

Learning Modern 3D Graphics Programming

[Next](#)

Learning Modern 3D Graphics Programming

Jason L. McKesson

Copyright © 2012 Jason L. McKesson

Table of Contents

[About this Book](#)

- [Why Read This Book?](#)
- [What You Need](#)
- [Organization of This Book](#)
- [Conventions used in This Book](#)

[Building the Tutorials](#)

[I. The Basics](#)

- [Introduction](#)
- [1. Hello, Triangle!](#)
- [2. Playing with Colors](#)

[II. Positioning](#)

- [3. OpenGL's Moving Triangle](#)
- [4. Objects at Rest](#)
- [5. Objects in Depth](#)
- [6. Objects in Motion](#)
- [7. World in Motion](#)
- [8. Getting Oriented](#)

[III. Illumination](#)

- [9. Lights On](#)
- [10. Plane Lights](#)
- [11. Shinies](#)
- [12. Dynamic Range](#)
- [13. Lies and Impostors](#)

[IV. Texturing](#)

- [14. Textures are not Pictures](#)
- [15. Many Images](#)

[16. Gamma and Textures](#)

[17. Spotlight on Textures](#)

[V. Framebuffer](#)

[VI. Advanced Lighting](#)

[A. Basic Optimization](#)

[Vertex Format](#)

[Textures](#)

[Object Optimizations](#)

[Finding the Bottleneck](#)

[Vertex Format](#)

[Synchronization](#)

[B. Further Study](#)

[Debugging](#)

[Topics of Interest](#)

[C. History of PC Graphics Hardware](#)

[Voodoo Magic](#)

[Dynamite Combiners](#)

[Vertices and Registers](#)

[Programming at Last](#)

[Dependency](#)

[Modern Unification](#)

[D. Getting Started with OpenGL](#)

[Manual Usage](#)

List of Figures

1. [Position Vectors](#)
2. [Direction Vectors](#)
3. [Vector Addition](#)
4. [Vector Addition Head-to-Tail](#)
5. [Vector Negation](#)
6. [Vector Subtraction](#)
7. [Vector Scaling](#)
8. [An Image](#)
9. [Normalized Device Coordinate Space](#)
10. [Scan Converted Triangle](#)
11. [Shared Edge Scan Conversion](#)
- 1.1. [Data Flow to Vertex Shader](#)
- 1.2. [Data Flow to Rasterizer](#)
- 2.1. [Fragment Position](#)
- 2.2. [Vertex Array Memory Map](#)
- 2.3. [Multiple Vertex Attributes](#)
- 2.4. [Interpolated Vertex Colors](#)

- 4.1. [Triangle Winding Order](#)
- 4.2. [Orthographic Prism](#)
- 4.3. [2D to 1D Orthographic Projection](#)
- 4.4. [2D to 1D Perspective Projection](#)
- 4.5. [Viewing Frustum](#)
- 4.6. [2D to 1D Perspective Projection Diagram](#)
- 4.7. [Camera to NDC Transformation in 2D](#)
- 4.8. [Perspective Prism](#)
- 4.9. [Perspective Matrix](#)
- 4.10. [Bad Aspect Ratio](#)
- 4.11. [Widescreen Aspect Ratio Frustum](#)
- 5.1. [Overlapping Objects](#)
- 5.2. [Three Overlapping Triangles](#)
- 5.3. [Depth Buffering](#)
- 5.4. [Mild Overlap](#)
- 5.5. [Major Overlap](#)
- 5.6. [Triangle Clipping](#)
- 5.7. [Near Plane Clipping](#)
- 5.8. [Depth Clamping](#)
- 5.9. [Depth Clamp With Overlap](#)
- 6.1. [Two 2D Coordinate Systems](#)
- 6.2. [Coordinate System Translation in 2D](#)
- 6.3. [Translation Project](#)
- 6.4. [Coordinate System Scaling in 2D](#)
- 6.5. [Scale Project](#)
- 6.6. [Coordinate Rotation in 2D](#)
- 6.7. [Rotation Project](#)
- 6.8. [Transform Order Diagram](#)
- 6.9. [Hierarchy Project](#)
- 7.1. [Full Vertex Transformation Pipeline](#)
- 7.2. [World Space Scene](#)
- 7.3. [Spherical Coordinates](#)
- 7.4. [Triangle Fan](#)
- 7.5. [Triangle Strip](#)
- 7.6. [Triangle Strips with Winding Order](#)
- 7.7. [Uniform Buffer and Block Binding Points](#)
- 8.1. [Gimbal Lock Project](#)
- 8.2. [Parallel Gimbals](#)
- 8.3. [Quaternion YPR Project](#)
- 8.4. [Camera Relative Project](#)
- 8.5. [Interpolation Directions](#)
- 9.1. [Surface Light Absorption](#)
- 9.2. [Perpendicular Light](#)
- 9.3. [Light at an Angle](#)
- 9.4. [Diffuse Reflectance](#)
- 9.5. [Near and Far Lights](#)
- 9.6. [Basic Lighting](#)
- 9.7. [Circle Scaling](#)
- 9.8. [Circle Scaling with Normals](#)

