 526 MBps transferred over Ethernet to a four port switch connected to the machine. To be able to store this bandwidth of data we purchased an Intel SSDSC2CW120A3K5 SSD capable of performing sequential writes of 520 MBps. It is usual for the reported speed of hardware manufacturers to only be theoretically possible. To compensate for this we installed 32GB of RAM in the system. This RAM can be used as a buffer to store recorded data rapidly, and then later recorded to the SSD.

On top of the custom hardware configuration, we created custom multi-threaded software that is able to multiplex all the streams together. Each of the cameras is captured using a separate thread which loads the data directly in RAM to enable maximum throughput. As the RAM buffer is filled, it is written to disk by a separate thread in a rolling fashion. The capture machine is fitted with an eight-core CPU to be able to run the multi-threaded software with one core per thread.

For the operating system, the machine was installed with a Linux distribution allowing several benefits to the capture process. It first allows all sub-systems, including the graphics system, to be shut down to minimize any external loads on the CPU. In addition, the capture software can be elevated to the highest possible thread priority meaning that only kernel based operations can interrupt the capture.

The final part of the capture hardware is the camera control circuit, constructed from an Arduino micro-controller. The controller was programmed to emit a clock pulse to all the cameras via camera I/O controller. The full details of the required implementation are contained within the camera’s manual [AVT14].

### 6.3 Polarisation

![Circular Polariser](image)

Figure 6.3: Construction of a circular polariser as a composite of a linear polariser angled at 45°, and a quarter-wave retarder.

In Chapter [3] we used linear polarisers to remove specular reflection from the surface of the skin. For maximum efficacy linear polarisation require that the light source and camera lie in the same plane defined by the axis of polarisation. This is because polarisation is preserved by specular reflection, and any movement outside of the plane constitutes a relative rotation with respect to the original polarisation direction (see Appendix [A]). In constructing a multi-view stereo system it
becomes impossible to keep both light sources and cameras in the same plane (see Figure 6.1) and so linear polarisation techniques become less effective. To circumvent this problem, we instead use circular polarisation.

We construct a circular polariser from the composition of a linear polariser and a half-wave retarder. The linear polariser first creates a 45° polarised wave with equal components in both the s and t components in the plane of oscillation. When the polarised wave passes through the half-wave retarder one component of polarisation becomes out of phase with the other by a half wavelength. When the components are out of phase, the oscillations describe a circle in the plane of oscillation. When incident upon a specular reflector, the direction of circular oscillation becomes reversed, i.e. left-handed rotation to right-handed rotation. Therefore, to filter out the circularly polarised light we must place a filter of the same handedness over the camera lens.

For linear polarisers, when cross polarisation is used to remove specularity, their efficiency depends on their relative rotation. At 90° relative rotation all specular reflectance is filtered out, at 0° relative rotation none is filtered. As a component of our circular polarisers is essentially a linear polariser, it also exhibits some rotational properties. As the main reason for using circular polarisers is to avoid this rotational component, we must measure it to ensure that our results are accurate.

### 6.3.1 Evaluation

![Graphs showing angular and spectral efficiency of polariser](image)

(a) Angular dependence of polarisation efficiency  
(b) The spectral efficiency of polarising filter efficiency

Figure 6.4: Overall and spectral efficiency of the polariser as a function of rotation.

We evaluate the efficiency of the polariser using a spectrometer and a halogen light source (see Section 6.4). We place a circular polarisation filter over the light source, such that the radiated light source is circularly polarised. We then bounce this light off a mirror to reverse its chirality, and pass it through a circular polariser placed over the spectrometer probe. We use blackout curtain material to ensure that the only light entering the probe has been first reflected off the mirror. We then rotate the polariser over the probe and measure how the spectrum changes as it is rotated.

Figure 6.4a plots the average polarisation efficiency with reference to the angular rotation of the probe filter. We first compute the percentage of un-filtered light by taking the ratio of the captured
6.4 Light selection

The light sources in the capture set-up (see Figure 6.1) must satisfy two different requirements. Firstly, the light source must be suitable for haemoglobin analysis. Secondly, the light source must be bright enough to enable video capture. For this purpose, we decide between a white LED and a pair of 1000w halogen lights.

6.4.1 Smooth Spectrum

Figure 6.5 shows the difference in the spectra between the halogen lamps and the LED light source. It can be seen clearly that the LED light source suffers from a large spike near the blue end of the spectrum by the halogen light spectrum without a filter. We then take an average of this over the visible spectrum and plot this against rotation angle, we also show the best fit cosine curve. We use a cosine curve, as the amount of cross-polarised light that is transmitted through a polarisation filter is based on the cosine law. This law stems from the projection of the light’s polarisation vector, and the polarisation axis of the filter. We can see that the efficiency of the polarisers ranges from approximately 6% to 11% of specularly reflected light un-filtered. However, specular reflectance accounts for approximately only 4 to 7% of total skin reflectance [IN07, DJ06]. That means that in total maximally only 0.77% of the reflected light is error due to specular reflectance. Figure 6.4b shows that the polarisation filter is very efficient in the visible range (400nm to 700nm). The worst efficiency of the polariser is in the 750nm to 800nm range and therefore does not affect our measurements.

The calibration step (see Section 3.2.3) in our haemoglobin analysis method requires the computation of a set of combined light source and camera sensitivity curves from an image of a colour checker. As part of the calibration process we solve a set of linear equations involving the patch spectra of the color checker and the light source spectrum. To regularize this solution, we enforce the assumption that the required spectra are smooth. As a result the accuracy of the solution is dependant on the smoothness of the light source spectrum.

Figure 6.5 shows the difference in the spectra between the halogen lamps and the LED light source. It can be seen clearly that the LED light source suffers from a large spike near the blue end of the spectrum.
the spectrum. The calibration step revolves around solving a set of linear equations. These discrete number of linear equations naturally produce a quantisation of the frequency space. Therefore any thin spikes naturally inhibit the accuracy of the solution. In addition, the calibration step smooths the spectrum to be able to avoid a degenerate solution to the linear system. A spike will hence be difficult to resolve in this circumstance.

### 6.4.2 Brightness

The use of polarisers greatly reduces the amount of light that passes into the camera lens. Light from the source first passes through the circular polariser in front of it reducing its intensity by a half. Of this, depolarised light that is reflected diffusely by skin then enters the cameras lens through an additional polarizing filter. The overall effect is to reduce the amount of light entering the lens by a quarter, or 4 f-stops. The intensity of the light source is further reduced by the area of the circular polariser through which the light initially passes.

Therefore the light source must emit a high intensity of light. For this purpose, we use two industrial-strength halogen lamps utilising 1000W halogen bulbs. Each bulb outputs a total of 21,000 lumens totalling 42,000 lumens in total. For comparison, a standard 13W LED light bulb produces approximately 900 lumens, thereby requiring approximately 40 LED light bulbs to produce the same brightness [Lig, Ear].

### 6.4.3 Evaluation

Given the requirements of a smooth spectrum and high brightness we chose the 1000w halogen lamps. Due to the nature of the halogen lamps, we are presented with two sources of error for our calibration and also for the quality of our reconstruction. The first is the stability of the halogen lamp over time which affects the accuracy before, after and during any capture. The second is the heat generated by the halogen lamps increasing the noise on the camera sensor.

![Figure 6.6: Stability of the halogen spectrum over time](image)

**Stability of halogen lamp** The light output of a halogen lamp is dependent on heating a tungsten filament to high temperatures. The spectral output of the heated filament approximates a black body
6.4. Light selection

The stability of the spectral curve calibration is two-fold: first, the lamp will take a certain time to heat up to full operating temperature; second, the lamp will change temperature over a prolonged period of time. If there is sufficient deviation from the calibration and over the duration the capture the results could be adversely affected.

To test the stability of the halogen lamp we make spectral measurements of the lamp over the period of one minute. We measure the spectral output of the lamp using a spectrometer which has an attached fibre optic probe. The spectrometer is set to capture 100 samples of the 60 seconds from start to finish.

Figure 6.6 shows the results of the curve measurements superimposed on each other over time. By inspection, we can see that the curve is very stable, with a marginal deviation from the average. We can therefore surmise that the calibration will be applicable across the capture process.

Heat problems

The light efficiency of halogen bulbs is only 3%, which means that approximately 97% of the energy input to the bulb is radiated as heat. This is because the choice of a black-body radiator as a smooth spectral light source is always associated with significant heat production. In fact, the maximal efficiency of a black-body radiator is 14% [Kee07].

The heat production causes significant problems for safety and the comfort of our subjects. To help with heat flow, we leave the back of the physical set-up open to allow the heat to escape. This also protects the subject from any blood flow changes in their skin as a result of heating. The curtain in front of the subject reduces the amount of dissipation towards them, and restricts the flow to only away from the subject.

In addition, it is noted that the possible detrimental effect on the polarizing filters is small. A linear polarizing filter (one of the components of a circular one) creates polarisation by only allowing electron flow along the length of polymers within the material [Lan51]. However, as the temperature rises for polyvinyl alcohol (a typical polarisation material) the chain confirmation increases [Wis12]. This means that temperature does not decrease the efficacy of the polarisers up to approximately 200°C. As such, the effect of temperature in our case is negligible.

