# Contents

## Preface  xix

## 1 A Survey of Computer Graphics  2

1-1 Graphs and Charts  3
1-2 Computer-Aided Design  5
1-3 Virtual-Reality Environments  10
1-4 Data Visualizations  12
1-5 Education and Training  19
1-6 Computer Art  23
1-7 Entertainment  28
1-8 Image Processing  31
1-9 Graphical User Interfaces  32
1-10 Summary  33

## 2 Overview of Graphics Systems  34

2-1 Video Display Devices  35
• Refresh Cathode-Ray Tubes  36
• Raster-Scan Displays  39
• Random-Scan Displays  41
• Color CRT Monitors  42
• Flat-Panel Displays  44
• Three-Dimensional Viewing Devices  47
• Stereoscopic and Virtual-Reality Systems  48

2-2 Raster-Scan Systems  50
• Video Controller  50
• Raster-Scan Display Processor  52

2-3 Graphics Workstations and Viewing Systems  54

2-4 Input Devices  57
• Keyboards, Button Boxes, and Dials  57
• Mouse Devices  57
• Trackballs and Spaceballs  59
• Joysticks  59
• Data Gloves  60
• Digitizers  60
• Image Scanners  62
• Touch Panels  64
• Light Pens  65
• Voice Systems  65

2-5 Hard-Copy Devices  66
2-6 Graphics Networks  68
2-7 Graphics on the Internet  68
2-8 Graphics Software  69
• Coordinate Representations  69
• Graphics Functions  71
• Software Standards  72
• Other Graphics Packages  73

2-9 Introduction to OpenGL  73
• Basic OpenGL Syntax  74
• Related Libraries  74
• Header Files  75
• Display-Window Management Using GLUT  76
• A Complete OpenGL Program  77

## Contents

### 2-10 Summary 81
- References 82
- Exercises 82

### 3 Graphics Output Primitives 84

#### 3-1 Coordinate Reference Frames 86
- Screen Coordinates 86
- Absolute and Relative Coordinate Specifications 87

#### 3-2 Specifying a Two-Dimensional World-Coordinate Reference Frame in OpenGL 87

#### 3-3 OpenGL Point Functions 88

#### 3-4 OpenGL Line Functions 91

#### 3-5 Line-Drawing Algorithms 92
- Line Equations 93
- DDA Algorithm 94
- Bresenham’s Line Algorithm 95
- Displaying Polylines 99

#### 3-6 Parallel Line Algorithms 99

#### 3-7 Setting Frame-Buffer Values 101

#### 3-8 OpenGL Curve Functions 102

#### 3-9 Circle-Generating Algorithms 103
- Properties of Circles 103
- Midpoint Circle Algorithm 105

#### 3-10 Ellipse-Generating Algorithms 109
- Properties of Ellipses 109
- Midpoint Ellipse Algorithm 110

#### 3-11 Other Curves 117
- Conic Sections 117
- Polynomials and Spline Curves 119

#### 3-12 Parallel Curve Algorithms 119

#### 3-13 Pixel Addressing and Object Geometry 120
- Screen Grid Coordinates 120
- Maintaining Geometric Properties of Displayed Objects 121

#### 3-14 Fill-Area Primitives 123

#### 3-15 Polygon Fill Areas 124
- Polygon Classifications 124
- Identifying Concave Polygons 125
- Splitting Concave Polygons 125

### 3-16 OpenGL Polygon Fill-Area Functions 134

### 3-17 OpenGL Vertex Arrays 139

### 3-18 Pixel-Array Primitives 142

### 3-19 OpenGL Pixel-Array Functions 143

### 3-20 Character Primitives 147

### 3-21 OpenGL Character Functions 149

### 3-22 Picture Partitioning 151

### 3-23 OpenGL Display Lists 151

### 3-24 OpenGL Display-Window Reshape Function 154

### 3-25 Summary 157

### 3-26 Example Programs 160

### 3-27 References 168

### 3-28 Exercises 168

## 4 Attributes of Graphics Primitives 172

#### 4-1 OpenGL State Variables 174
- Color and Gray Scale 174

#### 4-2 RGB Color Components 174
- Color Tables 175
- Gray Scale 176
- Other Color Parameters 176

#### 4-3 OpenGL Color Functions 177
- The OpenGL RGB and RGBA Color Modes 177
- OpenGL Color-Index Mode 178
Contents ix

4-4 Point Attributes 183
4-5 Line Attributes 183
• Line Width 183
• Line Style 185
• Pen and Brush Options 186
4-6 Curve Attributes 187
4-7 OpenGL Point-Attribute Functions 189
4-8 OpenGL Line-Attribute Functions 190
• OpenGL Line-Width Function 190
• OpenGL Line-Style Function 190
• Other OpenGL Line Effects 192
4-9 Fill-Area Attributes 193
• Fill Styles 193
• Color-Blended Fill Regions 194
4-10 General Scan-Line Polygon-Fill Algorithm 196
4-11 Scan-Line Fill of Convex Polygons 200
4-12 Scan-Line Fill for Regions with Curved Boundaries 201
4-13 Fill Methods for Areas with Irregular Boundaries 201
• Boundary-Fill Algorithm 201
• Flood-Fill Algorithm 205
4-14 OpenGL Fill-Area Attribute Functions 205
• OpenGL Fill-Pattern Function 206
• OpenGL Texture and Interpolation Patterns 207
• OpenGL Wire-Frame Methods 207
• OpenGL Front-Face Function 210
4-15 Character Attributes 211
4-16 OpenGL Character-Attribute Functions 213
4-17 Antialiasing 214
• Supersampling Straight-Line Segments 215
• Subpixel Weighting Masks 217
• Area Sampling Straight Line Segments 217
• Filtering Techniques 217
• Pixel Phasing 218
• Compensating for Line-Intensity Differences 218
4-18 OpenGL Antialiasing Functions 221
4-19 OpenGL Query Functions 222
4-20 OpenGL Attribute Groups 223
4-21 Summary 224
References 226
Exercises 227

