UNIVERSITY OF CALIFORNIA, SAN DIEGO

Controlling the Appearance of Specular Microstructures

A dissertation submitted in partial satisfaction of the

requirements for the degree

Doctor of Philosophy

in

Computer Science

by

Iman Sadeghi

Committee in charge:

Henrik Wann Jensen, Chair
Sheldon Brown
Samuel R. Buss
David J. Kriegman
Jurgen P. Schulze

2011

The dissertation of Iman Sadeghi is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Chair

University of California, San Diego

2011

To my parents
Jila and Khosrow,

and my brothers
Behrang and Nima.

کار ما نیست شناسایی راز گل سرخ،
کار ما شاید این است
که در افسون گل سرخ شناور باشیم

سهراب سپهری

TABLE OF CONTENTS

Signature Page iii

Dedication iv

Epigraph v

Table of Contents vi

List of Figures x

List of Tables xvi

Acknowledgements xvii

Vita and Publications xx

Abstract of the Dissertation xxii

Chapter 1 Introduction 1
1.1 Summary of Original Contributions 10
1.1.1 Light Interactions with Microspheres 10
1.1.2 Light Interactions with Microcylinders 11
1.1.3 Light Interactions with Interwoven Microcylinders 11
1.2 Organization of the Dissertation 12

Chapter 2 Nature and Behavior of Light 14
2.1 Different Optical Models 14
2.1.1 Ray Optics 15
2.1.2 Wave Optics 16
2.1.3 Electromagnetic Optics 18
2.1.4 Quantum Optics 19
2.2 Light Interactions with Matter 20
2.2.1 Reflection and Refraction 20
2.2.2 Interference 25
2.2.3 Diffraction 27
2.2.4 Polarization 28
2.2.5 Absorption 29
2.3 Light in Computer Graphics 30
2.3.1 Radiometry 30
2.3.2 The BRDF 33
2.3.3 The Rendering Equation 35
2.3.4 The Phase Function 35

2.3.5 The Volume Rendering Equation 37

Chapter 3 Light Interactions with Microspheres 38
3.1 Introduction 40
3.2 Background Theory 43
3.2.1 Dispersion 43
3.2.2 Interference 44
3.2.3 Diffraction 46
3.2.4 Non-Spherical Water Drops 47
3.3 Previous Work 48
3.3.1 Based on Ray Optics 49
3.3.2 Lorenz-Mie Theory 50
3.3.3 FDTD Methods 53
3.4 Our Approach 53
3.4.1 Overview 55
3.4.2 Simulating Dispersion 57
3.4.3 Computing Interference and Polarization 58
3.4.4 Approximating Diffraction 65
3.5 Results 66
3.6 Conclusion and Future work 73
3.7 Acknowledgments 74

Chapter 4 Light Interactions with Microcylinders 75
4.1 Introduction 76
4.2 Background Theory 77
4.2.1 Notations of Hair Geometry 77
4.2.2 Radiometry of Hair Fibers 78
4.2.3 Physical Properties of Hair Fibers 80
4.3 Previous Work 81
4.3.1 Physically Based Shading Models 81
4.3.2 Ad Hoc Shading Models 86
4.4 Artist Friendly Control Parameters 88
4.4.1 Intuitive behavior 89
4.4.2 Decoupling 90
4.4.3 Going beyond reality 91
4.5 Our Approach 93
4.6 Applying Our Approach 96
4.6.1 Single Scattering 96
4.6.2 Multiple Scattering 103
4.6.3 Decoupling Single and Multiple Scattering 108
4.7 Multiple Scattering Implimentation 109
4.7.1 Revised Dual Scattering Model 110
4.7.2 Implementation 117

4.8 Results 119
4.9 Comparisons 123
4.9.1 Setup 124
4.9.2 Ranking 126
4.9.3 Discussion 127
4.9.4 Performance 128
4.10 Conclusion and Future Work 129
4.11 Acknowledgments 130

Chapter 5 Light Interactions with Interwoven Microcylinders 131
5.1 Introduction 132
5.2 Background Theory 133
5.2.1 Notations of Cloth and Thread Geometry 133
5.2.2 Cloth Fibers 135
5.2.3 Cloth Threads 136
5.2.4 Weaving Patterns 137
5.3 Previous Work 137
5.3.1 Empirical Shading Models 138
5.3.2 Virtual Goniometric Methods 138
5.3.3 Microfacet Based Models 139
5.3.4 Data Acquisition Approaches 140
5.3.5 Physically Based Models 140
5.4 Light Scattering from Fabrics 141
5.4.1 Acquisition Setup 141
5.4.2 BRDF Measurements and Observations 143
5.5 Light Scattering from Threads 148
5.5.1 Acquisition Setup 149
5.5.2 BRDF Measurements 149
5.5.3 A Light Scattering Model for Threads 152
5.5.4 Matching Measured BRDFs 156
5.6 An Appearance Model for Cloth 158
5.6.1 Shading Model 158
5.6.2 Shadowing and Masking 160
5.6.3 Reweighting 163
5.7 Results 165
5.8 Limitations 174
5.9 Conclusion and Future Work 175
5.10 Acknowledgments 176

Chapter 6 Conclusions and Future Work 177
6.1 Contributions 177
6.2 Future Research Directions 178

Appendix A Rainbow Rendering Results Parameters 180

Appendix B Artist Friendly Vocabulary for Hair Rendering 188

Appendix C RSL Code for the Dual Scattering Model 189

Appendix D Hair Rendering User Study Supplementary Data 193

Appendix E Tangent Distribution Curves for Cloth Rendering 210

Bibliography 213

LIST OF FIGURES

Figure 1.1: In this dissertation we focus on the appearances caused by the following microstructures: microspheres, microcylinders, and interwoven microcylinders. 2
Figure 1.2: In this dissertation we focus on the appearances of rainbows, hair, and cloth. 3
Figure 1.3: Different types of rainbows: a double rainbow, multiple supernumerary bows, and a rare twinned rainbow. 4
Figure 1.4: Different types of human hair: blonde hair, red hair and dark brown hair. 6
Figure 1.5: Different types of cloth fabrics: red linen plain, blue silk satin and yellow polyester charmeuse. 8

Figure 2.1: Different optical models for describing the nature of light: Ray optics, wave optics, electromagnetic optics and quantum optics. 15
Figure 2.2: Electric wave E is a scalar function that oscillates perpendicular to the direction of its propagation. 18
Figure 2.3: Electromagnetic wave is a paired couple of vector waves: an electric wave $\vec{E}$ and a magnetic wave $\vec{H}$. These two wave vectors are perpendicular to the direction of wave vector $\vec{k}$. 19
Figure 2.4: The geometry of the reflection and refraction of light. 21
Figure 2.5: The geometry of light reflection. 22
Figure 2.6: The geometry of light refraction. 23
Figure 2.7: A computer generated image that uses Fresnel's equations in order to simulate reflection and refraction of light. 24
Figure 2.8: Effect of different optical phenomena on the appearance of rainbows: Dispersion, interference, and diffraction. 25
Figure 2.9: Different types of interference: Constructive, destructive and the general case. 26
Figure 2.10: Thin film interference of light. 26
Figure 2.11: Examples of iridescent surfaces caused by interference: A soap bubble, a Morpho butterfly and a beetle. 27
Figure 2.12: Diffraction of light as it hits a surface with a small hole or a sharp edge. 28
Figure 2.13: Polarization state of the light is perpendicular to the direction of the wave vector $\vec{k}$. 29
Figure 2.14: The BRDF expresses the reflection profile from a point on the surface for any pair of viewing and lighting directions. 34
Figure 2.15: Phase function expresses the angular distribution of light scattering at a given point inside a medium. 36

Figure 3.1: Our rendering results for different types of rainbows. 38

Figure 3.2: Different rainbows seen in nature. 41
Figure 3.3: Examples of twinned rainbows. 42
Figure 3.4: Dispersion of light as it gets reflected and refracted by the water drop causes the primary and the secondary bow. 44
Figure 3.5: Generation of rainbows from the point of view of geometric and wave optics: 45
Figure 3.6: A focal line defines the curve along which all the differential focal points of the wavefront lie. 46
Figure 3.7: Graph of intensity as a function of scattering angle for the primary rainbow. 47
Figure 3.8: Nonspherical raindrop shape with increasing radii as proposed by Beard and Chuang [BC87]. 48
Figure 3.9: Hand drawn sketches of Rene Descartes and Isaac Newton on explaining how primary and secondary rainbows are formed . . 50
Figure 3.10: Lee diagram showing the variation in appearance of primary and secondary rainbows. 51
Figure 3.11: Steps of our algorithm. 54
Figure 3.12: Comparison between the different phenomena simulated by our method (dispersion, interference and diffraction) and the same simulation from the Lorenz-Mie theory. 59
Figure 3.13: Direction **r** intersects patches $k1$ and $k2$. 62
Figure 3.14: Comparison between our method and Lorenz-Mie theory for different water drop sizes. 64
Figure 3.15: The inserts in these images show how our model can reproduce the rainbows in the underlying photographs. 67
Figure 3.16: Comparison of renderings of rainbows owed to different water drop radii between Lorenz-Mie and our solution. 70
Figure 3.17: The effect of different water drop radii on the apparent geometry of the rainbow. 71
Figure 3.18: The effect of the inclination of the sun on a non-spherical water drop alters the aparent geometry of the rainbow. 72
Figure 3.19: The effect of the different polarization states on the perception of a rainbow from 0.5mm radius non-spherical water drop. . . 72
Figure 3.20: Photograph of a twinned rainbow as well as a twinned rainbow simulated using our algorithm. 73

Figure 4.1: Rendering results using our novel hair shading system for different lighting and viewing directions 75
Figure 4.2: The geometry notations used for describing the light scattering form cylinders. 79
Figure 4.3: The structure of human hair fibers. Human hair consists of three main components: the cortex, the medulla, and the cuticle. 81
Figure 4.4: Reflection cone from a smooth cylinder. 82

Figure 4.5: Light reflections off of a cylindrical hair fiber in the azimuthal and longitudinal direction. 84
Figure 4.6: Evolution of azimuthal scattering functions. 85
Figure 4.7: Evolution of longitudinal functions. 86
Figure 4.8: Some visual development art references from the animated feature film Tangled. 87
Figure 4.9: A frame from the film 101 Dalmatians. 88
Figure 4.10: Final hair colors and their corresponding RGB absorption coefficients. Coming up with appropriate values for the absorption coefficients to get a desired hair color is not intuitive. 89
Figure 4.11: Comparison between coupled and decoupled control parameters. 92
Figure 4.12: Unexpected behavior of a physically based shading model after some non-physical modifications to the underlying hair scattering functions. 93
Figure 4.13: A schematic visualization of our approach. 95
Figure 4.14: Single scattering subcomponents: R, TT, and TRT. 98
Figure 4.15: Visualizing the primary highlight's control parameters on its corresponding azimuthal scattering function and a frontlit rendering. 100
Figure 4.16: Visualizing the transmission component's control parameters on the corresponding azimuthal scattering function and a backlit rendering. 101
Figure 4.17: Visualizing the secondary highlight's control parameters on the corresponding azimuthal scattering function and a frontlit rendering. 102
Figure 4.18: Different quantities involved in the computation of the Dual Scattering method. 104
Figure 4.19: Visualizing the decomposed components of our hair shading model. 105
Figure 4.20: The dual scattering method separates the multiple scattering component into global multiple scattering and local multiple scattering. 111
Figure 4.21: Some rendering results based on our implementation of the dual scattering model. 115
Figure 4.22: Sub-components of the dual scattering method. 118
Figure 4.23: Applying blond hair to one of our existing characters. 119
Figure 4.24: The hair shading system has been used successfully for rendering the long blond hair of the character Rapunzel, the short brown hair of the character Flynn, and the white hair/fur on the horse. 121
Figure 4.25: Reproducing the look for the character Penny using the original hair shader, and our new hair shader 122

Figure 4.26: Painted art reference and a corresponding rendering using our new hair shader. 123
Figure 4.27: Reference photographs used for the user study. 125
Figure 4.28: Normalized distribution of the rankings for all the shading models.126
Figure 4.29: Comparing one of the results of the New shader with one of the results of the Research shader. 128
Figure 4.30: Comparing one of the results of the New shader with one of the results of the Production shader. 129

Figure 5.1: Image rendered using our microcylinder shading model with intuitive parameters for different fabric types. 131
Figure 5.2: Cloth appearance for three different fabrics which are the main focus of our study in this paper. 133
Figure 5.3: Different weaving patterns of the fabrics shown in Figure 5.2. . 133
Figure 5.4: Notations and geometry of light reflection from a cylindrical fiber.134
Figure 5.5: Longitudinal angles ψ_i and ψ_r. 135
Figure 5.6: Microscope photos of different types of cloth fibers. 136
Figure 5.7: Microscope photos of different types of cloth threads and weaving patterns. 137
Figure 5.8: UCSD Gantry. 142
Figure 5.9: An illustration of the acquistion setup and coordinate system for measuring cloth. 143
Figure 5.10: Microscope pictures of the Linen Plain fabric. 144
Figure 5.11: BRDF measurements for Linen Plain for 0^o, 30^o, and 60^o incident angles. 144
Figure 5.12: Microscope pictures of the Silk Crepe de Chine fabric. 145
Figure 5.13: BRDF measurements for Silk Crepe de Chine for 0^o, 30^o, and 60^o incident angle. 146
Figure 5.14: Microscope pictures of the Polyester Satin Charmeuse fabric. . 147
Figure 5.15: The front and back of the Polyester Satin Charmeuse fabric. . . 147
Figure 5.16: BRDF measurements for Polyester Charmeuse for 0^o, 30^o, and 60^o incident angles. 148
Figure 5.17: An illustration of the acquistion setup for measuring the BRDF of threads. 150
Figure 5.18: Polar plots of measured incidence plane BRDF for different threads. 151
Figure 5.19: Reflection cone off of a smooth cylinder. 153
Figure 5.20: Incidence plane thread BRDF measurements in the top row matched by thread model in the bottom row. 157
Figure 5.21: Our shading model treats the fabric as a mesh of microcylinders oriented in two orthogonal directions. 159
Figure 5.22: The weaving pattern and a sample tangent curve for the Polyester Satin Charmeuse fabric. 160

Figure 5.23: Fabric as two different directions of threads with views from above and from grazing angles. 162
Figure 5.24: To capture the anisotropic behavior of different fabrics, we have wrapped the fabric around a cylinder in three different orientations. 165
Figure 5.25: Photograph and the rendering result for the Linen Plain fabric. 167
Figure 5.26: Photograph and the rendering result for the Silk Crepe de Chine fabric. 168
Figure 5.27: Photograph and the rendering result for the front side of Polyester Satin Charmeuse fabric. 169
Figure 5.28: Photograph and the rendering result for the back side of Polyester Satin Charmeuse fabric. 170
Figure 5.29: Matching a BRDF measurement of the Polyester Satin Charmeuse fabric with our model. 171
Figure 5.30: Comparison between photographs and rendered samples for Silk Shot Fabric. 171
Figure 5.31: Comparison between photographs and rendered samples for Velvet Fabric. 172
Figure 5.32: Our rendered results for different fabrics: Linen Plain, Silk Crepe de Chine, front side of Polyester Satin Charmeuse, back side of the Polyester Satin Charmeuse, Silk Shot Fabric, Velvet, and two imaginary fabrics. 173
Figure 5.33: The effect of shadowing and masking and the reweighting process on the final results. 174
Figure 5.34: Screenshots from the GPU implementation of our model. Each image was rendered in roughly 100 ms on a mobile GT420 GPU. 175

Figure A.1: Parameters used for rendering of Figure 3.1 181
Figure A.2: Parameters used for rendering of Figure 3.14 182
Figure A.3: Parameters used for rendering the top row images of Figure 3.15 183
Figure A.4: Parameters used for rendering the bottom row images of Figure 3.15 184
Figure A.5: Parameters used for rendering of Figure 3.16 185
Figure A.6: Parameters used for rendering of Figure 3.17 186
Figure A.7: Parameters used for rendering of Figure 3.20 187

Figure D.1: Photograph references used in the user study. 194
Figure D.2: Rendering results of artist A using the new shader. 195
Figure D.3: Rendering results of artist B using the new shader. 196
Figure D.4: Rendering results of artist C using the research shader. 197
Figure D.5: Rendering results of artist D using the new shader. 198
Figure D.6: Rendering results of artist E using the new shader. 199
Figure D.7: Rendering results of artist F using the production shader. . . . 200

Figure D.8: Rendering results of artist G using the new shader. 201
Figure D.9: Rendering results of artist H using the production shader. . . . 202
Figure D.10: Rendering results of artist I using the research shader. 203
Figure D.11: Rendering results of artist J using the production shader. . . . 204
Figure D.12: Rendering results of artist K using the research shader. 205
Figure D.13: Rendering results of artist L using the research shader. 206
Figure D.14: Rendering results of artist M using the new shader. 207
Figure D.15: Normalized distribution of the rankings for our hair rendering user study. 209

Figure E.1: The tangent curves of all fabrics presented in our cloth appearance model. 211

LIST OF TABLES

Table 2.1: List of index of refraction η for different media. 17

Table 3.1: Water drop polar curve coefficients [BC87]. 48

Table 3.2: List of index of refraction ρ of water for diffrent wavelengths of light λ. The values are derived from the data presented in [HQ73] 57

Table 3.3: Standard deviation of the Gaussian filter diffraction approximation for various water drop sizes. 66

Table 3.4: The list of parameters used to produce the rendering results in Figure 3.15 68

Table 4.1: Summary of all terms used in our implementation of the dual scattering model. 112

Table 5.1: Description of important symbols used in our thread BRDF model.153

Table 5.2: The list of parameters obtained from our measured cloth samples.166

Table B.1: Technical terms and their corresponding artist friendly terms. . . 188

Table D.1: Complete list of evaluations for the hair rendering user study. . . 208

Table E.1: The list of parameters used to produce the tangent curves for each fabric in our cloth appearance model. 212

ACKNOWLEDGEMENTS

[To be completed in the final version.]

FIRST and foremost I would like to thank my parents Jila and Khosrow, for they not only brought me into this world, but taught me how to observe, investigate, and learn from it. I also like to thank my brothers Behrang and Nima whom I grow up with and learned the mysteries of the world around me. This dissertation is dedicated to my family.

I would like to thank my Ph.D. advisor, Henrik Wann Jensen, for his support and guidance throughout my study. It has been a great privilege working with him. Thanks are also due to the other members of my dissertation committee, Sheldon Brown, Samuel Buss, David Kriegman, and Jurgen Schulze for agreeing to take their valuable time to read and evaluate this dissertation.

I would like to thank other faculty members at UCSD whom I have learned a lot during my graduate studies. Particularly, I like to thank Matthias Zwicker, Steve Rotenberg, Falko Kuester, and Serge J. Belongie.

I would like to especially thank Toshiya Hachisuka for he has been a lab mate and a teacher at the same time. I have enjoyed stimulating (and sometimes heated) discussions with him over the years.

Personally, I would like to thank my friends who have been influential on my research, sometimes directly and sometimes indirectly: Aria Arianezhad, Krystle de Mesa, Hani Goodarzi, Bridgette Wiley, Will Chang, Wan-Yen Lo, Brian Whited, Denis Kovacs, Emily Whiting, Hao Lee, Jess Ponce, Kooshiar Azimian, Hourieh Fakourfar, Hanieh Mirzaee, and Farnaz Shariat.

I would also like to thank all of my coauthors from which I have learned a lot during our collaborations: Oleg Bisker, Bin Chen, Joachim De Deken, Mohammad Ghodsi, Diego Gutierrez, Wojciech Jarosz, Henrik Wann Jensen, Ramtin Khosravi, Philip Laven, Shahriar Mirabbasi, Adolfo Munoz, Derek Nowrouzezahrai, Heather Pritchett, Farshad Rostamabadi, Nima Sadeghi, Francisco Seron, Rasmus Tamstorf, and Charles Tappan.

Other members of UCSD graphics group, and also VISGroup at CalIT^2 deserve special recognition, in particluar: Kai-Uwe Doerr, Carlos Dominguez, Craig Donner, Neel Joshi, Cyrus Jam, Alex Goldberg, Jason Kimball, Daniel Knoblauch, Iman Mostafavi, Marios Papas, Vid Pertrovic, Kevin Ponto, Volker Schoenefeld, Robert ThomaS, Tom Wypych, So Yamaoka.

I would like to thank Walt Disney Animation Studios for providing me with two extremely exciting and challenging internship opportunities. In particular I like to thank Rasmus Tamstorf, Joe Marks, Heather Pritchett, Lewis Siegel ...

I would like to thank Industrial Light & Magic for providing me a fun and stimulating internship opportunity. In particular, I like to thank Olivier Maury, David Bullock, ...

I like to thank Pixar Animation Studios for their Geriś Game short which inspired me to pursue computer graphics.

And last but not least, I like to thank the awesome whether of San Diego and its beautiful beaches!

Portions of this dissertation are based on papers which I have co-authored with others. My contributions to each of these papers is listed below.

- Chapter 3 is a reproduction of the material submitted for publication in the article:

  > Iman Sadeghi, Adolfo Munoz, Philip Laven, Wojciech Jarosz, Francisco Seron, Diego Gutierrez, and Henrik Wann Jensen. "Physically-Based Simulation of Rainbows." Conditionally Accepted to *ACM Transactions on Graphics*, 2011.

  I was the primary investigator and author of this paper.

- Chapter 4 is based on material published in the following paper and extended technical report:

  Iman Sadeghi, Heather Pritchett, Henrik Wann Jensen, and Rasmus Tamstorf. “An Artist Friendly Hair Shading System.” In *ACM Transactions on Graphics , (Proceedings of SIGGRAPH 2010)*, 29(4), 2010.

  Iman Sadeghi, and Rasmus Tamstorf. “Efficient Implementation of the Dual Scattering Model in RenderMan.” Technical Report. *Walt Disney Animation Studios*, 2010.

  I was the primary investigator and author of both papers.

- Chapter 5 is a reproduction of the material submitted for publication in the article:

  Iman Sadeghi, Oleg Bisker, Joachim De Deken, and Henrik Wann Jensen. “A Practical Microcylinder Appearance Model for Cloth Rendering.” Submitted to *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2011)*, 2011.

  I was the primary investigator and author of this paper.

VITA

6/2006	B.Sc. in Computer Engineering, Sharif University of Technology, Tehran, Iran
6/2008	M.Sc. in Computer Science, University of California, San Diego
6/2008 - 9/2008	R&D Intern, Walt Disney Animation Studios, Burbank, CA
6/2009 - 9/2009	R&D Intern, Walt Disney Animation Studios, Burbank, CA
6/2010 - 9/2010	R&D Intern, Industrial Light & Magic, San Francisco, CA
6/2011	Ph.D. in Computer Science, University of California, San Diego

PUBLICATIONS

Iman Sadeghi, Oleg Bisker, Joachim De Deken, and Henrik Wann Jensen. "A Practical Microcylinder Appearance Model for Cloth Rendering." Submitted to *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2011)*, 2011.

Iman Sadeghi, Adolfo Munoz, Philip Laven, Wojciech Jarosz, Francisco Seron, Diego Gutierrez, and Henrik Wann Jensen. "Physically-Based Simulation of Rainbows." Conditionally Accepted to *ACM Transactions on Graphics*, 2011.

Nima Sadeghi, Iman Sadeghi, and Shahriar Mirabbasi. "Analysis and Design of Monolithic Resistors with Desired Temperature Coefficient." Submitted to *IEEE Transactions on Electron Devices*, 2011.

Wojciech Jarosz, Derek Nowrouzezahrai, Iman Sadeghi, and Henrik Wann Jensen. "A Comprehensive Theory of Volumetric Radiance Estimation using Photon Points and Beams." In *ACM Transactions on Graphics*, 30(1). 2011.

Iman Sadeghi, Heather Pritchett, Henrik Wann Jensen, and Rasmus Tamstorf. "An Artist Friendly Hair Shading System." In *ACM Transactions on Graphics , (Proceedings of SIGGRAPH 2010)*, 29(4), 2010.

Iman Sadeghi, and Rasmus Tamstorf. "Efficient Implementation of the Dual Scattering Model in RenderMan." Technical Report. *Walt Disney Animation Studios*, 2010.

Iman Sadeghi, Bin Chen, and Henrik Wann Jensen. "Coherent Path Tracing." In *Journal of Graphics, GPU, & Game Tools*, 14(2), 2009.

Iman Sadeghi. "Photorealistic Rendering of Human Hair Fibers." Masters Research Exam. *University of California San Diego*, 2008.

Farshad Rostamabadi, Iman Sadeghi, Mohammad Ghodsi, and Ramtin Khosravi. “Optimal Point Removal in Closed-2PM Labeling.” In *Information Processing Letters*, 105(3), 2008.

PATENTS

Iman Sadeghi, Charles T. Tappan, and Rasmus Tamstorf. “System and Method for Artist Friendly Controls for Hair Shading”, *US Patent Pending*, 2010.

ABSTRACT OF THE DISSERTATION

Controlling the Appearance of Specular Microstructures

by

Iman Sadeghi

Doctor of Philosophy in Computer Science

University of California, San Diego, 2011

Henrik Wann Jensen, Chair

THE focus of this dissertation is the appearance modeling of specular microstructures. We limit our study on microspheres, microcylinders, and a mesh of interwoven microcylinders. These microstructures are involved in the appearance of rainbows, human hair, and cloth fabrics. There has been a lot of previous research on the appearance modeling of these three subjects. However, previous models either fail to reproduce specific appearances or they do so at the high price of profuse and time-consuming parameter tweaking. The goal of this dissertation, first and foremost, is to present novel practical appearance models that do not suffer from these shortcomings. Our novel models have robust physical basis and incorporate intuitive parameters for controlling the final appearance. Our appearance models facilitate the reproduction of a wider range of desired appearances for rendering rainbows, hair fibers, and cloth fabrics.

Fist, we present a novel appearance model for rendering rainbows. We introduce a ray tracing framework for simulating light interactions by particles with arbitrary shapes. We validate our approach against Lorenz-Mie theory for the case of spherical water drops, and show that our model can predict the light scattering behavior of large water drops which have non-spherical shapes. By considering the physically based shape of water drops as well as the effect of sun's inclination we present a simulation of twinned rainbows for the first time.

Next, we present a novel hair appearance model which addresses the problem of art-directability of physically-based appearance models. We introduce a novel approach for creating an art-directable hair shading model from existing physically based models. Through a user study we show that this system is easier to use compared to both physically based and ad hoc shading models. Our appearance model has been integrated into the production pipeline at the Walt Disney Animation Studios and has been used in the production of the animated feature film Tangled.

Finally, we present a practical appearance model for rendering cloth fabrics. Our model is based on extensive Bidirectional Reflectance Distribution Function (BRDF) measurements of several cloth samples and threads. Based on these measurements we present a novel analytical BRDF model for threads. Using this model and statistical tangent distribution of cloth threads inside a weaving pattern we can reproduce the appearance of wide range of cloth fabrics. We also introduce a novel shadowing and masking term for cloth fabrics which is important in grazing angle viewing and lighting.

1

Introduction

OVER the centuries, mankind has been fascinated by the beauty and complexity of natural phenomena. On one hand, scientists have been pursuing a better understanding of the intricacy and harmony of nature; on the other hand, artists have been trying to replicate the beauty of natural phenomena by mimicking their appearances using lines, paint and forms. Computer graphics serves as an effective bridge between these two seemingly irreconcilable disciplines.

One of the main goals of computer graphics is to synthesize digital images that present a visual depiction of the real world. One particularly important branch of computer graphics is appearance modeling, which focuses on reproducing a specific desired appearance. The underlying reference appearance can be a photograph or even an art-reference which itself is a translation of a natural phenomenon by the mind of an artist. The goal of appearance modeling is to reproduce important visual features of a given reference appearance as closely as possible.

In order to produce realistic appearances we need to understand the manner in which light interacts with matter. Without a robust physically based foundation, the final synthesized appearance will look artificial and unrealistic. In addition, in order to reproduce a given appearance, we need to be able to control the output and adjust its visual features. For tailoring the final result we need intuitive control parameters with direct impact on visual features of the output. Having a physically based model with adjustable control parameters is essential for appearance modeling. Without a physical basis or adjustable controls, reproducing a desired appearance would be extremely challenging.

In this dissertation we focus on the appearances formed by the following microstructures: microspheres, microcylinders, and interwoven microcylinders. Many natural phenomena with various exotic appearances are formed by the interaction of light with these simple geometric shapes. For example, light interacting with spherical water drops produces the color variations in different types of rainbows, glories, and coronas. All of the variations in human hair appearance can be described by the way light interacts with individual cylindrical hair strands. Cloth fabrics are another interesting example that exhibit a wide range of different appearances which is due to the interaction of light with a mesh of interwoven cylindrical yarns.

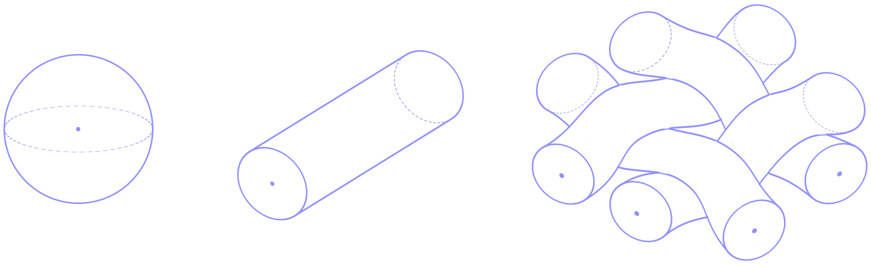


Figure 1.1: In this dissertation we focus on the appearances caused by the following microstructures: (left) microspheres, (middle) microcylinders, and (right) interwoven microcylinders.

The abovementioned microstructures have similar geometric shapes, therefore we encounter similar derivations when describing the physics of their behaviors. However, the variations in their physical properties cause a wide range of complex and fascinating appearances. These properties include both the material properties (e.g. index of refraction and absorption coefficients) and their structure (e.g. size, curvature, and weaving pattern). Theoretically, it is possible to reproduce any appearance through a rigorous, physically based simulation of these properties. However, we would need to acquire the exact value of these physical properties in order to reproduce a desired appearance. Besides, for adjusting the final appearance we would need to work with unintuitive control parameters which can be a tedious process.

We have access to a rich repertoire of prior research on the appearance modeling of rainbows, human hair, and cloth fabrics. However, as we will see in the following paragraphs, these implemented models either fail to reproduce specific appearances or they do so at the high price of profuse and time-consuming parameter tweaking. The goal of this dissertation, first and foremost, is to present novel practical appearance models that do not suffer from these shortcomings. These novel models have robust physical basis and incorporate intuitive parameters for controlling the final appearance. Our shading models facilitate the reproduction of a wider range of desired appearances for rendering rainbows, hair fibers, and cloth fabrics.

Figure 1.2: In this dissertation we focus on the appearances of rainbows, hair, and cloth.

Our first appearance model is formulated for rendering rainbows, which are caused by the interaction of light and water drops. These simple microstructures are responsible for different variations in the appearance of rainbows, one of the most beautiful displays of nature [Gre90; Min93; LL01]. Primary, secondary, and multiple supernumerary bows (See Figure 1.3 left and middle) are the result of light interacting with small spherical water drops. Twinned rainbows, however, are believed to appear due to the water drops that form imperfect spheres. Non-spherical water drops form when their sizes increase and the air resistance overcomes the surface tension of the drops. Supposedly, this deformation changes the light scattering behavior of the water drops and causes the formation of twinned rainbows (which have a pair of primary bows and a single secondary bow as seen in Figure 1.3 right). However, there has been no solid theory to confirm this hypothesis.

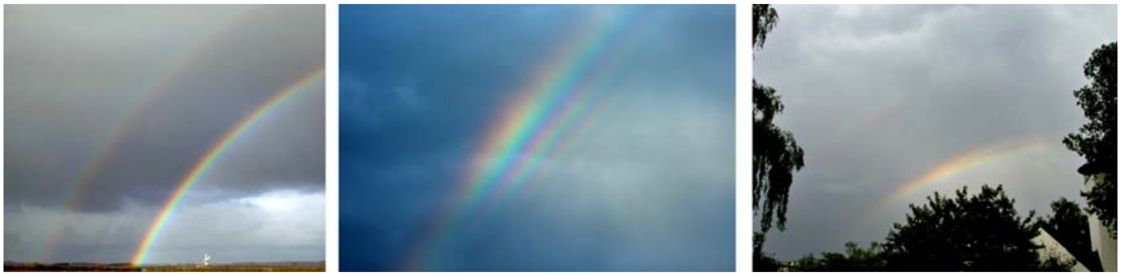


Figure 1.3: Different types of rainbows: (left) a double rainbow consisting of a bright primary bow and a dimmer secondary bow with flipped colors, (middle) multiple supernumerary bows, (right) a rare twinned rainbow.

The study of rainbows can be traced back more than two thousand years [LF01]. Ray optics can explain the variation of color in the appearance of rainbows as the result of dispersion of light as it passes through water drops. However, due to the small size of water drop particles, we have to take into account the wave nature of light in order to correctly simulate other phenomena such as polarization, interference, and diffraction.

There have been some approximation methods based on ray optics for modeling the appearance of rainbows. Musgrave [Mus89] follows Descartes' model from a classical geometrical optics perspective. Whereas, Frisvad and colleagues [FCF07] present a real-time simulation using Aristotle's rainbow formation theory based on reflections in clouds. Since, these models are developed based on ray optics, they fail to reproduce the appearance components that are related to the wave nature of light (e.g. supernumerary arcs).

Lorenz-Mie theory [Lor90; Mie08] provides an exact solution for scattering by spherical particles in non-absorbing media. However, this theory as well as other approximations of this model (e.g. Airy theory [Air38]), fails to predict the light scattering behavior of large non-spherical water drops. Therefore, Lorenz-Mie theory cannot reproduce the appearance of twinned rainbows.

The standard brute force method for simulating the light scattering from non-spherical water drops is the Finite Difference Time Domain method [Yee66; Taf98]. This method simulates Maxwell's Equations on a voxelized grid of the water drop particle. These methods are computationally expensive and impractical for three dimensional grids.

We have implemented a ray tracing framework to compute the light scattering profile of physically based water drops, based on which we have successfully explained and reproduced the appearance of twinned rainbows for the first time. We simulate different optical phenomena that are responsible in the appearance of rainbows. We present an accurate simulation of dispersion and interference of light as well as an approximation for the diffraction phenomenon. The results of our model match the results of the Lorenz-Mie theory for spherical water drops. However, our model can also predict the light scattering behavior of non-spherical water drops. Our approach can be generalized to other atmospheric effects where scattering particles have non-spherical shapes (e.g. Halos caused by ice crystals) where Lorenz-Mie theory is not applicable.

Due to the simple structure of spherical water drops the only physical property that affects their appearance is their size. We show that by varying the size of spherical water drops one can reproduce the appearance of wide range of rainbows including the primary and secondary bows, supernumerary bows, multiple supernumerary bows and even fogbows. We also show that in the case of non-spherical water drops, the inclination of the sun impacts the appearance of rainbows as well. Therefore, non-spherical water drops have a wider range of appearances. By taking into account the physically based shape of water drops and the sun's inclination, we have successfully explained and reproduced the appearance of twinned rainbows for the first time.

The second appearance model that we present is derived for rendering human hair. The variations in the appearance of hair can be captured by modeling the light interactions with individual hair fibers. Each hair fiber can be approximated by a transparent microcylinder. In the azimuthal direction microcylinders show similar light scattering behavior as microspheres. However, in the longitudinal direction they behave drastically different. This extra geometric dimension (which results in an anisotropic behavior) as well as variations in hair color and structure adds more complexity to the appearance of hair. Therefore, we need more control over their appearances in order to successfully reproduce a given appearance. For examples of different hair types refer to Figure 1.4.

Hair rendering is particularly important in motion pictures and the gaming industry. This is due to the fact that most computer generated characters carry hair or fur on their bodies. In addition, our eyes are very sensitive to the appearance of hair and we are sensitive to inaccuracies in its appearance. In fact, hair appearance has been shown to be one of the most important features of avatar personalization [DWYW09]. In these creative environments, having intuitive control parameters for appearance models becomes extremely important since art directors usually request specific changes regarding the produced appearances. Therefore, it is crucial for the artists to be able to readily fine-tune the final appearance.

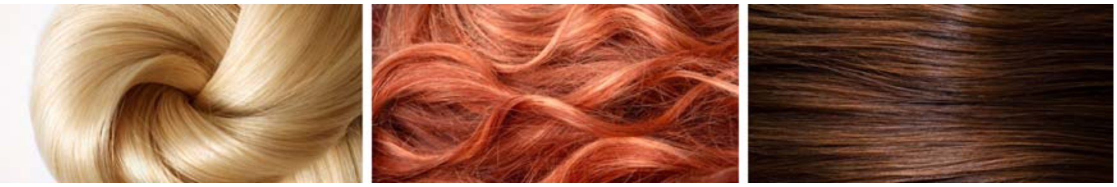


Figure 1.4: Different types of human hair: (left) blonde hair, (middle) red hair, (right) dark brown hair.

Hair appearance modeling has been an active area of research for more than two decades. Many studies have focused on physically based hair rendering both in the case of single scattering [KK89; Kim02; MJC$^+$03] and multiple scattering [MM06; ZW06; MWM08; ZYWK08]. Single scattering of light is responsible for the more noticeable features of hair appearance; For example primary and secondary highlight are the result of light reflection off of the outer and inner surface of the hair fibers respectively. The multiple scattering is responsible for the overall color of a volume of hair and is particularly important for light colored hair. Without multiple scattering, an assembly of blonde hairs will appear dark. Mentioned physically based models provide relatively accurate simulation of light inside individual hair strands and between an assembly of hair fibers based on the physical properties of hair strands (e.g. absorption coefficient, and index of refraction). However it remains challenging for the artists to directly benefit from these results, mainly because matching a desired appearance by tweaking the physically based parameters is a time-consuming and tedious task [ZRL$^+$09; BPvdP$^+$09].

There has been much effort to estimate the values of those physically based parameters by analyzing a single photograph [ZRL$^+$09; BPvdP$^+$09]. All of these methods enable artists to render hair with an appearance similar to a photographic reference. However, they fail to provide artists with any controls for further adjustments.

Because of the lack of suitable controls in physically based models, most production work in movies and games has used ad hoc shading models [AGL$^+$00; AGP$^+$02; AGP$^+$01; Neu04]. These shading models have more intuitive control parameters and are known to be more controllable. However, ad hoc shading models fail to capture the details of light scattering inside the hair volume and often produce inconsistent results under different lighting conditions.

We present a novel hair appearance model that provides intuitive artist control while maintaining the realistic appearance of the final result. Our appearance model is based on the light scattering behavior of real hair but it incorporates artist defined control parameters to control important visual features of the hair appearance. Through a user study we show that our model is easier to use compared to existing physically based and ad hoc models. Our hair shading model has been integrated into the production pipeline at the Walt Disney Animation Studios and have been used in the production of the animated feature film Tangled.

The last appearance model that we discuss in this dissertation is a practical appearance model for rendering cloth. Cloth fabrics are made out of a mesh of interwoven threads. Threads are composed of smaller fibers but we approximate each thread as a microcylinder with similar light scattering behaviors as hair strands. We justify this approximation by matching our analytical thread BRDF against measured data. Variations in the structure of threads as well as different weaving patterns add an extra level of complexity to the appearance of cloth fabrics compared to hair strands. Therefore we need more control over the appearance of cloth fabrics in order to be able to reproduce the variations in their appearance.

Cloth appearance is very important in computer graphics and in particular in movies and games. Cloth fabrics can be found in most computer graphics scenes in forms of clothing, furniture, and decoration. Cloth rendering is challenging due to the variations and complexity of fiber types, thread structures, and different weaving patterns. Therefore, it is crucial to have shading models with intuitive control parameters which enable us reproduce the complex appearance of cloth fabrics.

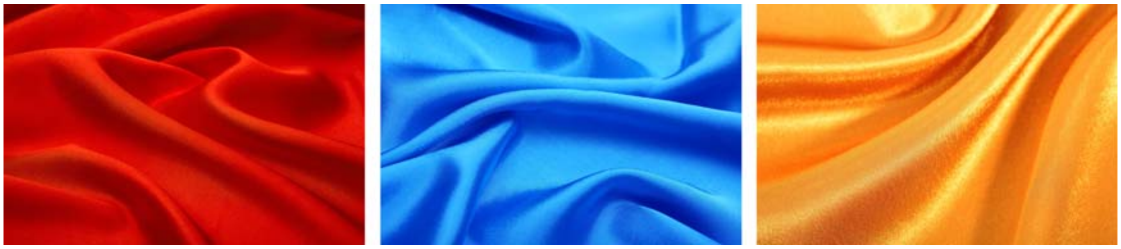


Figure 1.5: Different types of cloth fabrics: (left) red linen plain, (middle) blue silk satin, (right) yellow polyester charmeuse.

Cloth appearance modeling has been an active area of research for more than 25 years. The earliest approaches as well as more recent work are based on empirical shading models [Wei86; DLHS01; GD04]. These models are not concerned with the physical accuracy of the their results and fail to reproduce the complexity seen in real cloth fabrics.

Virtual goniometric methods can capture complex behavior of cloth fabrics through a statistical ray tracing of a geometric model of a patch of cloth fabric [WAT92; VKKK97; DC04]. However, these models require a 3D reconstruction of the fabric at the microstructure level which can be a tedious task. Another method for replicating cloth appearance is to use microfacet-based shading models. Microfacet models have been used by Ashikhmin et al. to model satin and velvet [APS00]. Adabala et al. continued this work by including support for weave patterns [AMTF03c]. Wang et al. [WZT$^+$08] introduced their own microfacet-based BRDF for modeling spatially-varying anisotropic reflectance using data captured from a single view. While microfacet models can be effective at capturing a complex appearance, these models are difficult to control as they depend strongly on the right microfacet normal distribution function. Formulating the correct normal distribution function for the microfacets can be a challenging task.

Another approach for simulating cloth is based on physically based simulation of the structure of the cloth fabric. Yasuda et al. [YYTI92] modeled the gloss seen in cloth by accounting for the internal structure. Others presented different volumetric approaches in order to render high quality renderings of cloth [XCL$^+$01; CLZ$^+$03; JAM$^+$10]. These models require a high quality representation of the cloth microstructures and therefore are less practical. Zhao et al. [ZJMB11] presented a CT scanning method for acquiring an accurate representation of the cloth fabrics.

Irawan et al. developed a comprehensive model for reproducing both the small-scale (BTF) and large-scale (BRDF) appearance of woven cotton cloth [IM06; Ira08]. This model is current state of the art and capable of reproducing a wide range of appearances. The basis of the model is a complex empirical model for light interacting with a cloth thread. This model is evaluated numerically to fit with measured data. The numerical fit is rather costly and to reduce the number of parameters in the model only a specific set of cloth tangents can be accounted for which in turn limits the accuracy of the model. Furthermore, this model ignores shadowing and masking between different threads of cloth, which limits the accuracy at grazing angles. Finally, the complexity of the model makes it difficult to control in order to achieve a specific appearance. We present a practical appearance model for rendering cloth fabrics. Our model is based on extensive measurements of the scattered light by cloth samples as well as individual threads and incorporates intuitive control parameters. We take into account the shadowing and masking by the individual threads in the cloth which is important for grazing angle viewing and lighting. Our model is easy to control and it can reproduce a wide range of cloth types including linen, silk, polyester, and velvet with varying weaving patterns. We can also, model the appearance of novel physically plausible cloth fabrics. Finally, we demonstrate that our appearance model can run in real-time on a GPU.

1.1 Summary of Original Contributions

The contributions of this dissertation are related to the appearance modeling of three different microstructures: microspheres, microcylinders, and interwoven microcylinders. These three microstructures are involved in producing the appearance of rainbows, human hair, and cloth fabrics respectively. We outline the major contributions of our work below.

1.1.1 Light Interactions with Microspheres

We present a comprehensive study on the appearance of rainbows. We explain the light scattering events that cause the formation of rainbows, and develop a novel ray tracing simulation approach that can handle spherical and non-spherical water drops. Our model accounts for the full spectrum of optical effects including dispersion, interference, polarization, and an efficient approximation for diffraction. We can reproduce primary, secondary and multiple supernumerary bows as well as supernumerary arcs, redbows and fogbows. We also show that by considering the physically based shape of water drops, we can explain, for the first time, the effect of the angle of incidence of the light and some unusual appearances like twinned rainbows.

1.1.2 Light Interactions with Microcylinders

We describe the requirements of appearance modeling systems in creative environments. According to those requirements, we define main characteristics of a desired appearance model. These characteristics include realistic appearance, intuitive controls, decoupled behavior and the capability of extrapolating a plausible appearance beyond physically based ranges. We show that ad hoc shading models fail to produce rich appearances while physically based hair shading models fail to provide the controllability requirements. We then present a novel User-Centered Design approach for creating a controllable hair appearance model from existing physically based models. Through a user study we show that our model is easier to control and can be used to match desired appearances faster compared to both physically based and ad hoc shading models.

1.1.3 Light Interactions with Interwoven Microcylinders

We present a novel framework that can reproduce a wide range of cloth appearances using analytical approximations of cloth threads and statistical geometric properties of the weaving patterns. Our model is based on extensive measurements of the scattered light by cloth samples as well as individual threads. Based on these measurements, we have developed a novel analytical model for light scattering by cloth threads. Our far-field appearance model can reproduce a wide range of complex appearances which have not been fully addressed in computer graphics literature before. We also introduce a novel shadowing and masking term for cloth fabrics. We validate our model by comparing predicted and measured values and show how we can use the model to recover parameters for different cloth samples.

1.2 Organization of the Dissertation

This dissertation is divided into six chapters. In Chapter 2, we provide some theoretical background on the physics of light and how it interacts with matter. We describe existing optical models and point out different phenomena that can be described by each model. We subsequently explain different interactions of light with matter that are relevant to specular microstructures. We conclude that section by describing the way light is being simulated in computer graphics.

In Chapter 3, we focus on the light interactions with microsphers and present our framework for modeling the appearance of rainbows. We show that we can control the final appearance of rainbows by adjusting the water drop size for spherical water drops. In the case of non-spherical water drops, both the size and the inclination of the sun affect the appearance of rainbows. By considering the physically based geometry of large water drops we present a simulation of twinned rainbows for the first time.

Chapter 4 focuses on the light scattering behavior of microcylinders. In that chapter we present a highly controllable hair appearance model. We show that our model is easier to control and enables users to match a given reference in less time compared to both physically based and ad hoc shading models. We explain how our hair appearance model has been successfully used in a production environment.

We then focus on interwoven microcylinders and introduce a practical appearance model for cloth fabrics in Chapter 5. We present a novel analytical model for light scattering by individual threads. We then present a novel cloth appearance model by combining our thread BRDF model with statistical tangent distribution of threads. We show that our model can reproduce a wide range of cloth appearances through intuitive and physically meaningful control parameters.

Finally, we conclude this dissertation in Chapter 6 by summarizing our contributions and discussing possible avenues for future work.

At the end we present supplemental materials for our work in five appendices.

Appendix A provides the necessary information for reproducing our results for rendering rainbows. In Appendix B, we present artist friendly terms for the context of hair rendering. We provide an efficient implementation of the Dual Scattering model in RenderMan Shading Language (RSL) in Appendix C. In Appendix D, we present the data regarding the user study of our hair shading model. In the end, we provide the tangent distribution curves used in the results of our cloth appearance model in Appendix E.

2

Nature and Behavior of Light

LIGHT interactions with matter determine the appearance of natural phenomena and materials. In order to understand these interactions, however, we need to first understand the physical properties of light. The nature of light can be described by a series of progressively complete optical models. As shown in Figure 2.1, as these models become more sophisticated, they can describe broader optical phenomena [ST07].

In this chapter we start by exploring different optical models in Section 2.1 in order to better understand the nature of light. We then go over different light interactions with matter which are relevant to specular microstructures in Section 2.2. Finally, in Section 2.3, we present important quantities, definitions and formula in computer graphics that are used for simulating light and its interactions with the environment.

2.1 Different Optical Models

Light is an electromagnetic wave and travels in the space in the form of a pair of coupled vector waves: an electric wave $\vec{E}$ and a magnetic wave $\vec{H}$, which are

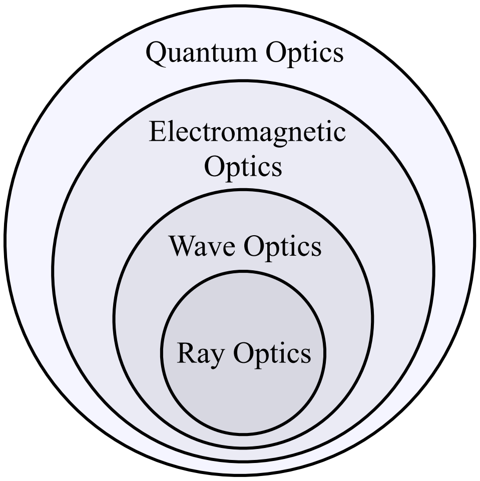


Figure 2.1: Different optical models for describing the nature of light. The theory of quantum optics is the most complete optical model to date. It subsumes electromagnetic optics, which in turn, subsumes wave optics. Ray optics is the simplest and the most widely used model in computer graphics.

perpendicular to each other and to the wave propagation direction $\vec{k}$. This complete model is called the *electromagnetic optics*. *Wave optics* is an approximation of the electromagnetic optics where we assume light as a scalar wave function. If light is interacting with objects much larger than its wavelength we can ignore the wave nature of light and assume it propagates on a straight line. This simplified model is called *ray optics*. Quantum physics have introduced some phenomena that cannot be explained by any of the above-mentioned optical models. These phenomena can be described in terms of *quantum optics*.

In the following sections, we go over these optical models from simplest to the most complete. This order, not surprisingly, reflects the historical development of these optical models.

2.1.1 Ray Optics

Ray optics, also known as the *geometric optics*, is the simplest theory concocted for describing the nature of light. This model views light as a stream of particles of energy traveling through space on straight lines called *rays*. It can describe the most common light interactions that we encounter in our daily lives: namely reflection and refraction. Ray optics can be derived from wave optics by taking the limit in which the wavelength of light becomes infinitesimally small. However, in practice this model can describe a wide range of optical phenomena for light rays with varying wavelengths. This model is the most commonly used model of light in computer graphics and is the main focus of this dissertation. However, in many cases, we incorporate information regarding the wave nature of light into our ray based models to simulate phenomena that cannot be described by ray optics.

Ray optics assumes that in a homogenous medium light travels in a straight line (i.e ray . In this model, any optical medium has a property called *index of refraction*. This quantity is the ratio of the speed of light in a vacuum c relative to that in the considered medium $c^{'}$:

$$\eta = c/c^{'}. \tag{2.1}$$

The refractive index of different media is a value greater than one. Table 2.1 shows a list of index of refraction values for various materials.

The last assumption of ray optics is called *Fermat's Principle* which states that light rays follow the path that can be traversed in the least time.

As we will describe in Section 2.2.1, these assumptions enable us to derive simple rules that determine the behavior of light rays through optical media. As long as the light is interacting with objects much larger than its wavelength, ray optics successfully predicts and describes the behavior of light. When the objects have sizes comparable to the wavelength of the light, we need to consider the wave nature of light which is the subject of next section.

2.1.2 Wave Optics

Wave optics, also known as *physical optics*, is an intermediate approximation between ray optics and electromagnetic optics. This model can describe all light

Table 2.1: List of index of refraction η for different media. The values are taken from [Hec01; Rob94; Taj04; MM98; Sar06; GK95]

Medium	**Index of Refraction η**
Vacuum	1.0
Air	1.00029
Ice	1.31
Water	1.333
Silk Fibers	1.34 to 1.35
Cotton Fibers	1.4
Polyester Fibers	1.456 to 1.539
Glass	1.52
Hair Strands	1.55
Diamond	2.417
Gallium Phosphide	3.50
Silicon	3.723 to 6.548

interactions handled by ray optics as well as more phenomena like interference and diffraction. Wave optics considers light as a scalar wave function u of position x and time t that satisfies *the wave equation*:

$$c^2 \ \bigtriangledown^2 u = \frac{\partial^2 u}{\partial t^2}, \tag{2.2}$$

where $\bigtriangledown^2$ is the Laplacian operator. The wave equation can be derived from Maxwell's equations which forms the foundation of classical optics. Since the wave equation is linear, the sum of any two waves u_1 and u_2 will be an optical wave itself:

$$u(x, t) = u_1(x, t) + u_2(x, t). \tag{2.3}$$

This relation is known as the *principle of superposition.* The most widely used form of waves are *harmonic waves*, which have the profile of sine or cosine curves. This type of wave function has special significance since any arbitrary wave function $u(x, t)$ can be synthesized by a superposition of harmonic waves using a Fourier

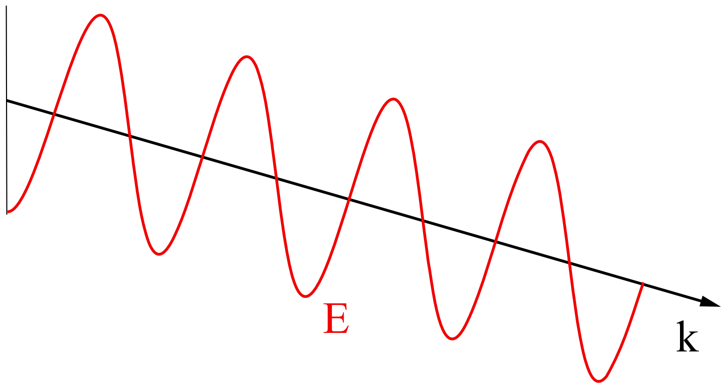


Figure 2.2: Electric wave E is a scalar function that oscillates perpendicular to the direction of its propagation $\vec{k}$.

transform [ST07].

2.1.3 Electromagnetic Optics

Electromagnetic Optics is the most complete optical model in the area of classical optics. It is based on Maxwell's equations and describes light as energy that propagates in the form of a paired couple of vector waves: an electric wave $\vec{E}$ and a magnetic wave $\vec{H}$. These two wave vectors are perpendicular to the direction of light $\vec{k}$ (See Figure 2.3). Wave optics is an approximation of electromagnetic optics since it treats light as a single scalar function. Effects such as polarization of light can be described using electromagnetic optics, but not wave optics.

In this dissertation we do not simulate the complete wave optics model which can be done using grid based numerical methods such as finite difference time domain method [Yee66; Taf98]. However, in many occasions, we incorporate wave information with light rays in order to simulate the wave nature of light. In Chapter 3, we take into account the phase of light in order to properly simulate interference. In that Chapter, we consider the polarization of light by incorporating electromagnetic information within each light ray.

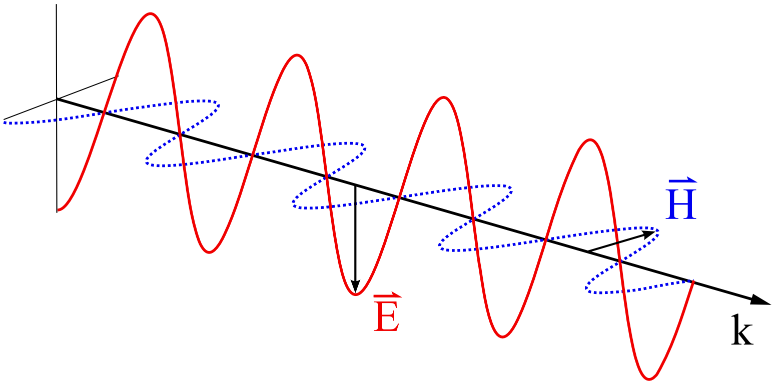


Figure 2.3: Electromagnetic wave is a paird couple of vector waves: an electric wave $\vec{E}$ and a magnetic wave $\vec{H}$. These two wave vectors are perpendicular to the direction of wave vector $\vec{k}$.

2.1.4 Quantum Optics

Light is a phenomenon that has a dual behavior: sometimes it behaves as if it is a wave and sometimes as if it is a stream of particles. Quantum optics is the fundamental theory of light that can describe this dual behavior. According to quantum optics, light is an electromagnetic wave and also a stream of photons at the same time. A photon can be considered as a wave packet (a wave function defined over a small region of space) [Gla95]. Unlike any other particles, photons cannot be seen directly. We can observe them by detecting the effect they have on the environment as they get created and destructed [Hec01]. In Quantum Optics, we are only able to calculate the probability that a photon will hit a detector, without having a satisfying explanation of how it actually happens [Fey88].

Effects such as phosphorescence and fluorescence can be explained only by quantum optics. Phosphorescence occurs when light gets absorbed by a substance and gets emitted at a different time. Fluorescence happens when absorbed and emitted light have different wavelengths.

Quantom optics is considered to be able to explain virtually all known optical phenomena [ST07]. However, quantum optics is too detailed for digital image synthesis and is not commonly used in computer graphics [DBB06]. In this dissertation we do not consider quantum optics and instead limit ourselves to classical optics.

2.2 Light Interactions with Matter

There are different ways that light interacts with matter. These interactions determine the appearance of different phenomena and materials. The study of behavior of light dates back to ancient Greek philosophers and mathematicians. The Greek word “optikos” means “sight” or “vision” [BW99].

In this section we will go over the light interactions that are relevant to the appearance of specular microstructures.

2.2.1 Reflection and Refraction

When light propagates from a medium to another, *reflection* and *refraction* of light may occur at the boundary of the two media (See Figure 2.4). These two phenomena are the most common light interactions that we see in our daily lives. Reflection and refraction can be explained by all optical models mentioned in Section 2.1. In this section we consider light as a ray and follow the rules of ray optics throughout the section.

Reflection is the change in the direction of light at the interface between two different media so that it returns to the originating medium (see Figure 2.4). One important property of the reflected ray is that it stays within the *incidence plane.* The incidence plane is the plane that contains the light ray as well as the normal of the interface at the point of incidence.

Based on Fermat’s principle, the reflection direction should lead to the path with the shortest time. Since the light has remained in the same medium, the path of minimum time would be the path of minimum distance (this is also known as *Hero’s principle*). Let us assume the light ray starts from point A and reaches

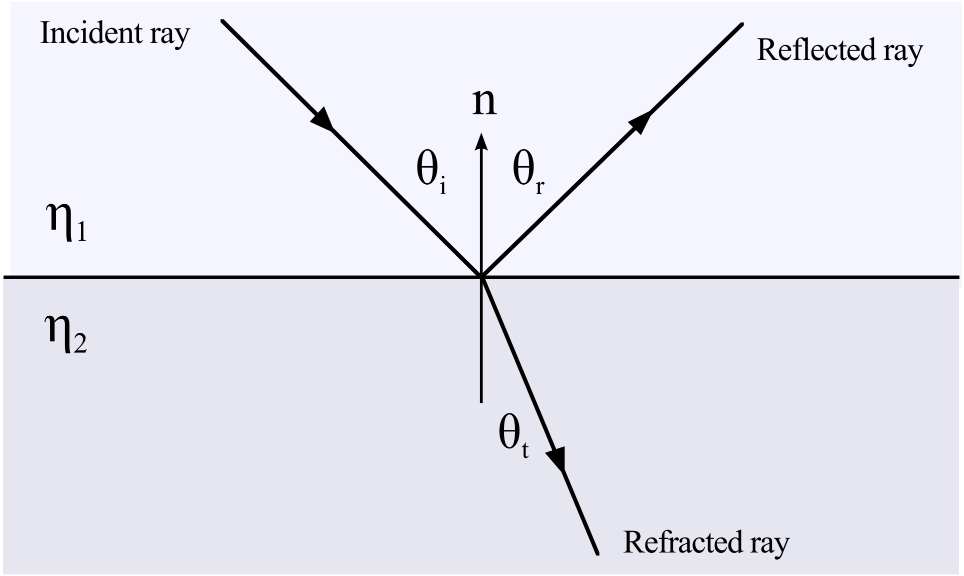


Figure 2.4: The geometry of the reflection and refraction of light: The incident ray has angle θ_i with respect to the surface normal $\vec{n}$. the reflected and refracted rays have angle θ_r and θ_t respectively.

point B as shown in Figure 2.5. The distance that light has to travel is equal to the distance between point A and point $B^{'}$, the mirror reflection of point B.

In order to minimize the distance between point A and point B we need to find the minimum distance between A and $B^{'}$ which is a straight line. Therefore, the incidence angle θ_i and reflection angle θ_r should be equal. This is also known as the *law of reflection*:

$$\theta_i = \theta_r. \tag{2.4}$$

Refraction is the change in the speed and direction of light as it travels between two different media (see Figure 2.4). The refracted ray will also stay in the incidence plane. The refracted angle θ_r is dependent on the incidence angle θ_i and the refractive indices of both media η_1 and η_2 (See Figure 2.6). Fermat's principle states that light that gets refracted from point A to point B will follow the path with the shortest time. The time taken by the light ray to travel from A to B can

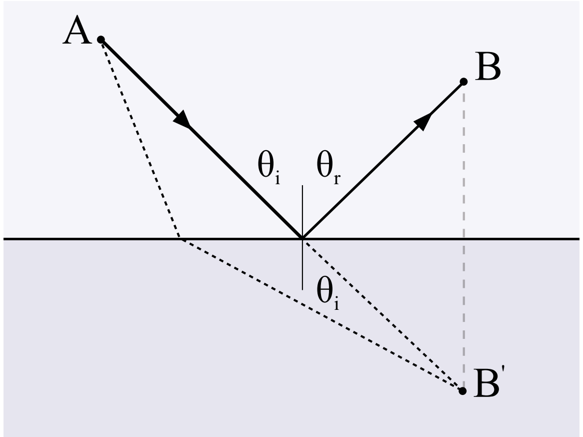


Figure 2.5: The geometry of light reflection as it goes from point A to point B. Using Fermat's principle, one can show that the reflected angle θ_r is equal to the incident angle θ_i.

be calculated as follows:

$$t(x) = \frac{AO}{v_1} + \frac{OB}{v_2} = \frac{\sqrt{h_1^2 + x^2}}{v_1} + \frac{\sqrt{h_2^2 + (d - x)^2}}{v_2} \tag{2.5}$$

where v_1 and v_2 represent the speed of light in medium 1 and 2 respectively.

We can minimize the time by setting its derivative with respect to x to zero:

$$\frac{dt}{dx} = \frac{x}{v_1\sqrt{h_1^2 + x^2}} - \frac{d - x}{v_2\sqrt{h_2^2 + (d - x)^2}} = 0 \tag{2.6}$$

Considering that the speed of light in any medium with refractive index η is c/η, we conclude the following relation which is also known as the *Snell's law*:

$$\frac{\sin\theta_i}{\sin\theta_t} = \frac{\eta_2}{\eta_1}. \tag{2.7}$$

For more details on deriving Snell's law refer to [BP91].

