

Spiel: Broken – Gruppe: Broken – Repo Glasshouse_Games

Erklärung zur Implementierung

Notiz

WE DID IT!

Gameplay

Steuerung

Die Steuerung ist programmiert für den X-Box 360 Controller. Es besteht eine Möglichkeit, wenn kein Controller angeschlossen ist, diesen über die Tastatur zu simulieren.

Aktion	X-Box 360 Controller	Tastatur-Simulation
Laufen	Linker Analogstick	W-A-S-D
Kamera drehen	Rechter Analogstick	I-J-K-L
Springen	A-Knopf	Leertaste
Dashen/Stürmen	X-Knopf	N
Angriff	B-Knopf	M
Kamera ausrichten	Linker Trigger	Y

Auf Distanzangriff wurde verzichtet.

Inhalt

Das Gameplay ist derzeit sehr simpel gehalten, man steuert einen Helden (Zylinderartiger Körper) mit einem Schwert. Es ist ein Arcade-Game ohne Winning condition – da man aber mit der Steuerung sehr gut spielen kann – ist es ein Spiel!

Effekte

Folgende Effekte wurden implementiert:

Shadow Maps

Die Shadowmaps sind an das OpenGL-Tutorial #16 angelehnt und wurde mit einer eigenen Idee zu PCF ausgebaut. Im Prinzip wird aus Richtung des Directional Lights die Szene einmal mit Tiefeninformationen auf eine Textur gerendert und diese wird mit Multisampling abgefragt um Schatten auf die eigentliche Szene zu zeichnen.

Normalmapping

OpenGL-Tutorials wurden verwendet um auf die richtigen Ideen zu kommen, und da Assimp bereits die Transformation/Generierung der Tagentspace-Komponenten supportet, konnten wir diese Nutzen um einfach zu implementieren.

Vertex Skinning

Vertex Skinning wurde auf maximal zwei Bones per Vertices getrimmt. Allgemein ist es möglich eine Animation abzuspielen. Die Struktur wurde sehr stark an Assimp angelehnt.

Complex Objects

Das Schwert und die Katze.

Animated Objects

Der Hero mit Node-Propaganda in unserer Scene und die Katze mit Vertex Skinning.

View-Frustum-Culling

Wurde nicht implementiert.

Transparency

Außer im Interface wird keine Transparenz derzeit verwendet.

Experimenting with OpenGL

Folgende Features wurden verwendet:

VBOs, VAOs, FBOs, Mip-Mapping, Texture-Sampling-Quality

Features

XBOX360 Controller Steuerung

Tools

Alle Texturen sind per Hand mit GIMP gezeichnet, alle Modelle selbst mit Blender modelliert.

Zusätzliche Libraries

Assimp - <http://assimp.sourceforge.net/>
für Modelloading

GLFW - <http://www.glfw.org/>
für Output, Input, Imageloading

Xinput
für Input unter Windows

GLM - <http://glm.g-truc.net/>
für Mathematik

flexGL – <http://github.com/ginkgo/flexGL>
für OpenGL-Bindings

lodepng - <http://lodev.org/lodepng/>
für PNG-Image loading