5 Geometric Transformations 230
5-1 Basic Two-Dimensional Geometric Transformations 232
• Two-Dimensional Translation 232
• Two-Dimensional Rotation 234
• Two-Dimensional Scaling 236
5-2 Matrix Representations and Homogeneous Coordinates 237
• Homogeneous Coordinates 238
• Two-Dimensional Translation Matrix 239
• Two-Dimensional Rotation Matrix 239
• Two-Dimensional Scaling Matrix 239
5-3 Inverse Transformations 240
5-4 Two-Dimensional Composite Transformations 241
• Composite Two-Dimensional Translations 241
• Composite Two-Dimensional Rotations 241
• Composite Two-Dimensional Scalings 242
• General Two-Dimensional Pivot-Point Rotation 242
• General Two-Dimensional Fixed-Point Scaling 243
• General Two-Dimensional Scaling Directions 244
• Matrix Concatenation Properties 244
• General Two-Dimensional Composite Transformations and Computational Efficiency 245
## Contents

- Two-Dimensional Rigid-Body Transformation 247
- Constructing Two-Dimensional Rotation Matrices 248
- Two-Dimensional Composite-Matrix Programming Example 248

5-5 Other Two-Dimensional Transformations 253
  - Reflection 253
  - Shear 255

5-6 Raster Methods for Geometric Transformations 257

5-7 OpenGL Raster Transformations 258

5-8 Transformations Between Two-Dimensional Coordinate Systems 259

5-9 Geometric Transformations in Three-Dimensional Space 261

5-10 Three-Dimensional Translation 262

5-11 Three-Dimensional Rotation
  - Three-Dimensional Coordinate-Axis Rotations 264
  - General Three-Dimensional Rotations 266
  - Quaternion Methods for Three-Dimensional Rotations 272

5-12 Three-Dimensional Scaling 275

5-13 Composite Three-Dimensional Transformations 278

5-14 Other Three-Dimensional Transformations
  - Three-Dimensional Reflections 281
  - Three-Dimensional Shears 281

5-15 Transformations Between Three-Dimensional Coordinate Systems 282

5-16 Affine Transformations 283

5-17 OpenGL Geometric-Transformation Functions 283
  - Basic OpenGL Geometric Transformations 284
  - OpenGL Matrix Operations 285
  - OpenGL Matrix Stacks 287
  - OpenGL Geometric-Transformation Programming Examples 288

5-18 Summary 291

<table>
<thead>
<tr>
<th>References</th>
<th>Exercises</th>
</tr>
</thead>
<tbody>
<tr>
<td>293</td>
<td>293</td>
</tr>
</tbody>
</table>

## 6 Two-Dimensional Viewing 296

6-1 The Two-Dimensional Viewing Pipeline 297

6-2 The Clipping Window 299
  - Viewing-Coordinate Clipping Window 300
  - World-Coordinate Clipping Window 300

6-3 Normalization and Viewport Transformations 301
  - Mapping the Clipping Window into a Normalized Viewport 301
  - Mapping the Clipping Window into a Normalized Square 303
  - Display of Character Strings 305
  - Split-Screen Effects and Multiple Output Devices 305

6-4 OpenGL Two-Dimensional Viewing Functions 305
  - OpenGL Projection Mode 306
  - GLU Clipping-Window Function 306
  - OpenGL Viewport Function 307
  - Creating a GLUT Display Window 307
  - Setting the GLUT Display-Window Mode and Color 308
  - GLU Display-Window Identifier 309
  - Deleting a GLUT Display Window 309
  - Current GLUT Display Window 309
  - Relocating and Resizing a GLUT Display Window 309
  - Managing Multiple GLUT Display Windows 310
  - GLUT Subwindows 311
  - Selecting a Display-Window Screen Cursor Shape 311
  - Viewing Graphics Objects in a GLUT Display Window 311
  - Executing the Application Program 312
  - Other GLUT Functions 312
  - OpenGL Two-Dimensional Viewing Program Example 313
6-5 Clipping Algorithms 315
6-6 Two-Dimensional Point Clipping 315
6-7 Two-Dimensional Line Clipping 316
   • Cohen-Sutherland Line Clipping 317
   • Liang-Barisky Line Clipping 322
   • Nichol-Lee-Nichol Line Clipping 325
   • Line Clipping Using Nonrectangular Polygon Clip Windows 328
   • Line Clipping Using Nonlinear Clipping-Window Boundaries 329
6-8 Polygon Fill-Area Clipping 329
   • Sutherland-Hodgman Polygon Clipping 331
   • Weiler-Atherton Polygon Clipping 335
   • Polygon Clipping Using Nonrectangular Polygon Clip Windows 337
   • Polygon Clipping Using Nonlinear Clipping-Window Boundaries 338
6-9 Curve Clipping 338
6-10 Text Clipping 339
6-11 Summary 340
References 342
Exercises 343