Based on Snell's law when light gets refracted into a medium with lower index of refraction it gets bent further from the surface normal. This indicates that at some angle, called the *critical angle*, refracted light direction lies exactly on the interface. All greater incident angles gets reflected back and undergo a *total*

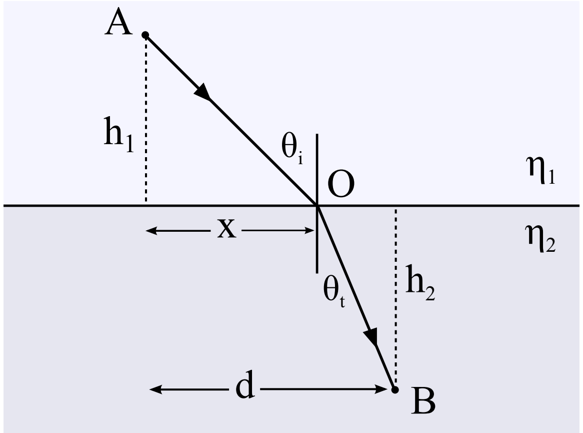


Figure 2.6: The geometry of light refraction as it goes from point A to point B. Using Fermat's principle, one can show that the refracted angle θ_t and the incident angle θ_i follow Snell's law.

internal reflection. The critical angle θ_c can be derived directly from Snell's law by setting θ_i to $\pi/2$:

$$\theta_c = \sin^{-1}(\frac{\eta_1}{\eta_2}) \tag{2.8}$$

When light reaches the boundary of a transparent material, it gets partially reflected and refracted. The fraction of the incident power of light that is reflected or refracted can be derived using *Fresnel Equations*:

$$\rho_{\|} = \frac{\eta_2 \cos\theta_i - \eta_1 \cos\theta_t}{\eta_2 \cos\theta_i + \eta_1 \cos\theta_t} \tag{2.9}$$

$$\rho_{\perp} = \frac{\eta_1 \cos\theta_i - \eta_2 \cos\theta_t}{\eta_1 \cos\theta_i + \eta_2 \cos\theta_t} \tag{2.10}$$

where $\rho_{\|}$ ($\rho_{\perp}$) is the amplitude reflection coefficient for the case where the electric wave vector $\vec{E}$ is parallel (perpendicular) to the incidence plane . This property of light is related to the polarization of light. Refer to Section 2.2.4 for more details. Although, Fresnel equations are defined for polarized electromagnetic waves, we can use their results when we are focusing on ray optics by considering unpolarized light. In such cases we can compute the fraction of reflected F_r and refracted F_t

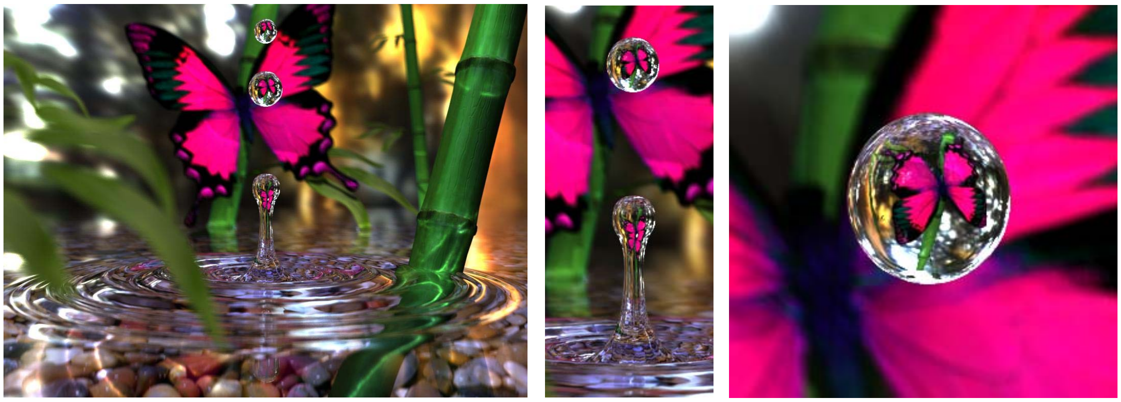


Figure 2.7: A computer generated image that uses Fresnel's equations in order to simulate reflection and refraction of light.

light based on the following formula:

$$F_r = \frac{1}{2}(\rho_{\parallel}^2 + \rho_{\perp}^2) \tag{2.11}$$

$$F_t = 1 - F_r \tag{2.12}$$

Correct simulation of reflection and refraction of light plays an important role in modeling the appearance of most phenomena and materials. Figure 2.7 shows a rendered image using Fresnel equations for computing the amount of reflections and refractions. In Chapter 3 we show that rainbows form due to the refraction and internal reflection of light by water drops.

In Chapter 4 we show that different highlights in the appearance of hair are caused by reflection and refraction by the hair surface. Finally, in Chapter 5, we show that important visual features of many cloth fabrics can be modeled by considering the tangents of the fibers and computing the reflection angle.

2.2.1.1 Dispersion

If different wavelengths of light travel with different speeds inside a medium (which means that the index of refraction in that medium is a function of wavelength), they will get separated after getting refracted. This phenomena is said to be *dispersion* and the materials that show this behavior in the visible spectrum range are called *dispersive*. In Chapter 3 we use wavelength dependent index of refractions to simulate the color variations present in rainbows (See Figure 2.8).

Figure 2.8: Effect of different optical phenomena on the appearance of rainbows: Top) Dispersion. Middle) Dispersion plus interference. Bottom) Dispersion plus interference plus diffraction.

2.2.2 Interference

Interference results from the superposition of two light waves. Therefore interference can be thought of as the interaction of light with light. This phenomenon usually happens when light can travel from a point to another by two or more paths of different lengths. Interference cannot be explained with ray optics since it relies on the phase difference of the superposed light waves [ST07].

Consider two interfering waves with the same wavelength. If the phase difference of the two light waves is zero, they will have *constructive interference* which will increase the amplitude of the resultant wave. If the phase difference is π then they will experience *destructive interference* which will reduce the amplitude of the resultant wave (potentially to zero). The general case lies between these two extreme cases. See Figure 2.9 for a visualization of these three cases.

Interference can be observed on the surface of thin films such as soap bubbles (See Figure 2.11) and oil film on water. The color variation on the surface of thin films is caused by interference of internally and externally reflected light waves. The

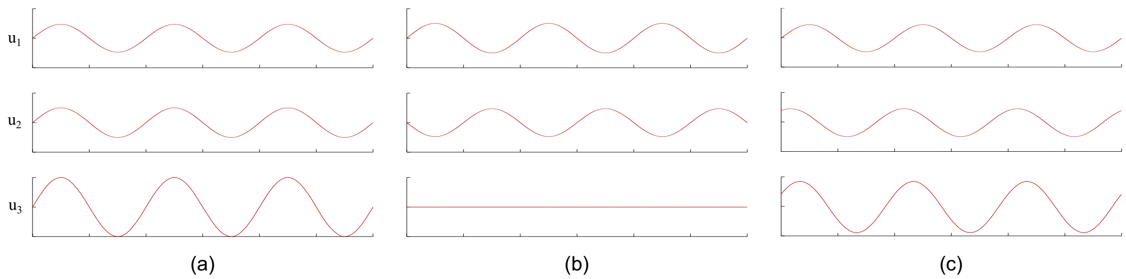


Figure 2.9: Different types of interference: Right) Constructive. Middle) Destructive. Right) the general case.

thickness of the film d, which should be comparable to the wavelength of light, influences the phase difference of the two paths of light as shown in Figure 2.10.

Full simulation of wave optics have been addressed in computer graphics [Yee66; Taf98]. These methods require a high density voxelized representation of space and are computationally expensive. We incorporate the wave information (phase, and wavelength) in light rays and use it to simulate interference which is needed to simulate the supernumerary arcs of rainbows in Chapter 3. See Figure 2.8 to see the effect of interference on appearance of rainbows.

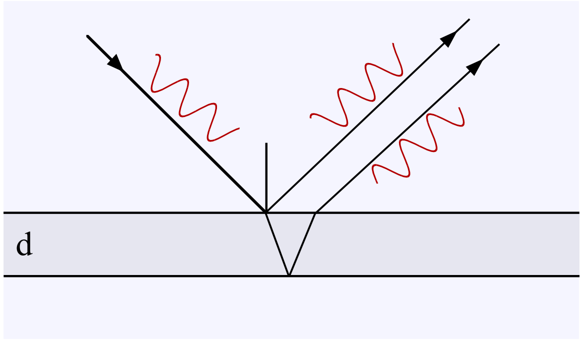


Figure 2.10: Thin film interference of light. The light gets reflected off the external and internal surfaces of a thin film. The reflected lights have a phase difference causing them to interfere.

2.2.2.1 Iridescence

Iridescence, also called *structural color*, is a surface property in which the observed hue of the surface depends on the viewing direction. The word iridescence is derived from the Greek word “iris” which means rainbow. However, the source of color variation in rainbows is due to dispersion while iridescence is caused by interference. Iridescence happens when some wavelengths get intensified through constructive interference and some wavelengths get attenuated through destructive interference. Probably, the most recognizable example of iridescence is the wings of Morpho butterflies with their vivid blue colors (Figure 2.11 middle). Other examples of iridescent surfaces are shells of many insects (see Figure 2.11 right), sea shells, and peacock feathers.

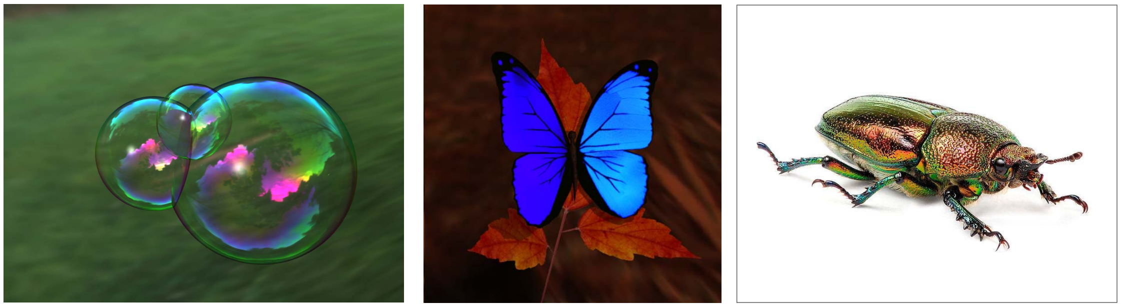


Figure 2.11: Examples of iridescent surfaces caused by interference: Left) A soap bubble. Middle) A Morpho butterfly. Right) A beetle.

2.2.3 Diffraction

Diffraction is another phenomena that is related to the wave nature of light. Similar to any other wave (e.g. sound waves, radio waves, and ripples on the surface of water), light waves get bent around small obstacles and small openings. This phenomena is called the diffraction of light [Gla95].

Based on *Huygens-Fresnel principle*, every point of a wave serves as a point source of spherically symmetric waves. The amplitude of the light wave at any given point is the superposition of all of these spherical waves [BW99]. Figure 2.12 shows the effect of diffraction as light hits a surface with a small hole or a sharp edge.

Although diffraction happens in nature frequently, we can ignore its effect in computer graphics unless the size of the objects are comparable to the wavelength of the light wave. Diffraction can be observed in natural phenomena, like rainbows and glories, as well as artificial objects, such as holograms and the back surface of compact disks.

In Chapter 3 we present a rough approximation for diffraction effect that happens around the rainbow angle. See Figure 2.8 for a visualization of this effect on the appearance of rainbows.

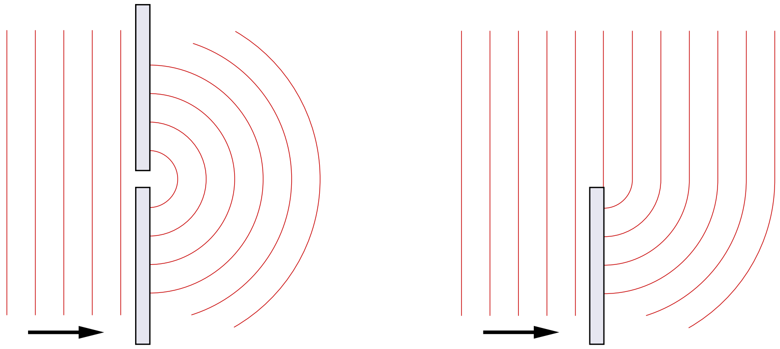


Figure 2.12: Diffraction of light as it hits a surface with a small hole or a sharp edge. The arrow shows the direction of the propagation direction of the wave. Every point of a wave serves as a point source of spherically symmetric waves.

2.2.4 Polarization

As we discussed in Section 2.1.3, electromagnetic waves consist of a pair of an electric wave vector $\vec{E}$ and a magnetic wave vector $\vec{H}$ (see Figure 2.3). The polarization is perpendicular to the wave's direction of travel $\vec{k}$. If the electric field stays in one direction, the electromagnetic wave has *linear polarization*. If the orientation of the electric wave changes over time, the electromagnetic wave will have *elliptical polarization* (Figure 2.13). The size of the ellipse represents the

the intensity of the light wave and the orientation and eccentricity of the ellipse represents the polarization of the light [ST07]. If the polarization state of different waves is not correlated then the light is *unpolarized*.

A common coordinate system for projecting the electromagnetic waves when they interact with a surface is the incidence plane. The component of the electric field parallel to this plane is called *p-polarized* and the component perpendicular to this plane is called *s-polarized*. Fresnel equations (Section 2.2.1) are defined based on these two components of the electric wave.

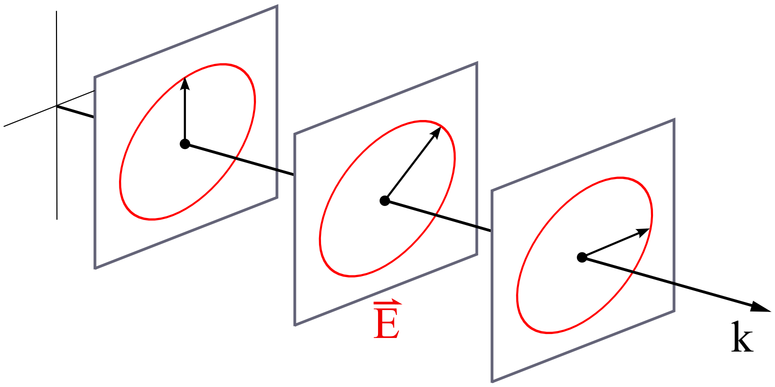


Figure 2.13: Polarization state of the light is perpendicular to the direction of the wave vector $\vec{k}$.

Rainbows are highly polarized phenomena and to simulate them we keep track of the polarization state of light as they interact with water drops (Chapter 3). For modeling the appearance of hair fibers (Chapter 4) and cloth fabrics (Chapter 5) we ignore the polarization state of light.

2.2.5 Absorption

When the light wave propagates inside a medium it loses a portion of its energy. This energy gets transformed into other forms of energy, like heat. This phenomena is known as *absorption* or *attenuation*. In many cases, absorption is the cause of the coloration of different objects. For example plant leaves appear green because they absorb the red and blue wavelengths of the light. This means that plants will appear very dark under red or blue light and that they cannot survive under green light.

In a homogeneous media, the amount of absorption of light as it travels through a distance of x can be derived from the *Beer-Lambert law*:

$$L_o = L_o e^{-\sigma_a x} \tag{2.13}$$

where L_o and L_i are outgoing and incoming radiance values (see Section 2.3.1.4) and σ_a is the absorption coefficient of the medium.

In Chapter 4 we show that absorption of light inside hair strands is the cause of variations in hair color. Most color variations in cloth fabrics are also due to the light absorption (Chapter 5). However, the effect of absorption on rainbow formation is negligible as we will show in Chapter 3.

2.3 Light in Computer Graphics

In this section we present the background material that is related to the simulation of light in computer graphics. Most of the algorithms used for digital image synthesis is based on quantities and derivation from the field of *radiometry*. We will go over the most important radiometric quantities and the relationship between them in the next section. Then we show the role of these terms and formulae in rendering algorithms.

2.3.1 Radiometry

Radiometry is a branch of optics that focuses on the measurement of electromagnetic radiation, including visible light. We devote this section to radiometry since it forms the basis of the derivations and algorithms that are widely used in computer graphics.

It is interesting to note that this field was built on the assumption of ray optics and therefore does not naturally handle electromagnetic properties of light (e.g. polarization). However, it has been evolved over the years with respect to Maxwell's equations and now has a solid physical basis [PH04].

In this section we will go over important radiometric terms which are relevant to computer graphics.

2.3.1.1 Flux

Flux, also known as *radiant power*, is the total amount of electromagnetic radiation flowing across a region of space per unit time. Its unit are watts (W) or joules per second (J/s) and is denoted by symbol Φ. Note that the total amount of flux for any closed surface around any electromagnetic source is the same.

2.3.1.2 Irradiance

Irradiance is the amount of incident flux per unit surface area. This quantity has unit of $W \cdot m^{-2}$ and is denoted by symbol E. We can calculate irradiance from the amount of incoming flux based on the following formula:

$$E = \frac{d\Phi}{dA}. \tag{2.14}$$

Irradiance is a surface property and can be measured at any point on the surface.

2.3.1.3 Intensity

Another important quantity is the *intensity* of the electromagnetic wave which is defined as the amount of flux per *solid angle*:

$$I = \frac{d\Phi}{d\vec{\omega}} \tag{2.15}$$

Solid angle is the extension of two dimensional angles on a plane to three dimensional angles on a sphere. It represents how large an object appears from the point of view of the center of the sphere. Solid angles are measured in *steradians* (sr). Intensity expresses the directional distribution of light for point light sources.

2.3.1.4 Radiance

The final and the most important radiometric term is *radiance*. This quantity is denoted by symbol L and is defined as flux density per unit area, per unit solid angle:

$$\begin{aligned} L &= \frac{d\Phi}{d\vec{\omega}\ dA^{\perp}} && (2.16) \\ &= \frac{d\Phi}{d\vec{\omega}^{\perp}\ dA}, && (2.17) \end{aligned}$$

where $A^{\perp}$ represents the projection of the differential surface area $d\ A$ onto a plane perpendicular to direction $d\vec{\omega}$, and $\omega^{\perp}$ is the projected solid angle with respect to the area dA. A more convenient formula for calculating radiance can be derived based on the cosine angle between surface normal $\vec{n}$ and the direction $\vec{\omega}$:

$$L = \frac{d\Phi}{d\vec{\omega}\ dA\ (\vec{\omega}.\vec{n})}. \tag{2.18}$$

Radiance describes how much light arrives from a very small cone around direction $d\vec{\omega}$ onto a very small area perpendicular to that direction. Radiance has the unit $W.sr^{-1}.m^{-2}$.

In computer graphics we are particularly interested in radiance values since it is the quantity that determines the color of the pixels in synthesized images. Another important property of radiance is that it remains constant along light rays. This property makes radiance the ideal quantity to be used in ray tracing.

In addition, radiance can be considered the most fundamental radiometry term since all other quantities can be computed from it. The following equation computes flux based on radiance by integrating both sides of equation 2.18 over a hemisphere of directions Ω and area A:

$$\Phi = \int_A \int_\Omega L(x, \vec{\omega})\ (\vec{n}.\vec{\omega})\ d\vec{\omega}\ dA(x). \tag{2.19}$$

Irradiance can also be derived from radiance based on equations 2.14 and 2.19:

$$E = \int_\Omega L(x, \vec{\omega})(\vec{n}.\vec{\omega})\ d\vec{\omega} \tag{2.20}$$

2.3.2 The BRDF

In computer graphics we often make the simplifying assumption that light hitting a surface at a specific point will get reflected at the same location. The *Bidirectional Reflectance Distribution Function* (BRDF) expresses the reflection profile from a point on the surface for any pair of viewing and lighting directions (see Figure 2.14). In terms of radiometric terms, the BRDF is the differential ratio between the reflected radiance at point x divided by the irradiance at that point:

$$f_r(x, \vec{\omega}, \vec{\omega}^{'}) = \frac{dL(x, \vec{\omega})}{dE(x, \vec{\omega}')} \tag{2.21}$$

$$= \frac{dL(x, \vec{\omega})}{L(x, \vec{\omega}')\ (\vec{n}.\vec{\omega}')\ d\vec{\omega}'}. \tag{2.22}$$

where $\vec{\omega}$ and $\vec{\omega}^{'}$ are viewing and lighting directions. As we will see shortly, their order does not matter.

Physically based BRDFs have the following properties:

- **Reciprocity:** For all pairs of directions $\vec{\omega}$ and $\vec{\omega}^{'}$ we will have:

  $$f_r(x, \vec{\omega}, \vec{\omega}^{'}) = f_r(x, \vec{\omega}^{'}, \vec{\omega}). \tag{2.23}$$

  This is known as *Helmholtz's law of reciprocity.*

- **Incident and Reflected Radiance Relation:** Reflected radiance can be computed from the BRDF and the incident radiance by integrating over all directions on a hemisphere Ω around point x:

$$L(x,\vec{\omega}) = \int_{\Omega} f_r(x,\vec{\omega},\vec{\omega}')\; L(x,\vec{\omega}')\; (\vec{n}.\vec{\omega}')\; d\vec{\omega}' \tag{2.24}$$

- **Energy Conservation:** The total amount of energy reflected from point x should be less than or equal to the incident energy arriving at that point. This means for all points x and for all direction $\vec{\omega}$ we will have:

$$\int_{\Omega} f_r(x,\vec{\omega},\vec{\omega}')\; (\vec{n}.\vec{\omega}')\; d\vec{\omega}' \leq 1. \tag{2.25}$$

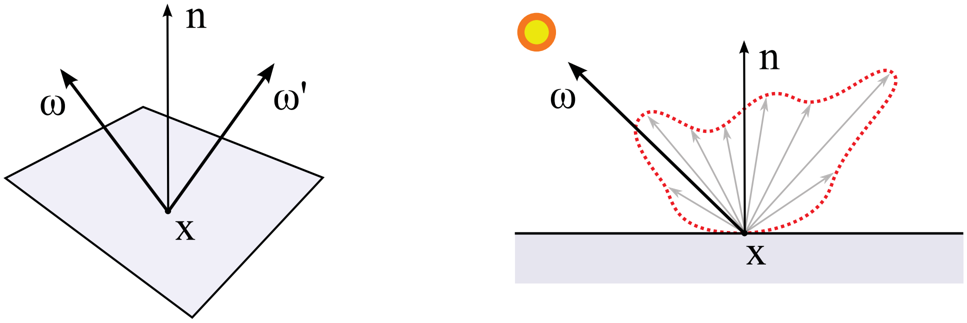


Figure 2.14: Left) The BRDF expresses the reflection profile from a point on the surface for any pair of viewing and lighting directions. Right) An example BRDF and its response to the incoming light direction $\vec{\omega}$.

The BRDF is a surface property and determines the observed appearance of different materials. In Chapter 4 we explore the behavior of hair BRDFs and in Chapter 5 we will investigate the BRDF of yarn fibers and present a novel BRDF for rendering cloth fabrics. When the appearance is defined for participating media (and not a surface), we use a different definition called the *phase function* (see Section 2.3.4).

The earliest and the most basic BRDF models include Lambertian [Lam60], Phong [Pho75] and Blinn-Phong [Bli77] models. Strauss [Str90] and Ward [War92] presented empirical models with intuitive controls. More sophisticated BRDF models include the ones based on wave optics [Kaj85; HTSG91] and microfacet theory [TS92; ON94; APS00].

2.3.3 The Rendering Equation

The rendering equation, also known as the *Light Transport Equation* (LTE), calculates the total reflected radiance from a point on a surface based on surface emission, the BRDF, and the distribution of incident radiance [Kaj86; ICG86]. This equation expresses the outgoing radiance $L(x, \vec{\omega})$ of any point x based on the emitted radiance L_e and the BRDF:

$$\begin{aligned} L(x, \vec{\omega}) = & L_e(x, \vec{\omega}) + \\ & \int_{\Omega} f_r(x, \vec{\omega}, \vec{\omega}^{'}) \; L(x, \vec{\omega}^{'}) \; (\vec{n}.\vec{\omega}^{'}) \; d\vec{\omega}^{'}. \end{aligned} \tag{2.26}$$

It is important to note that the incoming radiance from all other points should be calculated based on the same equation, making the rendering equation a recursive relation. Calculating this recursive integral is the task of all *global illumination algorithms.*

Many methods have been developed with the goal of efficiently solving the rendering equation. One set of these approaches is the radiosity algorithms [GTGB84; CG85; NN85; ICG86; SAWG91; CSS96] which are based on the finite element methods. Another set of approaches are based on Monte Carlo integration which include path tracing [Kaj86], bidirectional path tracing [LW93; VG94], Metropolis light transport [VG97], photon mapping [Jen01] and progressive photon mapping [HOJ08].

2.3.4 The Phase Function

The *phase function* expresses the angular distribution of light scattering at a given point inside a medium (see Figure 2.15). It has a similar role as the BRDF, but different in that it is defined for participating media (whereas the BRDF is defined for surfaces).

Physically based phase functions have the following properties:

- **Reciprocity:** Similar to physically based BRDFs, phase functions follow Helmholtz's law of reciprocity:

$$p(x, \vec{\omega}, \vec{\omega}') = p(x, \vec{\omega}', \vec{\omega}) \tag{2.27}$$

- **Normalization:** Unlike BRDFs, phase functions must integrate to one for all points x and all directions $\vec{\omega}$ over the whole sphere of directions $\Omega_{4\pi}$:

$$\int_{\Omega_{4\pi}} p(x, \vec{\omega}, \vec{\omega}')\, d\vec{\omega}' = 1 \tag{2.28}$$

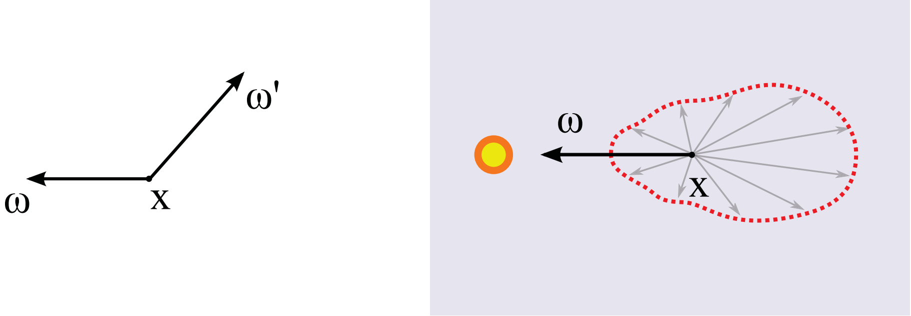


Figure 2.15: Left) Phase function expresses the angular distribution of light scattering at a given point inside a medium. Right) An example phase function and its response to the incoming light direction $\vec{\omega}$.

The most widely used phase functions in computer graphics include the Henyey-Greenstein phase function [HG41], Rayleigh Scattering [Ray71], and Lorenz-Mie Theory [Lor90; Mie08].

In Chapter 3 we focus on generating a physically based phase function for water drops inside the air in order to simulate rainbows. We compare our results for spherical waterdops with Lorenz-Mie results and show that our models extends the results to non-spherical water drops which cannot be handled using Lorenz-Mie.

2.3.5 The Volume Rendering Equation

The volume rendering equation, also known as the *Radiative Transfer Equation* (RTE), evaluates the outgoing radiance in a participating medium [Cha60; KVH84].

$$\begin{aligned} L(x,\vec{\omega}) \ &= T_r(x \leftrightarrow x_s)\ L(x_s,\vec{\omega}) + \\ &\int_0^s T_r(x \leftrightarrow x^{'})\ \sigma_a(x^{'})\ L_e(x^{'})\ dx^{'} + \\ &\int_0^s T_r(x \leftrightarrow x^{'})\ \sigma_s(x^{'})\ \int_{\Omega_{4\pi}} p(x,\vec{\omega},\vec{\omega}^{'})\ L(x,\vec{\omega}^{'})\ d\vec{\omega}^{'}\ dx^{'} \end{aligned} \tag{2.29}$$

where s is the depth of the medium, $T_r(a \leftrightarrow b)$ is the transmittance between point a and b, σ_s and σ_a are the scattering and absorption coefficients respectively, and p is the phase function of the medium. The first term represents the radiance from the surface at the end of the medium, the second term represents the accumulated emitted radiance and the last term is the accumulated in-scattered radiance.

Similar to the rendering equation 2.3.3, this equation is recursive in nature and computing the radiance at any points in medium depends on the radiance values of all other points in the medium and on the surfaces. This dependency makes evaluation of the volume rendering equation computationally expensive.

In computer graphics, many models have been developed for solving the volume rendering equation. Some of these methods are based on finite element methods and radiosity algorithm [RT87; BT92; Max94]. Other methods are mainly based on Monte Carlo integration which include approaches based on path tracing [PM93; HK93], bidirectional path tracing [LW96], Metropolis light transport [PKK00], photon mapping [JC98], and photon beams [JNSJ11].

3

Light Interactions with Microspheres

RAINBOWS are among the most visually stunning phenomena in nature. They are caused by the interaction of sunlight with small water drops in the atmosphere, and they appear in the form of multicolored arcs. The appearance of rainbows can vary significantly depending on the lighting conditions and the raindrop distribution.

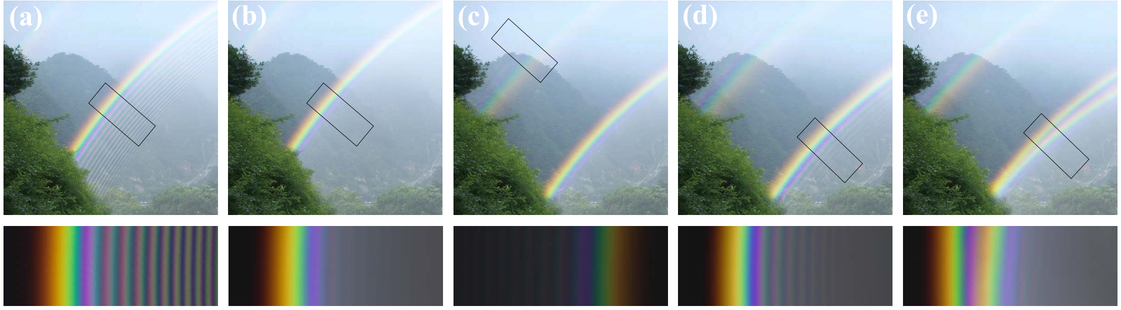


Figure 3.1: Our rendering results for different types of rainbows: **(a)** Rainbow derived from Lorenz-Mie theory. **(b)** Single primary rainbow with considering the angular view of the sun. **(c)** Double rainbow with a flipped secondary rainbow. **(d)** Multiple supernumerary rainbows caused by small water drops with uniform sizes. **(e)** Twinned rainbow resulted from mixture of non-spherical water drops and spherical ones.

Rainbows are one of the most interesting scattering phenomena in nature, owing to the rich interactions of light and water drops. Not only the dispersion caused by a wavelength-dependent index of refraction results into very beautiful shading gradients (primary and secondary rainbows), but other kinds of interactions at electromagnetic level are perceivable as well. For instance, wave interference results into a set of violet arcs close to the rainbow, and the effect of diffraction can transform the rainbow into a pure white arc for very small water drops. It is one of the very few outdoor phenomena in which such a rich set of microscopic wave interactions are perceivable by the human eye.

In this paper we derive a physically based model for simulating rainbows. Previous techniques for simulating rainbows has used either geometrical optics (ray tracing) or Lorenz-Mie theory, which computes the scattered wavefront from a spherical particle. Lorenz-Mie theory is by far the most accurate technique as it takes into account optical effects such as dispersion, polarization, interference, and diffraction. These effects are critical to accurately simulate rainbows. Unfortunately, the shape of real raindrops is non-spherical, especially for larger raindrops, and there is no alternative theory that can handle this case. We present the first comprehensive technique for simulating the interaction of a wavefront of light with a physically-based water drop shape. Our technique is based on ray tracing extended to account for dispersion, polarization, interference, and diffraction. Our model matches Lorenz-Mie theory for spherical particles, but it also enables the accurate simulation of non-spherical particles. It can simulate many different rainbow phenomena including double rainbows and supernumerary bows. We show how the non-spherical raindrops influences the shape of the rainbows, and we provide the first comprehensive simulation of the rare twinned primary bow, which is believe to be caused by non-spherical water drops.

Figure 3.2 shows examples of a full double rainbow, a close-up showing the darkening of Alexander's band, supernumerary bows and a rare twinned primary rainbow.

3.1 Introduction

Even though the study of rainbows can be traced back more than two thousand years [LF01], they are still not fully understood. For instance, twinned rainbows (which strangely are visible on the primary bow but not the secondary, as seen in Figure 3.20), are believed to appear due to water drops not being perfect spheres, but there is no solid theory to confirm this.

The first studies of rainbows assumed simple geometric optics where light is refracted as it enters or leaves the water drop. This model can explain the basic primary and double rainbow configuration, but it fails at explaining supernumerary bows that are caused by interference. To account for interference it is necessary to consider the sunlight as a wavefront interacting with the raindrop. This can be accomplished using Lorenz-Mie theory, which accounts for reflection, refraction, dispersion, polarization, interference, and diffraction, and it turns out that all of these optical effects are necessary to accurately simulate rainbows. Unfortunately, Lorenz-Mie theory is limited to spherical water drops, and this not only yields wrong predictions in some cases, but it ultimately limits the types of rainbows that can be explained as well. It is well-known that water drops become non-spherical as they get larger, and this heavily influences the distribution of the scattered light. Unfortunately, there is no theory available that can explain the consequence of physical water drops, and this is one of the reasons why rainbows continue to be an active research area.

In this paper we develop the first comprehensive model for rainbows in computer graphics. We explain the optical events that cause rainbows, and we develop an accurate ray tracing algorithm that accounts for the full spectrum of optical effects including dispersion, polarization, interference, and an efficient approximation for diffraction. We show how our model matches the results of Lorenz-Mie theory for spherical water drops, and how it extends to also account for non-spherical water drops. The result is the first accurate simulation of sunlight scattered by water drops of realistic non-spherical shape. We show how even a slight variation in the

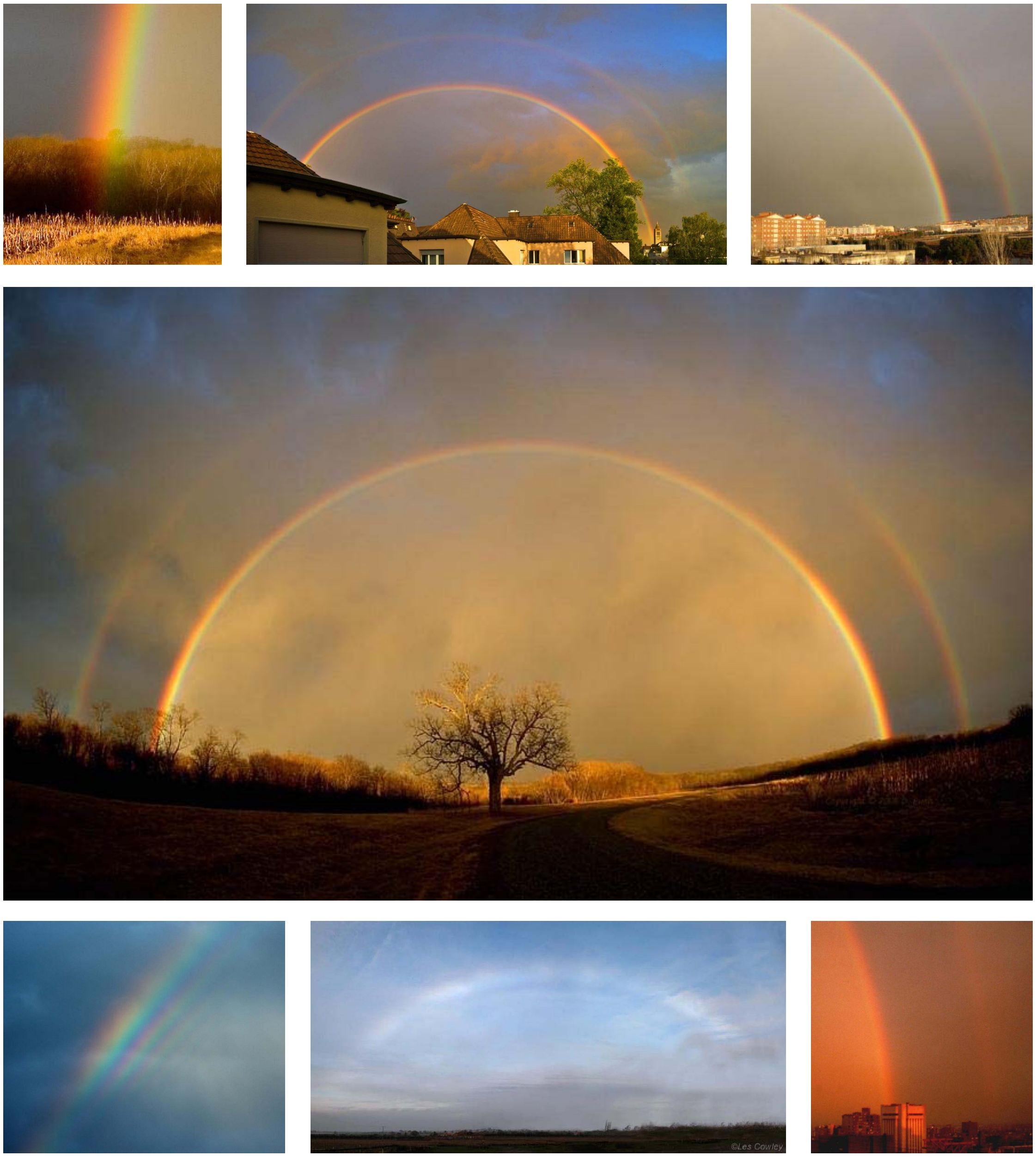


Figure 3.2: Different rainbows seen in nature. From top to bottom right to left: Primary rainbow, reproduced with permission ©Dan Bush [Bus]. Double rainbow (primary and secondary rainbows). Alexander's dark band. Double rainbow, reproduced with permission ©Dan Bush [Bus]. Supernumerary bows, reproduced with permission ©Ian Goddard [God]. Cloud bow. Red bow.

raindrop shape gives rise to changes in the position and shape of the rainbow. Our simulation can be used to explain the twinned primary bow shown in Figure 3.3, or show the effect of the angle of incidence of the light. We also use our model to explain the more common double rainbow as well as supernumerary arcs also shown in Figure 3.2. We also provide the full set of parameters that is needed to reproduce our results in Appendix A as well as a database of tabulated phase phase functions to the academic community[1].

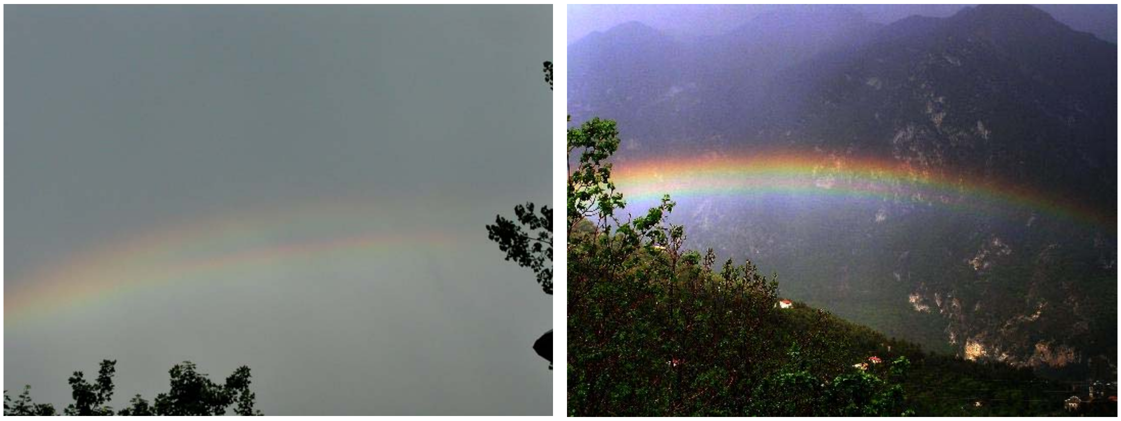


Figure 3.3: Examples of twinned rainbows. (right) Reproduced with permission ©Benjamin Khne [Kh]. (left) Reproduced with permission ©Vincent Jacques [Jac].

In summary, we present the following novel contributions:

- An in-depth analysis of the physical conditions and optical events which influence the appearance of rainbows.
- The first ray tracing model capable of accurately simulating rainbows by both spherical and non-spherical raindrops, including precise computation of the phase function.
- The first simulation of twinned rainbows due to large water drops of realistic non-spherical shape.

Our work falls in the domain of precise light simulation beyond the traditional

[1] http://graphics.ucsd.edu/ iman/Rainbows?

limitations of pure geometric optics. Similar approaches have been undertaken before in graphics, for instance in the field of gem modeling [GS04] or, the simulation of interference in thin layers [GMN94]. While we focused here on rainbows, we believe the model we develop can be used for other phenomena involving scattering by small particles (for example halos). Furthermore, accurate predictive rendering models of atmospheric phenomena, like the one we present, can have wider-reaching impact in areas such as meteorology e.g., by providing a key component in deducing the size of water drops from photographs [NN03].

3.2 Background Theory

Rainbows are created from the interaction between light and a participating medium composed of water drops suspended in the air. The most important visual effects are due to single scattering (note that we consider multiple light bounces within a single water drop as single-scattering). The complex phase function in this interactions produces the rich and varied angular distribution of radiance we observe as rainbows. Multiple scattering is responsible for the grayish background that appears behind the rainbows themselves. The effect of absorption on rainbow formation is negligible since the absorption of light in water reaches a maximum of 3.5×10^{-8} (expressed as the imaginary part of its refractive index [PF97]). In the following we describe the formation of rainbows, from geometric optics to wave effects, and introduce the actual shape of water drops, all of which will become the physical basis for our simulation algorithm described in Section 3.4.

3.2.1 Dispersion

The basic formation of the primary and secondary rainbow can be understood using simple geometric optics, considering ray paths within the circular cross-section of a spherical drop of water. For spherical drops, due to symmetry, the phase function is a 1D function of the scattering angle θ between the incident and

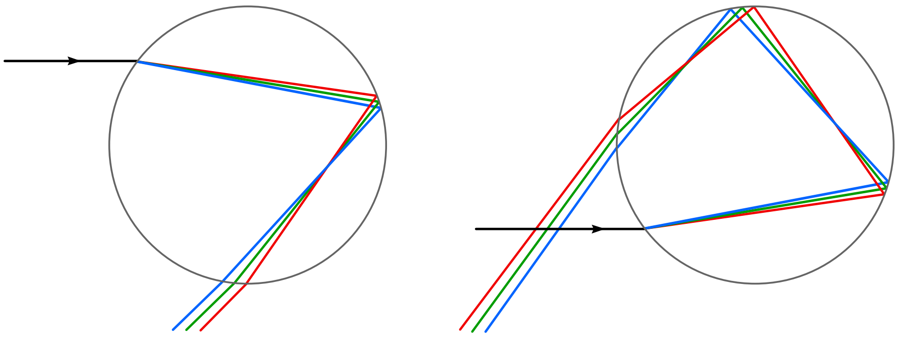


Figure 3.4: Dispersion of light as it gets reflected and refracted by the water drop causes the primary bow (left) and the secondary bow (right).

outgoing directions. Light rays that undergo one internal reflection in the water drop produce the primary rainbow for red light ($700\,nm$ wavelength) for an index of refraction of $\eta = 1.3314$ at a scattering angle of $\theta_{\text{rainbow}} = 137.7°$ and for violet light ($400\,nm$ wavelength) for $\eta = 1.3445$ at $\theta_{\text{rainbow}} = 139.6°$ (see Figure 3.4). Rainbows can also be generated by light rays that undergo two or more internal reflections: in the case of two internal reflections, the resulting secondary rainbow varies between 129.5° for red light and 126.1° for violet light (see Figure 3.4).

Note the order of the colors of the secondary rainbow (red on the inside of the arc and violet on the outside) is reversed compared to the primary rainbow, as seen in Figure 3.2. The darker area between both rainbows is known as *Alexander's dark band* (better perceived in Figure 3.2 top right).

3.2.2 Interference

Although ray optics can provide a good, basic explanation of the formation of the primary and secondary rainbows, real rainbows exhibit some features that cannot be explained with this model. For example, additional arcs (known as *supernumerary arcs*) occasionally appear on the inside of the primary rainbow (and the outside of the secondary): such arcs are typically violet or blue (Figure 3.2 bottom left). Supernumerary arcs caused great consternation, as stated in Chapter 8 of the book by Lee and Fraser [LF01], because they were not predicted by ray

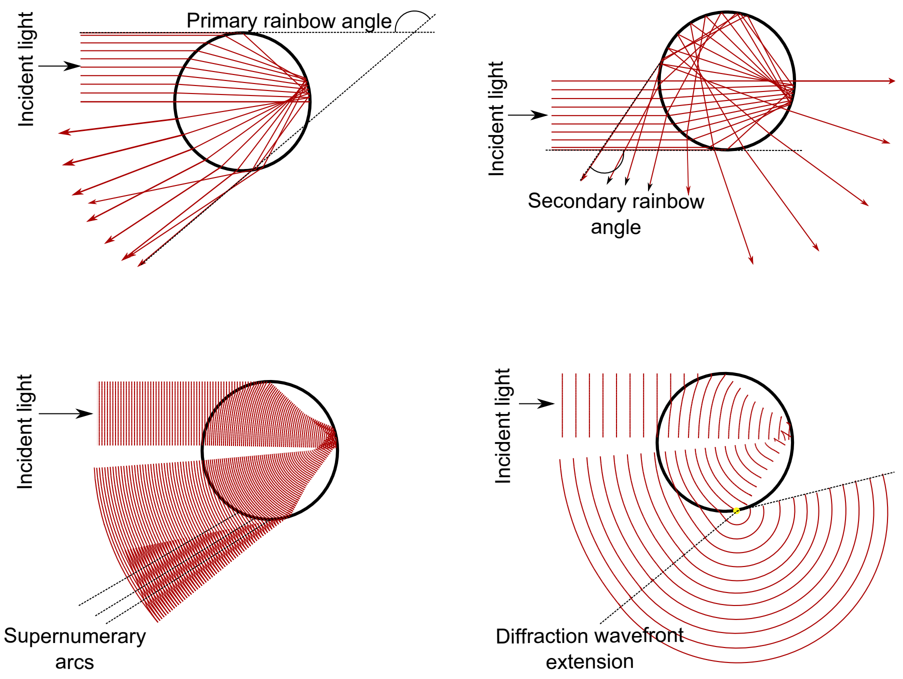


Figure 3.5: Generation of rainbows from the point of view of geometric and wave optics: **(a)** primary rainbow angle, after a single internal reflection, **(b)** secondary rainbow angle, after two internal reflections, **(c)** supernumerary rainbows are generated from constructive and destructive interference patterns (inspired by Lee and Fraser [LF01]), **(d)** diffraction extends the wavefront and avoids abrupt intensity changes.

optics. However, in the 1830s, scientists such as Young and Airy realized that they were a consequence of the wave theory of light: two rays that have different path lengths, must also have different phases – which result in an interference pattern consisting of a series of maxima and minima as a function of the scattering angle (see Figure 3.5). This phase difference between the two rays is also influenced by phase changes due to reflection.

Focal lines [van57] As a wavefront interacts with the water drop, it gets deformed. The surface that represents this wavefront can be differentially defined by the curvature at each point. This curvature defines a radius with respect to a corresponding *focus* or *focal point*. These focal points are actually internal caustics within the water drop. As light passes through a water drop, the collection of all focal points lie along a *focal line* (see Figure 3.6). Each passage through a focal line along the path results in a phase advance of $\pi/2$ [van57]. This traversal of focal lines needs to be considered as well for an accurate estimation of the phase difference between two interfering rays.

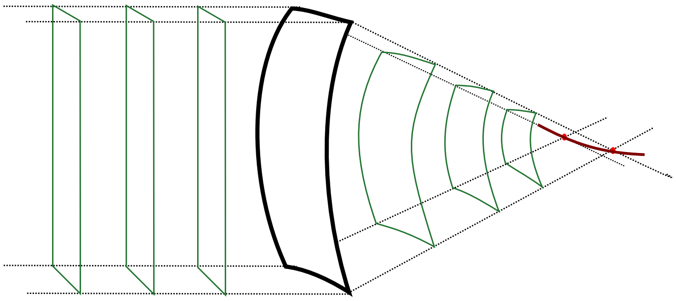


Figure 3.6: A focal line defines the curve along which all the differential focal points of the wavefront lie. The thick black represents a portion of the surface of the water drop. The green patches represent the wavefront itself. Notice how they converge to the red line, which is the focal line.

3.2.3 Diffraction

Another failure of ray optics is that it predicts infinite intensity at θ_{rainbow} and no scattering light when $\theta < \theta_{\text{rainbow}}$, while diffraction predicts that this abrupt radiance gradient cannot happen in reality [van57] (see Figure 3.5). Airy [Air38] produced an elegant mathematical solution which avoided both of these problems and proved that the peak intensity of the rainbow does not occur at $\theta_{\text{rainbow}} = 137.86°$, but at a slightly higher value of approximately $\theta = 138.9°$.

Another consequence of diffraction is that scattering from small droplets of water (such as fog in which the droplet radius is typically between $5\,\mu m$ and $20\,\mu m$) can generate rainbows that are essentially white. As the diffraction pattern for small droplets has very broad maxima in terms of θ, the rainbows corresponding to different wavelengths in the visible spectrum tend to overlap each other, thus creating white *fog bows* or *cloud bows*. Therefore, diffraction becomes more relevant as the water drops get smaller.

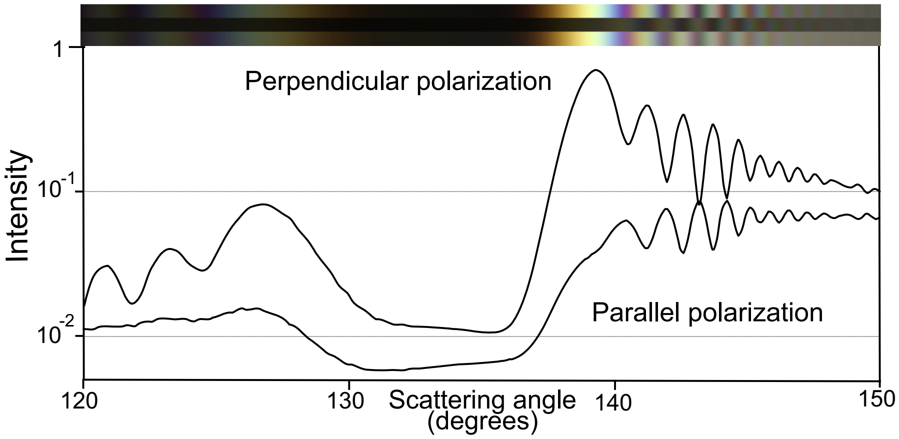


Figure 3.7: Graph of intensity as a function of scattering angle for the primary rainbow caused by scattering of sunlight by a spherical drop of water of radius $0.1\,mm$. The color stripes on top represent, from top to bottom, the phase function for perpendicular polarization, parallel polarization and unpolarized light, respectively.

3.2.4 Non-Spherical Water Drops

We consider physically-based water drops that are the same size and shape throughout the medium. Though water drops suspended in air are never homogeneous (the distribution of shapes and sizes can even be time-varying), given the ability to simulate phase functions for arbitrary geometry, such distributions can be trivially accounted for by calculating and combining a set of phase functions according to the distribution. For each water drop size, we use the model by Beard and Chuang [BC87] which accounts for surface tension as well as hydrostatic and aerodynamic pressure. Beard and Chuang proposed a cosine series fit to the model, with the shape of the raindrop profile given by the polar curve

$$r = a[1 + \sum c_n \cos(n\theta)], \tag{3.1}$$

where a is the radius of the equivalent volume sphere, while the coefficients c_n are listed as tabulated values (see Table 3.1). Figure 3.8 shows visualizations of several water drop shapes based on this work. Other models and approaches (and even water drop distributions) can be trivially used in our simulations, as the algorithm can handle arbitrary geometries.

Figure 3.8: Nonspherical raindrop shape with increasing radii as proposed by Beard and Chuang [BC87].

Table 3.1: Water drop polar curve coefficients [BC87]. The $a = 0.4$ row has been added to account for spherical water drops. Intermediate values are obtained through linear interpolation.

$a(mm)$	c_0	c_1	c_2	c_3	c_4	c_5	c_6	c_7
0.4	0	0	0	0	0	0	0	0
1.0	-0.0131	-0.0120	-0.0376	-0.0096	-0.0004	0.0015	0.0005	0
1.5	-0.0282	-0.0230	-0.0779	-0.0175	0.0021	0.0046	0.0011	-0.0006
2.0	-0.0458	-0.0335	-0.1211	-0.0227	0.0083	0.0089	0.0012	-0.0021
2.5	-0.0644	-0.0416	-0.1629	-0.0246	0.0176	0.0131	0.0002	-0.0044
3.0	-0.0840	-0.0480	-0.2034	-0.0237	0.0297	0.0166	-0.0021	-0.0072

3.3 Previous Work

In this section we describe the previous work that is relevant to rendering rainbows and similar atmospheric effects. Rainbows have traditionally been considered a fascinating topic, from scientists to philosophers, and are arguably one of the most beautiful displays of nature [Gre90; Min93; LL01]. Different theories have been developed over the centuries, and some of them have been adopted by the computer graphics community to simulate rainbows with varying degrees of realism. Some techniques are based on a simplification of the process, in order to achieve interactive frame rates, while others present rainbow simulations in the context of atmospheric modeling. However, the complete (and quite complex) physics of rainbow formation has not been fully researched in the field of computer graphics.

3.3.1 Based on Ray Optics

Earliest studies on the formation of rainbows date back to the ancient Greek philosophers. Aristotle (384-322 BCE) has the first documented study on rainbows. Although his explanation had many flaws, his qualitative description remained the main theory for the formation of rainbows for centuries [JF01]. The first accurate explanation of both primary and secondary bows is given by Theodoric of Freiberg in 1307. He explained the secondary rainbow through a similar analysis involving two refractions and two reflections [Lin66]. Consequently, in 1637, Rene Descartes extended the previous models in his book his book "Discourse on the Method" [Des37]. He presented the fist analytical calculation of the scattering angles for primary and secondary bows. For a sketch of his explanation for the formation of rainbows see Figure 3.9 right. In 1704, Isaac Newton refined previous models and explained the dispersion of light as a result of light interaction with water drops. In his book "Opticks" [New04], Newton showed that red light gets refracted less than green light which in turn gets refracted less than blue light. he showed that white light contains all the color variations that can be seen in rainbows (see Figure 3.9 left).

In computer graphics, simplified solutions include the work by Musgrave [Mus89], which follows Descartes' model from a classical geometric optics perspective. Frisvad and colleagues [FCF07] presented a real-time simulation using Aristotle's rainbow formation theory based on reflections in clouds. Although these models may provide intuitive explanations about rainbow formation, ray optics by itself fails to

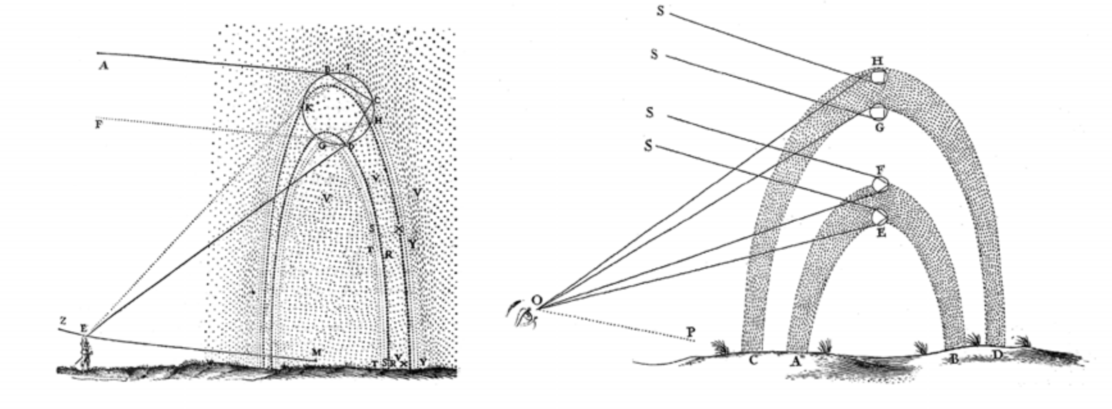


Figure 3.9: Hand drawn sketches of (left) Rene Descartes, from [Des37], and (right) Isaac Newton, from [New04], on explaining how primary and secondary rainbows are formed.

capture more complex aspects such as supernumerary arcs (see Figure 3.2 bottom left) which are due to the inetreferance of relected rays.

3.3.2 Lorenz-Mie Theory

Lorenz-Mie theory [Lor90; Mie08] developed a rigorous solution to the problem of scattering of light from spheres, taking into account not only interference, but polarization and radius distribution as well. Figure 3.7 shows the result of Lorenz-Mie theory calculations to simulate the scattering of sunlight by a water drop with radius $100\,\mu m$. It shows the primary rainbow near $\theta \approx 139°$, the secondary rainbow near $\theta \approx 127°$ and Alexander's dark band between $\theta \approx 130°$ and $\theta \approx 136°$. Note that the primary and secondary rainbows are strongly polarized: the dominant polarization is given by the perpendicular component of the electric field (with respect to the scattering plane). The colored horizontal bars above the graph in Figure 3.7 show the colors and relative brightness of the rainbows: the top bar represents perpendicular polarization, the middle bar represents parallel polarization whilst the lower bar represents unpolarized light (the combination of the two).

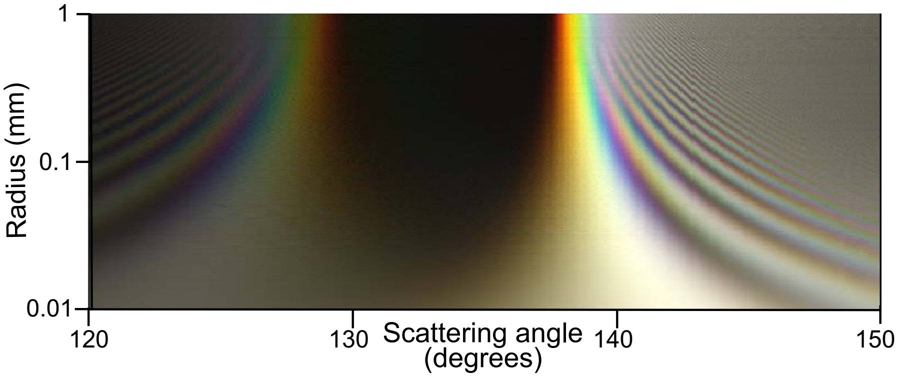


Figure 3.10: Lee diagram showing the variation in appearance of primary and secondary rainbows caused by scattering of sunlight by a spherical water drop as a function of radius (Lorenz-Mie theory calculations)

Figure 3.10 illustrates how the appearance of the primary and secondary rainbows varies with the radius of the (spherical) water drop, according to Lorenz-Mie theory: this type of diagram was first shown by Lee [Lee98] and is consequently known as a *Lee diagram*. It shows the complexity of rainbows and their supernumerary arcs.

When the size of spherical particles is larger than the wavelength of light, ray optics solutions are satisfactory. When particles are smaller than the wavelength of light, simple Rayleigh scattering can describe the behavior of light. For particles with sizes comparable to the wavelength of light, however, we need the more complex Lorenz-Mie theory to describe the behavior of light [Gla95].

Figure 3.7 shows the result of Lorenz-Mie theory calculations to simulate the scattering of sunlight by a water drop with radius $100\,\mu m$. It shows the primary rainbow near $\theta \approx 139°$, the secondary rainbow near $\theta \approx 127°$ and Alexander's dark band between $\theta \approx 130°$ and $\theta \approx 136°$. Note that the primary and secondary rainbows are strongly polarized: the dominant polarization is given by the perpendicular component of the electric field (with respect to the scattering plane). The colored horizontal bars above the graph in Figure 3.7 show the colors and relative brightness of the rainbows: the top bar represents perpendicular polarization, the middle bar represents parallel polarization whilst the lower bar represents unpolarized light (the combination of the two).

Lorenz-Mie theory [Lor90; Mie08] provides an exact solution for scattering by spherical particles in non-absorbing media. Given its computational complexity, it was not deemed useful until van de Hulst published results as tabulated data [van57]. Unfortunately, this work is limited to very small spheres, and thus not directly suitable for rainbows. This theory was later introduced to the graphics community by Rushmeier [Rus95], and was used recently to compute scattering properties of different materials [FCJ07].

Lee [Lee98] investigated the differences between results obtained using Lorenz-Mie theory and Airy theory [Air38], including perceptual issues. Jackèl and Walter [JW97] simulate rainbows by adding a rain layer to the atmosphere and making use again of Lorenz-Mie theory to compute phase functions for single scattering. In their work, raindrop sizes follow a normal distribution. A similar approach with a log-normal distribution was introduced by Riley et al. [REK$^+$04], who achieve interactive frame rates with simplified lighting models. Phase functions are obtained based on the work by Laven [Lav03], which implements the algorithm from Bohren and Huffman [BH83] to obtain scattered intensities. A simplified, texture-based GPU implementation has also been developed [nVI04]. Recently, Gedzelman [Ged08] explored the influence of the atmospheric environment on the appearance of rainbows; although valid conclusions on overall brightness and visibility were reached, the results did not aim to be photorealistic.

Most of these approaches are based on Lorenz-Mie theory which, unfortunately, can only provide an accurate solution in the case of spherical water drops. However, real water drops diverge from perfect spheres due to the combined effects of gravity and surface tension [BC87; BKI91; BCX91; VB09]. This translates into inaccurate simulations in the best case, and the impossibility to simulate certain effects like the twinned rainbow in the worst case. In this paper, we introduce a novel algorithm based on a ray-tracing approach, which for the first time matches the predictions of Lorenz-Mie theory for the ideal case of spherical water drops, but naturally generalizes to handle actual, real-world water drop geometries. This allows us to produce excellent simulations of rainbows, while extending the validity of such simulations to include scattering from non-spherical drops of water.

3.3.3 FDTD Methods

The most practical solution for computing the scattering properties of particles with arbitrary shapes, are the *Finite Difference Time Domain* (FDTD) methods [Yee66; Taf98]. These methods can simulate Maxwell's time-dependent equations on a discrete lattice in order to compute the behaviour of the electromagnetic wave. These brute-force methods have been used to simulate the light scattering behaviour of complex objects [UT82] as well as ice crystals [YL95; YL96].

The disadvantage of FDTD methods is that they are computationally very expensive for three-dimensional grids and can take days on multi-core processors. Furthermore, the generalizations of Lorenz-Mie theory to non-spherical particles by Frisvad et al. [FCJ07] cannot be used as it only applies to the computed scattering cross-section, while the appearance of rainbows are caused by variations in the angular scattering profile (the phase function).

3.4 Our Approach

A key aspect for an accurate simulation of rainbows is the precise computation of the phase function, which defines the angular distribution of radiance for every wavelength. Some of the approaches discussed in the previous work propose efficient methods to render rainbows, but they do not actually *simulate* precise phase functions, which they take from available simulators such as AirySim[2], BowSim[3] and MiePlot[4]. While these simulators do a great job at approximating the phase function of rainbows under some conditions, they all have hard limitations: none of

[2] http://www.atoptics.co.uk/rainbows/airysim.htm
[3] http://www.atoptics.co.uk/rainbows/bowsim.htm
[4] http://www.philiplaven.com/mieplot.htm

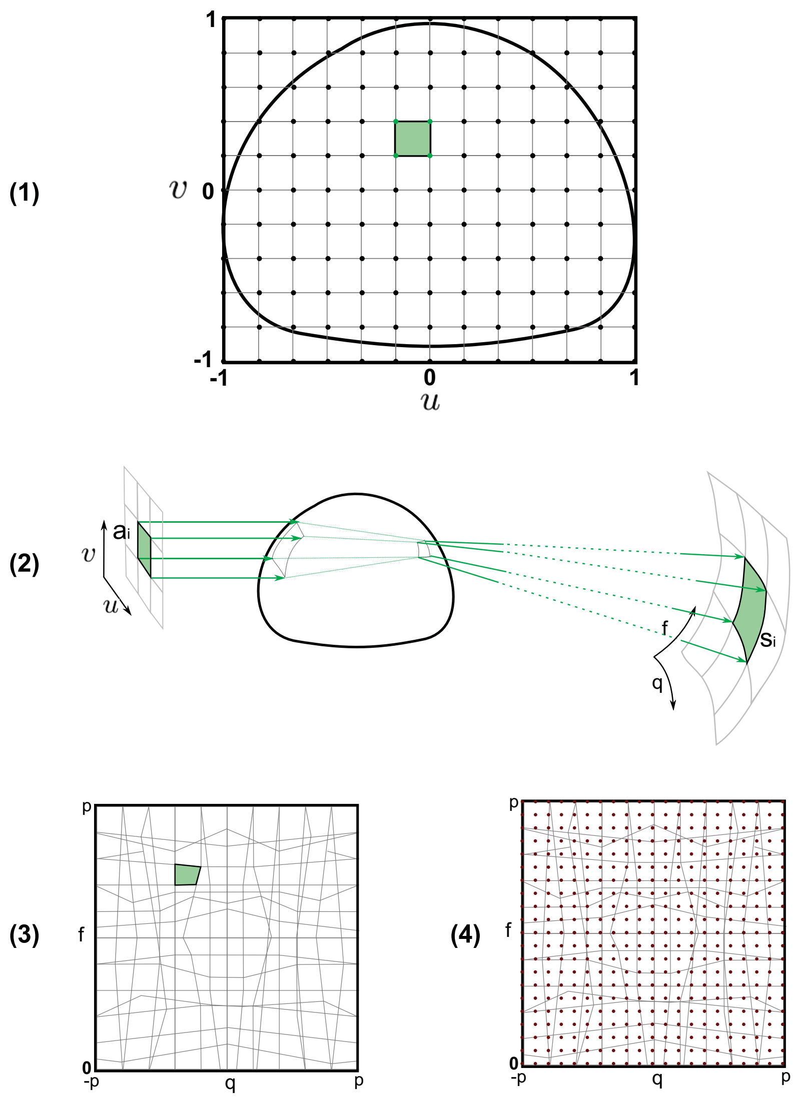


Figure 3.11: Steps of the algorithm. **(1)** Casting the grid of rays towards the particle. **(2)** Rays are reflected and refracted towards the water drop, forming patches. **(3)** Outgoing patches are collected in an infinite collecting sphere. **(4)** The stored patches in the collecing sphere are queried at specific directions, sampling the phase function.

them can handle physically-based water drop shapes, limiting the computations to spheres BowSim can additionally handle ellipsoids, but it does not consider interference for its simulations. AirySim, however, approximates interference using Airy functions. In this section we focus on this key aspect of rainbow simulation and our proposed solution for arbitraty geometries. This is our primary contribution.