![Figure 6.7: Noise levels over time as the temperature on the image sensor increases. We capture a series of frames of increasing temperature (left). However, we can see that the standard deviation stays constant over time (right).](image-url)
Increasing heat also increases the amount of dark current noise on the camera sensors [PKD’08]. To measure this effect we capture a series of dark noise images over time with the lens cap on. The cameras are placed under the halogen bulbs to simulate the hot environment. Figure 6.7 shows the temperature of the sensor over the duration of the test demonstrating a clear increase. For each of the frames we compute the standard deviation of the signal over the image a measurement of noise. The results clearly show that even though the temperature increases the standard deviation stays roughly constant. Consequently to compensate for the noise, we adjust for a dark image that is the average of the dark images over the entire measurement.

### 6.5 Curtain and aluminium frame

The aluminium frame was constructed from standard Rexroth aluminium parts [Rex]. These parts allow a great deal of flexibility in their configuration with connectors allowing them to be attached at right angles to each other. The frame holding the curtains was suspended from the ceiling by inserting drilling into and inserting expanding bolts into the concrete.

The polarisers were pinned to a wooden frame with a hole in the middle to allow light through. The frame was held at a height in line with the halogen lamps to allow the maximum amount of light through by an aluminium frame. The cameras were attached to same frame using wooden back plates. These backplates provided two functions: the first was to ensure that the cameras were aligned in the same plane reducing the error of calibration; the second was the allow the cameras to be rotationally adjusted to allow the maximal facial area of the subject to be captured.

To stop any uncontrolled light leakage within the set-up we utilize curtains hung from the frame. The curtains must therefore have the properties of highly absorbing and not reflect light specularly which may produce highly polarised light. For this purpose, we could not find a suitable material for the curtains that was within budget. We therefore constructed our own curtains by sewing together different materials with complimentary properties. The first material presented a dense highly absorbing layer, but exhibited some specular reflective properties. The second material was a black semi transparent material but exhibited no specular reflectivity. In addition, the denser material was fire retardant and therefore present an extra safety measure for the high temperature halogen lamps. To hang the curtains from the frame we bought runners that fit inside the groove within the aluminium struts. The runners allowed the curtains to be adjusted and for the subjects and experimentalist to move between sections of the construction.

### 6.6 Summary

In this chapter we have discussed the process of the physical construction required for the combined geometry and haemoglobin capture device. The construction consists of several different parts; cameras, polarisation filters, custom capture hardware and software, halogen light sources, curtains and an aluminium frame.
6.6. Summary

To select the cameras we utilize our analysis from Chapter 3 describing the cameras that produce the highest accuracy of calibration and haemoglobin reconstruction. Within these constraints we select three candidate cameras with a variety of lenses and resolutions. Although a higher resolution camera can have higher shot noise, on sample data it performs as well as a camera with lower resolution. We therefore opted to go for the highest possible resolution camera the Prosilica GT2750C.

To cope with the data output of four cameras in a multi-view stereo set-up we build a custom computer and create custom software. Even the fastest type of SSD is incapable of recording data at the throughput level\(^1\). We therefore install a high amount on memory in the machine to act as a buffer before writing it to disk. Finally, to synchronize the cameras we create a simple control circuit that produces a digital pulse at a fixed period.

In the first capture set-up in Chapter 3 we used linear polarisers to remove specular reflection for our subject’s skin. However, due to the multi-view stereo set-up this is no longer sufficient. Due to budgetary restriction purchasing pre-made circular polarisers became prohibitive. To get around this problem we construct our own polarisers out of a linear polariser, and a quarter-wave plate retarder. Experiments show that the results are highly successful and produce good results even over the range of rotational variance.

To capture a properly illuminated subject through both sets of polarizing filter we require a high powered light source. On top of this, because of the calibration step in our haemoglobin reconstruction pipeline, we require a smooth spectrum source. To satisfy both these requirements we use a high-powered industrial halogen light source. However, this does come with several drawbacks namely the stability of the spectrum and the heat that it generates. We measure the stability over time and find that the halogen lamp is remarkably stable and so presents no problem to the accuracy of reconstruction. The heat generation poses problems for our subject’s comfort and safety, and noise produced on the cameras’ sensors. We solve the first problem by adapting our set-up to allow heat to flow away. For the second, we show that the dark image noise does not vary significantly with temperature.

Lastly, we use a strong aluminium frame drilled into the ceiling to suspend the equipment and curtains from. The curtains are custom made to stop unpolarised light from leaking from the outside, and to stop polarised light from depolarizing as a result of diffuse reflectance.

\(^1\)At the time of construction
Chapter 7

Geometry Processing

In the previous chapter we described the physical construction of our capture setup that allows us to capture both haemoglobin concentration maps and geometry simultaneously. In this chapter we describe the geometry processing pipeline that takes the images from the multi-view camera setup and produces facial meshes.

The first step in the pipeline is to prepare the images for stereo reconstruction. This step consists of first removing the blue channel from the image to reduce noise, and second to transform the images from all the cameras into the same colour space.

The second step is to calibrate the extrinsic and intrinsic parameters of the camera for the purposes of stereo reconstruction. To do this we use standard calibration techniques from literature. However, due to the polarisers in our setup the light levels reaching the camera are very low. This means that the camera’s aperture is wide open, significantly reducing the depth of field. To compensate for this we use a custom target that deals sufficiently with depth of field blurring. The final steps are
the mesh reconstruction steps, which we take from previous research.

7.1 Image preparation

To allow us to perform geometry reconstruction we must first prepare our images to maximize the accuracy of our reconstruction. The first step is to transform all our images into the same colour space. Typically, to transfer between RGB spaces it is via the XYZ colour space [Pae03]. To compute the transform from the camera RGB space to XYZ space we use the calibration data capture during the haemoglobin extraction stage (see Section 3.2.3). We then use a transform from XYZ to the Best RGB colour space.

To compute the transform from XYZ to RGB space we use the fact that for each patch, $P^n$, in our calibration target we have a known XYZ coordinate, $P^n_{XYZ}$, from manufacturer’s data, and a measured RGB coordinate, $P^n_{RGB}$. We then solve a system of equations to find the linear transform, $T$, between the two spaces:

$$
\begin{bmatrix}
  P^1_X & P^n_X \\
  P^1_Y & P^n_Y \\
  P^1_Z & P^n_Z \\
\end{bmatrix} = T \cdot
\begin{bmatrix}
  P^1_R & P^n_R \\
  P^1_G & P^n_G \\
  P^1_B & P^n_B \\
\end{bmatrix}.
$$

(7.1)

For the rest of reconstruction pipeline we only use two of the three colour channels. As a result of the particular choice of halogen light (see Figure 6.6) there is very low light level in the blue portion of the spectrum. This means that the signal to noise ratio in the blue channel is very high, as shown in Figure 7.2. We therefore only use the red and green channels for any reconstruction and calibration steps below.
7.1. Image preparation

7.1.1 Calibration

In this section we describe the steps taken to reconstruct the point cloud of our subject’s expression from our multi-view captures. To be able to reconstruct point clouds from multi-view stereo images we must first perform calibration the cameras. Once we have the calibration we perform a dense patch-based reconstruction step using Furukawa and Ponce [FP10a].

Next we describe the camera calibration steps where a brief description of the intrinsic and extrinsic camera parameters described in Appendix B. We use standard calibration techniques to obtain the parameters.

Intrinsic Calibration

To measure the intrinsic parameters of the system, we use the MATLAB toolbox provided by Bouguet [Bou04]. This is combined with the calibration pattern provided by Vogiatzis and Hernandez [VH10], see Figure 7.3, which provides more accurate detection and calibration in a wider set of circumstances than the original chequerboard pattern. The calibration of the intrinsic parameters allows us to remove lens distortion from any captured images, and to normalize the focal length in pixels across the different cameras. Here we use the manufacturer data denoting the cell size of the photosites on the sensor which is 3.45\(\mu m\), and the lens focal length of 25mm to give 7246 pixels. We remove the distortion and normalizing the focal length using the OpenCV library [BK08].

Extrinsic Calibration

To calibrate for the extrinsic parameters we use the VisualSFM pipeline created by Wu [Wu11]. Typically the VisualSFM software is used to both calibrate intrinsic and extrinsic parameters, and perform dense stereo reconstruction all using the same set of images. However, we found that the Bouguet method is more accurate. We therefore fix the intrinsic parameters to match those of the corrected images and run the extrinsic calibration.
For our purposes the choice of calibration target is key to the accuracy of the extrinsic calibration. The sparse point cloud reconstruction step relies on SIFT feature \cite{Low99} detection and subsequent matching between views. The use of polarisers, reduced illumination, and the requirement of video footage, meaning fast shutter times, reduces the illumination of the subject significantly. This means that the aperture settings of the cameras must be set to as wide as possible. Under these circumstances the depth of field is greatly reduced. To compensate for this we adjust the focal plane for every subject at the start of every capture. However, due to the position of the cameras, it became impossible to have the calibration target fully in focus for all views. It was therefore important to choose a calibration target (see Figure 7.4) with a large amount of SIFT features leading to a large amount of matches eventually. After the cameras are refocused the intrinsic parameters will change, however, as noted we have kept them fixed for the extrinsics. To compensate for this we perform an additional bundle adjustment step using the sparse point cloud generated by VisualSFM.