7 Three-Dimensional Viewing 344
7-1 Overview of Three-Dimensional Viewing Concepts 345
   • Viewing a Three-Dimensional Scene 345
   • Projections 346
   • Depth Cueing 346
   • Identifying Visible Lines and Surfaces 347
   • Surface Rendering 348
   • Exploded and Cutaway Views 348
   • Three-Dimensional and Stereoscopic Viewing 348
7-2 The Three-Dimensional Viewing Pipeline 348
7-3 Three-Dimensional Viewing-Coordinate Parameters 351
   • The View-Plane Normal Vector 351
7-4 Transformation from World to Viewing Coordinates 355
7-5 Projection Transformations 356
7-6 Orthogonal Projections 357
   • Axonometric and Isometric Orthogonal Projections 358
   • Orthogonal Projection Coordinates 358
   • Clipping Window and Orthogonal-Projection View Volume 359
   • Normalization Transformation for an Orthogonal Projection 360
7-7 Oblique Parallel Projections 362
   • Oblique Parallel Projections in Drafting and Design 362
   • Cavalier and Cabinet Oblique Parallel Projections 364
   • Oblique Parallel-Projection Vector 364
   • Clipping Window and Oblique Parallel-Projection View Volume 366
   • Oblique Parallel-Projection Transformation Matrix 366
   • Normalization Transformation for an Oblique Parallel Projection 367
7-8 Perspective Projections 368
   • Perspective-Projection Transformation Coordinates 368
   • Perspective-Projection Equations: Special Cases 369
   • Vanishing Points for Perspective Projections 370
   • Perspective-Projection View Volume 371
   • Perspective-Projection Transformation Matrix 373
   • Symmetric Perspective-Projection Frustum 374
   • Oblique Perspective-Projection Frustum 378
   • Normalized Perspective-Projection Transformation Coordinates 380
7-9 The Viewport Transformation and Three-Dimensional Screen Coordinates 382
Contents

7-10 OpenGL Three-Dimensional Viewing Functions 383
  • OpenGL Viewing-Transformation Function 383
  • OpenGL Orthogonal-Projection Function 384
  • OpenGL Symmetric Perspective-Projection Function 386
  • OpenGL General Perspective-Projection Function 386
  • OpenGL Viewports and Display Windows 387
  • OpenGL Three-Dimensional Viewing Program Example 387

7-11 Three-Dimensional Clipping Algorithms 389
  • Clipping in Three-Dimensional Homogeneous Coordinates 389
  • Three-Dimensional Region Codes 390
  • Three-Dimensional Point and Line Clipping 391
  • Three-Dimensional Polygon Clipping 394
  • Three-Dimensional Curve Clipping 395
  • Arbitrary Clipping Planes 395

7-12 OpenGL Optional Clipping Planes 397

7-13 Summary 398

References 400
Exercises 400

8 Three-Dimensional Object Representations 402

8-1 Polyhedra 404

8-2 OpenGL Polyhedron Functions 404
  • OpenGL Polygon Fill-Area Functions 404
  • GLUT Regular-Polyhedron Functions 404
  • Example GLUT Polyhedron Program 406

8-3 Curved Surfaces 408

8-4 Quadric Surfaces 408
  • Sphere 408
  • Ellipsoid 408
  • Torus 409

8-5 Superquadrics 410
  • Superellipses 410
  • Superellipsoids 411

8-6 OpenGL Quadric-Surface and Cubic-Surface Functions 411
  • GLUT Quadric-Surface Functions 412
  • GLUT Cubic-Surface Teapot Function 413
  • GLU Quadric-Surface Functions 414
  • Example Program Using GLUT and GLU Quadric-Surface Functions 416

8-7 Blobby Objects 418

8-8 Spline Representations 420
  • Interpolation and Approximation Splines 420
  • Parametric Continuity Conditions 421
  • Geometric Continuity Conditions 422
  • Spline Specifications 423
  • Spline Surfaces 424
  • Trimming Spline Surfaces 424

8-9 Cubic-Spline Interpolation Methods 425
  • Natural Cubic Splines 426
  • Hermite Interpolation 426
  • Cardinal Splines 429
  • Kochanek-Bartels Splines 431

8-10 Bézier Spline Curves 432
  • Bézier Curve Equations 433
  • Example Bézier Curve-Generating Program 435
  • Properties of Bézier Curves 437
  • Design Techniques Using Bézier Curves 438
  • Cubic Bézier Curves 439
  • Bézier Surfaces 441
  • B-Spline Curves 442
  • Uniform Periodic B-Spline Curves 443
  • Cubic Periodic B-Spline Curves 444
  • Open Uniform B-Spline Curves 448
  • Nonuniform B-Spline Curves 451
  • B-Spline Surfaces 452
  • Beta Splines 452
  • Beta-Spline Continuity Conditions 452
Contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8-15</td>
<td>Cubic Periodic Beta-Spline Matrix Representation</td>
<td>453</td>
</tr>
<tr>
<td>8-16</td>
<td>Conversion Between Spline Representations</td>
<td>456</td>
</tr>
<tr>
<td>8-17</td>
<td>Displaying Spline Curves and Surfaces</td>
<td>457</td>
</tr>
<tr>
<td>8-18</td>
<td>OpenGL Approximation-Spline Functions</td>
<td>461</td>
</tr>
<tr>
<td>8-19</td>
<td>Sweep Representations</td>
<td>473</td>
</tr>
<tr>
<td>8-20</td>
<td>Constructive Solid-Geometry Methods</td>
<td>474</td>
</tr>
<tr>
<td>8-21</td>
<td>Octrees</td>
<td>476</td>
</tr>
<tr>
<td>8-22</td>
<td>BSP Trees</td>
<td>479</td>
</tr>
<tr>
<td>8-23</td>
<td>Fractal-Geometry Methods</td>
<td>479</td>
</tr>
<tr>
<td>8-24</td>
<td>Shape Grammars and Other Procedural Methods</td>
<td>507</td>
</tr>
<tr>
<td>8-25</td>
<td>Particle Systems</td>
<td>510</td>
</tr>
<tr>
<td>8-26</td>
<td>Physically Based Modeling</td>
<td>511</td>
</tr>
<tr>
<td>8-27</td>
<td>Visualization of Data Sets</td>
<td>514</td>
</tr>
<tr>
<td>8-28</td>
<td>Summary</td>
<td>521</td>
</tr>
</tbody>
</table>