3.4.1 Overview

We compute phase functions for non-spherical water drops by taking a *virtual gonioreflectometer* approach. In essence, we simulate the way a collection of light rays scatter off a water drop and gather the resulting information on an infinite collecting sphere. The problem at hand is thus similar to rendering caustics, and therefore a pure Monte Carlo approach would be impractical. Unfortunately, photon mapping would not work either, since interference in this configuration produces extremely high frequency details which the radiance estimation technique would fail to reproduce. To include all the important optical properties of real rainbows we augment our ray tracing computation to account for dispersion, polarization, interference, and diffraction. Unlike Lorenz-Mie theory, which is limited to spherical drops, our approach allows us to use the real shape of the drops and thus produce more accurate simulations.

Our algorithm simulates the phase function by following several steps for each wavelength:

1. We cast a grid of rays from an emitting plane that represents the wavefront of a directional light source. Each ray carries wave information represented using phasors.

2. These rays interact with the water drop (through reflection and refraction) a number of times and exit the water drop forming patches.

3. The outgoing patches are deposited on an infinite collecting sphere and stored in an acceleration structure.

4. The phase function is discretized into tabulated form by querying the acceleration structure along a 2D set of sampling directions uniformly distributed in longitude-latitude.

The stored tabulated phase function is later used for rendering. All these steps are illustrated in Figure 3.11.

In the following, we first explain the basis of our approach from a classic ray-tracing perspective, for the sake of clarity; we then introduce our phasor notation which allows us to efficiently compute interference and polarization.

Casting Rays Inspired by the beam tracing technique [HH84], and similar to the work by Collins [Col94], we follow a wavefront of light by casting a grid of rays (3000×3000 rays for our results). This way, rays that are contiguous and represent the same wavefront can be identified. Rays are perpendicular to a reference emitting plane, representing a collimated light source that emits a parallel wave train (see Figure 3.11(1)). Each of these rays is propagated through interactions with the water drop, which can be of arbitrary geometry. For our tests we use a physically-based geometric model that accounts for different particle sizes [BC87] although any other model or specific geometry could be considered instead.

When a ray interacts with the water drop, its path is reflected and refracted according to the law of reflection and Snell's law, respectively. We account for up to four consecutive interactions: a single reflection, two refractions, two refractions with an internal reflection (primary rainbow) and two refractions plus two internal reflections (secondary rainbow). Though further bounces could easily be handled, they have a negligible effect on the resulting phase function.

water drop, we store the outgoing rays and their corresponding adjacency information as a set of patches on an infinite virtual collecting sphere (see Figure 3.11(3)). Each vertex of a patch thus represents one outgoing ray and contains wave data.

The energy of the emitting plane is split among all the grid cells according to each grid cell's area a_i. When this energy exits the particle and reaches the collecting sphere, it is transformed into radiance by considering the solid angle s_i of the resulting patch. Therefore, the ratio $\frac{a_i}{s_i}$ determines a patch's contribution to the phase function, which amounts to density estimation. While we could apply this relation directly to photometric units, this would not account for many of the effects that contribute to rainbows such as interference, polarization, focal lines, and diffraction. Instead, we apply this ratio to the corresponding wave data, which is described in the following section.

3.4.2 Simulating Dispersion

For simulating the dispersive behavior of water drops we need to consider the index of refraction of water ρ as a function of light's wavelength λ. During our ray casting step, we compute the refraction angle of each wavelength of light based on the corresponding index of refraction. There are many sources that provide the refractive index of water for different wavelenghts of light either as tabulated data [HQ73] or by providing mathematical equations [ftPoWS]. We have listed the the index of refraction of light for 33 samples of visible spectrum which we have used for our simulation in Table 3.2.

Table 3.2: List of index of refraction ρ of water for diffrent wavelengths of light λ. The values are derived from the data presented in [HQ73]

$\lambda(nm)$	ρ	$\lambda(nm)$	ρ	$\lambda(nm)$	ρ
380	1.3406	490	1.3354	600	1.332
390	1.3398	500	1.335	610	1.332
400	1.339	510	1.3346	620	1.332
410	1.3386	520	1.3342	630	1.3318
420	1.3382	530	1.3338	640	1.3314
430	1.3378	540	1.3334	650	1.331
440	1.3374	550	1.333	660	1.331
450	1.337	560	1.333	670	1.331
460	1.3366	570	1.333	680	1.331
470	1.3362	580	1.3328	690	1.331
480	1.3358	590	1.3324	700	1.331

3.4.3 Computing Interference and Polarization

To account for interference and polarization, we characterize light in terms of an electromagnetic field **E** perpendicular to the direction of the ray. Defining a coordinate system with the z-axis along the direction of propagation, we can define it in terms of two orthogonal *phasors* of the electromagnetic field [Gia89]:

$$\begin{aligned} E_x &= A_x e^{i\left(\frac{2\pi}{\lambda}z-\omega t+\delta_x\right)} \\ E_y &= A_y e^{i\left(\frac{2\pi}{\lambda}z-\omega t+\delta_y\right)}, \end{aligned} \tag{3.2}$$

where A_x and A_y are the amplitudes, λ is the wavelength, ω is the angular frequency, t is time and δ_x and δ_y represent phase offsets. The irradiance carried by a planar electromagnetic wave represented by two phasors is $A_x^2 + A_y^2$. Notice that Stokes notation would fail to simulate interference because it just takes into account the relative phase difference between the two wave components.

We assume that all the waves have traveled the same optical path from the sun (and therefore z is a constant reference path) and also consider a stationary simulation of the phase function, where ωt becomes constant. Furthermore, we sample fixed values of λ along the visible spectrum and simulate each independently. As a consequence, the only relevant information for each phasor is the amplitude A and the corresponding phase offset δ, which is the polar representation of the phasor.

Following Euler's formula, $e^{ix} = \cos x + i \sin x$, we can represent a phasor $Ae^{i\delta}$ by a complex number (rectangular representation) for which the real part is $A\cos\delta$ and its complex part is $A\sin\delta$. This rectangular representation is efficient for phasor addition (interference) and phasor interpolation. Furthermore, it enables a very straightforward simulation of the interactions between the electromagnetic wave and the water drop, by applying the corresponding Fresnel coefficients (as explained below in the text).

Each ray in our algorithm carries the following information:

- Two phasors E_x and E_y rectangularly represented by their corresponding complex numbers.

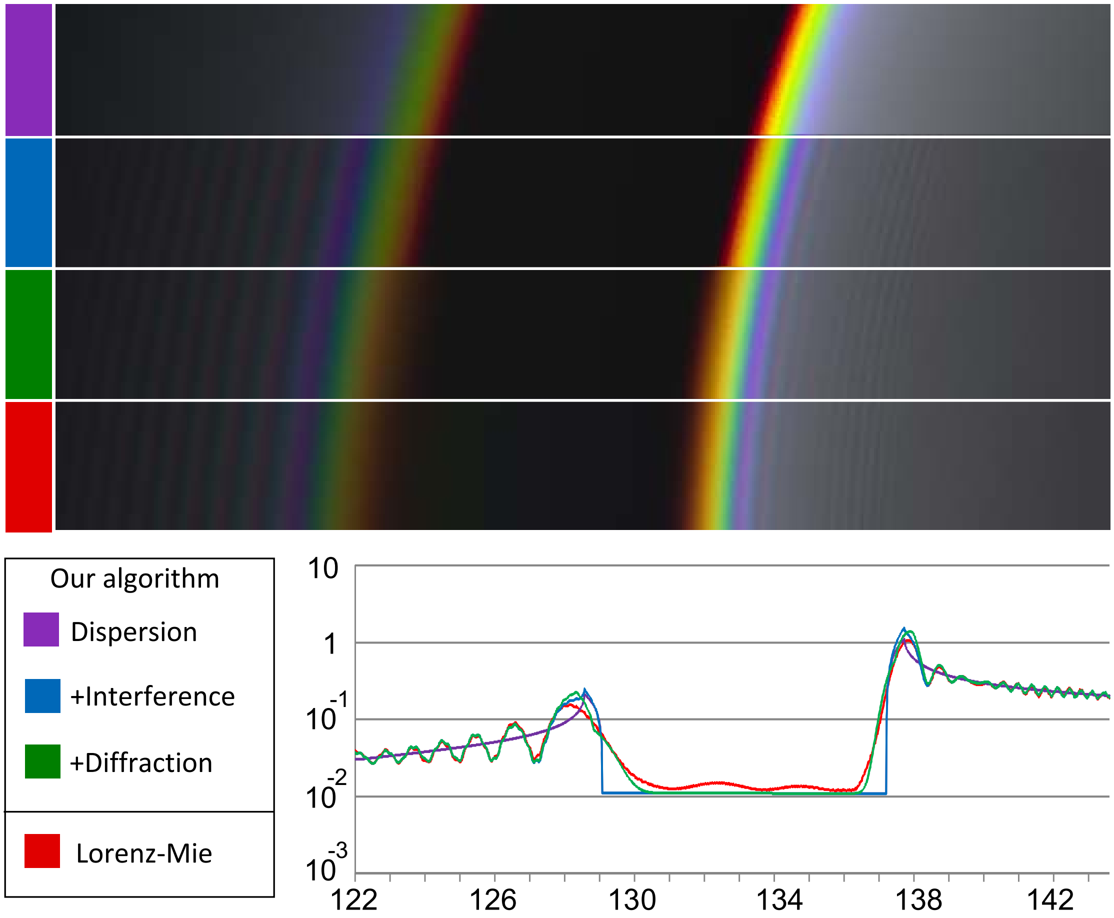


Figure 3.12: Comparison between the different phenomena simulated by our method (dispersion, interference and diffraction) and the same simulation from the Lorenz-Mie theory, for a 0.4mm radius spherical water drop. Top: Renderings for 33 wavelength. Bottom: Graphs for 650nm wavelength.

- The traversed optical path l.

Additionally, during ray tracing, we consider the frame that represents the coordinate system of the two axes of the electromagnetic wave (perpendicular to the propagation direction). These axes are rotated as needed for the different interactions.

Phase shifts need to be taken into account; these occur at the interaction with the water drop, along the traversed optical path, and at focal lines. As rays are traced towards the drop, we modify the phasors at the interactions with the water drop. Phase shifts due to optical path and focal lines are included after the bilinear interpolation at each patch (see Subsection 3.4.3.1).

There has been some previous work for ray tracing polarization effects [WK90; TTW94; WTP01], all based on coherency matrices. Our approach, on the other hand is similar to using Jones vectors [JON41] and can account for interference.

Ray-Water Drop Interactions Light interacting with a water drop gets both reflected and refracted, with the total amplitude divided between both rays in terms of the parallel and perpendicular components with respect to the plane of incidence. We rotate the coordinates of the two components of the wave to a parallel-perpendicular coordinate system. As in previous work by Gondek et al. [GMN94], the respective amplitudes are multiplied by the Fresnel coefficients $t_{\|}, t_{\perp}, r_{\|}$ and $r_{\perp}$, which can be found in most optics books [LLT95] and can become complex in the case of total internal reflection. We multiply phasors with these potentially complex coefficients using the rectangular representation of complex numbers explained before.

Optical Path The optical path l is defined as $l = \int_P \eta \, dp$, where η is the index of refraction and p refers to the differential traversed path. In our case the total optical path traversed by a ray is $l = \eta_i p_i + \eta_t p_t$, where p_i and p_t are the total distances traversed outside and inside the water drop, respectively. Given the impossibility of computing infinite path lengths from the sun, we rely on the fact that interference computations require just *relative* optical paths between different rays; we thus consider the common casting plane to be placed at a distance d from the center of the water drop (which would represent a distance z from the sun in Equation 3.2). In a similar fashion, we set a second reference plane perpendicular to each outgoing ray, placed at a distance d' from the center of the water drop (*not* from the original of the ray). The optical path l is accumulated as the ray traverses the water drop by simply adding the Euclidean distances between interactions outside and inside the particle. We account for this effect on phase change during the bilinear interpolation step in Subsection 3.4.3.1.

Focal Lines Focal lines must also be considered for an accurate simulation of the phase carried by each ray, given that each passage through a focal line along the path results in a phase advance of $\pi/2$ [van57]. Unfortunately, both computing focal lines caused by arbitrary geometry and detecting which rays actually traverse a focal line are very complex tasks. However, we can approximate the exact solution by leveraging the fact that we only need to take into account the area close to the rainbow. Furthermore, for interference, it is again only the phase difference that needs to be taken into account. We thus analyze the sign of the derivative of the outgoing angle θ with respect to the impact parameter $b = u^2 + v^2$ (where u and v are the parameters that define the projection plane from which the rays are cast, as illustrated in Figure 3.11(1)). When this derivative is positive, we consider one extra focal line than when the derivative is negative. The derivative (and therefore the number of focal lines) is easily computed at each of the patches from its corners. We consider that any direction inside a patch represents a ray that has crossed that number of focal lines.

3.4.3.1 Interference

In order to save the phase function to the hard drive (so it can be later used in a renderer), we tabulate it per wavelength by generating a 2D set of directions uniformly distributed in longitude-latitude coordinates. Each of these samples corresponds to a direction $\mathbf{r}$. We compute the outgoing radiance for a specific direction $\mathbf{r}$ within a patch using bilinear interpolation of the data stored at the four vertices of the patch. This bilinear interpolation is equivalent to assuming that the wavefront at each of the patches is planar, and the error we commit by making this assumption becomes negligible as the resolution of patches increases.

We then combine the interpolated data at each of the patches that contain a direction to account for interference between wavefronts. For efficiency we consider

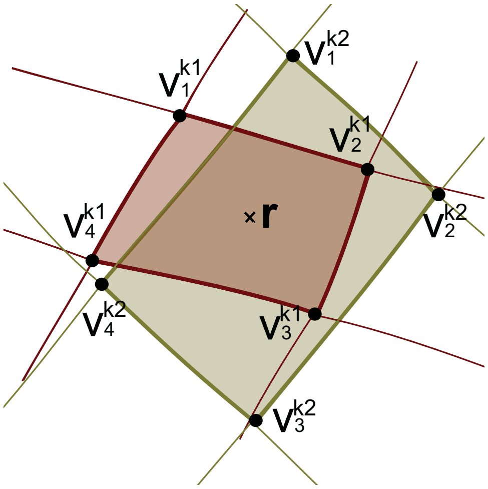


Figure 3.13: Direction **r** intersects patches $k1$ (red) and $k2$ (yellow) (two parts of two different wavefronts). At each of both patches, bilinear interpolation from all the corners v_i^k result into an interpolated electromagnetic wave per patch. Interference is then computed by combining (adding) all the interpolated electromagnetic waves.

the whole set of patches as a virtual geometry and we create a bounding volume hierarchy (BVH) over them. Given an outgoing direction **r**, we find the set of patches $\Upsilon(\mathbf{r})$ that contains **r** by tracing a ray from the center of the collecting sphere in that direction. We consider all intersected patches for interpolation and interference.

In Figure 3.13, we show an example for two patches $k1$ (red) and $k2$ (yellow), which represent two different interfering wavefronts. The ray at each of the four corners v_i^k of each patch contains information about the two corresponding phasors E_{xk} and E_{yk} and the optical path l_k. At each of the patches $k \in \Upsilon(\mathbf{r})$ we bilinearly interpolate this information from the four corners (at the specific direction **r**). Furthermore, we calculate the number of traversed focal lines f_k for that patch, as explained above. As the irradiance carried by a planar electromagnetic wave represented by two phasors is $A_x^2 + A_y^2$ we account for the corresponding radiance (applying the $\frac{a_i}{s_i}$ factor, as stated in the previous section) by multiplying each of the amplitudes by $\sqrt{\frac{a_i}{s_i}}$. We then obtain the new phasor information E'_{xk} and E'_{yk}

including the phase shift due to the traversed optical path and to the number of traversed focal lines for each interpolated wave as:

$$\begin{aligned} E'_{xk} &= E_{xk} e^{i\left(\frac{2\pi}{\lambda} l_k + \frac{\pi}{2} f_k\right)} \\ E'_{yk} &= E_{yk} e^{i\left(\frac{2\pi}{\lambda} l_k + \frac{\pi}{2} f_k\right)}. \end{aligned} \tag{3.3}$$

We then compute the final outgoing radiance due to interference by adding all the traversed phasors:

$$\begin{aligned} E_x(\mathbf{r}) =& \sum_{k \in \Upsilon(\mathbf{r})} E'_{xk} \\ E_y(\mathbf{r}) =& \sum_{k \in \Upsilon(\mathbf{r})} E'_{yk}, \end{aligned} \tag{3.4}$$

where $E_x(\mathbf{r})$ and $E_y(\mathbf{r})$ are the two components of the wave that exits the water drop towards $\mathbf{r}$.

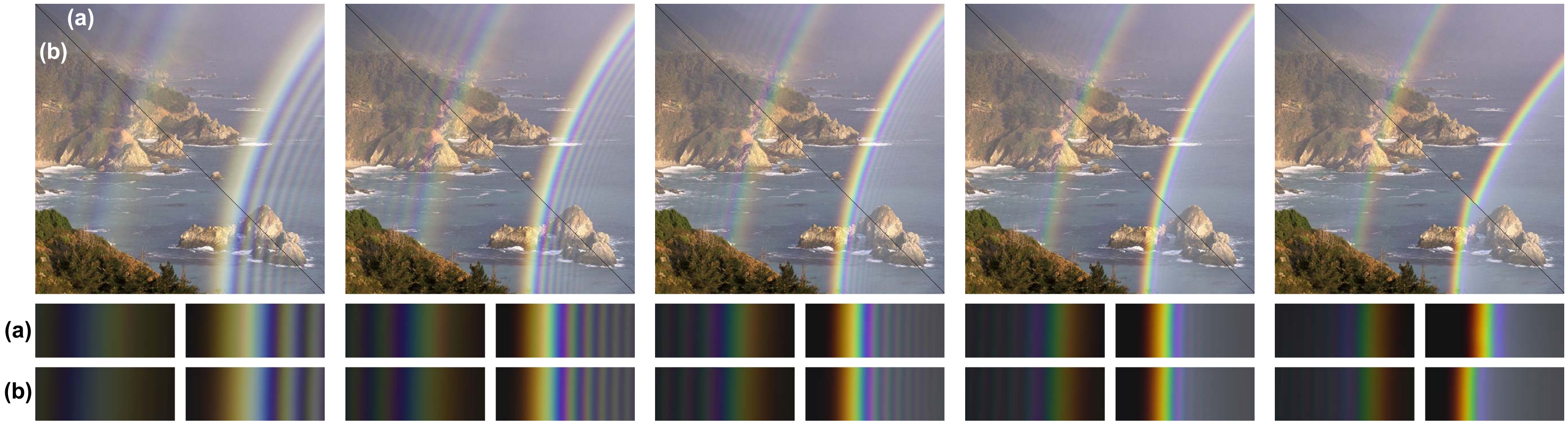


Figure 3.14: Comparison between our method **(a)** and Lorenz-Mie theory **(b)** for different water drop sizes. Top: Renders. Bottom: Plots of the phase function for the regions of the primary and secondary rainbows. From left to right: Water drops of radius 0.1mm, 0.2mm 0.3mm, 0.4mm and 0.5mm, respectively. Our method matches Lorenz-Mie theory for small water drops, which are spherical, but predicts different behavior as the radius increases by accounting for non-spherical drop shapes.

3.4.4 Approximating Diffraction

A fundamental problem with using ray-tracing techniques to simulate rainbows lies in the fact that geometric optics predicts *infinite* intensity at the rainbow angle with a very abrupt transition to zero intensity, as shown by the blue curve in Figure 3.12. In contrast, Lorenz-Mie theory predicts that maximum intensity occurs slightly above the geometric rainbow angle (see green curve at about 138.5° in Figure 3.12). Note also that this transition is softened so the intensity at the geometric rainbow angle is less than the maximum intensity, with some light being scattered into the zone below the geometric rainbow angle where no geometric rays can penetrate.

This process is very similar to diffraction by a knife edge in which some light appears in the shadow zone. An accurate calculation of the effect of diffraction on the rainbow light field would require the application of the Huygens-Fresnel principle for each differential point on each wavefront, which is time-consuming and impractical. Such techniques would also be able to predict the supernumerary arcs, but this is not necessary as Figure 3.14 shows that the supernumerary arcs predicted by our ray-tracing technique are already in very close agreement with Lorenz-Mie theory. Hence, we need only to address the diffraction effect, which can be efficiently approximated by performing a post process on the computed phase function, by first identifying very sharp transitions in intensity at a given wavelength and then smoothing out the sharp peaks by applying a domain specific kernel. The size of the chosen kernel depends on the size of the water drop. For efficiency reasons, we choose a simple Gaussian kernel, summarized in Table 3.3 for different radii. The values have been obtained from Lorenz-Mie theory for spherical drops, which our results show offer a good approximation. For the secondary rainbow we double the standard deviation of the kernel, to account for the fact that light has been reflected twice inside the water drop.

The addition of the diffraction filter produces some fairly subtle changes in our simulations, as can be seen in the simulations shown in Figure 3.12. In essence, the

Table 3.3: Standard deviation of the Gaussian filter diffraction approximation for various water drop sizes.

Radius (mm)	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
σ (degrees)	0.70	0.45	0.30	0.25	0.22	0.20	0.18	0.17	0.16	0.15

diffraction filter softens the transitions near the rainbow angle thus giving a better match to the Lorenz-Mie simulations. It is important to acknowledge that the parameters for our diffraction filter have not been thoroughly validated. Further work based on the application of the Huygens-Fresnel principle could overcome such concerns, but the increased accuracy of such techniques would be outweighed by an immense increase in computational complexity. In these circumstances, the diffraction filter seems to be a sensible approximation that adequately addresses a fundamental limitation of ray optics.

3.5 Results

We have used our technique to simulate several phase functions, and then used those phase functions to render images depicting various types of rainbows. Unless stated otherwise, each of the results shown on this section has been simulated by casting rays from a 3000 × 3000 grid for each wavelength, uniformly sampling 33 different wavelengths between 380 and 720 nm. The resulting phase functions were sampled at an angular resolution of 1800 × 14400 (which is dense enough to account for the cusp of the rainbow and the high frequency details of interference) and stored on disk. Rainbows are obtained by ray marching and computing single scattering along the volume, importance-sampling the sun (which is modeled as a disc subtending a solid angle of 0.5°). On an Intel(R) Xeon(R) CPU X5570 at 2.93GHz, using 8GB of RAM, our simulations took an average of 350 minutes to compute for all 33 wavelengths.

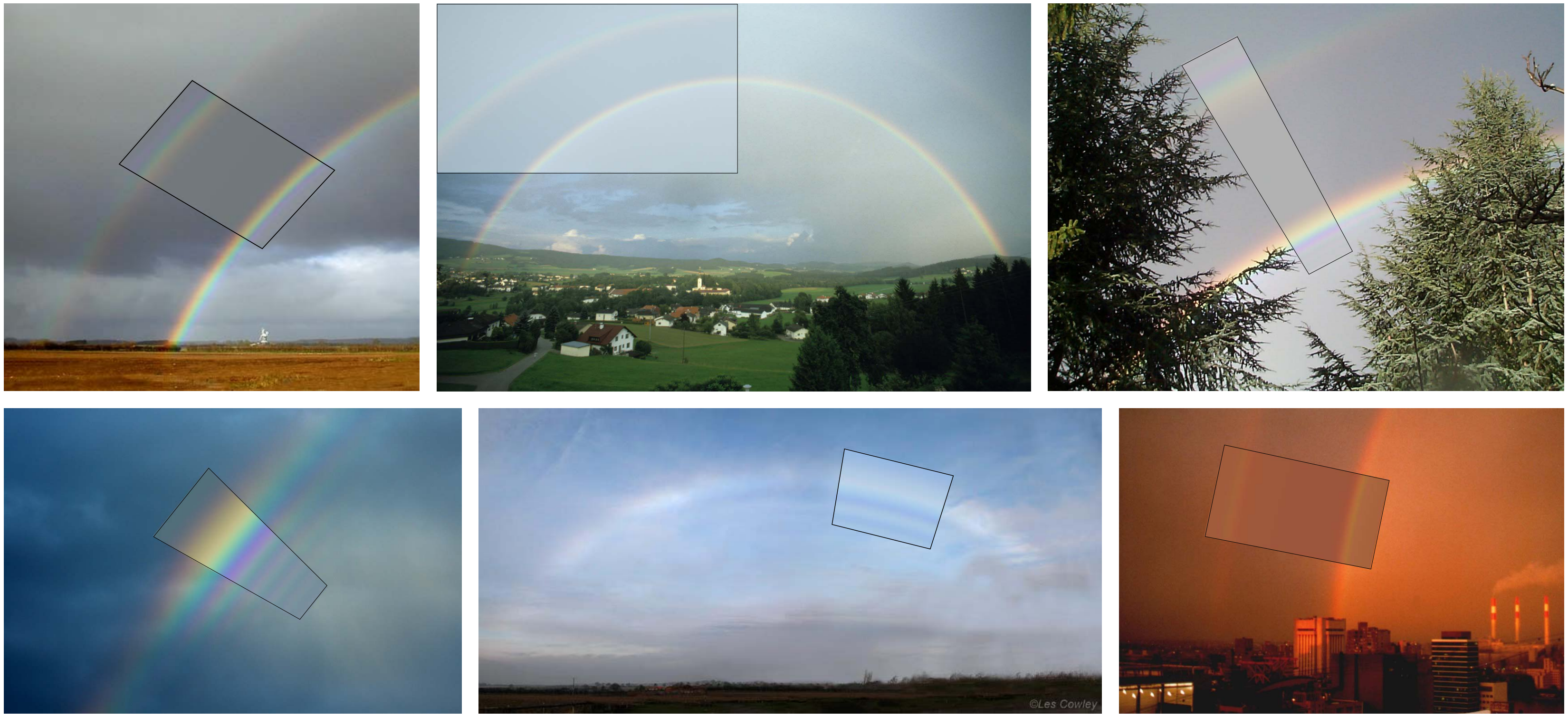


Figure 3.15: The inserts in these images show how our model can reproduce the rainbows in the underlying photographs. Only the background color of the insert has been matched to the specific photograph. Top row, from left to right: double rainbow (background reproduced with permission ©Les Cowley [Cowa]), full double rainbow (background reproduced with permission ©Karl Kaiser [Kai]) and supernumerary bows. Bottom row, from left to right: Multiple supernumerary bows, cloud bow (background reproduced with permission ©Les Cowley [Cowb]) and red bow.

Table 3.4: The list of parameters used to produce the rendering results in Figure 3.15

Figure	Top Left	Top Middle	Top Right	Bottom Left	Bottom Middle	Bottom Right
Water Drop Size	0.4 mm	0.4 mm	0.3 mm	0.3 mm	0.1 mm	0.4 mm
FOV	20^o	100^o	30^o	10^o	100^o	30^o
Lens Type	Rectilinear	Fisheye	Rectilinear	Fisheye	Rectilinear	Rectilinear
Background Color	(107,114,118)	(183,202,212)	(172,172,172)	(69,99,112)	(141,180,223)	(154,83,58)
Intensity	55%	100%	90%	80%	60%	80%
Illumination	D65	D65	D65	D65	D65	Rayleigh Scattering Applied to D65

To validate our algorithm, we simulated the phase function of spherical water drops of different sizes, comparing our results with the predictions of Lorenz-Mie theory. Figure 3.12 shows this comparison for a $0.4\,mm$ spherical water drop on a log-scale: purple represents dispersion, the pure geometric interpretation of the phase function of the rainbow; blue adds interference, and therefore the oscillations of the supernumerary arcs appear; green adds diffraction, eliminating the high intensity peaks at the geometric rainbow angle; and red represents the simulation from Lorenz-Mie theory. Notice the similarity between the green line (our complete simulation) and the red line (Lorenz-Mie simulation). The main differences are observed in the Alexander band, due to our diffraction approximation. Figure 3.14 shows rainbow renderings from the simulated phase functions, again exploring variations in size; for small sizes, where drops can be considered spherical, our results match Lorenz-Mie's predictions. However, larger drop sizes ($0.5\,mm$ in the figure) stop being spherical and consequently our algorithm predicts a different behavior.

Our method accurately reproduces several rainbow-related phenomena seen in nature. For instance, our algorithm can trivially reproduce the primary and secondary rainbows as seen in Figure 3.15 top-left and top-middle). Also, by simulating interference we are able to simulate supernumerary bows (Figure 3.15 top-right and bottom-left). By including also the effect of diffraction on the rainbow, we can simulate phenomena such as the cloud bow (Figure 3.15 bottom middle) in which the colors of the rainbow disappear into a whitish bow. Simulating the effect of Rayleigh scattering allows us to mimic the effect of a sunset on a rainbow (Figure 3.15 bottom-right).

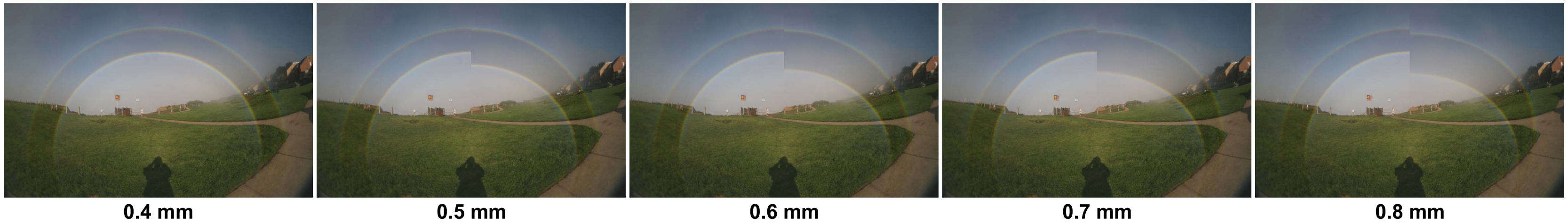


Figure 3.16: Comparison of renderings of rainbows owed to different water drop radii between Lorenz-Mie (left region on each image) and our solution (right region of each image). As the 0.4mm radius water drop is spherical, both algorithms lead to equal phase functions. As the particle gets bigger, the geometry becomes non-spherical and therefore Lorenz-Mie is unable to simulate it, while our solution takes it into account. Notice, also, that the variation on the secondary rainbow is quite unnoticeable compared to the variation on the primary rainbow, in agreement with the formation of twinned rainbows.

When superimposing simulations on images of natural rainbows, as in Figure 3.15, it is necessary to estimate the focal length of the camera lens. Fortunately, this information is often available in the EXIF data embedded in digital images. The angular performance of most camera lenses can be approximated by a rectilinear mapping function, except for fisheye lenses which are better modeled by equidistant, equal-area, stereographic or orthographic mapping functions. However, even with full information about the camera and its lens, it is also necessary to know the aiming point of the camera relative to the direction of the Sun or the anti-solar point. In practice, some of this information is typically missing (along with the obvious fact that the photographer is unlikely to provide any information about the size of the water drops causing the rainbows). Consequently, the simulation parameters generally need to be adjusted by trial and error to get a good match with the original image. The parameters used to produce the rendering results in Figure 3.15 is listed in Table 3.4.

Figure 3.1 shows an overview of the different rainbow-related phenomena our algorithm is able to simulate.

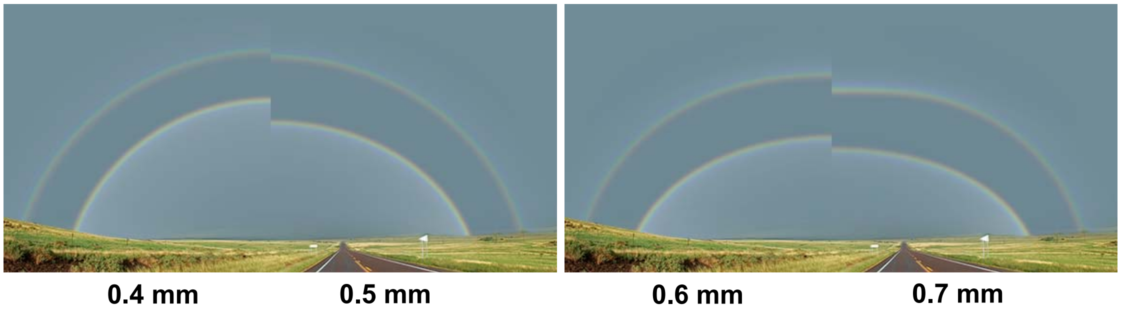
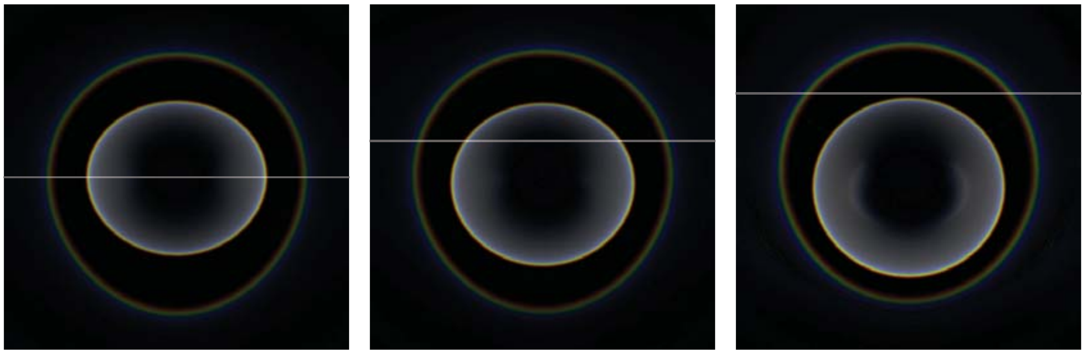


Figure 3.17: The effect of different water drop radii on the apparent geometry of the rainbow.

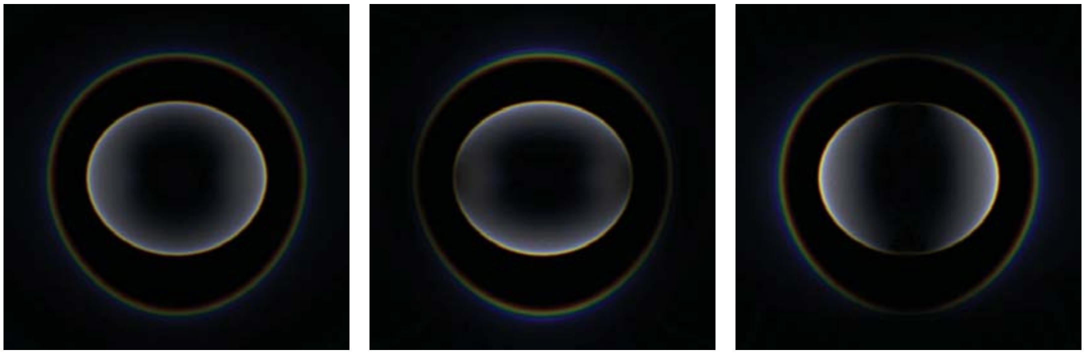
As discussed in Section 3.2.4, as water drops get larger they get deformed due to the impact of air resistance. This drastically affects the appearance of the final rainbow as shown in Figure 3.16. Producing these physically accurate phase functions for large water drops is, to our knowledge, not possible using any other method. Figure 3.17 shows the effect of the size of the water drop on the apparent geometry of a full rainbow.



(a) 0^o (b) 20^o (c) 40^o

Figure 3.18: The effect of the inclination of the sun on a non-spherical water drop (radius 0.5mm) alters the aparent geometry of the rainbow. This would not be the case for spherical water drops. The gray line indicates the horizon line.

While the phase function for spherical water drops is invariant to the inclination of the sun, non-spherical water drops produce very different phase functions for each incident direction of light. In other words, rainbows are actually the result of an anisotropic phase function within an anisotropic medium. Figure 3.18 shows the effect of the inclination of the sun on a $0.5\,mm$ non-spherical particle. We have set up the viewing direction to be parallel to the direction from the center of the sun, and we show the full (theoretical) 360° rainbow. For reference, the gray line indicates the horizon. Note that a rainbow due to spherical water drops would look identical in all these images.



(a) Circular (b) Horizontal (c) Vertical

Figure 3.19: The effect of the different polarization states on the perception of a rainbow from 0.5mm radius non-spherical water drop.

Furthermore, our algorithm naturally takes into account polarization as well. This enables us to explore the effects of different light polarization states and filters on the light field of the rainbow given by a water drop of arbitrary geometry. Figure 3.19 shows an example of this for circular, vertical and horizontal polarizations.

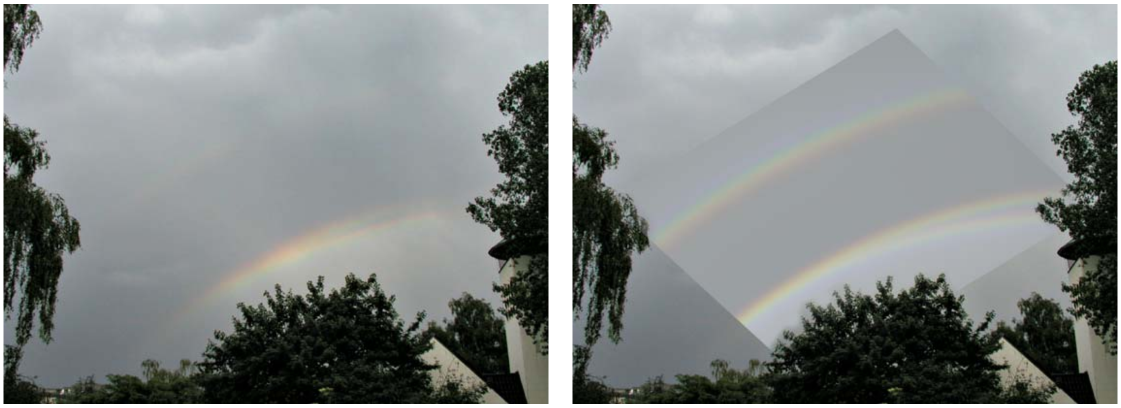


Figure 3.20: Left: Photograph of a twinned rainbow, reproduced with permission ©Benjamin Khne [Kh]. Right: Twinned rainbow simulated using our algorithm, generated from a two showers of 0.4mm and 0.45mm radius water drops, respectively.

Lastly, twinned rainbows can only be explained by a combination of two types of water drops with different sizes where at least one of them is non-spherical. Figure 3.20 shows a simulation of a twinned rainbow. To our knowledge, this is the first time that such a complex rainbow is simulated, based on an accurate computation of the phase function.

3.6 Conclusion and Future work

We have presented the first comprehensive model of rainbows suitable for computer graphics applications. We have validated it against Lorenz-Mie theory for the case of spherical water drops, and shown how it naturally overcomes the limitations of such theory.

As mentioned in Section 3.5, matching a reference photograph with rendering results is a manual process. A potential extension to our research would aim to use computer vision techniques to automate this process.

This research also opens other potential lines of investigation. Though we did not focus on performance in our work, we believe that our algorithm could be adapted to the GPU, greatly accelerating the phase function simulation. Furthermore, it would be interesting to explore other approaches for estimating focal lines and diffraction. Further development on our phase function simulator could lead to new and generalized global illumination algorithms, taking into account phenomena such as interference or diffraction. We foresee that a wide set of disciplines, such as meteorology or remote sensing, could benefit from our technique.

3.7 Acknowledgments

We would like to thank Maria Pilar for the rainbow picture in Figure 3.2 on the top right, Jurgen Schulze for referring us to the facility of Project GreenLight, Toshiya Hachisuka for constructive discussions and proofreading this paper, Krystle de Mesa for proofreading a draft of this paper. And finally, we like to thank the anonymous reviewers for their helpful suggestions and comments.

This work has been funded by NSF Project GreenLight (award no.: 0821155), a Marie Curie grant from the Seventh Framework Programme (grant agreement no.: 251415), the Spanish Ministry of Science and Technology (TIN2010-21543) and the Gobierno de Aragn (projects OTRI 2009/0411 and CTPP05/09).

The material in this chapter is, in part, a reproduction of the material submitted for publication in Iman Sadeghi, Adolfo Munoz, Philip Laven, Wojciech Jarosz, Francisco Seron, Diego Gutierrez, and Henrik Wann Jensen. “Physically-Based Simulation of Rainbows.” Conditionally Accepted to *ACM Transactions on Graphics*, 2011. The dissertation author was the primary investigator and author of this paper.

4

Light Interactions with Microcylinders

HAIR appearance plays an important role in character personalization and therefore is very important in creative environments. Despite much research on physically based hair rendering, it is currently challenging to benefit from this work mainly because physically based shading models do not offer artist friendly controls. Consequently, much production work so far has used ad hoc shaders that are easier to control, but often lack the richness seen in real hair.

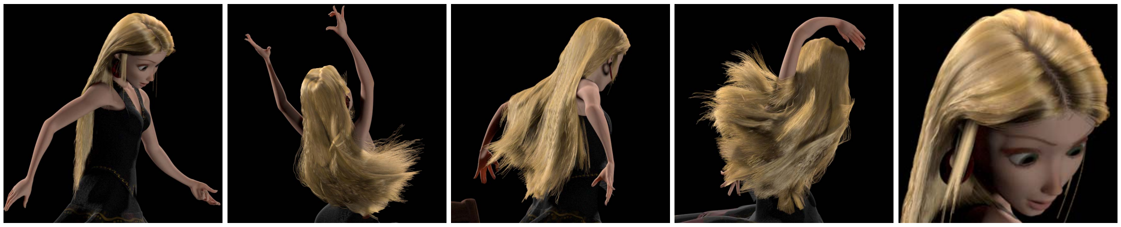


Figure 4.1: Rendering results using our novel hair shading system for different lighting and viewing directions. For film production it is important to see that the overall appearance is maintained for both still and animated images, as shown here.

In this chapter we show that physically based shading models fail to provide intuitive artist controls and we introduce a novel approach for creating an art-directable hair shading model from existing physically based models. Through auser study we show that this system is easier to use compared to existing systems. Our shader has been integrated into the production pipeline at the Walt Disney Animation Studios and has been used in the production of the animated feature film Tangled.

4.1 Introduction

Almost all characters in movies, games and other digitally created content have a representation of hair or fur on their bodies. Our eyes are very sensitive to the appearance of hair and we can observe subtle inaccuracies in its appearance. In fact, hair appearance has been shown to be one of the most important features of avatar personalization [DWYW09]. As such, it is critical to be able to render good looking hair and fur. In this context, it is important to recognize that "good looking" does not necessarily imply scientifically accurate. The appearance is the result of a creative process and the most important criteria is that it is aesthetically pleasing and that it fits within the universe of the character. This definition subsumes "photo-realistic" as a special case, but in general it is much broader.

Hair rendering is challenging because it requires capturing the complex behavior of light scattering events inside the hair volume, which is computationally very expensive. There has been much research on physically based hair rendering, but it remains difficult for artists to benefit from these results. The main drawback of using physically based shaders in creative endeavors is the lack of suitable controls. Consequently, most production work so far has used ad hoc shaders which are typically more art-directable. However, ad hoc shaders fail to capture the details of light scattering inside the hair volume and often produce inconsistent results under different lighting conditions. Overall, this leads to a less rich appearance which then limits the universe in which the characters can live.

In the following sections we illustrate how physically based shaders fail to satisfy the controllability required in a creative environment. We subsequently present a novel approach that can produce an art-directable hair shading model using the physical properties of hair fibers. Our model also handles different lighting conditions while giving full control over all visually important aspects of the hair appearance to the artists. In practice it has been shown to be versatile enough to handle all types of hair as well as fur. Our evaluation shows that using our shading model, artists tend to achieve the desired appearance more easily compared to both a physically based hair shader and an ad hoc hair shader which has previously been used in feature film production.

The outline of the remainder of the chapter is as follows. In the next section we consider related work, while Section 4.4 defines the controls needed by artists. We also explain why physically based shading models fail to satisfy these needs. In Section 4.5 we present our novel approach for producing an art-directable hair shading model based on existing physically based models. In Section 4.6 we show how we have applied our approach to both the single scattering and multiple scattering components of hair. In Section 4.7 we present the implementation details of our multiple scattering component. We proceed by presenting some rendering results in Section 4.8. The results of our evaluation are summarized in Section 4.9, and we end with the conclusion and future work in Section 4.10.

4.2 Background Theory

In this section we summarize important background knowledge that is needed throughout the chapter.

4.2.1 Notations of Hair Geometry

We start by summarizing the notations used in the hair rendering community and throughout this chapter. These notations are mainly taken from the works by

Marschner et al. [MJC+03] and Zinke et al. [ZW07].

We treat hair fibers as smooth microcylinders. The tangent direction of the cylinder is shown by vector $\vec{t}$ which points toward the tip of the hair. The plane perpendicular to this direction is referred to as the *normal plane.* The direction of illumination is ω_i and the viewing direction or the direction in which the scattered light is being measured is ω_r. Both the viewing direction and the illumination direction vectors point away from the center point.

For parameterizing ω_i and ω_r we use spherical coordinates. The inclinations of ω_i and ω_r with the normal plane are referred to as θ_i and θ_r respectively. All directions where $\theta = 0$ lie on the normal plane. Also, $\theta = \pi/2$ represents the direction of t vector and $\theta = -\pi/2$ indicates the opposite direction of t.

The azimuthal angles of ω_i and ω_r around the hair fiber are denoted by ϕ_i and ϕ_r respectively. The relative azimuthal angle $\phi_r - \phi_i$ is denoted as ϕ. Due to the radial symetry of the cylinders the angle *phi* can be used to indicate the azimuthal angle around the hair fiber. Similarly, the half longitudinal angle θ_h and relative longitudinal angle θ_d are defined as $\frac{\theta_r+\theta_i}{2}$ and $\frac{\theta_r-\theta_i}{2}$ respectively. All mentioned geometry notations of light scattering from fibers are summarized in Figure 4.2.

4.2.2 Radiometry of Hair Fibers

Hair fibers can be treated as 1D curve entities. Therefore, the description of light scattering from hair fibers should be described in a different manner than the usual light scattering parameters for surface reflection. Recall from Section 2.3.2 that for 2D surfaces, we define the light scattering behavior as the conventional BRDF $f_r(\omega_i, \omega_r)$. The BRDF is defined to be the ratio of the surface radiance exiting the surface in direction ω_r to surface irradiance falling on the surface from a differential solid angle in the direction ω_i:

$$f_r(\omega_i, \omega_r) = \frac{dL_r(\omega_r)}{dE_i(\omega_i)} \tag{4.1}$$

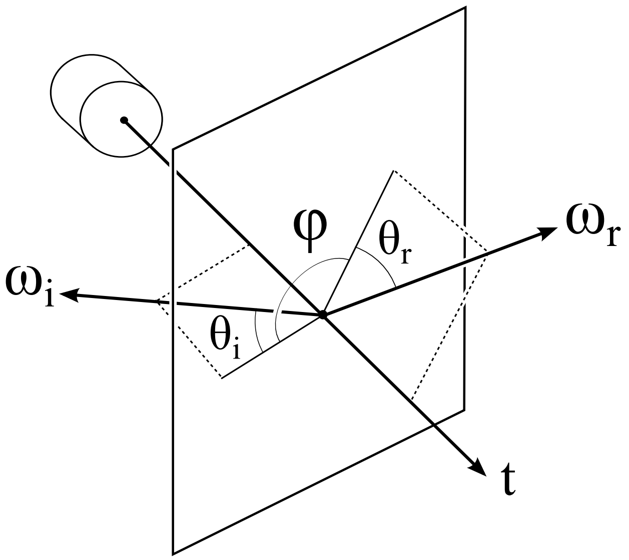


Figure 4.2: The geometry notations used for describing the light scattering form cylinders.

Therefore, the scattered radiance due to an incident radiance distribution $L_i(\omega_i)$ is calculated by the following integral over all possible incoming directions for ω_i on a hemisphere Ω:

$$L_r(\omega_r) \;=\; \int_{\Omega} f_r(\omega_i, \omega_r)\; L_i(\omega_i) \cos\theta_i \; d\omega_i \tag{4.2}$$

Similarly, Marschner et al. [MJC$^+$03] defined the light scattering from fibers $f_s(\omega_i, \omega_r)$ by defining the *curve radiance*, L_r^c, and *curve irradiance*, E_i^c. Curve radiance and curve irradiance are similar to the conventional radiance and irradiance but have different units. Curve radiance is defined as power per projected length per solid angle and curve irradiance is defined as power per unit length. Consequently, the *curve scattering function* f_s is the ratio of curve radiance exiting the curve in direction ω_r to curve irradiance falling on the curve from a differential solid angle in the direction ω_i:

$$f_s(\omega_i, \omega_r) = \frac{dL_r^c(\omega_r)}{dE_i^c(\omega_i)} \tag{4.3}$$

Thus, the outgoing curve radiance L_r^c, due to illumination from an incoming radiance distribution L_i, would be computed as the following integral over all possible directions for ω_i on a whole sphere $\Omega 4\pi$:

$$L_r^c(\omega_r) \;=\; D \int_{\Omega 4\pi} f_s(\omega_i, \omega_r)\, L_i(\omega_i) \cos\theta_i \, d\omega_i \tag{4.4}$$

In the above integral, D is the diameter of the hair fiber. This indicates that although identical fibers of different widths have the same scattering function, thick fibers appear brighter than thin fibers.

4.2.3 Physical Properties of Hair Fibers

Human hair fiber consists of three main components: the *cortex*, the *cuticle* and the *medulla* (see Figure 4.3). The cortex is the main part of the hair fiber and gives hair its physical strength. The cuticle covers the surface of the fiber and protects the inner cortex. The cuticle is the main interface between hair and light and is thus responsible for the behavior of light scattering by hair fibers. This thin layer is composed of flat cells that overlap on top of each other and has a structure similar to that of roof shingles [MJC$^+$03]. On the microscopic level, the surface of hair fibers appear as a nested set of cones (see Figure 4.3).

The overlapping structure of the hair cells causes the fiber's surface normals to be directed slightly away from the expected normal direction. Experiments have shown that these tilted cells shift the normal of the surface toward the root of the fiber by approximately five degrees [BS91; Rob94]. The medulla sometimes exists near the axis of the hair and its functionality is still unknown. The medulla and cortex contain pigments that absorb light differently at various different wavelengths. Together they produce the coloration of hair fibers. As observed by Robbins, the cross section of hair fibers can vary from circular to elliptical to irregular [Rob94]. He also discovered that hair is composed of amorphous proteins which act as transparent medium with index of refraction $\eta = 1.55$ (refer to Section 2.1.1for more

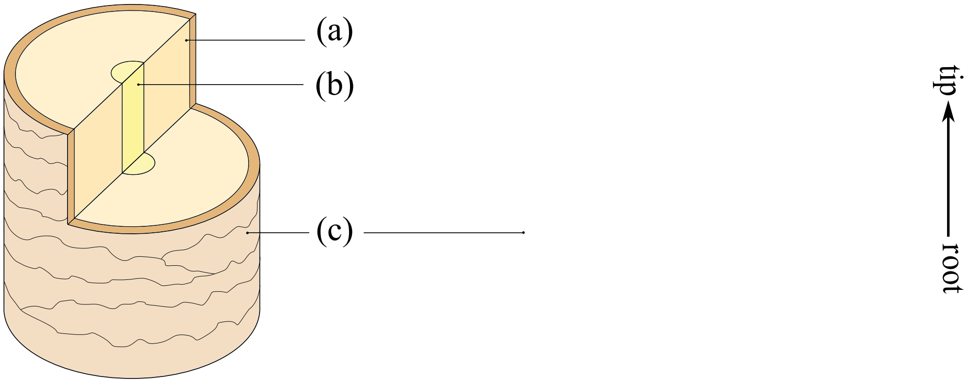


Figure 4.3: The structure of human hair fibers. Human hair consists of three main components: (a) the cortex, (b) the medulla, and (c) the cuticle. To the right is an electron micrograph of a hair fiber showing the structure of the cuticle layer. In this image the fiber is oriented with the root at the bottom and the tip at the top. The micrograph image is reprinted from [Rob94].

details).

4.3 Previous Work

In this section we will go over the previous work in hair rendering which includes physically based shading models, which are the main focus of academic community, and ad hoc shading models, which are mainly used in creative environments. In the rest of this section we will go over the most important work in each category.

4.3.1 Physically Based Shading Models

Extensive work has been done in the research community to capture the exact behavior of light scattering by human hair fibers.

The first prominent work in the field of hair rendering is the classical model of Kajiya and Kay [KK89]. They pointed out that the specular reflection of a parallel bundle of rays off of a smooth cylinder would lie within a cone with an apex angle equal to the incident angle. Consider the intersection of a parallel beam of light with a cylinder. Surface normals on the cylinder point in all directions around the hair and lie on the normal plane. Therefore, the reflected light is independent of the azimuthal angle of the incident direction. As a result, all the reflected rays will make the same angle with the normal plane and hence lie on a cone (see Figure 4.4). Kajiya and Kay considered constant intensity around this cone of reflections.

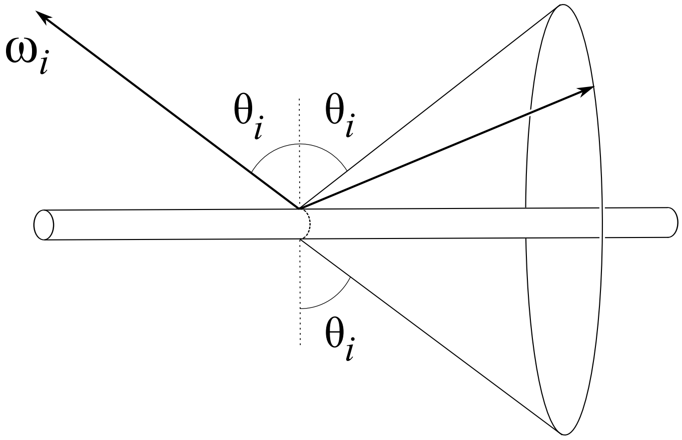


Figure 4.4: Reflection cone from a smooth cylinder. Kajiya and Kay pointed out that light reflections off of a cylinder form a cone.

Goldman [Gol97] and Kim [Kim02] extended the model of Kajiya and Kay by introducing azimuthal dependencies for the amount of light that gets scattered on the reflection cone.

In 2003, Marschner et al. [MJC$^+$03] presented a physically based scattering model for rendering hair fibers which remains the basis of all modern shading models in the field of hair rendering. In addition to developing a novel shading model, they have also presented the most comprehensive light scattering measurements from hair fibers.

Marschner et al. [MJC$^+$03] noted that when a ray of light interacts with a cylindrical surface, it will leave the cylinder on the reflection cone regardless of the number and sequence of refractions and reflections it undergoes. They also used *Bravais's law* to show that one may decompose the scattering function of a hair fibers into a longitudinal function (dependent on θ angles) and an azimuthal function (dependent on ϕ) which is dependent on a new effective of refraction. This new index of refraction is a function of longitudinal angle *theta* and is known as the *Bravais index*. For more information regarding the Bravais's law refer to [TRI70].

This observation enabled them to define their hair scattering function as a combination three longitudinal functions $M(\theta)$ and three azimuthal functions $N(\phi)$ by adjusting the effective index of refraction. They described the separation of the surface reflection highlight (R), transmitted component (TT) and internal-reflection component (TRT). They also presented an analytical analysis for predicting azimuthal variation based on the ray optics of smooth cylinders.

Marschner et al.'s improvements on azimuthal scattering functions include the following: predicting the intensity of reflected light based on the ray optics of a circular cross section, considering the Fresnel term (Section 2.2) when light interacts with the boundary of the fiber, and accounting the volume absorption (Section 2.2.5) of internal medium. Figure 4.5 shows the light interactions with a smooth cylinder in the azimuthal direction. This diagram is very similar to the depiction of light scattering from spherical water drops (refer to Section 3.2.1). However, cylindrical geometries show an extremely different behavior in the longitudinal direction.

For the longitudinal direction, Marschner et al. considered the effect of tilted cuticles and explained the separation of different highlights. As previously mentioned, the cuticles are tilted towards the root of the hair (approximately by 5^o) and shift the reflection cones of the primary and secondary (internally reflected) highlight slightly off the ideal mirror reflection cone. As illustrated in Figure 4.5, this slight shift in the normals' direction will result in shifts in opposite direction for the R and TRT components, causing the separation of the primary and secondary highlights respectively. The primary highlight is always shifted towards the root of the hair, whereas secondary highlight is shifted towards the tip of the hair. Therefore the primary highlight always appears below the secondary highlight.

Zinke et al. [ZSW04; ZW07] extended the model of Marschner et al. for close-

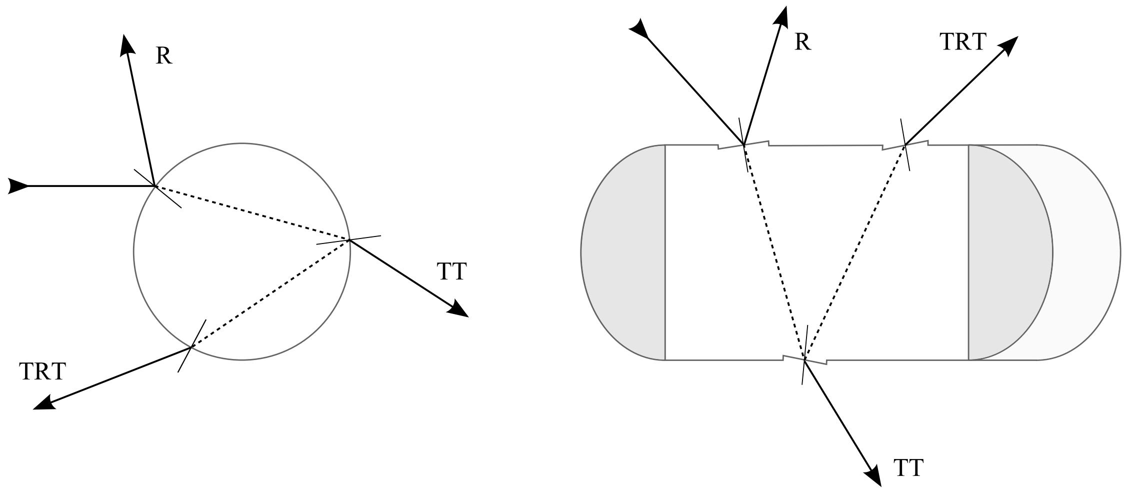


Figure 4.5: Light reflections off of a cylindrical hair fiber in the azimuthal (left) and longitudinal (right) directions. Tilted cuticles on the surface of the hair fiber deviate the direction of light reflections and refractions and cause the separation of primary (R) and secondary (TRT) highlights.

up renderings by considering the effect of microscale azimuthal and longitudinal offsets. To see the evolution of hair shading models with regards to azimuthal and longitudinal scattering functions, refer to Figure 4.6 and 4.7 respectively. For a survey on hair rendering refer to [WBK$^+$07].

There have been many attempts to simulate multiple scattering in hair rendering by using Monte Carlo path tracing [ZSW04; ZW07]. Path tracing approaches are computationally expensive and their rendering result converges very slowly. There have been two concurrent works by [MM06] and [ZW06] which try to simulate the effect of multiple scattering in hair rendering based on photon mapping [Jen01]. Both of these methods extend the conventional photon mapping algorithm to make it more suitable for the complex optical and geometrical properties of hair fibers.

In 2008, Zinke et al. [ZYWK08] presented an approximation method for computing the multiple scattering of light inside a hair volume. Their work is similar in spirit to the works in the field of participating media [PAT$^+$04] and rendering translucent materials [CTW$^+$04]. They introduced the concept of *dual scattering*, which splits the multiple scattering component into *global multiple scattering* and *local multiple*

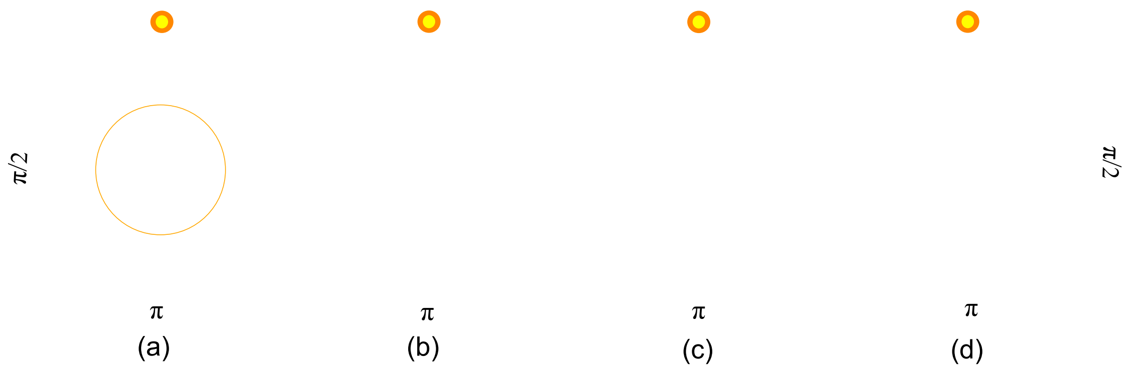


Figure 4.6: Evolution of azimuthal scattering functions. The values around the circles indicate the value of the azimuthal angle ϕ. (a) The azimuthal independent model of Kajiya and Kay. (b) The effect of Goldman's refinements to the Kajiya and Kay's model. (c) The combined phase function of Kim's model. The upper lobe is the reflected component and the lower sharper lobe is the transmitted component. (d) The physically based model of Marschner et al. The gray plot in Marschner's model resembling a flipped heart is the R component, the three colored lower lobes represent the TT component, and the three colored upper lobes represent the TRT component (glints).

scattering components. The global multiple scattering component is responsible for computing the irradiance arriving at the neighborhood of a point inside the hair volume and the local multiple scattering component approximates the multiple scattering of incoming irradiance within this local neighborhood.

All of these methods accentuate the physical correctness of the results and do not consider the controllability of the hair shading model. Unfortunately, matching a desired appearance by tweaking the physically based parameters (e.g. absorption coefficient, and index of refraction) is a time-consuming and tedious task [ZRL+09; BPvdP+09].

In addition, there have been efforts to estimate the values of physically based parameters by analyzing a single photograph [ZRL+09; BPvdP+09]. These methods enable artists to render hair with an appearance similar to a photographic reference. However, they do not provide artists any controls for further adjustments.

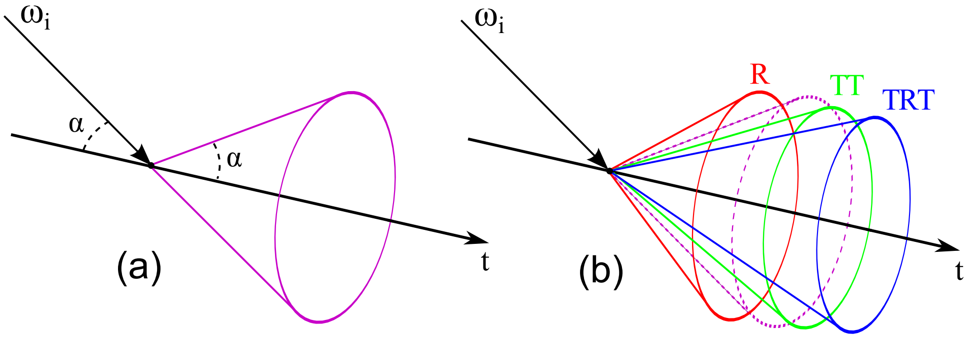


Figure 4.7: Evolution of longitudinal functions: (a) Kajiya and Kay defined the reflection cone and it was used by Goldman and Kim in their shading model. (b) Marschner et al. introduced the separation of different scattering components on three different cones. Their longitudinal scattering function was later used by Zinke.

Also, they can only produce results for which there is already a photographic reference. In practice, art direction often goes beyond what can be captured in a photo from the real world, so it is important for a hair shader to be able to extrapolate beyond the physical range.

Furthermore, it is important to realize that a physically accurate model requires equally accurate input to generate realistic results. When the goal is to create aesthetically pleasing imagery rather than a scientific simulation of light interaction with hair, it may therefore be necessary to allow for non-physical behavior in order to compensate for imperfect input-geometry, animation, or lighting (See Figure 4.8).

4.3.2 Ad Hoc Shading Models

In addition to physically based shading models, there have been efforts towards implementing ad hoc shaders in production environments.

Goldman [Gol97] introduced his *fakefur* rendering method by modifying the model of Kajiya and Kay. He provided simple parameters for controlling the amount of

Figure 4.8: Some visual development art references from the animated feature film Tangled. When the goal is to create aesthetically pleasing imagery rather than a scientific simulation, it may be necessary to allow for non-physical behavior for the shading model. © Disney Enterprises, Inc.

light scattering in frontlit and backlit scenes in order to render more realistic fur (see Figure 4.9).

Other hair rendering methods have been introduced that are based on simple shading models like the Lambertian model [AGL$^+$00; AGP$^+$02] or the Kajiya and Kay model [AGP$^+$01; Neu04]. These methods have introduced numerous tricks to handle the backlighting situation better as well as finding a hair tangent based on the surface normal and the hair geometry to get more pleasing results. Petrovic et al. [PHA05] presented a volumetric representation that was easier for the artist to light. They used the model of Kajiya and Kay for shading.

