



Advanced

Lighting and Materials with Shaders



Kelly Dempski
Emmanuel Vieille



TEAM Ling - Live, Informative, Non-cost and Genuine!

Advanced Lighting and Materials with Shaders

Kelly DempSKI and Emmanuel Viale

Wordware Publishing, Inc.

Library of Congress Cataloging-in-Publication Data

Dempski, Kelly.

Advanced lighting and materials with shaders / by Kelly Dempski and Emmanuel Viale.

p. cm.

Includes bibliographical references and index.

ISBN 1-55622-292-0 (pbk., companion cd-rom)

1. Computer graphics. 2. Light—Computer simulation. I. Viale, Emmanuel. II. Title.

T385.D467 2004

006.6'9—dc22

2004018415

© 2005, Wordware Publishing, Inc.

All Rights Reserved

2320 Los Rios Boulevard

Plano, Texas 75074

No part of this book may be reproduced in any form or by any means
without permission in writing from Wordware Publishing, Inc.

Printed in the United States of America

ISBN 1-55622-292-0

10 9 8 7 6 5 4 3 2 1

0409

All brand names and product names mentioned in this book are trademarks or service marks of their respective companies. Any omission or misuse (of any kind) of service marks or trademarks should not be regarded as intent to infringe on the property of others. The publisher recognizes and respects all marks used by companies, manufacturers, and developers as a means to distinguish their products.

This book is sold as is, without warranty of any kind, either express or implied, respecting the contents of this book and any disks or programs that may accompany it, including but not limited to implied warranties for the book's quality, performance, merchantability, or fitness for any particular purpose. Neither Wordware Publishing, Inc. nor its dealers or distributors shall be liable to the purchaser or any other person or entity with respect to any liability, loss, or damage caused or alleged to have been caused directly or indirectly by this book.

All inquiries for volume purchases of this book should be addressed to Wordware Publishing, Inc., at the above address. Telephone inquiries may be made by calling:

(972) 423-0090

TEAM LinG - Live, Informative, Non-cost and Genuine!

In memory of Helen Gaede
KD

To Alexandre, Nicolas, and Valérie
EV

This page intentionally left blank.

Contents

Foreword	xvii
Acknowledgments	xviii
Introduction	xix
Chapter 1 The Physics of Light	1
Introduction	1
1.1 The Duality of Light.	1
1.2 Light as Particles	2
1.3 Light as Waves	4
1.3.1 Wavelength and Color.	5
1.3.2 Phase and Interference	6
1.4 Light as Energy	7
Conclusion and Further Reading	8
References.	8
Chapter 2 Modeling Real-World Lights	9
Introduction	9
2.1 Ideal Point Lights	10
2.1.1 Point Lights as Energy Sources	10
2.1.2 Geometric Attenuation	12
2.1.3 Attenuation through Matter	13
2.1.4 Point Lights and Lighting Equations	14
2.2 Directional Lights	15
2.2.1 The Relationship between Point and Directional Lights	15
2.2.2 Directional Lights and Lighting Equations	16
2.3 Area Lights.	17
2.3.1 The Relationship between Point and Area Lights.	17
2.3.2 Attenuation and Area Lights.	18

2.4 Spotlights	21
2.4.1 Spotlights as Physical Lights	21
2.4.2 Spotlights and Lighting Equations.	22
2.4.3 Other Spotlight Models	24
2.5 Global Illumination	25
2.5.1 Global Illumination vs. Local Illumination.	25
2.5.2 Ambient Light	26
Conclusion	28
Chapter 3 Raytracing and Related Techniques.	29
Introduction	29
3.1 The Raytracing Algorithm	30
3.1.1 Backward Raytracing	30
3.1.2 Camera Models	34
3.1.3 The Different Types of Rays.	36
3.1.4 Recursion	38
3.1.5 Ray and Object Intersections	38
3.1.6 Texturing and Shading.	39
3.1.7 Problems and Limitations	43
3.1.8 Solutions and Workarounds	45
3.1.9 The Algorithm	50
3.2 Extending the Raytracing Algorithm	51
3.2.1 Stochastic Sampling	51
3.2.2 Path Tracing and Related Techniques	55
3.2.3 Photon Mapping	58
3.3 Real-time Raytracing.	59
3.4 Raytracing Concepts for Other Techniques	60
Conclusion	61
References	61
Other Resources	62