The calibration step is of great importance to the overall reconstruction process. A lot of time was spent optimizing the calibration process by tweaking the VisualSFM configuration files. For our purposes, with our particular calibration target, we found that maximizing the number of features passed to the GPU was important. On average the SIFT GPU library identified between 40000 to 90000 features points per calibration image. To drive the SIFT feature detection requires setting various parameters. We found the most important of these for calibration accuracy were the matching distance and the maximum matching distance ratio. The first parameter controls the threshold for rejecting a SIFT feature match based on its distance. The second controls a feature match based on the ratio of its distance to that of the second-best match in the pool of matches.

We tweaked the parameters such that there was a relatively high distance threshold, but a close to unity ratio between the matches and the second-best match. This means that in the case of a blurred image, we allow a margin of error for the points, but utilize our large pool of features to pick only the best of those available. Typically, the parameters would reduce the total amount of matching features over the images to between 4000 and 6000. We then perform an additional bundle adjustment step with a fixed focal length to ensure the accuracy of the calibration. The adjustment of the parameters was perform for each subject to obtain the best results.

### 7.2 Point Cloud Reconstruction

To perform the reconstruction of the point cloud we use the Patch-based multi-view stereo (PMVS) reconstruction software \cite{FP16,FP10a}. The software recreates a dense point cloud reconstruction from a set of multi-view stereo images coupled with an extrinsic calibration. The algorithm works by matching a patch from one camera image to the other camera images. This is performed by computing a photometric consistency measure between the original patch, and its re-projection in the other camera views’ image spaces.

The choice of patch size becomes important for the quality of reconstruction. The patch
approximates the surface as a plane, and so a large patch assumes that the surface is very smooth and
neglects sharp detail. For a small patch the error increases as there are less sample points, but allows
for finer detail. For this part of the pipeline, we set the software to use a patch size of 19, a fairly large
patch. Later in the pipeline we perform a mesh refinement step to extract finer detail. We set a very
high rejection threshold, so that only the very closest matches are accepted. This helps us overcome
any noise problems associated with the low exposure values.

For the point cloud reconstruction we use only the red and green channel images described
above. The PMVS software was designed to only read and use 8-bit images. However, one of the
great advantages of the camera choice is the high bit depth. We therefore modified the software to
utilize 16-bit images. In addition, we produce a set of masks, one for each of the four camera images,
that segment the subject’s skin from the rest of the images. Serendipitously, the separation of skin
tones from the image is eased by using only the red and green channels\footnote{especially in contrast to
the black background and white hairnet. In these images, we only need to restrict the hue channel
to perform the segregation. The range of hue values was decided for each subject upon inspection,
as the value varied greatly between subjects. The segmentation naturally produces a mask with a
lot of unnecessary speckles. To clean this up, we first select the largest connected component in the
mask, which should be the skin area. We then smooth the boundary of the mask by performing image
closing and the opening morphological operations. After this we use a hole filling algorithm to fill in
parts of the mask that may have been removed from the centre of the skin area, such as the eyes.

Initially, we performed some simple point cloud cleaning steps to improve the output of PMVS.
However, we found that the use of high bit depth images, image masks and our calibration techniques
meant that this was no longer required. An example of reconstructed point cloud can be seen in
Figure 7.5b.
7.3 Mesh Reconstruction

The initial mesh reconstruction is performed using Poisson reconstruction [KBH06] on the point cloud. This reconstruction step is used throughout Computer Graphics and is very successful. It works by fitting an implicit function to the point cloud such that its distance spatially and in respect to the point cloud normals is minimised. Such a fitting obeys the Poisson equation, where the normals are related to the gradient of vertex position function. For the normals, we allow the software to recompute the normals from the point cloud directly. At this point in the pipeline the Poisson reconstruction only needs to be an approximation to the true surface as we perform a mesh refinement step. However, we found in most cases that the reconstruction was very accurate, due to the accuracy of the point cloud reconstruction. For a particular subject, the results of the Poisson reconstruction can be seen in Figure 7.5c.

7.3.1 Mesh refining steps

The mesh refinement pipeline is based on a pipeline described in Beeler et al. [BBB+10]. Their pipeline is a state-of-the-art reconstruction pipeline. We use parts of the pipeline that are not primarily their research, however their paper collates the steps into a reproducible form. As the software is not open-source we reimplement it here and as such provide a brief overview of the algorithmic steps. The refinement is based on a set of papers ranging from Scharstein and Szeliski [SS98] to [WTRF09]. As the Poisson reconstruction results from the previous section are very accurate, and the point clouds of high quality, we only employ the surface refinement steps for the paper.

Surface Refinement

The surface refinement algorithm works by maximizing the photometric consistency of the mesh with the camera images. To compute photometric consistency a patch based algorithm is used. For every vertex, we find the camera which is most face on with the vertex’s normal. That is the camera whose view vector to a vertex is most aligned with its normal. A square patch is formed around the vertex’s projection in that camera’s image, where the patch is of size \( n \). This patch is then projected into the other camera views and the photometric consistency measured between the original and the patch’s projection.

The measure of photometric consistency used by Scharstein and Szeliski [SS98] is Normalized cross correlation (NCC). For two patches \( p_1 \) and \( p_2 \) where each one consists of a set of projected pixels \( p_1(u,v) \) and \( p_2(u,v) \) it is computed as:

\[
\text{NCC} = \frac{1}{n} \sum_{u,v} \frac{(p_1(u,v) - \mu_{p_1})(p_2(u,v) - \mu_{p_2})}{\sigma_{p_1} \sigma_{p_2}},
\]

where \( \mu_{p_i} \) and \( \sigma_{p_i} \) are the mean and standard deviation of the patch in one image, and analogously

---

This is roughly the color of the dominant chromophores in blood; red for haemoglobin, and brown (a mixture of red and green) for melanin.
for patch two. Note, here $u$ and $v$ represent corresponding samples in the patches of size $n$. The value of the NCC is turned into an error value, $\xi$, where $\xi = 0.5(1 - \text{NCC})$.

This photometric consistency is used to perturb the surface along its normal direction. The maximal perturbation distance is defined as $\delta$. For each vertex we measure the photometric consistency at its original point, $-\delta$ along the normal, and $\delta$ along the normal obtaining $\xi_{-1}$, $\xi_{0}$ and $\xi_{+1}$ respectively. Using this an optimal value for the perturbation, $\delta_p$, is found by interpolation between $-d$ and $d$ quadratically:

$$\delta_p = \begin{cases} 
-\delta & \xi_{-1} < \xi_{0}, \xi_{+1}, \\
\delta \frac{\xi_{-1} - \xi_{0}}{\xi_{+1} - \xi_{-1}} & \xi_{0} < \xi_{-1}, \xi_{+1}, \\
\delta & \xi_{-1} < \xi_{0}, \xi_{+1}.
\end{cases} \quad (7.3)$$

This new vertex position is then computed from the original position, $X$, as a weighted combination of a smoothed vertex position, $X_s$ and the perturbed position, $X_p$, where $X_p = X + d_p$:

$$X' = \frac{(w_p X_p + w_s X_s)}{(w_p + w_s)}, \quad (7.4)$$

where the smoothness weight $w_s$ is fixed, and the perturbation weight $w_p$ is calculated:

$$w_p = \begin{cases} 
\xi_{-1} - \xi_{0} & \xi_{-1} < \xi_{0}, \xi_{+1}, \\
0.5(\xi_{-1} + \xi_{+1} - 2\xi_{0}) & \xi_{0} < \xi_{-1}, \xi_{+1}, \\
\xi_{+1} - \xi_{0} & \xi_{-1} < \xi_{0}, \xi_{+1}.
\end{cases} \quad (7.5)$$

For the purposes of smoothing the mesh we use mean curvature flow. Desbrun et al. [DMSB99] and Meyer et al. [MDSB03] describe the computation of mean curvature flow for a triangulated mesh. The mean curvature flow is movement along the normal in proportion to the negative mean curvature of the surface. This means that the movement acts to move the mean curvature towards zero, which is a property of minimal surfaces. To compute it we use an open source implementation of Desbrun’s algorithm [rp10]. The movement computed by mean curvature flow requires an implicit integration, and so we manually create a small time period over which to integrate.

To automate the process further we add an additional step to halt the mesh refining process. The delta movement of the algorithm is fixed to a maximum of 0.1 mm, when the average movement change over the entire surface is less than this threshold we stop the optimisation.