Visual Representations for Vector Fields | 517
Visual Representations for Tensor Fields | 519
Visual Representations for Multivariate Data Fields | 520

References | 524
Exercises | 525

9 Visible-Surface Detection Methods | 528

9-1 Classification of Visible-Surface Detection Algorithms | 529
9-2 Back-Face Detection | 530
9-3 Depth-Buffer Method | 531
9-4 A-Buffer Method | 534
9-5 Scan-Line Method | 535
9-6 Depth-Sorting Method | 537
9-7 BSP-Tree Method | 540
9-8 Area-Subdivision Method | 541
9-9 Octree Methods | 543
9-10 Ray-Casting Method | 544
9-11 Comparison of Visibility-Detection Methods | 545
9-12 Curved Surfaces | 545
Curved-Surface Representations | 546
Surface Contour Plots | 546
9-13 Wire-Frame Visibility Methods | 547
Wire-Frame Surface-Visibility Algorithms | 547
Wire-Frame Depth-Cueing Algorithm | 548
OpenGL Visible-Visible-Detection Functions | 549
OpenGL Polygon-Culling Functions | 549
OpenGL Depth-Buffer Functions | 549
OpenGL Wire-Frame Surface-Visibility Methods | 551
OpenGL Depth-Cueing Function | 552
Summary | 552
References | 553
Exercises | 554

# 10 Illumination Models and Surface-Rendering Methods

## 10-1 Light Sources
- **Point Light Sources**
- **Infinitely Distant Light Sources**
- **Radial Intensity Attenuation**
- **Directional Light Sources and Spotlight Effects**
- **Angular Intensity Attenuation**
- **Extended Light Sources and the Warn Model**

## 10-2 Surface Lighting Effects

## 10-3 Basic Illumination Models
- **Ambient Light**
- **Diffuse Reflection**
- **Specular Reflection and the Phong Model**
- **Combined Diffuse and Specular Reflections**
- **Diffuse and Specular Reflections from Multiple Light Sources**
- **Surface Light Emissions**
- **Basic Illumination Model with Intensity Attenuation and Spotlights**
- **RGB Color Considerations**
- **Other Color Representations**
- **Luminance**

## 10-4 Transparent Surfaces
- **Translucent Materials**
- **Light Refraction**
- **Basic Transparency Model**

## 10-5 Atmospheric Effects

## 10-6 Shadows

## 10-7 Camera Parameters

## 10-8 Displaying Light Intensities
- **Distributing System Intensity Levels**
- **Gamma Correction and Video Lookup Tables**
- **Displaying Continuous-Tone Images**

## 10-9 Halftone Patterns and Dithering Techniques
- **Halftone Approximations**
- **Dithering Techniques**

## 10-10 Polygon-Rendering Methods
- **Constant-Intensity Surface Rendering**
- **Gouraud Surface Rendering**
- **Phong Surface Rendering**
- **Fast Phong Surface Rendering**

## 10-11 Ray-Tracing Methods
- **Basic Ray-Tracing Algorithm**
- **Ray-Surface Intersection Calculations**
- **Ray-Sphere Intersections**
- **Ray-Polyhedron Intersections**
- **Reducing Object-Intersection Calculations**
- **Space-Subdivision Methods**
- **Simulating Camera Focusing Effects**
- **Antialiased Ray Tracing**
- **Distributed Ray Tracing**

## 10-12 Radiosity Lighting Model
- **Radiant-Energy Terms**
- **The Basic Radiosity Model**
- **Progressive Refinement Radiosity Method**
- **Environment Mapping**
- **Photon Mapping**
- **Adding Surface Detail**
- **Modeling Surface Detail with Polygons**
- **Texture Mapping**
- **Linear Texture Patterns**
- **Surface Texture Patterns**
- **Volume Texture Patterns**
- **Texture Reduction Patterns**
- **Procedural Texturing Methods**

