Dokumentation

CSE Projekt – Martin Steinberger

Bei dem Programm handelt es sich um eine schlichte Demo welche die Anforderungen unserer Übung beinhält. In den folgenden Punkten wird kurz erläutert wie die einzelnen Schritte behandelt wurden.

# Aufbau

Das Programm wurde mit OpenGL in Verbindung mit SDL geschrieben. Grafische Objekte werden im Programm als „Vertex Array Objects“ an die Shader geschickt, dort werden diese Transformiert und der Lichteinfall darauf berechnet. Der Code stammt zu einem großen Teil von <https://learnopengl.com/> da dort Ausgezeichnete Tutorials für die Verwendung von OpenGL in Verbindung mit Shadern vorgestellt werden. Mein Ansatz versucht die in den Tutorials beschriebenen Abläufe Objektorientiert umzusetzen daher ergeben sich einige Klassen welche für die Funktionalitäten des Programms zuständig sind.

## Mesh.h / Mesh.cpp

Die Mesh Klasse ist ein Container welcher Referenzen auf Vertex Array Objects, Texturen sowie eine Transformationsmatrix und Methoden um diese einfach zu bearbeiten enthält. Ich habe versucht die Mesh Klasse so gut wie möglich von OpenGL zu entkoppeln und alle OpenGL Funktionen in den Referenzierten MyVAOs unterzubringen.

Der Vector welcher die Referenzen auf VAOs (Vertex Array Objects) enthält stellt die verschiedenen LODs dar. Bevor ein VAO gezeichnet wird, wird die Entfernung von der Kamera berechnet und dementsprechend ein bereits auf der Grafikkarte hochgeladenes VAO gezeichnet. Bevor das VAO gezeichnet wird, setzt das Mesh-Objekt die Textur, Transformationsmatrix und Transparenz des VAO neu. So können mit demselben VAO auf der Grafikkarte optisch unterschiedliche Objekte gezeichnet werden.

## MyVAO.h / MyVAO.cpp

MyVAO ist eine Klasse welche die OpenGL Funktionen zum Zeichnen von VAOs in Methoden verpackt. Die Klasse lädt in ihrem Konstruktor die benötigten Daten auf die Grafikkarte