### 7.4 Results

For our purposes, the capture setup is also required to reconstruct haemoglobin concentration maps. As we have seen previously the use of polarisers greatly reduces the exposure. This causes problems in the calibration, but also in the geometry reconstruction as shown in Figure 7.6 We can see that
where the image has a low exposure, the geometry becomes especially noisy. To compensate for this we could increasing smoothing in this area. However, as the geometry is used to later drive a model we must preserve accuracy as much as possible. For this purpose we must accept that where there is excessive noise in parts of the image the true geometry is not obtainable. This means that parts where the accuracy of the geometry is very high the results are not over smoothed.

In addition, the results of the capture can be affected by a mis-calibration in the extrinsic parameters of the cameras. This results in a ridge forming between two mismatched camera views.
7.5 Summary

In this chapter, we described the mesh reconstruction pipeline. The pipeline consists of four steps; calibration, point cloud reconstruction, mesh reconstruction and mesh refinement. All fours of these steps are based on established techniques from previous research and are not individually novel. However, we had to make adjustments to the pipelines to be able to adapt it to our particular measurement setup.

The adaptions are required because of the combination of polarisers and our halogen light source. The use of polarisers greatly reduces the amount of light that illuminates our subject, and the halogen source means that there is very little light to start with in parts of the spectrum. The low light level means that there is an increased noise level in all the images. To mitigate this we increase the aperture size to its maximal settings. The effect of this is two fold; first, there is a greatly reduced depth of field in the images; secondly, there is a higher propensity for motion blur.

To cope with the increased noise levels we perform a number of preparation steps. The first adjustment is the introduction of a specialised noise model \(^2\) that allows us to cope with temperature changes on the sensor. After we have reduced the noise in the Bayer pattern, we transform all the images into a common colour space. In this space, we also remove the blue channel as its noise level is visibly very high.

To cope with the reduced depth of field, we use a specially chosen calibration target. The very low depth of field meant that even focussing on the calibration target in all of the views became very difficult. Despite this, the calibration method provided high accuracy results, and after parameters were tweaked became a one-click solution for most of the captured subjects.

\(^2\)See Appendix C
Chapter 8

Metric Computation

In the previous chapters, we have described how to measure facial geometry with corresponding haemoglobin concentrations. This is achieved using our unique, combined haemoglobin concentration and Multiview Stereo (MVS) capture system. Capturing both simultaneously allows us to test the thesis’ overarching hypothesis; that there is a correlation between the changes in facial geometry and changes in haemoglobin concentration. In this chapter, we describe how to compute a series of metrics that are phenomenologically correlated to haemoglobin concentration changes. In the next chapter, we then analyse the correlation between the metrics and changes in haemoglobin concentration.

In the first section of this chapter, we describe the motivation for several geometry-based metrics. This choice is driven by phenomenological observations from different subjects. We observe that curvature and surface divergence exhibit correlation with haemoglobin changes. We then use the rest of the chapter to describe how to compute these metrics.

In the second section, we describe the preparation steps for our captured images. We first project
the geometry and measured haemoglobin values into a common image space. This space is defined by a cylindrical mapping of the subject’s facial geometry into a $uv$ space. We use this mapping to project the haemoglobin and geometry into a two-dimensional image for each of the cameras. We then use **Unstructured Lumigraph Rendering (ULR)** to blend between the different camera views to create a single combined image. Finally, to be able to measure changes in the metrics over time, we first use a tracking algorithm to find correspondences between frames. We then use image warping techniques to align the images with each other.

In the third section, we describe how to compute the metrics directly from the geometry image [GGH02]. It is simply the surface geometry mapped into an image, where the RGB colour channels map to XYZ coordinates. The embedding of the geometry in an image does not detract from the fact that it is still a set of vertices, and so we can use standard techniques to compute curvature and divergence over the cloud. We imply a smooth enough surface such that we can use standard image processing kernels as differential surface operators.

In the case of curvature we can pull the algorithm directly from previous research. In the case of temporal divergence, however, the situation is different. Divergence acts on a vector field defined by the movement in geometry over time. In our case the vector field is defined over the set of pixel coordinates, rather than a triangulated surface. We therefore compute the divergence directly in image space, rather than perform additional triangulation steps.

### 8.1 Observations

In this section, we describe empirical observations made from the captured data. These observations determine the set of metrics that we compute namely curvature and divergence.

#### 8.1.1 Curvature

We can observe sharp changes of curvature in the skin around points of wrinkling. At a wrinkle point, muscle movements compress skin together create wrinkling in the surface. This wrinkling forms a wave-like pattern where peaks bulge out from the surface with troughs in between. The wavelength of the wrinkle pattern is approximately related to the Youngs modulus and thickness of the skin [CM03]. Therefore thicker tissue forms longer wavelengths, and thinner tissue shorter wavelengths. Generally, the troughs in the wrinkling pattern have a much sharper curvature than the peaks, and the wavelength represents the average of the two. This higher gradient results in a much higher curvature value for the troughs, and a lower curvature value for the peaks.

As can be observed in Figure 8.2 and Figure 8.3, there are blood concentration changes associated with wrinkling. Here the figures show two sets of photographs showing a subject moving from a neutral to an emotional expression; the set on the left shows the original colour photographs of the change, and the set on the right shows blood concentration changes. The blood concentrations are

---

1 Here, wavelength is a property of the waveform present in the curved profile of the wrinkle.
Figure 8.2: Subject showing wrinkles changing curvature on facial expression. Color images (left), Haemoglobin Concentration (right)
Figure 8.3: Subject showing curvature changes in cheek on facial expression. Color images (left), Haemoglobin Concentration (right)
high where the image is light, and low where the image is dark. At points where the skin wrinkles inwards we can see that the blood concentration drops as the blood is pushed away by the inward compression. Surrounding the wrinkles, we can observe an increase in the blood concentration as the blood is pushed away. The formation of the wrinkle troughs are instances of increasing negative curvature, and the wrinkle peaks are instances of positive curvature. We can therefore posit that changes in curvature caused by wrinkling are correlated to changes in blood concentration with negative curvature increases causing a reduction in blood concentration, and positive causing an increase.

8.1.2 Divergence

In Figure 8.4 we observe the correlation between surface contraction and blood concentration changes. In particular, we note the changes on the subject’s forehead; we can see that at the points the skin’s surface converges, the blood concentration levels rise; at the point they diverge, the levels drop. This is independent of any curvature changes, which is restricted where the forehead’s skin layer is tightly coupled to the skull.

8.2 Image preparation

In this section, we describe the steps that will take our multiview images and transform them into a single image per-frame. This allows us to track corresponding points between frames to investigate the temporal nature of the changes. We first transform each of the viewpoint image into a shared coordinate system. This allows us to blend the images together into a single image per-frame of our capture. We then apply an algorithm by Sundaram et al. [SBK10] to perform the tracking. The algorithm incorporates assumptions that mean the tracker performs most accurately when structures are smoothly moving between frames, and have largely consistent size and shape. To ensure this we take several precautions to enable the most accurate and dense tracking possible, which we discuss.

8.2.1 UV Mapping

UV coordinate mapping is the process of creating a two-dimensional mapping of a three-dimensional surface. Typically it is used in computer graphics for the purposes of texture mapping. However, we use it here to provide a common coordinate space onto which to combine our multiview images into a single image per frame in our capture. The image based rendering is performed using ULR described in the next subsection.

Surface parametrisation, such as UV mapping, is a well studied topic [SPR06]. Techniques such as ABF++ [SLMB05] try to preserve angles during the mapping process, and some try to preserve both angles and area [LPRM02], whereas some look to preserve distances [SSGH01]. However, although the use of these techniques can be advantageous for texture mapping it is detrimental in our case: the tracking algorithm is highly dependent on consistent size and shape between frames. For our pipeline, we cannot guarantee a consistent triangulation between frames, and so these algorithms
Figure 8.4: Shrinking in skin shows reduction in blood concentration. Color images (left), Haemoglobin Concentration (right)
8.2. Image preparation

To enable a consistent mapping, we choose cylindrical projection. A human face provides an approximate homeomorphism to a hemi-cylinder, making this type of projection a natural choice. Some points, such as under the nose, do not provide an injective mapping under a cylindrical map as they lie on approximately the same horizontal plane. However, such points in general are not important to our analysis as they tend to show very little blood perfusion.

We now provide a definition of the cylindrical mapping, by first defining our mesh, $M_n$, which is composed of a set of triangles, $T$, and a set of vertices, $V$ for a frame $F_n$. Each vertex $v \in \mathbb{R}^3$ is a tuple:

$$v = (v_x, v_y, v_z).$$  \hfill (8.1)

To produce its UV coordinates, $(u, v)$, using the cylindrical mapping we use:

$$u = v_y,$$ \hfill (8.2)

$$v = \frac{1}{2\pi} \arctan \left( \frac{v_x}{v_z} \right).$$ \hfill (8.3)

The assumptions of the tracking algorithm also mean we have to keep the subject’s head as still as possible between frames. Even though we have a chin rest to stop the subject’s face making large movements left and right, it is inevitable that there will be some small movements. However, the chin rest does stop the subject moving in and out of the shallow depth of field. To compensate for the movements we perform a series of alignment steps on our meshes, $M_n$. Our first step is to align all meshes, $M_n$, to the neutral, or zeroth mesh $M_0$. We compute this alignment using ICP [BM92], resulting in a transformation, $T_n$ between the two frames. We then compute the transformation that aligns the neutral frame to the axes such that the subject is facing outwards on the z-axis.

