CGI: BERICHT ZU AUFGABE 2, RAYTRACER

VON SEBASTIAN DASSÉ, MAX NOVICHKOV, SIMON LISCHKA

1. Aufgabenstellung

1.1. Implementierung der beiden Kameras OrtographicCamera und PerspectiveCamera

1.2. Schreiben einer Color-Klasse

1.3. Entwerfen einer abstrakten Geometrie-Superklasse Geometry und der entsprechenden Unterklassen Plane, AxisAlignedBox, Triangle und Sphere sowie eines Hit Objekts.

1.4. Entwerfen einer Welt-Klasse, die alle Objekte der Szene enthlt

1.5. Entwerfen der Raytracer-Klasse, die ber die Objekte der Welt iteriert

1.6. Testen der Geometry-Objekte durch entsprechende Beispielkonfigurationen

2. Lösungsstrategien

2.1. Die Klassen wurden zunächst als leere Vorlagen generiert.

2.2. Die entsprechenden Klassen wurden zunächst auf Papier gerechnet und dann gemeinsam im linearen Zeitablauf implementiert.

2.3. Die Raytracer und View Objekte wurden möglichst generisch modularisiert.

3. Besondere Probleme

3.1. Einsetzen der Werte aus der selbsterstellten Colorklasse in das WriteableRaster.

3.2. Implementierung der Axis-Aligned Box.

4. Implementierung

4.1. Die Kameras, Color-Klasse und Geometrieklassen, sowie die Hit Klasse wurden anhand der im Unterricht vorgegebenen Methodik und dem Buch Ray Tracing From Ground Up von Kevin Suffern umgesetzt.

4.2. Der Raytracer gibt mit der trace() Methode ein BufferedImage Objekt aus, das mit der UI-Klasse ShowImage auf einem Canvas dargestellt wird. Zur Bequemen Erzeugung der Objekte Graphikobjekte wurde die Factoryklasse rayter.tests.graphical.Factory erstellt.

Date: 24.11.2013.

5. Zeitbedarf

5.1. Der Zeitbedarf wurde bei allen Teammitgliedern mit jeweils 25 Stunden angegeben. Dabei wurde viel Zeit mit mathematischen Überlegungen aufgewandt.