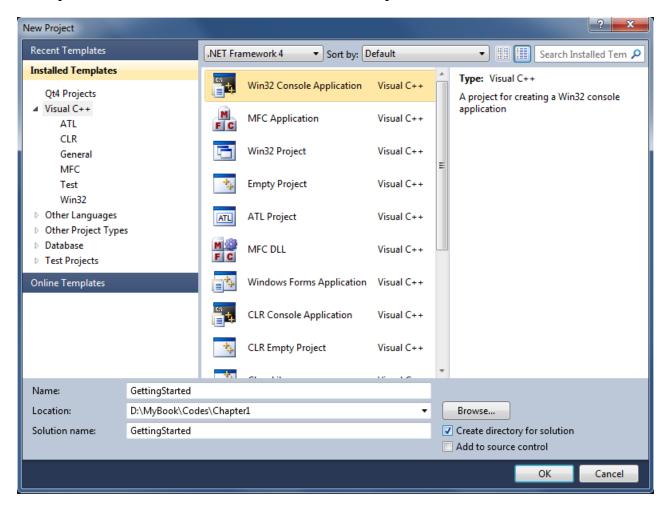
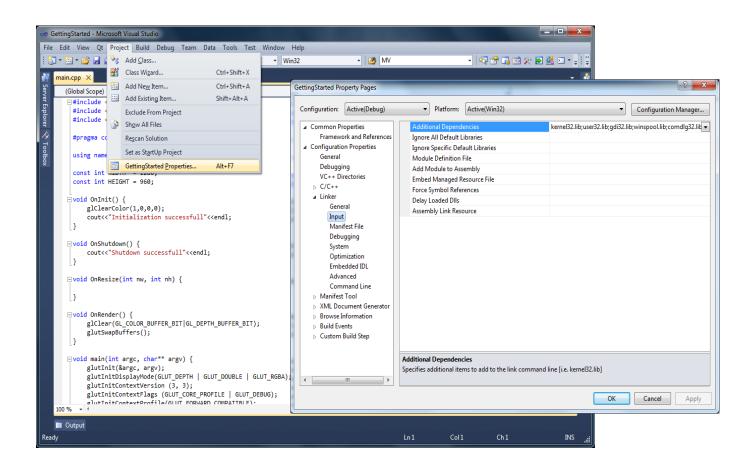
### **Chapter 1: Introduction to Modern OpenGL**



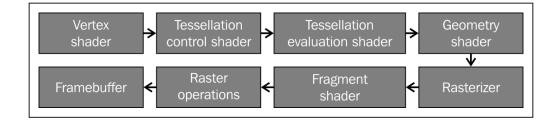
| Win32 Application Wizard - Get | tingStarted   | ? ×                          |
|--------------------------------|---|------------------------------|
| Applicat                       | tion Settings   |                              |
| Overview                       | Application type:   | Add common header files for: |
| Application Settings           | <ul> <li>Windows application</li> <li>Console application</li> <li>DLL</li> <li>Static library</li> <li>Additional options:</li> <li>Empty project</li> <li>Export symbols</li> <li>Precompiled header</li> </ul> | ☐ ATL<br>☐ MFC               |
|                                | < Previous  | Next > Finish Cancel         |

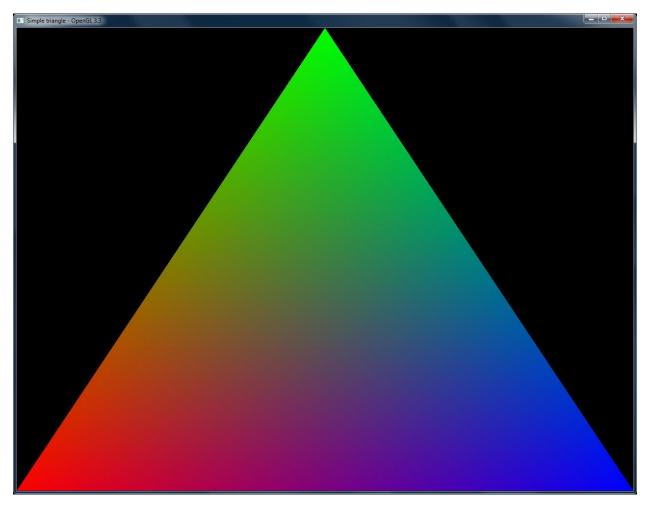
| 8 ×      |
|----------|
|          |
| *        |
| •        |
|          |
| Macros>> |
|          |
|          |

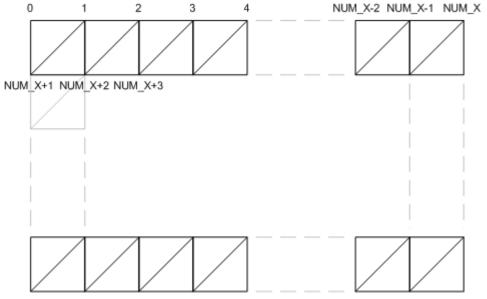
| ibrary Directories  | 8 ×      |
|---|----------|
|   |          |
| D:\Libraries\glew-1.9.0\lib\<br>D:\Libraries\freeglut-2.8.0\lib\x86\          | A        |
| ٠ III   | •        |
|   |          |
| \$(VCInstallDir)atlmfc\lib<br>\$(WindowsSdkDir)lib<br>\$(FrameworkSDKDir)\lib | -        |
| \$(WindowsSdkDir)lib  | Macros>> |