Chapter 4 Objects and Materials 63

Introduction 63

4.1 Plastics 64

4.2 Wood 66

 4.2.1 Trees 66

 4.2.2 Lumber. 67

 4.2.3 Finished Wood 67

4.3 Leaves and Vegetation 68

4.4 Metals 70

4.5 Concrete and Stone 71

 4.5.1 Concrete 71

 4.5.2 Brick 72

 4.5.3 Natural Stone 72

4.6 Skin. 73

4.7 Hair and Fur 74

4.8 Air and the Atmosphere 75

4.9 Transparent Materials 76

 4.9.1 Glass 76

 4.9.2 Water. 77

4.10 Paint. 77

4.11 Worn Materials 78

Conclusion 79

Chapter 5 Lighting and Reflectance Models 81

Introduction 81

5.1 The Rendering Equation. 82

5.2 Basic Illumination Definitions 83

 5.2.1 Irradiance and Illuminance. 83

 5.2.2 Radiance and Luminance 83

5.3 Lambert’s Law for Illumination 84

5.4 Bidirectional Reflectance Distribution Functions (BRDFs). 87

 5.4.1 Parameters to a BRDF. 88

5.4.2 Isotropic vs. Anisotropic Materials	89
5.4.3 BRDFs vs. Shading Models	90
5.5 Diffuse Materials	90
5.5.1 A Simple Diffuse Shading Model	90
5.5.2 Diffuse Materials and Conservation of Energy	91
5.5.3 Purely Diffuse Materials	93
5.6 Specular Materials	93
5.6.1 Purely Specular Materials	93
5.6.2 Specular and the Phong Model	94
5.6.3 The Blinn-Phong Model	95
5.6.4 Combining Diffuse and Specular Reflection	97
5.7 Diffuse Reflection Models	97
5.7.1 Oren-Nayar Diffuse Reflection	97
5.7.2 Minnaert Reflection	99
5.8 Specular and Metallic Reflection Models	100
5.8.1 Ward Reflection Model	100
5.8.2 Schlick Reflection Model	102
5.8.3 Cook-Torrance Model	103
5.8.3.1 The Geometric Term	103
5.8.3.2 The Fresnel Term	104
5.8.3.3 The Roughness Term	104
5.8.3.4 The Complete Cook-Torrance Model	105
Conclusion	106
References	107
Chapter 6 Implementing Lights in Shaders	109
Introduction	109
6.1 Basic Lighting Math	110
6.1.1 Color Modulation	110
6.1.2 Object Space Light Vectors	111
6.1.3 Putting the Basics Together	113
6.2 Per-Vertex Warn Lights	113
6.2.1 The Warn Shader	114

6.2.2 The Warn Application 116

6.2.3 The Results. 118

6.3 Per-Pixel Warn Lights 120

6.3.1 PS2.0 Lighting 120

6.3.2 The Results. 123

6.3.3 Lookup Textures 123

Conclusion 125

References 126

Other Resources 126

Chapter 7 Implementing BRDFs in Shaders 127

Introduction. 127

7.1 Basic Setup and Diffuse Materials 128

7.1.1 Basic Application Code 128

7.1.2 Basic Diffuse Material 130

7.2 Specular Materials 132

7.2.1 The Phong Shaders. 132

7.2.2 The Blinn-Phong Shaders 136

7.3 Oren-Nayar Materials. 138

7.4 Minnaert Materials 143

7.5 Ward Materials 145

7.5.1 Isotropic Ward Materials 145

7.5.2 Anisotropic Ward Materials 149

7.6 Schlick Materials 152

7.7 Cook-Torrance Materials 154

Conclusion 156

References 156

Chapter 8 Spherical Harmonic Lighting 157

Introduction. 157

8.1 Understanding the Need for Spherical Harmonics. 158

8.1.1 Hemispheres of Light 158

8.1.2 Representations of Light 160

8.1.3 Compressing Data Signals	160
8.1.4 Compressing Hemispheres of Light	163
8.2 Spherical Harmonics Theory	164
8.2.1 Definition	164
8.2.2 Projection and Reconstruction	167
8.2.3 Main Properties	170
8.2.4 The Spherical Harmonic Lighting Technique	171
8.2.4.1 The Rendering Equation	171
8.2.4.2 SH Diffuse Lighting	173
8.2.4.3 SH Diffuse Shadowed Lighting	175
8.2.4.4 SH Diffuse Shadowed Inter-Reflected Lighting	177
8.3 Sample Implementations in OpenGL.	179
8.3.1 Introduction.	179
8.3.2 Associated Legendre Polynomials 2D Display.	180
8.3.2.1 Design	180
8.3.2.2 Implementation	181
8.3.2.3 Command-line Parameters	182
8.3.2.4 Results.	182
8.3.3 Spherical Harmonics 3D Display.	183
8.3.3.1 Design	183
8.3.3.2 Implementation	185
8.3.3.3 Command-line Parameters	186
8.3.3.4 Keyboard Mapping and Mouse Usage	186
8.3.3.5 Results.	187
8.3.4 Function Approximation and Reconstruction Using Spherical Harmonics	187
8.3.4.1 Design	187
8.3.4.2 Implementation	189
8.3.4.3 Command-line Parameters	192
8.3.4.4 Keyboard Mapping and Mouse Usage	192
8.3.4.5 Results.	193
8.3.5 HDR Images Loading and Display	194
8.3.5.1 Design	194
8.3.5.2 Implementation	196

