

Computergrafik 2 / Aufgabe 2: *Raytracer-Basis*

Implementieren Sie die Basis für einen einfachen Raytracer. Das Programm soll im ersten Schritt lediglich das Sichtbarkeitsproblem lösen und Bilder produzieren, die etwa folgendermaßen aussehen:



Abbildung: Der Raytracer soll mindestens Kugeln und Ebenen einfarbig ohne Beleuchtungsberechnung darstellen können.

Der Raytracer soll folgende Anforderungen erfüllen:

- Eine Szene wird als Menge von Szeneobjekten spezifiziert, die absolut im Weltkoordinatensystem positioniert werden.
- Jedes Objekt kann mit einem Material versehen werden, welches zunächst nur aus einer einfachen RGB-Farbe besteht (sozusagen ein „Selbstleuchten“ des Objekts). Später wird die Farbe durch einen Satz von Materialparametern für ein Beleuchtungsmodell ersetzt.
- Als erstes sollen die Objekttypen *Kugel* und *Ebene* implementiert werden. Es soll leicht möglich sein, die Software durch hinzufügen einer Klasse um einen neuen Objekttypen zu erweitern, ohne den eigentlichen Raytracing-Code zu ändern.
- Das System verwendet als Abstraktion eine virtuelle Kamera. Zunächst ist diese Kamera fix: der Betrachterstandpunkt befindet sich im Nullpunkt; die Blickrichtung ist entlang der negativen Z-Achse; die Y-Achse zeigt nach oben. Entwerfen Sie die Kamera so, dass man nur die Kamera-Klasse ändern müsste, um eine allgemeinere Kamera zu implementieren.
- Die Szene wird im Hauptprogramm durch entsprechende Konstruktoraufrufe spezifiziert, d.h. Sie instanzieren zunächst eine Szene und dann die gewünschten Objekte, die Sie der Szene durch entsprechende Methoden hinzufügen.
- Beim Aufruf des Programms wird ein Bild berechnet, auf der Festplatte gespeichert und dann angezeigt (siehe Warmup-Aufgabe).

- Für eine sehr gute Note implementieren Sie bitte auch den Objekttyp *Axis Aligned Box* oder weitere geometrische Primitive Ihrer Wahl.

Vorgehensweise

Bearbeiten Sie die Aufgabe in folgenden Schritten:

- Lesen Sie aufmerksam die Aufgabenstellung und die Anforderungen.
- Betrachten Sie den in der Vorlesung gezeigten Raytracing-Algorithmus und entwerfen Sie eine Menge von Klassen, die alle Bestandteile des Algorithmus abbilden. Wählen Sie vernünftige Namen wie z.B.: Scene, Shape, Sphere, Plane, Camera, Ray, Hit, usw. Spezifizieren Sie dann die notwendigen Operationen als Funktionen auf diesen Klassen.
- Ein „Hit“-Objekt soll alle Informationen speichern, die Sie benötigen, um für einen Trefferpunkt eine Farbe auszurechnen, z.B. die Position des Treffers auf dem Strahl oder die Farbe des getroffenen Objekts.
- Verwenden Sie als Grundlage für Ihre Implementierung wieder die Klasse `ImageGenerator`, die Sie bereits aus dem Warmup kennen.
- Verwenden Sie möglichst auch die Klassen `cg2.vecmath.vector` und `cg2.vecmath.color` aus dem zu Verfügung gestellten Material.
- Definieren Sie eine geeignete Testszene. Generieren Sie für jeden Pixel des Bildes einen Strahl und schneiden Sie ihn mit der Geometrie der Szene. Färben Sie den Pixel mit der Farbe der getroffenen Geometrie ein.

Gruppeneinteilung

Falls Sie Ihre Arbeitsgruppe noch nicht angemeldet haben (2 Personen oder einzeln), holen Sie dies bitte unverzüglich nach: teilen Sie mir Namen, Emailadresse und Matrikelnummern der Gruppenmitglieder per Email an `hschirmacher@beuth-hochschule.de` mit.

Abgabe

Die Abgabe erfolgt bis spätestens zum 26.10.11 für die Mittwochsübungen und bis zum 31.10.11 für die Montagsübungen. Bitte demonstrieren und erläutern Sie die Lösung spätestens am Abgabetag in der Übung und schicken Sie die Quellen am gleichen Tag spätestens um 22 Uhr an `hschirmacher@beuth-hochschule.de`. Erstellen Sie für die Abgabe ein Archiv des gesamten lauffähigen Projekts, in dem nur die Quellen und generierten Bilder, aber keine binären Kompilate enthalten sind.

Wenn Sie die o.a. Zeiten nicht einhalten, gilt die Übung als verspätet abgegeben; dies wird mit einem Abschlag auf die Note belegt. Bitte beachten Sie die Hinweise auf dem Handout der ersten Vorlesung.