- 9.9. [Lighting and Scale](#)
- 9.10. [Half Lit](#)
- 9.11. [Ambient Lighting](#)
- 9.12. [Cube Position Topology](#)
- 9.13. [Cube Normal Topology](#)
- 9.14. [Full Cube Topology](#)
- 10.1. [Vertex Point Lighting](#)
- 10.2. [Light Near Surface](#)
- 10.3. [Triangle Interpolation](#)
- 10.4. [Triangle Edge Interpolation](#)
- 10.5. [Two Triangle Quadrilateral](#)
- 10.6. [Two Triangle Interpolation](#)
- 10.7. [Fragment Point Lighting](#)
- 10.8. [Close Lit Cylinder](#)
- 10.9. [Adjacent Gradient](#)
- 10.10. [Gradient Intensity Plot](#)
- 10.11. [High Light](#)
- 10.12. [Fragment Attenuation](#)
- 11.1. [Perfect Specular Reflection](#)
- 11.2. [Smooth and Rough Microfacets](#)
- 11.3. [Phong Lighting](#)
- 11.4. [Phong with Dark Diffuse](#)
- 11.5. [Phong Clipping](#)
- 11.6. [Phong Distortion](#)
- 11.7. [Large View and Reflect Angle](#)
- 11.8. [Geometric Half-Angle Vector](#)
- 11.9. [Perfect Reflection Half-Angle Vector](#)
- 11.10. [Blinn Lighting](#)
- 11.11. [Blinn vs. Phong Lighting](#)
- 11.12. [Light Edge](#)
- 11.13. [Improved Light Edge](#)
- 11.14. [Gaussian Probability Distribution Curves](#)
- 11.15. [Gaussian with Sharp Highlight](#)
- 12.1. [Scene Lighting](#)
- 12.2. [Darkness, Day vs. Night](#)
- 12.3. [Light Clipping](#)
- 12.4. [HDR Lighting](#)
- 12.5. [Gamma Function Graph](#)
- 12.6. [Gamma Correction](#)
- 12.7. [Gamma Lighting](#)
- 12.8. [Gamma Shadow Details](#)
- 13.1. [Basic Impostor](#)
- 13.2. [Circle Point Computation](#)
- 13.3. [Bad Impostor](#)
- 13.4. [Circle Projection](#)
- 13.5. [Bad vs. Good](#)
- 13.6. [Bad Intersection](#)
- 13.7. [Depth Correct Impostor](#)
- 14.1. [Basic Texture](#)