## 10-13 Environment Mapping

## 10-14 Photon Mapping

## 10-15 Adding Surface Detail

## 10-16 Modeling Surface Detail with Polygons

## 10-17 Texture Mapping

## 10-18 Bump Mapping

## 10-19 Frame Mapping

## 10-20 OpenGL Illumination and Surface-Rendering Functions
- **OpenGL Point Light-Source Function**
- **Specifying an OpenGL Light-Source Position and Type**
- **Specifying OpenGL Light-Source Colors**
- **Specifying Radial-Intensity Attenuation Coefficients for an OpenGL Light Source**
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OpenGL Directional Light Sources (Spotlights)</td>
<td>640</td>
</tr>
<tr>
<td>OpenGL Global Lighting Parameters</td>
<td>641</td>
</tr>
<tr>
<td>OpenGL Surface-Property Function</td>
<td>642</td>
</tr>
<tr>
<td>OpenGL Illumination Model</td>
<td>643</td>
</tr>
<tr>
<td>OpenGL Atmospheric Effects</td>
<td>644</td>
</tr>
<tr>
<td>OpenGL Transparency Functions</td>
<td>645</td>
</tr>
<tr>
<td>OpenGL Surface-Rendering Functions</td>
<td>646</td>
</tr>
<tr>
<td>OpenGL Halftoning Operations</td>
<td>647</td>
</tr>
<tr>
<td>10-21 OpenGL Texture Functions</td>
<td>648</td>
</tr>
<tr>
<td>OpenGL Line-Texture Functions</td>
<td>648</td>
</tr>
<tr>
<td>OpenGL Surface-Texture Functions</td>
<td>651</td>
</tr>
<tr>
<td>OpenGL Volume-Texture Functions</td>
<td>653</td>
</tr>
<tr>
<td>OpenGL Color Options for Texture Patterns</td>
<td>653</td>
</tr>
<tr>
<td>OpenGL Texture-Mapping Options</td>
<td>654</td>
</tr>
<tr>
<td>OpenGL Texture Wrapping</td>
<td>655</td>
</tr>
<tr>
<td>Copying OpenGL Texture Patterns from the Frame Buffer</td>
<td>655</td>
</tr>
<tr>
<td>OpenGL Texture-Coordinate Arrays</td>
<td>655</td>
</tr>
<tr>
<td>Naming OpenGL Texture Patterns</td>
<td>656</td>
</tr>
<tr>
<td>OpenGL Texture Subpatterns</td>
<td>657</td>
</tr>
<tr>
<td>OpenGL Texture Reduction Patterns</td>
<td>657</td>
</tr>
<tr>
<td>OpenGL Texture Borders</td>
<td>658</td>
</tr>
<tr>
<td>OpenGL Proxy Textures</td>
<td>659</td>
</tr>
<tr>
<td>Automatic Texturing of Quadric Surfaces</td>
<td>659</td>
</tr>
<tr>
<td>Homogeneous Texture Coordinates</td>
<td>659</td>
</tr>
<tr>
<td>Additional OpenGL Texture Options</td>
<td>660</td>
</tr>
<tr>
<td>10-22 Summary</td>
<td>660</td>
</tr>
<tr>
<td>References</td>
<td>664</td>
</tr>
<tr>
<td>Exercises</td>
<td>665</td>
</tr>
<tr>
<td>11 Interactive Input Methods and Graphical User Interfaces</td>
<td>668</td>
</tr>
<tr>
<td>11-1 Graphical Input Data</td>
<td>669</td>
</tr>
<tr>
<td>11-2 Logical Classification of Input Devices</td>
<td>669</td>
</tr>
<tr>
<td>Locator Devices</td>
<td>670</td>
</tr>
<tr>
<td>Stroke Devices</td>
<td>670</td>
</tr>
<tr>
<td>String Devices</td>
<td>670</td>
</tr>
<tr>
<td>11-3 Input Functions for Graphical Data</td>
<td>673</td>
</tr>
<tr>
<td>11-4 Interactive Picture-Construction Techniques</td>
<td>675</td>
</tr>
<tr>
<td>11-5 Virtual-Reality Environments</td>
<td>679</td>
</tr>
<tr>
<td>11-6 OpenGL Interactive Input-Device Functions</td>
<td>679</td>
</tr>
<tr>
<td>GLUT Mouse Functions</td>
<td>680</td>
</tr>
<tr>
<td>GLUT Keyboard Functions</td>
<td>684</td>
</tr>
<tr>
<td>GLUT Tablet Functions</td>
<td>689</td>
</tr>
<tr>
<td>GLUT Spaceball Functions</td>
<td>689</td>
</tr>
<tr>
<td>GLUT Button-Box Function</td>
<td>690</td>
</tr>
<tr>
<td>GLUT Dials Function</td>
<td>690</td>
</tr>
<tr>
<td>OpenGL Picking Operations</td>
<td>690</td>
</tr>
<tr>
<td>11-7 OpenGL Menu Functions</td>
<td>696</td>
</tr>
<tr>
<td>Creating a GLUT Menu</td>
<td>696</td>
</tr>
<tr>
<td>Creating and Managing Multiple GLUT Menus</td>
<td>699</td>
</tr>
<tr>
<td>Creating GLUT Submenus</td>
<td>700</td>
</tr>
<tr>
<td>Modifying GLUT Menus</td>
<td>703</td>
</tr>
<tr>
<td>11-8 Designing a Graphical User Interface</td>
<td>703</td>
</tr>
<tr>
<td>The User Dialogue</td>
<td>703</td>
</tr>
<tr>
<td>Windows and Icons</td>
<td>704</td>
</tr>
<tr>
<td>Accommodating Multiple Skill Levels</td>
<td>704</td>
</tr>
<tr>
<td>Consistency</td>
<td>705</td>
</tr>
<tr>
<td>Minimizing Memorization</td>
<td>705</td>
</tr>
<tr>
<td>Backup and Error Handling</td>
<td>705</td>
</tr>
<tr>
<td>Feedback</td>
<td>705</td>
</tr>
<tr>
<td>11-9 Summary</td>
<td>706</td>
</tr>
<tr>
<td>References</td>
<td>709</td>
</tr>
<tr>
<td>Exercises</td>
<td>709</td>
</tr>
</tbody>
</table>
12 Color Models and Color Applications 712

12-1 Properties of Light 713
  • The Electromagnetic Spectrum 713
  • Psychological Characteristics of Color 715

12-2 Color Models 716
  • Primary Colors 716
  • Intuitive Color Concepts 716

12-3 Standard Primaries and the Chromaticity Diagram 717
  • The XYZ Color Model 717
  • Normalized XYZ Values 718
  • The CIE Chromaticity Diagram 718
  • Color Gamuts 719
  • Complementary Colors 719
  • Dominant Wavelength 719
  • Purity 720

12-4 The RGB Color Model 720

12-5 The YIQ and Related Color Models 722
  • The YIQ Parameters 722
  • Transformations Between RGB and YIQ Color Spaces 722
  • The YUV YC,Cb Systems 723

12-6 The CMY and CMYK Color Models 723
  • The CMY Parameters 723
  • Transformations Between CMY and RGB Color Spaces 724