These shaders have more intuitive and easy-to-understand controls and they can produce art-directed appearances. However, as previously mentioned, the results are often inconsistent under different lighting conditions and tend to lack the visual complexity which characterizes real hair.

Our work is a form of Participatory Design, where all end users of a system are involved in the design process to ensure the usability of the final product [SN93]. Participatory design is a model of User-Centered Design [ND86; Nor02] and is

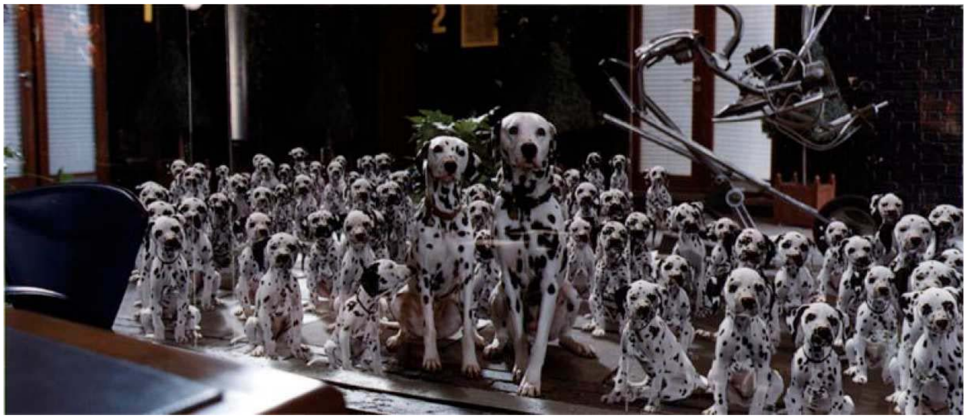


Figure 4.9: A frame from the film 101 Dalmatians. Only the two adult dalmatians were photographed and all of the puppies in the scene are rendered using Goldman's shading model. From [Gol97]. © Disney Enterprises, Inc.

known to be a challenging process in large development environments [Gru93]. To the best of our knowledge, we are the first to present a user-centered design approach for a physically-based rendering component. However, unlike most work in this area, we focus on reformulating the underlying computations rather than modifying the user interface to enhance the usability of the system.

4.4 Artist Friendly Control Parameters

Making it easy to control the behavior of shading modules is critical in most creative applications. Art directors usually have specific comments about the appearance of characters and how they want to modify the appearance, and it is important that artists have tools with the right controls to achieve this creative vision.

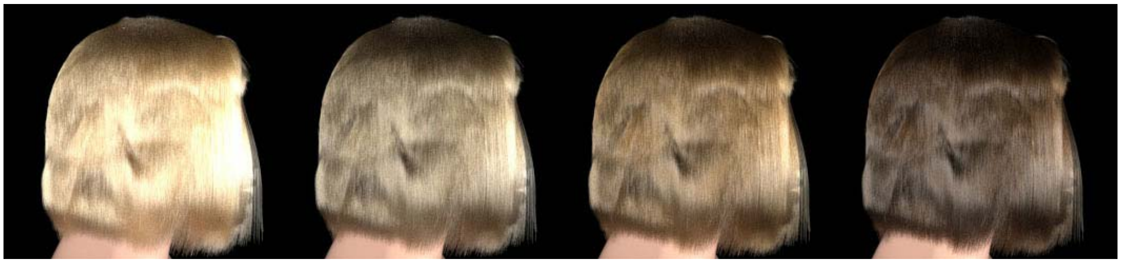
Having said that, it is important to note that there is no universal "artist friendly" system. Different artists often have different needs and concerns regarding the final appearance. These concerns vary over time, and they are different between production departments as well as between individuals. In particular, we have found the needs early on in the creative process to be different compared to those late in the process. Early on, when the design space is being explored, a few controls tend to be favored, while a lot of controls are desired late in the process when specific details are being tweaked.

In general, it is therefore impossible to satisfy the needs of all artists in one shading model. However, there are some simple criteria which are common among most users, and usually physically based shaders fail to satisfy these criteria as described below.

4.4.1 Intuitive behavior

The first requirement is that the control parameters should correspond to visually distinct features and behave predictably.

In the physically based world, the appearance of materials is being determined by intrinsic properties (e.g., index of refraction and absorption coefficients). These physically based properties have complex and unintuitive effects on the final hair appearance [ZRL$^+$09; BPvdP$^+$09]. This makes it very hard even for trained artists, to guess the shader parameter values in order to get a desirable appearance [MTLB05].



(0.03, 0.07, 0.15) (0.15, 0.2, 0.3) (0.2, 0.3, 0.5) (0.3, 0.6, 1.2)

Figure 4.10: Final hair colors and their corresponding RGB absorption coefficients. Coming up with appropriate values for the absorption coefficients to get a desired hair color is not intuitive. Image is reproduced from [ZRL$^+$09] © 2009 Association for Computing Machinery, Inc. Reprinted by permission.

One such example is hair color which is determined by absorption coefficients. These measure the wavelength based attenuation of light as it passes through the hair medium, but there is no obvious relation between the absorption coefficients and the hair color. Adding to the complexity is the fact that the final hair color (especially for light-colored hair) also depends on the number of light scattering events inside the hair volume. Figure 4.10 shows some hair rendering results and their corresponding absorption coefficient values.

A more intuitive control parameter for changing the color of hair would be a simple color variable that has direct impact on the color of hair. However, for a physically based shader this may not be easy to implement as the relationship is often highly non-linear especially when multiple scattering is involved.

The second requirement for the control parameters is that they should have intuitive names. Technical terms taken from the computer graphics literature are in general not artist-friendly and in some cases can be ambiguous and even misleading for artists. Terms like “azimuthal” and “longitudinal”, as an example, are technically accurate, but tend to be unfamiliar to many non-technical people. Another term, like “highlight” may be familiar to most but is also ambiguous since it can refer to where the light is reflecting or the result of having parts of the hair dyed.

Unfortunately, coming up with intuitive names is not a simple process with a unique answer. We came up with a list of names by brainstorming with our look-development and lighting artist, and we have listed these terms in Appendix B. However, this is by no means *the* correct set of names. The point is that the names should be chosen with the users in mind such that they can relate to them.

4.4.2 Decoupling

From an artist’s point of view, changes to one visually distinct feature (e.g. brightness of primary highlight, color of the secondary highlight) should not affect other visually distinct features. This is critical because in many cases art directors ask for a change in one feature and expect the rest of the appearance to remain unchanged.

Physical material properties are unfortunately inherently interconnected: Changing one physical property will in most cases affect all the visually distinct features of the final appearance. As an example, changing the index of refraction of hair will affect the color, intensity, and position of different highlights. Even if only one of these needed to change. See Figure 4.11 (top row) for a visualization of these effects.

Also, energy conservation forces any physically based scattering function to integrate to a value less than (or equal to) one. Therefore, if an artist makes one of the subcomponents of the scattering function very large, other subcomponents have to become smaller. As an example, if we increase the intensity of one of the highlights, the intensity of the other two highlights must be reduced to conserve energy. Similarly, if we consider a single highlight, increasing its width will reduce the observed intensity since the energy gets distributed over a larger area. See Figure 4.11 (middle row) for a visualization of this effect on the primary highlight. These coupled behaviors reduce art-directability and are undesirable from a usability point of view. Instead, in this example, we would like to be able to change the width of the highlight while keeping the color, intensity, and position of the highlight intact. See Figure 4.11 (bottom row).

4.4.3 Going beyond reality

Physically based models follow the rules of physics and can therefore not produce non-physical results. However, in the creative world the only limitation is the imagination, and art directors are often interested in appearances which are not feasible in the real world. To complicate matters, with a physically based shader if one attempts to modify the hair scattering functions to accommodate such non-physical requests, undesirable side effects may occur. For instance, if the modified scattering function is energy absorbent, the multiple scattering component (which

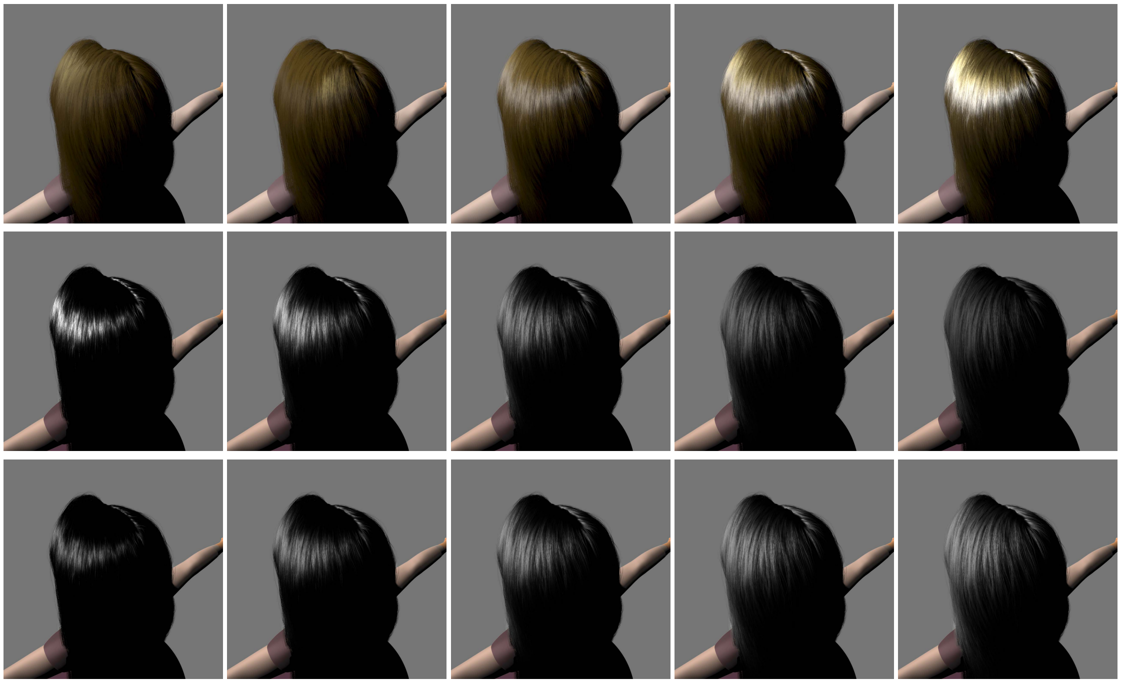


Figure 4.11: Comparison between coupled and decoupled control parameters. Top row : Changing the index of refraction in a physically based shader affects all visual components of the appearance at the same time and is an example of a coupled (and unintuitive) control parameter. Middle row : Increasing the width of a highlight in a physically based shader must reduce the intensity to preserve energy. This coupled behavior is undesirable from an artist's point of view. Bottom row : An example of decoupled control parameter for changing the width of a highlight which will not affect any other aspects of the highlight.

is very important for the overall hair color) might disappear. On the other hand, if the scattering function is energy producing, then the multiple scattering component will blow up (Figure 4.12). This happens because the overall energy gets increased with every bounce. These results, which are consistent with the rules of physics, are just as undesirable from an artist's point of view as the coupling and unintuitive behavior mentioned above. Our goal is therefore to create controls which cover the physically correct domain, but which extend seamlessly into the non-physical domain. For discussion on how we address this problem refer to Section 4.6.2.3.

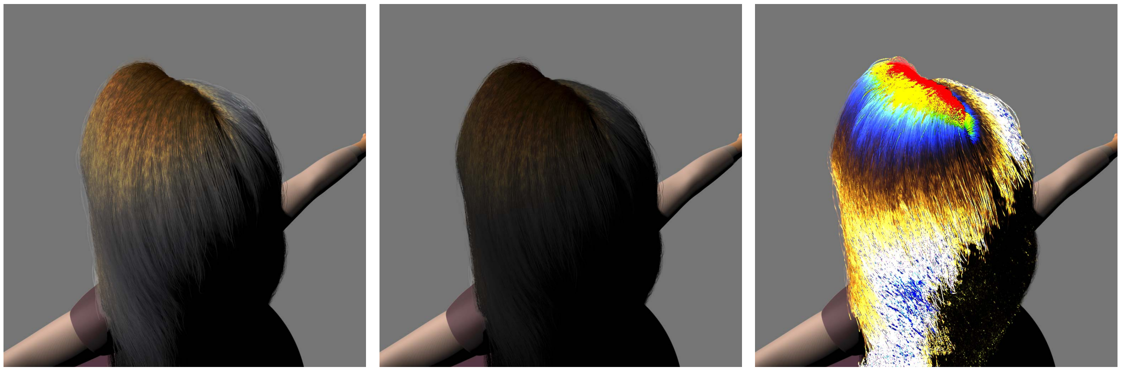


Figure 4.12: Unexpected behavior of a physically based shading model after some non-physical modifications to the underlying hair scattering functions. Left : A physically based setting for the shader will result in a reasonable multiple scattering component. Center : When the modified single scattering functions absorb energy the multiple scattering component might disappear. Right : When the single scattering functions produce energy, the multiple scattering component might blow up.

4.5 Our Approach

As explained above, the reason that physically based shading models have limited artist controls is that physically based scattering functions (f_s) are defined over the domain of material properties. We refer to these material properties as *Physically Based Controls (PBC)* which include parameters like index of refraction η, absorption coefficient σ_a, etc.

$$f_s = f(\omega_i, \omega_r, \sigma_a, \eta, ...) = f(\omega_i, \omega_r, \{PBC\}) \tag{4.5}$$

where ω_i and ω_r are lighting and viewing directions.

Our goal is to produce a pseudo scattering function f'_s that approximates f_s but is defined on a different domain of parameters which have intuitive visual meanings to the artists and are separate for all visually meaningful components. We refer to these intuitive, decoupled, and meaningful parameters as *Artist Friendly Controls (AFC)*.

$$f'_s = f(\omega_i, \omega_r, \{AFC\}) \approx f_s \tag{4.6}$$

The pseudo scattering function f'_s is not limited to the rules of physics and can produce a larger range of appearances than f_s. Therefore, f'_s can produce supernatural appearances as well as physically based results.

To produce f'_s from f_s, we propose the following steps:

1. Examination: Examine the exact behavior of a physically based scattering function f_s over the domain of some material properties $\{PBC\}$.

2. Decomposition: Decompose the behavior of f_s into visually separate and meaningful scattering sub-functions f_{s_i}. Defining meaningful subcomponents is a subjective task and should be done with the help of the artists who will be the end users of the system [ND86].

3. Defining AFCs: For each sub-component f_{s_i}, define artist friendly control parameters AFC_{ij} which are intuitive, decoupled, and visually meaningful for the artists. These AFCs define qualities like color, intensity, size, shape, and position of the visual features. This step is also subjective and should be done with the help of artists that will be using the system.

4. Reproduction: Create pseudo scattering functions f'_{s_i} that approximate the qualitative behavior of decomposed scattering functions f_{s_i} over the domain of $\{AFC_{ij}\}$.

5. Recombination: Combine the approximated pseudo scattering functions f'_{s_i} to get one pseudo scattering function f'_s. The final pseudo scattering function f'_s approximates f_s and is defined over the domain of artist friendly control parameters $\{AFC_{ij}\}$.

See Figure 4.13 for a schematic visualization of this approach.

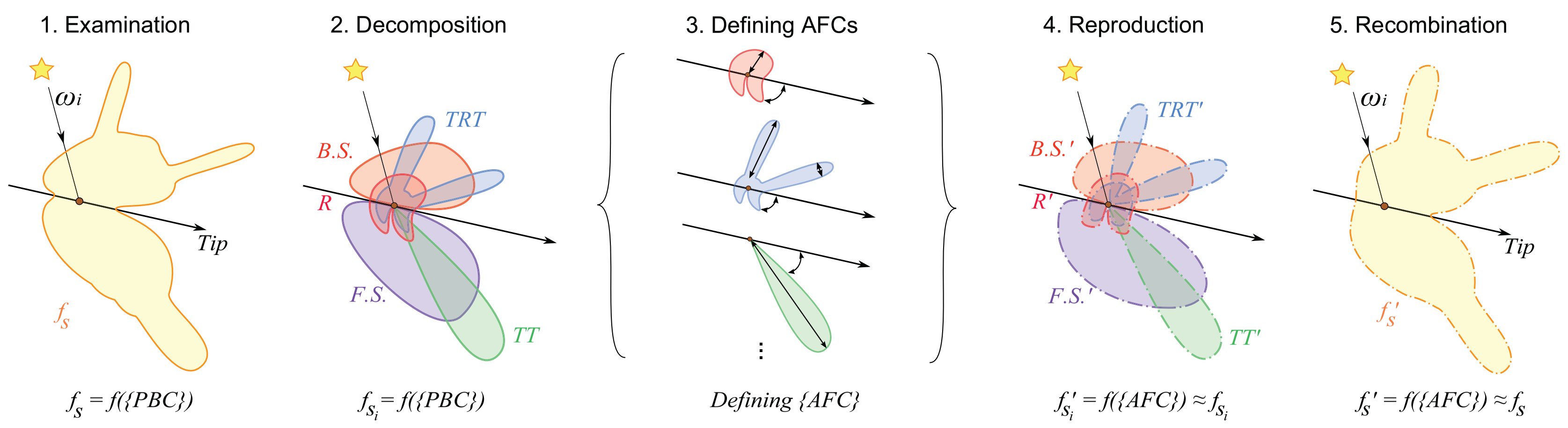


Figure 4.13: A schematic visualization of our approach . The first column shows the physically based hair scattering function f_s which is defined over a set of physically based parameters $\{PBC\}$. The second column shows decomposed scattering functions f_{s_i}. In the third column, meaningful artist friendly controls AFC_{ij} are being defined for each subcomponent. In the fourth column, new pseudo scattering functions f'_{s_i} are introduced to approximate each f_{s_i} function. These functions are defined over the domain of AFC_{ij}. In the last column, all of these pseudo scattering functions are combined to give us the final pseudo scattering function f'_s that approximates f_s.

4.6 Applying Our Approach

In this section we explain how we have applied our approach to the single and multiple scattering components.

4.6.1 Single Scattering

4.6.1.1 Examination

In computer graphics, the prominent work on single scattering properties of hair fibers is by Marschner et al. [MJC$^+$03]. According to their measurements, single scattering has three main subcomponents: 1) The light that reflects off the surface of hair (aka primary highlight), 2) light that has transmitted through the hair medium (aka transmission highlight), and 3) light that has been internally reflected off the inner surface of the hair (aka secondary highlight). We will refer to these components as R, TT, and TRT, respectively (Figure 4.14 left).

Due to the presence of tilted cuticles, these three components will be reflected in three different angles around the hair fiber, forming 3 different cones. The R component has the color of the light source and usually appears as a bright highlight. The TT component appears in back lighting situations and is the bright halo around the hair. The TRT component appears above the primary highlight and has the color of the hair. This component contains some randomized looking sharp peaks that are basically caustics formed as the light passes through the hair fibers. Their randomized appearance is due to the fact that hair fibers have elliptical cross sections and are oriented randomly.

Marschner et al. [MJC$^+$03] showed that one can decompose the scattering function of a hair fiber into three longitudinal functions $M(\theta)$ and three azimuthal functions $N(\phi)$. See Figure 4.14 for a qualitative visualization of these six functions. Marschner et al. [MJC$^+$03] defined the final hair scattering function f_s as:

$$f_s(\theta, \phi) = \sum_X M_X(\theta) N_X(\phi) / \cos^2 \theta \tag{4.7}$$

where subscript $X \in \{R, TT, TRT\}$ represents one of the three subcomponents.

The longitudinal scattering functions $M_X(\theta)$ have been modeled as unit-integral, zero-mean Gaussian functions. The variance of these Gaussian functions represents the longitudinal width of each highlight:

$$M_X(\theta) = g(\beta_X^2, \theta_h - \alpha_X) \tag{4.8}$$

Here g is a unit-integral zero-mean Gaussian function, β_X^2 represents the variance of the lobe, α_X represents its longitudinal shift, and θ_h is the longitudinal half angle between incoming and outgoing light directions.

Marschner et al. [MJC+03] proceed to compute the azimuthal scattering functions assuming that the hair fibers have circular cross sections. The important observation in this context is that the final shape of these scattering functions is relatively easy to characterize, and that it is qualitatively similar for different types of hair (Figure 4.14 right). The exception is the behavior of the glints which is very complex. However, for our purposes it is sufficient to use a simplified model as described below.

4.6.1.2 Decomposition

For the decomposition step, we asked a team of artists to identify appearance properties that they want to control. They came up with four components: primary highlight, secondary highlight, glints, and the rim light when the hair is backlit.

These four components align with the underlying physically based calculations very nicely. They are basically the R, TT, and TRT components where the TRT com-

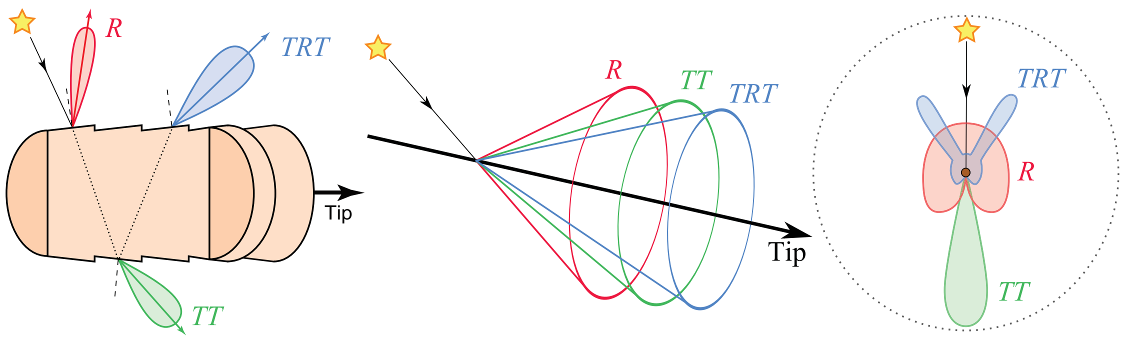


Figure 4.14: Single scattering subcomponents: R, TT, and TRT. (left) Three different paths that light can take after intersecting a hair fiber. (middle) Longitudinal scattering functions $M_x(\theta)$ which are three cones with different apex angles. (right) Azimuthal scattering functions $N_x(\phi)$.

ponent is being decomposed into two subcomponents: glints and TRT excluding the glints.

4.6.1.3 Defining AFCs

Defining decoupled and meaningful artist friendly controls for each component means defining qualities like color, intensity, size, shape, and position of each decomposed component. The artists came up with the following AFCs:

R: Color, intensity, longitudinal position, and longitudinal width.

TT: Color, intensity, longitudinal position, longitudinal width, and azimuthal width.

TRT minus glints: Color, intensity, longitudinal position, and longitudinal width.

Glints: Color, intensity, and frequency of appearance.

Please note that the decomposition and the choice of control parameters are subjective choices and could be done differently depending on the needs of the artists.

4.6.1.4 Reproduction

Our task is to approximately reproduce all of the decomposed subcomponents based on their defined AFCs.

We do this by simulating the longitudinal and azimuthal scattering functions separately. To simulate the longitudinal function, we use Gaussian functions similar to the original paper with the only difference being that the new functions are unit height instead of unit-area. This way, changing the width will not affect the brightness of the highlight. We have thus defined the pseudo longitudinal scattering functions as follows:

$$M'_X(\theta) = g'(\beta_X^2, \theta_h - \alpha_X) \tag{4.9}$$

where $X \in \{R, TT, TRT\}$, g' is a unit-height zero-mean Gaussian function, β_X^2 represents the longitudinal width of component X, and α_X is its longitudinal shift.

The azimuthal scattering functions are more complex and we have to simulate each one separately. We have to keep in mind that we want to keep the peak of these functions constant so that they will not affect the brightness of each component.

The azimuthal scattering function for the primary highlight $N_R(\phi)$ is like an upside down heart shape (Figure 4.15). We reproduce a simple approximation of this component according to the work by Kim [Kim02] as the equation below :

$$N'_R(\phi) = \cos(\phi/2) \qquad 0 < \phi < \pi \tag{4.10}$$

With this approximation we are ignoring the Fresnel term for simplicity but it can be added if more accurate results are desired.

The azimuthal scattering function of the transmission component N_{TT} is a sharp

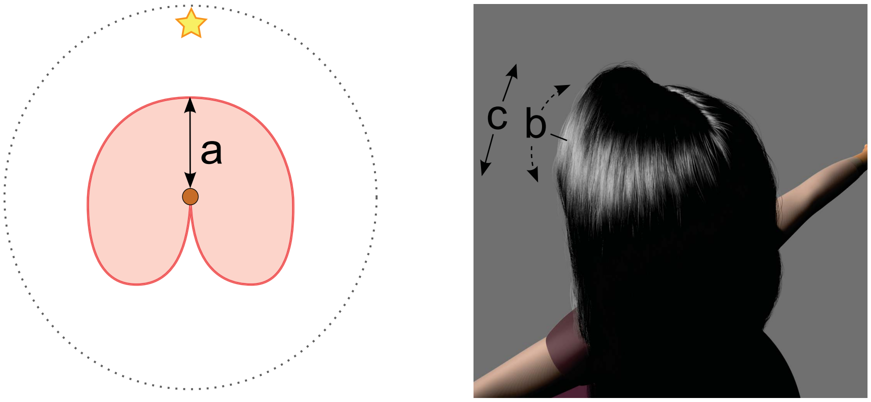


Figure 4.15: Visualizing the primary highlight's control parameters on its corresponding azimuthal scattering function (left) and a frontlit rendering (right). (a) The intensity I_R, (b) the longitudinal shift α_R, and (c) the longitudinal width β_R^2 of the primary highlight.

forward directed lobe (Figure 4.16). We simply reproduce it as a Gaussian with unit height and controllable azimuthal width:

$$N'_{TT} = g'(\gamma_{TT}^2, \pi - \phi), \tag{4.11}$$

where γ_{TT}^2 is the azimuthal width.

For the secondary highlight we have more control parameters because of glints. Due to the eccentricity of the human hair fibers, the number, intensity, and the azimuthal direction of the glints varies based on the orientation of each hair. However, since we are only concerned with the final visual impact of the glints, we assume that glints are two sharp peaks with the same intensity that are always coming back toward the incoming light direction. We add a random shift to their azimuthal direction to get the randomized appearance. This very simplified model for glints produces visually acceptable results for our purpose. We give the artist controls over the relative brightness of the glints and the frequency of their appearance.

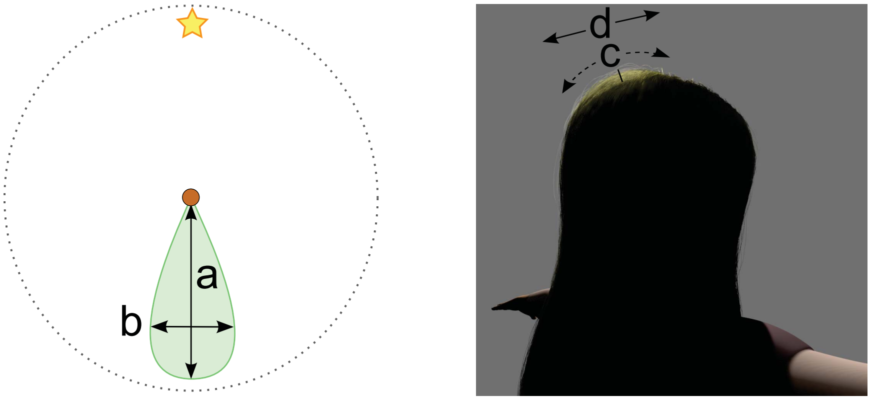


Figure 4.16: Visualizing the transmission component's control parameters on the corresponding azimuthal scattering function (left) and a backlit rendering (right). (a) The intensity I_{TT}, (b) the azimuthal width γ^2_{TT}, (c) the longitudinal shift α_{TT}, and (d) the longitudinal width β^2_{TT} of the transmission highlight.

$$N'_{TRT-G} = \cos(\phi/2) \tag{4.12}$$

$$N'_G = I_g g'(\gamma^2_g, G_{angle} - \phi) \tag{4.13}$$

$$N'_{TRT} = N'_{TRT-G} + N'_G \tag{4.14}$$

Here, I_g is the relative intensity of glints over the intensity of secondary highlight, and γ^2_g is the azimuthal width of the glints. Increasing the azimuthal width of the glints makes them appear more often and decreasing their width will reduce their presence. G_{angle} is the half angle between two glints. To give a randomize appearance to the glints, G_{angle} is different for each hair strand and has a randomized value between 30^o and 45^o. See Figure 4.17.

To embed the control for color and brightness of each component we simply multiply each one by a scalar variable and a color variable :

$$f'_X = C_X I_X M'_X(\theta) N'_X(\phi) \tag{4.15}$$

where $X \in \{R, TT, TRT\}$ and C_X and I_X are the color and intensity of component

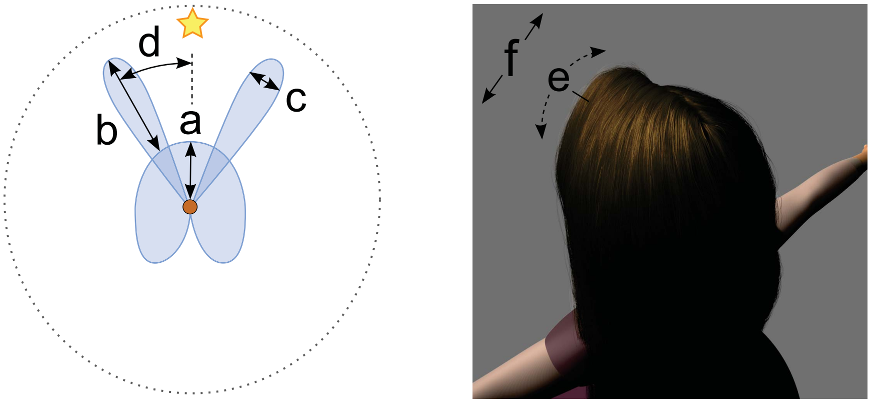


Figure 4.17: Visualizing the secondary highlight's control parameters on the corresponding azimuthal scattering function (left) and a frontlit rendering (right). (a) The intensity of the secondary highlight I_{TRT}, (b) the relative intensity of glints I_g, (c) the azimuthal width of the glints γ_g^2, (d) the half angle between the glints G_{angle}, (e) the longitudinal shift α_{TRT}, and (f) the longitudinal width β_{TRT}^2 of the secondary highlight.

X, respectively. In practice these values can be controlled manually, procedurally or through painted maps.

4.6.1.5 Recombination

To combine the results we have to add all the components together and divide by cos^2 to account for the projected solid angle of the specular cone [MJC+03].

$$f'_s = \sum_X f'_X / \cos^2(\theta) \tag{4.16}$$

Note that f'_s is not energy preserving, which we will discuss further in Section 4.6.2.3.

4.6.2 Multiple Scattering

Considering multiple scattered light is critical for correct reproduction of hair color, especially for light colored hair. To capture the exact behavior of multiple scattered light one needs elaborate methods like brute force path tracing [ZW07], photon mapping [MM06; ZW06], or other grid based approaches [MWM08]. Path tracing approaches are computationally expensive and their results converge very slowly. Photon mapping and grid based approaches are faster than path tracing methods but are still relatively expensive. All of these methods require ray tracing capabilities and are very costly in a production environment.

Another class of methods try to approximate the multiple scattering component by considering the physical properties of human hair fibers. The most prominent work in this category is the Dual Scattering model [ZYWK08]. This method is fast and relatively accurate, and with some considerations it can be used efficiently in production without the use of any extra data structures or any ray tracing steps [ST10].

We chose the Dual Scattering model as our physically based scattering function. Here we explain how we have applied our approach to this model to produce the pseudo scattering function for the multiple scattering component.

4.6.2.1 Examination

The Dual Scattering method approximates the multiple scattering function as a combination of two components: global multiple scattering and local multiple scattering.

Global multiple scattering accounts for the light that reaches the neighborhood of the shading point. It is dependent on the orientations of all the hairs between the light source and that point. It requires calculating the forward scattering transmittance and the spread of the light that reaches the shading point from all light sources. Global multiple scattering will be computed for different points separately.

Local multiple scattering approximates the scattering events within the local neighborhood of the shading point. It is only dependent on the longitudinal inclination of the hair strand at the shading point, and assumes that all surrounding hairs in the shading region have the same orientation and that there is an infinite number of them. For more details about the Dual Scattering method refer to the original paper [ZYWK08]. Decomposing the dual scattering component into meaningful components is not as straight-forward as it is for the single scattering component. To find meaningful components, we visualized all the terms involved in the computation of the final result of the model (Figure 4.18) and asked our artists to choose the ones that have intuitive meanings for them. The artists came up with two groups of components:

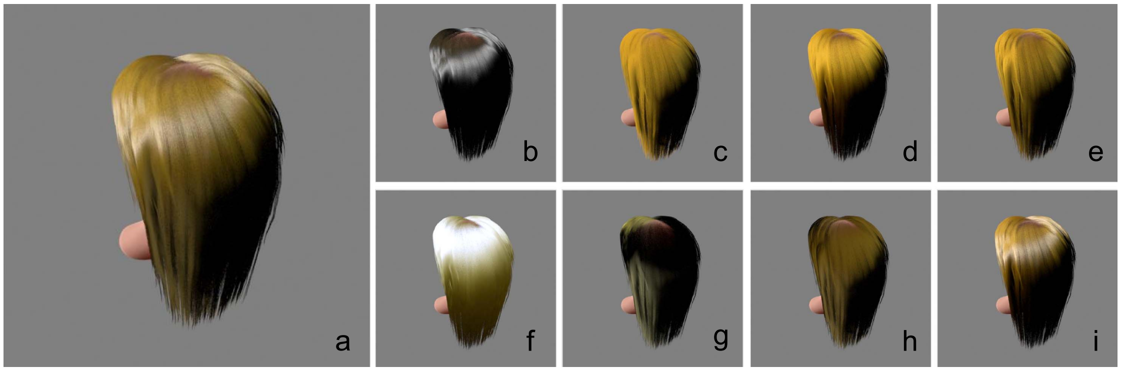


Figure 4.18: Different quantities involved in the computation of the (a) final results of the Dual Scattering method: (b) Single scattering components f_s, (c) average backscattering attenuation A_b, (d) multiple backscattering distribution function for direct lighting f_{back}^{direct}, (e) multiple backscattering distribution function for indirect lighting $f_{back}^{scatter}$, (f) average backscattering spread S_b, (g) single scattering for indirect lighting $f_s^{scatter}$, (h) $F^{scatter}$ term, and (i) F^{direct} term.

Forward Scattering component (F.S.) This includes the $f_s^{scatter}$ term from the dual scattering model, which computes the light that scatters forward and maintains its forward directionality inside the hair volume. This component is very important in backlit situations.

Backscattering component (B.S.) This includes the terms f_{back}^{direct} and $f_{back}^{scatter}$ from the dual scattering model. These are multiple backscattering distribution functions that represent the light that goes into the hair volume and comes back to the surface. f_{back}^{direct} computes this quantity for direct illumination and $f_{back}^{scatter}$ computes this for indirect illumination. Both of these components are smooth Gaussian functions in the Dual Scattering model.

Note that $f_{back}^{scatter}$ and f_{back}^{direct} are very similar quantities. f_{back}^{direct} is being used in the computation of F^{direct} while $f_{back}^{scatter}$ is being used in the computation of $F^{scatter}$ and accounts for the variance of forward scattering in the longitudinal directions $\bar{\sigma}_f^2$ (Figure 4.19 d and e). In the original paper [ZYWK08], the term f_{back} is being used for both of these quantities. For further discussion of this refer to [ST10].

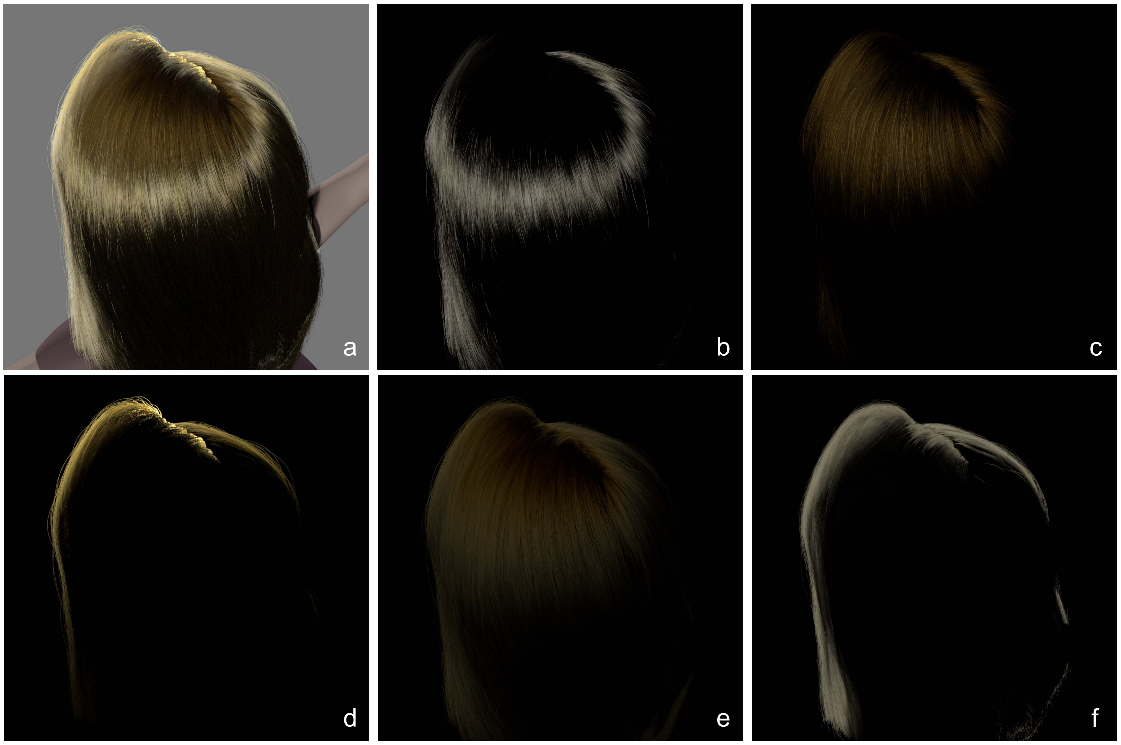


Figure 4.19: Visualizing the decomposed components of our hair shading model. (a) Final rendering result. (b) Primary highlight component R', (c) secondary highlight component TRT', (d) transmission component TT', (e) backscattering component $B.S.'$, and (f) forward scattering component $F.S.'$.

4.6.2.2 Defining AFCs

At this point we need to define artist friendly controls like the color, intensity, size, shape, and position for all of the decomposed components. However, all of these components are already indirectly defined by the values chosen for the single-scattering components. By overriding these values with artist defined values, we will lose the richness and details in the appearance of those components. Therefore, instead of overriding the color variable we provide adjustment control parameters for modifying these components. The artists came up with the following AFCs:

F.S. Color Adjust and Intensity Adjust for modifying the computed color and intensity values.

B.S. Color Adjust, Intensity Adjust, Longitudinal Shift Adjust, and Longitudinal Width Adjust.

Setting all these control parameters to their default values gives the original results of the dual scattering model.

4.6.2.3 Reproduction and Recombination

To reproduce the dual scattering results we use the original algorithm and replace the single scattering component f_s^{direct} with our pseudo scattering function f_s' and embed the defined artist controls into the f_{back}^{direct}, $f_{back}^{scatter}$ and $f_s^{scatter}$ components.

However, as mentioned in Section 4.4.3, replacing the physically based scattering function of f_s^{direct} with the non-physically based model f_s' can cause problems since f_s' is not necessarily energy conserving. For the purpose of computing the multiple scattering component we solve this problem by normalizing the pseudo scattering function f'_s to make it energy conserving :

$$f'^{norm}_s(\theta, \phi) = \frac{f'_s(\theta, \phi)}{\int_\Omega f'_s(\theta', \phi')\ d\theta'\ d\phi'} \tag{4.17}$$

Here Ω is the full sphere around the shading point. We use this normalized version of the single scattering component for all the computations in the dual scattering model. This will prevent the multiple scattering component from disappearing or blowing up (Figure 4.12). For single scattering we still use the un-normalized function f'_s which means that the final result of the hair shader is not guaranteed to be energy conserving. If this is a problem, another normalization step similar to the one described above can be applied to the final result of the shader.

Combining all of the above, we get the pseudo code shown below for reproducing the results of dual scattering with embedded artist controls. The modifications to the physically based version given in [ZYWK08] are highlighted in blue.

```
// Precompute $\bar{A}_b(\theta)$, $\bar{\Delta}_b(\theta)$, and $\bar{\sigma}^2_b(\theta)$ from ${f'}^{norm}_s$ for $0 < \theta < \pi$

$\mathbf{F}(T_f,\bar{\sigma}^2_f,$ directFraction$)$

    // Backscattering for direct and indirect lighting
    $f^{direct}_{back} \Leftarrow 2\bar{A}_b(\theta) g(\theta_h - \bar{\Delta}_b(\theta) + \alpha_{Back}, \bar{\sigma}^2_b(\theta) + \beta_{Back})$
            $/(\pi \cos^2\theta)$
    $f^{scatter}_{back} \Leftarrow 2\bar{A}_b(\theta) g(\theta_h - \bar{\Delta}_b(\theta) + \alpha_{Back}, \bar{\sigma}^2_b(\theta) + \bar{\sigma}^2_f(\theta) + \beta_{Back})$
            $/(\pi \cos^2\theta)$

    $f^{direct}_{back} \Leftarrow C_{Back} I_{Back} f^{direct}_{back}$
    $f^{scatter}_{back} \Leftarrow C_{Back} I_{Back} f^{scatter}_{back}$

    // Longitudinal functions for direct and indirect lighting
    $M'_X \Leftarrow g'(\theta_h - \alpha_X,\ \beta^2_X)$
    $M^G_X \Leftarrow g'(\theta_h - \alpha_X,\ \beta^2_X + \bar{\sigma}^2_f)$

    // Azimuthal functions for indirect lighting
    $N^G_X(\theta,\phi) \Leftarrow \frac{2}{\pi} \int_{\pi/2}^{\pi} N'_X(\theta,\phi')\ d\phi'$

    // Single scattering for direct and indirect lighting
    $f^{direct}_s \Leftarrow \sum M'_X N'_X(\theta,\phi)$
    $f^{scatter}_s \Leftarrow \sum M^G_X N^G_X(\theta,\phi)$

    $f^{scatter}_s \Leftarrow C_{Forward} I_{Forwrd} f^{scatter}_s$

    $F^{direct} \Leftarrow$ directFraction$(f^{direct}_s + d_b f^{direct}_{back})$
    $F^{scatter} \Leftarrow (T_f$ - directFraction$)\ d_f\ (f^{scatter}_s + \pi d_b f^{scatter}_{back})$

    // Combine the direct and indirect scattering components
    return $(F^{direct} + F^{scatter}) \cos\theta_i$
```

The symbols used here are as follows : $\bar{A}_b$ for the average backscattering attenuation, $\bar{\Delta}_b$ for the average longitudinal shift, $\bar{\sigma}_b^2$ for the average backscattering variance, T_f for the front scattering transmittance, and $\bar{\sigma}_f^2$ for the front scattering variance. M_X^G and N_X^G are the averaged forward scattered longitudinal and azimuthal scattering functions, respectively. All of these terms are taken directly from the Dual Scattering paper [ZYWK08].

The terms M'_X and N'_X are the pseudo scattering functions defined in Section 4.6.1.4. $I_{Forward}$ and I_{Back} are control parameters for adjusting the intensity values, while $C_{Forward}$ and C_{Back} are control parameters for adjusting the color values of the forward scattering and backscattering components respectively. Finally, α_{Back} and β_{Back} are control parameters for adjusting the longitudinal shift and the longitudinal width of the back scattering components.

4.6.3 Decoupling Single and Multiple Scattering

Multiple scattering computations are based on the single scattering functions, and there is an inherent relationship between these two components since multiple scattering is basically the effect of many single scattering events.

However, this relationship can be problematic for art direction. An art director might request a change of the appearance of the single scattering (e.g. color of the primary highlight) and yet want to keep everything else untouched. Unfortunately, if the artist changes the single scattering components, it will also affect the multiple scattering component.

To address this problem, we have provided the ability to break the link between single and multiple scattering by having two sets of parameters for the single scattering components. One of these sets will feed into the computations of multiple scattering and one will be used as the parameters of the single scattering. These two sets are linked together by default, but the artist has the ability to break this link at any point through the shader GUI.

4.7 Multiple Scattering Implimentation

Most of the existing models for simulating the exact behavior of multiple scattered light are computationally expensive and require additional data structures. The Dual Scattering model [ZYWK08] is a fast and relatively accurate model that approximates the behavior of multiple scattering component. In this section we provide the necessary details needed for implementing the Dual Scattering model efficiently without any ray tracing steps and any extra data structures. We have implemented this model in RenderMan and used it for the final results in this chapter.

It is critical to consider multiple scattering of light to ensure the correct perception of hair color, especially for light colored hair [MM06; ZW06]. To capture the exact behavior of the multiple scattered light one needs elaborate methods like brute force path tracing [ZW07], photon mapping [MM06; ZW06], or other grid based approaches like [MWM08]. Path tracing approaches are computationally expensive and their results converge very slowly. Photon mapping and grid based approaches are faster than path tracing methods but are still relatively expensive. Besides, the latter two are two pass methods which need ray tracing capabilities in their first pass and extra data structures to store extra information in the scene for the second pass (e.g. photons or spherical harmonics coefficients). Therefore, integrating these methods into RenderMan is more complicated than a typical surface shader.

Another class of methods try to approximate the multiple scattering component by considering the physical properties of human hair fibers. The most recent method in this category is the Dual Scattering method [ZYWK08]. It is a fast and good approximation that produces results very similar to path tracing. We decided to use this method as the basis of our multiple scattering component because it does not require extra data structures and with careful considerations one can implement this method without any ray tracing steps.

In Section 4.7.1 we explain the details of our implementation. We show how one can implement the global multiple scattering component without any ray tracing steps in Section 4.7.1.1. We also explain our implementation of the local multiple scattering component and provide extra clarifications and formula needed for implementation of this component in Section 4.7.1.2. In Section 4.7.2 we present the details of our implementation and highlight its differences with the original dual scattering method [ZYWK08]. We show some rendering results of our model and present a summarized version of our RenderMan Shading Language code in Appendix C.

4.7.1 Revised Dual Scattering Model

To implement the Dual Scattering method we follow the instructions of the original paper [ZYWK08]. However, there are some notes and clarifications which might be useful for anyone who wants to implement this method. In particular, in the original paper, the longitudinal inclination angles θ, θ_h, θ_d, θ_i and θ_o have been used interchangeably. We try to state the subscripts of all θ angles explicitly. Throughout this section we have highlighted all modifications and formula which are not present in the original paper in red. Table 4.1 summarizes all the terms and their corresponding descriptions that will be used in this report. In addition, we present necessary modifications in order to implement the dual scattering model without any raytracing steps which is very expensive in production environment.

The Dual Scattering method approximates the multiple scattering component Ψ as a combination of two components: global multiple scattering Ψ^G and local multiple scattering Ψ^L. The global multiple scattering component approximates the amount of light that reaches the neighborhood of the shading point after it travels through the hair volume. The local multiple scattering approximates the scattering events within the neighborhood of the shading point (See Figure 4.20).

The multiple scattering component is derived according to the following equation :

$$\Psi(x, \omega_d, \omega_i) = \Psi^G(x, \omega_d, \omega_i)(1 + \Psi^L(x, \omega_d, \omega_i)) \tag{4.18}$$

where ω_d is the direction of the incoming light entering the hair volume and ω_i is the direction of the multiple scattered light reaching the shading point x. This equation states that the multiple scattered light reaching the eye is either the result of global multiple scattering or its combination with the scattering events around the shading point (i.e. local multiple scattering).

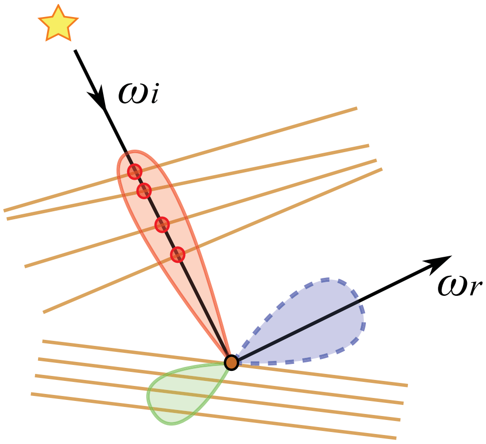


Figure 4.20: The dual scattering method separates the multiple scattering component (blue) into global multiple scattering (red) and local multiple scattering (green). The global multiple scattering depends on the orientation of all hair strands between the shading point and the light sources while the local multiple scattering depends only on the orientation of the hair strand at the shading point.

4.7.1.1 Global Multiple Scattering

Global multiple scattering Ψ^G is especially important for rendering light colored hair since it accounts for the light that penetrates the hair volume. Since the dominant subcomponent of the hair scattering function is the transmission component (TT), Ψ^G is approximated by only the front scattered light. In other words, Ψ^G is the contribution of one or more TT scattering events along the line connecting the shading point to the light source. Therefore, Ψ^G at any point is dependent

Table 4.1: Summary of all terms used in our implementation of the dual scattering model.

Symbol	**Description**
Ψ	Multiple scattering function
Ψ^G / Ψ^L	Global/local multiple scattering function
ω_i	Incoming light direction
ω_o	Outgoing view direction
ω_d	Direct lighting direction
θ_i / θ_o	Incoming/outgoing longitudinal angle
θ_h / θ_d	Half/difference angle between θ_i and θ_o
T_f	Forward scattering transmittance
S_f	Forward scattering spread
$\bar{S}_b$	Average backward scattering spread
$\bar{A}_b$	Average backscattering attenuation
$\bar{\Delta}_b$	Average backscattering longitudinal shift
d_f / d_b	Forward/backward scattering density factor
$\bar{a}_f$ / $\bar{a}_b$	Average forward/backward attenuation
$\bar{\alpha}_f$ / $\bar{\alpha}_b^2$	Average forward/backward scattering shift
$\bar{\beta}_f^2$ / $\bar{\beta}_b^2$	Average forward/backward scattering variance
$\bar{\sigma}_f^2$ / $\bar{\sigma}_b^2$	Total variance of forward/backward scattering
f_s^{direct} / $f_s^{scatter}$	Single scattering for direct/indirect lighting
f_{back}^{direct} / $f_{back}^{scatter}$	Backscattering for direct/indirect lighting
M_X	Longitudinal scattering function for component X
N_X	Azimuthal scattering function for component X
α_X	Longitudinal shift for component X
β_X	Longitudinal width for component X

on the orientations of all the hairs between the light source and that point. It is approximated by the multiplication of the forward scattering transmittance T_f and spread S_f of the light that reaches the shading point from all light sources.

$$\Psi^G(x, \omega_d, \omega_i) \approx T_f(x, \omega_d) S_f(x, \omega_d, \omega_i) \tag{4.19}$$

Global multiple scattering will be computed separately for different points. One of the easiest way of computing this component is by using a ray shooting method. In this method, for a given shading point, we shoot a ray toward the light and intersect it with any hair that occludes the light source. Then from the orientations of those intersecting hairs we compute the forward scattering transmittance according to :

$$T_f(x, \omega_d) = d_f \prod_{k=1}^{n} \bar{a}_f(\theta_d^k) \tag{4.20}$$

Here d_f is the forward scattering density factor which accounts for the hair density around the shading point. In the original paper, this density factor is set to 0.7. The term θ_d^k represents the longitudinal inclination at the k'th scattering event and $\bar{a}_f(\theta_d)$ is the average attenuation which is computed from the fiber scattering function f_s. The computation of the single scattering components is identical to [MJC⁺03] and is the product of longitudinal scattering functions M, which are modeled as Gaussian functions, with precomputed azimuthal scattering functions N.

We also compute the forward scattering spread as follows

$$S_f(x, \omega_d, \omega_i) = g(\theta_h, \bar{\sigma}_f^2(x, \omega_d))/(\pi \cos \theta_d) \tag{4.21}$$

Here $\bar{\sigma}_f^2$ is the total variance of forward scattering in the longitudinal directions and is the sum of variances of all scattering events along the shadow path:

$$\bar{\sigma}_f^2(x, \omega_d) = \sum_{k=1}^{n} \bar{\beta}_f^2(\theta_d^k) \tag{4.22}$$

Ray shooting is very straightforward in the ray tracing context but it is very expensive when using a REYES based renderer like RenderMan. We therefore approximate the global multiple scattering component by assuming that the hairs in front of any point have the same orientation as the shading point. This is a good approximation for long flat hair style which was the main focus for our production. Moreover, in a production environment, the inaccuracies in this approximation can be compensated by the extra artistic controls. At each shading point we estimate the number of hair strands n in front of a shading point by looking up the shadow opacity value stored in the deep shadow maps. We then approximate T_f and $\bar{\sigma}_f^2$ according to the following formula:

$$T_f(x, \omega_d) \approx d_f \ \bar{a}_f(\theta_d)^n \tag{4.23}$$

$$\bar{\sigma}_f^2(x, \omega_d) \approx \bar{\beta}_f^2(\theta_d) \times n \tag{4.24}$$

Here θ_d is the longitudinal inclination of the hair strands at the shading point.

These approximations have been acceptable based on our production needs. However, an exact implementation is possible if it is required. The goal is to find $\prod_{k=1}^{n} \bar{a}_f(\theta_d^k)$ and $\sum_{k=1}^{n} \bar{\beta}_f^2(\theta_d^k)$ where n is the number of hairs in front of the shading point and θ_d^k is the longitudinal angle of the hair strand with respect to the light source. Here, $\bar{a}_f$ and $\bar{\beta}_f^2$ are precomputed functions.

The main idea is to use deep shadow maps to calculate the list of all θ_d^k angles. Then we can use these values to compute all the values of $\bar{a}_f(\theta_d^k)$ and $\bar{\beta}_f^2(\theta_d^k)$. For the exact solution, we introduce a virtual opacity function inside the hair surface shader and let it return opacity values κ as a function of its longitudinal orientation (Any one-to-one mapping form $-\pi/2 \leq \theta_d \leq \pi/2$ to $0 < \kappa < 1$ works). In the first step we compute the deep shadow map of the hair volume using these opacity functions. In the second step we do a post process on the deep texture file generated for the deep shadow maps. We can retrieve the actual opacity values from the stored shadow function inside the deep texture. Also, from those opacity values we can compute the actual orientations of the hair fibers. This will provide us with list of θ_d^k values.

4.7.1.2 Local Multiple Scattering

The local multiple scattering function Ψ^L approximates the light scattering events within the neighborhood of the shading point. Light paths contributing to Ψ^L must include at least one backward scattering event since other light paths are already

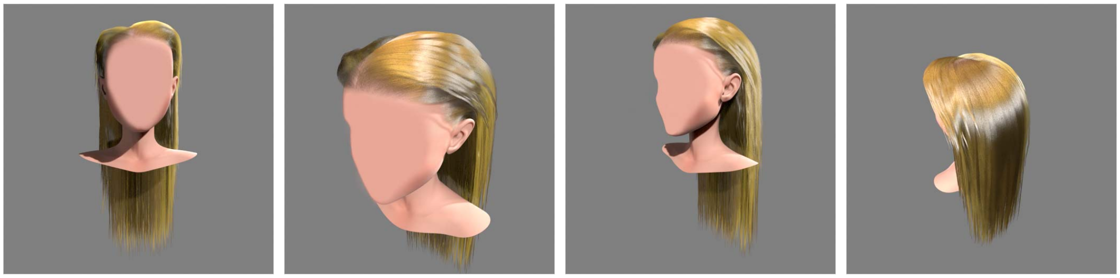


Figure 4.21: Some rendering results based on our implementation of the dual scattering model.

accounted for in Ψ^G. For simplification, we assume that all the hairs surrounding the shading region have the same orientation and that there is an infinite number of them. Therefore Ψ^L is only dependent on the longitudinal inclination of the hair strand at the shading point. This enables us to pre-compute Ψ^L for all the longitudinal inclination angles :

$$\Psi^L(x, \omega_d, \omega_i) f_s(\omega_i, \omega_o) \approx d_b f_{back}(\omega_i, \omega_o) \tag{4.25}$$

Here d_b is the backward scattering factor and similar to d_f it is set to 0.7. f_{back} is the backscattered light which is the product of an average backscattering attenuation function $\bar{A}_b$ and an average backscattering spread function $\bar{S}_b$:

$$f_{back}(\omega_i, \omega_o) = 2\bar{A}_b(\theta_d)\bar{S}_b(\omega_i, \omega_o)/\cos\theta_d \tag{4.26}$$

The average backscattering attenuation function $\bar{A}_b$ is given by

$$\bar{A}_b(\theta_d) = \frac{\bar{a}_b \bar{a}_f^2}{1 - \bar{a}_f^2} + \frac{\bar{a}_b^3 \bar{a}_f^2}{(1 - \bar{a}_f^2)^2} \tag{4.27}$$

where $\bar{a}_f$ and $\bar{a}_b$ are the average forward and backward scattering attenuations. Similar to S_f (Equation 4.21), $\bar{S}_b$ is computed as

$$\bar{S}_b(x, \omega_d, \omega_i) = g(\theta_h - \bar{\Delta}_b(\theta_d), \bar{\sigma}_b^2(\theta_d))/(\pi \cos\theta_d)$$

where $\bar{\Delta}_b$ is the average longitudinal shift given by:

$$\bar{\Delta}_b \approx \bar{\alpha}_b\left(1 - \frac{2\bar{a}_b^2}{(1 - \bar{a}_f^2)^2}\right) + \bar{\alpha}_f\left(\frac{2(1 - \bar{a}_f^2)^2 + 4\bar{a}_f^2\bar{a}_b^2}{(1 - \bar{a}_f^2)^3}\right) \tag{4.28}$$

and $\bar{\sigma}_b$ is the average backscattering standard deviation given by:

$$\bar{\sigma}_b \approx (1 \ + \ d_b \ \bar{a}_f^2) \frac{\bar{a}_b \sqrt{2\bar{\beta}_f^2 \ + \ \bar{\beta}_b^2} \ + \ \bar{a}_b^3 \sqrt{2\bar{\beta}_f^2 \ + \ \bar{\beta}_b^2}}{\bar{a}_b \ + \ \bar{a}_b^3 (2\bar{\beta}_f \ + \ 3\bar{\beta}_b)} \tag{4.29}$$

Here $\bar{\alpha}_f(\theta_d)$ and $\bar{\alpha}_b(\theta_d)$ are the average forward and backward scattering shifts respectively. They are basically weighted averages of longitudinal shifts, α_R, α_{TT}, and α_{TRT} for the front and back scattering hemispheres. They can be computed from the following equations:

$$\bar{\alpha}_f(\theta_d) = \frac{\int_{\Omega_f} f_R \alpha_R + f_{TT} \alpha_{TT} + f_{TRT} \alpha_{TRT}(\theta_d)}{\int_{\Omega_f} f_R + f_{TT} + f_{TRT}(\theta_d)} \tag{4.30}$$

$$\bar{\alpha}_b(\theta_d) = \frac{\int_{\Omega_b} f_R \alpha_R + f_{TT} \alpha_{TT} + f_{TRT} \alpha_{TRT}(\theta_d)}{\int_{\Omega_b} f_R + f_{TT} + f_{TRT}(\theta_d)} \tag{4.31}$$

where Ω_f and Ω_b represent the forward and backward scattering hemispheres. f_X represents the X portion of the single scattering component f_s for $X \in \{R, TT, TRT\}$.

Similarly, for the average forward and backward scattering variances $\bar{\beta}_f^2(\theta)$ and $\bar{\beta}_b^2(\theta)$ we have:

$$\bar{\beta}_f(\theta_d) = \frac{\int_{\Omega_f} f_R \beta_R + f_{TT} \beta_{TT} + f_{TRT} \beta_{TRT}(\theta_d)}{\int_{\Omega_f} f_R + f_{TT} + f_{TRT}(\theta_d)} \tag{4.32}$$

$$\bar{\beta}_b(\theta_d) = \frac{\int_{\Omega_b} f_R \beta_R + f_{TT} \beta_{TT} + f_{TRT} \beta_{TRT}(\theta_d)}{\int_{\Omega_b} f_R + f_{TT} + f_{TRT}(\theta_d)} \tag{4.33}$$

Please note that $\bar{\alpha}_{f/b}$, $\bar{\beta}_{f/b}$, $\bar{a}_{f/b}$, $\bar{\Delta}_b$ and $\bar{\sigma}_b^2$ are color variables and have to be computed for different color channels (in general for different wavelengths). This means that different wavelengths will be shifted and spread differently.

For efficient computation, Zinke et al. computed the shading differently depending on whether or not the shading point receives illumination directly F^{direct} or through other hairs (points in shadow) $F^{scatter}$. For indirect versions they introduced $f_s^{scatter}$ which is a modified version of single scattering component f_s^{direct}.

Similar to the single scattering BRDF, $f_s^{scatter}$ has three longitudinal function called M_X^G and three azimuthal functions N_X^G for $X \in \{R, TT, TRT\}$. These functions are computed directly from the longitudinal and azimuthal functions of the hair BRDF as follows:

$$M_X^G(\theta_h) = M_X(\theta_h - \alpha_X, \bar{\beta}_X^2 + \bar{\sigma}_f^2) \tag{4.34}$$

$$N_X^G(\theta_d, \phi) = \frac{2}{\pi}\int_{\pi/2}^{\pi} N_X(\theta_d, \phi')d\phi' \tag{4.35}$$

Note that N_X^G should be the averaged version of N_X over the front scattering hemisphere but the authors of the paper considered a simpler approximation to be the average over the front semi-circle.

The front hemisphere is dominated by the TT component. There are some contributions from the R component as well, but the contributions from the TRT component in the front scattering hemisphere are very small. Therefore, N_{TT}^G is larger than N_R^G and both of them are larger than N_{TRT} which means that the color of the $f_s^{scatter}$ component is being dominated by the color of the TT component.