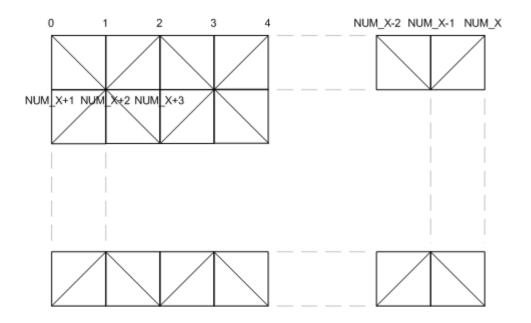


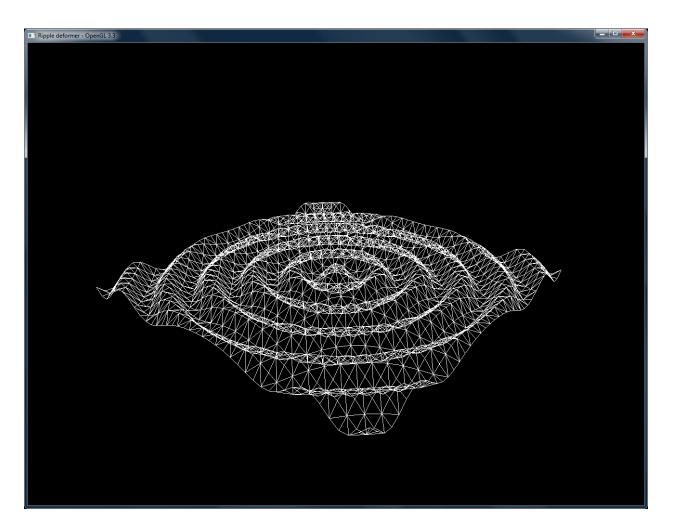


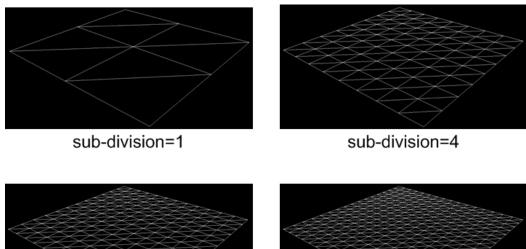


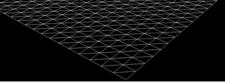












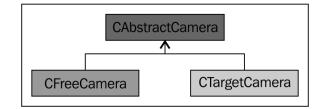
sub-division=8

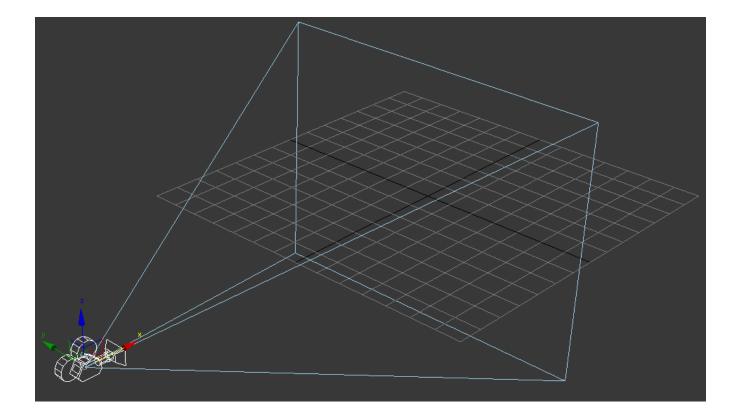
sub-division=6

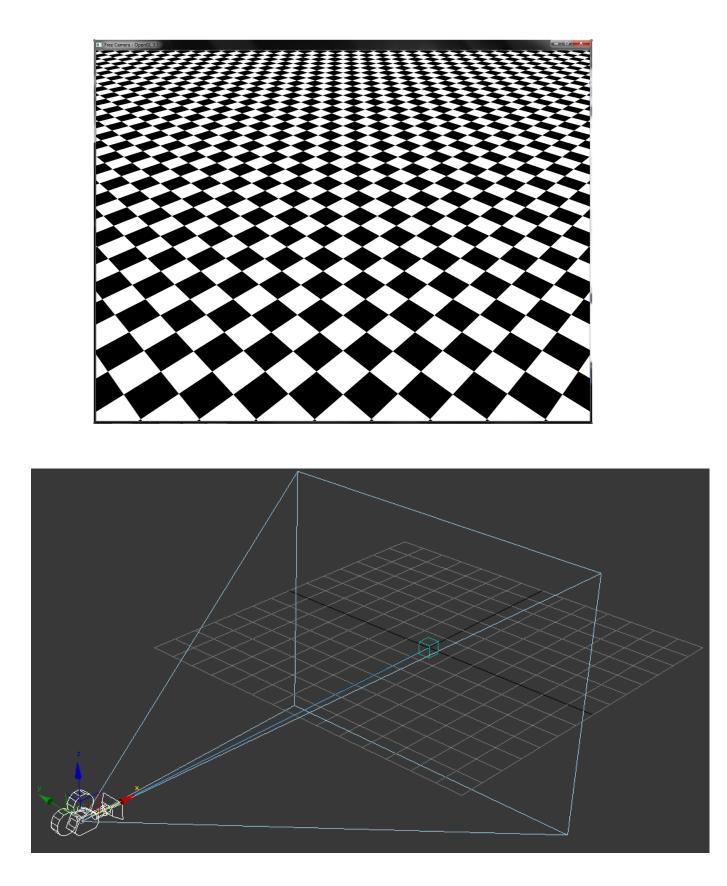


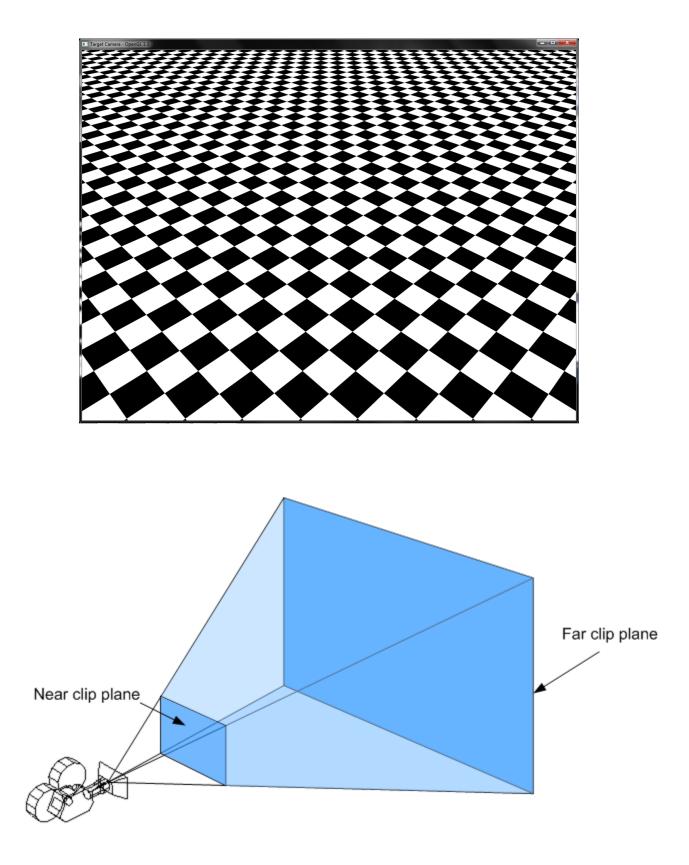
$$y = A.\sin(2\pi ft + \varphi)$$
$$d(x, y, z) = \sqrt{x^2 + y^2 + z^2}$$
$$F(x, y, x) = A.\sin(-\pi fd(x, y, z) + \varphi)$$

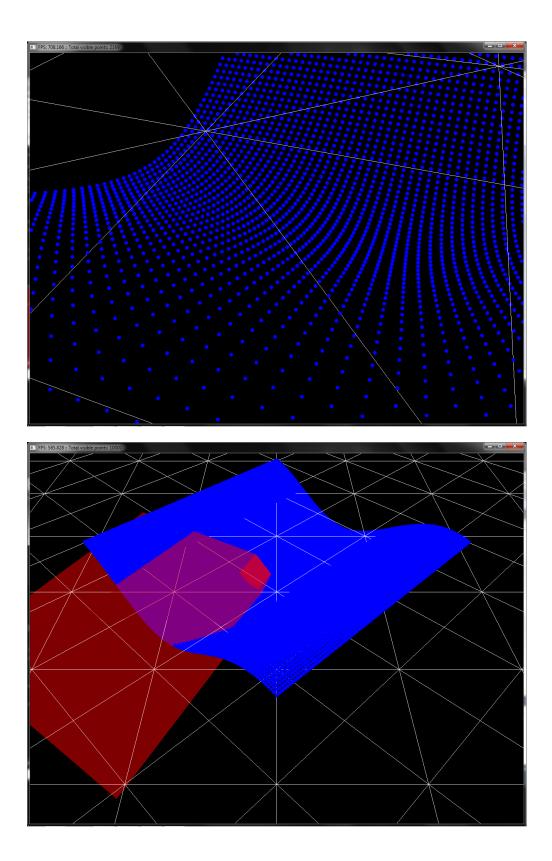
## **Chapter 2: 3D Viewing and Object Picking**

