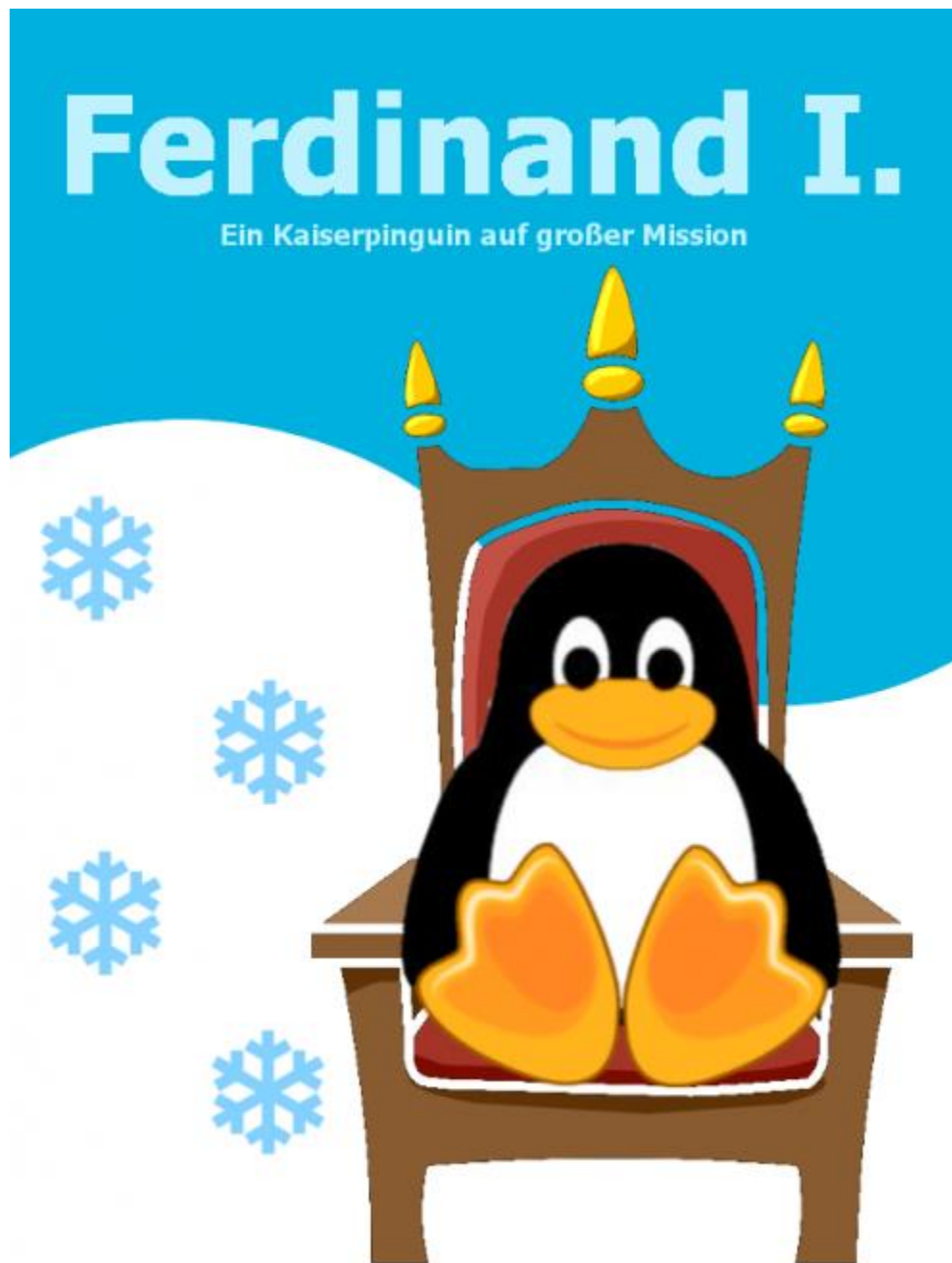


Dokumentation Abgabe 2



Einleitung

Erreiche mit dem Pinguin Ferdinand das Schloss am unteren Ende der Eistrutsche, trotz aller Hindernisse und sammle möglichst viele Medaillen auf dem Weg nach unten.