Similarly, f_{back} should be computed separately for direct and indirect lighting. We have decomposed the f_{back} term form the original paper into two terms f_{back}^{direct} and $f_{back}^{scatter}$. The latter term accounts for the change in variance of the forward scattered light in the longitudinal direction and will be used in the calculation of $F^{scatter}$. See Figure 4.22 for a visualization of different terms in the dual scattering model.

$$f_{back}^{direct}(\omega_i, \omega_o) = \frac{2\bar{A}_b(\theta)g(\theta_h - \bar{\Delta}_b(\theta), \bar{\sigma}_b^2(\theta) + \bar{\sigma}_f^2(x, \omega_d))}{\pi \cos^2\theta_d} \tag{4.36}$$

$$f_{back}^{scatter}(\omega_i, \omega_o) = \frac{2\bar{A}_b(\theta)g(\theta_h - \bar{\Delta}_b(\theta), \bar{\sigma}_b^2(\theta))}{\pi \cos^2\theta_d} \tag{4.37}$$

4.7.2 Implementation

The following pseudo code is our revised version of the pseudo code presented in Figure 5 of the original paper [ZYWK08]. The corresponding RenderMan pseudo code is shown in Appendix C.

```
// Pre-compute $\bar{A}_b(\theta_d)$, $\bar{\Delta}_b(\theta_d)$ and $\bar{\sigma}_b^2(\theta_d)$ from $f_s$ for $0 < \theta_d < \pi$
```

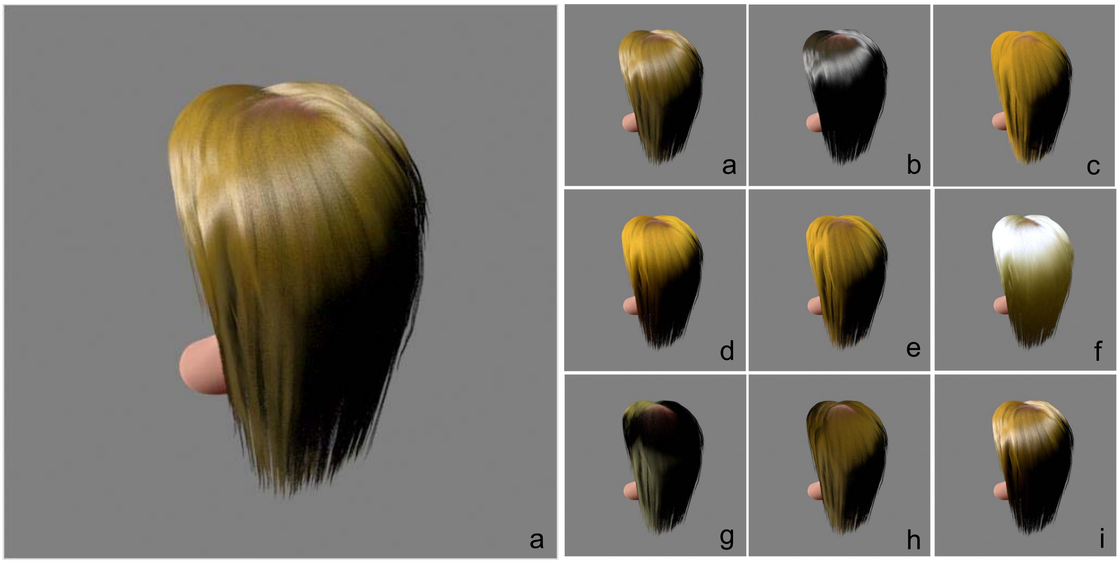


Figure 4.22: Sub-components of the dual scattering method. (a) Dual scattering model. (b) Single scattering component (f_s). (c) Average backscattering attenuation ($\bar{A}_b$). (d) Local multiple scattering for direct light lighting (f_{back}^{direct}). (e) Local multiple scattering for indirected lighting ($f_{back}^{scatter}$). (f) Average backscattering spread ($\bar{S}_b$). (g) Single scattering for indirect lighting ($f_s^{scatter}$) (h) $F^{scatter}$ (i) F^{direct}

```
// Compute the amount of direct lighting $directFraction$
// Compute the number of hairs in front of the shading point $n$

$T_f(x, \omega_d) = d_f \ \bar{a}_f(\theta_d)^n$
$\bar{\sigma}_f^2(x, \omega_d) = \bar{\beta}_f^2(\theta_d) \times n$

$\mathbf{F(}\ T_f, \bar{\sigma}_f^2, directFraction)$

    // Backscattering for direct/indirect lighting
    $f_{back}^{direct} \Leftarrow 2\bar{A}_b(\theta_d)g(\theta_h - \bar{\Delta}_b(\theta_d), \bar{\sigma}_b^2(\theta_d))/(\pi \cos^2\theta_d)$
    $f_{back}^{scatter} \Leftarrow 2\bar{A}_b(\theta_d)g(\theta_h - \bar{\Delta}_b(\theta_d), \bar{\sigma}_b^2(\theta_d) + \bar{\sigma}_f^2(\theta_d))/(\pi \cos^2\theta_d)$

    // Longitudinal functions for direct/indirect lighting
    $M_X \Leftarrow g(\theta_h - \alpha_X, \bar{\beta}_X^2)$
    $M_X^G \Leftarrow g(\theta_h - \alpha_X, \bar{\beta}_X^2 + \bar{\sigma}_f^2)$

    // Azimuthal functions for indirect lighting
    $N_X^G(\theta_d, \phi) \Leftarrow \frac{2}{\pi}\int_{\pi/2}^{\pi} N_X(\theta_d, \phi')d\phi'$
```

```
// Single scattering for direct/indirect lighting
$f_s^{direct} \Leftarrow \sum M_X(\theta_h) N_X(\theta_d, \phi)$
$f_s^{scatter} \Leftarrow \sum M_X^G(\theta_h) N_X^G(\theta_d, \phi)$

$F^{direct} \Leftarrow directFraction(f_s^{direct} + d_b f_{back}^{direct})$
$F^{scatter} \Leftarrow (T_f - directFraction) d_f (f_s^{scatter} + \pi d_b f_{back}^{scatter})$

// Combine the direct and indirect scattering components
**return** $(F^{direct} + F^{scatter}) \cos \theta_i$
```

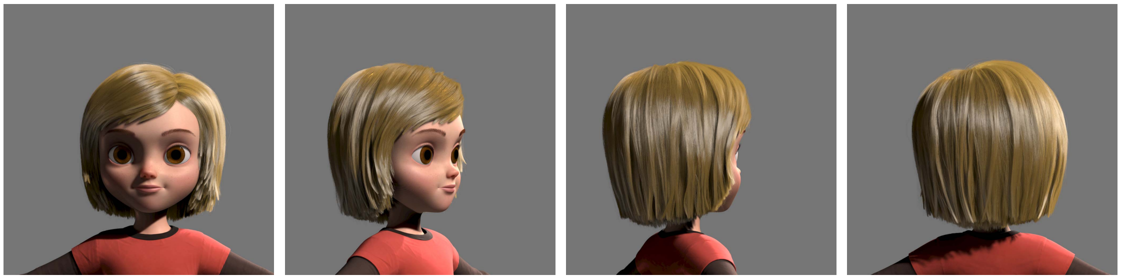


Figure 4.23: Applying blond hair to one of our existing characters.

Figure 4.21 shows some rendering results from different viewing angles and lighting directions. Figure 4.22 illustrates the contribution of different terms involved in the computations of the dual scattering model. Finally, Figure 4.23 shows some rendered frames of an animation using our dual scattering implementation.

4.8 Results

We have implemented our hair shading system in RenderMan and it has been fully integrated into the production pipeline at the Walt Disney Animation Studios. It has been used in the production of the animated feature film Tangled and has proven to be very versatile in handling different types of hair and fur successfully. Figure 4.24 shows some frames from the film.

Figure 4.1 shows frames from a short animation which is also using our hair shader, and which is included in the accompanying video. This shows that the overall appearance holds up nicely when animated. This is important in film production since many shading models can produce convincing still images but perform poorly with animation.

A common concern when introducing a physically inspired shading system in a production environment is that it might not have the same amount of flexibility as an existing ad hoc system. To address this issue, we have recreated the look for a character which was originally created using an ad hoc shader. Figure 4.25 shows the look of this character using both the original shader and our new shader. Using our new shader, we were able to replicate the art direction and the overall appearance of the character. Note that the groom for this character was created specifically for the ad hoc shader. Since the new shader tends to reveal many more details in the groom than the ad hoc shader, a more exact match between the two results would require changes to the underlying groom. We did not do that for this test as we wanted to focus on the effect of the shader itself. However, the underlying grooms *were* adapted to the new shader in order to produce the results in Figure 4.1.

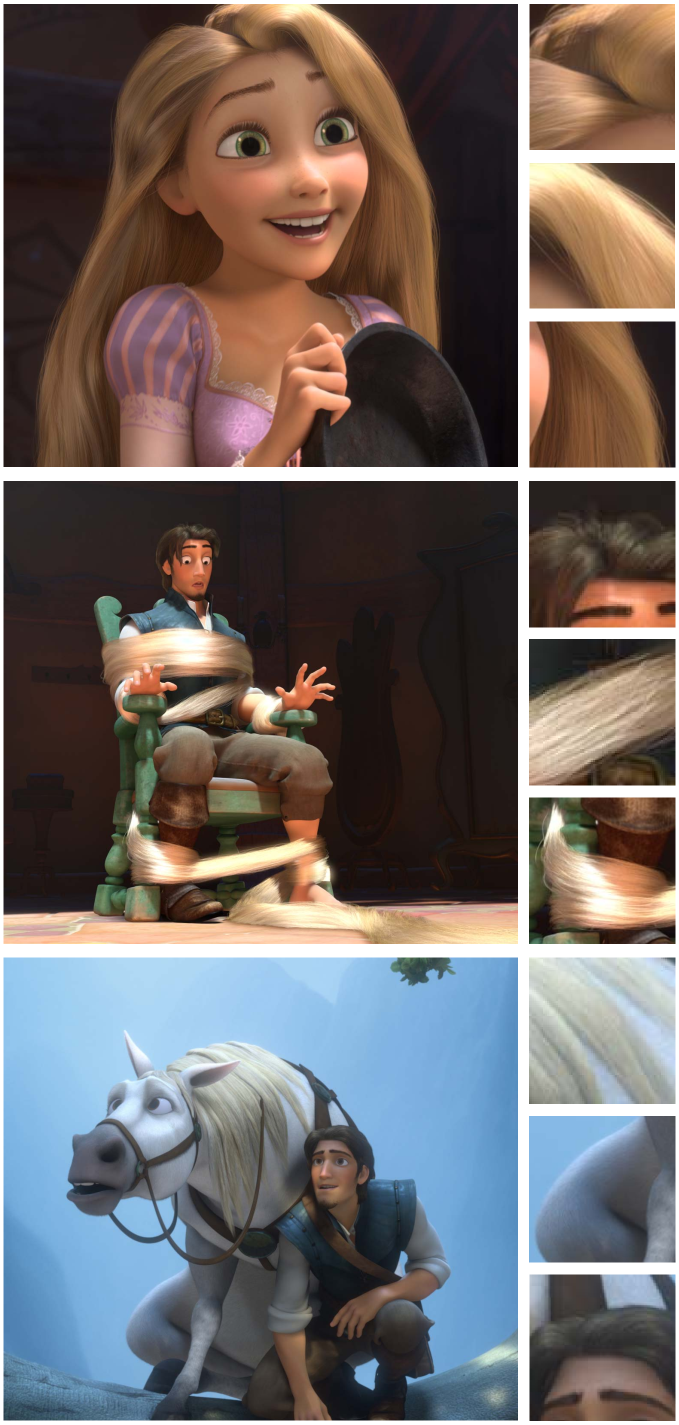


Figure 4.24: The hair shading system has been used successfully for rendering the long blond hair of the character Rapunzel, the short brown hair of the character Flynn, and the white hair/fur on the horse. The three frames are from the animated feature film Tangled.

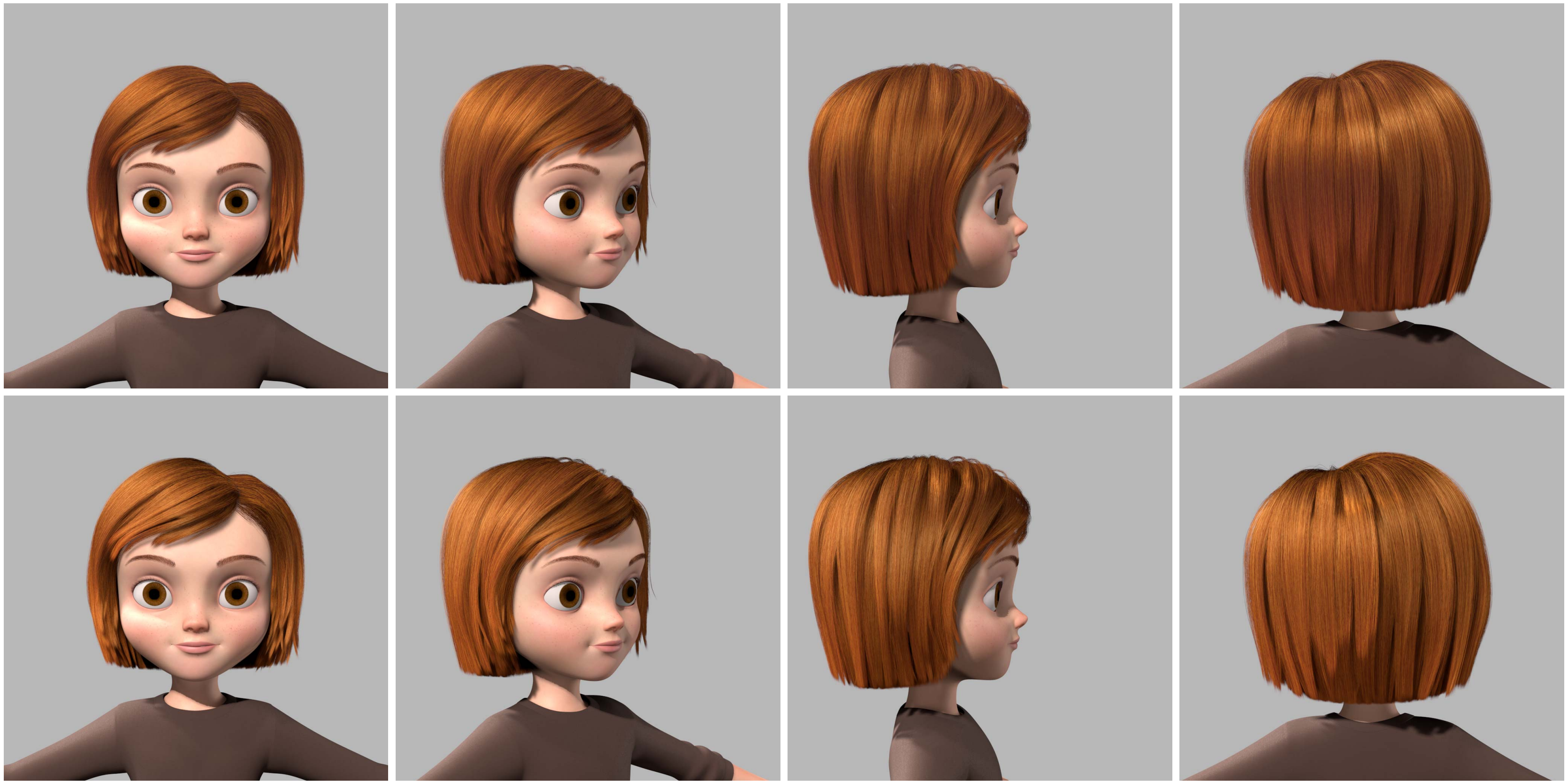


Figure 4.25: Reproducing the look for the character Penny using the original hair shader (top row), and our new hair shader (bottom row).

Figure 4.26 shows an art reference and a matching rendering using our new shader. It is important to note that in this case, the art directors were not looking for an exact match between the rendered result and the painted reference image. The painted reference only served as the initial guide for the overall look and feel of the rendered image.

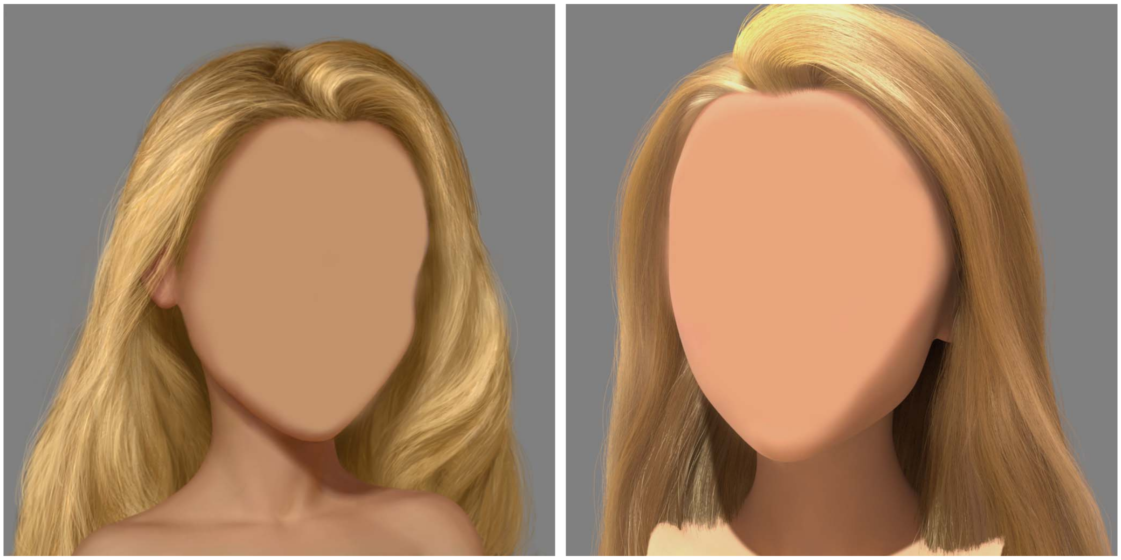


Figure 4.26: Painted art reference (left) and a corresponding rendering (right) using our new hair shader. Faces have been been taken out intentionally.

4.9 Comparisons

To evaluate the usability of our shader we organized auser study at our studio. Our goal was to compare our shader to the current state of the art, by comparing to the existing ad hoc production shader used in the feature film BOLT and a physically based shader that has been used in production. The former is based on Kajiya-Kay's model, but has many layers of tweaks and controls built on top of it, and is based on years of accumulated experience from numerous feature films. The latter is an implementation of Marschner's model and Zinke's BCSDF model for single scattering. It uses a version of what later became the dual-scattering model for the multiple scattering. We will refer to the three shaders as the New shader, the Production shader and the Research shader.

4.9.1 Setup

We gathered three groups of artists from our look development and lighting departments with varying levels of experience. We assigned one of the shaders to each group of artists and trained each group on using their assigned shader. Among those who finished the user study, 6 artists were assigned to the New shader, 4 artists were assigned to the Research shader, and 3 artists were assigned to the Production shader.

As reference material for the test we captured photos of a natural blond hair wig illuminated by a single directional light. Blond hair provides the most challenging test for most hair shaders, since its appearance depends on both single and multiple scattering. We provided 4 photos to the artists corresponding to different lighting directions, but captured 8 photos to be used for the evaluations. The goal was to evaluate the behavior of the shaders under both known and unknown lighting directions. In particular we wanted to be able to identify situations in which a shader could be fine tuned to match any given reference, but would require additional tweaks for every new lighting situation. See Figure 4.27. The reason we chose photographic reference was to ensure a fair comparison for the physically based shader and to give us an unbiased ground truth.

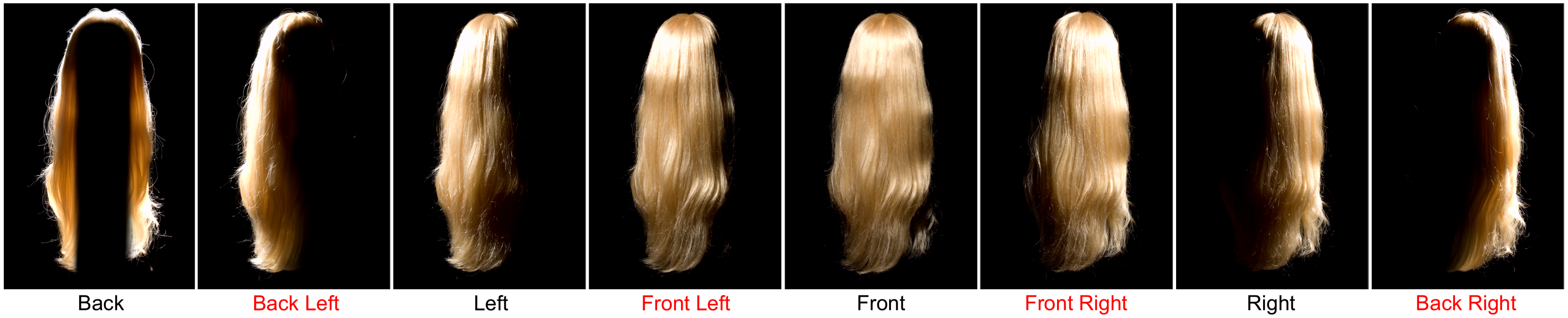


Figure 4.27: Reference photographs used for the user study. The labels indicate the location of the light. Only images with black labels were given to the artists at the time of the user study. All images, however, were used to evaluate the results.

We groomed a hair model similar to the wig in our photo shoot setup, and placed it in a scene which contained 8 directional lights with the same position, orientation and intensity as the ones used in our photo shoot. Given this we asked the artists to tweak the parameters of their assigned shader in order to come up with a fixed set of parameter values to match the appearance of all 4 given photo references. We limited the amount of time that each artist could spend on this task to 4 hours which is comparable or a little less than what they would spend on a similar task in production.

4.9.2 Ranking

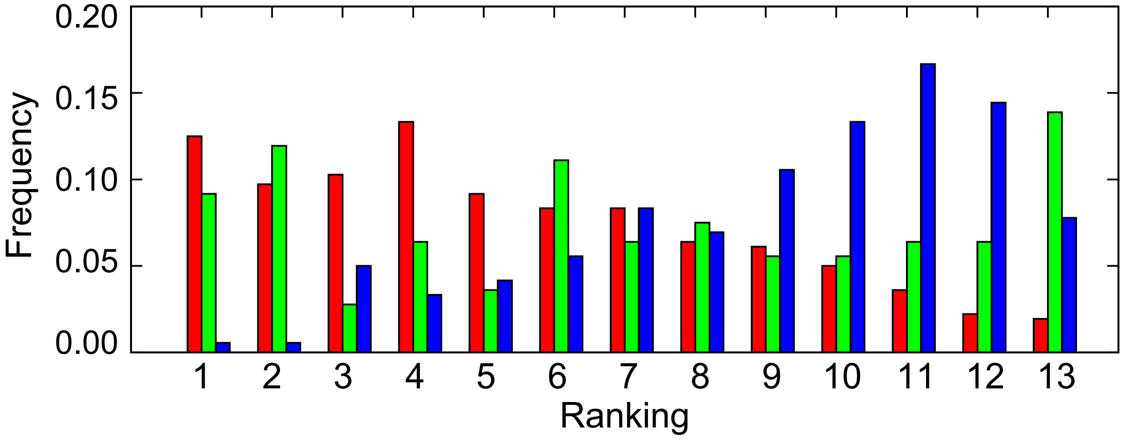


Figure 4.28: Normalized distribution of the rankings for all the shading models (1 is best, 13 is worst). The New shader (red) is generally ranked better than the Research shader (blue) while the Production shader (green) is somewhat inconsistent with really good results and some not so good. The numbers have been normalized based on the number of samples in each group.

Upon completion of the assignment we used the submitted shader parameters of each artist to render all 8 lighting directions. We then anonymized and randomized the order of the rendering results and encouraged everyone at our studio to rank the results from best to worst based on the photographic references. We got 35 responses from a mix of expert and non-expert volunteers. Figure 4.28 visualizes the distribution of rankings for each shader. All rendering results and the details of their evaluation ranks are available in Appendix D.

4.9.3 Discussion

There are many confounding factors in the evaluation described above which prevent us from drawing any statistically significant conclusions. One is the small sample size, which is difficult to address since the population of people with the right skillset is fairly small. Another is the fact that every artist did not repeat the experiment with all three shaders. This means that an exceptionally good artist can skew the results toward one shader.

With these caveats in mind, there are still a number of trends apparent in the results which are consistent with the experiences we have had with the use of these shaders in production. The Research shader generally performs poorly. While the artists are able to get the desired appearance for some lighting directions, they fail to get a consistent appearance under all lighting directions. See Figure 4.29. In practice they were also a lot more frustrated with the overall experience using this shader.

The Production shader can do surprisingly well with an experienced user, and the best results are clearly better than the best result produced by the Research shader. However, it often fails to recreate the appearance of real hair. In particular, it misses the secondary highlight, and has problems with the bright transmission component. It also tends to miss the variation in lighting inside the hair volume and it often gives a flat appearance to the hair. This is shown in Figure 4.30, but has also been observed during production use.

In contrast to the above, the new shader generally performs very well and is able to produce physically plausible results for all the lighting directions. While it is not flawless, its intuitive controls let most artists produce good results with a minimal amount of training. In our video we show that this is also true for different

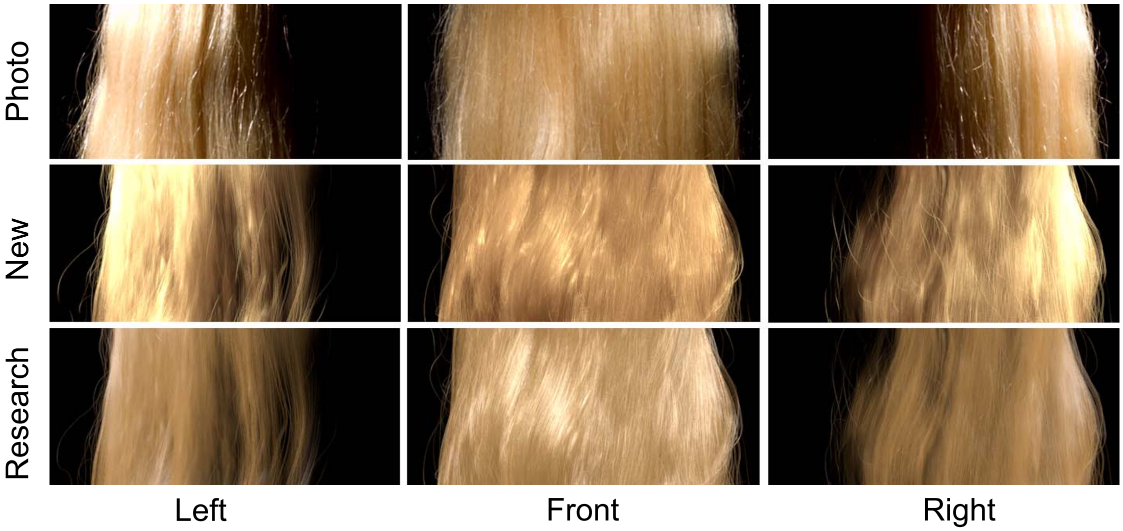


Figure 4.29: Comparing one of the results of the New shader (middle row) with one of the results of the Research shader (bottom row). The artists have successfully matched the frontlit image (middle column) with the photo reference. However, due to lack of control when using the Research shader, the artist was not able to match the hair appearance in the other two lighting setups (left and right columns).

lighting conditions (indoor / outdoor, direct / indirect, etc.), and in practice we have received almost exclusively positive feedback from its use in production.

4.9.4 Performance

Performance was not the main focus of of our work but our shading model has turned out to be more efficient than the other two shaders. We compared the performance of the different shaders by rendering the frontlit image of the user study in 1024×1024 resolution on a 2.66 GHz Intel Xeon 5150 machine. The hair model consisted of 140 guide hairs which were used to generate more than 100000 hairs procedurally at render time. We used the instances of each shader with the highest rankings for these measurements. The results indicate that our new shader is around 3.3 times faster than the Production shader and 1.5 times faster

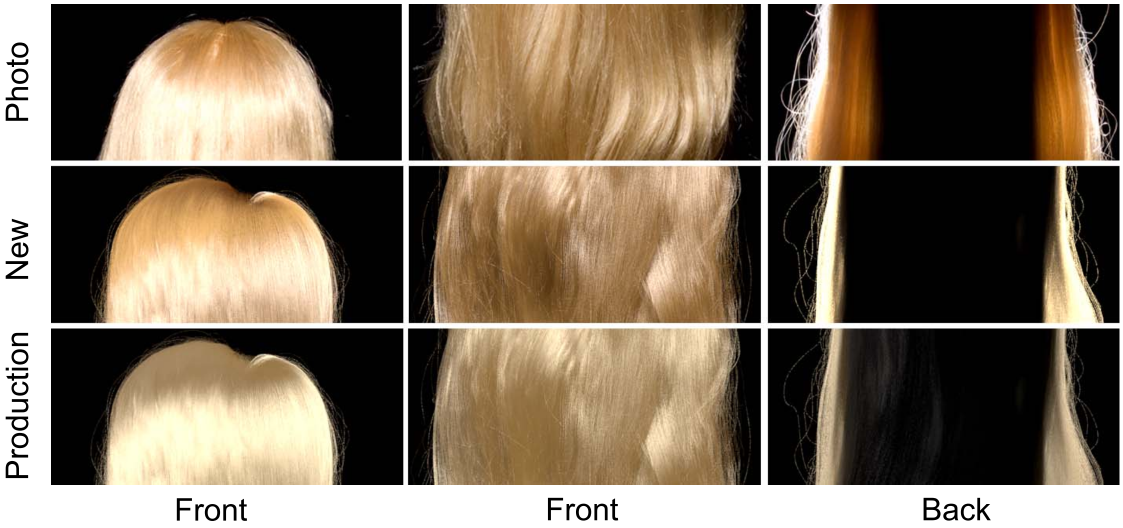


Figure 4.30: Comparing one of the results of the New shader (middle row) with one of the results of the Production shader (bottom row). Unlike the New shader, the Production shader fails to correctly produce the secondary highlight (left column), to capture the details of the light scattering inside the hair volume (middle column), and to produce the bright halo around the hair in the backlit situation (right column).

than the Research shader. This is a significant improvement, especially in movie production where even small improvements add up quickly. The Research shader consumes 1.6 times more memory than our shader and the Production shader uses 1.3 times more memory.

4.10 Conclusion and Future Work

We have addressed the problem of art-directability of physically-based hair shading models. In particular, we have defined the basic requirements for artist friendly systems and have shown that physically based models fail to satisfy these requirements. Accordingly, we have introduced a new approach for creating a physically inspired but art-directable hair shading model. In our experience the new shading model is more intuitive to use and produces better results than previous shading models used in production.

While these results are encouraging, it should, however, be noted that none of the shaders obtained perfect scores. Part of the reason for this, most likely, has to do with the underlying groom and the interplay between shader and groom, which we hope to address in future work.

Another avenue for future work is to investigate the applicability of our approach to materials other than hair. We speculate that our approach is applicable to a much broader range of materials in appearance modeling.

4.11 Acknowledgments

We would like to thank everyone at the Walt Disney Animation Studios who helped us with this project as well as Maryann Simmons, Will Chang and Krystle de Mesa for proof reading a draft of this paper. We would also like to thank the anonymous reviewers for their helpful comments. All images © Disney Enterprises, Inc. unless otherwise noted.

The material in this chapter is, in part, a reproduction of the material published in Iman Sadeghi, Heather Pritchett, Henrik Wann Jensen, and Rasmus Tamstorf. "An Artist Friendly Hair Shading System." In *ACM Transactions on Graphics , (Proceedings of SIGGRAPH 2010)*, 29(4), 2010; as well as the extended technical report Iman Sadeghi, and Rasmus Tamstorf. "Efficient Implementation of the Dual Scattering Model in RenderMan." Technical Report. *Walt Disney Animation Studios*, 2010. The dissertation author was the primary investigator and author of both papers.

5

Light Interactions with Interwoven Microcylinders

CLOTH fabrics present one of the most sophisticated appearances in our everyday life. Their appearance is dependent on the type of fibers and the structure of the weaving patterns. In this chapter we present a practical appearance model for rendering cloth fabrics.

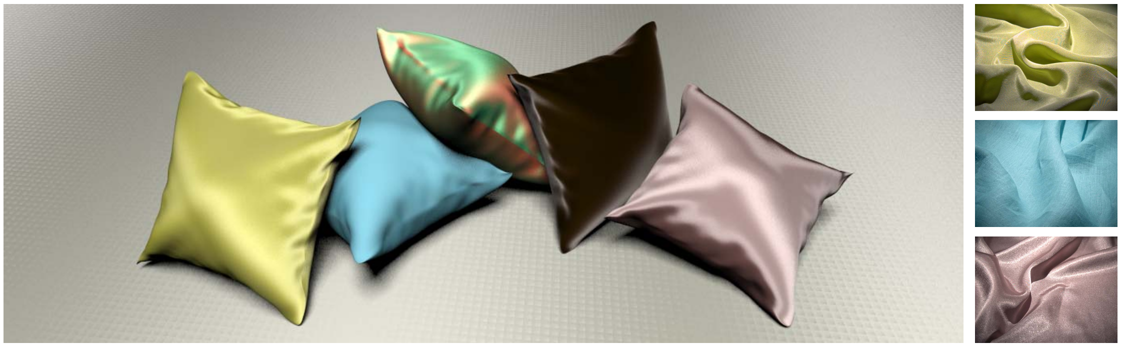


Figure 5.1: Image rendered using our microcylinder shading model with intuitive parameters for different fabric types. Left to right the fabrics are: Silk Crepe de Chine, Linen Plain, Silk Shot Fabric, Velvet, and Polyester Satin Charmeuse. (right) Reference photos for three of the fabrics rendered.

The model can simulate both the anisotropic highlights on cloth as well as the complex color shifts seen in cloth made of different colored threads. Our model is based on extensive Bidirectional Reflectance Distribution Function (BRDF) measurements of several cloth samples. We have also measured the scattering profile of several different individual cloth threads. Based on these measurements we have derived a simple model capable of predicting the light scattered by different threads. We model cloth as a collection of threads in a weave pattern which provides information about the coverage of the different thread types as well as their tangent directions. Our model also accounts for shadowing and masking by the threads. We validate our model by comparing predicted and measured values and we show how we can use the model to recover parameters for different cloth samples including silk, velvet, linen, and polyester with varying weaving patterns. We can also, model the appearance of novel physically plausible cloth fabrics. Finally, we demonstrate that our model can run in real-time on a GPU.

5.1 Introduction

Cloth is a complex material made of interwoven threads of different types. Cloth appearance can vary from matte diffuse to highly specular and anisotropic. Existing models for simulating cloth in graphics are either too simplistic to capture this appearance or too complex for practical use.

In this paper we present a practical appearance model for cloth. Our model is based on extensive measurements of the scattered light by cloth samples as well as individual threads. Based on these measurements, we have developed a simple model for light scattering by threads as well as cloth made of a given weave pattern. Our appearance model simulates cloth at a distance and ignores the appearance of individual threads unlike recent work on cloth [Ira08]. This makes the model significantly simpler and more robust in terms of matching measured data. In addition, our model takes into account the shadowing and masking by the individual threads in the cloth. Our model is easy to control and it can reproduce a wide range of cloth types including linen, silk, polyester, and velvet with varying weaving patterns. We can also, model the appearance of novel physically plausible cloth fabrics. We include measured parameters for these cloth types including the weave patterns for our cloth samples.

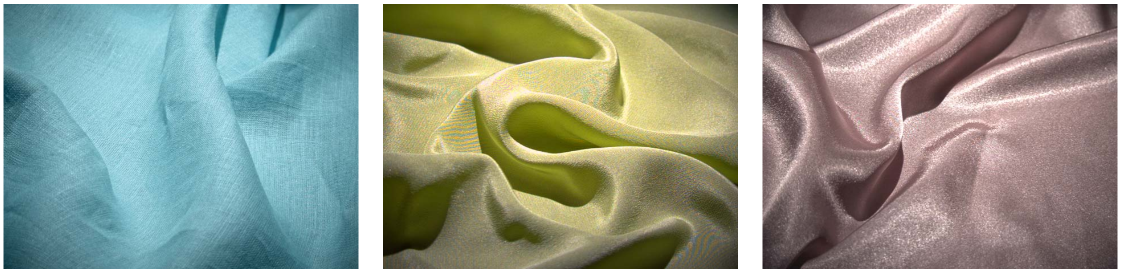


Figure 5.2: Cloth appearance for three different fabrics which are the main focus of our study in this paper: (left) Linen Plain, (middle) Silk Crepe de Chine, and (right) Polyester Satin Charmeuse.

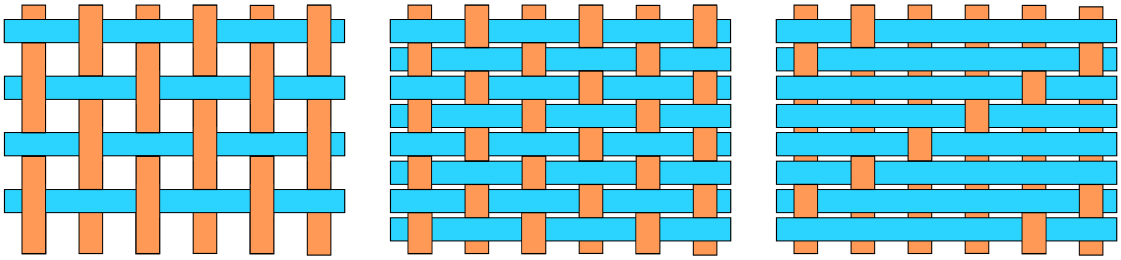


Figure 5.3: Different weaving patterns of the fabrics shown in Figure 5.2: (left) Plain, (middle) Crepe de Chine, and (right) Satin Charmeuse.

5.2 Background Theory

5.2.1 Notations of Cloth and Thread Geometry

The notations used in this chapter are similar to the ones introduced in Chapter 4 with some adjustments. The view direction ω_r and light direction ω_i point away from the point on the surface of cloth. We compute azimuthal directions ϕ_r and ϕ_r with respect to the surface normal of the cloth n. The longitudinal directions

θ_r and θ_i are being computed with respect to the normal plane of the thread at the shading point as shown in Figure 5.4. Directions with zero θ lie on the normal plane of the thread while θ value of $\pi/2$ is the tangent direction of the tread t.

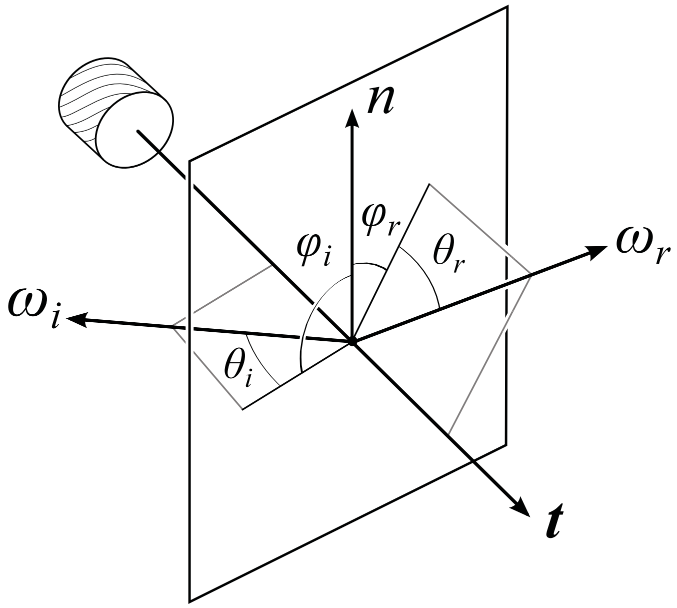


Figure 5.4: Notations and geometry of light reflection from a cylindrical fiber. Longitudinal angles θ are computed with respect to the normal plane and the azimuthal angles ϕ are computed based on the local surface normal direction n. When the thread is not part of a fabric n can be any arbitrary direction within the normal plane.

We define to new longitudinal angle ψ_i (and ψ_i) as the angle between local surface normal n and the projection of ω_i (ω_r) on to the plane that contains t and n (see Figurer 5.5).

We define the difference angles according to the following formula:

$$\phi_d = \phi_i - \phi_r \tag{5.1}$$

$$\theta_d = \theta_i - \theta_r \tag{5.2}$$

$$\psi_d = \psi_i - \psi_r \tag{5.3}$$

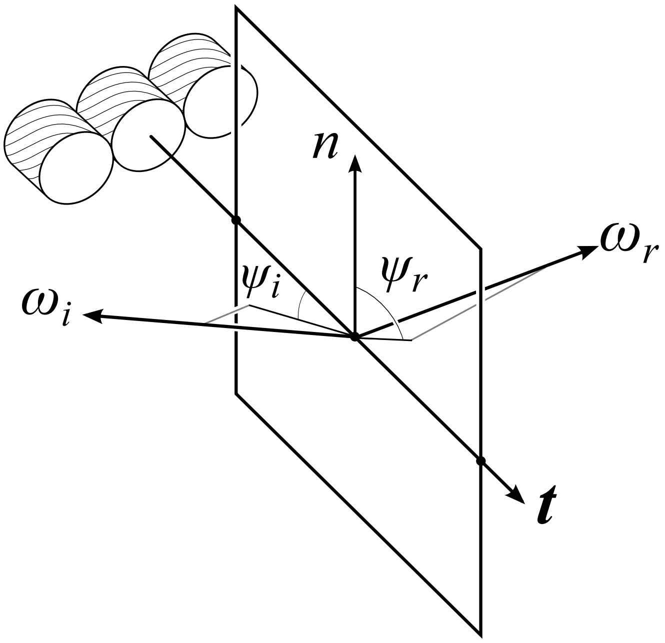


Figure 5.5: Longitudinal angles ψ_i and ψ_r are the angles between local surface normal n and the projection of ω_i and ω_r on to the plane spanned by t and n vectors.

Similarly, we define the half angles as:

$$\phi_h = (\phi_i + \phi_r)2 \tag{5.4}$$

$$\theta_h = (\theta_i + \theta_r)/2 \tag{5.5}$$

5.2.2 Cloth Fibers

Cloth fibers are the building blocks of cloth threads and play an important role in the final appearance of cloth. There is a wide range of variations in the type and structure of cloth fibers. Some fibers are made out of plants. For example cotton fibers grow around the seeds of the cotton plant [SJK06] while Flax fibers are extracted from the skin of the stem of the flax plant [MW03]. Threads made from flax fibers are called linen. Many types of fibers have animal sources. Silk fibers are made out of the cocoon of the silk worm [Par92]. Wool fibers are obtained from sheep and some other mammals [(Ed04]. In addition to natural fibers, synthetic fibers are widely used in fabrication of fibers. Examples of synthetic fibers include Polyester, Nylon, and Rayon. Figure 5.6 shows micrograph images of different cloth fibers.

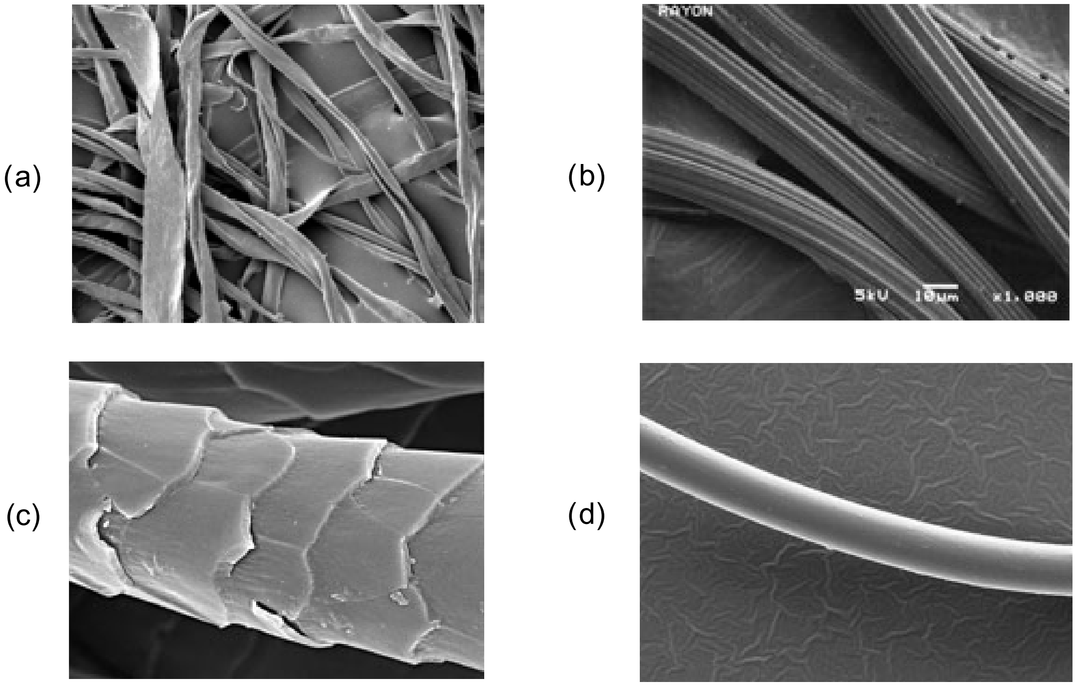


Figure 5.6: Microscope photos of different types of cloth fibers: (a) cotton, (b) rayon, (c) wool, and (d) polyester fibers. Images are from [Dek10].

5.2.3 Cloth Threads

Cloth threads, also referred to as *Yarns*, are long assembly of cloth fibers. Threads can be constructed from fibers in two different ways. Staple threads are made out of twisted fibers so that they stay together by friction (e.g. cotton threads). In contrast, filament threads are made out of straight long fibers (e.g. silk threads). Figure 5.7 shows different types of threads in a cloth fabric.

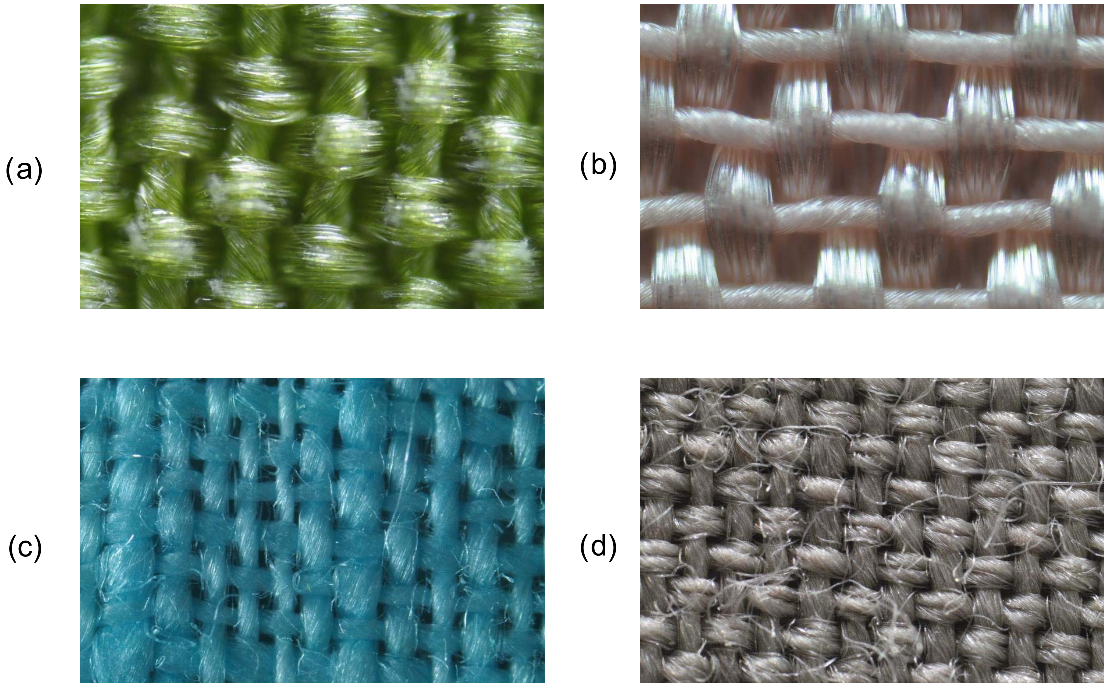


Figure 5.7: Microscope photos of different types of cloth threads and weaving patterns; (a) Silk threads interwoven with Crepe de Chine pattern. (b) Polyester threads interwoven in Satin Charmeuse pattern. (c) Linen threads interwoven in Plain pattern. (d) Wool threads interwoven in Tropical weave pattern.

5.2.4 Weaving Patterns

The complex and versatile appearance of cloth is due to the variations in cloth threads as well as the way they are interwoven together. Fabrics made out of same thread type but with different weaving patterns can have drastically different appearances. Some examples of weaving patterns can be seen in Figure 5.7. For more information regarding cloth fabrics refer to [SJK06].

5.3 Previous Work

Rendering cloth has been an active area of research for more than 25 years. Rendering realistic cloth is a considerable challenge because of its physical complexity and diversity. Consequently, the capabilities of most cloth models are limited; they can reproduce the appearance of only specific types of cloth or are unable to accurately match all of the observed reflectance properties. In the following section we go over prominent work in cloth rendering and mention their limitations.

5.3.1 Empirical Shading Models

The earliest approaches as well as more recent work are based on empirical shading models, where the primary goal is to achieve believable shading rather than physical accuracy. One suggested approach for rendering cloth employs Lambertian shading model and bump mapping [Wei86]. Daubert et al. [DLHS01] proposed a material representation based on the Lafortune reflectance function coupled with an occlusion term to handle several view dependent effects of coarsely woven and knitted fabrics. Glumac and Doepp [GD04] developed an anisotropic shader that extends a general lighting model to use thread direction parameters to achieve believable looks for several silk-like fabrics. Not concerned with physical accuracy, these models only provide an imitation of cloth appearance and typically lack the ability to replicate some of its more complex features.

5.3.2 Virtual Goniometric Methods

Westin et al. [WAT92] computed BRDFs for velvet and plain weave nylon fabrics through statistical ray tracing of a geometric model of the small-scale cloth structure. They modeled the threads of a nylon fabric as flat interwoven cylinders. They could simulate the appearance of velvet by modeling its microstructure as a collection of randomly orientated cylinders coming out of the surface of the fabric. Volevich et al. [VKKK97] extend this work by modeling individual fibers of the cloth in their simulation. Drago and Chiba [DC04] took a similar approach to simulate the optical properties of woven canvas fabrics. These models require a reconstruction of the fabric at the microstructure level.

5.3.3 Microfacet Based Models

Another method for replicating cloth appearance is to use microfacet-based shading models. Although these techniques are typically capable of producing accurate appearances at various levels of detail, the general nature of microfacet-based models makes it difficult to produce the right distribution without a densely sampled BRDF to use in a fitting step.

Ashikhmin et al. [APS00] presented a BRDF generator that could model satin and velvet. Others extended this work to handle a variety of weave patterns [AMTF03a; AMTF03b; AMTF03c]. Wang et al. [WZT^{+}08] introduced their own microfacet-based BRDF for modeling spatially-varying anisotropic reflectance using data captured from a single view.

While microfacet models can be effective at capturing a complex appearance these models are difficult to control as they depend strongly on the right microfacet distribution function. Since cloth is often anisotropic it is difficult to obtain this distribution from measured data. These distribution functions can be very complex and unintuitive.

Our work is similar to the model of Ashikhmin et al. [APS00] in the sense that they can reproduce a specular reflection in any direction by distributing the microfacets in the appropriate direction. We reproduce specular peaks by orienting the tangents so that their reflection cone lies in the desired direction. The input of our model is intuitively based on the weaving pattern of the fabric, while their model requires a complicated mathematical representation of the surface normals. For example, for producing the appearance of Velvet fabric, Ashikhmin et al. propose $c \times \exp(-\cot^2\theta/\sigma^2)$ to be the distribution of normals (where c and σ are control parameters). Formulating the correct equation can be a challenging task. In our model we define the tangents to be almost perpendicular to the surface normal and we can reproduce the appearance of Velvet.

5.3.4 Data Acquisition Approaches

Due to the complexity of the cloth appearance many attempted to model its appearance using data captured from a cloth fabric from varying directions [SSK03; WRG$^+$09]. These data get stored as compressed bidirectional texture function (BTF) [DvGNK99] and will be used during the rendering. Ngan et al. [NDM05] used measured BRDF of velvet and satin fabrics and fitted an analytical models to the measured data. All of these methods require data acquisition setup and high quality measurements of a specific fabric in order to model its appearance.

5.3.5 Physically Based Models

Another approach for simulating cloth is based on modeling the structure of the cloth. Yasuda et al. [YYTI92] modeled the gloss seen in cloth by accounting for the internal structure, but assumed a very simplified model of the structure and the results lacked verification. Xu et al. [XCL$^+$01] presented a volumetric approach called lumislice method in order to render close-up renderings of free-form knitwear. This work was later extended by Chen et al. [CLZ$^+$03]. While these methods can reproduce a wide range of appearances they can be difficult to control. Jakob et al. [JAM$^+$10] presented a volumetric rendering approach that could be applied to rendering cloth fabrics. Zhao et al. [ZJMB11] extended this work and presented a CT scanning method for acquiring an accurate representation of the cloth fabrics. Their model produces high quality renderings but it is limited to reproducing specific cloth samples. The requirement for CT scanning different fabrics make this model impractical.

Irawan et al. developed a comprehensive model for reproducing both the small-scale (BTF) and large-scale (BRDF) appearance of woven cotton cloth [IM06; Ira08]. This model is current state of the art and capable of reproducing a wide range of appearances. The basis of the model is a complex empirical model for light interacting with a cloth thread. He presents a rigorous model for computing the light reflection off of yarn threads, which are simulated as an assembly of specular fibers. His model incorporates numerical integrations (with no performance reports) and a fitting process to estimate the value of different control parameters. The numerical fit is rather costly and to reduce the number of parameters in the model only a specific set of cloth tangents can be accounted for. This model approximates the tangent distribution of threads with conic sections, which in turn limits the ability of his model to match complex or irregular tangent distributions. Due to this limitation he cannot model any BRDFs with more than two specular highlights (e.g. the Polyester Satin Charmeuse as shown in Figure 5.27 and 5.28) or discontinuous distributions such as Velvet. They do not provide such complex appearances in their rendering results. Furthermore, the model ignores shadowing and masking between different threads of cloth, which limits the accuracy at grazing angles (which is not presented in their measurements). Finally, the complexity of the model makes it difficult to control in order to achieve a specific appearance.

5.4 Light Scattering from Fabrics

5.4.1 Acquisition Setup

The cloth measurements presented in this paper were acquired with a fully automatic, four-axis image-based gonioreflectometer at University of California San Diego (UCSD). The device consists of two robotic arms, each with two degrees of freedom. Each degree of freedom has a minimum displacement of 0.01 degrees, allowing the arms to move freely to nearly any desired position about the hemisphere which surrounds the measurement platform. In order to capture the data, the outer arm is mounted with a CCD camera and the inner arm is mounted with a light source (see Figure 5.8).

In addition to quantitative analysis of cloth reflectance, we investigated different fabric types and their constituent threads under a microscope to gain further intuition about their behavior.

Figure 5.8: UCSD Gantry.

The terms used throughout this section derive from the setup in Figure 5.9. The incoming and outgoing directions are defined relative to the surface normal. The angle between the normal and the incoming light direction is θ_i and the angle between the normal and camera viewing direction is θ_r. Only in-plane measurements of the cloth samples were obtained (the light and camera rotate about the hemisphere in the same plane), so the relative position of the light and camera need only be represented by the incoming light direction ω_i and the outgoing direction ω_o. The angles ϕ_i and ϕ_r are collapsed into a fixed $\phi_d = \phi_i - \phi_r$ which is equal to 180^o for in-plane measurements.

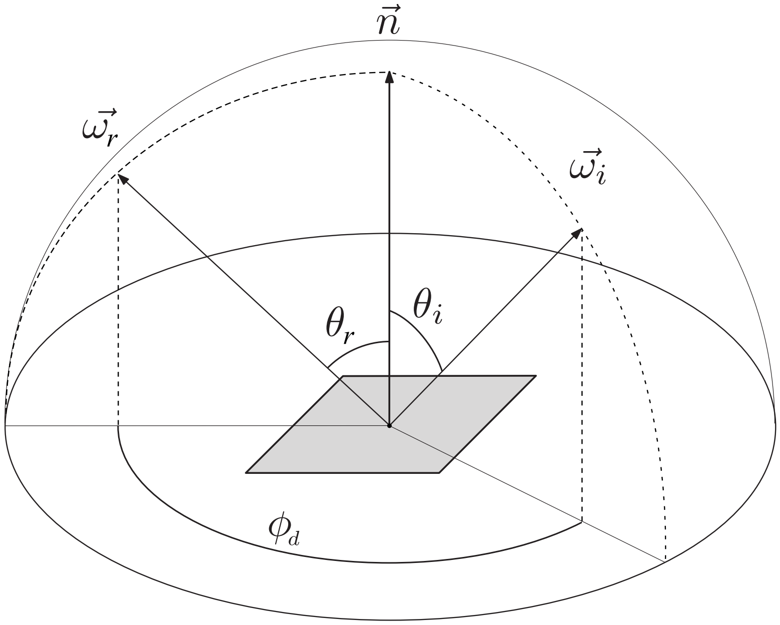


Figure 5.9: An illustration of the acquistion setup and coordinate system for measuring cloth.

5.4.2 BRDF Measurements and Observations

Our measurements show that the appearance of cloth is dominated by some combination (in varying amounts) of diffuse reflectance, specular reflectance, shadowing/masking, and grazing angle sheen. While many samples were measured, the three fabrics in Figure 5.2 are the focus of our analysis. we focused on three fabrics: Linen Plain, Silk Crepe de Chine, and Polyester Charmeuse. Since each has a unique combination of fiber type, thread structure, and weaving pattern, their measurements provide insight as to which physical characteristics are responsible for the variations in their appearance. The resulting set of observed behaviors produced by these fabrics have not been fully addressed in literature nor have they been validated with ample physical measurements.

5.4.2.1 Linen Plain

The measured Linen sample is a plain weave fabric assembled with a single type of thread (see Figure 5.3 left). This particular construction causes the material to look the same both front and back, as well as from perpendicular viewing angles. Under the microscope, as shown in Figure 5.10, we observe a repeating grid of twisted threads.

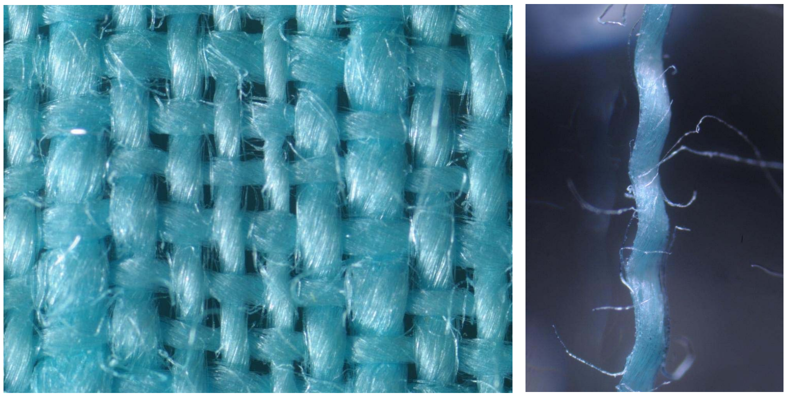


Figure 5.10: Microscope pictures of the Linen Plain fabric. The fabric (left) is made from one type of thread (right) with varying thinknesses.

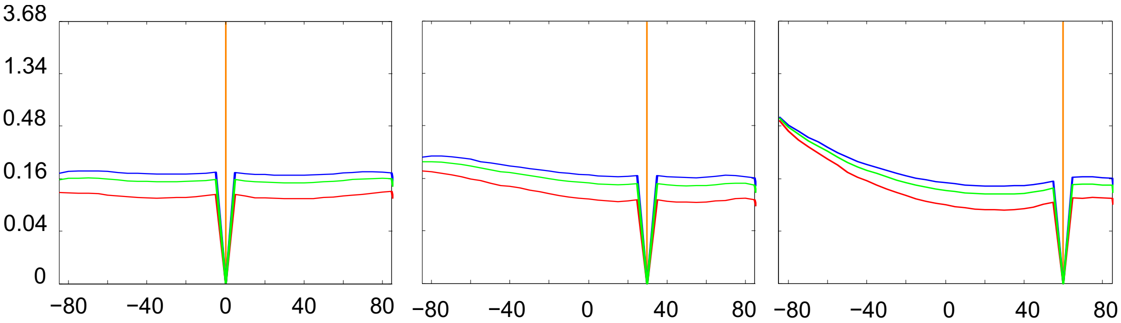


Figure 5.11: BRDF measurements for Linen Plain for 0^o, 30^o, and 60^o incident angles.

Due to its orthogonally symmetric structure, linen was measured in only one direction. Figure 5.11 shows the normal plain BRDF measurements of this fabric along one of its threads. The orange line indicates the direction of incident light. The drop in the plots are due to the occlusion of the light source by the camera.

The measurements confirm that linen produces a smooth reflection with no specular peaks under most lighting conditions, except when the fabric is observed at a grazing angle and the light is also grazing. As seen in Figure 5.11 (right), at these grazing angles, reflectance increases substantially. The measurements also show that the effect of shadowing/masking, manifested by dips at the plot edges, is minimal.

5.4.2.2 Silk Crepe de Chine

The measured Silk Crepe de Chine sample is assembled with two different types of thread (Figure 5.12). The first type of thread is made of densely twisted fibers. This thread remains straight and uniformly spaced in the fabric. The second type of thread is made of thin and untwisted fibers, and passes above and below the first type (see Figure 5.3 middle). This thread exhibits sharp surface reflection and very little absorption resulting in its translucent appearance. While moving a light around the microscope, a strong specular reflection in two incidence directions is visible (see Figure 5.12 right).

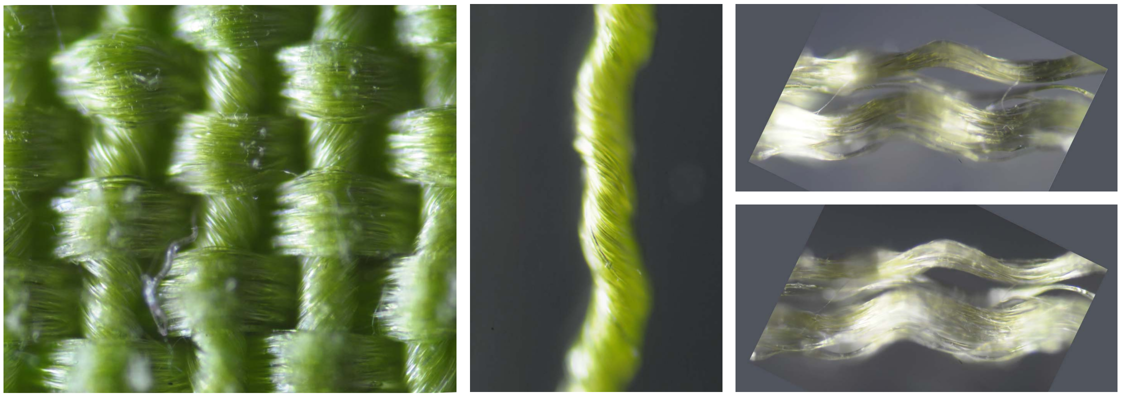


Figure 5.12: Microscope pictures of the Silk Crepe de Chine fabric. The fabric (left) is made out of two different types of thread: A set of dense twisted threads (middle) and a series of flat shiny threads (right) which go above and below the other threads.

The variation in thread type as well as the weaving pattern structure result in an asymmetrical surface which causes this fabric to appear significantly different in perpendicular viewing directions. To demonstrate this, two perpendicular planes of silk were measured (Figure 5.13). Measurements in the plane parallel to the flat threads (top row) show two off-specular peaks, while the perpendicular plane measurements (bottom row) exhibit two grazing angle peaks. Furthermore, the parallel measurements clearly indicate a drop in reflectance as the eye approaches grazing angle, suggesting the contribution of shadowing/masking. In contrast, the perpendicular measurements maintain the grazing angle peaks under all lighting conditions.

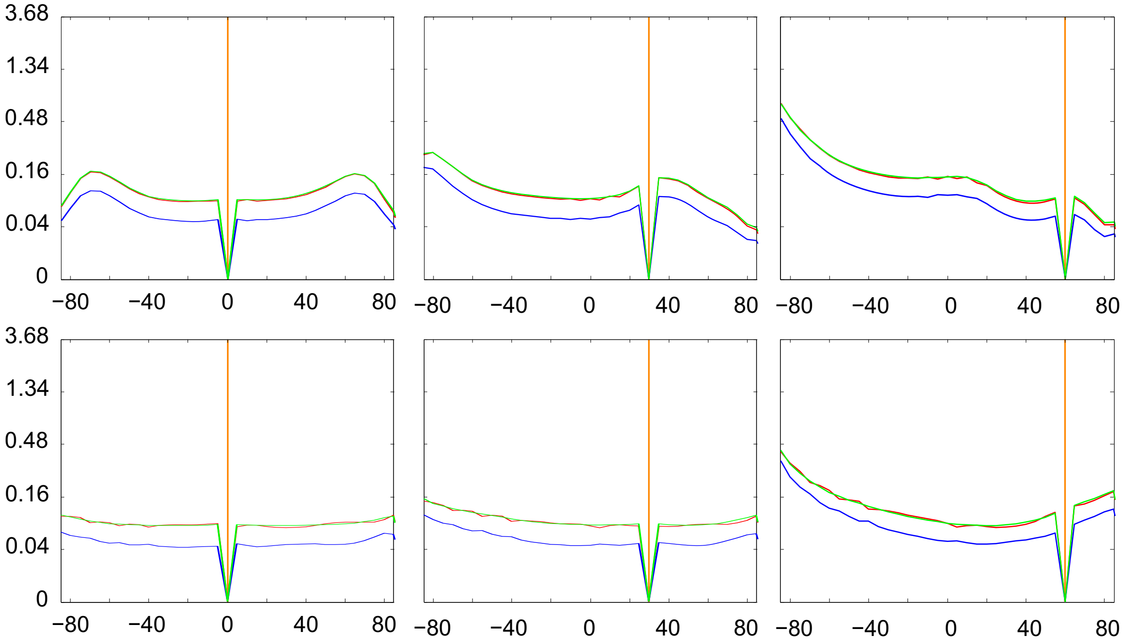


Figure 5.13: BRDF measurements for Silk Crepe de Chine for 0^o, 30^o, and 60^o incident angle. The top row corresponds to the in plane measurements parallel to the direction of flat threads, and the bottom row represents the measurements in the perpendicular direction.

5.4.2.3 Polyester Satin Charmeuse

The measured Polyester Satin Charmeuse sample is a satin weave fabric, meaning that the threads in one direction cross over most of the threads in the other direction (see Figure 5.3 right). Like Silk, this fabric is made out of two distinct (polyester) threads. The flat threads go above and below the twisted threads, but remain longer above than below. This asymmetry in the weaving pattern causes the fabric to have two different sides (Figure 5.15). While moving the light around the microscope, we noticed strong reflections in three different direction of light as can be seen in Figure 5.14 right.

