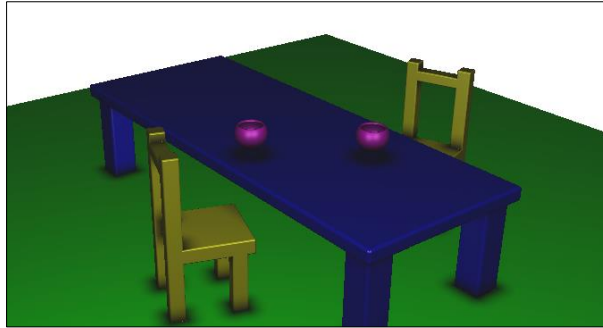


Computergrafik Hausaufgabe WS2024

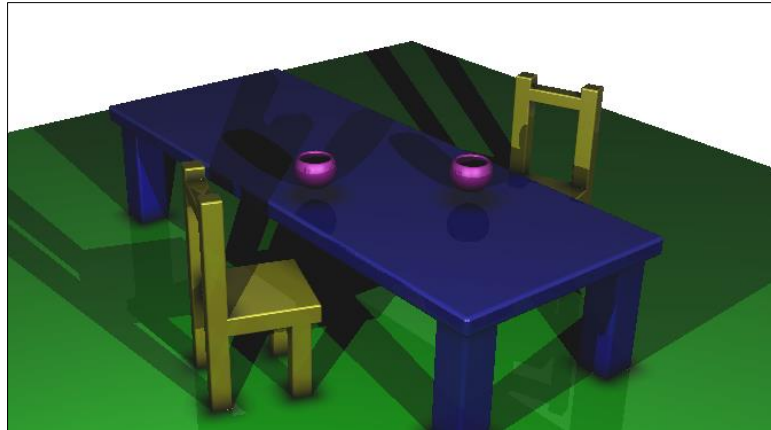
Rendering mit Raytracing und Signed Distance Functions

Gegeben ist ein WebGL/GLSL Programm, welches in der Übung besprochen wurde. Die Aufgaben sind mit Hilfe von SDF (Signed Distance Functions) und Raycasting zu lösen. Als Grundlage soll dazu der Code aus dem **Base.zip** Archiv genutzt werden.



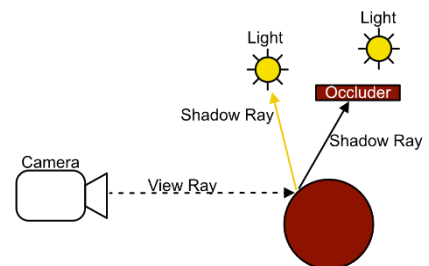
1. Modellierung

- a. Modellieren Sie eine einfache Szene mit einer flachen Box als Bodenplatte
- b. Modellieren Sie einen Tisch mit 2 Stühlen sowie eine Tasse mit Henkel auf dem Tisch. Nutzen Sie für die einzelnen Objekte CSG (Constructive Solid Geometry) Operationen.
- c. Animieren sie ein Objekt durch eine einfache zyklische Animation. Z. b. ein Stuhl, der umherrückt oder ein Objekt auf dem Tisch das sich bewegt.
- d. Die Szene soll mit dem Phong'schen Beleuchtungsmodell mit 2 Punktlichtquellen beleuchtet werden.
- e. Geben Sie den einzelnen Objekten verschiedene Materialparameter für den *Diffusen* und *Spekularen* Anteil des Beleuchtungsmodells.
Hinweis: Sie können *structs* aus C nutzen, um mehrere Werte aus einer Funktion zurückzugeben.
- f. Die Kamera soll statisch sein und so positioniert werden, dass Sie einen guten Überblick über die Szene gibt.

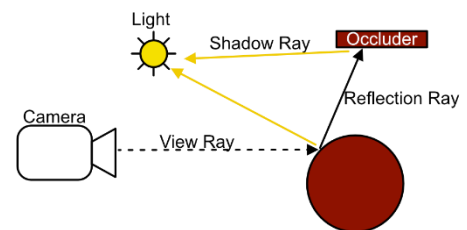


2. Reflexionen und Schatten

- a. Implementieren Sie Schattenberechnung, indem Sie von jedem Schnittpunkt mit der Szene einen Lichtstrahl zu den Lichtquellen berechnen. Falls der Lichtstrahl die Szene schneidet, fällt auf den Schnittpunkt kein Licht.



- b. Platzieren Sie ein weiteres Objekt auf den Tisch mit einer stark reflektierenden Oberfläche. Implementieren Sie das reflektive Material, indem Sie ähnliche wie bei der Schattenberechnung einen Strahl vom Schnittpunkt mit der Oberfläche reflektieren und bei Kollision mit der Szene das Material der Szene darstellen.



- c. Ermöglichen Sie eine Kamerabewegung mit der Maus (z.B. Orbit um die Szene). Nutzen Sie dazu die Uniform Variable `vec3 iMouse`, welche die Parameter der Mausposition in xy und ob die linke Maustaste gedrückt wird in z enthält.

3. Texturen und Rotationskörper

Dieser Aufgabenblock erfordert etwas mehr Eigenanteil, da Texturierung nicht in der Übung behandelt wurde und die Distanzfunktion für Rotationskörper selbst entwickelt werden muss.

- a. Texturieren Sie Ihre Szene entweder mit prozeduralen Texturen oder mit Bildern die als Texturen geladen werden.
- b. Entwickeln und implementieren Sie eine SDF-Funktion zur Berechnung der Distanz zu einem Rotationskörper. Modellieren Sie einen Rotationskörper als Vase und platzieren Sie diese auf den Tisch.

Hinweise:

1. Es wird eine OpenGL 3.0 kompatible Grafikkarte benötigt
2. Die Rechner im PC-Pool (3. Stock) können für diese Aufgabe verwendet werden
3. Für die Aufgabenblöcke 1 und 2 muss nur der Fragmentshader in der **index.html** verändert werden. Für die Verwendung von Texturen in Aufgabeblock 3 muss auch die **render.js** Datei angepasst werden.
4. Zur Entwicklung der Lösung kann jeder beliebige Texteditor verwendet werden, empfohlen wird Visual Studio Code
5. Zur Anzeige der Anwendung wird der Browser Google Chrome oder Firefox empfohlen, es sollten aber auch alle anderen gängigen Browser funktionieren
6. Die Aufgaben sind selbstständig zu lösen. Gruppenarbeiten werden als Plagiate gewertet.
7. Stellen Sie sicher, dass Ihre Richtungsvektoren normalisiert sind, sonst kommt es zu Artefaktbildung.

Abgabetermin der Lösung: 31. Januar 2025, 12 Uhr!

An: sven.kluge@uni-rostock.de, Betreff: *Computergrafik Hausaufgabe*

Abgabeform (wichtig!):

- Mail mit Zip-Datei im Anhang in der Form „**Name_Vorname.zip**“
- Es müssen alle benötigten Daten angefügt sein und die Webapplikation muss selbstständig lauffähig sein
- Nicht lauffähige Programme können nicht bewertet werden

Hilfreiche Links:

- <https://iquilezles.org/articles/distfunctions/>
- <https://iquilezles.org/articles/>
- <https://jamie-wong.com/2016/07/15/ray-marching-signed-distance-functions/>
- Shadertoy Beispiele