- 14.2. [Texture Binding and Context](#)
- 14.3. [Sampler Binding and Context](#)
- 14.4. [Projection and Interpolation](#)
- 14.5. [Perspective Correct Interpolation](#)
- 14.6. [Material Texture](#)
- 14.7. [A Torus](#)
- 14.8. [Surface smudges](#)
- 15.1. [Basic Checkerboard Plane](#)
- 15.2. [Jagged Texture Edge](#)
- 15.3. [Nearest Sampling](#)
- 15.4. [Linear Filtering](#)
- 15.5. [Large Minification Sampling](#)
- 15.6. [Mipmapped Minification Sampling](#)
- 15.7. [Hallway with Mipmapping](#)
- 15.8. [Hallway with Special Texture](#)
- 15.9. [Linear Mipmap Linear Comparison](#)
- 15.10. [Main Diagonal](#)
- 15.11. [Long Fragment Area](#)
- 15.12. [Long Fragment with Sample Area](#)
- 15.13. [Parallelogram Sample Area](#)
- 15.14. [Anisotropic Filtering](#)
- 15.15. [Max Anisotropic Filtering](#)
- 16.1. [Gamma Ramp](#)
- 16.2. [Gamma Ramp with sRGB Images](#)
- 16.3. [Gamma Checkers](#)
- 16.4. [Gamma Correct with Gamma Mipmaps](#)
- 16.5. [Gamma Landscape](#)
- 17.1. [Double Projection](#)
- 17.2. [Top View Projection](#)
- 17.3. [Near View Projection](#)
- 17.4. [Projected Light](#)
- 17.5. [Back Projected Light](#)
- 17.6. [Colored Spotlight](#)
- 17.7. [Edge Clamped Light](#)
- 17.8. [Border Clamped Light](#)
- 17.9. [Cube Map Face Orientation](#)
- 17.10. [Cube Point Light](#)

List of Tables

- 6.1. [Hierarchy Tutorial Key Commands](#)
- 7.1. [World Space Controls](#)
- 10.1. [Transform Legend](#)
- 12.1. [Scene Lighting Values](#)
- 13.1. [Sphere Impostor Control Key Map](#)

List of Examples

- 1.1. [The display Function](#)
- 1.2. [Buffer Object Initialization](#)

- 1.3. [Vertex Shader](#)
- 1.4. [Reshaping Window](#)
- 1.5. [Fragment Shader](#)
- 1.6. [Program Initialization](#)
- 1.7. [Shader Creation](#)
- 1.8. [Program Creation](#)
- 2.1. [FragPosition's Fragment Shader](#)
- 2.2. [New Vertex Array Data](#)
- 2.3. [Buffer Object Initialization](#)
- 2.4. [Rendering the Scene](#)
- 2.5. [Vertex Arrays](#)
- 2.6. [Draw Arrays Implementation](#)
- 2.7. [Multi-input Vertex Shader](#)
- 2.8. [Fragment Shader with Input](#)
- 3.1. [Computation of Position Offsets](#)
- 3.2. [Adjusting the Vertex Data](#)
- 3.3. [Updating and Drawing the Vertex Data](#)
- 3.4. [Offsetting Vertex Shader](#)
- 3.5. [Draw with Calculated Offsets](#)
- 3.6. [Offset Computing Vertex Shader](#)
- 3.7. [Rendering with Time](#)
- 3.8. [Loop Duration Setting](#)
- 3.9. [Time-based Fragment Shader](#)
- 3.10. [More Shader Creation](#)
- 4.1. [Face Culling Initialization](#)
- 4.2. [ManualPerspective Vertex Shader](#)
- 4.3. [Program Initialization](#)
- 4.4. [MatrixPerspective Vertex Shader](#)
- 4.5. [Program Initialization of Perspective Matrix](#)
- 4.6. [Square-only Viewport](#)
- 4.7. [Reshape with Aspect Ratio](#)
- 5.1. [Draw Arrays Implementation](#)
- 5.2. [Draw Elements Implementation](#)
- 5.3. [VAO Initialization](#)
- 5.4. [VAO and Indexed Rendering Code](#)
- 5.5. [Vertex Attribute Data Abridged](#)
- 5.6. [Array Drawing of Two Objects with One VAO](#)
- 5.7. [MultiObject Element Buffer](#)
- 5.8. [Base Vertex Single VAO](#)
- 5.9. [Base Vertex Rendering](#)
- 5.10. [Depth Buffer Setup](#)
- 5.11. [Depth Buffer Clearing](#)
- 5.12. [Depth Clamping On/Off](#)
- 6.1. [Translation Shader Initialization](#)
- 6.2. [Frustum Scale Computation](#)
- 6.3. [Translation Matrix Generation](#)
- 6.4. [Rotation Transformation Building](#)
- 6.5. [Hierarchy::Draw](#)
- 7.1. [Window Resizing](#)