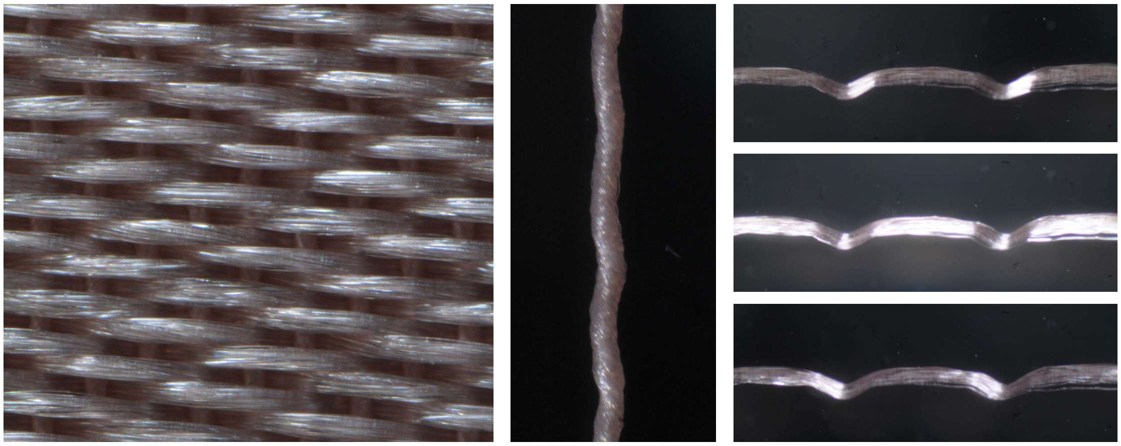


Figure 5.14: Microscope pictures of the Polyester Satin Charmeuse fabric. The fabric (left) is made out of two different types of thread: A set of dense twisted threads (middle) and a series of flat shiny threads (right) which go above and below of the the other threads. Inset shows the reverse side of the fabric.

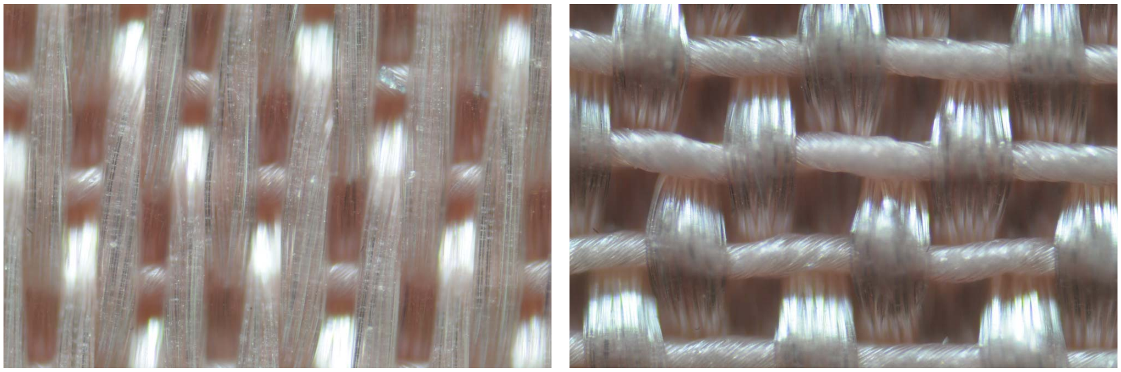


Figure 5.15: The front and back of the Polyester Satin Charmeuse fabric.

The variation in thread type and the asymmetric weaving pattern result in strong anisotropic reflectance. The fabric was measured in two perpendicular planes (Figure 5.16). In plane measurements along the direction of flat threads exhibit three specular peaks, one in the reflection direction and the other two in equal but opposite off-specular directions. Measurements in the perpendicular plane exhibit grazing angle peaks which are visible under all lighting conditions.

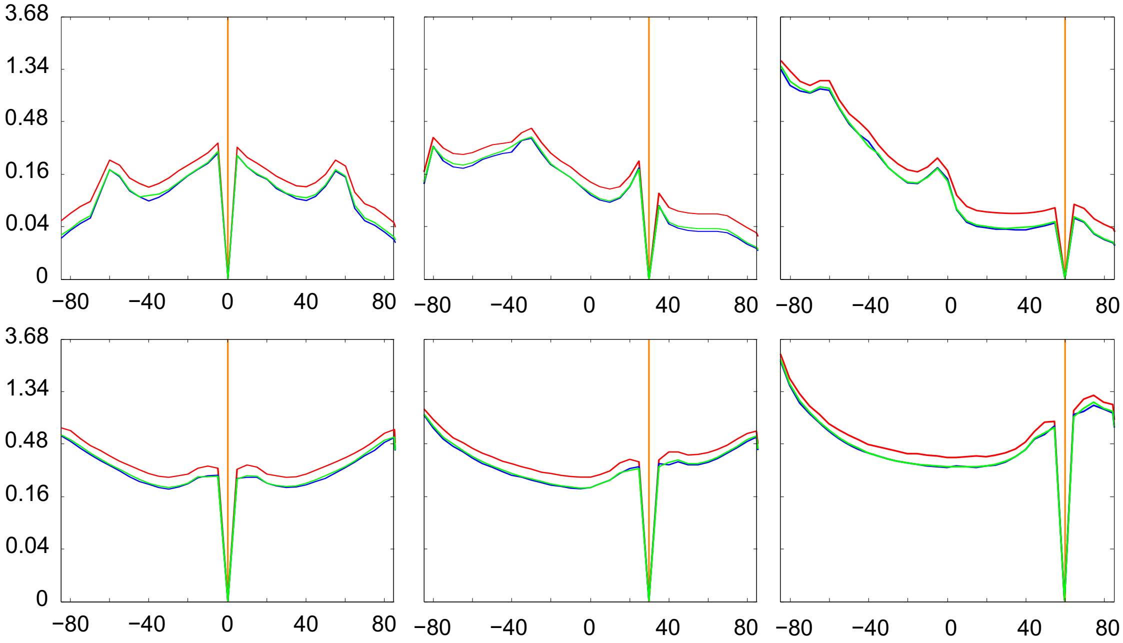


Figure 5.16: BRDF measurements for Polyester Charmeuse for 0^o, 30^o, and 60^o incident angles. Top row corresponds to in plane measurements along the direction of flat threads, and the bottom row represents measurements in the perpendicular direction.

5.5 Light Scattering from Threads

In the previous section we noted that the there are two different types of threads that contribute to the overall appearance of cloth fabrics. The first and most common type of threads are densely twisted threads. These threads have many varieties that differ by twist level and constituent fiber count. The twist level of threads affects the compactness and density of fibers that compose them [Sav99]. The second type of threads have a minimal amount of twist in their construction and we will refer to them as flat threads. These are usually less dense and have a greater diameter due to their loosely packed structure. Both thread categories are fed by a diverse selection of raw materials such as silk, cotton, wool, flax, and synthetic filaments. We further investigate the light scattering properties of cloth fabrics by measuring the BRDF of individual threads of different type.

5.5.1 Acquisition Setup

We measured the radiance distribution of several thread types using the same spherical gantry as used for the cloth measurements and a unique suspension apparatus. The results served to validate our analytical model as well as provide a qualitative basis for reasoning about threads and cloth in general. In our measurements we illuminate an 8 cm section of thread with a collimated light beam and collect radiance scattering measurements with a CCD camera.

To procure a thread sample, we first remove a single strand from a finished fabric. When a thread is removed from fabric it is no longer straight, but retains the shape that it had in the fabric. In order to obtain accurate scattering measurements, the thread must be extended to its maximal length. This type of procedure is common in fabric quality testing and requires standard tension, which has the general goal of non-destructively pulling on one end of the thread. In our experimental thread mount, we clamp one end of the thread to a poseable arm, and let the rest of it hang, weighed down by a magnetic set of spheres at the unclamped end. Hanging the thread in mid-air allows the gantry to measure a full 4D BRDF with minimal occlusions and no background to contaminate the measurements. Additionally, gravity provides a vertically straight thread orientation which eliminates pose calibration.

5.5.2 BRDF Measurements

We measured a complete 3D BRDF by varying the longitudinal angles θ_i, θ_r, and the azimuthal difference angle $\phi_d = \phi_i - \phi_r$. Figure 5.4 shows the notations used in

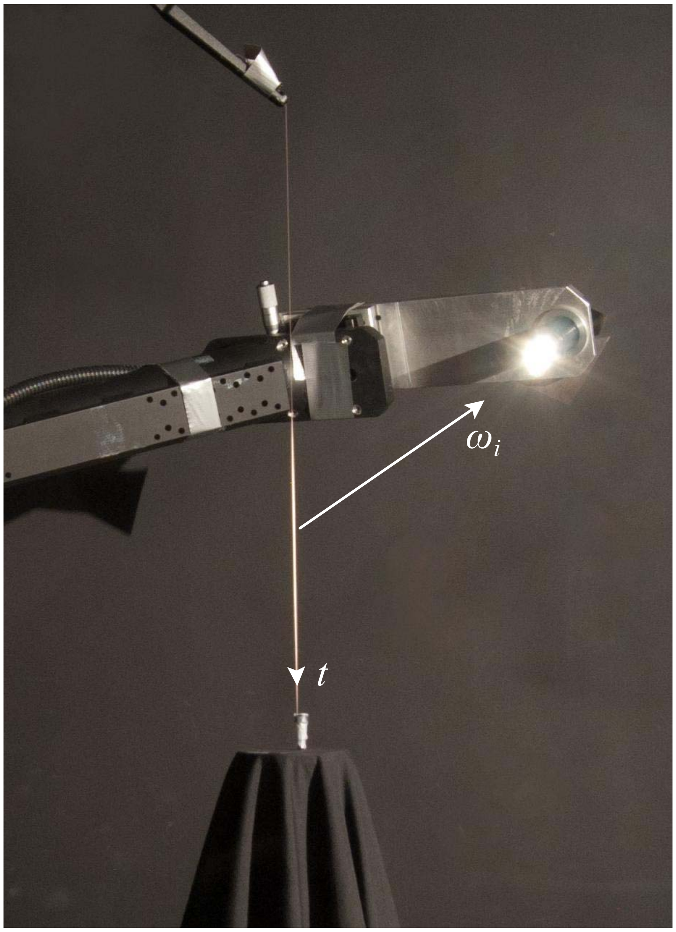


Figure 5.17: An illustration of the acquistion setup for measuring the BRDF of threads.

our paper. We did not measure a 4D BRDF because we assumed symmetry of the BRDF with respect to ϕ. Since threads are not perfect cylinders, this assumption is somewhat violated, however, it allows us to capture less data while still observing the salient thread scattering features. We present a planar slice of the resulting measurements in Figure 5.18. Here the BRDF is a function of two angles θ_i and θ_r). We present several θ_i angles and plot a continuous range of BRDF measurements for θ_r. The threads were not treated with any dyes and no polarizing filters were used. As a result, the BRDF plots represent the natural visible combination of surface reflection and internal scattering. To facilitate intuition about the plots we can state the following: surface reflection results in a lobe in the specular reflection direction ($\theta_r = -\theta_i$), and internal scattering results in a wider lobe that is more decoupled from appearing in the specular direction.

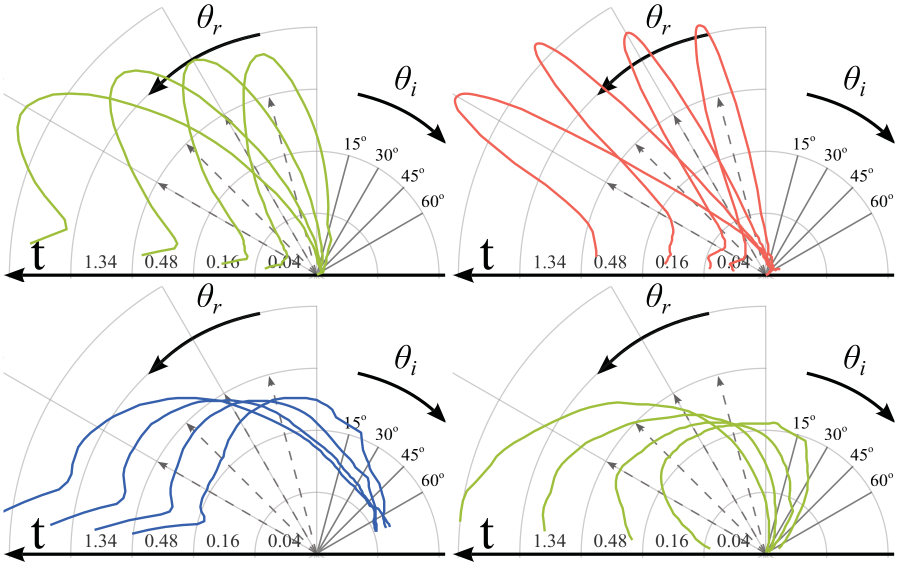


Figure 5.18: Polar plots of measured incidence plane BRDF for different threads. Each quadrant of the figure contains the rgb average BRDF for a thread type. Starting at the top row, and moving left to right we have flat silk thread, flat polyester thread, twisted linen thread, and twisted silk thread.

The top row of Figure 5.18 demonstrates the similarity among flat threads and their disparity from twisted threads. In the top row, both the polyester and the silk thread possess narrow specular lobes oriented at the exact specular reflection direction. This result can be attributed to their low surface roughness as well as minimal internal scattering. The fact that the lobe is oriented at the exact specular reflection direction means that, unlike hair, threads have no consistent cuticle that displaces their specular reflection. The polyester thread is the more specular of the two flat threads, as evidenced by its narrower and brighter reflection lobe. This can be attributed to the synthetic vs. organic fibers that they are composed of, where polyester has fewer natural imperfections and irregularities due to its industrial fabrication process.

At first glance the twisted threads in the bottom row of Figure 5.18 appear nearly identical. They both exhibit a characteristic wide scattering lobe that slowly increases as the light goes to glancing angle. Focusing on glancing incidence angles, we observe that the twisted linen thread scatters more light in the non-specular directions. This type of BRDF can be attributed to either a very rough surface or isotropic internal scattering. We address these behaviors in subsequent sections when we present our thread BRDF model.

The incidence plane plots in Figure 5.18 and 5.20 were produced by holding the light fixed at a specified angle and moving the camera in-plane from 6^o to 160^o. This range is determined by the angular limitations of the gantry as well as the CCD camera limitations at observing glancing angles.

5.5.3 A Light Scattering Model for Threads

Based on our measurements of individual threads, we observed an optical behavior that is similar to hair and more generally, smooth dielectric cylinders. We observed that the reflection of a collimated light beam from a taut thread sample forms a cone centered on the thread axis. Additionally, the surface reflection is framed by a subtle color reflection that is also centered at the cone. This type of reflection from cylinders has been previously studied by [KK89; Kim02; MJC+03], where the normal plane around the tangent is used as the coordinate frame for computing light scattering behavior. Unlike hair, threads do not have tilted cuticles on their surface and therefore the reflected rays will stay on the reflection cone regardless of the number of bounces inside the thread.

In our model we abstract the thread geometry and optical behavior with a cylinder. When a beam from ω_i consisting of parallel rays of light strikes a thread cylinder running along the vector t, each ray in the beam reflects at the surface according to the surface normal of the cylinder. These surface normals are all perpendicular to the thread tangent vector t and lie in the normal plane. For a smooth specular cylinder, a beam incident at θ_i will be reflected in the ideal specular direction $-\theta_i$ across the normal plane and, due to the circular cross section of the cylinder, will be spread into a cone [KK89]. See Figure 5.19 for a visualization of the reflection

Table 5.1: Description of important symbols used in this section.

Symbol	Description
f_s	Thread scattering function
F_r	Fresnel reflectance
F_t	Fresnel transmittance
γ_s	Surface reflectance Gaussian width
γ_v	Volume scattering Gaussian width
k_d	Isotropic scattering coefficient
A	Colored albedo coefficient

cone. The refracted light will enter the cylinder and after any number of internal reflections and refraction will emit into the same cone as the surface specular reflection [MJC$^+$03].

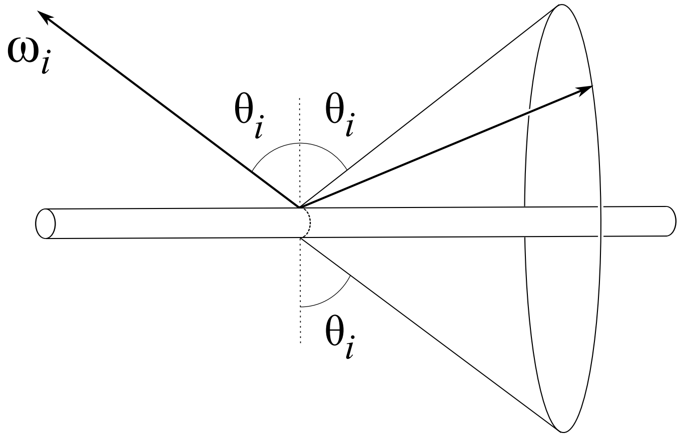


Figure 5.19: Reflection cone off of a smooth cylinder.

To establish radiometric notation for our cylinder based model we use the curve radiance integral from [MJC$^+$03]:

$$L_r = \int f_s(t, \omega_i, \omega_r) L_i(\omega_i) \cos\theta_i \, d\omega_i \tag{5.6}$$

Note that unlike the standard radiance integral on a surface, the reflected radiance from a cylinder differs by the fact that it is defined over a unit length instead of a unit area. This difference arises from the fact that the cylinder scattering function accounts for all the light scattered around the circumference of the cylinder.

As in previous treatments of BSDFs [HK93], we separate our scattering function $f_s(t, \omega_i, \omega_r)$ into a surface scattering component $f_{r,s}$, and volume scattering component $f_{r,v}$. In addition to the angles in Figure 5.4, we introduce $\phi_d = \phi_i - \phi_r$ and $\theta_h = (\theta_i + \theta_r)/2$ to define the two scattering functions.

5.5.3.1 Surface Reflection

We model surface reflection similarly to [MJC+03], except we do not decompose our computation into longitudinal and azimuthal planes.

$$f_{r,s}(t, \omega_i, \omega_r) = F_r(\eta, \vec{w_i}) \cos(\phi_d/2) g(\gamma_s, \theta_h) \tag{5.7}$$

The $\cos(\phi_d/2)$ term arises due to projection of the circular cylinder cross-section, as demonstrated by [Kim02], and previously used by [SPJT10] for hair rendering. To break away from the idealized smooth cylinder representation of threads, we employ a unit area Gaussian g with width γ_s to simulate surface roughness. Finally, we add physical accuracy to the model by attenuating the power by a Fresnel term. The actual angle used to compute the Fresnel term is based on the reflection normal on the cylinder as well as a half-angle between the light and the eye, yielding an exact expression:

$$F_r(\eta, \arccos(\cos(\theta_d/2) \cos(\phi_d/2))). \tag{5.8}$$

This model produces a glossy reflection on a cone around the thread with physical and geometric attenuation. We considered using the full micro-facet specular formulation, but found that it did not improve matching to our measured results.

5.5.3.2 Volume Scattering

Real threads are composed of fibers that are either twisted together or lay flat next to each other. We make a unifying assumption that all fiber types are cylindrical with minimal eccentricity. This is generally true with the exception of cotton, which resembles a flat ribbon. To summarize, our model is a large thread cylinder composed of tiny fiber subcylinders. This enables us to use the fact that smooth cylinders emit light due to internal scattering into the ideal reflection cone. Therefore, light that enters the thread volume and undergoes any type of scattering interaction with the fiber subcylinders will result in a surface emission into the same cone as the surface reflection.

One thing to note is that the orientation of the fiber subcylinders differs from that of the thread cylinder. We model this deviation as a normal distribution centered on the thread tangent. Therefore, a flat thread will have a much smaller variance than a twisted thread. We model these behaviors while maintaining physical constraints in the following equation:

$$f_{r,v}(t, \omega_i, \omega_r) = F \; \frac{(1 - k_d) \; g(\gamma_v, \theta_h) + k_d}{\cos\theta_i + \cos\theta_r} \; A \tag{5.9}$$

Here F is the product of two transmission Fresnel terms:

$$F = F_t(\eta, \vec{w_i}) F_t(\eta', \vec{w_r'}) \tag{5.10}$$

We define the subcylinder tangent deviation with a Gaussian lobe g with width γ_v. The Gaussian lobe controls the width the forward scattering cone. For twisted threads, which consist of fibers that deviate from the thread tangent direction, this Gaussian is wider than for flat threads which mainly consist of parallel filaments. Additionally, we define a tunable isotropic scattering term k_d and a color albedo term A. We added an isotropic scattering term to account for cellulose based fibers such as cotton and linen, which predominantly yield isotropic volume scattering instead of a forward scattering cone. The division by the sum of projected cosines comes from [Cha60], in his derivation for diffuse reflectance due to multiple scattering in a semi-infinite medium. Adding this normalization term gave us better matches with our measured results. The complete thread scattering model is a sum of the surface and volume components:

$$f_s(t, \omega_i, \omega_r) = \big(f_{r,s}(t, \omega_i, \omega_r) + f_{r,v}(t, \omega_i, \omega_r)\big) / \cos^2 \theta_d \tag{5.11}$$

Note that the complete scattering formulation contains a division by $cos^2\theta_d$, which is necessary to account for the solid angle attenuation of the specular cone [MJC+03]. Previous work has addressed volume scattering in threads with a cylindrical phase function in [Ira08] as well as the Henyey-Greenstein phase function in [AMTF03c]. We experimented with various phase functions as well, but found them inadequate due to their decoupled behavior from the direction of the thread. Our approach is similar in spirit to [JAM+10], which defines phase functions oriented to the direction of fibers to achieve highly anisotropic volume scattering.

We have defined a complete BSDF for individual threads, which matches well to our measured results. It was our goal to define as few non-physical control parameters as possible to enable the physical and geometric scattering constraints to drive its behavior. We note that the model is only suitable for distant viewing of threads since it assumes that the rays of light incident on the thread cylinder are parallel and that the thread is locally straight.

5.5.4 Matching Measured BRDFs

In this section we provide evidence for the validity of our model by comparing it to measured BRDFs of thread. We accomplish this by manually fitting our model parameters to measured results. We did not consider automatic fitting approaches due to the minimal number of control parameters and their predictable nature.

In Figure 5.20, we demonstrate the performance of our model with measured BRDF

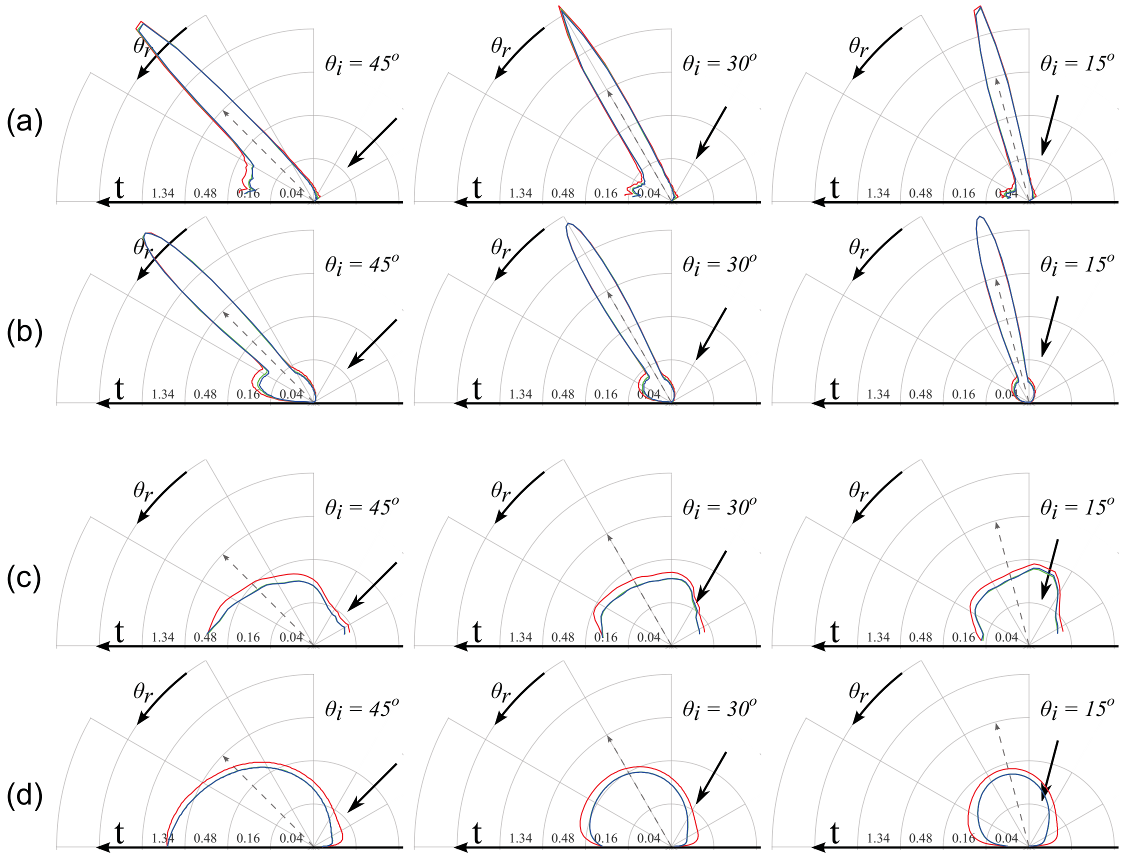


Figure 5.20: Incidence plane thread BRDF measurements in the top row matched by thread model in the bottom row. In each row the first three plots are for a twisted thread and the last three for a flat thread. The two threads were extracted from the same Polyester Satin Charmeuse cloth sample. The plots show scattering as a function of view angle.

results in the top row and our model in the bottom row. Each row shows BRDF measurements for three incident light angles of two thread types: one twisted, and one flat. The first three plots in each row correspond to a twisted polyester thread extracted from the Polyester Satin Charmeuse cloth sample. Our model results in the bottom row are able to closely match the measured results in the top. We achieve this by observing a wide surface reflectance Gaussian supplemented by an even wider volume Gaussian and a red tinted albedo coefficient.

The second set of three plots in Figure 5.20 correspond to a flat polyester thread from the same cloth sample. Our model closely simulates the scattering profile of this thread by setting a very narrow surface reflectance Gaussian and a small red tinted albedo. Our model is validated by being able to closely simulate the scattering behavior of different thread types under various incident light angles.

5.6 An Appearance Model for Cloth

We consider cloth fabric as a mesh of interwoven smooth cylinders oriented in two orthogonal directions. These cylinders, which we will refer to them as microcylinders, are considered to be very small compared to the geometry of the fabric. We use texture UV coordinates of the mesh as the direction of microcylinders but any other direction can be used. As discussed in section 5.5.3, we do not rely on a specific surface normal in our cylinder scattering model and therefore need only tangent directions at the cloth level (Section 5.6.1). However, the surface normal does come into play in shadowing/masking calculations (Section 5.6.2).

5.6.1 Shading Model

In order to render cloth fabrics, we evaluate the outgoing radiance from the *smallest patch* of the weaving pattern. This patch is the smallest portion of the weaving pattern which has the following property: the complete weave can be constructed by repeating this patch. Note that the smallest patch is not unique since all of them contain the same set of tangents even though they come from different threads (see Figure 5.22 left).

We assume that the smallest patch is locally flat and smaller than a pixel in the image plane. Additionally, for clarity, we constrain our discussion to cloth patches that contain exactly two threads, one orthogonal to the the other, as is common in most weaving patterns. However, the formulations in this section can be trivially extended to compute the contribution from any number of threads in a smallest patch.

We define the outgoing radiance of the smallest patch to be the weighted average of the outgoing radiance of constituent threads based on their local orientation and coverage inside the smallest patch (See Figure 5.21):

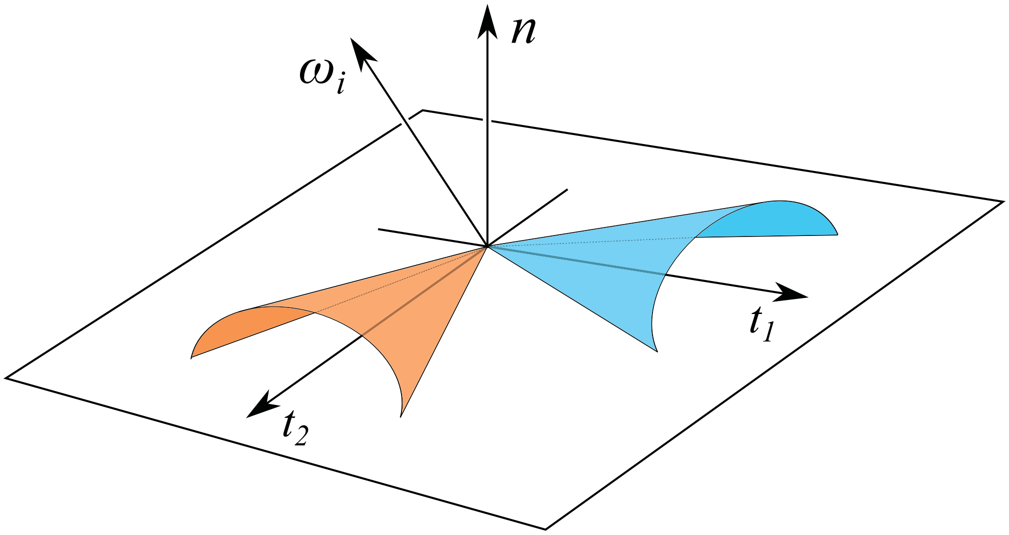


Figure 5.21: Our shading model treats the fabric as a mesh of microcylinders oriented in two orthogonal directions.

$$L_r(\omega_r) \; = a_1 \times L_{r,1}(\omega_r) + a_2 \times L_{r,2}(\omega_r) \tag{5.12}$$

where a_1 and a_2 represent the area coverage ratio of the first and second thread within the smallest patch respectively. If the weaving pattern has no empty spaces, these two numbers add up to one.

For each thread, we define a *tangent curve* that describes its tangent distribution inside the smallest patch (Figure 5.22 right). We specify the tangent curve by setting the tangent values at discrete control points. In order to compute the total radiance of each thread, we sample its corresponding tangent curve at T equidistant points and compute a weighted radiance contribution of the samples according to the following formula:

$$\begin{aligned} L_{r,j}(\omega_r) &= \int \frac{\sum_{t \in C_j} f_s(t, \omega_i, \omega_r) L_i(\omega_i) \cos\theta_i}{2\ T}\, d\omega_i \\ &= \frac{1}{2\ T} \int L_i(\omega_i) \Big(\sum_{t \in C_j}^{T} f_s(t, \omega_i, \omega_r) \cos\theta_i \Big)\, d\omega_i \end{aligned} \tag{5.13}$$

where j can be either 1 or 2 and represents the thread direction, C_j is the set that contains the T sampled tangents from the tangent curve of the corresponding thread, and f_s is the analytical thread BSDF model introduced in Section 5.5.3 . We are dividing by 2 T since there are T tangent samples for each direction of threads.

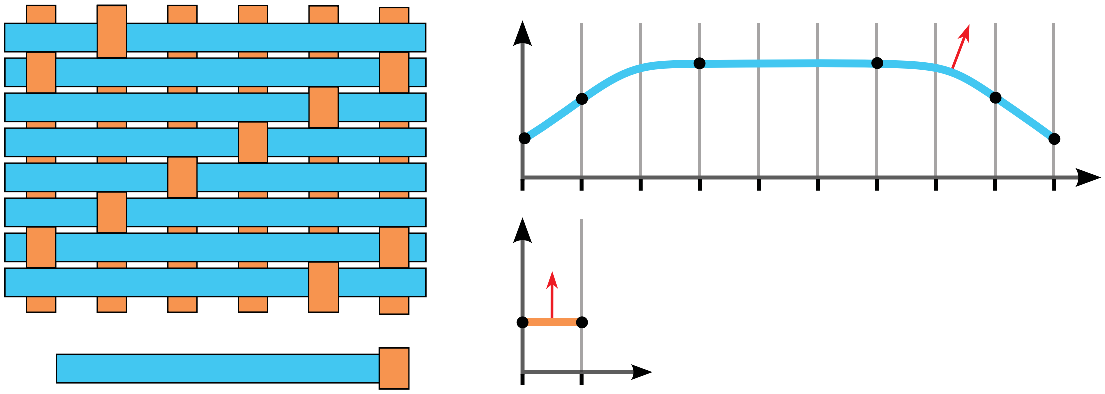


Figure 5.22: The weaving pattern and a sample tangent curve for the Polyester Satin Charmeuse fabric: (left top) the weaving pattern, (left bottom) a smallest patch, (right) the tangent curve for the two types of threads. The red arrows indicate the local normal of the tangent which is used in the shadowing and masking calculations.

5.6.2 Shadowing and Masking

Shadowing and masking are very important for the correct evaluation of the outgoing radiance especially at grazing angle viewing and lighting directions. Poulin and Fournier [PF90] derived a shadowing and masking term for grooved surfaces composed of cylinders. However, their approach is not applicable to our model since they assumed that the cylinders have a surface patch BRDF and integrated all of the reflected light scattered toward a viewer. Since our formulation treats cylinders as one-dimensional entities, we do not compute the explicit reflectance variation across their circumference.

Shadowing and masking are very similar concepts; shadowing can be thought of as masking from the point of view of the light source. We interchangably refer to both of these quantities as masking M in the rest of this section. We only compute the masking between the same types of threads (i.e. threads with same overall directions). Shadowing between threads with different orientations is more involved and is left as future work.

Consider the setup shown in Figure 5.23 where the fabric is wrapped around a cylinder. Let us first focus on the horizontal threads only (Figure 5.23 middle). Threads along this direction never occlude each other from the viewer even at grazing angles. Therefore, the cylinder BSDF defined in Section 5.5.3 alone can be used to compute the correct outgoing radiance from these types of threads with no masking adjustment.

Now let us consider the vertical threads (Figure 5.23 right). At grazing angles each thread partially masks the thread behind it and gets masked by the thread in front of it. The amount of masking is relative to the cosine of the viewing direction projected to the thread normal plain and the surface normal. This angle is equal to ϕ_i (see Figure 5.4).

$$M(t, \omega_i) = \max(\cos\phi_i, 0) \tag{5.14}$$

If the cosine is negative, the surface is a backface and is being self-masked. The same argument holds for the light direction and results in shadowing.

$$M(t, \omega_r) = \max(\cos\phi_r, 0) \tag{5.15}$$

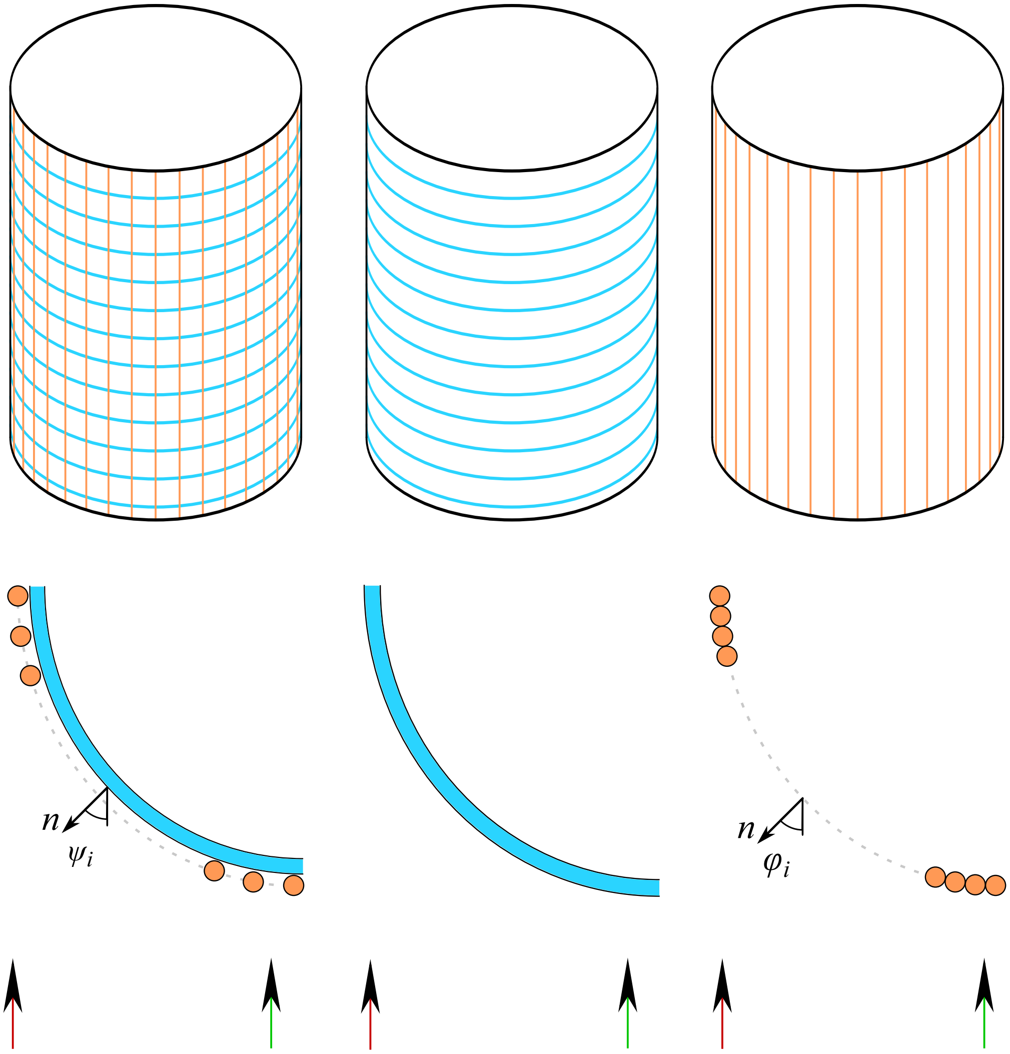


Figure 5.23: Fabric as two different directions of threads with views from above (green arrows) and from grazing angles (red arrows): (left) The contribution of different threads in the smallest patch is related to the orientation of the patch. At grazing angles, the blue thread contributes less than the orange thread. (middle) In the longitudinal direction, there is no masking and no adjustment needed.(right) In the azimuthal direction, the amount of masking in grazing angles is dependent on the $\cos \phi_i$.

Here ϕ_i and ϕ_r are computed with respect to the local normal of the tangent t. If the tangent deviates from the surface tangent by α degrees then its normal will deviate from the surface normal by α degrees as well. See Figure 5.22 for an illustration.

When ω_i and ω_r are not correlated, the overall shadowing and masking amount is equal to the multiplication of $M(t, \omega_i)$ and $M(t, \omega_r)$. In cases where these two directions are close to each other (e.g. driving at night), we use the adjustment introduced by Ashikhmin et al. [APS00] to compute the overall shadowing and masking term $M(t, \omega_i, \omega_r)$:

$$\begin{aligned} M(t, &\omega_i, \omega_r) = \\ &(1 - u(\phi_d))\ M(t, \omega_i) \times M(t, \omega_r) + \\ &u(\phi_d)\ \min(M(t, \omega_i), M(t, \omega_r)) \end{aligned} \tag{5.16}$$

where u is a unit height Gaussian function with standard deviation between 15^o and 25^o [APS00]. We will refer to $M(t, \omega_i, \omega_r)$ in short as $M(t)$.

We can rewrite Equation 5.13 to include the effect of shadowing and masking:

$$\begin{aligned} &L_{r,j}(\omega_r) \ = \\ &\frac{1}{2T} \int L_i(\omega_i) \Big(\sum_{t \in C_j} f_s(t, \omega_i, \omega_r) M(t) \cos\theta_i \Big)\ d\omega_i. \end{aligned} \tag{5.17}$$

To see the effect of shadowing and masking see Figure 5.33.

5.6.3 Reweighting

So far we have considered that the contribution of a tangent to the overall reflection of the smallest patch is based on its length (i.e. area coverage). This is only correct when the ω_r and ω_i are near surface normal n. We need to adjust the contribution of each tangent t based on its projected length $P(t, \omega_i)$ onto the image plane. Tangents that are more visible inside the smallest patch will have a higher contribution (for that viewing angle). We refer to this adjustment as reweighting. This process determines the contribution of each tangent curve sample to the overall reflectance of the smallest patch.

Projection of the tangents onto the image plane is based on the cosine of the longitudinal angle ψ_i. As shown in Figure 5.5.

$$P(t, \omega_i) = \max(\cos\psi_i, 0) \tag{5.18}$$

When the cosine is negative, the tangent is being self-masked and contributes zero to the overall reflection of the patch. Similar to the masking term, we calculate the projection for both ω_i and ω_r directions. This means that tangents receive energy based on their visibility from the point of view of the light source. We combine these two projections to get the final projection term $P(t, \omega_i, \omega_r)$:

$$\begin{aligned} P(t,&\omega_i, \omega_r) = \\ &(1 - u(\psi_d))\ P(t, \omega_i) \times P(t, \omega_r) + \\ &u(\psi_d)\ \min(P(t, \omega_i), P(t, \omega_r)) \end{aligned} \tag{5.19}$$

where ψ_d is the difference between ψ_i and ψ_r. We refer to $P(t, \omega_i, \omega_r)$ in short as $P(t)$. Finally we can rewrite Equation 5.17 to get our complete shading model:

$$\begin{aligned} &L_{r,j}(\omega_r) \ = \\ &\int L_i(\omega_i) \Big(\frac{\sum_{t \in C_j} f_s(t, \omega_i, \omega_r) M(t) P(t) \cos\theta_i}{\sum_{k \in C} P(k)} \Big)\ d\omega_i. \end{aligned} \tag{5.20}$$

where $C = C_1 \cup C_2$ is the set that contains 2 T sampled tangents from the tangent curves of both threads. To see the effect of reweighting on the final result see Figure 5.33.

5.7 Results

We have implemented the cloth model in a ray tracer and on the GPU. This section contains rendered results for several different cloth fabrics. The parameters for each cloth sample is summarized in Table 5.2 and the tangent distribution is given in Appendix E.

We have matched our rendered results against photographs. To capture the anisotropic behavior of different fabrics, we have wrapped the fabrics around a cylinder in three different directions. We label each mode based on the orientation of the flat threads as vertical, horizontal, and diagonal (see Figure 5.24). For the Linen Plain fabric, the vertical and horizontal modes are identical. For comparison, we present our rendered results of different fabrics in the same setup.

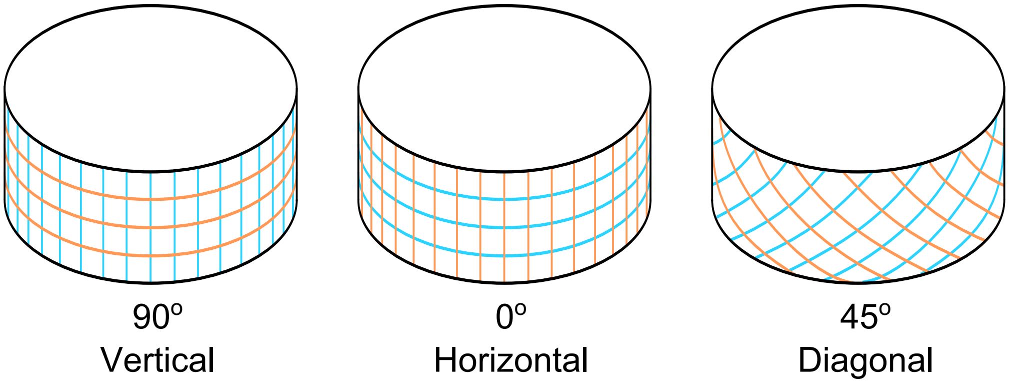


Figure 5.24: To capture the anisotropic behavior of different fabrics, we have wrapped the fabric around a cylinder in three different orientations where the flat threads stay (left) vertical, (middle) horizontal, and (right) diagonal.

Table 5.2: The list of parameters obtained from our measured cloth samples. The γ parameters are measured in degrees.

Fabric	η	**Thread**	A	k_d	γ_s	γ_v
Linen Plain	1.46	Both	$(0.20, 0.80, 1.00) \times 0.3$	0.3	12	24
Silk Crepe de Chine	1.345	Flat	$(1.00, 0.95, 0.05) \times 0.12$	0.2	5	10
		Twisted	$(1.00, 0.95, 0.05) \times 0.16$	0.3	18	32
Polyester Satin Charmeuse	1.539	Flat	$(1.00, 0.37, 0.30) \times 0.035$	0.1	2.5	5
		Twisted	$(1.00, 0.37, 0.30) \times 0.2$	0.7	30	60
Silk Shot Fabric	1.345	Dir 1	$(0.10, 1.00, 0.40) \times 0.2$	0.1	4	8
		Dir 2	$(1.00, 0.00, 0.10) \times 0.6$	0.1	5	10
Velvet	1.46	Both	$(0.05, 0.02, 0.00) \times 0.3$	0.1	6	12

Figure 5.25 shows our results for the Linen Plain fabric. The top image is the photograph of the fabric and the bottom image is our rendering. The graphs show the average values of the pixels on the y-axis. This fabric shows similar behavior on the vertical and horizontal mode due to the symmetry of the plain weaving pattern. However, it has a different appearance in the diagonal mode, which demonstrates the subtle anisotropic behavior of this fabric. Our renderings qualitatively match the photographs in all three cases.

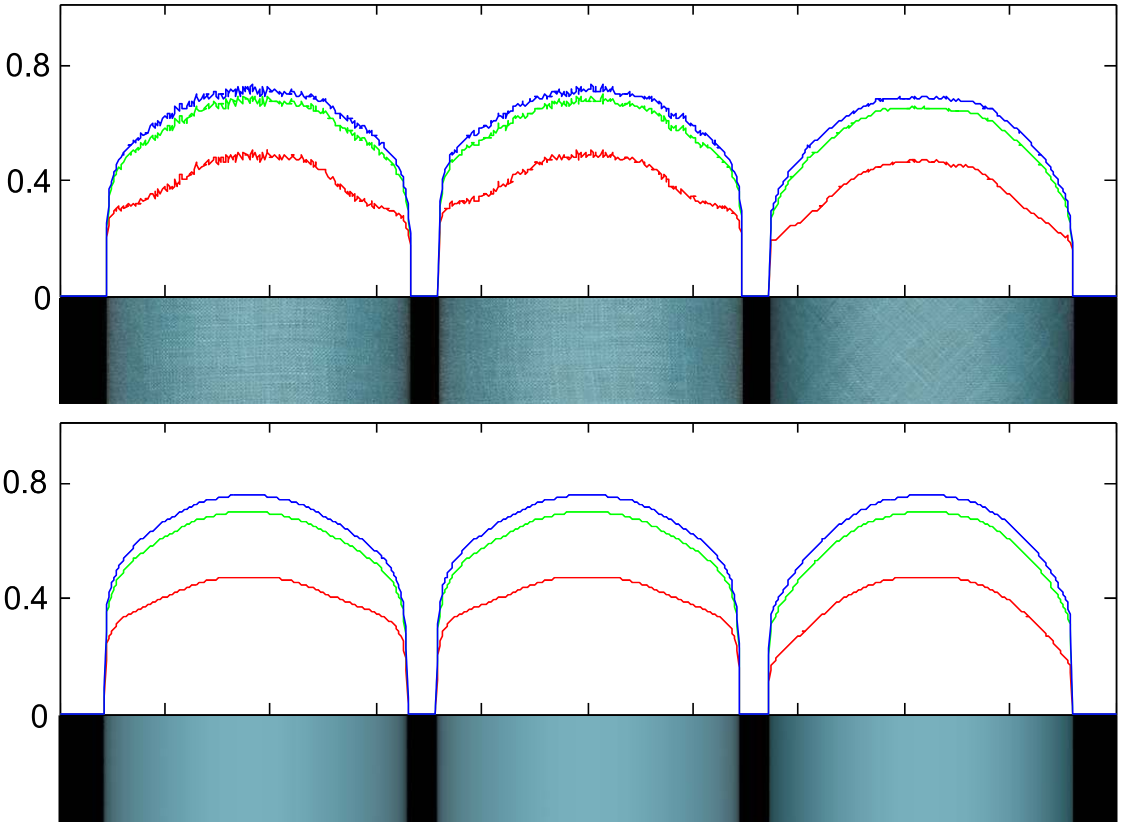


Figure 5.25: Photograph and the rendering result for the Linen Plain fabric.

Our rendering result for the Silk Crepe de Chine fabric is shown in Figure 5.26. This fabric presents grazing angle highlights in the vertical mode and shows two off specular highlights in the horizontal mode. The two off specular peaks are due to the tangent distribution of the flat threads (refer to Appendix E). These behaviors can be seen in the BRDF measurements of this fabric as well (see Figure 5.13). However, it is important to note that the these plots are essentially different; in the BRDF measurements, the ω_i and surface normal n are fixed and the ω_r is changing, while in these graphs the ω_i and ω_r are fixed and n is changing.

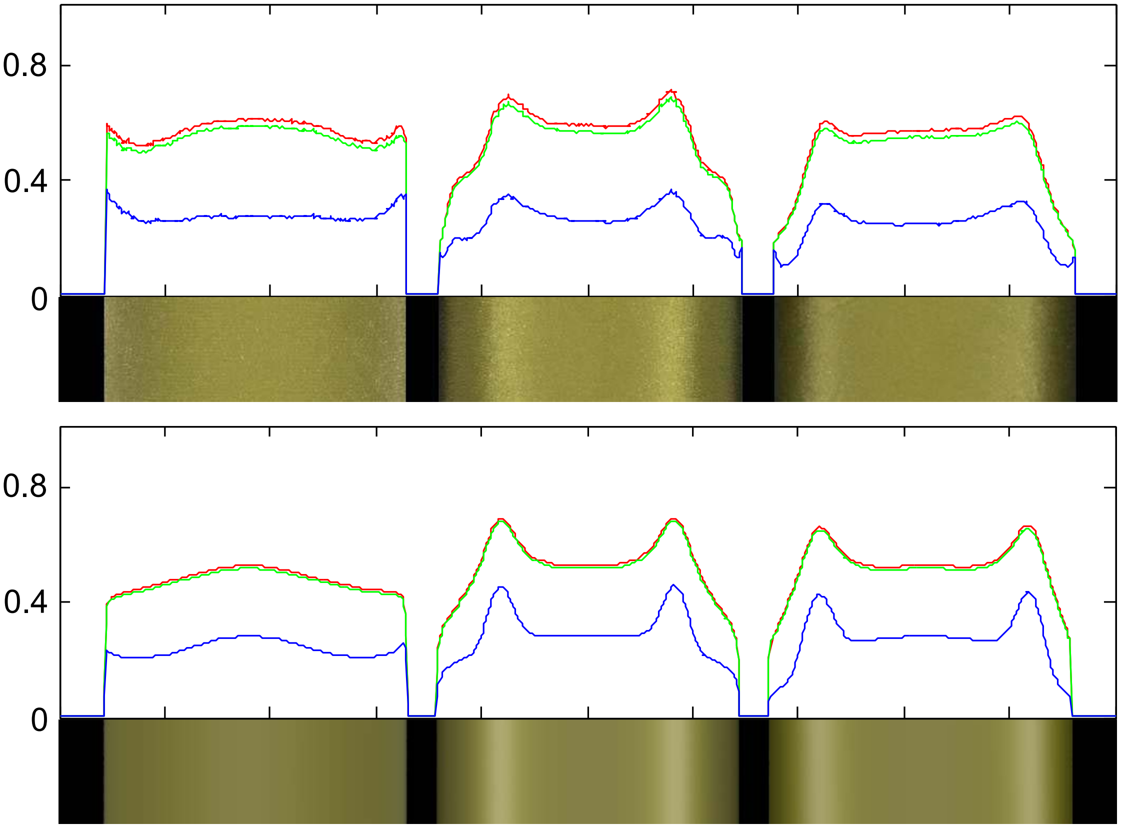


Figure 5.26: Photograph and the rendering result for the Silk Crepe de Chine fabric.

The photograph and the renderings for the front side of Polyester Satin Charmeuse fabric are shown in Figure 5.27. The back side of this fabric has a different appearance due to the asymmetry of this weaving pattern. Our result for the back side of this fabric is presented in Figure 5.28. On the front side, this fabric has a flat appearance in the vertical mode and presents three sharp highlights in the horizontal mode. On the back side, we can see four highlights in the horizontal mode. All of these highlights are due to the tangent distribution of the flat threads in the weaving pattern (refer to Appendix E). Our renderings reproduce the qualitative behavior of this fabric for both sides and in all directions.

Note, how our model is able to capture the variation in the highlights across the

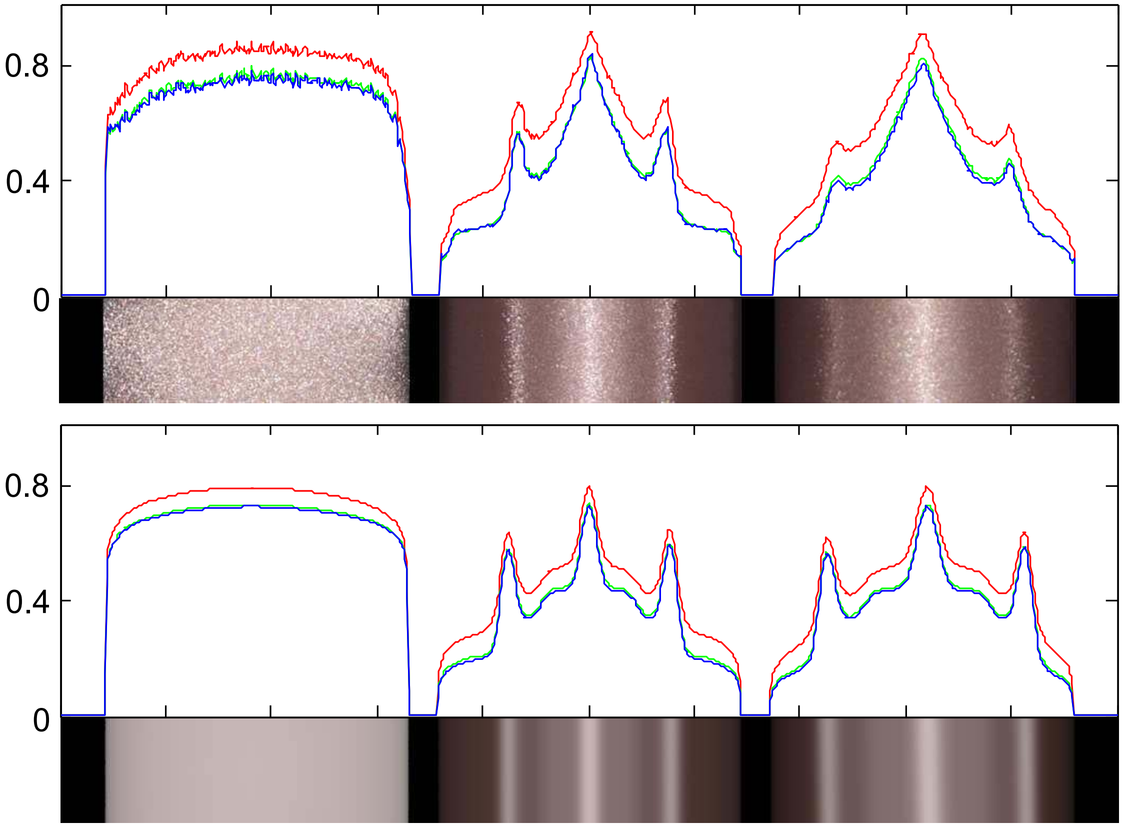


Figure 5.27: Photograph and the rendering result for the front side of Polyester Satin Charmeuse fabric.

different fabrics including the grazing angle highlight seen on the silk, and the split highlight seen on the backside of the Satin. The back side of this fabric has a different appearance due to an asymmetry of the weaving pattern. It can be seen from the first two columns that all four cloth samples show anisotropic behavior (even slightly visible for the linen), and our model is able to not only capture this, but also accurately predict the appearance when the cloth sample is rotated 45 degrees.

Figure 5.29 show a comparison of our BRDF with the measured BRDF of the fabric (Figure 5.29 top) for the front side of the fabric and along the direction of flat threads. Note, how the model is able to capture the variation in the location of the highlights and the overall shape of the reflected light as the light source moves from normal incidence to 30 and 60 degrees.

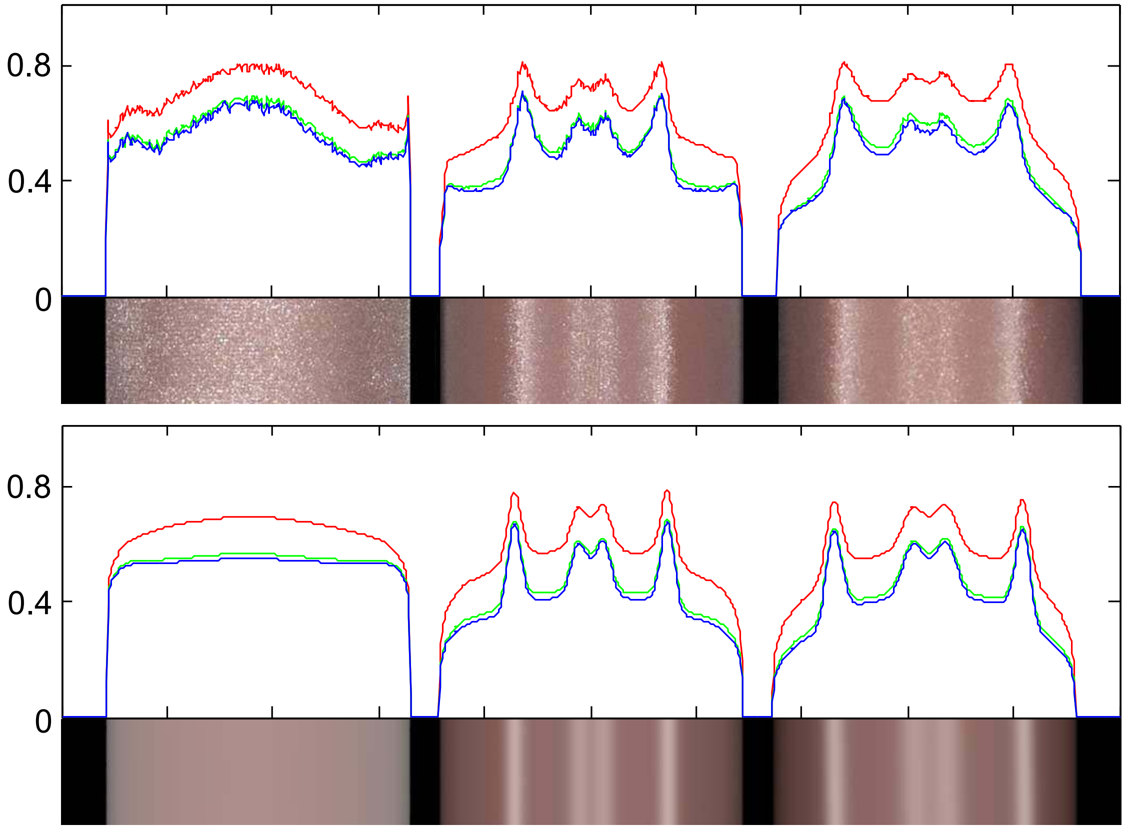


Figure 5.28: Photograph and the rendering result for the back side of Polyester Satin Charmeuse fabric.

Figure 5.30 and 5.31 demonstrate how our model can reproduce the appearance of other fabrics that have been previously studied. We have successfully matched a Silk Shot Fabric (presented in [PK03]) and a Velvet fabric (presented in [Ash01]). The Silk Shot fabric is composed of threads with two different colors (in this case red and green) resulting in a complex anisotropic appearance. Our model we can reproduce this appearance using anisotropic volume scattering by the colored threads rather than the shadowing and masking effect as it was assumed by Pont and Koenderink [PK03]. Our model can easily reproduce the appearance of Velvet (see Figure 5.32) by setting the tangent distribution to be near perpendicular to the surface of the fabric.

Figure 5.32 shows the set of studied fabrics. Table 5.2 summarizes the parameters of our analytic thread BSDF and Appendix E describes the tangent distributions

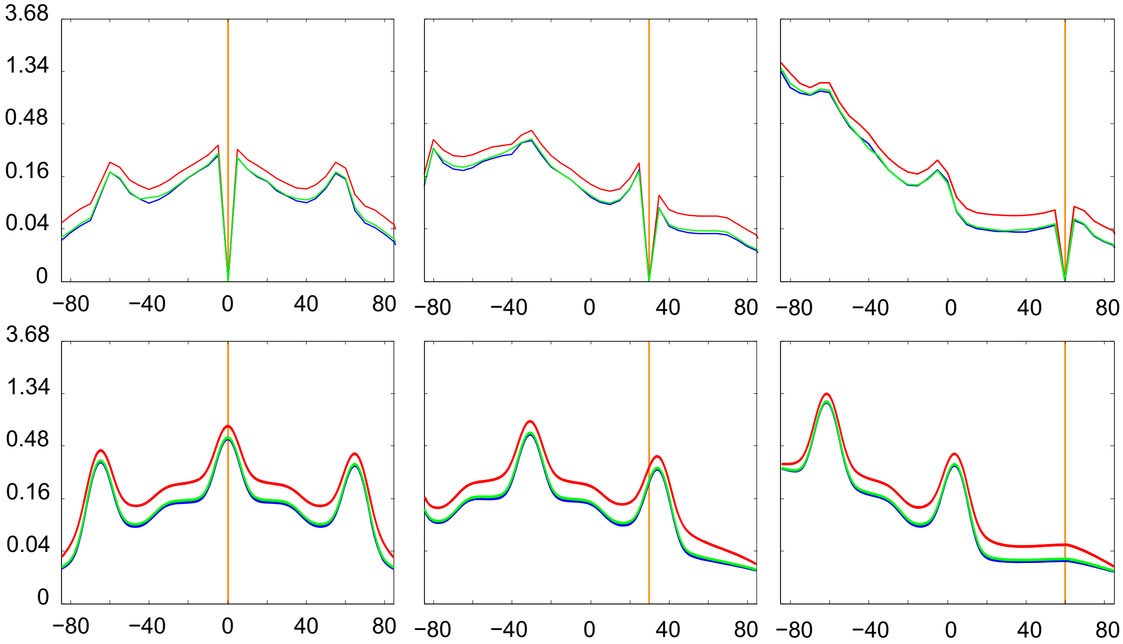


Figure 5.29: Matching a BRDF measurement of the Polyester Satin Charmeuse fabric with our model. (top) normal plain BRDF measurement of the front side of Polyester Satin Charmeuse fabric along the direction of flat threads compared to (bottom) the result of our appearance model.

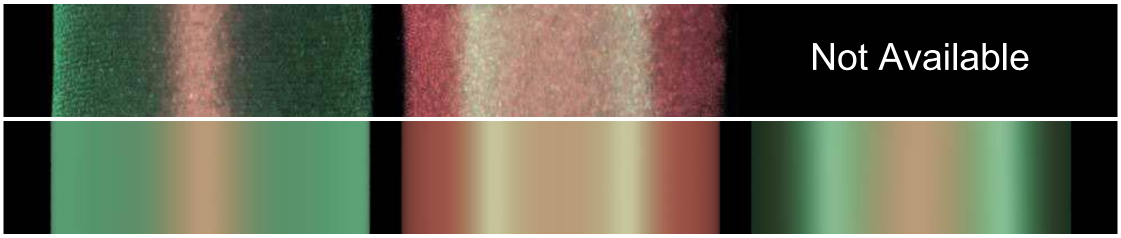


Figure 5.30: Comparison between photographs (top row) and rendered samples (bottom row) for Silk Shot Fabric (from [PK03]). Only the vertical and horizontal mode were available and our model is predicts the appearance for the diagonal mode.

used for our rendering results. For the Velvet fabric, we are using a texture map to specify the groom direction. We have also rendered two imaginary cloth fabrics: one is a weaving of silk and polyester threads using a Shantung weaving pattern and the other is using an imaginary fabric with asymmetric specular peaks. Note

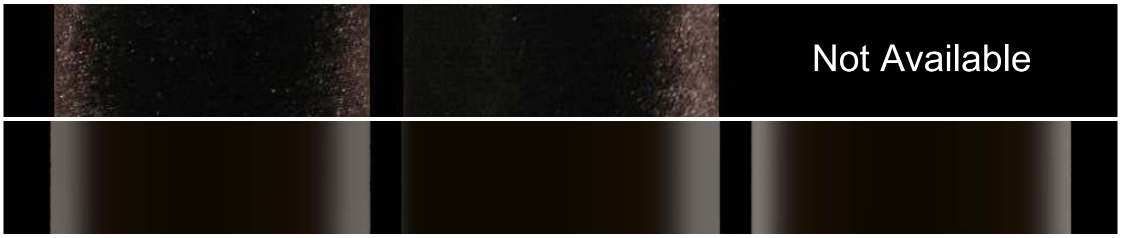


Figure 5.31: Comparison between photographs (top row) and rendered samples (bottom row) for Velvet Fabric (from [PK03]). Only the vertical and horizontal mode were available and our model is predicts the appearance for the diagonal mode.

that our renders are meant to purely demonstrate a surface BRDF. Cloth is a material, with strong textural cues giving a feel to the material. Our model is able to specify this texture parameter at the thread level as demonstrated in the velvet image, which is demonstrating a specular reflection in the shape specified by a texture.

