

## Bücherliste

 Wichtig: Alle Bücher sind in der Fakultätsbibliothek vorhanden und zum Teil ausleihbar!

- [1] Tomas Akenine-Möller, Eric Haines: **Real-time rendering**, 2. Auflage, Peters, 2006. Die 3. Auflage kann in der Arbeitsgruppe erfragt werden.
- [2] Michael Mortenson: **Mathematics for Computer Graphics Applications**, 2. Auflage, Industrial Press, 2007.
- [3] Richard Wright, Benjamin Lipchak, Nicholas Haemel: **OpenGL SuperBible : comprehensive tutorial and reference**, 4. Auflage, Addison-Wesley, 2007.
- [4] Randi Rost: **OpenGL shading language**, 2. Auflage, Addison-Wesley, 2007.
- [5] Dave Shreiner: **OpenGL programming guide: the official guide to learning OpenGL**, 6. Auflage, Addison-Wesley, 2008.

## Internetquellen

 Wichtig: Es ist empfehlenswert, nicht nur ausschließlich mit Internetquellen zu arbeiten!

 Wichtig: Die Wiki von Referenz [9] ist eine deutsche Übersetzung der OpenGL Reference Pages und stellt zusätzliche Tutorien zur Verfügung.

 Wichtig: Die Folien zu unserer Vorlesung **Advanced Computer Graphics** sind unter Referenz [10] zu finden.

- [6] Song Ho Ahn (2005-2008): **OpenGL**. <http://www.songho.ca/opengl/index.html>.
- [7] Gamedev.net (1997-2011): **NeHe Productions - Everything OpenGL**. <http://nehe.gamedev.net/>.
- [8] Sudhanshu Semwal (2008): **OpenGL Tutorial**. <http://www.cs.uccs.edu/~semwal/indexGLTutorial.html>.
- [9] Florian Sievert (2010): **Delphi Open GL Wiki**. <http://wiki.delphigl.com/index.php/>.
- [10] Matthias Teschner (2011): **Image Processing and Computer Graphics**. <http://cg.informatik.uni-freiburg.de/teaching.htm>.
- [11] Unreal (2000): **OpenGL Programming Guide (Addison-Wesley Publishing Company): Table of Contents**. <http://fly.cc.fer.hr/~unreal/theredbook/>.

# Vorträge

**⚠**Wichtig: **Unbedingt die PowerPoint-Einführung unter <http://cg.informatik.uni-freiburg.de/teaching/powerpoint.ppt> lesen!**

## 1. Einführung (Rendering Pipeline, State Machine, Syntax)

- Theorie, geschichtlicher Abriss
- Rendering Pipeline
- kurz erklärtes Beispielprogramm
- mögliche Quellen:
  - [1], Kapitel 2
  - [3], Kapitel 1 und 2
  - [11], Kapitel 1

## 2. Homogene Koordinaten, Transformationen

- homogene Notation für Punkte, Richtungen, Transformationen
- affine Transformationen (Translation, Rotation, Skalierung, Scherung)
- inverse Transformationen, Verkettung von Transformationen
- mögliche Quellen:
  - [1], Kapitel 1
  - [2], Kapitel 3
  - [8], „Geometric Objects“ und „Viewing“
  - [11], Kapitel 3

## 3. Viewing, Vertex Processing

- Modelview Transformation, Projektive Transformation, Perspektivische Division, Screen Mapping (Viewport Transformation)
- View Volume, Kanonisches View Volume, Orthographische Projektion, Perspektivische Projektion
- mögliche Quellen:
  - [3], Kapitel 4
  - [6], „Transformation“
  - [11], Kapitel 3

## 4. Rasterisierung, Fragment Processing

- Framebuffer
- Pixel, Fragment, Buffer-Arten (Depth, Color, Stencil, ...)
- Rasterisierung von Strecken, Kreisen, Polygonen (Konzept)
- Fragment-Operationen (Scissor Test, Alpha Test, Stencil Test, Depth Test, Blending, Dithering, Logische Operationen and Masking)
- mögliche Quellen:
  - [1], Kapitel 2
  - [3], Kapitel 10 (und 8)
  - [6], Frame Buffer Object (optional)

## 5. Geometrische Primitive, Dreiecksnetze

- Punkte, Linien, Dreiecke zeichnen. Funktion: glVertex(), glBegin() und glEnd()
- Wireframe vs. gefüllte Flächen
- Unterschiede GL\_POINTS, GL\_LINES etc.
- Dreiecksnetz inkl. Datenstruktur, Dreiecksnetz zeichnen
- mögliche Quellen:
  - [8], „Geometric Objects“
  - [11], Kapitel 2
  - G.Scott Owen (1999): **Polygon Mesh Data Structure**.  
<http://www.siggraph.org/education/materials/HyperGraph/modeling/polymesh/polymesh.htm>

## 6. Shading

- Materialeigenschaften (ambient, diffus, spekulär)
- Lichtquellenarten (Punkt, gerichtet) und Lichtquelleneigenschaften
- Phong-Beleuchtungsmodell
- Shading (flat, Gouraud)
- mögliche Quellen:
  - [1], Kapitel 4
  - [7], Lektion 1.5
  - [11], Kapitel 5 und 6

## 7. Texture Mapping

- Textur, Texture Mapping, Texturkoordinaten
- Beispiele
- mögliche Quellen:
  - [1], Kapitel 5
  - [3], Kapitel 5
  - [11], Kapitel 9
  - Nate Miller (2000): **OpenGL Texture Mapping: An Introduction**.  
<http://www.gamedev.net/reference/articles/article947.asp>

## 8. GLUT

- Einfaches Einführungsbeispiel
- Öffnen eines Fensters mit GLUT
- Mainloop
- Wie wird schlussendlich gezeichnet (glutDisplayFunc, glutReshapeFunc)?
- Animation (mit idle)
- evtl. Benutzerevents verarbeiten
- mögliche Quellen:
  - [3], Kapitel 2
  - Lighthouse3D (2011): **GLUT Tutorial**.  
<http://www.lighthouse3d.com/opengl/glut/index.php?1>
  - Elie De Brauwer (2004): **An introduction to GLUT**.  
<http://mindfuck.de-brauwer.be/articles/glut/>

## 9. Shader

- Übersicht OpenGL Rendering Pipeline
- Welche Teile der Pipeline sind durch Shader veränderbar?
- Beispielshader, welche die ursprüngliche Pipeline simulieren
- Shader in der eigenen Applikation einbinden
- kleines Anwendungsbeispiel
- mögliche Quellen:
  - [3], Kapitel 6
  - [4], Kapitel 4 und 6
  - António Ramires Fernandes (2005): **GLSL Tutorial**.  
<http://www.lighthouse3d.com/opengl/glsl/>.