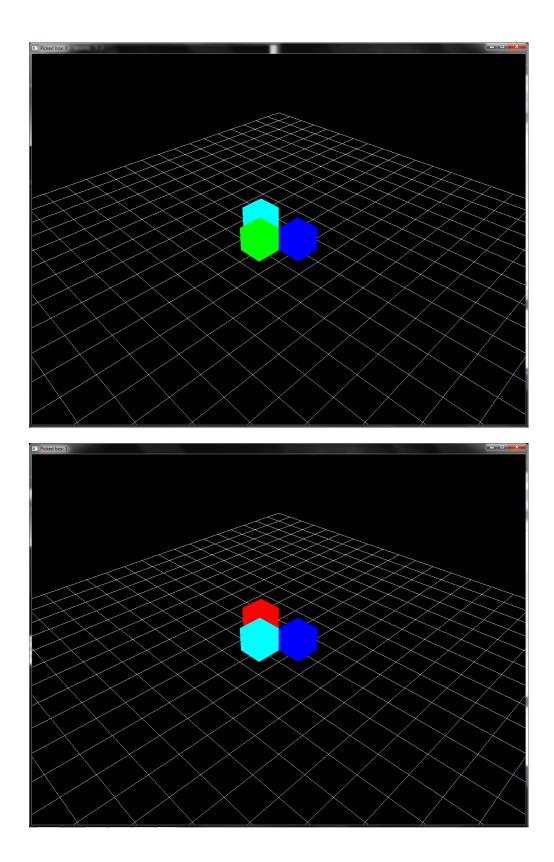


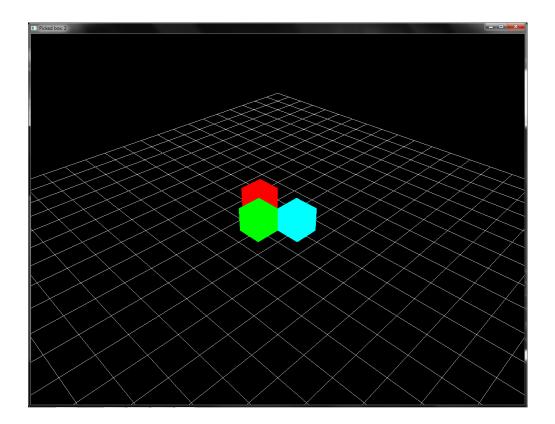


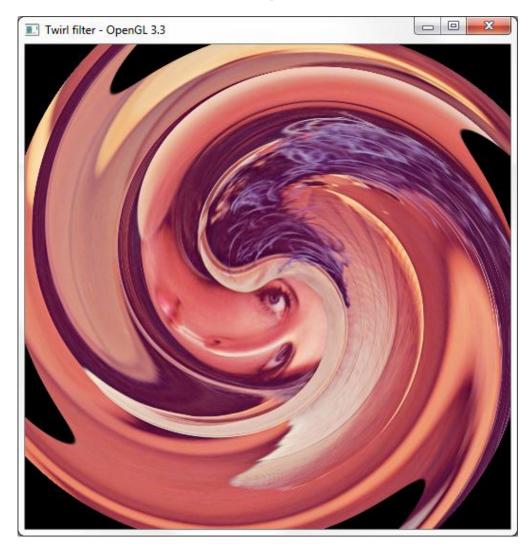






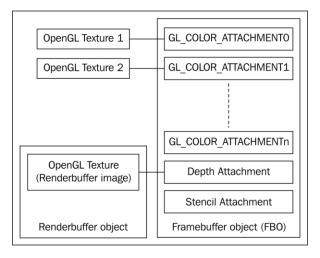


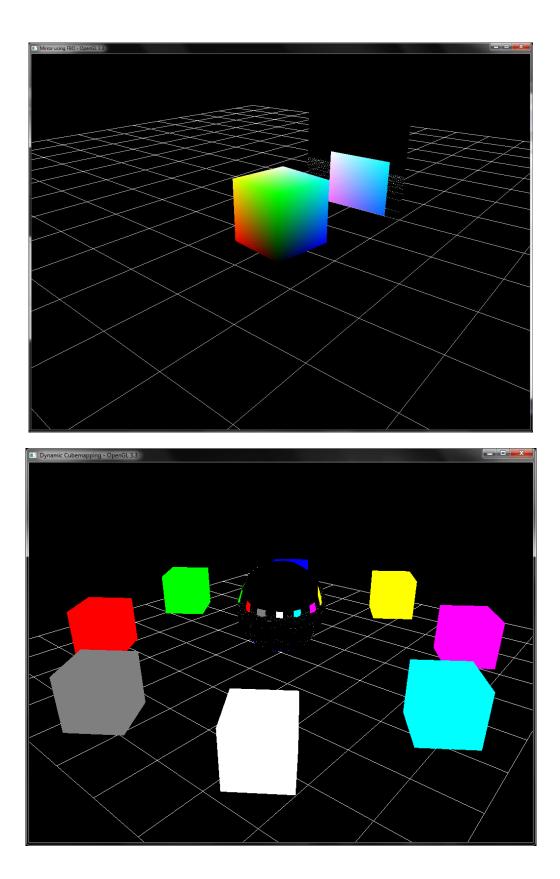


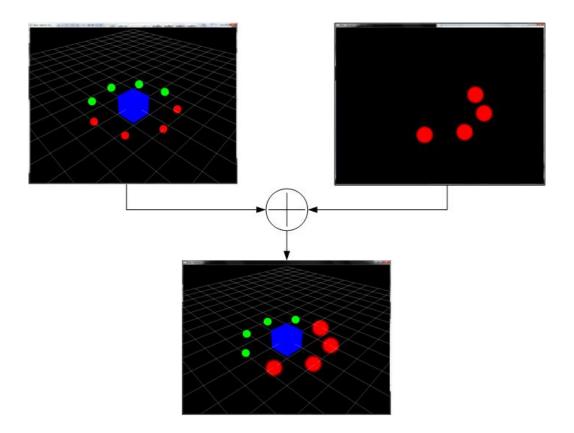


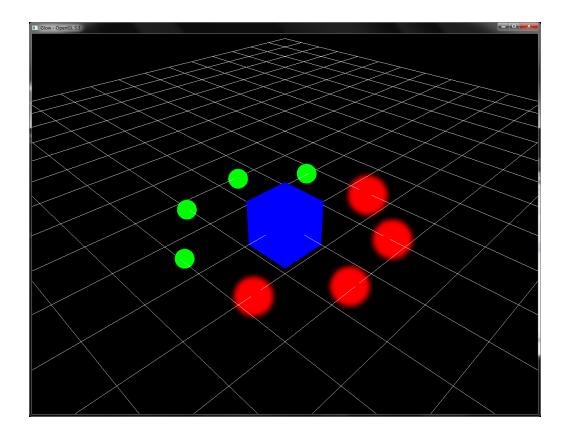
## **Chapter 3: Offscreen Rendering and Environment Mapping**





