Prof. Dr.-Ing. Hartmut Schirmacher
Fachbereich Informatik und Medien

Real Time Rendering Aufgabe 1: Simple Lighting

Diese Aufgabe dient zur Einarbeitung in die Verwendung der OpenGL Shading Language und von Qt/OpenGL. Sie beschäftigen sich mit der Entwicklung von Shader-Programmen zur Darstellung von verschiedenen Materialeigenschaften.

Abgabe: siehe Moodle! Demonstrieren Sie in der Übung nach der Abgabe Ihre Lösung. Alle Gruppenmitglieder müssen dazu anwesend sein. Ausschlaggebend ist neben der eigentlichen Lösung vor allem die Qualität Ihrer Erklärungen.

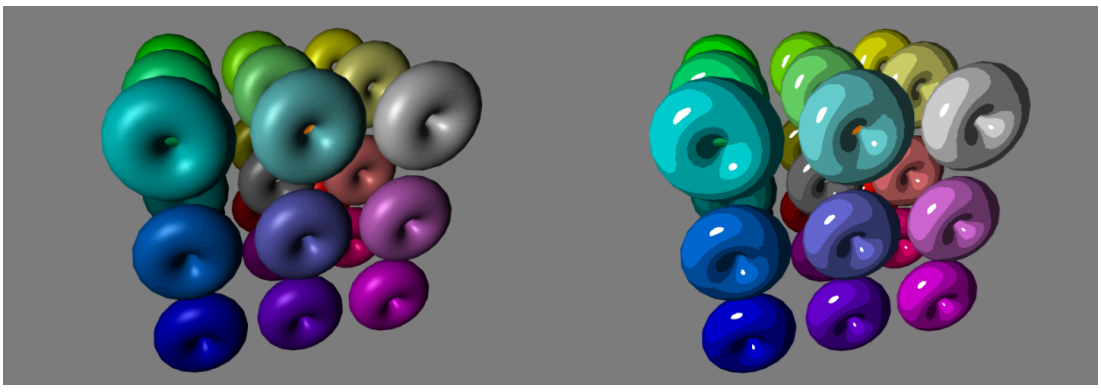
Bewertung: Verspätete Abgaben werden mit 2/3 Notenabzug pro angefangener Woche Verspätung belegt. Wer die geforderten Dinge funktionierend löst und gut erklärt, erhält mindestens eine 2.0. Eine 1.0 ist nur bei Demonstration von sehr gutem Verständnis der Materie erreichbar.

Aufgabe 1.1: Toon Shader

Recherchieren Sie die Technik des *Toon-Shading* (auch bekannt als *Cel-Shading*). Vorsicht: zu diesem Thema gibt es viel „Müll“ im Netz. Dokumentieren Sie im Sourcecode Ihre Quelle(n), und erklären Sie bei der Demo, was für die Qualität Ihrer Quelle spricht (siehe auch Qualitätskriterien weiter unten!)

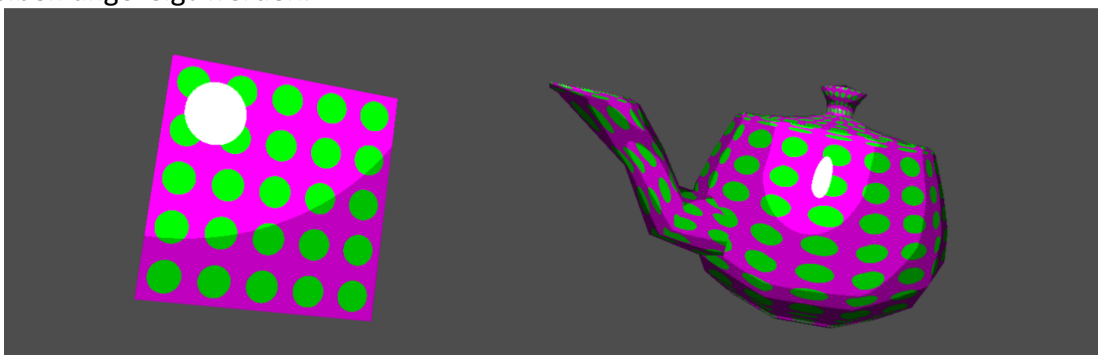
Kopieren und modifizieren Sie den Phong-Shader so, dass die berechnete Oberflächenintensität, wie im Bild rechts zu sehen, in mehreren Abstufungen diskretisiert wird (Verlauf wird durch einfarbige Stufen ersetzt). Achten Sie dabei auf folgende Qualitätskriterien:

- korrekte Wiedergabe des weißen Glanzlichts;
- gleichmäßige Verteilung der Farbstufen über die Einfallswinkel
  (Achtung genau überlegen was `ndotl` ist)
- die Bedeutung jeder Zahl, die im Code vorkommt, sollte man auch genau erklären können (keine „magic numbers“ – „naja mit 0.7236 hat es dann gut ausgesehen“)
- Man soll in der Applikation zwischen Phong und Toon hin- und herschalten können
- die Anzahl der Farbstufen soll interaktiv einstellbar sein
- da lange Branches schlecht für die Shaderperformanz sind, vermeiden Sie soweit wie möglich if-then-else im Shadercode.
-



Aufgabe 1.2: Dots

Fügen Sie einen weiteren Shader hinzu, zu welchem man umschalten kann[1]. Kopieren und modifizieren Sie Ihren Toon-Shader aus Aufgabe 1.1 so, dass die Oberfläche nicht einfarbig erscheint, sondern mit gleichmäßig verteilten andersfarbigen Punkten übersät ist. Farbe, Dichte und Radius der Punkte sollen über drei Uniform-Variablen von geeignetem Typ aus der Anwendung heraus über die Tastatur steuerbar sein. Es sollen keine unvollständigen Scheiben angezeigt werden.



Zur Implementierung benötigen Sie Zugriff auf die Texturkoordinaten für jedes Fragment. Dazu muss das entsprechende Attribut `in vec2 texcoord` im Vertex-Programm entgegengenommen werden (siehe auch Beispiele im SU).

Bitte beachten Sie: es werden lediglich Texturkoordinaten benötigt, nicht jedoch Texturen selbst. Die Verwendung eines Texture-Samplers und der dazugehörigen Look-Up Funktion ist zur Lösung der Aufgabe nicht erwünscht.

Beachten Sie weiterhin, dass einige OBJ-Modelle (Duck) eine recht unintuitive Verteilung der Texturkoordinaten verwenden. Wählen Sie geeignete Modelle zur Demonstration.

Aufgabe 1.3: Prozeduraler Veränderung der Oberfläche

Implementieren Sie einen weiteren prozeduralen Oberflächen-Shader ihrer Wahl. Lassen Sie sich dabei z.B. von Beispielen aus dem Web inspirieren. Beachten Sie, dass Sie die Funktionsweise Ihres Shaders bei der Demo erklären sollen.

Prozedurale Shader sind nicht notwendigerweise statisch, sondern lassen sich durch Übergabe eines Zeitparameters per Uniform-Variable auch animieren. Siehe das „Wobble“-Beispiel im SU (aber bitte nicht nur 1:1 kopieren).

Fügen Sie ihrer Szene weitere Modelle hinzu, die ihr prozedurales und/oder animiertes Material gut demonstrierbar machen.

[1] Leider gibt es unter manchen Windows-Versionen einen noch nicht gelösten Bug im RTR-Code oder in Qt, so dass die Methode `Mesh::replaceMaterial()` nicht funktioniert. Zum Umschalten zwischen zwei Materialien legen Sie am besten die Objekte jeweils doppelt an, für jedes Material einmal; oder Sie erzeugen die Szene nach dem Umschalten einfach neu.