12-7 The HSV Color Model 724
  • The HSV Parameters 724
  • Selecting Shades, Tints, and Tones 726
  • Transformations Between HSV and RGB Color Spaces 726

12-8 The HLS Color Model 728

12-9 Color Selection and Applications 728

12-10 Summary 730
  References 731
  Exercises 731

13 Computer Animation 732

13-1 Raster Methods for Computer Animation 734

13-2 Design of Animation Sequences 735

13-3 Traditional Animation Techniques 737

13-4 General Computer-Animation Functions 737

13-5 Computer-Animation Languages 739

13-6 Key-Frame Systems 739
  • Morphing 739
  • Simulating Accelerations 742
  • Direct Motion Specification 745
  • Goal-Directed Systems 745
  • Kinematics and Dynamics 746

13-7 Articulated Figure Animation 747
  • Periodic Motions 748
  • OpenGL Animation Procedures 749

13-8 Summary 752
  References 754
  Exercises 754

14 Hierarchical Modeling 756

14-1 Basic Modeling Concepts 757
  • System Representations 757
  • Symbol Hierarchies 759

14-2 Modeling Packages 760

14-3 General Hierarchical Modeling Methods 762
  • Local Coordinates 763
  • Modeling Transformations 763
  • Creating Hierarchical Structures 763

14-4 Hierarchical Modeling Using OpenGL Display Lists 765

14-5 Summary 765
  References 766
  Exercises 766

15 Graphics File Formats 768

15-1 Image-File Configurations 769

15-2 Color-Reduction Methods 770
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Color Reduction</td>
<td>770</td>
</tr>
<tr>
<td>Popularity Color Reduction</td>
<td>771</td>
</tr>
<tr>
<td>Median-Cut Color Reduction</td>
<td>771</td>
</tr>
<tr>
<td>File-Compression Techniques</td>
<td>771</td>
</tr>
<tr>
<td>Run-Length Encoding</td>
<td>772</td>
</tr>
<tr>
<td>LZW Encoding</td>
<td>772</td>
</tr>
<tr>
<td>Other Pattern-Recognition Compression Methods</td>
<td>773</td>
</tr>
<tr>
<td>Huffman Encoding</td>
<td>773</td>
</tr>
<tr>
<td>Arithmetic Encoding</td>
<td>776</td>
</tr>
<tr>
<td>Discrete Cosine Transform</td>
<td>776</td>
</tr>
<tr>
<td>Composition of the Major File Formats</td>
<td>778</td>
</tr>
<tr>
<td>JPEG: Joint Photographic Experts Group</td>
<td>779</td>
</tr>
<tr>
<td>CGM: Computer-Graphics Metafile Format</td>
<td>780</td>
</tr>
<tr>
<td>TIFF: Tag Image-File Format</td>
<td>781</td>
</tr>
<tr>
<td>PNG: Portable Network-Graphics Format</td>
<td>781</td>
</tr>
<tr>
<td>XBM: X Window System Bitmap Format &amp; XPM: X Window System Pixmap Format</td>
<td>781</td>
</tr>
<tr>
<td>Adobe Photoshop Format</td>
<td>782</td>
</tr>
<tr>
<td>MacPaint: Macintosh Paint Format</td>
<td>782</td>
</tr>
<tr>
<td>PICT: Picture Data Format</td>
<td>782</td>
</tr>
<tr>
<td>BMP: Bitmap Format</td>
<td>782</td>
</tr>
<tr>
<td>PCX: PC Paintbrush File Format</td>
<td>783</td>
</tr>
<tr>
<td>TGA: Truevision Graphics-Adapter Format</td>
<td>783</td>
</tr>
<tr>
<td>GIF: Graphics Interchange Format</td>
<td>783</td>
</tr>
<tr>
<td>Summary</td>
<td>783</td>
</tr>
<tr>
<td>References</td>
<td>784</td>
</tr>
<tr>
<td>Exercises</td>
<td>784</td>
</tr>
</tbody>
</table>

