

# Dokumentation

## Die Goldene Liese

### Implementierungsdetails/ Beschreibung

Man steuert einen "Farmer" aus der Egoperspektive. Der Farmer ist mit den Tasten w, a, s, d und der Maus frei steuerbar. Anfangs ist die Welt komplett weiß eingefärbt. Mittels Leertaste kann die Welt Stück für Stück freigespritzt werden. Ziel ist es die Kuh zu finden und zu berühren. (Tipp: Um die Kuh leichter zu finden in den wireframemodus (F3) umschalten und sich in die Nähe der Kuh bewegen)

### Features

Die F1 - F9 Tasten wurden gemäß den Anforderungen implementiert. (Ausgabe in der Konsole)

### Beschreibung von Lichtquellen und Texturen

Wir haben eine Lichtquelle mit phong shading implementiert. Für jedes Objekt sind die Parameter shininess und specularColor setzbar.

Sowohl die Heightmap, als auch die Kuh ist mit einer Textur belegt. Mittels UV Koordinaten hat die Kuh permanente Schattierungen erhalten. Die permanenten Schattierungen kommen durch Ecken, Kanten und Rundungen am Objekt zustande. Bei der Heightmap lässt sich leicht einstellen mit welcher Rate sich das geladene Oberflächenmuster wiederholen soll.

### Zusätzliche Libraries

- Um Objekte zu laden verwenden wir die Assimp Library (<http://assimp.sourceforge.net/>)
- Um Texturen und Heightmap zu laden verwenden wir die FreeImage\_Lib (<http://freeimage.sourceforge.net/>)
- Für die Collision Detection und die physikalische Berechnung der Objekte verwenden wir Bullet (<http://bulletphysics.org/>)

### Effekte (+Implementierungsdetails)

#### Shadow Mapping

- Wir erzeugen eine DepthTexture in der die Entfernung sowohl der Kuh, als auch des Terrain zur Sonne gespeichert werden. Mit Hilfe der DepthTexture werden im mask\_shader dynamisch Schatten erzeugt. Die Schatten sind erst sichtbar, wenn man die Farbe an der Stelle abgewaschen hat!

#### Lens flare

- In unserer Welt haben wir Lensflare (von der Sonne ausgehend) unter Zuhilfenahme von verschiedenen Tutorials implementiert. (<https://entwickler.de/online/opengl-weltraumeffekte-mit-motion-blur-lens-flare-co-how-to-114796.html>),

<http://john-chapman-graphics.blogspot.co.at/2013/02/pseudo-lens-flare.html> ) Dazu haben wir einen zweiten Renderpass eingeführt. Die komplette Welt (ausgenommen Sonne) wird zuerst in einen neuen Framebuffer in eine 2D Farb und Tiefentextur gerendert. Die Sonne wird wieder in einen extra Framebuffer gerendert.

- Anschließend wird die Distanz der Sonne zum Mittelpunkt des Fensters berechnet. (Berechnung in Screen Space). Ist die Sonne sichtbar, werden ghosts und halos der Sonne erzeugt. Dazu wird die Sonnentextur entlang des Vektors, der von Mittelpunkt des Fensters bis zur Sonne geht, dupliziert, transformiert und skaliert.
- Je weiter die Sonne vom Mittelpunkt entfernt ist, desto transparenter sind die angezeigten ghosts und halos.
- Die ghosts werden außerdem mithilfe einer farbigen Textur eingefärbt. (je nachdem an welcher Position sie sich befinden, ändert sich die Farbe)
- zusätzlich werden die verschiedenen Farbkanäle der ghosts um verschiedene kleine offsets verschoben, um einen realistischeren Effekt zu erzielen.

#### Partikelsystem

- Um den Wasserstrahl zum Abwaschen der Farbe zu simulieren haben wir ein Partikelsystem implementiert.
- Die einzelnen Partikel sind transparent und mit einer Textur belegt.
- Mit Parametern lassen sich Richtung, Dichte, Lebenszeit, etc. einstellen.

#### Welt Schritt für Schritt "freispritzen"

- Jedes Objekt ist Anfangs mit einer dynamisch erstellten Textur (= "Maske") belegt.
- Schießt der Spieler mit Wasser wird mittels Raytracing der Physikbibliothek Bullet berechnet wo der Strahl auftritt.
- Wird die Kuh getroffen, wird das Abwaschen der Kuh mit langsamen verschwinden (über ein paar Frames hinweg) der Maske simuliert.
- Wird das Terrain getroffen wird ein jedes mal leicht variierendes Muster generiert. Die Maske wird rund um den getroffenen Punkt um das generierte Muster entfernt und die darunter liegende Textur kommt zu Vorschein.

#### **Sonstige Special Features**

- Der Schwanz der Kuh wedelt fröhlich hin und her. Die Kuh bewegt sich willkürlich in der Welt.
- Wir haben eine Skybox implementiert.
- Wir können ein Terrain aus einem Grauwertbild erstellen, haben also eine Heightmap implementiert.

#### **Tools für die Modelcreation**

Die Models wurden in Blender erstellt und bearbeitet. Anschließend wurden sie als .obj Files und getrennten Textur Bildern (.jpeg/.png) exportiert.

Die Kuh wurde von google warehouse heruntergeladen und bearbeitet.

(<https://3dwarehouse.sketchup.com/model.html?id=ed279e2538fb1022aa2750145c6ad94f>)