8.3.5.3 Command-line Parameters	196
8.3.5.4 Results.	196
8.3.6 Spherical Harmonic Lighting Program.	197
8.3.6.1 Design	197
8.3.6.2 Implementation	201
8.3.6.3 Command-line Parameters	205
8.3.6.4 Keyboard Mapping and Mouse Usage	206
8.3.6.5 Results.	207
Conclusion and Further Reading	209
References	209
Chapter 9 Spherical Harmonics in DirectX.	211
Introduction.	211
9.1 Per-Vertex SH Data Generation with D3DX	212
9.1.1 The Main SH Simulator	212
9.1.2 Parameters and Performance Implications.	213
9.1.2.1 Vertex Count	213
9.1.2.2 Ray Count	214
9.1.2.3 Bounce Count	215
9.1.2.4 Order	215
9.1.3 Compressed SH Coefficients.	216
9.1.3.1 Generating Compressed SH Coefficients.	216
9.1.3.2 Using Compressed SH Coefficients.	219
9.2 Rendering the Per-Vertex SH Solution.	222
9.2.1 Encoding Lights for SH Rendering.	222
9.2.2 The Basic SH Vertex Shader.	225
9.3 SH with Cube Maps.	228
9.4 DX SH with HDRI	230
9.5 Multiple Meshes/Materials.	232
9.6 Subsurface Scattering.	235
9.7 Simple Specular Highlights.	237
9.7.1 The Basic Idea	237
9.7.2 The Implementation	240

Conclusion	242
References	243
Chapter 10 Toward Real-Time Radiosity	245
Introduction	245
10.1 Radiosity Background and Theory	246
10.1.1 What Radiosity Tries to Achieve	246
10.1.2 Historical Background and Evolution	246
10.1.3 Near Real-Time Radiosity	247
10.2 Radiosity Theory and Methods	248
10.2.1 Definitions	248
10.2.2 The Radiosity Equation	251
10.2.3 Form Factors	253
10.2.3.1 Properties	253
10.2.3.2 Determination	254
10.2.4 The Classic Radiosity Method	256
10.2.5 The Progressive Refinement Method	259
10.2.6 Radiosity and Subdivision	261
10.3 Sample Implementation in OpenGL	263
Conclusion	264
References	265
Other Resources	265
Appendix A Building the Source Code	267
Introduction	267
A.1 DirectX/HLSL Programs	267
A.1.1 Requirements	267
A.1.2 Building the Programs	268
A.1.3 Running and Testing the Programs and Shaders	270
A.2 OpenGL/CG Programs	271
A.2.1 Requirements	271
A.2.2 Building the Programs	271
A.2.2.1 Windows Platforms	271

