

Bericht zu den Schnittberechnungen

Trayracer: Oliver Kniejski, Steven Sobkowski, Marie Hennings

23. November 2015

Die Aufgabenstellung

Die aktuelle Aufgabe ist es einen ersten Raytracer zu programmieren. Dieser soll übergebene Szenen, also Welten mit einem oder mehreren Objekten, darstellen können. Zur Realisierung der Aufgabe müssen folgende Klassen geschrieben werden:

- Die abstrakte Kamera Klasse, die von perspektivischer und orthographischer Kamera Klasse implementiert wird.
- Die Ray Klasse, die einen bestimmten Strahl darstellt.
- Die abstrakte Klasse Geometry, die von folgenden Geometrien implementiert wird: Plane, Sphere, Triangle und Axis Aligned Box.
- Die Hit Klasse, die den Schnittpunkt eines Strahls mit einer Geometrie abbildet.
- Die Color Klasse, die eine RGB-Farbe darstellt und die einzelnen Farbanteile durch eine Zahl zwischen 0 und 1 angibt.
- Die World Klasse, zu der man Geometrien hinzufügen kann
- Die ausführbare Raytracer Klasse, die pixelweise die ausgewählte Welt mit ihren Objekten in ein vorgegebenes Fenster zeichnet.

Lösungsstrategien

Vor Beginn der Bearbeitung der Aufgabe wurden alle Klassen auf die einzelnen Teammitglieder aufgeteilt. Dabei wurde darauf geachtet, dass jeder etwa gleich komplexe Aufgaben erhält. Anschließend wurden alle Klassen gemeinsam überprüft und mögliche Fehler und Probleme zusammen gelöst.

Implementierung

Die Klassen wurden anhand der in der Aufgabe vorgegebenen Klassendiagramme implementiert. Einzig die Color Klasse erhielt eine zusätzliche Methode, um die Farbe als RGB-int auszulesen.

Für den Raytracer wurde als UI-Bibliothek das JavaFX-Framework verwendet.

Besondere Probleme oder Schwierigkeiten bei der Bearbeitung

Anfangs wurde im Raytracer das Bild auf dem Kopf stehend dargestellt. Die Lösung des Problems bestand darin, die Bildaufbauschleife vertikal umgekehrt zu durchlaufen. Bei der Visualisierung der Ebene fiel auf, dass die Überprüfung des t vergessen wurde. Auf die gleiche Art wurde ein Fehler bei der Würfelimplementierung festgestellt. Dies zeigt, dass das Debugging mittels Visualisierung im Raytracer um einen neuen Aspekt erweitert wird. Das α wurde in der Formel für die perspektivische Kamera in den Vorlesungsfolien unklar definiert. An dieser Stelle muss der Öffnungswinkel halbiert werden.

Zeitbedarf

Der Zeitbedarf war trotz weniger Probleme verhältnismäßig hoch. Jedes Gruppenmitglied verbrachte ca. 3 Stunden in der Übung und zusätzlich noch einmal 4-5 Stunden mit den Aufgaben.