

InsideTime

time for texel

Systemvoraussetzungen

Nichts Besonderes. Ein Rechner mit einer Nvidia Grafikkarte. Da wir aber in unserem Demo Cg shader language verwenden, wird vorausgesetzt, dass die Grafikkarte einen vertex und einen fragment shader unterstützt. Es gibt dadurch Probleme mit mobilen Pcs (Go series).

Die Demo wurde auf einem P3 1,6 GHz mit Geforce 3 Ti 500 entwickelt und auf einem P4 3,0 GHz mit Geforce FX 5600XT getestet. Treiberversion ist jeweils ab ForceWare 81.85 aufwärts. Verwendetes Betriebssystem ist Windows. Entwicklungsumgebung war Visual Studio .NET.

Effekte

Für das Demo haben wir folgende Effekte implementiert:

- (dot3 per-pixel bump mapping)
Mussten wir leider wieder rausnehmen, da es Probleme bei der Beleuchtung gab. Alles grau ist nicht so hübsch.
- Toon Shader
Läuft über ein Lookup in eine 1D Texture auf Grund der Beleuchtung und der Normale. Die 1D Texture enthält einen 3-stufigen Schattenverlauf (in unserem Fall). Die Silhouette machen wir mit Wireframe-Drawing wo wir nur die Rückseite des Objektes zeichnen und das dann auf das Bestehende blenden.
- Volumetric Lines
Läuft über ein Lookup in eine Texture die die diversen Ansichten auf die Line enthält. Je nach Richtung wird dann ein anderes Billboard gerendert.
- Particle System
- Glow Effect
Läuft über „Render-to-Texture“ und dann ein wiederholtes Radial-Blur dieser Texture.
- Dynamic cubemapping
Läuft über ein 6-fach Re-render aus den 6 90° Perspektiven und einem kopieren in eine Texture. Wegen der alten Hardware war Render-to-texture zu aufwändig (neuer OpenGL Kontext usw) und es läuft auch so für diese Szene flott genug.
Die Texture wird dann für das Hauptobjekt verwendet.

Handlung (Scenario)

Die Idee. Wir gelangen über den Bildschirm in das Computer. Drinnen wird uns die Geschichte der Computergraphik erzählt. ;)

1. Cube mit Screenshot
2. Particles
 - Blended Text
3. ToonShader
4. Pong Normal
5. Pong 3D mit Volume Line Shader (CG Shader)
6. Rotierender Text - Geschichte der CG (Glow – Effect)
7. Dynamic Cubic Environment Mapping