

Raytracing Aufgabe 04 - Beleuchtung II

Benjamin Schurian, Clemens Pade, Robert Dimitrov
Computergraphik I

WS2015/16

1 Einführung

In dieser Aufgabe ist der zweite Teil der Beleuchtung für den Raytracer zu implementieren - es werden nämlich das Schatten und die Reflexion hinzugefügt. Dafür sollten die Klassen, die in der vorherigen Aufgabe geschrieben wurden, um einige Elemente erweitert werden, sowie einige neue Klassen implementiert werden. Außerdem sollte man am Ende zwei Demo-Szenen erstellen, die als Tests für die Licht- und Materialklassen dienen. Jedes der Teammitglieder übernahm eine dieser drei kleinen Teilaufgaben.

2 Teilaufgabe 1

In der ersten Teilaufgabe sollten die Lichtklassen verändert werden. Ein zusätzliches Attribut `boolean castshadows` gibt an, ob das entsprechende Licht Schatten wirft. Alle Konstruktoren der Licht-Unterklassen wurden um dieses Attribut erweitert. Außerdem wird in der `illuminates`-Methode ein World-Object benötigt, damit man ermitteln kann, ob ein Objekt zwischen Lichtquelle und dem Punkt auf der Oberfläche ist.

Da diese Teilaufgabe relativ leicht ist, hat sie uns keine Probleme bereitet. Länger hat die Einarbeitung ins Thema gedauert.

- **Teammitglied:** Benjamin
- **Einlesen:** 30 Minuten
- **Programmieren:** 10 Minuten

3 Teilaufgabe 2

In dieser Teilaufgabe wurden die Klassen für das Material verändert - die Methode `colorFor` bekam als zusätzlichen Parameter einen Tracer, ein Objekt, das eine Funktion zum rekursiven Raytracen zur Verfügung stellt. Außerdem wurde die Klasse `ReflectiveMaterial` geschrieben, die das Material für einen perfekt diffus reflektierenden Körper mit Glanzpunkt und Reflektion darstellt.

Diese Klasse ist der `PhongMaterial`-Klasse sehr ähnlich, sie besitzt aber ein weiteres Attribut für die Farbe der spiegelnden Reflektion des Glanzpunkts. Zudem wird bei der `colorFor`-Methode auch die Farbe drauf addiert, die die Funktion $fr(r)$ des `Tracer`-s zurückgibt.

Diese Teilaufgabe ist uns am schwierigsten gefallen. Vor allem bei der `Tracer`-Klasse mussten wir lange über die Implementation nachdenken. Dabei haben alle Teammitglieder mitgeholfen. In der Material-Klasse sind auch ein paar Probleme (aus Versehen) entstanden, die aber leichter zu beseitigen waren.

- **Teammitglied:** Robert (Hilfe von Benjamin und Clemens)
- **Einlesen:** 30 Minuten
- **Programmieren:** 3 Stunden

4 Teilaufgabe 3

Abschließend sollten zwei vom Dozenten beschriebene Demo-Szenen erstellt werden, wie bei den vorherigen Aufgaben. Da alle Daten in der Aufgabenstellung genau beschrieben wurden, sind wir hier auf keine Probleme gestoßen.

- **Teammitglied:** Clemens
- **Bearbeitung:** 15 Minuten

5 Probleme

Bei der Erstellung der zweiten Szene haben wir einen Fehler in der `AxisAlignedBox`-Klassen-`hit(Ray r)`-Methode gefunden - wir haben immer von derselben Ecke der Box den Vektor zum Ursprung des Strahls berechnet, anstelle der Ecke der Ebene, die wir gerade testen (ob sie vom Rayursprung sichtbar ist). Wegen diesem Fehler hatten wir einige Schwierigkeiten mit dem Schatten. Außerdem haben wir angefangen, double-Rundungsfehler mithilfe von Epsilon-Rechnung auszugleichen.

6 Quellen und Literatur

- Computergraphik I Folien – Stephan Rehfeld