Steuerung

Spiel Steuerung

Taste	Effekt
A	Pinguin nach links bewegen
D	Pinguin nach rechts bewegen
Leertaste	Springen
Kamerabewegung	Mittels der Maus

Debug Steuerung (zusätzlich zu den vorgegebenen Funktionstastenfunktionen laut Wiki)

Taste	Effekt
C (gedrückt halten)	Stoppen
F6	Cascaded Shadow Map Bereiche anzeigen

Implementierung

Effekte

Cascaded Shadow Maps (mit PCF) – 2,5 Punkte

Für die Schatten unter dem Pinguin, den Medaillen und der Rutsche wird Shadow Mapping verwendet. Die Lichtquelle bildet die Sonne. Aufgrund der großen Ausmaße der Welt kam es in den ersten Versionen durch Perspective Aliasing zu Artefakten. Die zweite Abgabe verwendet daher mehrere Depth Maps in unterschiedlichen Größen, welche das ganze View Frustum abdecken sollen. Je näher die Schatten an dem Spieler sind desto detailreicher sind sie. Man kann sich die Bereiche der verschiedenen Depth Maps durch die Taste F6 einfärben lassen.

[https://msdn.microsoft.com/en-us/library/windows/desktop/ee416307\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ee416307(v=vs.85).aspx)

<http://www.opengldev.org/www/tutorial49/tutorial49.html>

<http://www.lighthouse3d.com/tutorials/view-frustum-culling/geometric-approach-extracting-the-planes/>

<http://www.opengl-tutorial.org/intermediate-tutorials/tutorial-16-shadow-mapping/>

Glow – 1 Punkt

Der Glow Effekt ist an den Medaillen sichtbar, welche sich auf der Rutsche befinden. Hierbei wird der goldene/gelbe Rand dieser für den Effekt verwendet.

http://http.developer.nvidia.com/GPUGems/gpugems_ch21.html

GPU-Partikel System (mit Transform Feedback, Instancing) – 1 Punkt

Die Simulation des Partikelsystems erfolgt anhand eines GPU Partikelsystems mittels Instancing und Transform Feedback. Im Spiel fallen weiße Quadrate mit blauem Rand herab. Die Textur wird auf Billboards angebracht, welche durch das Rendern des Partikelsystems entstehen. Diese Billboards

werden auf den Spieler ausgerichtet. Die Erzeugung der Partikel erfolgt an zufälligen Stellen im Raum und die Partikel fallen mit einer zufälligen Geschwindigkeit zwischen 1 und 5 nach unten.

<http://www.mbsoftworks.sk/index.php?page=tutorials&series=1&tutorial=26> (Der Algorithmus des Shaders zur Generierung einer Zufallszahl zwischen 0 und 1 wurde übernommen)

<http://prideout.net/blog/?tag=opengl-transform-feedback>

<http://ogldev.atspace.co.uk/www/tutorial28/tutorial28.html>

Komplexe Objekte

Der größte Teil der Szene besteht aus komplexen Objekten wie der Pinguin, die Rutschenteile oder das Schloss am Ende. Diese wurden von uns mittels Blender allesamt händisch modelliert und in eine Collada-Datei exportiert. Der Import in das Spiel erfolgt über den Assimp-Importer.

Animierte Objekte

Der Pinguin besitzt animierte Flügel welche sich mit ihm mit und gleichzeitig um ihn herum bewegen. Die Art der Bewegung ist noch nicht ganz perfekt und wird für das Game-Event noch in Richtung einer eher flatternden Bewegung umgeändert.

View-Frustum-Culling

Für jedes Objekt der Rutsche wird aus dem kleinsten und dem größten Punkt die Bounding-Box berechnet und mit dem View-Frustum auf Überschneidungen hin überprüft. Alle es berührenden Objekte werden gezeichnet.

<http://www.lighthouse3d.com/tutorials/view-frustum-culling/geometric-approach-extracting-the-planes/>

<http://www.lighthouse3d.com/tutorials/view-frustum-culling/geometric-approach-testing-boxes/>

Transparenz

Die Platten in der Hälfte der Strecke werden transparent dargestellt und erlauben so einen Blick auf die darunterliegenden Streckenteile.

Experimentieren mit OpenGL

FBOs werden für Effekte wie Glow verwendet.

Die verschiedenen Mip Mapping und Texture-Sampling Qualitätsstufen können mit den Tasten F4 und F5 variiert werden und sind z.B. an dem Himmel oder an der Rutsche sichtbar.

Beleuchtung und Texturen

Alle Objekte in der Szene besitzen Texturen und sind mit Ausnahme der Partikel von einer direktionalen Lichtquelle (der Sonne) von oben beleuchtet.

Zusätzliche Bibliotheken

- Bullet Physics –Physik des Spiels
- FreeImage – Laden der Bilddateien für die Texturen
- Assimp – Importieren der Spielwelt aus einer Collada-Datei

Tools

- Blender – Desgin der Spielelemente und des Levels
- Gimp – Erstellung und Bearbeitung der Texturen