We therefore have two transformations; one to align each of the frames with the neutral expression coordinate frame, and then from the neutral coordinate frame to the axis-aligned frame. By composing these two transformations for each frame we can transform the geometry from its original coordinate frame to an axis aligned one. More formally, for each vertex $v$ that is an element of our
vertices $V$ of our mesh $M_n$, we get a new set of axis-aligned vertices $v' \in V'$:

$$v' = T_N T_n v$$  \hspace{1cm} (8.4)

This alignment is performed manually using the mesh view GUI provided as part of the TriMesh [Rus] package as shown in Figure 8.6. The alignment is only necessary for the initial, neutral expression.

The alignment process above ensures that movements by our subject are greatly reduced. However, even small movements side-to-side by our subject can be magnified by our cylindrical UV mapping. Figure 8.5b shows that this magnification effect becomes greater the closer we are to the centre of projection. However, on the other hand, as a result the size of projection increases, meaning the resolution is enhanced the closer we are to the centre. We therefore find a balance on how far forward to move the meshes in the $z$ axis away from the origin.

### 8.2.2 View-dependent combination

To produce the blended image, we must compute the UV coordinates for every pixel associated with our four cameras views (see previous chapter), which we perform using GLSL shaders. The first step is to project the the $xyz$ coordinates of the geometry into the coordinate system of the original images. This is achieved using the camera calibration used in the last chapter, and standard OpenGL projection. The result is a geometry image where there are $xyz$ coordinates in place of the $RGB$ channels of our original captured images. We then apply the equations, Eq. 8.2 and Eq. 8.3, to transform each pixel into the UV coordinates. These images then provide a mapping from pixels in each one of our camera views to the corresponding points in UV space.

The final step is to blend the four different UV mapped camera views to create a single image containing the full geometry. Each pixel in the UV space is potentially shared by more than one
camera view that project to the same point in UV space. To create a single image, we must blend the different overlapping pixels together. To do this we use ULR interpolation, introduced by Buehler et al. [BBM*01], but we use the simplified version by Weyrich [WMP*06]. The algorithm described computes a blending weight for the images based on a set of relative weights, $r_i$. The maximal number of images blended is set to some $k$, in our case 3, where $i < k$. For the relative weights $r_i$ we use the dot product of the surface normal at the pixel with the viewing direction. Therefore a normal facing the direction of the camera has the highest weight and so the value which is most face on the camera has the most influence. To accentuate the effect we apply a gamma correction to this value to disproportionately weight those images with the highest weighting.

We use this UV projection and blending method to produce three images for use in our later processing pipeline; haemoglobin ($H$), melanin ($M$), irradiance ($I$) and geometry ($G$).

### 8.2.3 Tracking and warping

To measure metrics over time on the surface of a subject’s skin, we require the ability to track those points. To get tracking information we use the dense optical flow algorithm by Sundaram et al. [SBK10]. The algorithm is GPU-accelerated and produces results in parallel. Figure 8.8 shows the results of the tracking over a capture sequence. The density of points is reduced for visualisation purposes in the figure, but the algorithm produces a per pixel tracking.
Once we have the tracking points we use them to warp the images into alignment such that they match the first image in the capture. To perform the warping we use the inbuilt MATLAB transformation functions to construct a composable transform that warps between a frame and its predecessor. To compute the transform from a frame $n$ to the first frame, we then compose all the transformations $n, n-1, n-2$ to 1.

### 8.3 Metric Computation

In the previous section, we described how to produce a set of UV-coordinate- projected images from our warped meshes so that they are aligned with each other. Apart from the per-frame melanin, haemoglobin and irradiance images, we also produce a per-frame geometry image $G_n$. In the geometry image, the R, G and B channels contain the X, Y and Z coordinates of our geometry respectively. All of our metrics are computed directly on the geometry image, allowing the same metrics to be computed in GPU shaders for use in animation pipelines. Once we have a series of geometry images at different spatial frequency we compute our metrics over each spatial frequency in turn.

All metrics are based on computing derivatives in $uv$ space over the geometry image. To compute the derivatives we filter the images with $du$ and $dv$ kernels that use the finite differencing methods.

#### 8.3.1 Scale Space Decomposition

Before we compute the metrics on the geometry we first decompose it in frequency space by convolving the surface of the geometry with a series of Gaussian kernels, see Figure 8.10. This Gaussian space decomposition is a common technique used in Computer Graphics; we therefore follow the method of Mokhtarian et al. [MKYO]. However, where they perform the convolution on the mesh directly, we perform it in image space. The Gaussian must be applied to the surface of the geometry, and so we must first project it into the tangential plane at each point.

The two-dimensional Gaussian kernel is a square kernel with equal semi-major and semi-minor axis. When the kernel is projected onto the surface of the geometry in image space, the effect is to
8.3. Metric Computation

The blurring effect of the low-pass nature of the Gaussian kernel is determined by its standard deviation. The higher the standard deviation the lower the frequencies that are allowed to pass, and the smoother the geometry will appear. However, as the standard deviation increases so does the fall-off of the kernel and therefore the computation time goes up linearly with each standard deviation. To circumvent this, we apply the same Gaussian filter repeatedly. For a filter with standard deviation, $\sigma$, if we convolve with the same kernel again it it equivalent to filtering with a Gaussian kernel of standard deviation, $\sqrt{2}\sigma$.

8.3.2 Image Space Derivatives

These derivatives compute the change of the R,G, and B channels of our image wrt. a differential change in our image coordinate system $(u,v)$. These differential operators are applied using convolution.

The most basic convolution kernel to use is the central difference kernel. However, its use can lead to blocky artefacts as a result. To mitigate against this we use Gaussian smoothing. This involves convolving the image first with a Gaussian filter in the direction orthogonal to the central difference kernel. The two convolutions can be combined into a single operation, creating a two dimensional...
Chapter 8. Metric Computation

kernel. For use later in the thesis we denote the \( u \) central difference kernel as \( \partial_u \), and the \( v \) central difference kernel as \( \partial_v \).

All of our metrics are computed directly from the geometry image, \( G \), as a set of \( x, y, z \) coordinates parametrize by a set of \( u, v \) coordinates:

\[
G(u, v) = \begin{pmatrix} X(u,v) \\ Y(u,v) \\ Z(u,v) \end{pmatrix}.
\] (8.5)

Using the image derivatives we can compute partial derivatives over the geometry images. We define the partial derivatives with respect to \( u \) and \( v \) as \( G_u \) and \( G_v \) respectively. For a double partial derivative, we use \( G_{uu} \) for \( \delta u \delta u \) as an example.

### 8.3.3 Curvature computation

A full explanation of the derivation of the computation of mean and Gaussian curvature is beyond the scope of this thesis, however we will give a brief overview. The curvature computation is based on the work of Mokhtarian et al. [MKY01]. The method requires a parametrisation of the surface, for which our UV coordinates provide a natural one.

The computation of mean and Gaussian curvature is dependent on computing derivatives over the surface. This link between the derivatives and the curvature are defined by the fundamental forms. There are three fundamental forms that define the metric properties of a surface. To compute curvature we only require the first two, which we describe below.

For the first form, which is used as a map from parametric space to movement over the surface we have:

\[
E(u,v) = G_{uu}(u,v) \cdot \vec{n}(u,v),
\] (8.6)

\[
F(u,v) = G_{uv}(u,v) \cdot \vec{n}(u,v),
\] (8.7)

\[
G(u,v) = G_{vv}(u,v) \cdot \vec{n}(u,v).
\] (8.8)

For the second form, which maps changes in the tangent direction into the normal direction. We compute the following coefficients:

\[
L(u,v) = G_{uu}(u,v) \cdot \vec{n}(u,v),
\] (8.9)

\[
M(u,v) = G_{uv}(u,v) \cdot \vec{n}(u,v),
\] (8.10)

\[
N(u,v) = G_{vv}(u,v) \cdot \vec{n}(u,v),
\] (8.11)

where \( \vec{n}(u,v) \) is the computed normal vector at a point in the image.

They compute the above coefficients for each point in our geometry image. We combine the
coefficients to compute the curvature using the following equations for mean and Gaussian curvature:

\[ K(u, v) = \frac{(L(u, v)N(u, v) - M^2(u, v))}{(E(u, v)G(u, v) - F^2(u, v))}, \]

(8.12)

\[ H(u, v) = \frac{(E(u, v)N(u, v) + G(u, v)L(u, v) - 2F(u, v)M(u, v))}{2(E(u, v)G(u, v) - F^2(u, v))}. \]

(8.13)

We can compute the individual maximum and minimum curvatures that compute the mean and Gaussian curvatures with:

\[ M_{\text{max}}(u, v) = H(u, v) + \sqrt{H^2(u, v) - K}, \]

(8.14)

\[ M_{\text{min}}(u, v) = H(u, v) - \sqrt{H^2(u, v) - K}. \]

(8.15)

### 8.3.4 Divergence Computation

The second metric is divergence which is a measure of how the surface of the geometry is expanding or contracting. We compute this for every point on the surface of our subject’s skin using a local neighbourhood around that point. For a point whose neighbourhood is expanding, the divergence is positive; for a point whose neighbourhood is contracting, the divergence is negative. In Chapter 2 we provide a brief overview of divergence as an operator on a vector field. In our case this vector field is the displacement of point over the surface of the geometry.