- 7.2. [Position-only Vertex Shader](#)
- 7.3. [Upload World to Camera Matrix](#)
- 7.4. [Spherical to Euclidean Transform](#)
- 7.5. [Draw the Ground](#)
- 7.6. [DrawForest Function](#)
- 7.7. [Call to DrawParthenon](#)
- 7.8. [Draw Camera Target](#)
- 7.9. [Cylinder Mesh File](#)
- 7.10. [UBO-based Vertex Shader](#)
- 7.11. [Uniform Buffer Creation](#)
- 7.12. [UBO-based Perspective Matrix](#)
- 7.13. [UBO-based Camera Matrix](#)
- 7.14. [Viewing Point with UBO](#)
- 8.1. [Gimbal Lock Display Code](#)
- 8.2. [Quaternion YPR Display](#)
- 8.3. [OffsetOrientation Function](#)
- 8.4. [Camera Relative OffsetOrientation](#)
- 8.5. [Quaternion Linear Interpolation](#)
- 8.6. [Spherical Linear Interpolation](#)
- 9.1. [Display Camera Code](#)
- 9.2. [Ground Plane Lighting](#)
- 9.3. [Cylinder Lighting](#)
- 9.4. [Lighting Vertex Shader](#)
- 9.5. [Lighting with Proper Normal Transform](#)
- 9.6. [Ambient Vertex Lighting](#)
- 9.7. [Lighting Intensity Settings](#)
- 10.1. [Per-Vertex Point Light Rendering](#)
- 10.2. [Per-Vertex Point Light Vertex Shader](#)
- 10.3. [Initial Per-Fragment Rendering](#)
- 10.4. [Ground Plane Per-Fragment Rendering](#)
- 10.5. [Model Space Per-Vertex Lighting Vertex Shader](#)
- 10.6. [Model Space Per-Fragment Lighting Vertex Shader](#)
- 10.7. [Per-Fragment Lighting Fragment Shader](#)
- 10.8. [Light Attenuation Fragment Shader Definitions](#)
- 10.9. [Window to Camera Space Function](#)
- 10.10. [Light Intensity Application Function](#)
- 10.11. [Main Light Attenuation](#)
- 11.1. [Phong Lighting Shader](#)
- 11.2. [Blinn-Phong Lighting Shader](#)
- 11.3. [Gaussian Lighting Shader](#)
- 12.1. [Material Uniform Block](#)
- 12.2. [Material UBO Construction](#)
- 12.3. [Daytime Lighting](#)
- 12.4. [Light Uniform Block](#)
- 12.5. [Many Lights Main Function](#)
- 12.6. [HDR LightBlock](#)
- 12.7. [Gamma LightBlock](#)
- 12.8. [Fragment Gamma Correction](#)
- 13.1. [Basic Impostor Vertex Shader](#)

