

Commander Cookie Submission 2

Requirements:

- Gameplay

Im Spiel sollen die Cookies davor bewahrt werden, von den Fliegern durch Durchfliegen zerstört zu werden. Die Cookies haben jeweils 4 Livepoints und verlieren jedesmal einen davon, wenn es zum Kontakt mit einem Flugzeug kommt, nachdem sie das letzte Leben verlieren, ist das Spiel verloren. Wenn Flieger abgeschossen werden und man bisher weniger als 4 Flieger durch Abschießen zerstört hat, spawnt ein neuer auf einer von zwei zufälligen Flugrouten. Werden alle 4 Flieger abgeschossen, bevor einer der Kekse zerstört wird, hat man gewonnen.

- Effects

- Cel-Shading: Der für die Beleuchtung relevante Winkel zwischen Lichtquelle und Oberflächennormale wird auf wenige mögliche Werte diskretisiert. Dadurch entsteht eine abgestufte Beleuchtung.

- Contures (edge detection): Beim Rendern der Szene aus der Kameraperspektive, werden neben den Farben auch die Normalen der Oberflächen in eine Textur geschrieben. Aus dieser Textur werden durch Kantendetektion Konturen erzeugt, welche auf dem Farbbild sichtbar gemacht werden.

- Edge glow (bloom): Alle sichtbaren Flugzeuge der Szene werden beim Rendern zusätzlich, einfarbig in eine zusätzliche Textur gezeichnet. Durch Kantendetektion auf dieser Textur können so die Außenkanten der Flugzeuge erkannt werden und nach außen hin verwischt werden.

- Shadow-mapping: Bevor die Szene aus der Kameraperspektive gezeichnet wird, wird sie aus der Perspektive der Lichtquelle betrachtet. Durch die entstandene Textur kann festgestellt werden, ob sich ein Fragment im Schatten oder im Licht befindet. Alle nichtdurchsichtigen Objekte der Szene werden beleuchtet.

- Texture Mapping

Alle nichtdurchsichtigen Objekte der Szene werden durch Texturen eingefärbt. Das Einlesen der Texturen aus Bildfiles übernimmt die externe Bibliothek ResIL (<http://sourceforge.net/projects/resil/>).

- Complex Objects

Die Modelle wurden aus dem Internet (<http://tf3dm.com/> für Tank und Plane, <https://3dwarehouse.sketchup.com/> alles andere) heruntergeladen und mit Blender nachbearbeitet und zu einer Szene zusammengefügt. Mit der externen Library Assimp (<http://assimp.sourceforge.net/>) wurden diese Modelle dann einzeln mit dem Triangulation und dem Pre-Transform-Vertices Flag geladen und daraus GameObjectTemplates erstellt, die dazu benutzt werden um die GameObjects zu erstellen. Parallel zum ModelLoading werden im gleichen Schritt auch die Modelle verwendet um Informationen für die RigidBodyes, die für die CollisionDetection verwendet werden zu extrahieren..

- Animated Objects

Der Panzer und seine Bestandteile sind hierarchisch animiert. Der Turm und das Kanonenrohr folgen immer der Position des Panzers. Der Turm dreht sich immer mit der Kamera mit und das Kanonenrohr folgt immer der Neigung der Kamera.

- View-Frustum-Culling

Alle Spielobjekte besitzen eine umschließende „bounding sphere“. Befindet sich diese nicht innerhalb des view-frustums, so wird das Objekt auch nicht in die Szene gerendert. Ist der Wire-Frame-Mode(F3) aktiviert, so sieht man die Anzahl der gerenderten Objekte in der Konsole.

- Transparency

Nachdem alle nichtdurchsichtigen Objekte in der Szene gezeichnet wurden, wird zusätzlich eine durchsichtige Fensterscheibe in die Szene gezeichnet.

- Experiments

Um die implementierten Effekte umzusetzen, wurden mehrere FBOs verwendet, um die Szene auf unterschiedliche Wege zu rendern. Ein FBO nutzt „multiple rendertargets“ um in einem Durchlauf in mehrere Ausgabertexturen, unterschiedliche Daten zu schreiben.

Features:

- Flugzeuge folgen Bezierkurven

In CCMath.java sind Funktionen implementiert durch die Bezierkurvenverläufe berechnet werden können. Diese werden genutzt um die Flugpfade der Flugzeuge zu beschreiben.

- Bullet Collision Detection

Die Kollisionserkennung wird von der externen Bibliothek „Bullet“ übernommen.

- Hintergrund

Außerhalb des Zimmers befindet sich eine texturierte Ebene, mit der die Außenwelt dargestellt wird. Um den Anschein zu erwecken, dass die Häuser im Hintergrund weit vom Betrachter entfernt sind, bewegt sich die Ebene immer mit der Kamera mit.