To compute the displacement over the surface we need to transform displacements in \( \mathbb{R}^3 \) into displacements over the surface of the geometry. A differential displacement over the surface can be approximated by a differential displacement in a local tangent plane. Therefore we can compute divergence by projecting displacements of the local neighbourhood in \( \mathbb{R}^3 \) into the tangent plane of that point. For simplicity, in later sections, we will call this neighbourhood the divergence neighbourhood. A further description of tangential frames can be seen in Chapter 2.

To compute divergence in the tangent plane between frames we must also compensate for how the tangent plane itself changes between frames. For each point in the divergence neighbourhood its projection produces a set of coordinates \((t, b)\). To measure how the points have moved in the plane, when the plane is changing we must ensure two things. The first is that one plane is not a rotation of the other around if their normals were aligned. The second is the centre point of each of the planes. For the rotation compensation we simply ensure that one of the tangent vectors is always a projection of \( \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \) vector on the geometry surface. The compensation is automatically taken care of by the fact that the divergence operator is a differential operator, and is therefore invariant to affine transformation.

Using this equation we can compute the change of corresponding points in the neighbourhood between frames, \( n \) and \( n - 1 \), as:

\[ \Delta \mathcal{G}^n(u, v) = \mathcal{G}^n(u, v) - \mathcal{G}^{n-1}(u, v), (u, v) \in \mathcal{N}. \]

(8.16)
This equation results in a three component vector where the first two components represent displacement over the surface of the geometry, and the last component represent displacement in the direction of the normal to the surface:

\[
\Delta \mathcal{G}(u, v) = \begin{pmatrix} \Delta T(u, v) \\ \Delta B(u, v) \\ \Delta N(u, v) \end{pmatrix},
\]

where we omit the frame number \( n \) for brevity.

Combining this equation, Eq. \( 2.50 \) with our displacement field Eq. \( 8.17 \) and our local tangential basis Eq. \( 2.57 \) we obtain the equation describing surface divergence at a point, \( x \):

\[
\nabla \cdot \Delta \mathcal{G}(u, v) = \Delta \mathcal{T}_t(u, v) + \Delta \mathcal{B}(u, v).
\]

where \( \Delta \mathcal{T}_t \) represents the derivative of the displacement field tangent component with respect to the tangent direction.

### 8.3.5 Divergence Image Space Computation

We now describe how we compute the divergence equation shown in Eq. \( 8.18 \) using image space derivatives. Given our displacement field, \( \Delta \mathcal{G} \) (see Eq. \( 8.16 \)), we define the image space derivative operations that allow us to compute the divergence. To compute the divergence we first compute the Jacobian from the \( (u, v) \) coordinates to the \( (t, b) \) coordinate systems. This is a simple application of
we move all the models into a common cylindrical coordinate system. Then, images from different cameras are blended together to compute a complete image for our subject’s entire facial geometry.

The computation of the metrics is based on empirical observation of our subjects. The observations show that there is an apparent correlation between changes in curvature of the subject’s face, and divergence on the surface of their skin.

By a simple rearrangement of the equations, we can solve for the Jacobian using a sparse matrix solver.

Once we have the Jacobian, we can compute the directional derivative in the tangent plane using another application of the chain rule:

\[
\Delta T_{e}(u,v) = \Delta T_{u}(u,v) \frac{\partial u}{\partial t} + \Delta T_{v}(u,v) \frac{\partial v}{\partial t}, \tag{8.23}
\]

\[
\Delta B_{v}(u,v) = \Delta B_{u}(u,v) \frac{\partial u}{\partial b} + \Delta B_{v}(u,v) \frac{\partial v}{\partial b}. \tag{8.24}
\]

we then substitute them into Eq. 8.18 to compute the divergence.

### 8.4 Summary

The computation of the metrics is based on empirical observation of our subjects. The observations show that there is an apparent correlation between changes in curvature of the subject’s face, and divergence on the surface of their skin.

To explore this relationship, we must therefore compute the curvature changes and surface divergence from our model. We used an image space algorithm based on the fact that our models are embedded in image space. This image space embedding is a result of the preparation steps. First, we move all the models into a common cylindrical coordinate system. Then, images from different cameras are blended together to compute a complete image for our subject’s entire facial geometry.
<table>
<thead>
<tr>
<th>Frame</th>
<th>Haemoglobin</th>
<th>Divergence</th>
<th>Mean Curvature</th>
<th>Gaussian Curvature</th>
</tr>
</thead>
<tbody>
<tr>
<td>5th</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
<tr>
<td>10th</td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
</tr>
<tr>
<td>15th</td>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
</tr>
<tr>
<td>20th</td>
<td><img src="image13.png" alt="Image" /></td>
<td><img src="image14.png" alt="Image" /></td>
<td><img src="image15.png" alt="Image" /></td>
<td><img src="image16.png" alt="Image" /></td>
</tr>
<tr>
<td>25th</td>
<td><img src="image17.png" alt="Image" /></td>
<td><img src="image18.png" alt="Image" /></td>
<td><img src="image19.png" alt="Image" /></td>
<td><img src="image20.png" alt="Image" /></td>
</tr>
</tbody>
</table>

Figure 8.12: Change in haemoglobin concentration (left) against change in the computed metrics: Divergence, Mean Curvature, Gaussian Curvature (left to right) with respect to the initial frame.
Finally, we use tracking and warping on the frames to align them with the initial frame in the capture. Once we have a set of aligned geometry images we compute our metrics. For curvature, we leverage previous research under the realisation that the geometry image is simply a point cloud. For divergence, we present a simple image based method. Standard computer graphics techniques compute surface stretch using triangulated meshes. However, in our case we no longer have a triangulation, and it is unnecessary to generate one.

The results of the metric computation process are shown in Figure 8.12. These results show the change in each of the metrics; haemoglobin, divergence, Mean and Gaussian curvatures, compared to the initial frame. We can observe a strong correlation between divergence and Gaussian curvature changes with haemoglobin changes which agrees with our initial observations. We can observe that the correlation between Mean curvature changes and haemoglobin does not empirically exist, and therefore we do not use it further in this thesis.

In the next chapter, we further analyse the data to find a quantitative model that describes how to find the change in haemoglobin from these metrics.
Chapter 9

Analysis

In this chapter we create an analytic model of the relationship between the metrics and haemoglobin changes. In the previous chapter we computed the metrics for divergence and curvature in image space from our meshes. From observation we can see a strong correlation between divergence, Gaussian curvature and haemoglobin changes. In this chapter, we provide a quantitative analysis of this relationship. This culminates in our model that allows us to compute blood concentration changes from changes in facial geometry.

9.1 Measurements

To measure a wide range of facial movements, we refer back to our original work on simulating changes in haemoglobin (see Chapter [4]. We utilize the same wheel of emotions in this chapter. However, we now capture video footage using the capture software and hardware described in Chapter [5] and Chapter [7]. In addition, we add a Freestyle expression, where the subject was encouraged...
to change their expression at will and in an extreme way as possible. This allows us to capture geometric changes that were not part of our everyday expression gamut. Combining this set of seven per-subject expressions ensures that we capture a wide range of all possible movements to choose from.

Our measurement methodology consists of capturing seven different facial expressions: Anger, Disgust, Fear, Joy, Sad, Surprise and Freestyle. For each one of these expressions, we capture 150 frames, or 10 seconds at the capture frame rate of 15 fps. Due to the high aperture requirements (described in Chapter 6) of the capture system, we requested that the subjects make the facial expression at a slower speed than they would normally. This reduced the amount of motion blur experienced in the capture, but is still valid under our hypothesis. Under these conditions, we have found that a single facial expression typically takes no more than four seconds to complete giving us ample time for the user to start and end the expression. Finally, we process only the 10 seconds from the full recording, reducing the amount of computation required.

9.2 Observations

From Chapter 8, we have observed that there is an empirical correlation between changes in blood concentration, divergence and Gaussian curvature. The change in the metrics is in reference to the initial frame, or neutral expression in the capture. This change is computed simply as the numerical difference between the aligned metric value, or haemoglobin value, at a given pixel and the corresponding pixel in the initial image. In this section, we develop a more quantitative analysis of the relationship between the changes in the metrics and the haemoglobin changes.

From the empirical observations, we can observe that there is a greater correlation between the changes in divergence and the changes in haemoglobin concentration. In Figure 9.2, we plot the divergence over the frames alongside the changes in blood concentration. We can observe, and see from the scatter plot that the relationship between the variables is roughly linear. In fact, computing
the Pearson product-moment correlation coefficient (PCC) between the two variables gives strong values for the blue, magenta and red markers of 0.8476, 0.7958 and 0.9589 respectively. Here a value of -1 and 1 is a perfect correlation, with zero meaning no correlation.