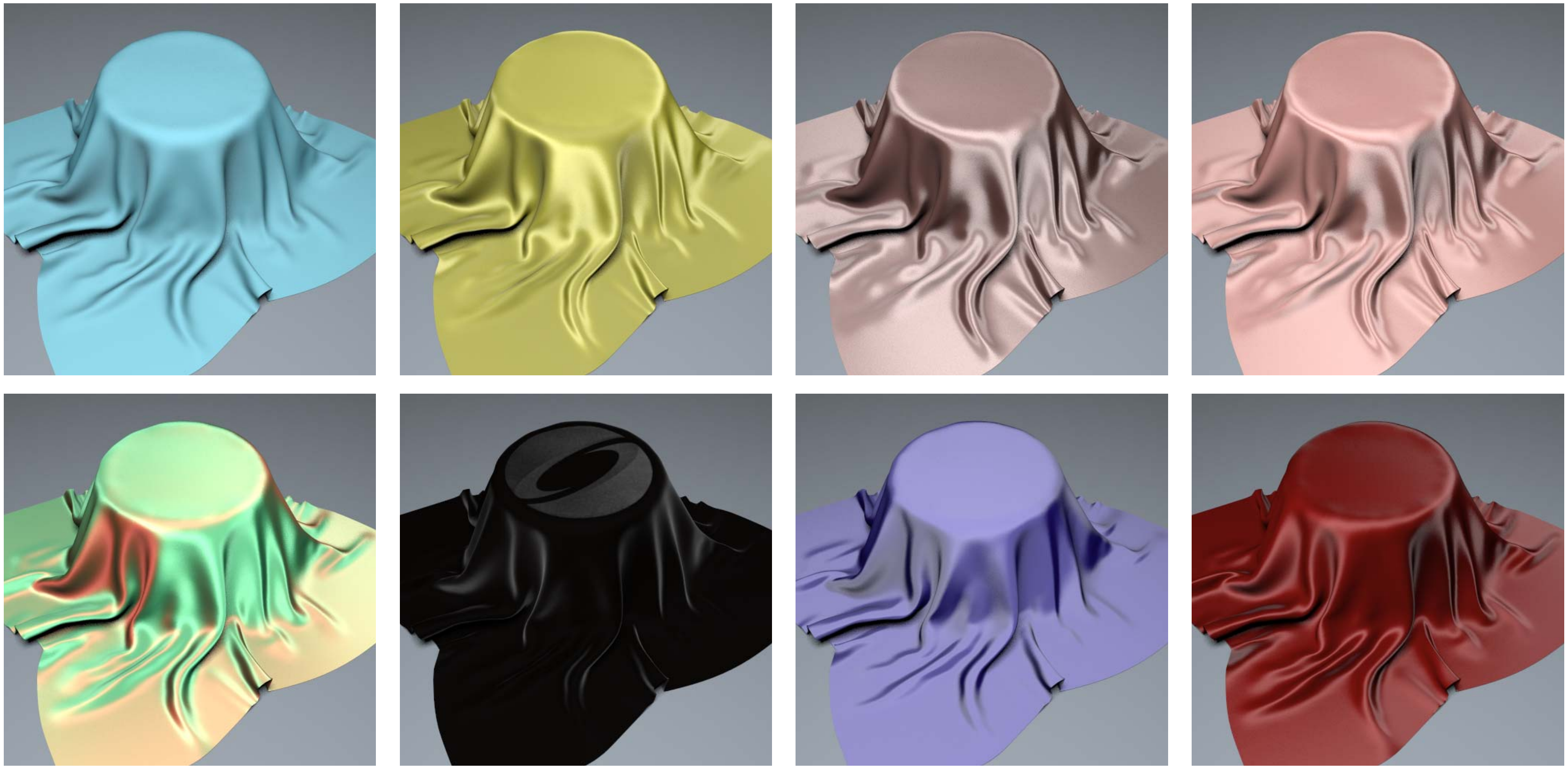


Figure 5.32: Our rendered results for different fabrics. From top to bottom, left to right: Linen Plain, Silk Crepe de Chine, front side of Polyester Satin Charmeuse, back side of the Polyester Satin Charmeuse, Silk Shot Fabric, Velvet, an imaginary fabric made out of silk and polyester threads with a Shantung weaving pattern, and an imaginary fabric with asymmetric specular peaks.

Figure 5.33 shows the effect of shadowing and masking as well as the reweighting process. Note the bright edges in the vertical mode which are the results of the contribution of all vertical flat fibers at grazing angles. The masking term corrects this effect by reducing the intensity of masked threads in grazing angles. The reweighting process intensifies the amount of contribution of the threads based on their projected length on to the image plane (or light plane). For example in the vertical mode, the flat threads will occupy more area of the smallest patch than the twisted threads at grazing angles. Therefore they will contribute more in those cases.

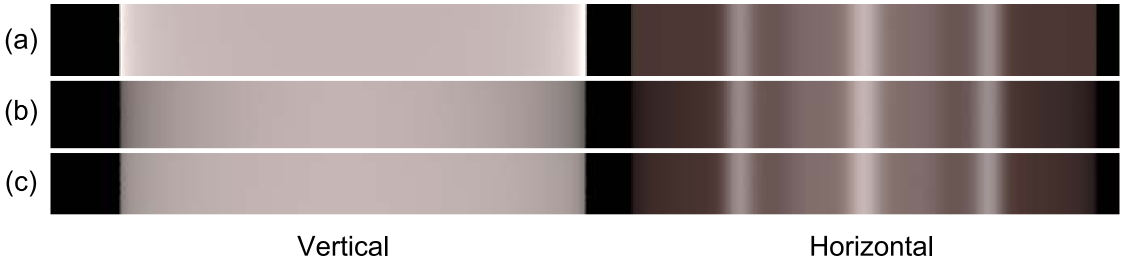


Figure 5.33: The effect of shadowing and masking and the reweighting process on the final results: (a) the result for the shading model, (b) the effect of shadowing/masking term, and (c) the final results after applying shadowing/masking and reweighting.

The images in Figure 5.32 have been rendered in 512 × 512 resolution with 144 samples per pixel in an unoptimized CPU ray tracer. The renderings took 51 minutes on average on an 2.83 GHz Intel Core 2 CPU. We have implemented the same model in a GPU shader. The images in Figure 5.34 were rendered in 100 ms. on a laptop with an Intel I5 M480 processor and a mobile NVIDIA GT420 GPU.

5.8 Limitations

As shown in the results, our model is able to reproduce the complex behavior of a variety of fabrics. Our appearance model is based on our analytical thread BSDF model for different types of thread and the tangent distribution of threads in a

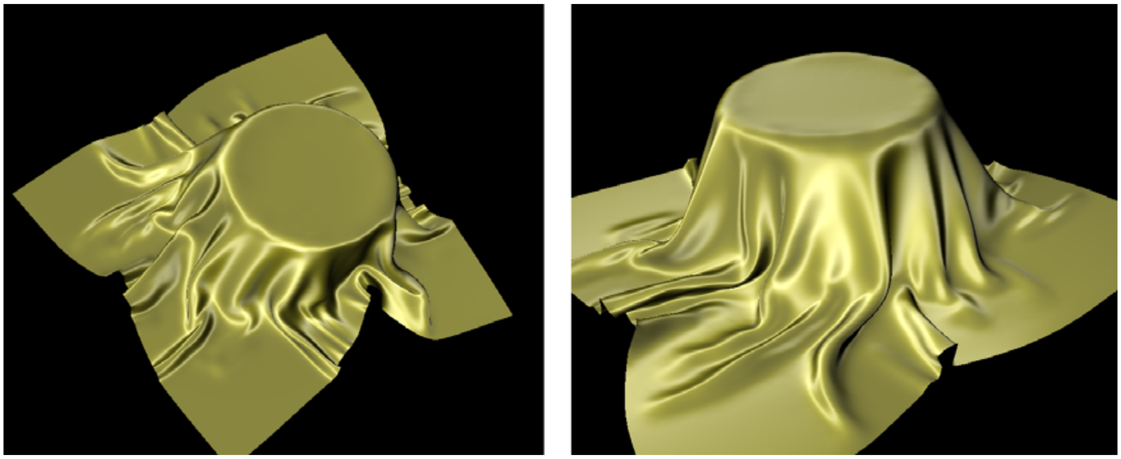


Figure 5.34: Screenshots from the GPU implementation of our model. Each image was rendered in roughly 100 ms on a mobile GT420 GPU.

weaving pattern. A limitation of our model in its current form is that it cannot accurately produce close-up renderings. It does not reproduce the appearance of single threads in a patch, or the reflectance variation across each thread, assuming that the smallest patch of the fabric is smaller than a pixel in the image plane. This limitation can be somewhat worked around with a texture, but a texture will fail for extreme close-ups, where it will be necessary to model actual geometry such as [ZJMB11]. Additionally, our shadowing and masking term does not handle masking between threads with orthogonal directions. This causes an underestimation of masking at extreme grazing angles. Finally, our model ignores the effect of multiple scattering between different threads.

5.9 Conclusion and Future Work

We have presented a practical appearance model for cloth fabrics. Our model is simple and efficient and can be used to reproduce the complex anisotropic appearance of cloth. We present both measurements and a novel scattering model for threads. We use this model to render cloth fabrics. Our cloth BRDF is based on the tangent distribution of interwoven threads, and it includes shadowing and masking terms that are important for grazing angle viewing and lighting. Our model has intuitive inputs and can run in real-time on a GPU.

One avenue for future research is investigating the shadowing and masking between threads with different directions. In addition, we are interested in testing automated fitting processes to estimate the parameters of our model based on photographs of a cloth fabric wrapped around a cylinder in different directions. Furthermore, we aim to investigate the transmission term and approximate the multiple scattering of light between different threads. Lastly, it would be interesting to investigate different ways of importance sampling our fabric BRDF.

5.10 Acknowledgments

We would like to thank Emily Caporello and Mcgill Hall Dept. of Neuroscience for permitting the use of their microscope facility, Hao Li and Carlos Dominguez for providing table cloth models, Sylvia Pont for the use of the Silk Shot Fabric photograph, Toshiya Hachisuka for constructive discussions.

We would also like to thank Hani Goodarzi, Viktoriya Karshenboyem, Krystle de Mesa, and Bridgette Wiley for proofreading this chapter.

The material in this chapter is, in part, a reproduction of the material submitted for publication in Iman Sadeghi, Oleg Bisker, Joachim De Deken, and Henrik Wann Jensen. "A Practical Microcylinder Appearance Model for Cloth Rendering." Submitted to *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2011)*, 2011. The dissertation author was the primary investigator and author of this paper.

6

Conclusions and Future Work

IN this dissertation we focused on three types of specular microstructures: microspheres, microcylinders, and interwoven microcylinders. These microstructures are responsible for the appearance of rainbows, human hair, and cloth fabrics respectively. We have presented three novel appearance models for each of these subjects by examining the light scattering behavior from their corresponding microstructures. Our appearance models have robust physical basis and emphasis on the controllability of the final shading model. Our shading models facilitate the reproduction of a wider range of desired appearances for rendering rainbows, hair fibers, and cloth fabrics.

6.1 Contributions

Our first contribution is a novel appearance model for rendering rainbows. We have presented a novel ray tracing approach for simulating light interactions by particles with arbitrary shapes. We have validated our framework against Lorenz-Mie theory for the case of spherical water drops, and shown that our model can predict the light scattering behavior of non-spherical large water drops. By considering the physically based shape of water drops we have presented a simulation of twinned rainbows for the first time.

In context of hair rendering, we have presented a novel hair appearance model that addresses the problem of art-directability of physically-based shading models. In particular we have defined the basic requirements for artist friendly systems and shown that physically based models fail to satisfy those requirements. Accordingly, we have introduced a novel approach for designing physically inspired but art-directable appearance models based on existing physically based shading models.

Finally, we have also presented a practical appearance model for cloth fabrics. We have introduced a novel analytical BRDF model for cloth fabrics. Using this model and statistical tangent distribution of cloth threads inside a weaving pattern we can reproduce the appearance of wide range of cloth fabrics. We also introduce a novel shadowing and masking term for cloth fabrics which is important in grazing angle viewing and lighting.

6.2 Future Research Directions

Each of our appearance models suggest different directions for future work and can be improved in different aspects.

As mentioned in Section 3.5 the process of matching the appearance of rainbows is not automatic. The parameters generally need to be adjusted by trial and error to get a good match with the original image. A potential extension to our research would aim to use computer vision techniques to automate this process. Furthermore, it would be interesting to explore other approaches for estimating focal lines and diffraction. Also, further development on our phase function simulator could lead to new and generalized global illumination algorithms, taking into account phenomena such as interference or diffraction. Finally, we believe that our algorithm could be adapted to the GPU, greatly speeding up the phase function simulation.

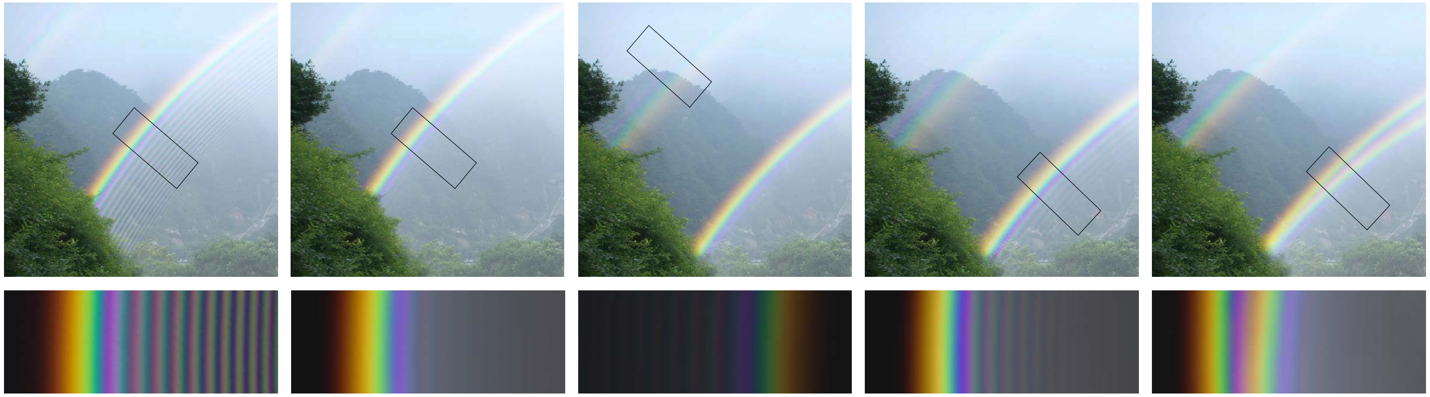
Our approach for creating an art-directable hair shading model is not based on any specific property of hair fibers. It would be interesting to investigate the applicability of our approach to materials other than hair. We speculate that our approach is applicable to a much broader range of materials in appearance modeling. Another avenue of work is to investigate the interplay of the underlying groom and the hair shading model.

Finally, our cloth appearance model opens new avenues for future research directions. One avenue for future research is investigating the shadowing and masking between threads with different directions. In addition, it would interesting to use computer vision techniques to estimate the parameters of our model based on photographs of a cloth fabric wrapped around a cylinder in different directions. Furthermore, we aim to investigate the transmission term and approximate the multiple scattering of light between different threads. Lastly, it would be interesting to automate the derivation of the tangent curves from the weaving patterns.

A

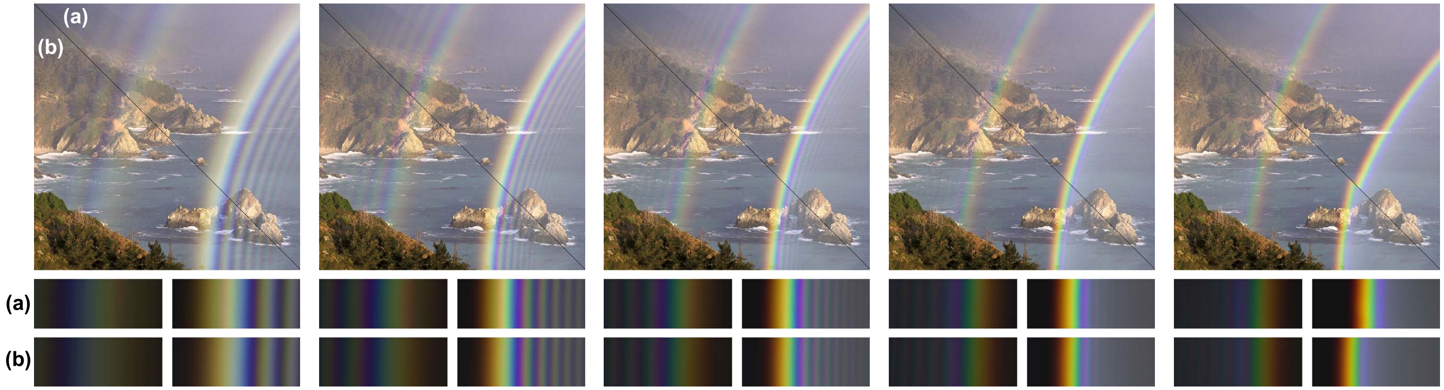
Rainbow Rendering Results Parameters

THE parameters required for producing the results of Chapter 3 are listed in this appendix. These parameters include the water drop size, field of view (FOV), camera lense type, sun's inclination (only relevant for non-spherical water drops), and the illumination model. For the superimposed images we also mention the intensity of the rendering and the background color which have been added to images.



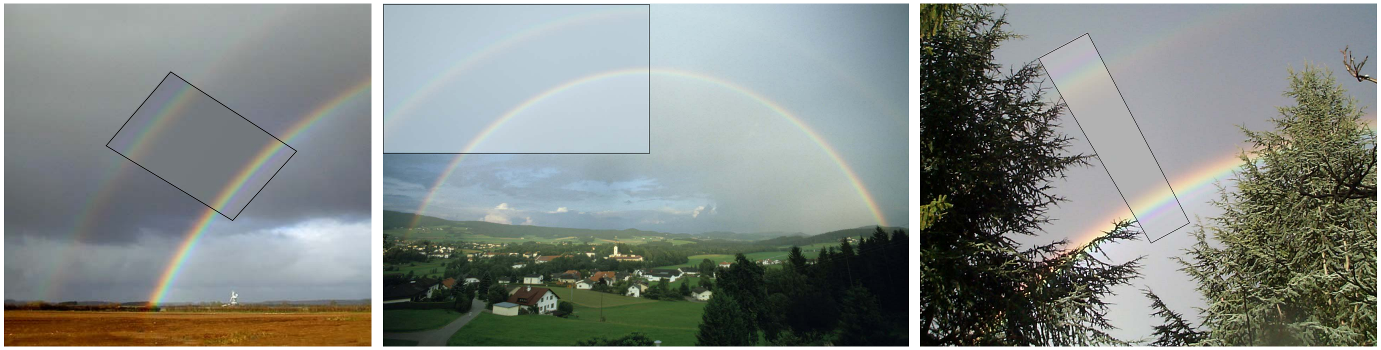
	(a)	(b)	(c)	(d)	(e)
Water Drop Size	0.4 mm	0.4 mm	0.4 mm	0.3 mm	0.40 mm & 0.45 mm
FOV	25^{o}	25^{o}	25^{o}	25^{o}	25^{o}
Lens Type	Rectilinear	Rectilinear	Rectilinear	Rectilinear	Rectilinear
Sun's Inclination	N/A	N/A	N/A	N/A	0^{o}
Sun's Angular View	0^{o}	0.5^{o}	0.5^{o}	0.5^{o}	0.5^{o}
Illumination	D65	D65	D65	D65	D65

Figure A.1: Parameters used for rendering of Figure 3.1



Water Drop Size	0.1 mm	0.2 mm	0.3 mm	0.4 mm	0.5 mm
FOV	30^o	30^o	30^o	30^o	30^o
Lens Type	Rectilinear	Rectilinear	Rectilinear	Rectilinear	Rectilinear
Sun's Inclination	N/A	N/A	N/A	N/A	0^o
Sun's Angular View	0.5^o	0.5^o	0.5^o	0.5^o	0.5^o
Illumination	D65	D65	D65	D65	D65

Figure A.2: Parameters used for rendering of Figure 3.14



Water Drop Size	0.4 mm	0.4 mm	0.3 mm
FOV	20^{o}	100^{o}	30^{o}
Lens Type	Rectilinear	Fisheye	Rectilinear
Sun's Inclination	N/A	N/A	N/A
Sun's Angular View	0.5^{o}	0.5^{o}	0.5^{o}
Background Color	(107,114,118)	(183,202,212)	(172,172,172)
Intensity	55%	100%	90%
Illumination	D65	D65	D65

Figure A.3: Parameters used for rendering the top row images of Figure 3.15

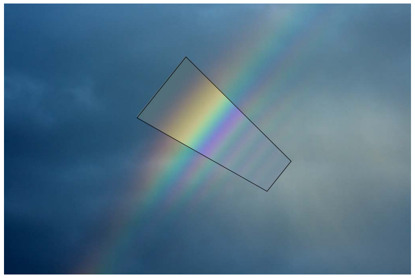
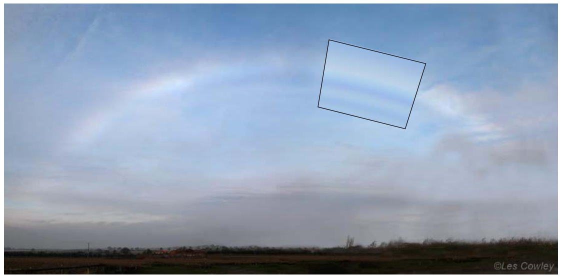
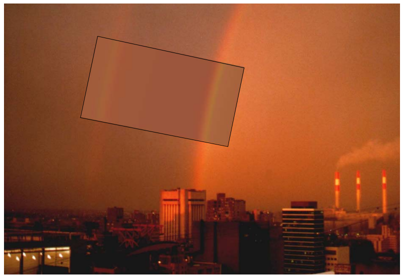
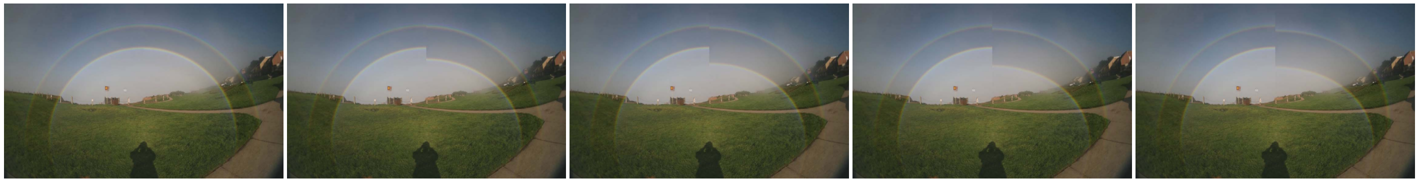
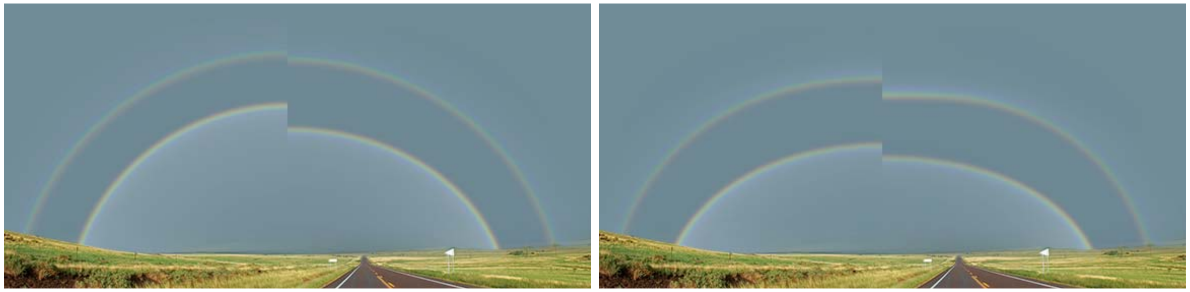
		 	
Water Drop Size	0.3 mm	0.1 mm	0.4 mm
FOV	10^{o}	120^{o}	30^{o}
Lens Type	Rectilinear	Fisheye	Rectilinear
Sun's Inclination	N/A	N/A	N/A
Sun's Angular View	0.5^{o}	0.5^{o}	0.5^{o}
Background Color	(69,99,112)	(141,180,223)	(154,83,58)
Intensity	80%	60%	80%
Illumination	D65	D65	Rayleigh Scattering Applied to D65

Figure A.4: Parameters used for rendering the bottom row images of Figure 3.15



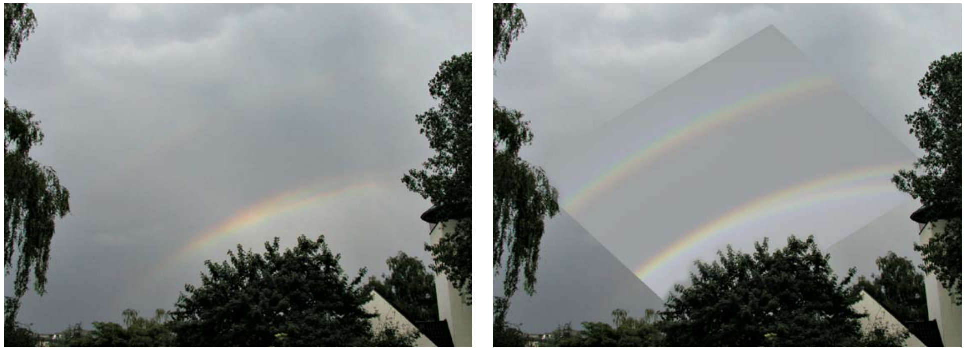
Water Drop Size	0.4 mm	0.5 mm	0.6 mm	0.7 mm	0.8 mm
FOV	120^{o}	120^{o}	120^{o}	120^{o}	120^{o}
Lens Type	Fisheye	Fisheye	Fisheye	Fisheye	Fisheye
Sun's Inclination	N/A	25^{o}	25^{o}	25^{o}	25^{o}
Sun's Angular View	0.5^{o}	0.5^{o}	0.5^{o}	0.5^{o}	0.5^{o}
Illumination	D65	D65	D65	D65	D65

Figure A.5: Parameters used for rendering of Figure 3.16



Water Drop Size	0.4 mm	0.5 mm	0.6 mm	0.7 mm
FOV	120^o	120^o	120^o	120^o
Lens Type	Fisheye	Fisheye	Fisheye	Fisheye
Sun's Inclination	N/A	0^o	0^o	0^o
Sun's Angular View	0.5^o	0.5^o	0.5^o	0.5^o
Illumination	D65	D65	D65	D65

Figure A.6: Parameters used for rendering of Figure 3.17



Water Drop Size	0.4 mm & 0.45 mm
FOV	30^o
Lens Type	Rectilinear
Sun's Inclination	0^o
Sun's Angular View	0.5^o
Background Color	(146,151,153)
Illumination	D65

Figure A.7: Parameters used for rendering of Figure 3.20

B

Artist Friendly Vocabulary for Hair Rendering

HERE we present the list of artist friendly terms for the context of hair rendering which are the result of many brainstorming sessions with our lighting and look-development artists in Walt Disney Animation Studios. We hope that they become a common vocabulary for researchers and artists in the future.

Table B.1: Technical terms and their corresponding artist friendly terms.

Technical Term	**Artist Friendly Term**
Primary Highlight	Specular Component
Secondary Highlight	Subspecular Component
Transmission Highlight	Transmission Component
Color	Color
Intensity	Scale
Longitudinal Width	Roughness
Longitudinal Position	Angular Offset
Azimuthal Width	Spread
Backscattering	Hair Volume Diffuse
Forward Scattering	Hair Volume Transmission

C

RSL Code for the Dual Scattering Model

IN this appendix we present our implementation of the Dual Scattering model in RenderMan Shading Language (RSL). The model has been modified to avoid any ray tracing steps to make it more practical for the production environment.

```
class dual_scattering(
    // Defining the shader parameters
    float enable = 1;
    /*
        type switch
        name {Enable}
        desc {Enables the hair shader.}
    */
    color absorption_coefficients = color (0.2, 0.3, 0.5);
    /*
        name {Absorption Coefficients}
        desc {Absorption coefficients of hair medium for R, G and B channels.}
    */
    float cuticle_angle = 5;
    /*
        name {Cuticle Angle}
        desc {The angle of cuticle scales on hair fiber.}
    */
    ...
)
{
    //Defining the uniform parameters used in the pre-computation step
    float $d_b$, $d_f = 0.7$;

    color $\bar{a}_f[\theta_h]$, $\bar{a}_b[\theta_h]$, $\bar{\alpha}_f[\theta_h]$, $\bar{\alpha}_b[\theta_h]$, $\bar{\beta}_f^2[\theta_h]$, $\bar{\beta}_b^2[\theta_h]$;
    color $\bar{\sigma}_b^2[\theta_h]$, $\bar{A}_b[\theta_h]$, $\bar{\Delta}_b[\theta_h]$, $N_R^G[\theta_h]$, $N_{TT}^G[\theta_h]$, $N_{TRT}^G[\theta_h]$;

    //Defining the varying parameters which have different values for each shading point
    varying float hairs_in_front;
    varying color $\bar{\sigma}_f^2$;
    varying color $T_f$;

    //Defining the auxiliary functions like normalized Gaussian function etc.
    float g(float variance; float x;)
    {
        return exp(-x*x/(2*variance))/sqrt(2*$\pi$*variance);
    }
    ...

    //Pre-computing and tabulating the uniform variables in the Constructor
    public void construct ()
    {
        //Pre-computations should be done in the following order:
        $\bar{a}_f[\theta_h]$, $\bar{a}_b[\theta_h]$ = ...
        $\bar{\alpha}_f[\theta_h]$, $\bar{\alpha}_b[\theta_h]$ = ...
        $\bar{\beta}_f[\theta_h]$, $\bar{\beta}_b[\theta_h]$ = ...
        $\bar{\Delta}_b[\theta_h]$ = ...
        $\bar{\sigma}_b^2[\theta_h]$ = ...
        $\bar{A}_b[\theta_h]$ = ...
        $N_R^G[\theta_h]$, $N_{TT}^G[\theta_h]$, $N_{TRT}^G[\theta_h]$ = ...
    }

    //Main body of the hair surface shader.
    public void surface( output color Ci, Oi;)
    {
```

```
//Compute Hair tangent U, and viewing direction $\omega_r$, $\theta_r$ and $\phi_r$
vector U = - normalize (dPdv);
$\omega_r$ = -normalize(I);
...

// Loop over all the lights in the scene
illuminance (P )
{
    //Compute light direction $\omega_i$, $\theta_i$, $\theta_h$, $\theta_d$, $\phi_i$ and $\phi$
    $\omega_i$ = normalize(L);
    ...

    //Compute the amount of shadow from the deep shadow maps
    float shadowed = 0;
    lightsource (”out_shadow”, shadowed);
    float illuminated = 1 - shadowed;

    //Estimate the number of hairs in front of the shading point
    hairs_in_front = shadowed * hairs_that_cast_full_shadow;

    //Use the number of hairs in front of the shading point to approximate $\bar{\sigma}_f^2$
    $\bar{\sigma}_f^2$ = hairs_in_front * $\bar{\beta}_f^2[\theta_h]$;

    //Use the number of hairs in front of the shading point to approximate $T_f$
    $T_f$ = $d_f$ * pow($\bar{a}_f[\theta_h]$, hairs_in_front);

    //Computing $f_s^{direct}$ and $f_s^{scatter}$
    $f_s^{direct}$ =
        $M_R(\theta_h)$ * $N_R(\theta_d, \phi)$ +
        $M_{TT}(\theta_h)$ * $N_{TT}(\theta_d, \phi)$ +
        $M_{TRT}(\theta_h)$ * $N_{TRT}(\theta_d, \phi)$;
    $f_s^{scatter}$ =
        $M_R^G(\theta_h)$ * $N_R^G(\theta_d, \phi)$ +
        $M_{TT}^G(\theta_h)$ * $N_{TT}^G(\theta_d, \phi)$ +
        $M_{TRT}^G(\theta_h)$ * $N_{TRT}^G(\theta_d, \phi)$;

    //Computing $f_{back}^{direct}$ and $f_back^{scatter}$
    $f_{back}^{direct}$ =
                2 *$\bar{A}_b[\theta_d]$ * g($\theta_h$-$\bar{\Delta}_b[\theta_d]$, $\bar{\sigma}_b^2[\theta_d]$)/($\pi$ * cos$^2$($\theta_d$));
            $f_{back}^{scatter}$ =
                2 *$\bar{A}_b[\theta_d]$ * g($\theta_h$-$\bar{\Delta}_b[\theta_d]$, $\bar{\sigma}_b^2[\theta_d]$+$\bar{\sigma}_f^2[\theta_d]$)/($\pi$ * cos$^2$($\theta_d$));

            // Computing $F^{direct}$
            color $F^{direct}$ = (illuminated) * ($f_s^{direct}$ + $d_b$ * $f_{back}^{direct}$);

            // Computing $F^{scatter}$
            color $F^{scatter}$ = ($T_f$ - illuminated) * $d_f$ * ($f_s^{scatter}$ + $\pi$ * $d_b$ * $f_{back}^{scatter}$);

            // Computing the final result
            Ci += ($F^{direct}$ + $F^{scatter}$) * cos($\theta_i$);
        }
        Oi = Os;
        Ci *= Oi;
    }
}
```

D

Hair Rendering User Study Supplementary Data

We present the complete set of evaluations obtained by each of the artists in the user study in this appendis. In addition we provide the rankings by a mix of expert and non-expert volunteers from Walt Disney Animation Studios. Each evaluator was asked to order the results from best (left) to worst (right) with parentheses indicating ties. The results were initially presented in alphabetical order without any information about which results were obtained with which shading model.

To decode the following evaluations, the following correspondenses apply:

- New shader: A-B-D-E-G-M
- Production shader: F-H-J
- Research shader: C-I-K-L

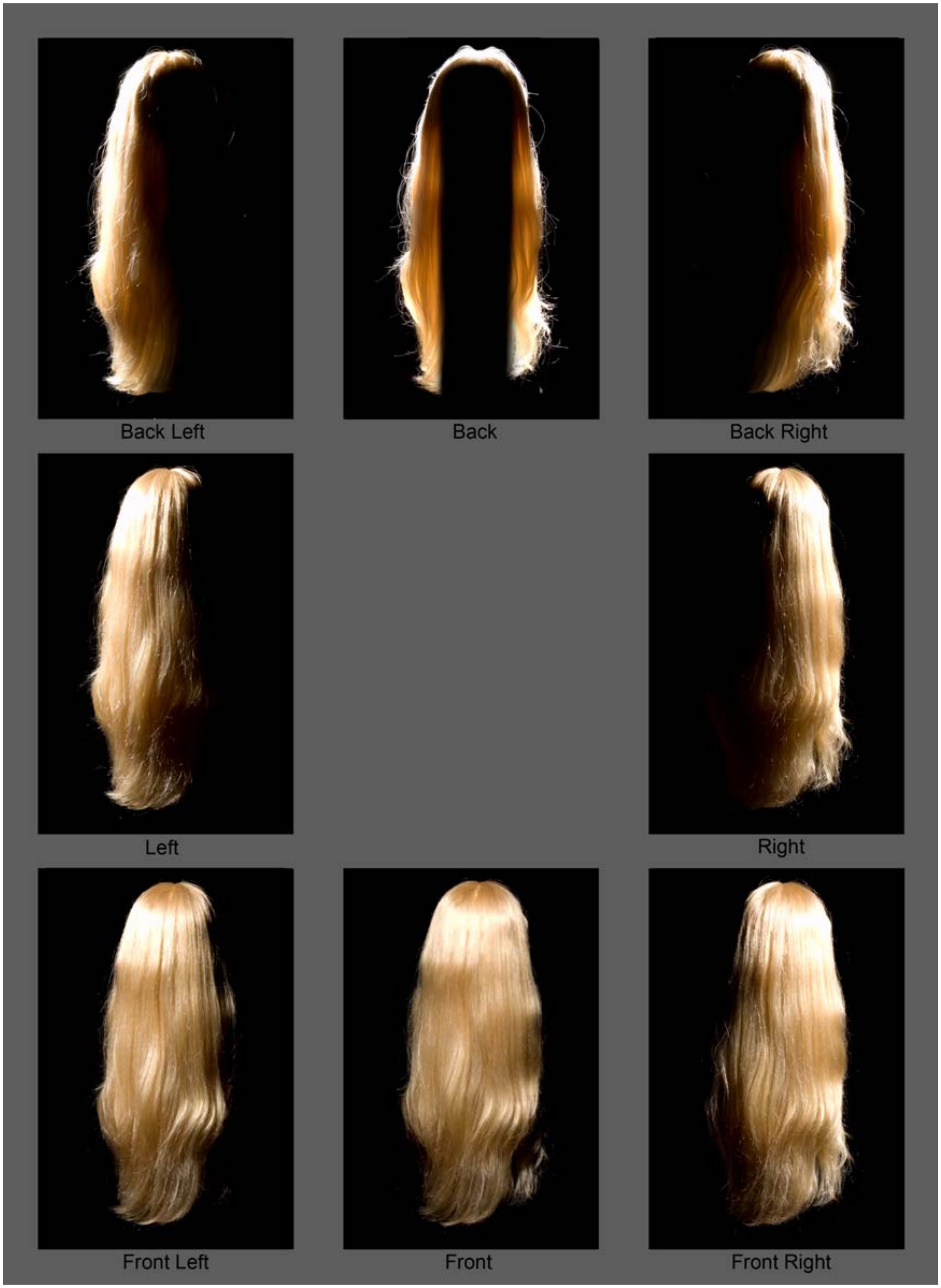


Figure D.1: Photograph references used in the user study.

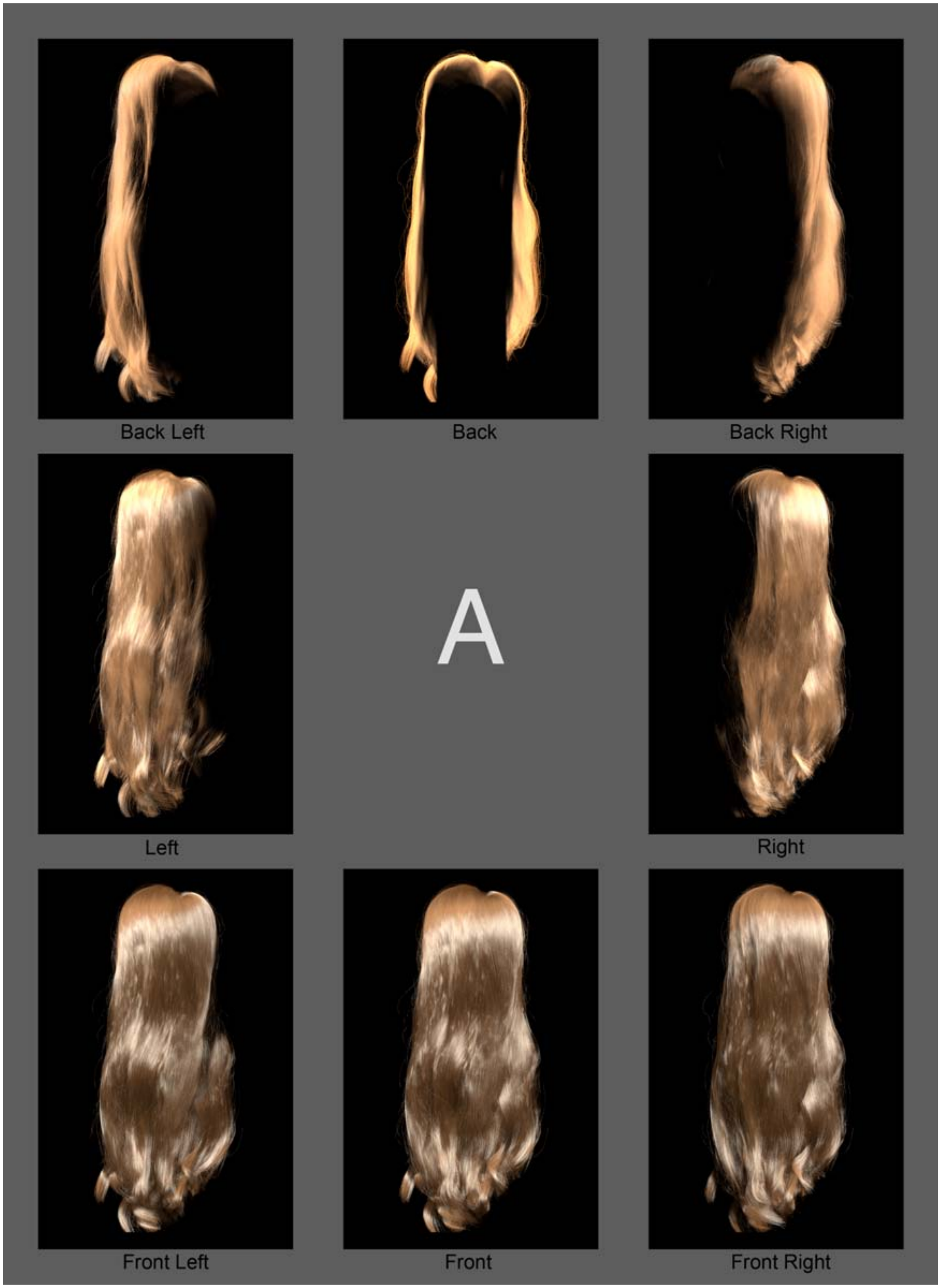


Figure D.2: Rendering results of artist A using the new shader.

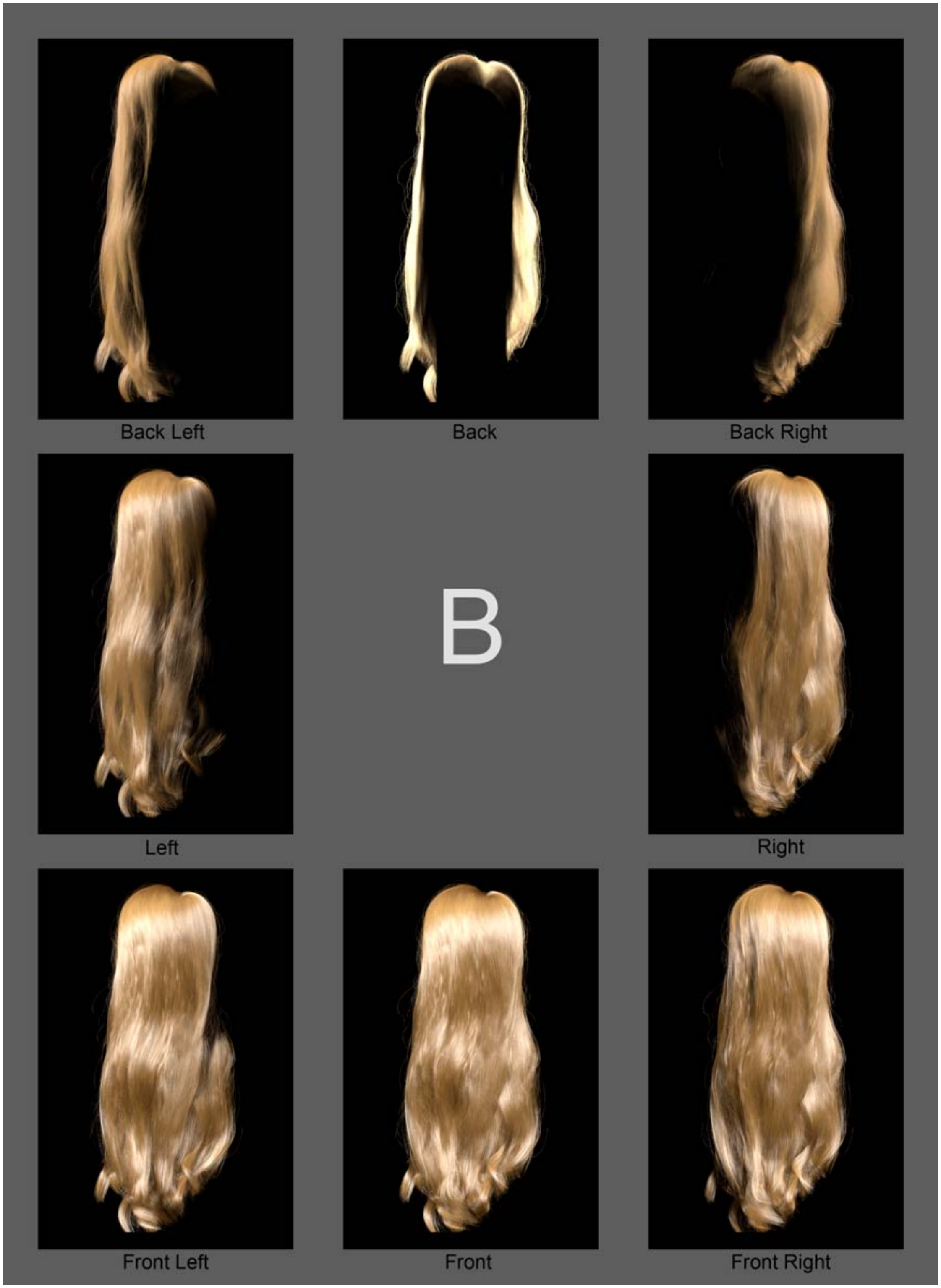


Figure D.3: Rendering results of artist B using the new shader.

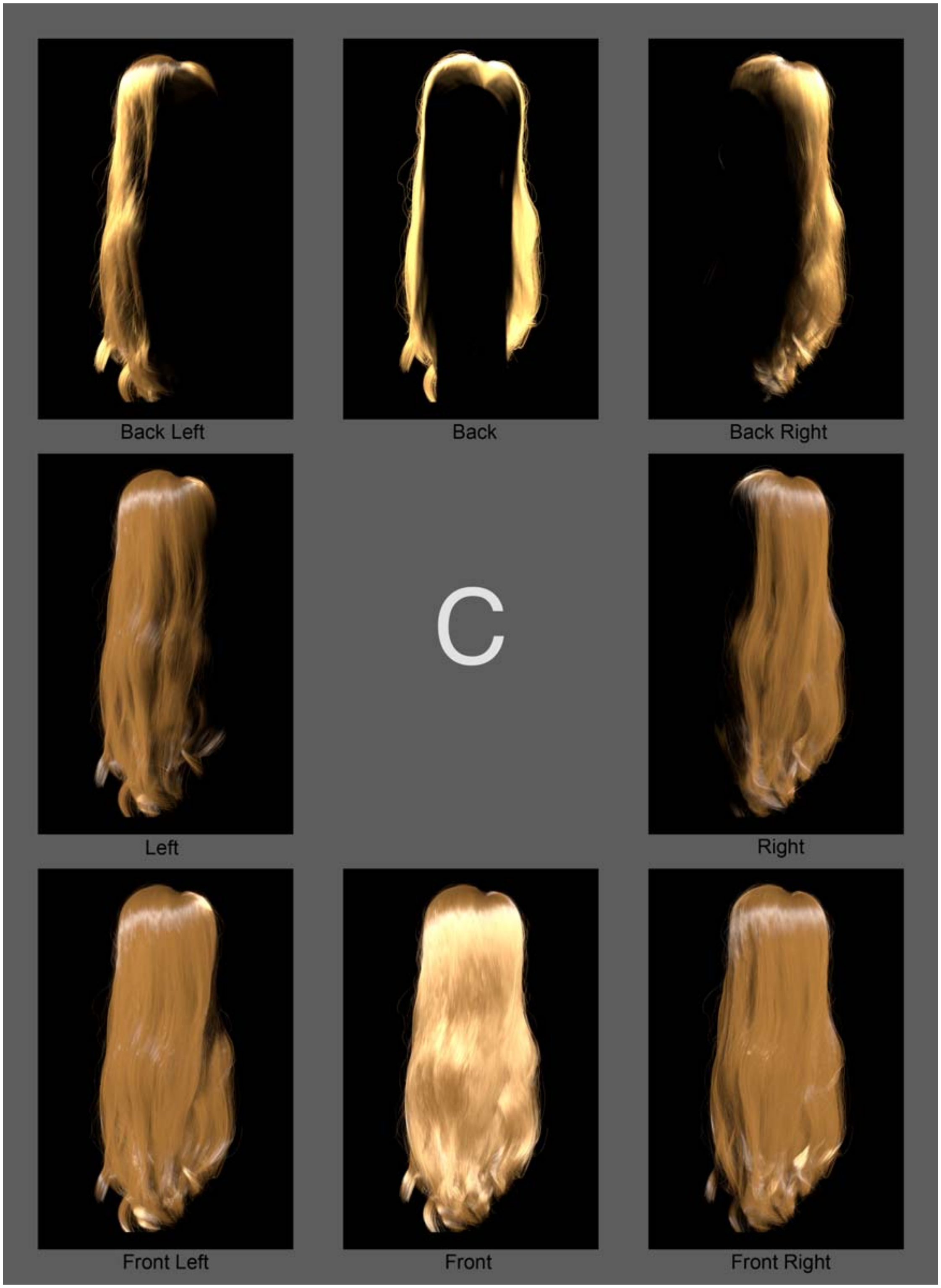


Figure D.4: Rendering results of artist C using the research shader.

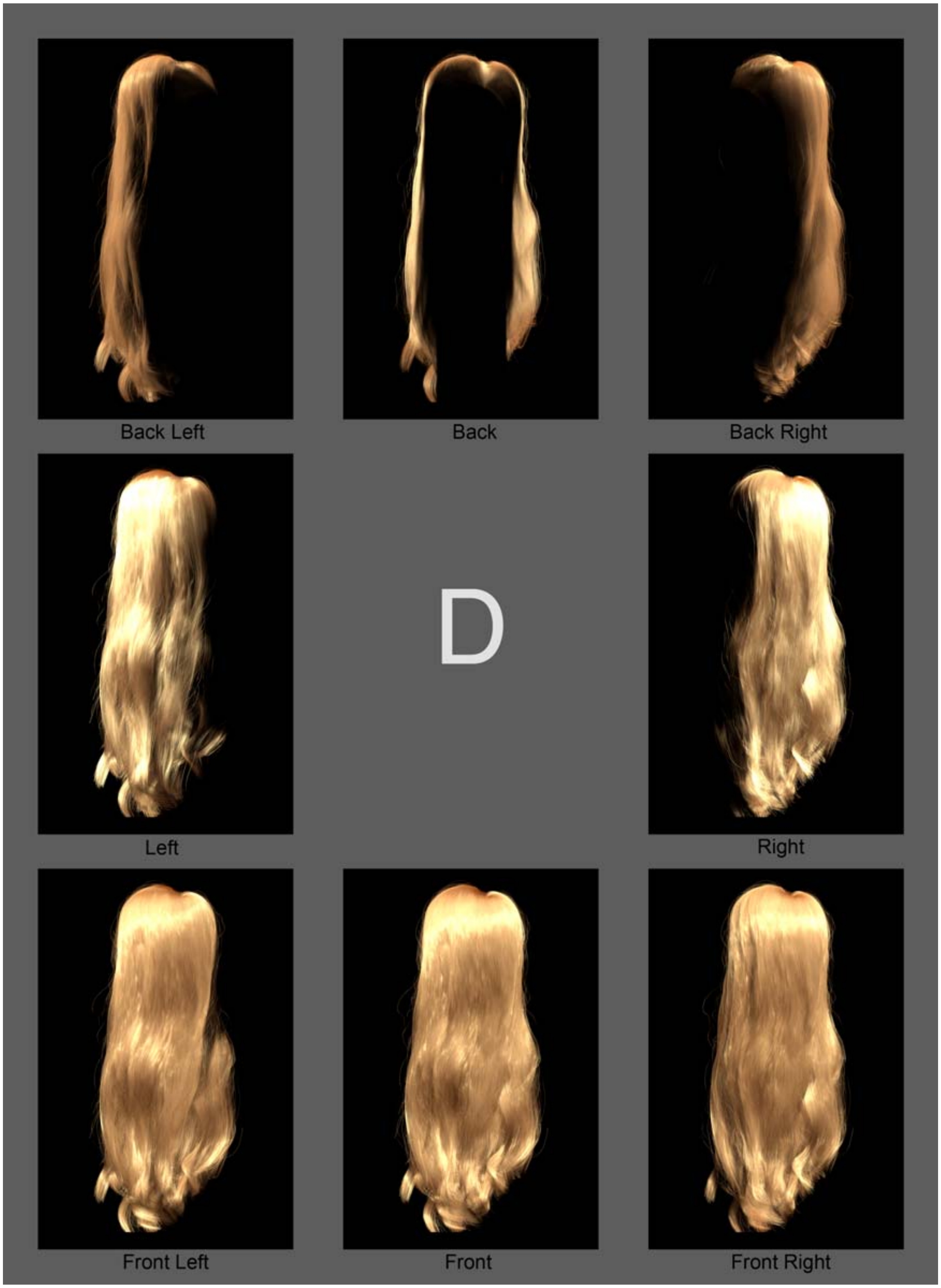


Figure D.5: Rendering results of artist D using the new shader.

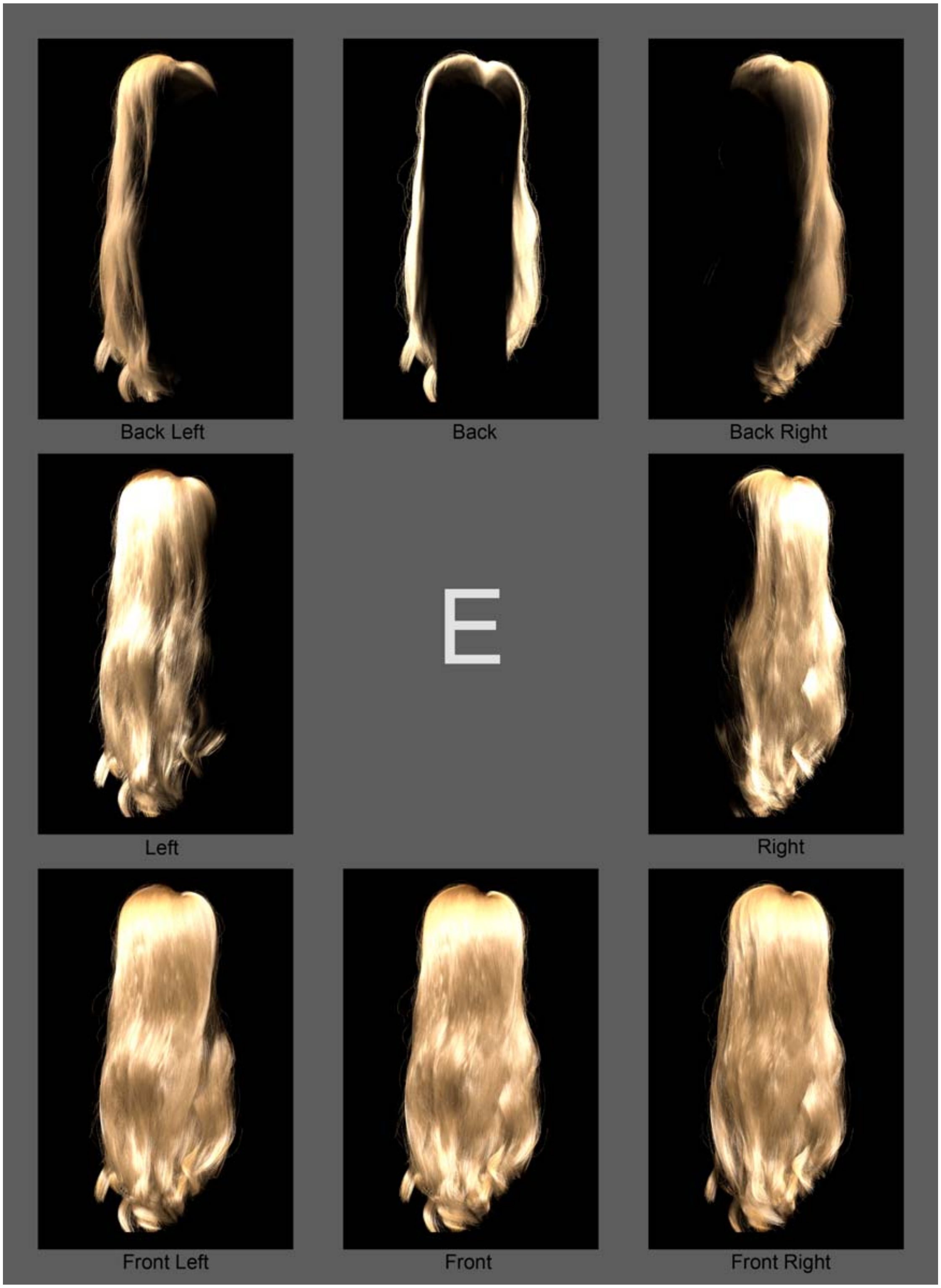


Figure D.6: Rendering results of artist E using the new shader.

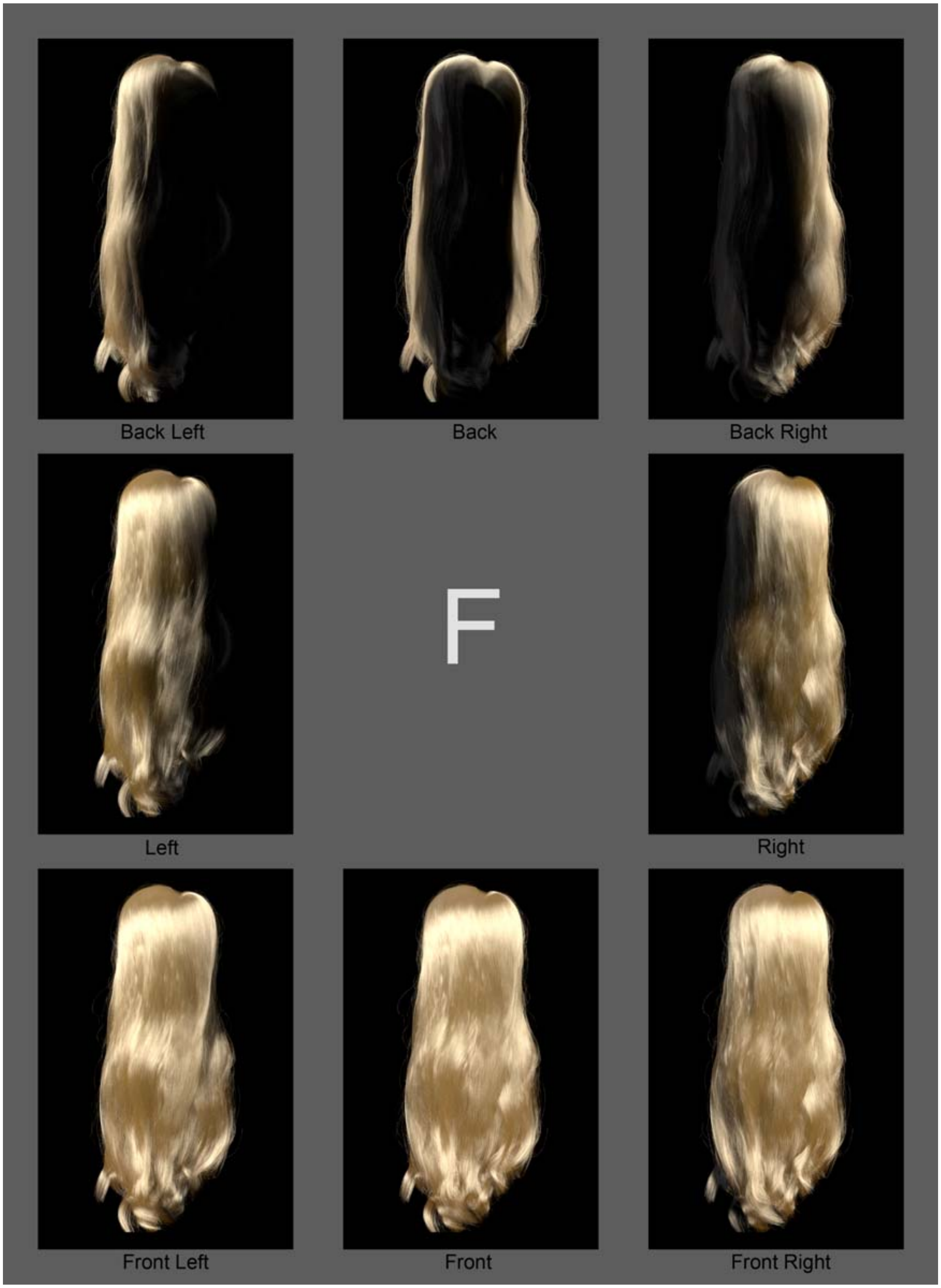


Figure D.7: Rendering results of artist F using the production shader.

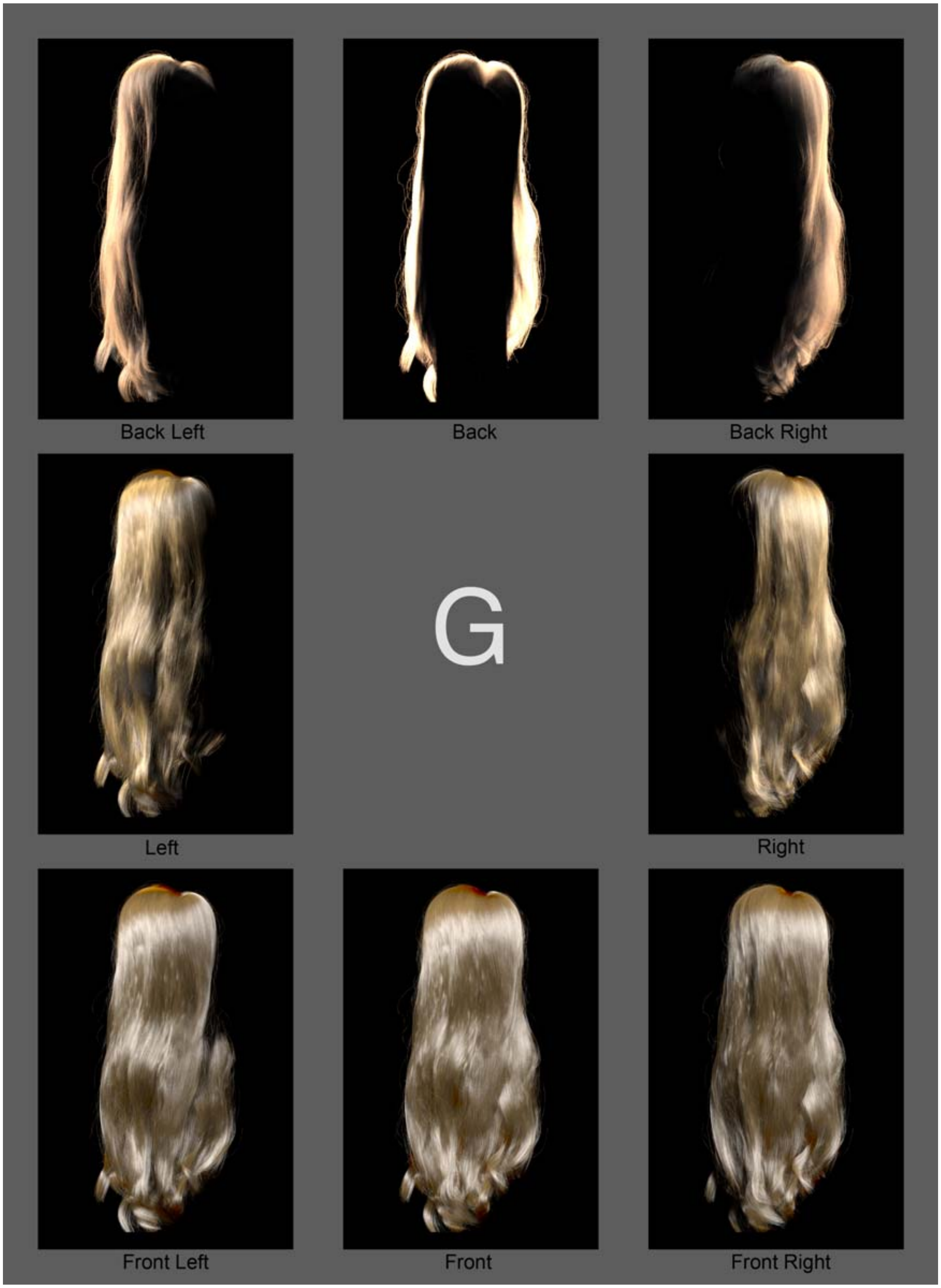


Figure D.8: Rendering results of artist G using the new shader.

