

CONTENTS

INTRODUCTION

xxi

CHAPTER 1: INTRODUCING WEBGL

1

The Basics of WebGL	1
So Why Is WebGL So Great?	2
Designing a Graphics API	3
An Immediate-Mode API	3
A Retained-Mode API	4
An Overview of Graphics Hardware	4
GPU	5
Framebuffer	5
Texture Memory	7
Video Controller	7
Understanding the WebGL Graphics Pipeline	7
Vertex Shader	8
Primitive Assembly	12
Rasterization	14
Fragment Shader	14
Per Fragment Operations	17
Comparing WebGL to Other Graphics Technologies	19
OpenGL	19
OpenGL ES 2.0	21
Direct3D	23
HTML5 Canvas	24
Scalable Vector Graphics	28
VRML and X3D	30
Linear Algebra for 3D Graphics	31
Coordinate System	31
Points or Vertices	31
Vectors	32
Dot Product or Scalar Product	33
Cross Product	34
Homogeneous Coordinates	35
Matrices	35
Affine Transformations	38
Summary	44

CHAPTER 2: CREATING BASIC WebGL EXAMPLES	45
Drawing a Triangle	46
Creating the WebGL Context	49
Creating the Vertex Shader and the Fragment Shader	51
Compiling the Shaders	52
Creating the Program Object and Linking the Shaders	53
Setting Up the Buffers	54
Drawing the Scene	56
Understanding the WebGL Coding Style	57
Debugging Your WebGL Application	58
Using Chrome Developer Tools	58
Using Firebug	65
WebGL Error Handling and Error Codes	67
WebGL Inspector	70
Troubleshooting WebGL	76
Using the DOM API to Load Your Shaders	78
Putting It Together in a Slightly More Advanced Example	80
Experimenting with Code	83
Summary	84
CHAPTER 3: DRAWING	85
WebGL Drawing Primitives and Drawing Methods	86
Primitives	86
Understanding the Importance of Winding Order	91
WebGL's Drawing Methods	93
Typed Arrays	99
Buffer and View	100
Supported View Types	101
Exploring Different Ways to Draw	102
gl.drawArrays() and gl.TRIANGLES	103
gl.drawArrays() and gl.TRIANGLE_STRIP	105
gl.drawElements() and gl.TRIANGLES	106
gl.drawElements() and gl.TRIANGLE_STRIP	108
Conclusions of the Comparison	111
Pre-Transform Vertex Cache and Post-Transform Vertex Cache	111
Interleaving Your Vertex Data for Improved Performance	114
Using an Array of Structures	115
Using a Vertex Array or Constant Vertex Data	123
A Last Example to Wind Things Up	124
Some Things to Experiment With	134
Summary	134

CHAPTER 4: COMPACT JAVASCRIPT LIBRARIES AND TRANSFORMATIONS	137
Working with Matrices and Vectors in JavaScript	138
Sylvester	139
WebGL-mjs	142
glMatrix	146
Using Transformations	150
How Transformations Are Used	150
Understanding the Complete Transformation Pipeline	157
Getting Practical with Transformations	158
Setting Up Buffers with Object Coordinates	159
Creating Transformation Matrices with JavaScript and Uploading Them to the Shader	160
Uploading the Transformation Matrices to the Vertex Shader in the GPU	161
Calling Your Drawing Methods	162
Understanding the Importance of Transformation Order	162
Using a Grand, Fixed Coordinate System	163
Using a Moving, Local Coordinate System	165
Pushing and Popping Transformation Matrices	167
A Complete Example: Drawing Several Transformed Objects	171
Creating a Cube with WebGL	173
Organization of the View Transformation and the Model Transformation	175
Summary	176
CHAPTER 5: TEXTURING	177
Understanding Lost Context	178
Understanding the Setup Required to Handle Lost Context	179
Factors to Consider When Handling Lost Context	181
Introducing 2D Textures and Cubemap Textures	183
Loading Your Textures	185
Creating a WebGLTexture Object	185
Binding Your Texture	185
Loading the Image Data	186
Uploading the Texture to the GPU	187
Specifying Texture Parameters	189
Understanding the Complete Procedure of Loading a Texture	190
Defining Your Texture Coordinates	193

Using Your Textures in Shaders	195
Working with Texture Image Units	197
Working with Texture Filtering	198
Understanding Magnification	199
Understanding Minification	200
Understanding Mipmapping	200
Understanding Texture Coordinate Wrapping	203
Using the gl.REPEAT Wrap Mode	203
Using the gl.MIRRORED_REPEAT Wrap Mode	205
Using the gl.CLAMP_TO_EDGE Wrap Mode	206
A Complete Texture Example	207
Using Images for Your Textures	210
Downloading Free Textures	210
Basing Textures on Your Own Photos	211
Drawing Images	211
Buying Textures	211
Understanding Same-Origin Policy and Cross-Origin Resource Sharing	212
Understanding Same-Origin Policy for Images in General	212
Understanding Same-Origin Policy for Textures	214
Understanding Cross-Origin Resource Sharing	215
Summary	217
CHAPTER 6: ANIMATIONS AND USER INPUT	219
Animating the Scene	219
Using setInterval() and setTimeout()	221
Using requestAnimationFrame()	222
Compensating Movement for Different Frame Rates	225
Creating an FPS Counter to Measure the Smoothness of Your Animation	226
Understanding the Disadvantages of Using FPS as a Measurement	228
Event Handling for User Interaction	230
Basic Event Handling with DOM Level 0	231
Advanced Event Handling with DOM Level 2	232
Key Input	234
Mouse Input	239
Applying Your New Knowledge	240
Summary	246
CHAPTER 7: LIGHTING	249
Understanding Light	249
Working with a Local Lighting Model	250

Understanding the Phong Reflection Model	251
Ambient Reflection	252
Diffuse Reflection	253
Specular Reflection	255
Understanding the Complete Equation and Shaders for the Phong Reflection Model	259
Using Lighting with Texturing	263
Understanding the JavaScript Code Needed for WebGL Lighting	267
Setting Up Buffers with Vertex Normals	268
Calculating and Uploading the Normal Matrix to the Shader	270
Uploading the Light Information to the Shader	270
Using Different Interpolation Techniques for Shading	271
Flat Shading	272
Gouraud Shading	273
Phong Shading	274
Understanding the Vectors That Must Be Normalized	278
Normalization in the Vertex Shader	278
Normalization in the Fragment Shader	279
Using Different Types of Lights	279
Directional Lights	280
Point Lights	280
Spot Lights	281
Understanding the Attenuation of Light	284
Understanding Light Mapping	288
Summary	289
 CHAPTER 8: WEBGL PERFORMANCE OPTIMIZATIONS	 291
WebGL under the Hood	292
Hardware that Powers WebGL	292
Key Software Components	294
WebGL Performance Optimizations	296
Avoiding a Typical Beginner's Mistake	296
Locating the Bottleneck	298
General Performance Advice	302
Performance Advice for CPU-Limited WebGL	305
Performance Advice for Vertex-Limited WebGL	307
Performance Advice for Pixel-Limited WebGL	308
A Closer Look at Blending	310
Introducing Blending	310
Setting the Blending Functions	311

Understanding Drawing Order and the Depth Buffer	314
Drawing a Scene with Both Opaque and Semi-Transparent Objects	315
Changing the Default Operator for the Blend Equation	315
Using Premultiplied Alpha	316
Taking WebGL Further	317
Using WebGL Frameworks	317
Publishing to the Google Chrome Web Store	318
Using Additional Resources	318
Summary	319
 INDEX	 321