![Figure 9.3: Pearson correlation coefficient between the change in divergence per pixel and the changed in haemoglobin value.](image)

![Figure 9.4: Pearson correlation coefficient between the change in Gaussian curvature per pixel and the changed in haemoglobin value.](image)

To test the applicability of this model to the entire face, we compute a per-pixel PCC with the divergence metric, and as a comparison the PCC with Gaussian curvature. The results of the correlation can be seen in Figure 9.4 and Figure 9.3 for each iteration of the frequency decomposition process (see Figure 8.10). The results show that where there are points of high haemoglobin change the divergence is strongly positively correlated with changes in haemoglobin concentrations. Similarly, the changes in curvature are strongly negatively correlated with changes in haemoglobin. However, where points above the brow change the results are not as strongly correlated as with divergence. Similarly, where the blood concentration goes down on the bridge of the nose, there is no correlation.
Figure 9.5: Absolute value PCC between divergence (left) and haemoglobin change (right). On the top a subject has demonstrated the disgust expression. On the bottom another subject is demonstrating the surprise expression.

We therefore base our model on the divergence calculations. Although, it is obvious that the curvature still shows strong correlation in parts. We posit that where there is convergence the skin will naturally bunch up causing curvature changes. Therefore, we can see that the divergence metric is a more general metric.

In Figure 9.5, we plot side-by-side the absolute PCC value across the expression (left) with the absolute haemoglobin change (right). This demonstrates that where the haemoglobin change is highest, the correlation is also equally as high. This presents the argument that those points with low correlation are simply so because there is no overall haemoglobin change and therefore no relationship to find.

9.3 Model

Based on the observation of the PCC values across the face of our subject, we construct a simple haemodynamic model. We know there is a strong linear relationship between the divergence and changes in haemoglobin concentration.

To fit a linear model we simply perform linear regression on a per-pixel basis. This linear regression is performed using an inverse matrix operation. Given a set of points of an independent
variable, \( x_i \), and a dependent variable, \( y_i \), where \( i \) is the \( i^{th} \) frame, we can find the best fitting line by solving the following matrix equation:

\[
\begin{bmatrix}
x_0 \\
x_1 \\
\vdots \\
x_n
\end{bmatrix}
\begin{bmatrix}
\alpha_0 \\
\alpha_1 \\
\vdots \\
\alpha_n
\end{bmatrix}
= 
\begin{bmatrix}
y_0 \\
y_1 \\
\vdots \\
y_n
\end{bmatrix}
\tag{9.1}
\]

giving us the gradient, \( \alpha_0 \) of the linear relationship. As the relationship stated between the independent and dependent variables is a differential we one, there is no offset, and so the relationship is a simple one of scale.

Figure 9.6: Regressed linear relationship coefficients between surface divergence and haemoglobin changes. On the top a subject has demonstrated the disgust expression. On the bottom another subject is demonstrating the surprise expression.

Examples of the fitted coefficients can be seen in Figure 9.6. The top figure shows an example surprise expression, and the bottom figure shows an example disgust expression. By inspecting the figure we can clearly see that where there is a negative gradient of the linear relationship, there is an
increase in blood concentration. Similarly, where there is a positive gradient there is a decrease.

9.4 Results and Discussion

To test the accuracy of the linear model, we use the coefficients to reconstruct the blood concentration levels over the duration of the capture. We then compute the absolute error relative to the captured blood concentration and plot the results. For two subjects with different expressions we show the results in Figure 9.7 and Figure 9.8.

In both figures we can see that the results are visually very accurate. In the first subject, Figure 9.7, we can see that the model correctly predicts the reduction of blood concentration at the top of the nose and above the eyebrows. In the second subject, Figure 9.8, we see that large errors appear in the model but they are sporadic. Closer inspection of the reconstructed blood concentration levels shows, especially around the wrinkled area, that there is a subtle blurring of the results. This could be due to the accuracy of either the alignment process, or the capture and smoothed geometry. However, for the purposes of use in animation pipeline the results are more than satisfactory visually.

In Figure 9.9, we examine the difference between the original blend model from Chapter 4 and the linear geometric model. Recall that the blend model takes a wheel of emotions, and uses barycentric coordinates to blend between the haemoglobin maps of the different bases, with the neutral map at the centre. In this comparison, we simply reduce the wheel to a line representing our captured sequence; the initial frame is our neutral expression, and the last frame is our target expression. This reduces Eq. 4.15 to a simple one dimensional equation, where $k = 1$, and as we are using barycentric coordinates then the single weight, $w_1$, must also be unity. We then show side-by-side comparisons with the same frame reconstructed using the linear geometric model, along with the ground truth captured frame. Note, the reconstructed frames appear different to those in Figure 9.8 and Figure 9.7, as we have warped them back to the original UV coordinates, rather than tracking space, as they would be used in a rendering pipeline.

As we can see, the linear model reconstructs a lot more of the finer detail, especially where there is a large amount of curvature. The blend model uses a circular kernel to calculate local mean and standard deviation parameters in the area around each pixel, this naturally blurs the blood concentration maps. In addition, the modulation is only applied to the neutral expression, so any details that are not in the original neutral expression will not appear in any target haemoglobin map. In the Disgust, Freestyle, and Surprise expressions show in Figure 9.9, we have highlighted places where the model is much more comparable to the ground truth when using the geometric model vs. the blend model. This highlights a great strength of using the new approach.

However, there are not sufficient results to justify a strong conclusion. This is due to the computational complexity of the pipeline, and the amount of manual effort in processing and tweaking of parameters as described in the previous chapters. In addition, the complexity of the pipeline means that there are a lot of failure points along the way. However, there are several key factors with the
Figure 9.7: The actual blood concentration maps vs the concentration maps generated by the model. On the right, we show the percentage error between the model and the measured blood concentration maps.
Figure 9.8: The actual blood concentration maps vs the concentration maps generated by the model. On the right, we show the percentage error between the model and the measured blood concentration maps.
results that allow us to bring a conclusion. Firstly, the results are for different subjects, and although a very small sample represents reproducibility across different skin types and geometry. Secondly, the per-pixel tracking means that every in the captures is not only a sample spatially, but across time. In effect, each pixel is therefore a independent demonstration of the linear model. Figure 9.2 shows that the correlation between blood concentration changes and the divergence metric is determined at every pixel, and across time. In effect, this means that every pixel represents an independent test for our hypothesis. However, as noted this is still not strong enough for a wide ranging conclusion, but does provide a strong hint that the hypothesis is correct.

In addition, although both results confirm the applicability of a linear relationship between blood concentration changes and temporal divergence of the skin, the question remains as to what the coefficient represents. Initial inspection of the results would suggest that the coefficient is related to curvature changes. However, statistical analysis of this does not bear this out. Another possible explanation is that the coefficient is related to how depressed or expanded the skin is, in the direction of the normal.
Chapter 10

Conclusions

In this thesis, we have investigated the link between changing facial expression and dynamic perfusion changes with the aim of making dynamic skin appearance more realistic. In the first instance, we describe a system in which we drive a dynamic perfusion model by interpolating between haemoglobin maps of a set of fundamental expressions. These are taken from psychological literature and form a wheel that spans a wide gamut of human expression. The interpolation is performed by using histogram matching; a process that preserves detail during the interpolation process. The method produces very good results, and its efficiency means it can be used in a real-time animation pipeline. However, we find the practical use of this approach is limited, as it requires specialist equipment to capture the initial haemoglobin maps. We therefore attempt to extend the model to a more general approach. In the new approach, we first hypothesize that there is a functional link between change in the surface geometry of the subject expression, and resulting changes in blood perfusion. To investigate this we extend the capture device to allow us to simultaneously capture both three-dimensional geometry measurements, and haemoglobin concentration values. We process the geometry to compute differential metrics over its surface and attempt to extract a relationship with perfusion changes. Investigations suggest a linear relationship, but there is not enough processed data to draw a strong conclusion. However, the results that have been processed are strong on their own, and provide enough evidence to warrant further investigation in the future.

In conclusion, we have demonstrated that blood perfusion changes can be incorporated into an animation pipeline. This is driven by several insights into the nature of blood perfusion derived from our capture pipeline. This pipeline is described in great technical detail, with insights into calibration and hardware selection to increase accuracy and applicability across different potential capture rigs. Subsequent to this, we have shown how to combine the chromophore capture process with a stereoscopic reconstruction pipeline. This final combination lead to insights into possible connections between geometric changes in the subject’s facial expressions and perfusion changes. Although there is not sufficient evidence to draw a strong conclusion; the nature of the per-pixel tracking means that each point represents a separate, independant experimental observation. Combined with the capture
resolution, this means that the conclusion can be strongly made, but only for these particular results. This, at least, hints that the model is applicable for different expressions and across different subjects.