A.2.2.2 Linux Platforms 273

A.2.3 Running and Testing the Programs and Shaders 274

A.3 OpenGL/GLSL Programs 274

 A.3.1 Requirements 274

 A.3.2 Building the Programs 275

 A.3.3 Running and Testing the Programs and Shaders 277

A.4 OpenGL Programs 278

 A.4.1 Platforms and Tools 278

 A.4.2 Installing the Libraries 278

 A.4.2.1 GLUT 278

 A.4.2.2 Lib3ds 278

 A.4.3 Building the Programs 280

 A.4.3.1 Unix Platforms 280

 A.4.3.2 Windows Platforms 280

References 281

Other Resources 281

Appendix B Sample Raytracer Implementation 283

Introduction 283

B.1 Design 283

 B.1.1 Introduction 284

 B.1.2 Data Types 284

 B.1.3 Main Functions and Program Flow 286

 B.1.4 Input File Format Specification 287

 B.1.4.1 Basic Data Types 287

 B.1.4.2 Primitives 289

B.2 Implementation. 295

 B.2.1 Scene Parser Overview 295

 B.2.2 Core Raytracing Functions. 295

 B.2.3 Ray/Primitive Intersection Functions 302

 B.2.4 File List and Auxiliary Functions 303

B.3 The Raytracing Program 305

 B.3.1 Renderings 305

B.3.2 Extending the Raytracer	306
Conclusion	307
References	307
Appendix C The Lighting and Shading Frameworks	309
Introduction	309
C.1 DirectX/HLSL Framework	310
C.1.1 Requirements	310
C.1.2 Design	310
C.1.2.1 Introduction	310
C.1.2.2 User Interface.	311
C.1.2.3 Data Structures and Instantiation.	311
C.2 OpenGL/Cg Framework	315
C.2.1 Requirements	315
C.2.2 Design	315
C.2.2.1 Introduction	315
C.2.2.2 User Interface.	316
C.2.2.3 Data Structures	316
C.2.2.4 Functions and Files	317
C.2.2.5 Program Flow.	319
C.2.3 Results	320
C.2.3.1 OpenGL Shading	320
C.2.3.2 Simple Shader.	320
C.2.3.3 Advanced Shader	321
C.3 OpenGL/GLSL Framework	322
C.3.1 Requirements	322
C.3.2 Design	322
C.3.2.1 Introduction	322
C.3.2.2 User Interface.	323
C.3.2.3 Data Structures	323
C.3.2.4 Functions and Files	324
C.3.2.5 Program Flow.	325

C.3.3 Results 326

 C.3.3.1 OpenGL Shading 326

 C.3.3.2 Simple Shader. 326

References 327

Other Resources 328

Index 329

This page intentionally left blank.

Foreword

The description of surface appearance is at the core of image synthesis, and creating convincing images with the computer remains to date a complex process, requiring advanced specialized skills. In reality, appearance is controlled by the complex interplay of light (electromagnetic radiation coming from various sources) and the materials composing the objects in the scene, with complex scattering effects taking place all along the path of light.

The physical laws governing this process have long been known at the scale of the light wavelength; yet in order to create a computational model that can be used by a computer, a discrete model is required, and the vast body of research conducted in computer graphics in the last three decades has aimed at providing simplified models for the calculation of lighting effects.

Nowadays computers are equipped with powerful graphics units specialized in massively parallel computations at the vertex or pixel level. The complex calculations needed for proper appearance rendering therefore become affordable in real-time applications, provided they can be incorporated in small programs called shaders. This book therefore provides

a very timely contribution by demonstrating in detail how to create these shaders for advanced lighting.

The authors have chosen an ambitious and rigorous path by first presenting the physics and detailed lighting equations. Their discussion of lighting effects, advanced rendering algorithms such as raytracing and radiosity, and material descriptions will be useful to anyone first approaching computer graphics applications.

The remainder of the book provides ample detail on how to incorporate the most advanced lighting effects in shaders. The authors went as far as covering recent advances such as “precomputed radiance transfer” techniques, in which the objects can be illuminated in real time from all directions.

The combination of the theoretical presentation of the underlying concepts and the practical implementation methods for direct usage make this book a complete and self-contained manual. I very much enjoyed reading it and I am certain that it will prove a valuable resource for a great many professionals in the gaming or special effects industry.

Francois Sillion
Grenoble, France

Acknowledgments

Every book of this kind includes the knowledge and expertise of many people, both directly and indirectly involved with the book itself. First, we are indebted to the people who developed the theory and techniques that are described here. Their work is referenced at the end of each chapter and we encourage readers to use those sources to explore further. Of the people who worked directly with the book, we would like to thank Jason Mitchell, Scott Thompson, Wes Beckwith, and Jim Hill. Their help and suggestions were excellent.

We would also like to thank our families and colleagues. Their support is invaluable when working on a book like this.

Introduction

The Importance of Light

When you take a careful look at the real world around you, you begin to realize that most of the detail you see is the result of subtle lighting effects. Areas of light and shadow define the furrowed bark of trees. An elderly face can appear wiser simply because of the depth of its creases. Before a storm, ordinary sunlight takes on an ominous tone, and objects appear less defined. The same room can look very different when it is lit by a television or by candlelight. Each of these examples is an instance where your perception of reality is affected by very subtle changes in the way light interacts with the objects around you. Your eyes and brain are tuned to look for those subtle cues. Therefore, if your goal is to recreate reality (or to create a new reality), you must include as many of those effects as possible.

However, most game development books dramatically understate the importance of lighting and the interactions between light and physical materials. This is largely the result of the fact that, until very recently, consumer graphics

hardware was not capable of rendering complex lighting effects in real time. This is changing, and game programmers are finding that they have the power to render complex effects that more closely approximate reality. In the past, the ability to render complex scenes was limited to the raw performance of the hardware. Performance will always be a concern, but complexity is now also dependent on the ability to use the capabilities of the hardware effectively.

With the correct algorithms and approaches, graphics hardware is capable of reproducing the subtle differences between elements of the virtual environment. These subtleties are the key to realism. Now characters need not be limited to simple textured models. They can be comprised of complex materials such as matte cloth, gleaming shields, and smooth skin. This level of realism is now possible, but it requires a deeper understanding of the underlying theory of how light interacts with matter.