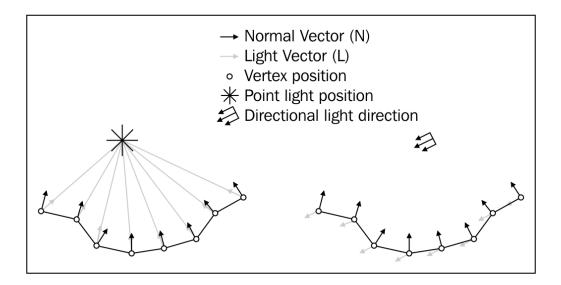


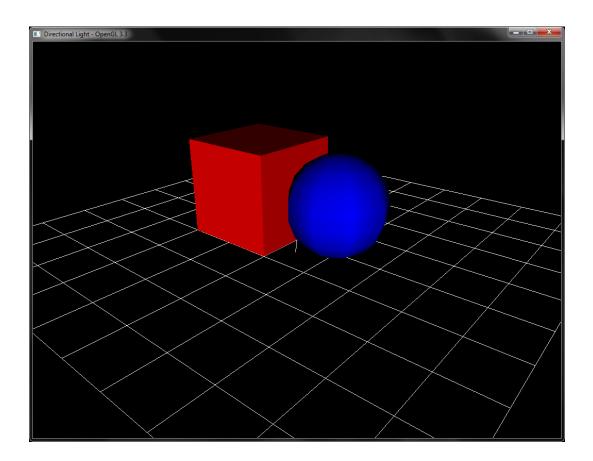


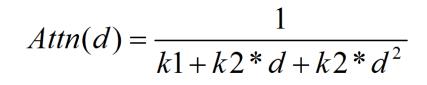


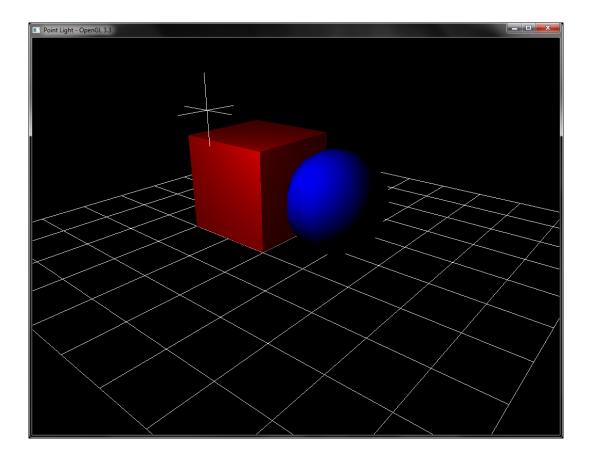
# Per-vertex Lighting - Op Per-fragment Lighting - Op