## A Mathematics for Computer Graphics | 787

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coordinate Reference Frames</td>
<td>787</td>
</tr>
<tr>
<td>Two-Dimensional Cartesian Screen Coordinates</td>
<td>787</td>
</tr>
<tr>
<td>Standard Two-Dimensional Cartesian Reference Frames</td>
<td>788</td>
</tr>
<tr>
<td>Polar Coordinates in the xy Plane</td>
<td>788</td>
</tr>
<tr>
<td>Standard Three-Dimensional Cartesian Reference Frames</td>
<td>789</td>
</tr>
<tr>
<td>Three-Dimensional Cartesian Screen Coordinates</td>
<td>790</td>
</tr>
<tr>
<td>Three-Dimensional Curvilinear-Coordinate Systems</td>
<td>790</td>
</tr>
<tr>
<td>Solid Angle</td>
<td>791</td>
</tr>
<tr>
<td>Points and Vectors</td>
<td>792</td>
</tr>
<tr>
<td>Point Properties</td>
<td>792</td>
</tr>
<tr>
<td>Vector Properties</td>
<td>792</td>
</tr>
<tr>
<td>Vector Addition and Scalar Multiplication</td>
<td>794</td>
</tr>
<tr>
<td>Scalar Product of Two Vectors</td>
<td>794</td>
</tr>
<tr>
<td>Vector Product of Two Vectors</td>
<td>795</td>
</tr>
<tr>
<td>Basis Vectors and the Metric Tensor</td>
<td>796</td>
</tr>
<tr>
<td>Determining Basis Vectors for a Coordinate Space</td>
<td>796</td>
</tr>
<tr>
<td>Orthonormal Basis</td>
<td>797</td>
</tr>
<tr>
<td>Metric Tensor</td>
<td>798</td>
</tr>
<tr>
<td>Matrices</td>
<td>799</td>
</tr>
<tr>
<td>Scalar Multiplication and Matrix Addition</td>
<td>800</td>
</tr>
<tr>
<td>Matrix Multiplication</td>
<td>800</td>
</tr>
<tr>
<td>Matrix Transpose</td>
<td>801</td>
</tr>
<tr>
<td>Determinant of a Matrix</td>
<td>801</td>
</tr>
<tr>
<td>Matrix Inverse</td>
<td>802</td>
</tr>
<tr>
<td>Complex Numbers</td>
<td>802</td>
</tr>
<tr>
<td>Basic Complex Arithmetic</td>
<td>803</td>
</tr>
<tr>
<td>Imaginary Unit</td>
<td>803</td>
</tr>
<tr>
<td>Complex Conjugate and Modulus of a Complex Number</td>
<td>804</td>
</tr>
<tr>
<td>Complex Division</td>
<td>804</td>
</tr>
<tr>
<td>Polar-Coordinate Representation for a Complex Number</td>
<td>805</td>
</tr>
<tr>
<td>Quaternions</td>
<td>805</td>
</tr>
<tr>
<td>Nonparametric Representations</td>
<td>807</td>
</tr>
<tr>
<td>Parametric Representations</td>
<td>807</td>
</tr>
<tr>
<td>Rate-of-Change Operators</td>
<td>808</td>
</tr>
<tr>
<td>Gradient Operator</td>
<td>809</td>
</tr>
<tr>
<td>Directional Derivative</td>
<td>809</td>
</tr>
<tr>
<td>General Form of the Gradient Operator</td>
<td>810</td>
</tr>
<tr>
<td>Laplace Operator</td>
<td>810</td>
</tr>
<tr>
<td>Divergence Operator</td>
<td>810</td>
</tr>
<tr>
<td>Curl Operator</td>
<td>811</td>
</tr>
</tbody>
</table>
## Contents

A-11 Rate-of-Change Integral Transformation Theorems 811
  • Stokes’s Theorem 811
  • Green’s Theorem for a Plane Surface 812
  • Divergence Theorem 813
  • Green’s Transformation Equations 814

A-12 Area and Centroid of a Polygon 814
  • Area of a Polygon 814
  • Centroid of a Polygon 815

A-13 Calculating Properties of Polyhedra 817

A-14 Numerical Methods 817
  • Solving Sets of Linear Equations 817
  • Finding Roots of Nonlinear Equations 819

• Evaluating Integrals 820
• Solving Ordinary Differential Equations 822
• Solving Partial Differential Equations 824
• Least-Squares Curve-Fitting Methods for Data Sets 825

Bibliography 827

Subject Index 839

OpenGL Function Index 856
  • Core Library Functions 856
  • GLU Functions 857
  • GLUT Functions 857
Computer graphics remains one of the most exciting and rapidly growing areas of modern technology. Since the appearance of the first edition of this book, computer graphics has become a standard feature in applications software and computer systems in general. Computer-graphics methods are routinely applied in the design of most products, in training simulators, in the production of music videos and television commercials, in motion pictures, in data analysis, in scientific studies, in medical procedures, and in numerous other applications. A great variety of techniques and hardware devices are now in use or under development for these diverse application areas. In particular, much of today’s computer-graphics research is concerned with improving the effectiveness, realism, and speed of picture generation. To produce a realistic view of a natural scene, a graphics program must simulate the effects of actual light reflections and refractions from physical objects. Therefore, the current trend in computer graphics is to incorporate improved approximations of physics principles into graphics algorithms, to better simulate the complex interactions between objects and a lighting environment.

Features of the Third Edition

The material in this third edition evolved from notes used in a variety of courses we have taught over the years, including introductory computer graphics, advanced computer graphics, scientific visualization, special topics, and project courses. When we wrote the first edition of this book, many graphics courses and applications dealt only with two-dimensional methods, so we separated the discussions of two-dimensional and three-dimensional graphics techniques. A solid foundation in two-dimensional computer-graphics procedures was given in the first half of the book, and three-dimensional methods were discussed in the second half. Now, however, three-dimensional graphics applications are commonplace, and many initial computer-graphics courses either deal primarily with three-dimensional methods or introduce three-dimensional graphics at an early stage. Therefore, a major feature of this third edition is the integration of three-dimensional and two-dimensional topics.

We have also expanded the treatment of most topics to include discussions of recent developments and new applications. General subjects covered in this third edition include: current hardware and software components of graphics
systems, fractal geometry, ray tracing, splines, illumination models, surface rendering, computer animation, virtual reality, parallel implementations for graphics algorithms, antialiasing, superquadrics, BSP trees, particle systems, physically based modeling, scientific visualization, radiosity, bump mapping, and morphing. Some of the major expansion areas are animation, object representations, the three-dimensional viewing pipeline, illumination models, surface-rendering techniques, and texture mapping.

Another significant change in this third edition is the introduction of the OpenGL set of graphics routines, which is now widely used and available on most computer systems. The OpenGL package provides a large and efficient collection of device-independent functions for creating computer-graphics displays, using a program written in a general-purpose language such as C or C++. Auxiliary libraries are available in OpenGL for handling input and output operations, which require device interactions, and for additional graphics procedures such as generating cylinder shapes, spherical objects, and B-splines.

**Programming Examples**

More than twenty complete C++ programs are provided in this third edition, using the library of graphics routines available in the popular OpenGL package. These programs illustrate applications of basic picture-construction techniques, two-dimensional and three-dimensional geometric transformations, two-dimensional and three-dimensional viewing methods, perspective projections, spline generation, fractal methods, interactive mouse input, picking operations, menu and submenu displays, and animation techniques. In addition, over one hundred C++/OpenGL program segments are given to demonstrate the implementation of computer-graphics algorithms for clipping, lighting effects, surface rendering, texture mapping, and many other computer-graphics methods.