The thesis raises several interesting observations about the nature of blood perfusion. Our initial experiments showed that perfusion is generally confined to a local scaling of an underlying vascular network. This allowed us to use local histogram matching to transfer statistical information between corresponding points in an animation pipeline. The locality of the perfusion effects is also demonstrated in the independence of the per-pixel observations in our linear model. The direct linear connection between the displacement metric and haemoglobin changes mean that the effect is also localised. That is, the effect can be entirely described by a scale coefficient at each pixel. The neighbours are only taken into effect in the differential displacement metric calculation, and possibly in the very nature of the coefficient. The linear model also raises another interesting point, that the relationship between perfusion and geometry is a holographic one. That is, the metric is purely defined by the surface of the facial expression; the very nature of the mathematical definition of the divergence implies a two-dimensionality. From this two different possibilities arise; that the vascular network is thin enough to be encoded as two-dimensional, or the surface encodes. However, further investigation would be required to confirm this.

In addition, although the model describes a linear relationship, it is not clear what the coefficient in the relationship is. As discussed in Section 9.4, it appears this may be related to curvature changes of the skin surface; although, an initial statistical analysis do not confirm this. Another possible explanation is it being related to the depression or expansion along the normal direction of the skin’s surface. However, at this point any answer to this question is mere speculation, and it could be the subject of future work to answer these questions.
Appendix A

Polarization

The common way to remove specularly reflected light from an object is to take advantage of a property of light known as polarisation. Light is an electromagnetic wave, where oscillations in the electric and magnetic fields occur in the plane perpendicular to direction of wave travel. The magnetic field oscillations are always perpendicular to the electric field oscillations in this plane, and so it is sufficient to describe a wave fully using only electric field oscillations. The direction of the oscillations in this plane over time describes the polarisation.

Within the perpendicular plane, the electric field strength will change in both direction and magnitude as the wave propagates through space. The vector in this plane can be further decomposed into two perpendicular vectors. If these vectors are in phase the addition of these vectors forms a line in our perpendicular plane which changes in magnitude but not in direction. This is known as linear polarisation. If one of these perpendicular vectors is out of phase, the magnitude doesn’t change but the direction does as this is known as circular polarisation. In between is known as elliptical.

The upper layers of skin preserve the linear polarisation of specularly reflected light. The surface skin acts like a dielectric, or insulator, which means charge will no flow across it. This means, that when the wave exerts an electromagnetic force on an atom on the surface, the electron shell of the atom temporarily distorts but no electrons flow. Once the force is removed the electron shell returns to its ground state, this distortion produces an electromagnetic wave of equal and opposite phases s and t waves. For a linearly polarised wave this produces a wave of the same linear polarisation but different phase, and for circular polarisation a circularly polarized wave but of different chirality.

The preservation of linear polarised light allows us to remove specularly reflected light using polarised filters. If we illuminate our subject with vertically polarised light, we know that specularly reflected light will also be vertically polarised. If we view the subject through a horizontal polarising filter we immediately cut out any vertically, and in this case specularly reflected light. We therefore design our system to have vertically polarised lights, and a horizontal filter over our measuring device.

For greatest effect, we need to place both the filters and light in a single plane containing our horizontal polarisation direction. As we move away from this plane, the basis in which our
polarisation oscillates becomes rotated in our perspective. This rotation means that our cross polarized filter will out remove the projection of the rotated basis. This means we design our such that the lights and the camera light on the same plane.
Appendix B

Camera Calibration Parameters

The minimum possible system for stereo reconstruction is a binocular system consisting of two
different camera viewpoints used to reconstruct an object. We can explore this system theoretically by
modeling each camera using the pinhole camera model \[HZ03\]. This model describes mathematically
the transformation from the three-dimensional world to the two-dimensional image plane. The model
identifies the origin of the camera which is the centre of projection from the three-dimensional world
to the two-dimensional image plane. This point corresponds to the aperture of the camera. The
triangulation is formed by three vectors; the vector connecting the two camera origins to each other,
and the two vectors from the respective origins to the point on the object’s surface.

To accurately describe the pinhole camera transformation we must measure the parameters of the
cameras. The first set of parameters are the camera’s intrinsic parameters. These parameters describe
the internal parameters of the camera such as the ratio of lengths in image space to real world space,
and lens distortion parameters. The second set of parameters are the camera’s extrinsic parameters.
These parameters describe the translation and rotation of the cameras relative to some fixed point in
the real world, normally a calibration target. Both sets of parameters are the only parameters required
to fully describe our system, and are typically found using calibration methods.

Once we have the calibration data describe the system, binocular stereo reconstruction becomes
a task of finding matching points in both images. This process is known as stereo correspondence.
Once matching points are found, we can reconstruct the triangulation in three-dimensional space,
using the calibration data. In binocular stereo systems, this is simplified by using the extrinsic
and intrinsic parameters to rectify both sets of images. This rectification process ensures epipolar
lines are horizontal in both images \[LF96\]. As a consequence, stereo correspondence can be found
by scanning along corresponding horizontal lines in both images and computing similarly metrics
between different points on each line in both images. This greatly reduces the complexity of the
problem.

This methodology for binocular reconstruction means advances in binocular stereo reconstruction
come largely from improvements in the stereo matching algorithms. A full discussion of stereo
matching cost functions is outside the scope of this thesis but is discussed in great detail in Scharstein and Szeliski [SS02].

### B.1 Linear Intrinsic Parameters

The first calibration step is to calibrate for the intrinsic parameters of the camera. The intrinsic calibration describe camera properties such as focal length, image sensor format and the principal point. Together the intrinsic parameters can be combined into the intrinsic matrix that transforms normalized camera coordinates into coordinates in the image:

\[
\begin{pmatrix}
\alpha_x & \gamma & u_0 \\
0 & \alpha_y & v_0 \\
0 & 0 & 1
\end{pmatrix}
\]  

(B.1)

The first parameters describe the focal length of the lens in terms of pixels. The size of the pixels is determined by their size of the photosites on the sensor. The photosites need not be square and so the focal length may be a different size in terms of pixels in the x-direction versus pixels in the y-direction; hence two parameters \(\alpha_x\) and \(\alpha_y\).

The second set of parameters describe the principal point of the camera. This point represents the pixel position in the image of the optical center of the lens. This is ideally the centre of the image and is represented by the coordinates \(u_0\) and \(v_0\).

The last parameter is the skew coefficient. This parameters represents the skew of the photosites where they do not have a perfect right-angled form. However, it is generally accepted that with modern manufacturing methods this can be approximated as zero. We do so in this calibration process which reduces the number of possible degrees of freedom to calibration for.

### B.2 Non-linear Intrinsic Parameters

The final set of intrinsic calibration parameters are the non-linear calibration parameters describing the lens distortion. Radial distortion can be approximated using the Brown-Conrady [Con19] distortion model. The model can have an infinite number of parameters describing a infinite number of spherical basis functions. For a lens with a relatively short focal length, such as those use with a DSLR camera, we only require a small number of coefficients to represent the largest of the spherical distortion components. In our case we use two coefficients:

\[
x_d = x_u \left(1 + K_1 r^2 + K_2 r^4\right)
\]  

(B.2)

\[
y_d = y_u \left(1 + K_1 r^2 + K_2 r^4\right)
\]  

(B.3)

Here \(x_d\) and \(y_d\) represent the pixel coordinates in the image after the lens distortion has taken
affect, where $x_u$ and $y_u$ are the original coordinates. The coefficients $K_1$ and $K_2$ represent the magnitude of the distortion based on the distance of pixel coordinates from the optical centre of the lens.

Typically the lens distortion model contains a series of tangential distortion components that control off-center distortion of the lens. However, due to the high manufacturing standards of the camera purchased this was deemed to be unnecessary, and remove of these coordinates generally improved the results of the calibration process.
Appendix C

Thermal Noise Model

The use of temperature variability of the capture environment can greatly affect the noise from the sensor. Dark current is a measure of the noise on a sensor when it is not exposed to light, and increases as temperature and exposure increase. During our capture process, the duration of the capture and the use of halogen lights causes the sensors to heat up over time. Therefore for every frame that we capture the dark current profile will be different.

To be able to compensate for dark current noise, we use a model that can describe it for each captured temperature. [JC09] shows that dark noise is linearly dependent on temperature. We capture the dark noise pattern over a steadily increasing temperature, with our lights acting as a heat source. We, then create a linear model of dark noise using the on-board temperature readings of the camera. This takes the form of a scale and bias linear relationship, where the scale and bias is determined per pixel:

\[ N(x, y) = S(x, y)T + B(x, y) \] (C.1)

where \( T \) is the temperature, \( S \) is the per pixel scale, \( B \) is the per pixel bias, and \( N \) is the resultant dark noise image.

We then correct for dark noise on a particular image, by using its recorded temperature and generating a noise image using the above equation. This is then subtracting from the image, before de-bayering takes place.

\[ ^1 \text{An image taken with the lens cap on} \]
Bibliography


[BBA+07] Bernd Bickel, Mario Botsch, Roland Angst, Wojciech Matusik, Miguel Otaduy, Hanspeter Pfister, and Markus Gross. Multi-scale capture of facial geometry and