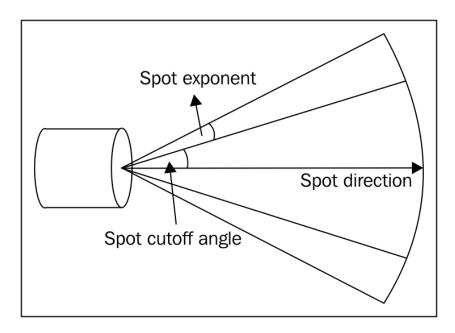
## **Chapter 4: Lights and Shadows**

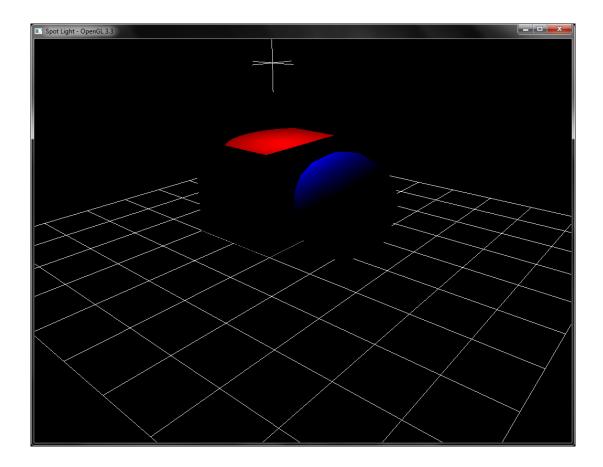




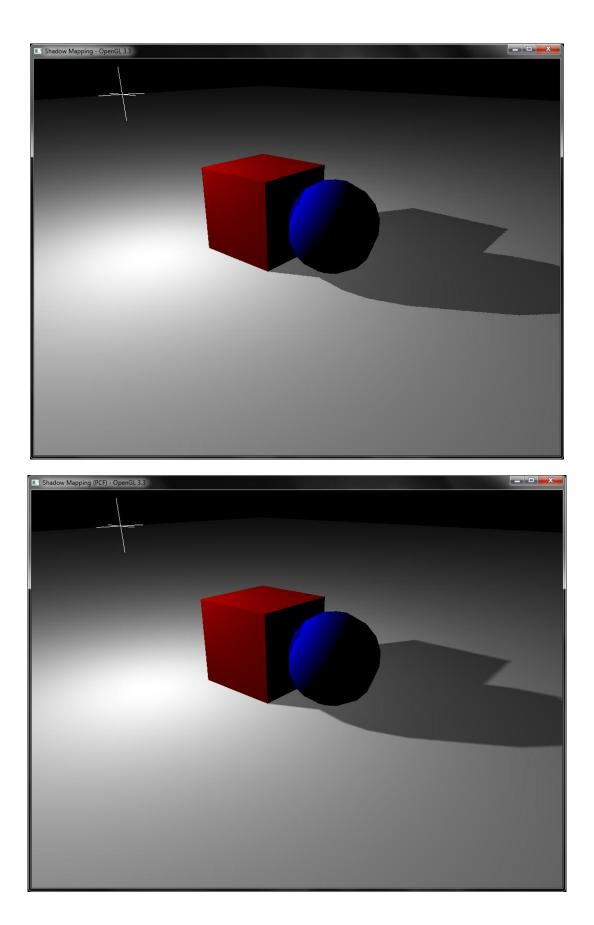


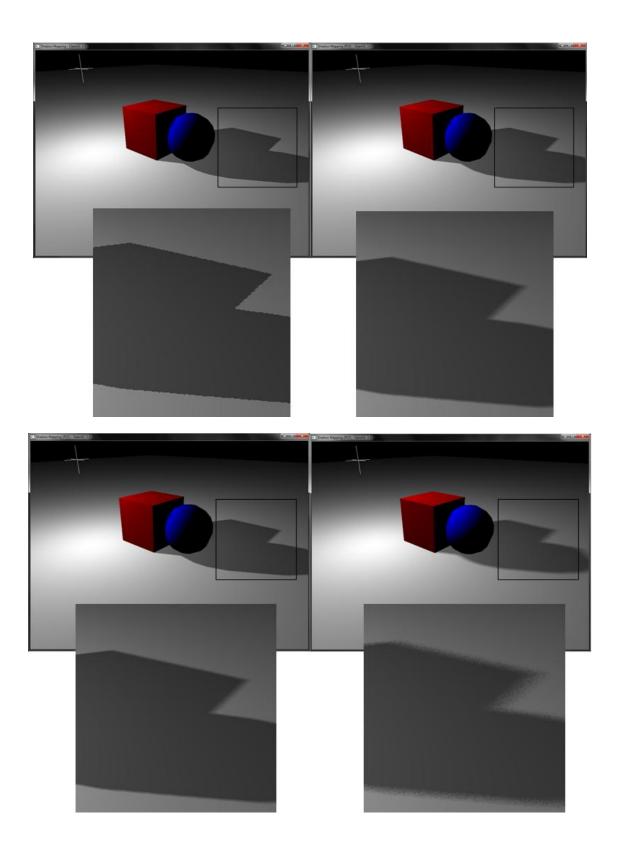


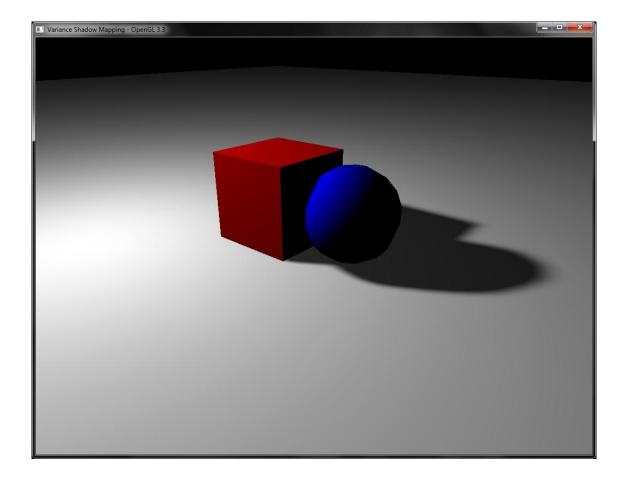


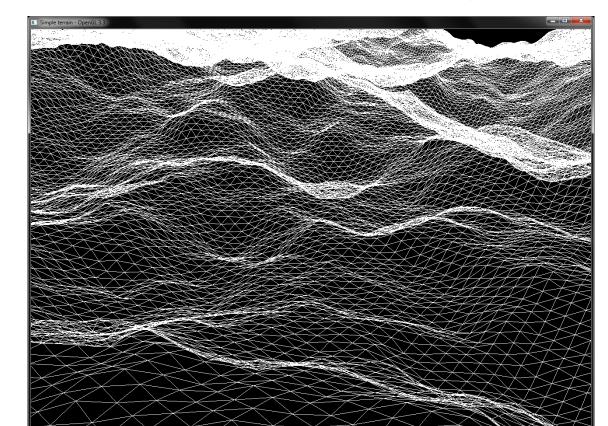


$$UV_{proj} = S * MV * V_{obj}$$
$$S = B * P_L * MV_L$$

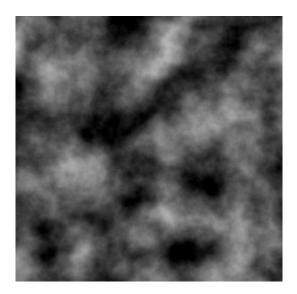


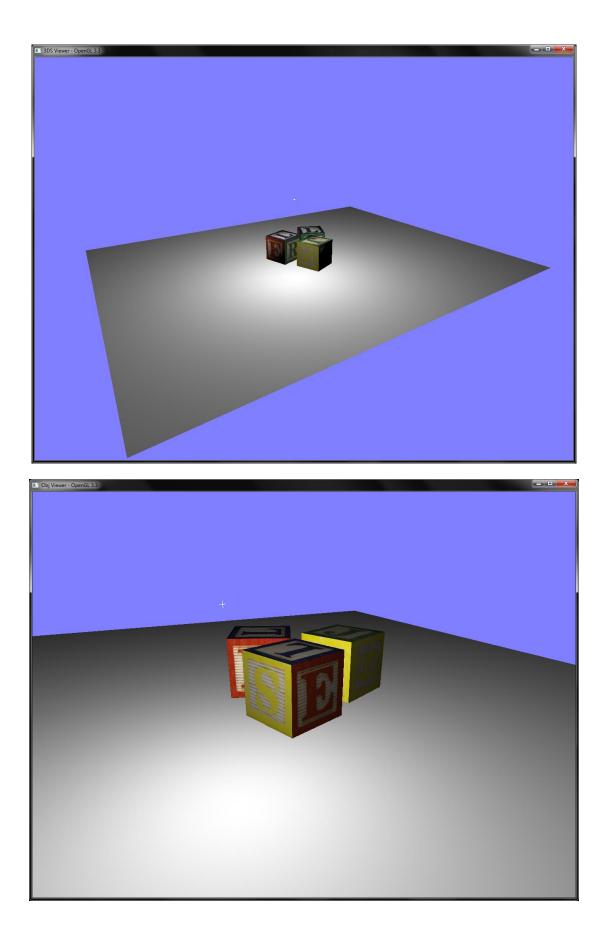