- 13.2. [Basic Impostor Fragment Shader](#)
- 13.3. [Ray Traced Impostor Square](#)
- 13.4. [Depth Correct Fragment Shader](#)
- 13.5. [Impostor Geometry Creation](#)
- 13.6. [Vertex Shader for Points](#)
- 13.7. [Geometry Shader Definitions](#)
- 13.8. [Geometry Shader Vertex Computation](#)
- 13.9. [Fragment Shader Changes](#)
- 14.1. [BuildGaussianData function](#)
- 14.2. [CreateGaussianTexture function](#)
- 14.3. [Shader Texture Access](#)
- 14.4. [Sampler Object Creation](#)
- 14.5. [BuildGaussianData in 2D](#)
- 14.6. [CreateGaussianTexture in 2D](#)
- 14.7. [CreateShininessTexture function](#)
- 14.8. [Shininess Texture Access](#)
- 14.9. [Gaussian Texture with Specular](#)
- 15.1. [DDS Texture Loading with Mipmaps](#)
- 15.2. [Special Texture Data](#)
- 16.1. [sRGB Image Format](#)
- 16.2. [Window to Clip Matrix Computation](#)
- 16.3. [Vertex Format](#)
- 16.4. [Gamma Landscape defaults Function](#)
- 16.5. [Enable sRGB Conversion](#)
- 17.1. [Scene Graph Shader Definition](#)
- 17.2. [Scene Graph Node Definition](#)
- 17.3. [Double Projection LoadAndSetupScene](#)
- 17.4. [Left Projection Matrix](#)
- 17.5. [Right Projection Matrix](#)
- 17.6. [View Camera to Projected Texture Transform](#)
- 17.7. [Border Clamp Sampler Objects](#)
- 17.8. [Cube Texture Loading](#)
- 17.9. [View Camera to Light Cube Texture](#)

List of Equations

- 1. [Vector Addition with Numbers](#)
- 2. [Vector Negation](#)
- 3. [Vector Multiplication](#)
- 4. [Vector-Scalar Multiplication](#)
- 5. [Vector-Scalar Addition](#)
- 6. [Vector Algebra](#)
- 7. [Vector Length](#)
- 8. [Vector Normalization](#)
- 4.1. [Perspective Computation](#)
- 4.2. [Depth Computation](#)
- 4.3. [Camera to Clip Equations](#)
- 4.4. [Camera to Clip Expanded Equations](#)
- 4.5. [Camera to Clip Matrix Transformation](#)
- 4.6. [Vector Matrix Multiplication](#)

- 5.1. [Perspective Computation](#)
 - 6.1. [Coordinate System](#)
 - 6.2. [Identity Matrix](#)
 - 6.3. [Translation Matrix](#)
 - 6.4. [Scaling Transformation Matrix](#)
 - 6.5. [Vectorized Matrix Multiplication](#)
 - 6.6. [Axial Rotation Matrices](#)
 - 6.7. [Angle/Axis Rotation Matrix](#)
 - 6.8. [Order of Transformation](#)
 - 8.1. [Angle/Axis to Quaternion](#)
 - 8.2. [Quaternion Multiplication](#)
 - 8.3. [Quaternion to Matrix](#)
 - 9.1. [Diffuse Lighting Equation](#)
 - 9.2. [Dot Product](#)
 - 9.3. [Dot Product from Vector Math](#)
 - 9.4. [Matrix Transpose](#)
 - 10.1. [Physical Light Attenuation](#)
 - 10.2. [Light Attenuation Inverse](#)
 - 10.3. [Camera to Window Transforms](#)
 - 10.4. [Window to Camera Transforms](#)
 - 11.1. [Phong Specular Term](#)
 - 11.2. [Vector Reflection](#)
 - 11.3. [Half-Angle Vector](#)
 - 11.4. [Blinn Specular Term](#)
 - 11.5. [Gaussian Distribution Function](#)
 - 11.6. [Gaussian Specular Term](#)
 - 12.1. [Display Gamma Function](#)
 - 12.2. [Gamma Correction Function](#)
 - 13.1. [Ray Equation](#)
 - 13.2. [Sphere Equation](#)
 - 14.1. [Gaussian as Function of One Variable](#)
-

[Next](#)

About this Book