**Required Background**

We assume no prior familiarity with computer graphics, but we do assume that the reader has some knowledge of computer programming and basic data structures, such as arrays, pointer lists, files, and record organizations. A variety of mathematical methods are used in computer-graphics algorithms, and these methods are discussed in some detail in the appendix. Mathematical topics covered in the appendix include techniques from analytic geometry, linear algebra, vector and tensor analysis, complex numbers, quaternions, basic calculus, and numerical analysis.

This third edition can be used both as a text for students with no prior background in computer graphics and as a reference for graphics professionals. The emphasis is on the basic principles needed to design, use, and understand computer-graphics systems, along with numerous example programs to illustrate the methods and applications for each topic.

**Suggested Course Outlines**

For a one-semester course, a subset of topics dealing with either two-dimensional methods or a combination of two-dimensional and three-dimensional topics can be chosen, depending on the requirements of a particular course. A two-semester course sequence can cover the basic graphics concepts and algorithms in the first semester and advanced three-dimensional methods in the second. For the
self-study reader, early chapters can be used to provide an understanding of graphics concepts, supplemented with selected topics from the later chapters.

At the undergraduate level, an introductory computer-graphics course can be organized using selected material from Chapters 2 through 6, 11, and 13. Sections could be chosen from these chapters to cover two-dimensional methods only, or three-dimensional topics could be added from these chapters along with limited selections from Chapters 7 and 10. Other topics, such as fractal representations, spline curves, texture mapping, depth-buffer methods, or color models, could be introduced in a first computer-graphics course. For an introductory graduate or upper-level undergraduate course, more emphasis could be given to three-dimensional viewing, three-dimensional modeling illumination models, and surface-rendering methods. In general, however, a two-semester sequence provides a better framework for adequately covering the fundamentals of two-dimensional and three-dimensional computer-graphics methods, including spline representations, surface rendering, and ray tracing. Special-topics courses, with an introductory computer-graphics prerequisite, can be offered in one or two areas, selected from visualization techniques, fractal geometry, spline methods, ray tracing, radiosity, and computer animation.

Chapter 1 illustrates the diversity of computer-graphics applications by taking a look at the many different kinds of pictures that people have generated with graphics software. In Chapter 2, we present the basic vocabulary of computer graphics, along with an introduction to the hardware and software components of graphics systems, a detailed introduction to OpenGL, and a complete OpenGL example program. The fundamental algorithms for the representation and display of simple objects are given in Chapters 3 and 4. These two chapters examine methods for producing basic picture components such as polygons and circles; for setting the color, size, and other attributes of objects; and for implementing these methods in OpenGL. Chapter 5 discusses the algorithms for performing geometric transformations such as rotation and scaling. In Chapters 6 and 7, we give detailed explanations of the procedures for displaying views of two-dimensional and three-dimensional scenes. Methods for generating displays of complex objects, such as quadric surfaces, splines, fractals, and particle systems are discussed in Chapter 8. In Chapter 9 we explore the various computer-graphics techniques for identifying the visible objects in a three-dimensional scene. Illumination models and the methods for applying lighting conditions to a scene are examined in Chapter 10. Methods for interactive graphics input and for designing graphical user interfaces are given in Chapter 11. The various color models useful in computer graphics are discussed in Chapter 12, along with color-design considerations. Computer-animation techniques are explored in Chapter 13. Methods for the hierarchical modeling of complex systems are presented in Chapter 14. And, in Chapter 15, we survey the major graphics file formats.

Acknowledgments

Many people have contributed to this project in a variety of ways over the years. To the organizations and individuals who furnished pictures and other materials, we again express our appreciation. We also acknowledge the many helpful comments received from our students in various computer-graphics and visualization courses and seminars. We are indebted to all those who provided comments, reviews, suggestions for improving the material covered in this book, and other input, and we extend our apologies to anyone we may have failed to mention. Our thanks to Ed Angel, Norman Badler, Phillip Barry, Brian Barsky,
Hedley Bond, Bart Braden, Lara Burton, Robert Burton, Greg Chwelos, John Cross, Steve Cunningham, John DeCatrel, Victor Duvaneko, Gary Eerkes, Parris Egbert, Tony Faustini, Thomas Foley, Thomas Frank, Don Gillies, Andrew Glassner, Jack Goldfeather, Georges Grinstein, Eric Haines, Robert Herbst, Larry Hodges, Carol Hubbard, Eng-Kiat Koh, Mike Krogh, Michael Laszlo, Suzanne Lea, Michael May, Nelson Max, David McAllister, Jeffrey McConnell, Gary McDonald, C. L. Morgan, Greg Nielson, James Oliver, Lee-Hian Quek, Laurence Rainville, Paul Ross, David Salomon, Günther Schrack, Steven Shafer, Cliff Shaffer, Pete Shirley, Carol Smith, Stephanie Smullen, Jeff Spears, William Taffe, Wai Wan Tsang, Spencer Thomas, Sam Uselton, David Wen, Bill Wicker, Andrew Woo, Angelo Yfantis, Marek Zaremba, Michael Zyda, and the many anonymous reviewers. We also thank our editor Alan Apt, Toni Holm, and the Colorado staff for their help, suggestions, encouragement, and, above all, their patience during the preparation of this third edition. And to our production editors and staff, Lynda Castillo, Camille Trentacoste, Heather Scott, Xiaohong Zhu, Vince O’Brien, Patricia Burns, Kathy Ewing, and David Abel, we extend our sincere appreciation for their many talented contributions and careful attention to detail.