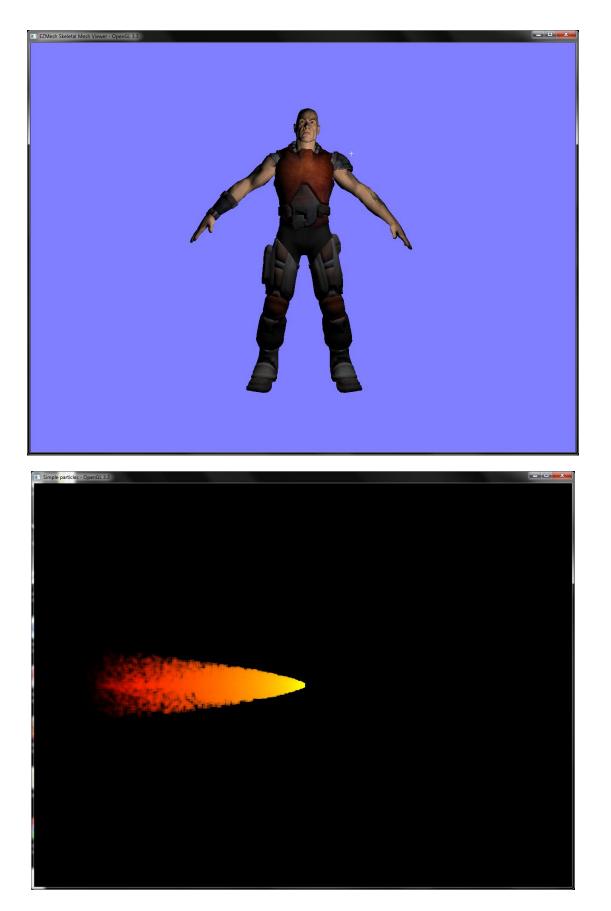


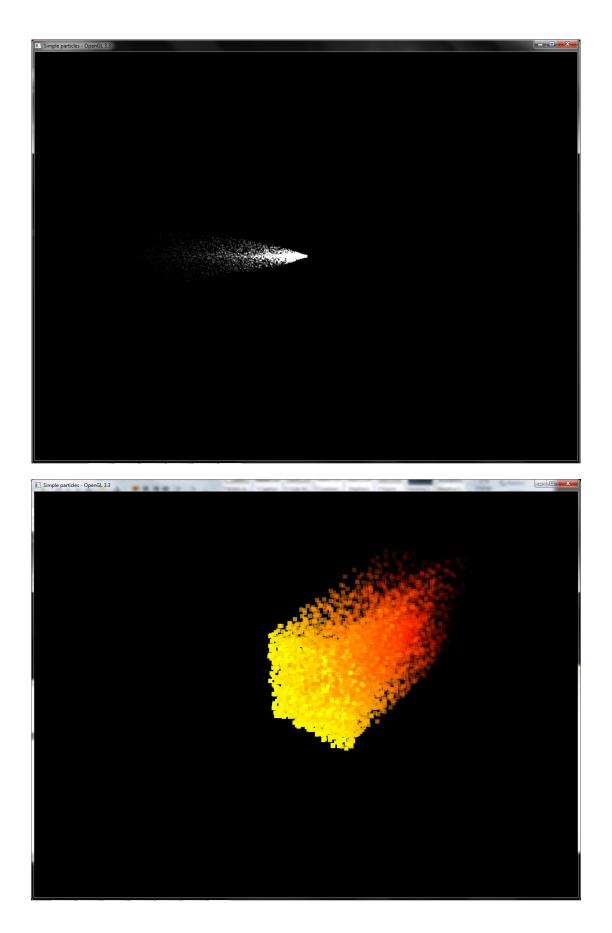


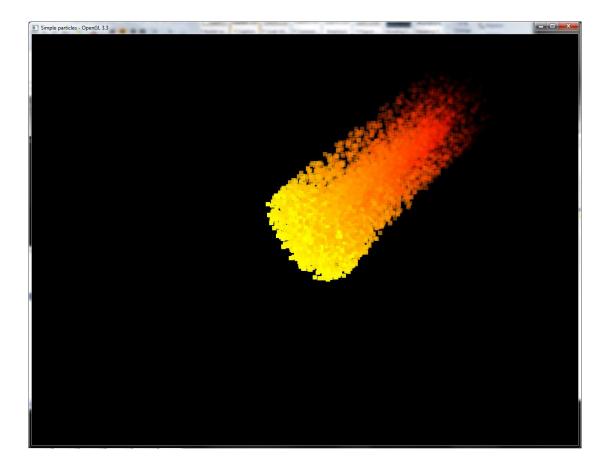
Chapter 5: Mesh Model Formats and Particle Systems



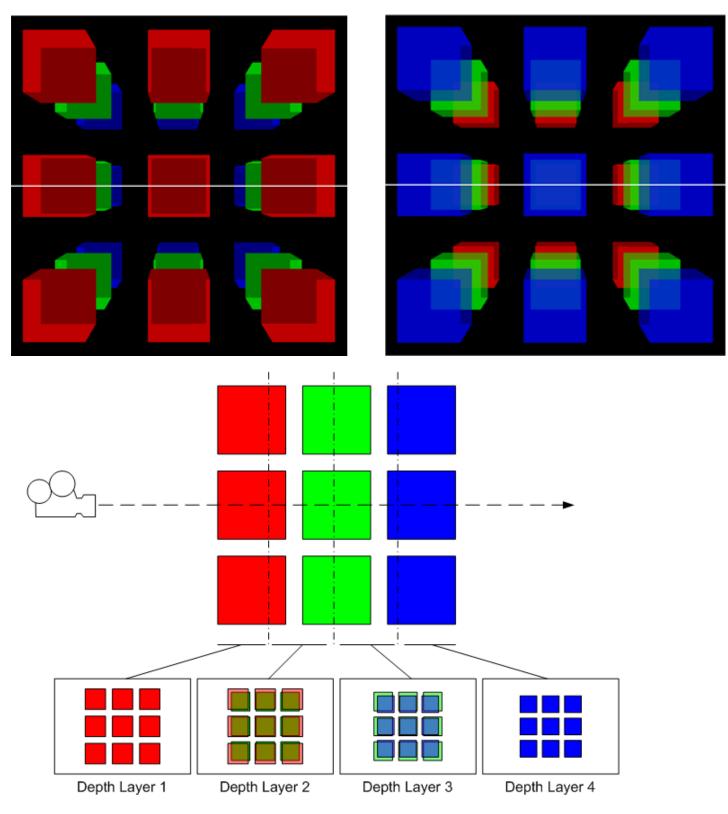


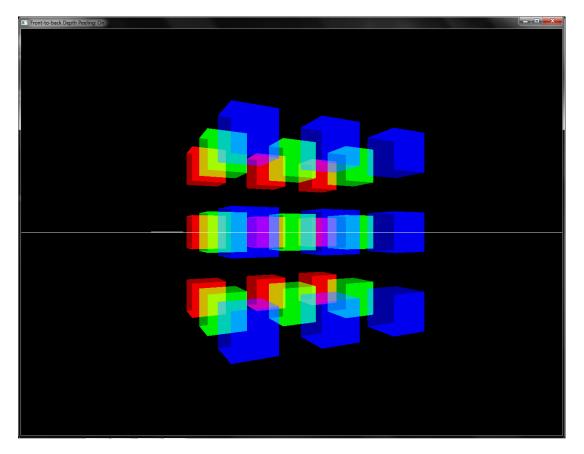


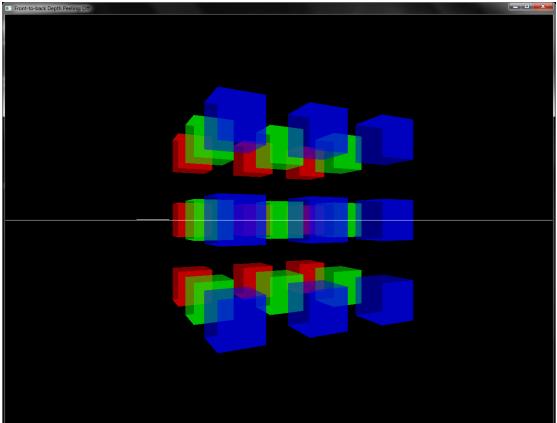


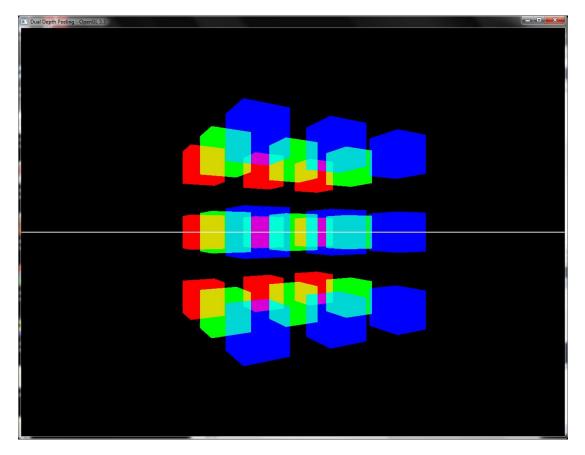


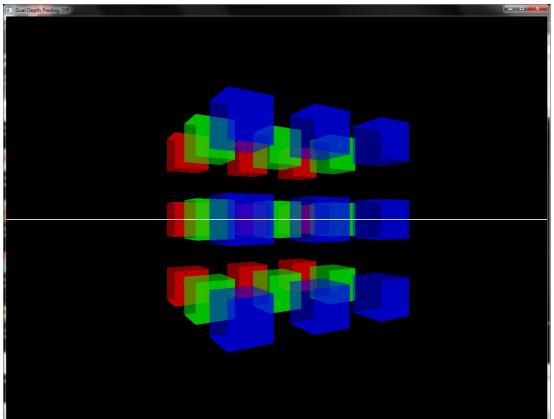
# Chapter 6: GPU-based Alpha Blending and Global Illumination