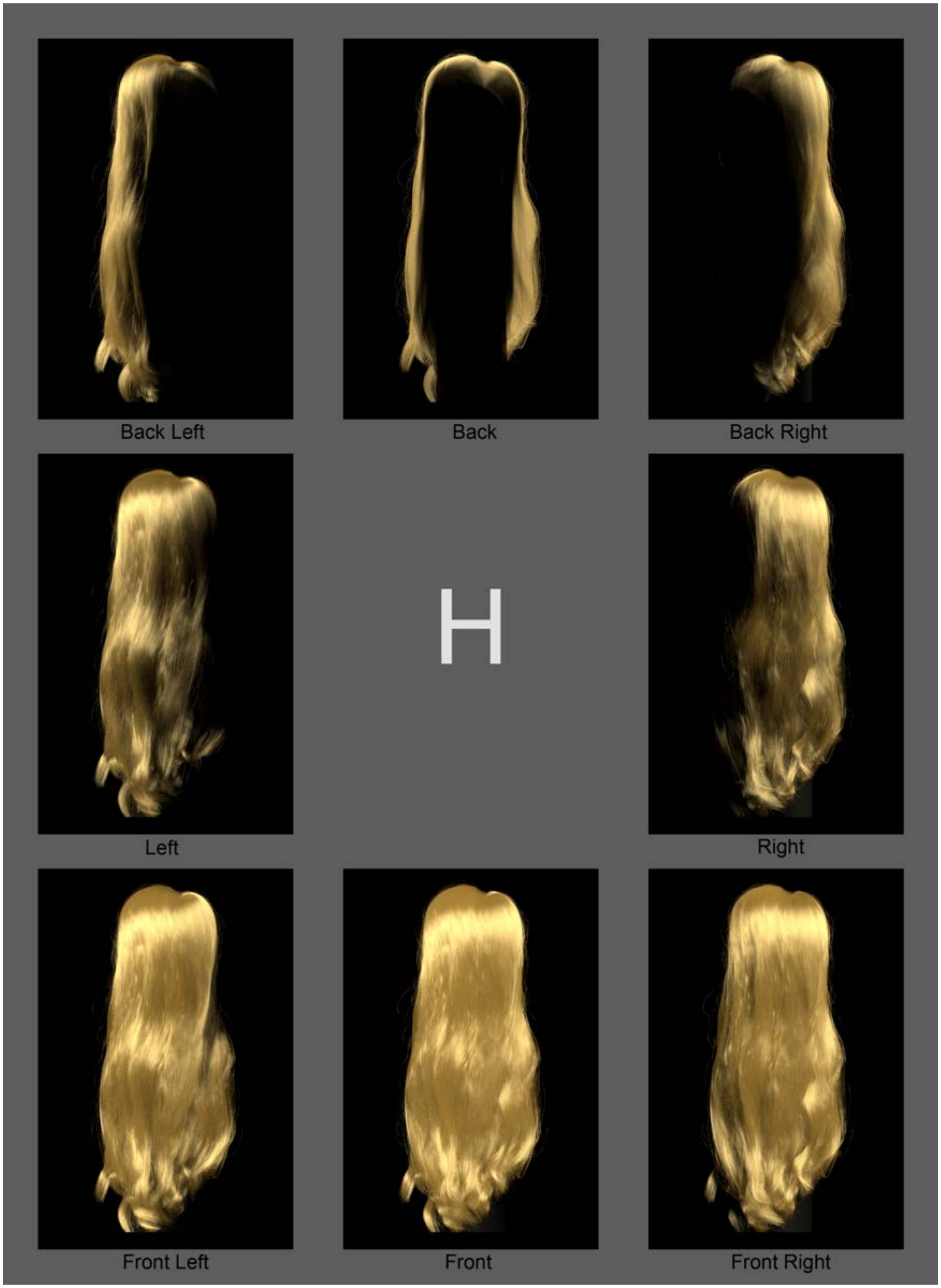


Figure D.9: Rendering results of artist H using the production shader.

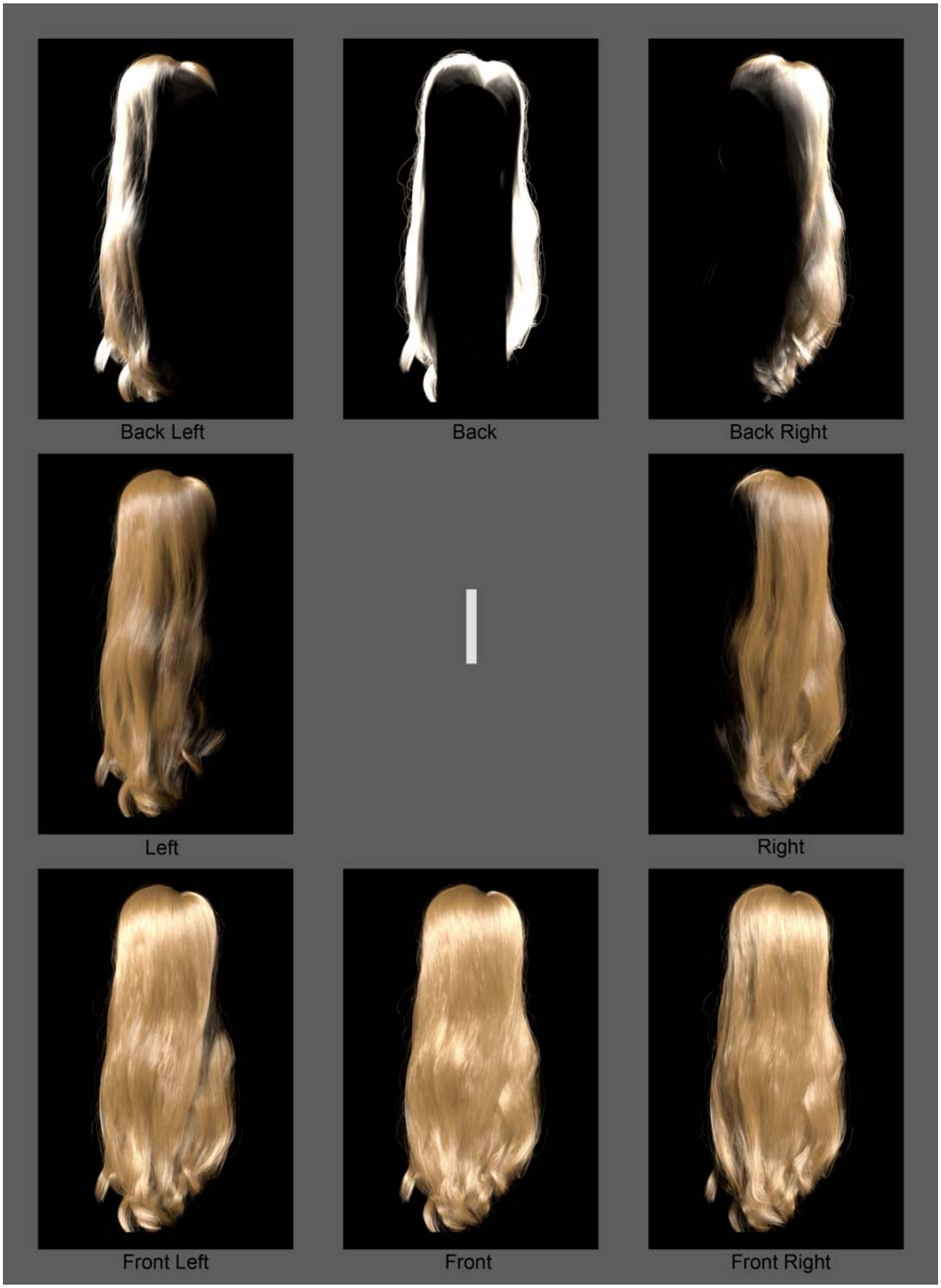


Figure D.10: Rendering results of artist I using the research shader.

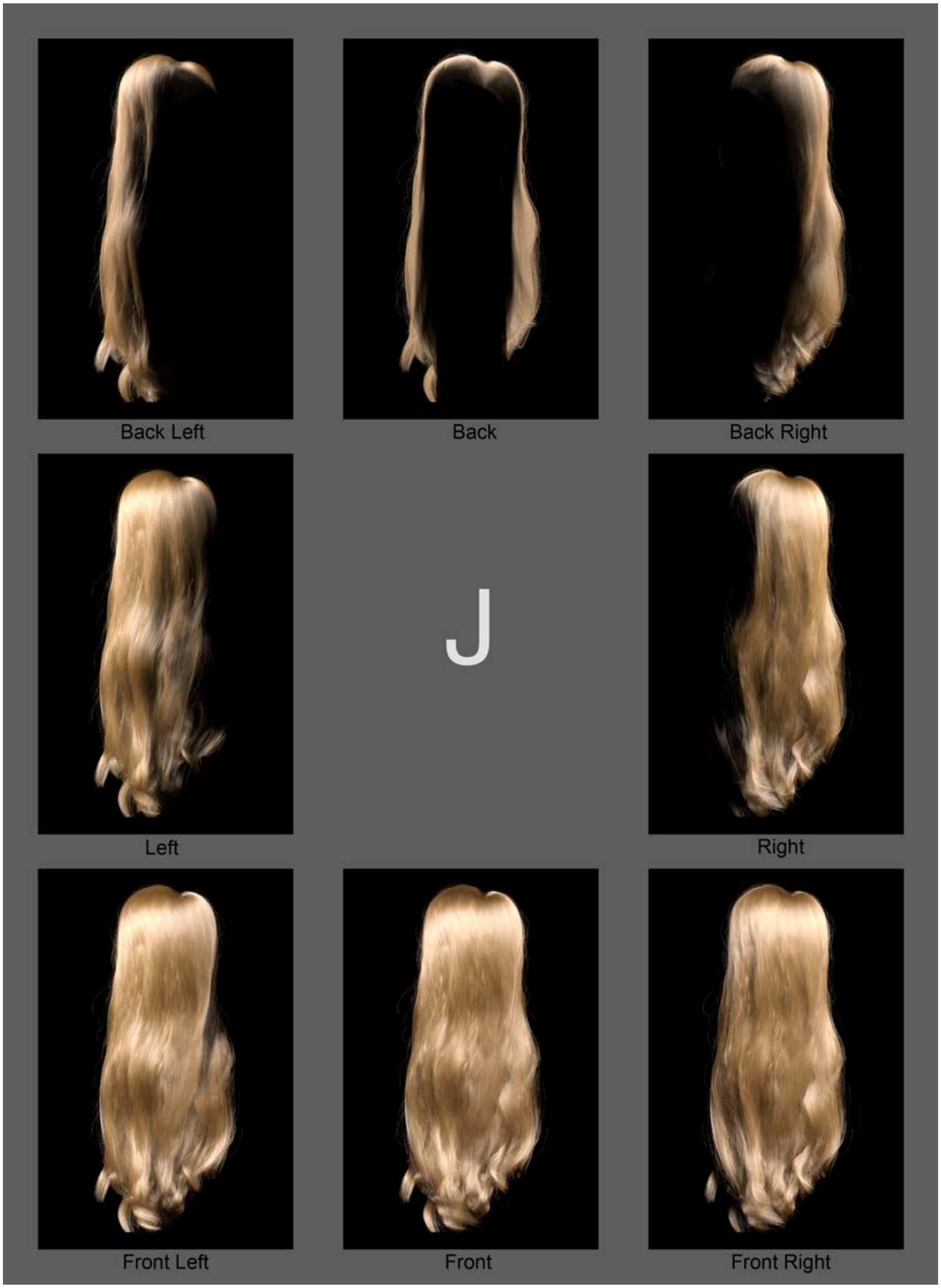


Figure D.11: Rendering results of artist J using the production shader.

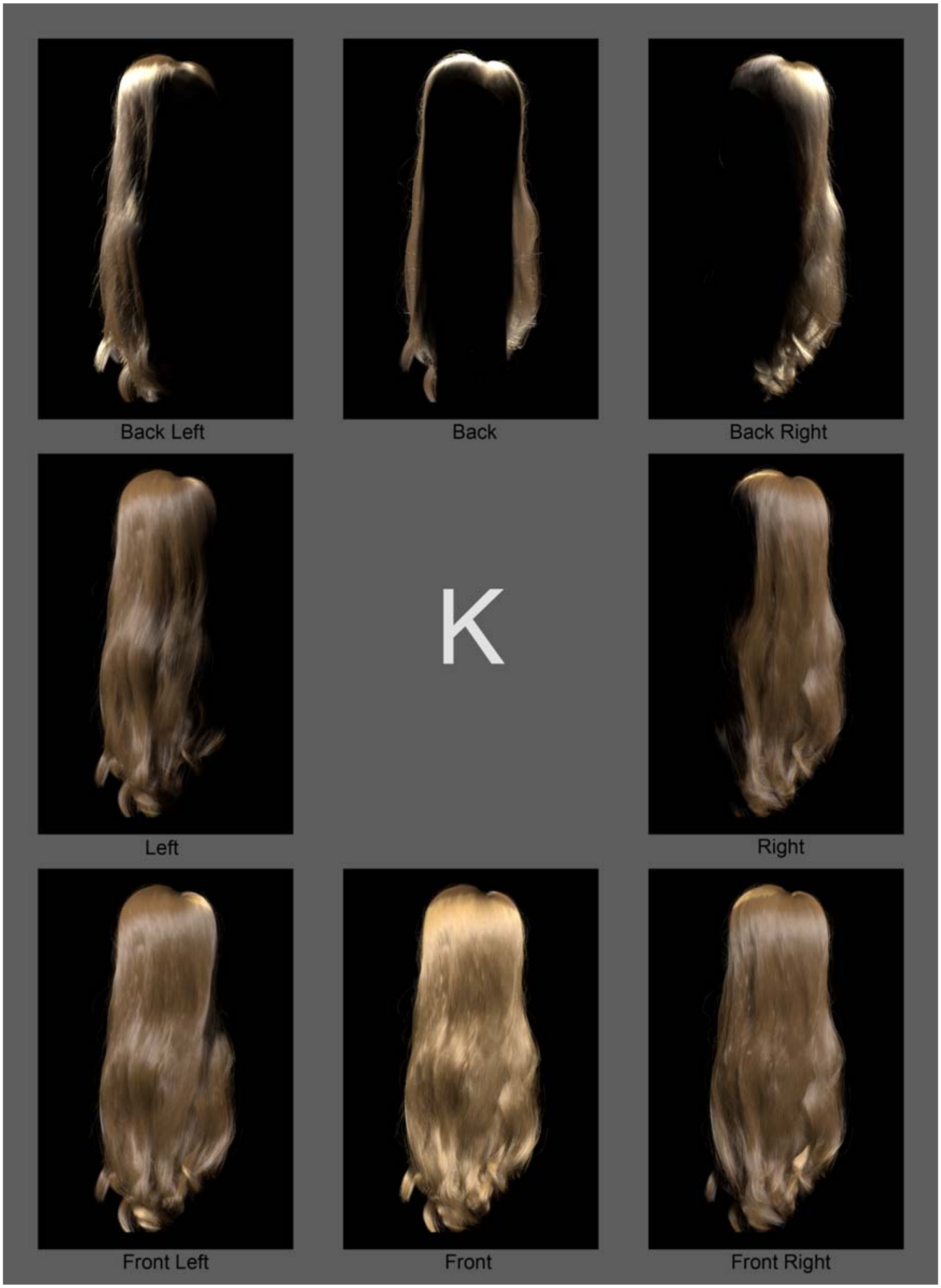


Figure D.12: Rendering results of artist K using the research shader.

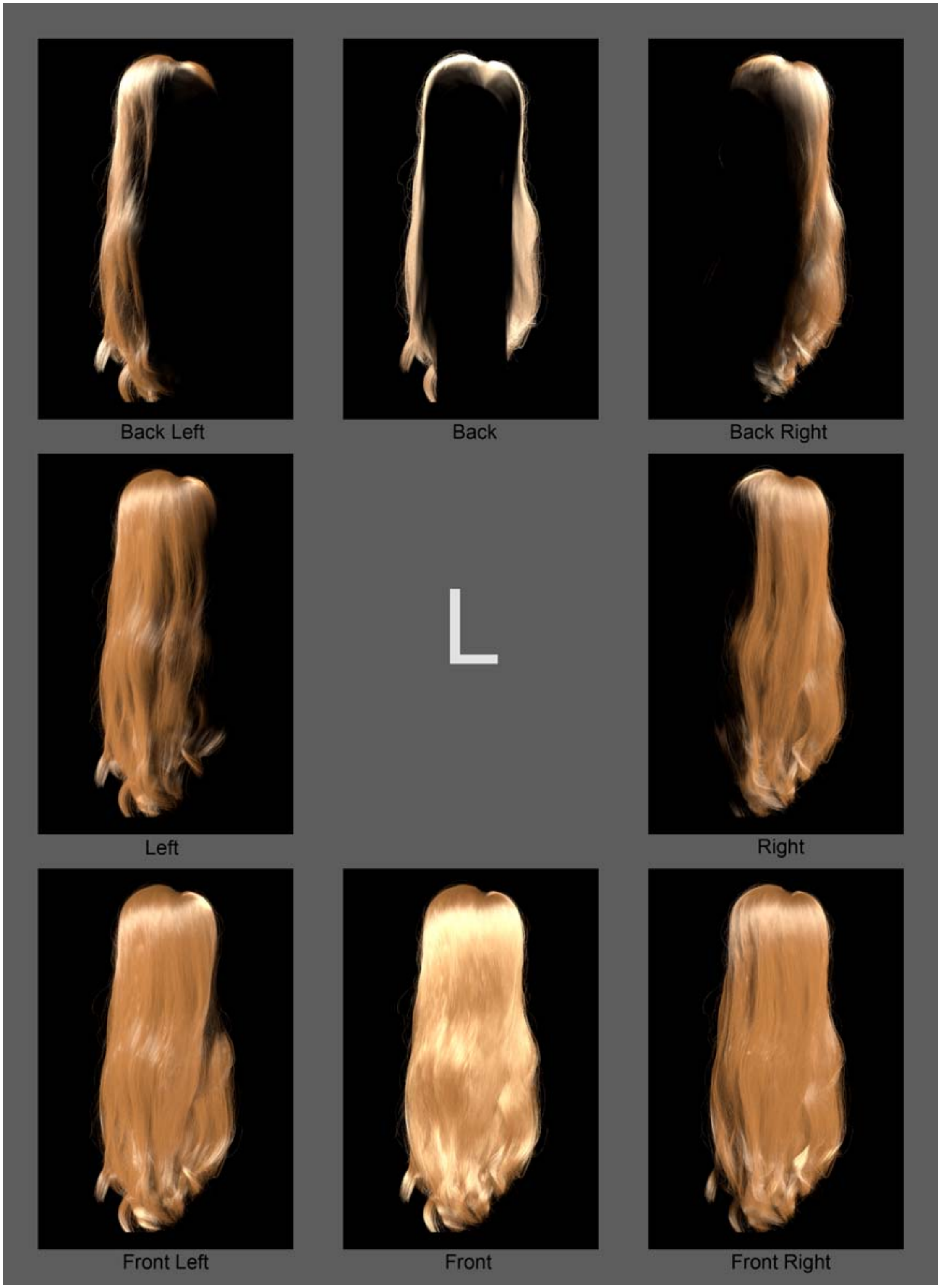


Figure D.13: Rendering results of artist L using the research shader.

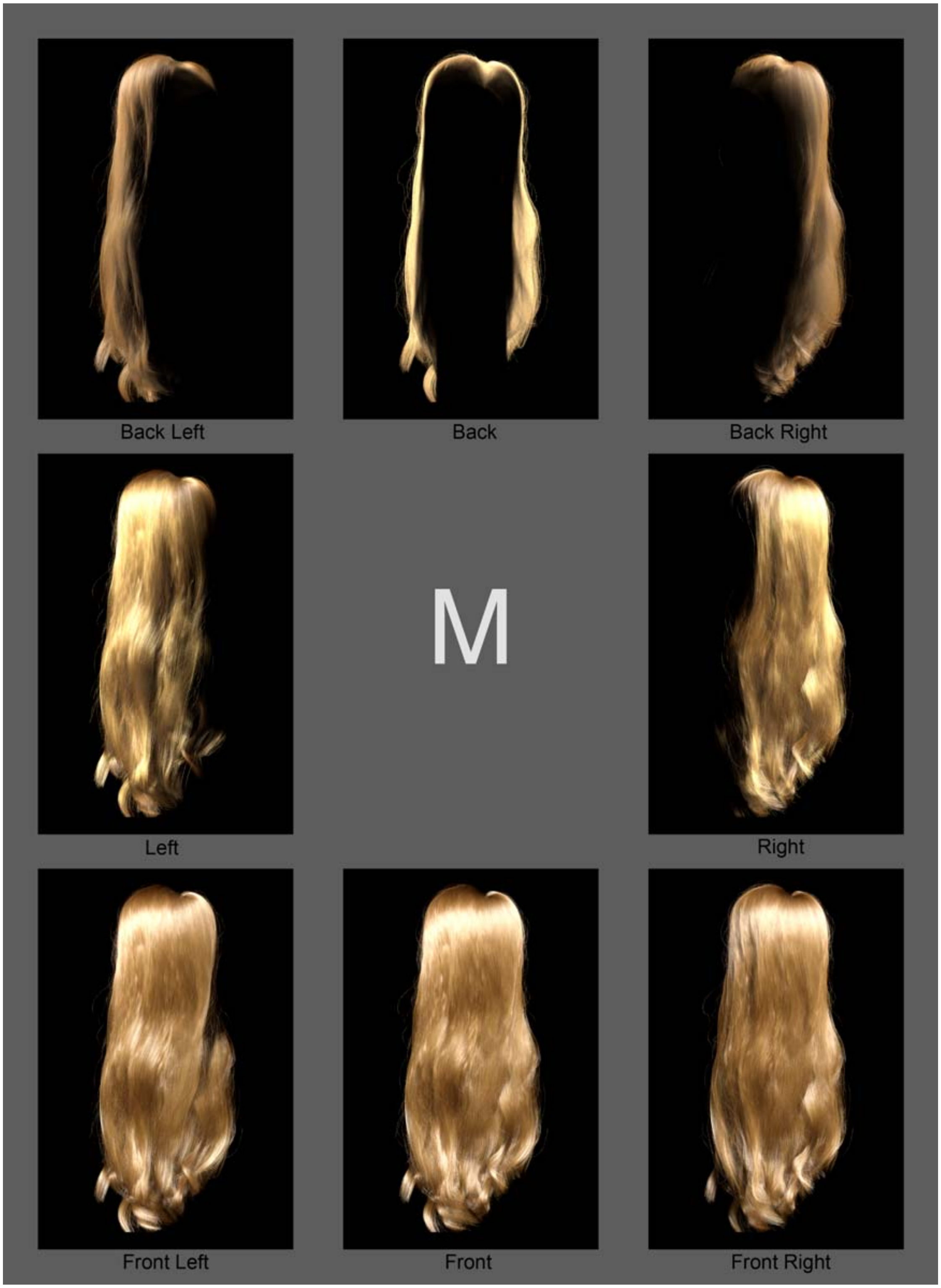


Figure D.14: Rendering results of artist M using the new shader.

Table D.1: Complete list of evaluations for the user study.

No.	**Evaluator**	**Ranking**
1	Designer	B-A-G-J-I-F-M-E-D-C-L-K-H
2	Software Engineer	B-J-M-A-E-G-I-F-K-D-H-L-C
3	Look Dev. Artist	B-J-A-M-D-I-F-E-G-K-C-L-H
4	Apprentice Animator	M-D-B-J-A-E-K-F-H-I-L-C-G
5	Layout Artist	J-B-E-M-I-G-F-K-D-A-L-C-H
6	Lighting TD	B-J-I-M-A-E-D-K-F-L-C-G-H
7	Lighting Artist	J-B-E-K-M-F-I-A-D-G-C-L-H
8	Layout Supervisor	M-B-E-D-I-J-A-K-G-F-C-L-H
9	Apprentice Animator	B-A-J-I-M-F-G-K-D-L-C-E-H
10	Efx Animator	G-F-B-M-A-J-E-D-I-H-C-L-K
11	Systems Engineer	J-F-B-A-M-D-I-K-L-C-G-E-H
12	Look Dev. Artist	(B,J)-A-G-M-E-I-F-D-K-C-L-H
13	Software Engineer	B-J-M-D-K-I-E-F-G-A-C-L-H
14	Systems Engineer	M-B-I-A-D-F-(E,J)-K-(C,L)-(G,H)
15	Modeler	B-D-M-J-E-I-A-F-L-K-G-C-H
16	Systems Engineer	J-B-A-K-M-F-E-D-I-C-G-L-H
17	Software Engineer	B-M-J-A-F-D-E-L-K-C-G-H-I
18	Systems Engineer	(J,I,B)-(E,M)-G-F-D-K-A-L-(C,H)
19	Software Engineer	D-M-L-A-B-J-C-E-I-K-F-H-G
20	Look Dev. TD	B-M-E-J-I-A-F-D-G-H-L-K-C
21	Software Engineer	B-J-A-E-M-I-D-F-K-L-G-H-C
22	Software Engineer	B-E-J-F-M-D-G-A-C-K-I-H-L
23	Lighting Artist	J-B-I-M-D-K-E-A-F-G-H-L-C
24	General TD	B-J-E-M-D-A-I-K-F-C-L-H-G
25	Systems Engineer	D-C-A-B-F-E-H-J-I-M-K-L-G
26	Look Dev. Artist	B-D-K-A-J-M-C-E-F-G-H-I-L
27	Software Engineer	B-J-A-F-M-G-D-I-E-H-K-C-L
28	Lighting Artist	J-B-A-M-L-K-D-F-G-I-E-C-H
29	Assistant TD	J-B-I-M-A-F-K-E-D-L-G-C-H
30	Lighting Apprentice	J-B-M-A-D-F-K-G-I-E-L-C-H
31	Character TD	B-J-A-(E,F)-I-G-D-K-M-(C,H,L)
32	Character TD	M-B-I-A-J-D-E-K-F-H-L-G-C
33	Look Dev. Artist	D-(M,J)-(I,E,A)-(B,F,K,L)-(C,G,H)
34	Lighting Supervisor	D-(M,J)-(I,E,A)-(B,F,K,L)-(C,G,H)
35	Art Director	B-(J,E)-(D,M)-(A,F)-I-(G,L)-(C,H,K)

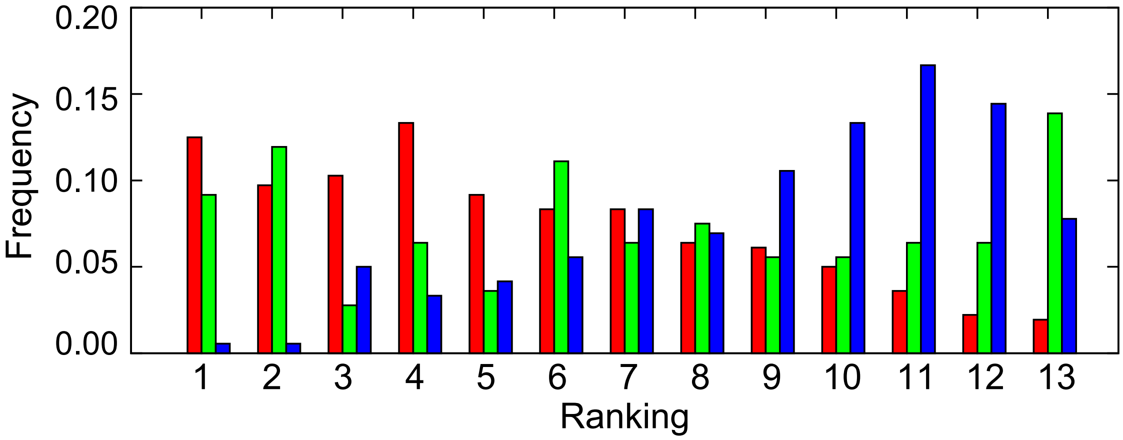


Figure D.15: Normalized distribution of the rankings for all the shading models (1 is best, 13 is worst). The New shader (red) is generally ranked better than the Research shader (blue) while the Production shader (green) is somewhat inconsistent with really good results and some not so good. The numbers have been normalized based on the number of samples in each group.

E

Tangent Distribution Curves for Cloth Rendering

IN this appendix we present the tangent distributions that is used for rendering each fabric in this paper. Figure E.1 shows the tangent curves of each fabric. The curves are defined by setting the tangent values only at the control points shown in the figure. Table E.1 summarizes the tangent offsets of all control points and the length of each segment. The tangent curves are based on the structure of the weaving pattern. Each flat section on the tangent curve is responsible for a highlight in the BRDF of the fabric.

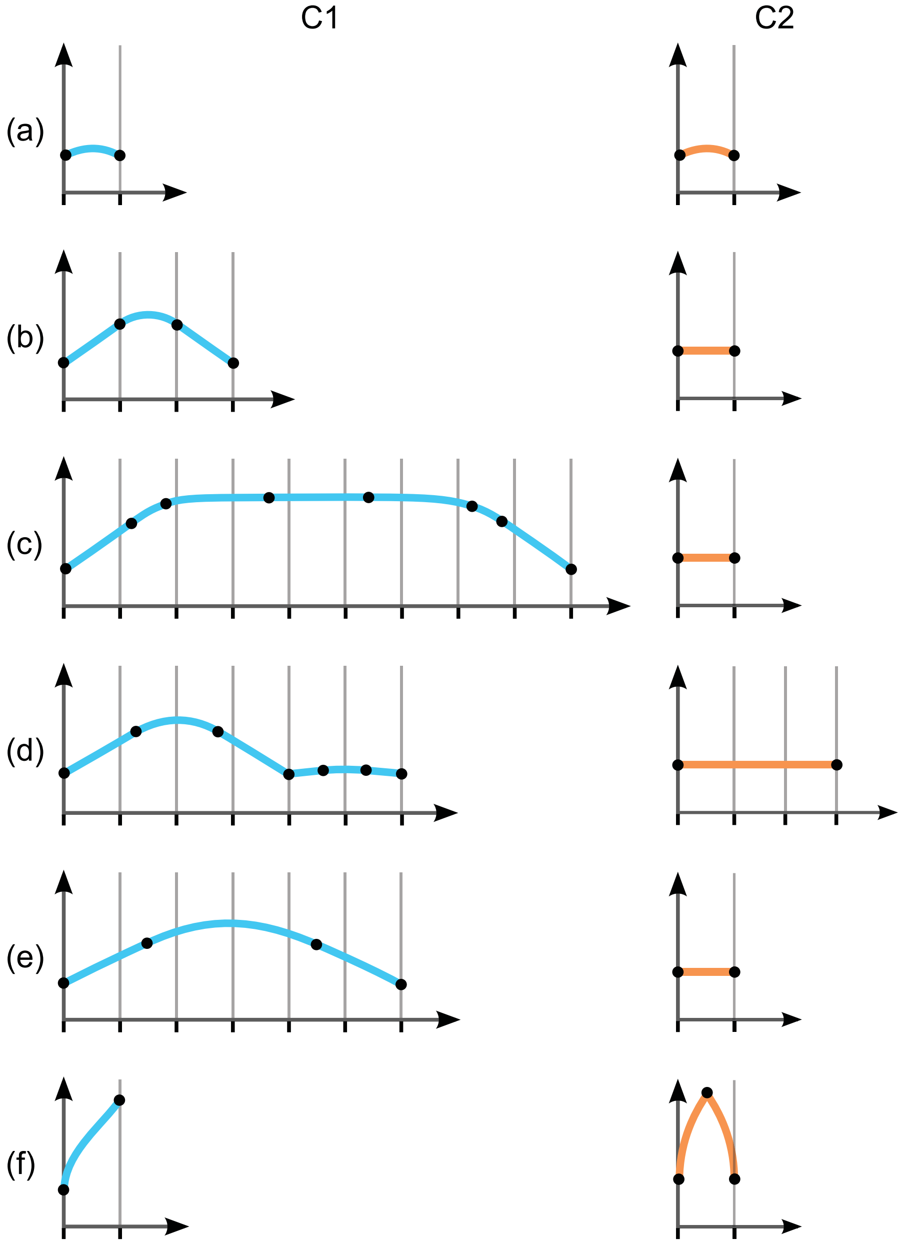


Figure E.1: The tangent curves of all fabrics presented in our cloth appearance model. The tangent curves are for (a) Linen Plain, (b) Silk Crepe de Chine, (c) front face of Polyester Satin Charmeuse, (d) back face of Polyester Satin Charmeuse, (e) Silk Shot fabric, and (f) Velvet. The numerical values are listed in Table E.1.

Table E.1: The list of parameters used to produce the tangent curves for each fabric. The tangent curves are visualized in Figure E.1.

Fabric	**Thread**	a	**Tangent Offsets (degrees)**	**Tangents Lenghts**
Linen Plain	Both	0.33	$\{25, 25\}$	$\{1\}$
Silk Crepe de Chine	Flat	0.75	$\{-35, -35, 35, 35\}$	$\{1, 1, 1\}$
	Twisted	0.25	$\{0, 0\}$	$\{1\}$
Polyester Satin Charmeuse (Back Side)	Flat	0.9	$\{-32, -32, -18, 0, 0, 18, 32, 32\}$	$\{1.33, 0.66, 2, 2, 2, 0.66, 1.33\}$
	Twisted	0.1	$\{0, 0\}$	$\{1\}$
Polyester Satin Charmeuse (Back Side)	Flat	0.67	$\{-30, -30, 30, 30, -5, -5, 5, 5\}$	$\{1.33, 1.33, 1.33, 0.67, 0.67, 0.67\}$
	Twisted	0.33	$\{0, 0\}$	$\{3\}$
Silk Shot Fabric	Dir 1	0.86	$\{-25, -25, 25, 25\}$	$\{1.33, 2.67, 1.33\}$
	Dir 2	0.14	$\{0, 0\}$	$\{1\}$
Velvet	Dir 1	0.5	$\{-90, -50\}$	$\{1\}$
	Dir 2	0.5	$\{-90, -55, 55, 90\}$	$\{0.5, 0, 0.5\}$

Bibliography

[AGL$^+$00] Tony Apodaca, Larry Gritz, Tal Lancaster, Mitch Prater, and Rob Bredow. Advanced RenderMan 2: To RI_INFINITY and Beyond. SIGGRAPH 2000 Course 40, 2000. 7, 87

[AGP$^+$01] Tony Apodaca, Larry Gritz, Matt Pharr, Christophe Hery, Kevin Bjorke, and Lawrence Treweek. Advanced RenderMan 3: Render Harder. SIGGRAPH 2001 Course 48, 2001. 7, 87

[AGP$^+$02] Tony Apodaca, Larry Gritz, Matt Pharr, Dan Goldman, Hayden Landis, Guido Quaroni, and Rob Bredow. RenderMan in Production. SIGGRAPH 2002 Course 16, 2002. 7, 87

[Air38] G.B. Airy. On the intensity of light in the neighbourhood of a caustic. *Trans. Cambridge Philos. Soc.*, 3(6):379–403, 1838. 5, 46, 52

[AMTF03a] Neeharika Adabala, Nadia Magnenat-Thalmann, and Guangzheng Fei. Real-time rendering of woven clothes. In *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '03, pages 41–47, New York, NY, USA, 2003. ACM. 139

[AMTF03b] Neeharika Adabala, Nadia Magnenat-Thalmann, and Guangzheng Fei. Real-time visualization of woven textiles. In *In Proceedings of Industrial Simulation Conference*, VRST '03, pages 502–508, 2003. 139

[AMTF03c] Neeharika Adabala, Nadia Magnenat-Thalmann, and Guangzheng Fei. Visualization of woven cloth. In *Proceedings of the 14th Eurographics workshop on Rendering*, EGRW '03, pages 178–185, Aire-la-

Ville, Switzerland, Switzerland, 2003. Eurographics Association. 9, 139, 156

[APS00] Michael Ashikhmin, Simon Premože, and Peter Shirley. A microfacet-based brdf generator. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '00, pages 65–74, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co. 9, 34, 139, 163

[Ash01] Michael Victor Ashikhmin. *Approximate methods for improving surface appearance.* PhD thesis, The University of Utah, 2001. AAI3012256. 170

[BC87] Kenneth V. Beard and Catherine Chuang. A new model for the equilibrium shape of raindrops. *Journal of the Atmospheric Sciences*, 44(11):1509–1524, 1987. xi, xvi, 47, 48, 52, 56

[BCX91] V.N. Bringi, V. Chandrasekar, and R. Xiao. Raindrop axis ratios and size distributions in florida rainshafts:an assessment of multiparameter radar algorithms. *IEEE Transactions on Geoscience and Remote Sensing*, 36(3):703–715, 1991. 52

[BH83] C.F. Bohren and D.R. Huffman. *Absorption and Scattering of Light by Small Particles.* Wiley, 1983. 52

[BKI91] Kenneth V. Beard, Rodney J. Kubesh, and Harry T. Ochs III. Laboratory measurements of small raindrop distortion. part i: Axis ratios and fall behavior. *Journal of the Atmospheric Sciences*, 48(5):698–710, 1991. 52

[Bli77] James F. Blinn. Models of light reflection for computer synthesized pictures. *SIGGRAPH Comput. Graph.*, 11:192–198, July 1977. 34

[BP91] Partha P. Banerjee and Ting-Chung Poon. *Principles of Applied Optics, The Aksen Associates Series in Electrical and Computer Engineering.* Aksen, 1991. 22

[BPvdP$^+$09] Nicolas Bonneel, Sylvain Paris, Michiel van de Panne, Frédo Durand, and George Drettakis. Single Photo Estimation of Hair Appearance. *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering)*, 28(4):1171–1180, 2009. 7, 85, 89

[BS91] H. K. Bustard and R. W. Smith. Investigation into the scattering of light by human hair. *Applied Optics*, pages 3485–3491, 1991. 80

[BT92] Neeta Bhate and A. Tokuta. Photorealistic volume rendering of media with directional scattering. In Alan Chalmers, Derek Paddon, and François Sillion, editors, *Rendering Techniques '92*, Eurographics, pages 227–246. Consolidation Express Bristol, 1992. 37

[Bus] Dan Bush. Missouri skies: Rainbow at elam bend. `http://www.missouriskies.org/rainbow/february_rainbow_2006.html`. 41

[BW99] Max Born and Emil Wolf. *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light.* Cambridge University Press, 7 edition, 1999. 20, 28

[CG85] Michael F. Cohen and Donald P. Greenberg. The hemi-cube; a radiosity solution for complex environments. *Computer Graphics (Proceedings of SIGGRAPH 85)*, 19(3):31–40, August 1985. 35

[Cha60] Subrahmanyan Chandrasekhar. *Radiative Transfer.* Dover Publications, 1960. 37, 156

[CLZ$^+$03] Yanyun Chen, Stephen Lin, Hua Zhong, Ying-Qing Xu, Baining Guo, and Heung-Yeung Shum. Realistic rendering and animation of knitwear. *IEEE Transactions on Visualization and Computer Graphics*, 9(1):43–55, 2003. 9, 140

[Col94] Steven Collins. Adaptive splatting for specular to diffuse light transport. In *In Fifth Eurographics Workshop on Rendering*, pages 119–135, 1994. 56

[Cowa] Les Cowley. Atmospheric optics: Alexander's dark band. `http://www.atoptics.co.uk/rainbows/adband.htm`. 67

[Cowb] Les Cowley. Atmospheric optics: Cloudbow. `http://www.atoptics.co.uk/rainbows/cldbow.htm`. 67

[CSS96] Per H. Christensen, Eric J. Stollnitz, and David H. Salesin. Global illumination of glossy environments using wavelets and importance. *ACM Transactions on Graphics*, 15(1):37–71, 1996. 35

[CTW$^+$04] Yanyun Chen, Xin Tong, Jiaping Wang, Stephen Lin, Baining Guo, and Heung-Yeung Shum. Shell texture functions. *ACM Trans. Graph.*, 23(3):343–353, 2004. 84

[DBB06] Philip Dutré, Philippe Bekaert, and Kavita Bala. *Advanced Global Illumination.* AK Peters, Ltd., second edition, 2006. 19

[DC04] Frédéric Drago and Norishige Chiba. Painting canvas synthesis. *Vis. Comput.*, 20(5):314–328, 2004. 9, 138

[Dek10] Joachim De Deken. A BRDF Analysis of Cloth. Master's thesis, University of California, San Diego, La Jolla, CA, 2010. 136

[Des37] Rene Descartes. Discourse on the method of rightly conducting one's reason and of seeking truth in the sciences. 1637. 49, 50

[DLHS01] Katja Daubert, Hendrik P. A. Lensch, Wolfgang Heidrich, and Hans-Peter Seidel. Efficient cloth modeling and rendering. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, pages 63–70, London, UK, 2001. Springer-Verlag. 8, 138

[DvGNK99] Kristin J. Dana, Bram van Ginneken, Shree K. Nayar, and Jan J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Trans. Graph.*, 18:1–34, January 1999. 140

[DWYW09] Nicolas Ducheneaut, Ming-Hui Wen, Nicholas Yee, and Greg Wadley. Body and Mind: A Study of Avatar Personalization in Three Virtual Worlds. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, pages 1151–1160, New York, NY, USA, 2009. ACM. 6, 76

[(Ed04] Valerie Steele (Editor), editor. *Encyclopedia of Clothing and Fashion (Scribner Library of Daily Life))*. Charles Scribners & Sons, 2004. 135

[FCF07] Jeppe Revall Frisvad, Niels Jørgen Christensen, and Peter Falster. The aristotelian rainbow: from philosophy to computer graphics. In *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, GRAPHITE '07, pages 119–128, New York, NY, USA, 2007. ACM. 4, 49

[FCJ07] Jeppe Revall Frisvad, Niels Jørgen Christensen, and Henrik Wann Jensen. Computing the scattering properties of participating media using lorenz-mie theory. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 60, New York, NY, USA, 2007. ACM. 52, 53

[Fey88] Richard P. Feynman. *QED: The Strange Theory of Light and Matter*. Princeton University Press, 1988. 19

[ftPoWS] The International Association for the Properties of Water and Steam. Release on the refractive index of ordinary water substance as a function of wavelength, temperature and pressure. `http://www.iapws.org/relguide/rindex.pdf`. 57

[GD04] Rick Glumac and David Doepp. Generalized approach to rendering fabric. In *ACM SIGGRAPH 2004 Sketches*, SIGGRAPH '04, pages 120–, New York, NY, USA, 2004. ACM. 8, 138

[Ged08] S.D. Gedzelman. Simulating rainbows in their atmospheric environment. *Applied Optics*, 47(34):176–181, 2008. 52

[Gia89] Douglas C. Giancoli. *Physics for Scientists and Engineers*. Prentice Hall, 1989. 58

[GK95] M. A. Green and M. J. Keevers. Optical properties of intrinsic silicon at 300 k. *Progress in Photovoltaics: Research and Applications*, 3:189 – 192, 1995. 17

[Gla95] Andrew S. Glassner. *Principles of Digital Image Synthesis*. Morgan Kaufmann, 1995. 19, 27, 51

[GMN94] Jay S. Gondek, Gary W. Meyer, and Jonathan G. Newman. Wavelength dependent reflectance functions. In *SIGGRAPH '94*, pages 213–220, New York, NY, USA, 1994. ACM. 43, 60

[God] Ian Goddard. Atmospheric optics: Multiple supernumerary bows. `http://www.atoptics.co.uk/rainbows/bowim46.htm`. 41

[Gol97] Dan B. Goldman. Fake Fur Rendering. In *Proceedings of SIGGRAPH 97*, pages 127–134, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co. 82, 86, 88

[Gre90] R. Greenler. *Rainbows, Halos, and Glories*. Cambridge University Press, 1990. 3, 48

[Gru93] Jonathan Grudin. Obstacles to Participatory Design in Large Product Development Organizations. In *Participatory design: principles and practices*, pages 99–119. L. Erlbaum Associates, Hillsdale, NJ, USA, 1993. 88

[GS04] Stéphane Guy and Cyril Soler. Graphics gems revisited. *ACM Transactions on Graphics (Proceedings of the SIGGRAPH conference)*, 2004. 43

[GTGB84] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modeling the interaction of light between diffuse surfaces. In *Computer Graphics (Proceedings of ACM SIGGEAPH 84)*, pages 213–222, New York, NY, USA, 1984. ACM Press. 35

[Hec01] Eugene Hecht. *Optics*. Addison Wesley, 4 edition, 2001. 17, 19

[HG41] Louis George Henyey and Jesse Leonard Greenstein. Diffuse radiation in the galaxy. *Astrophysics*, 93:70–83, 1941. 36

[HH84] Paul S. Heckbert and Pat Hanrahan. Beam tracing polygonal objects. *SIGGRAPH Comput. Graph.*, 18:119–127, January 1984. 56

[HK93] Pat Hanrahan and Wolfgang Krueger. Reflection from layered surfaces due to subsurface scattering. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '93, pages 165–174, New York, NY, USA, 1993. ACM. 37, 154

[HOJ08] Toshiya Hachisuka, Shinji Ogaki, and Henrik Wann Jensen. Progressive photon mapping. In *ACM SIGGRAPH Asia 2008 papers*, SIGGRAPH Asia '08, pages 130:1–130:8, New York, NY, USA, 2008. ACM. 35

[HQ73] George M. Hale and Marvin R. Querry. Optical constants of water in the 200-nm to 200-μm wavelength region. *Appl. Opt.*, 12(3):555–563, Mar 1973. xvi, 57

[HTSG91] Xiao D. He, Kenneth E. Torrance, François X. Sillion, and Donald P. Greenberg. A comprehensive physical model for light reflection. In *Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '91, pages 175–186, New York, NY, USA, 1991. ACM. 34

[ICG86] David S. Immel, Michael F. Cohen, and Donald P. Greenberg. A radiosity method for non-diffuse environments. *SIGGRAPH Comput. Graph.*, 20:133–142, August 1986. 35

[IM06] P. Irawan and S. Marschner. A simple, accurate texture model for woven cotton cloth. Technical report, Cornell University, Department of Computer Science, 2006. 9, 140

[Ira08] Piti Irawan. *Appearance of woven cloth*. PhD thesis, Cornell University, Ithaca, NY, USA, 2008. AAI3295837. 9, 132, 140, 156

[Jac] Vincent Jacques. Rainbows. `http://vjac.free.fr/skyshows/waterdroplets/rainbows.html`. 42

[JAM$^+$10] Wenzel Jakob, Adam Arbree, Jonathan T. Moon, Kavita Bala, and Steve Marschner. A radiative transfer framework for rendering materials with anisotropic structure. *ACM Trans. Graph.*, 29:53:1–53:13, July 2010. 9, 140, 156

[JC98] Henrik Wann Jensen and Per H. Christensen. Efficient simulation of light transport in scences with participating media using photon maps. In *Computer Graphics (Proceedings of SIGGRAPH 98)*, pages 311–320, New York, NY, USA, 1998. ACM Press. 37

[Jen01] Henrik Wann Jensen. *Realistic Image Synthesis Using Photon Mapping.* A. K. Peters, Ltd., Natick, MA, USA, 2001. 35, 84

[JF01] Raymond L. Lee Jr and Alistair B. Fraser. The rainbow bridge: Rainbows in art, myth, and science. 2001. 49

[JNSJ11] Wojciech Jarosz, Derek Nowrouzezahrai, Iman Sadeghi, and Henrik Wann Jensen. A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Trans. Graph.*, 30:5:1–5:19, February 2011. 37

[JON41] R. CLARK JONES. A new calculus for the treatment of optical systems. *J. Opt. Soc. Am.*, 31(7):488–493, Jul 1941. 60

[JW97] D. Jackèl and B. Walter. Modeling and rendering of the atmosphere using Mie-scattering. *Computer Graphics Forum*, 16(4):201–210, 1997. ISSN 1067-7055. 52

[Kai] Karl Kaiser. Der homepage von karl kaiser: Nature. `http://home.eduhi.at/member/nature`. 67

[Kaj85] James T. Kajiya. Anisotropic reflection models. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '85, pages 15–21, New York, NY, USA, 1985. ACM. 34

[Kaj86] James T. Kajiya. The rendering equation. In *Computer Graphics (Proceedings of SIGGRAPH 86)*, pages 143–150, New York, NY, USA, 1986. ACM Press. 35

[Kh] Benjamin Khne. Anomaler regenbogen am 03. juni 2002. `http://www.nachtwolke.de/temp/regenbogen2.htm`. 42, 73

[Kim02] Tae-Yong Kim. *Modeling, Rendering and Animating Human Hair.* PhD thesis, University of Southern California, Los Angeles, CA, USA, 2002. 6, 82, 99, 152, 154

[KK89] J. T. Kajiya and T. L. Kay. Rendering Fur With Three Dimensional Textures. In *Computer Graphics (Proceedings of SIGGRAPH 89)*, pages 271–280, New York, NY, USA, 1989. ACM. 6, 81, 152

[KVH84] James T. Kajiya and Brian P Von Herzen. Ray tracing volume densities. *SIGGRAPH Comput. Graph.*, 18:165–174, January 1984. 37

[Lam60] J. H. Lambert. Photometria sive de mensure de gratibus luminis, colorum umbrae. *Eberhard Klett*, 1760. 34

[Lav03] P. Laven. Simulation of rainbows, coronas, and glories by use of Mie theory. *Applied Optics*, 42(3):436–444, 2003. 52

[Lee98] Raymond L. Lee. Mie theory, Airy theory, and the natural rainbow. *Applied Optics*, 37(9):1506–1519, 1998. 51, 52

[LF01] Raymond L. Lee and Alistair B. Fraser. *The rainbow bridge: rainbows in art, myth, and science*. Pennsylvania State University Press, 2001. 4, 40, 44, 45

[Lin66] David C Lindberg. Roger bacon's theory of the rainbow: Progress or regress? 1966. 49

[LL01] D. Lynch and W. Livingstone. *Color and Light in Nature (second edition)*. Cambridge University Press, 2001. 3, 48

[LLT95] S.G. Lipson, H. Lipson, and D.S. Tanhauser. *Optical Physics*. Cambridge, third edition edition, 1995. 60

[Lor90] L. Lorenz. Lysbevgelser i og uden for en af plane lysblger belyst kugle. det kongelig danske videnskabernes selskabs skrifter. *Rkke, Naturvidenskabelig og Mathematisk Afdeling*, 6(1):262, 1890. 4, 36, 50, 52

[LW93] Eric P. Lafortune and Yves D. Willems. Bi-directional path tracing. In H. P. Santo, editor, *Proceedings of Third International Conference on Computational Graphics and Visualization Techniques (Compugraphics '93)*, pages 145–153, Alvor, Portugal, 1993. 35

[LW96] Eric P. Lafortune and Yves D. Willems. Rendering participating media with bidirectional path tracing. In Xavier Pueyo and Peter Schröder, editors, *Rendering Techniques '96*, Eurographics, pages 91–100. Springer-Verlag Wien New York, 1996. 37

[Max94] Nelson Max. Efficient light propagation for multiple anisotropic volume scattering. In Georgios Sakas, Peter Shirley, and Stefan Müller, editors, *Rendering Techniques '94*, Eurographics, pages 87–104. Springer-Verlag Berlin Heidelberg New York, 1994. 37

[Mie08] G. Mie. Beitr age zur optik tr uber medien, speziell kolloidaler metall osungen. *Annalen der Physik*, 25(3):377445, 1908. 4, 36, 50, 52

[Min93] M. Minnaert. *Light and Colour in the Outdoors.* Springer-Verlag, 1993. 3, 48

[MJC$^+$03] Stephen R. Marschner, Henrik Wann Jensen, Mike Cammarano, Steve Worley, and Pat Hanrahan. Light Scattering from Human Hair Fibers. *ACM Transactions on Graphics*, 22(3):780–791, 2003. 6, 78, 79, 80, 82, 96, 97, 102, 113, 152, 153, 154, 156

[MM98] Sivakumar Manickavasagam and M. Pinar Mengüç. Scattering-matrix elements of coated infinite-length cylinders. *Appl. Opt.*, 37(12):2473–2482, Apr 1998. 17

[MM06] Jonathan T. Moon and Stephen R. Marschner. Simulating Multiple Scattering in Hair Using a Photon Mapping Approach. *ACM Transactions on Graphics*, 25(3):1067–1074, 2006. 6, 84, 103, 109

[MTLB05] Tadao Mihashi, Christina Tempelaar-Lietz, and George Borshukov. Generating Realistic Human Hair for "The Matrix Reloaded". In *SIGGRAPH '05: ACM SIGGRAPH 2005 Courses*, page 17, New York, NY, USA, 2005. ACM. 89

[Mus89] F. Kenton Musgrave. Prisms and rainbows: A dispersion model for computer graphics. In *Graphics Interface '89*, pages 227–234, June 1989. 4, 49

[MW03] Alister D. Muir and Neil D. Westcott, editors. *Flax: The genus Linum (Medicinal and Aromatic Plants - Industrial Profiles).* CRC Press, 2003. 135

[MWM08] Jonathan T. Moon, Bruce Walter, and Steve Marschner. Efficient Multiple Scattering in Hair Using Spherical Harmonics. *ACM Transactions on Graphics*, 27(3):31:1–31:7, 2008. 6, 103, 109

[ND86] Donald A. Norman and Stephen W. Draper, editors. *User Centered System Design; New Perspectives on Human-Computer Interaction.* L. Erlbaum Associates, Hillsdale, NJ, USA, 1986. 87, 94

[NDM05] A Ngan, F Durand, and W Matusik. *Experimental Analysis of BRDF Models*, volume 2, pages 117–226. Eurographics Association, 2005. 140

[Neu04] Ivan Neulander. Quick Image-Based Lighting of Hair. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Sketches*, page 43, New York, NY, USA, 2004. ACM. 7, 87

[New04] Isaac Newton. Opticks. 1704. 49, 50

[NN85] Tomoyuki Nishita and Eihachiro Nakamae. Continuous tone representation of three-dimensional objects taking account of shadows and interreflection. *Computer Graphics (Proceedings of SIGGRAPH 85)*, 19(3):23–30, 1985. 35

[NN03] Srinivasa G. Narasimhan and Shree K Nayar. Shedding light on the weather. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 665 – 672, June 2003. 43

[Nor02] Donald A. Norman. *The Design of Everyday Things.* Basic Books, New York, NY, USA, 2002. 87

[nVI04] nVIDIA. Rainbows and fogbows: adding natural phenomena. Sdk white paper, nVIDIA Corporation, 2004. 52

[ON94] Michael Oren and Shree K. Nayar. Generalization of lambert's reflectance model. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, SIGGRAPH '94, pages 239–246, New York, NY, USA, 1994. ACM. 34

[Par92] Julie Parker. *All About Silk: A Fabric Dictionary and Swatchbook (Fabric Reference Series, Volume 1).* Rain City Publishing, 1992. 135

[PAT+04] S. Premoze, M. Ashikhmin, J. Tesendorf, R. Ramamoorthi, and S. NAYAR. Practical rendering of multiple scattering effects in participating media, 2004. 84

[PF90] Pierre Poulin and Alain Fournier. A model for anisotropic reflection. In *Proceedings of the 17th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '90, pages 273–282, New York, NY, USA, 1990. ACM. 160

[PF97] R.M Pope and E.S Fry. Absorption spectrum (380-700 nm) of pure water. ii. integrating cavity measurements. *Applied Optics*, 36(33):8710–8723, 1997. 43

[PH04] Matt Pharr and Greg Humphreys. *Physically Based Rendering: From Theory to Implementation.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004. 31

[PHA05] Lena Petrovic, Mark Henne, and John Anderson. Volumetric Methods for Simulation and Rendering of Hair. Technical Report 06-08, Pixar Animation Studios, 2005. 87

[Pho75] Bui Tuong Phong. Illumination for computer generated pictures. *Commun. ACM*, 18:311–317, June 1975. 34

[PK03] Sylvia C Pont and Jan J Koenderink. Split off-specular reflection and surface scattering from woven materials. *Appl Opt*, 42(8):1526–33, 2003. 170, 171, 172

[PKK00] Mark Pauly, Thomas Kollig, and Alexander Keller. Metropolis light transport for participating media. In *Proceedings of the Eurographics Workshop on Rendering Techniques 2000*, pages 11–22, London, UK, 2000. Springer-Verlag. 37

[PM93] Sumanta N. Pattanaik and Sudhir P. Mudur. Computation of global illumination in a participating medium by Monte Carlo simulation. *The Journal of Visualization and Computer Animation*, 4(3):133–152, July–September 1993. 37

[Ray71] John William Strutt Lord Rayleigh. On the scattering of light by small particles. *Philosophical Magazine*, 61:447–454, 1871. 36

[REK$^+$04] Kirk Riley, David S. Ebert, Martin Kraus, Jerry Tessendorf, and Charles Hansen. Efficient rendering of atmospheric phenomena. In Dieter Fellner and Stephen Spencer, editors, *Proceedings of the 2004 Eurographics Symposium on Rendering*, pages 375–386, 2004. 52

[Rob94] C. R. Robbins. *Chemical and Physical Behavior of Human Hair, third ed.* Van Nostrand Reinhold Co., 1994. 17, 80, 81

[RT87] Holly E. Rushmeier and Kenneth E. Torrance. The zonal method for calculating light intensities in the presence of a participating medium. *Computer Graphics (Proceedings of SIGGRAPH 87)*, 21(4):293–302, July 1987. 37

[Rus95] Holly Rushmeier. Input for participating media. Realistic Input for Realistic Images, ACM SIGGRAPH '95 Course Notes. Also appeared in the ACM SIGGRAPH '98 Course Notes - A Basic Guide to Global Illumination, 1995. 52

[Sar06] Sartomer. Refractive index and product selection. `http://www.sartomer.com`, 2006. 17

[Sav99] B. P. Saville. *Physical Testing of Textiles.* CRC Press, 1999. 148

[SAWG91] François X. Sillion, James R. Arvo, Stephen H. Westin, and Donald P. Greenberg. A global illumination solution for general reflectance distributions. *Computer Graphics (Proceedings of SIGGRAPH 91)*, 25(4):187–196, 1991. 35

[SJK06] Anna L. Langford Sara J. Kadolph. *Textiles.* Prentice Hall, 2006. 135, 137

[SN93] Douglas Schuler and Aki Namioka, editors. *Participatory Design: Principles and Practices.* L. Erlbaum Associates, Hillsdale, NJ, USA, 1993. 87

[SPJT10] Iman Sadeghi, Heather Pritchett, Henrik Wann Jensen, and Rasmus Tamstorf. An artist friendly hair shading system. *ACM Trans. Graph.*, 29:56:1–56:10, July 2010. 154

[SSK03] Mirko Sattler, Ralf Sarlette, and Reinhard Klein. Efficient and realistic visualization of cloth. In *Proceedings of the 14th Eurographics workshop on Rendering*, EGRW '03, pages 167–177, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association. 140

[ST07] Bahaa E. A. Saleh and Malvin Carl Teich. *Fundamentals of Photonics.* Wiley-Interscience, 2 edition, March 2007. 14, 18, 19, 25, 29

[ST10] Iman Sadeghi and Rasmus Tamstorf. Efficient Implementation of the Dual Scattering Model in RenderMan. Technical report, Walt Disney Animation Studios, 2010. 103, 105

[Str90] Paul S. Strauss. A realistic lighting model for computer animators. *IEEE Comput. Graph. Appl.*, 10:56–64, November 1990. 34

[Taf98] Allen Taflove. *Advances in Computational Electrodynamics: The Finite-Difference Time-Domain Method.* Artech House Inc, 1998. 5, 18, 26, 53

[Taj04] Fumiaki Tajima. Development of measurement system for diameter and refractive index of silk fibroin filament based on scattered light pattern. *Journal of Insect Biotechnology and Sericology*, 73(2):85–88, 2004. 17

[TRI70] R. A. R. TRICKER. Introduction to meteorological optics. Mills & Boon, London, 1970. 83

[TS92] K. E. Torrance and E. M. Sparrow. Radiometry. chapter Theory for off-specular reflection from roughened surfaces, pages 32–41. Jones and Bartlett Publishers, Inc., 1992. 34

[TTW94] David C. Tannenbaum, Peter Tannenbaum, and Michael J. Wozny. Polarization and birefringency considerations in rendering. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, SIGGRAPH '94, pages 221–222, New York, NY, USA, 1994. ACM. 60

[UT82] K. Umashankar and A. Taflove. A novel method to analyze electromagnetic scattering of complex objects. *Electromagnetic Compatibility, IEEE Transactions on*, EMC-24(4):397 –405, nov. 1982. 53

[van57] H. C. van de Hulst. *Light Scattering by Small Particles*. Dover Publications Inc., 1957. 45, 46, 52, 61

[VB09] Emmanuel Villermaux and Benjamin Bossa. *Nature Physics*, 5:697–702, 2009. 52

[VG94] Eric Veach and Leonidas J. Guibas. Bidirectional estimators for light transport. In Georgios Sakas, Peter Shirley, and Stefan Müller, editors, *Rendering Techniques '94*, Eurographics, pages 145–167. Springer-Verlag Berlin Heidelberg New York, 1994. 35

[VG97] Eric Veach and Leonidas J. Guibas. Metropolis light transport. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '97, pages 65–76, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co. 35

[VKKK97] Vladimir L. Volevich, Edward A. Kopylov, Andrei B. Khodulev, and Olga A. Karpenko. An approach to cloth synthesis and visualization. In Proceedings of GRAPHICON '97, 1997. 9, 138

[War92] Gregory J. Ward. Measuring and modeling anisotropic reflection. *SIGGRAPH Comput. Graph.*, 26:265–272, July 1992. 34

[WAT92] Stephen H. Westin, James R. Arvo, and Kenneth E. Torrance. Predicting reflectance functions from complex surfaces. *SIGGRAPH Comput. Graph.*, 26(2):255–264, 1992. 9, 138

[WBK+07] Kelly Ward, Florence Bertails, Tae-Yong Kim, Stephen R. Marschner, and Marie-Paule Cani. A Survey on Hair Modeling: Styling, Simulation, and Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):213–234, 2007. 84

[Wei86] Jerry Weil. The synthesis of cloth objects. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '86, pages 49–54, New York, NY, USA, 1986. ACM. 8, 138

[WK90] Lawrence B. Wolff and David J. Kurlander. Ray tracing with polarization parameters. *IEEE Comput. Graph. Appl.*, 10:44–55, November 1990. 60

[WRG$^+$09] Jiaping Wang, Peiran Ren, Minmin Gong, John Snyder, and Baining Guo. All-frequency rendering of dynamic, spatially-varying reflectance. In *ACM SIGGRAPH Asia 2009 papers*, SIGGRAPH Asia '09, pages 133:1–133:10, New York, NY, USA, 2009. ACM. 140

[WTP01] Alexander Wilkie, Robert F. Tobler, and Werner Purgathofer. Combined rendering of polarization and fluorescence effects. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, pages 197–204, London, UK, 2001. Springer-Verlag. 60

[WZT$^+$08] Jiaping Wang, Shuang Zhao, Xin Tong, John Snyder, and Baining Guo. Modeling anisotropic surface reflectance with example-based microfacet synthesis. In *ACM SIGGRAPH 2008 papers*, SIGGRAPH '08, pages 41:1–41:9, New York, NY, USA, 2008. ACM. 9, 139

[XCL$^+$01] Ying-Qing Xu, Yanyun Chen, Stephen Lin, Hua Zhong, Enhua Wu, Baining Guo, and Heung-Yeung Shum. Photorealistic rendering of knitwear using the lumislice. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, pages 391–398, New York, NY, USA, 2001. ACM. 9, 140

[Yee66] Kane Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *Antennas and Propagation, IEEE Transactions on*, 14(3):302 –307, may 1966. 5, 18, 26, 53

[YL95] Ping Yang and K. N. Liou. Light scattering by hexagonal ice crystals: comparison of finite-difference time domain and geometric optics models. *J. Opt. Soc. Am.*, 12:162–176, 1995. 53

[YL96] Ping Yang and K. N. Liou. Finite-difference time domain method for light scattering by small ice crystals in three-dimensional space. *J. Opt. Soc. Am.*, 13:2072–2085, 1996. 53

[YYTI92] Takami Yasuda, Shigeki Yokoi, Jun-ichiro Toriwaki, and Katsuhiko Inagaki. A shading model for cloth objects. *IEEE Comput. Graph. Appl.*, 12:15–24, November 1992. 9, 140

[ZJMB11] Shuang Zhao, Wenzel Jakob, Steve Marschner, and Kavita Bala. Building volumetric appearance models of fabric using micro ct imaging. *ACM Trans. Graph.*, July 2011. 9, 140, 175

[ZRL+09] Arno Zinke, Martin Rump, Tomás Lay, Andreas Weber, Anton Andriyenko, and Reinhard Klein. A Practical Approach for Photometric Acquisition of Hair Color. *ACM Transactions on Graphics*, 28(5):165:1–165:9, 2009. 7, 85, 89

[ZSW04] A. Zinke, G. Sobottka, and A. Weber. Photo-realistic rendering of blond hair. In *Vision, Modeling, and Visualization (VMV) 2004*, pages 191–198. Akademische Verlagsgesellschaft Aka GmbH, November 2004. 83, 84

[ZW06] Arno Zinke and Andreas Weber. Global Illumination for Fiber Based Geometries. In *Electronic proceedings of the Ibero American Symposium on Computer Graphics (SIACG)*, Santiago de Compostela, Spain, 2006. 6, 84, 103, 109

[ZW07] Arno Zinke and Andreas Weber. Light Scattering from Filaments. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):342–356, 2007. 78, 83, 84, 103, 109

[ZYWK08] Arno Zinke, Cem Yuksel, Andreas Weber, and John Keyser. Dual Scattering Approximation for Fast Multiple Scattering in Hair. *ACM Transactions on Graphics*, 27(3):32:1–32:10, 2008. 6, 84, 103, 104, 105, 107, 108, 109, 110, 117