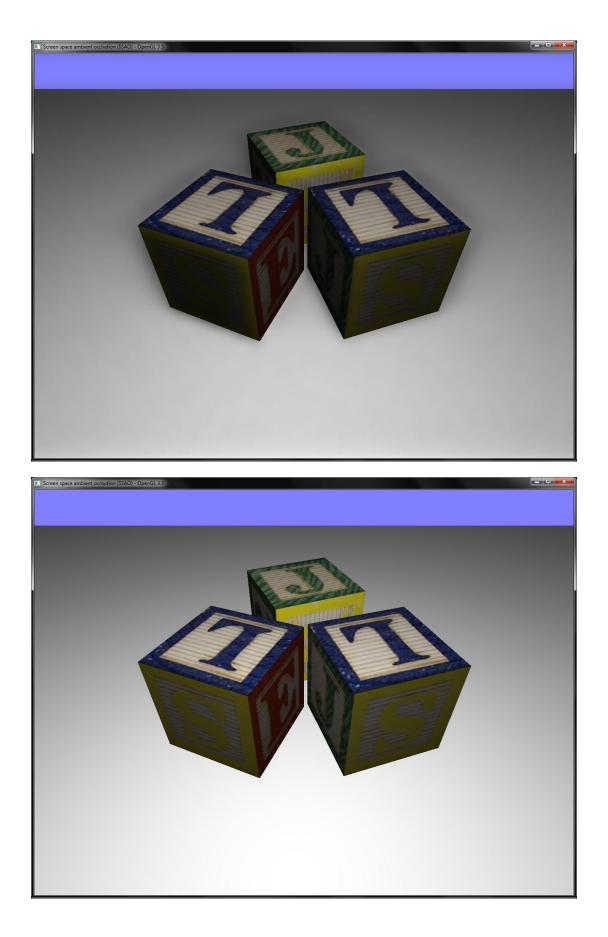


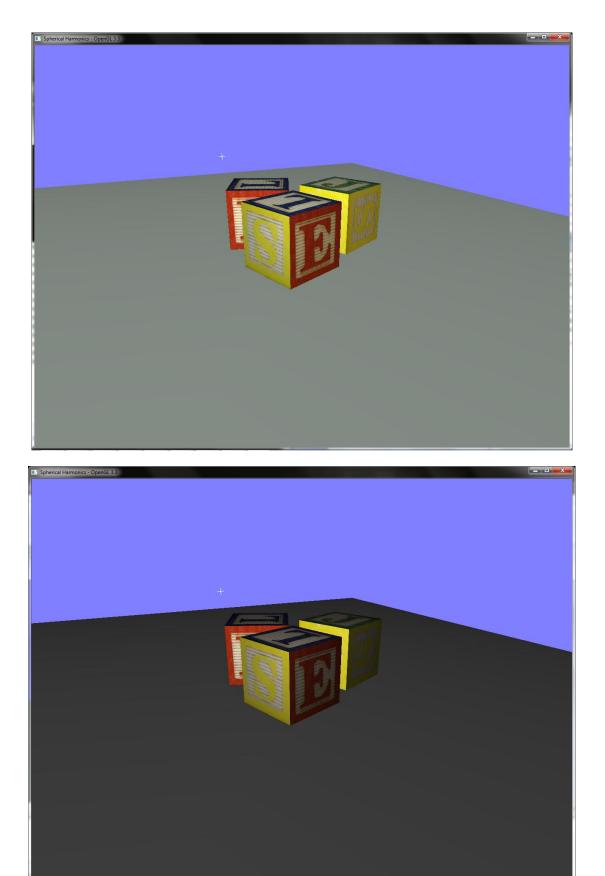


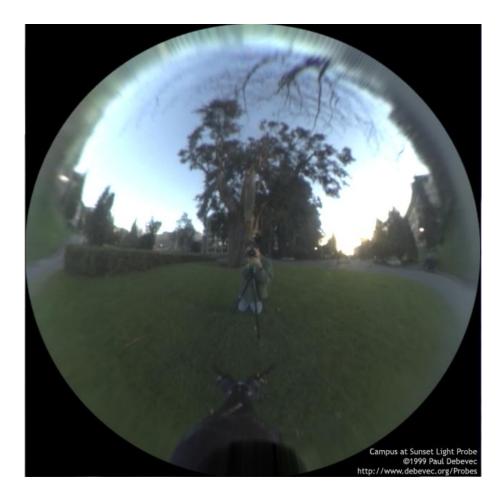


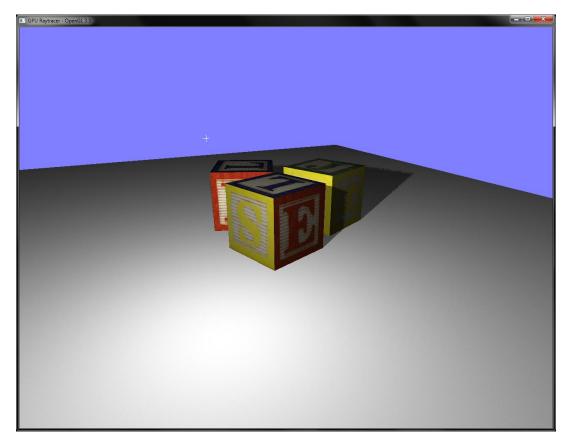


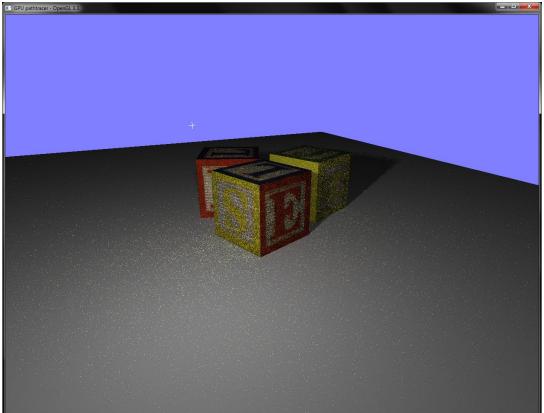




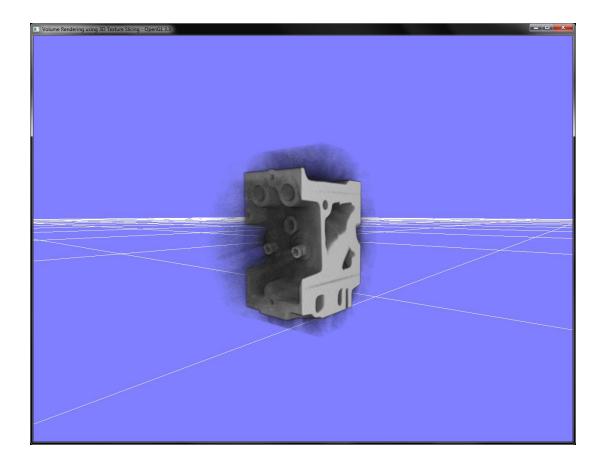


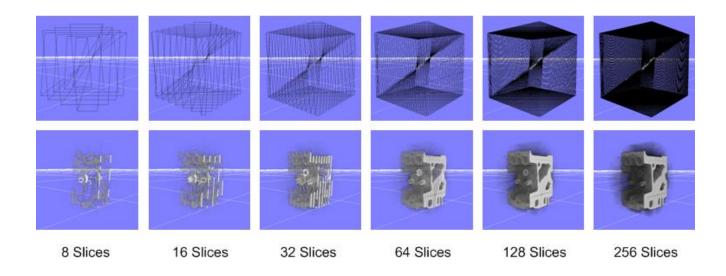


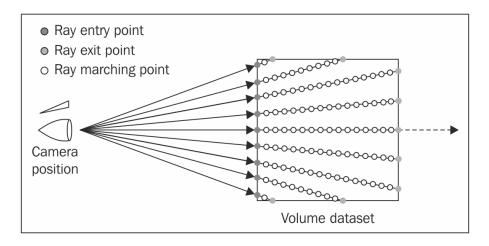


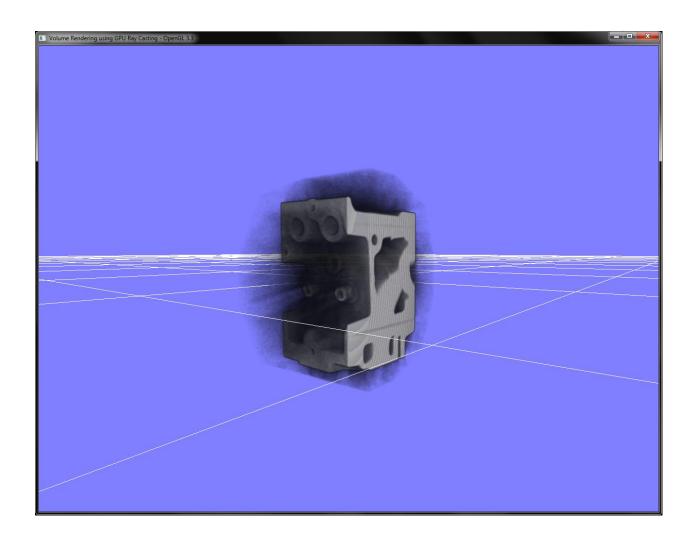


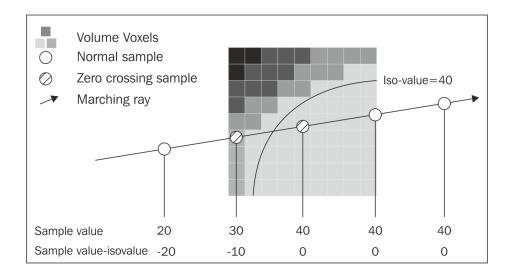
**Chapter 7: GPU-based Volume Rendering Techniques** 

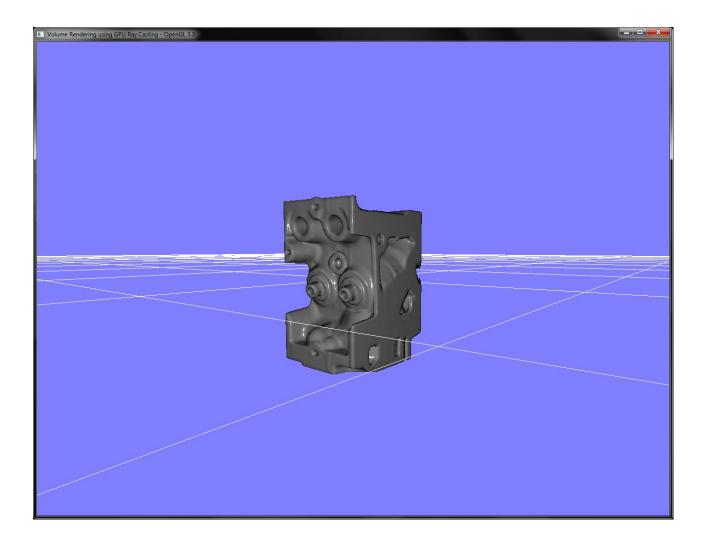


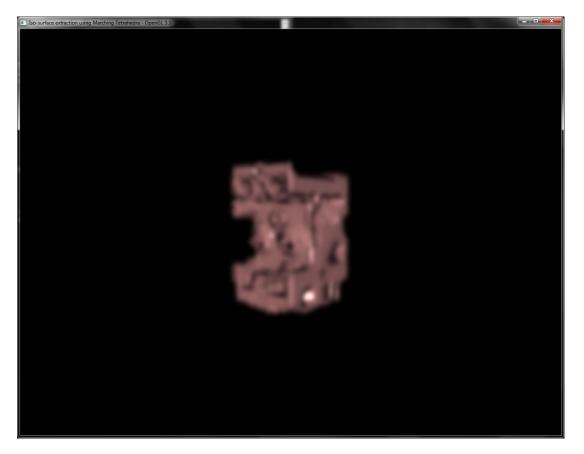


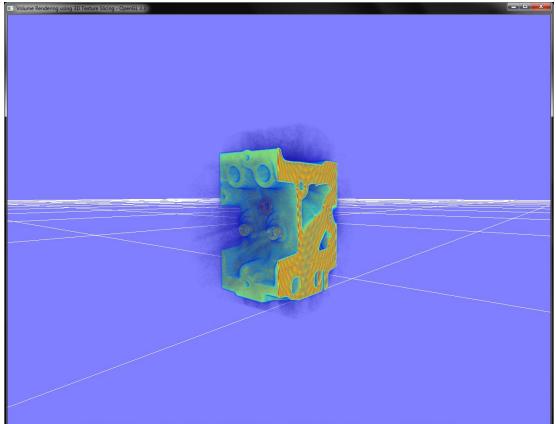


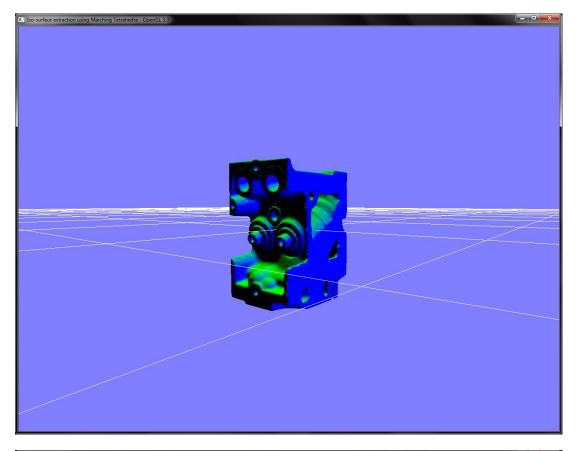


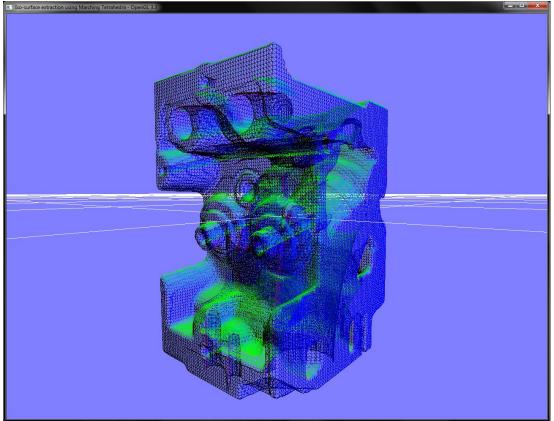


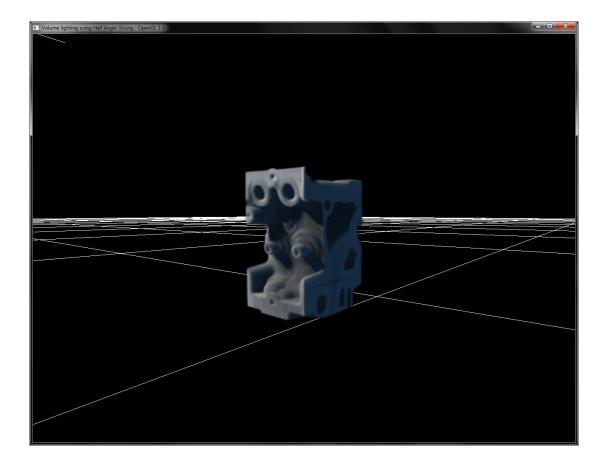












Chapter 8: Skeletal and Physically-based Simulation on the GPU



○ Input vertex

Linear blended vertex

Ø Dual quaternion blended vertex

