

Übung zu Computergraphik I

– Übungsblatt 5 –

Lehrstuhl für Computergraphik
und Multimediasysteme

Hendrik Hochstetter, Bianca Kretz, Rene Winchenbach

Abgabe: Für Studenten mit 5 LP verpflichtend bis spätestens 15. Mai 2014, 10 Uhr

Besprechung: Mittwoch 21. Mai und Donnerstag 22. Mai 2014

Hinweise: Die Lösungen der Programmieraufgaben senden Sie bitte per Mail mit Name und Matrikelnummer an Ihren jeweiligen Tutor. Geben Sie dabei bitte nur die modifizierte Quelltextdatei als Anhang ab.

Aufgabe 1 Phong-Beleuchtung (2 Punkte)

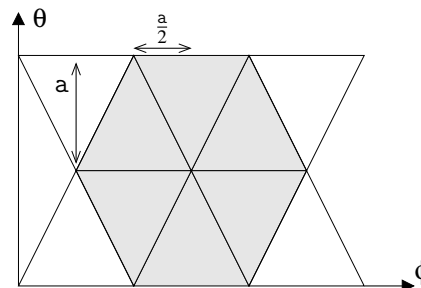
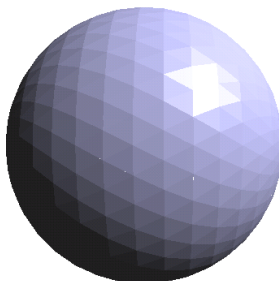
In der folgenden Aufgabe soll eine kleine OpenGL-Anwendung geschrieben werden, in der eine beleuchtete Sphäre dargestellt wird. Laden Sie hierzu zunächst das Programmgerüst `ueb05.zip` herunter:

<http://www.cg.informatik.uni-siegen.de/computergraphik-i-2014s/uebungen>

Das gegebene Programm enthält bereits eine Funktion `drawSphere()`, die Punkte auf der Oberfläche einer Sphäre berechnet und darstellt. Die Berechnung der Oberflächenpunkte erfolgt in Abhängigkeit zweier Winkel $\theta \in [0, \pi]$ und $\phi \in [0, 2\pi]$ nach der folgenden Formel, wobei r der Sphärenradius ist:

$$\mathbf{r}(\theta, \phi) = r \begin{pmatrix} \sin\theta \cos\phi \\ \sin\theta \sin\phi \\ \cos\theta \end{pmatrix}$$

Im Folgenden soll eine Triangulierung der Sphäre durchgeführt werden. Außerdem soll die Sphäre mit Hilfe des Phong-Modells beleuchtet werden. In der folgenden Abbildung ist die beleuchtete Sphäre sowie das Schema der Triangulierung zu sehen:



- 1.1 Ergänzen Sie die Funktion `drawSphere()`, so dass die Sphäre als Dreiecksnetz dargestellt wird (siehe Abbildung). Sie können hierzu die bereits definierten `for`-Schleifen verwenden.

Beachten Sie, dass der Winkel θ das Intervall $[0, 2\pi]$ durchläuft, damit in der Triangulierung in zwei Richtungen ausgerichtete Dreiecke berechnet werden. Definieren Sie die Eckpunkte der Dreiecke mit dem Befehl `glVertex3f(...)` und zu jedem Dreieck eine Oberflächennormale mittels `glNormal3f(...)`. Schalten Sie anschließend mit `glEnable(GL_LIGHTING)` die Lichtquelle ein, um die Beleuchtung von der Graphik-Hardware berechnen zu lassen.

- 1.2 Schalten Sie die Lichtquelle aus und berechnen Sie für jeden Eckpunkt einen Phong-Beleuchtungswert (entspr. der Formel aus der Vorlesung) durch Setzen des jeweiligen Farbwertes mit Hilfe der Funktion `glColor4f(...)`. Geeignete Werte für die Licht- und Materialeigenschaften sind am Anfang des Programms als globale Variablen vordefiniert.

Hinweis: Mit Hilfe der Taste `1` kann zwischen manueller und Hardware-unterstützter Beleuchtung gewechselt werden. Der aktuelle Zustand ist durch die Boole'sche Variable `g_compute_lighting` festgelegt.

- 1.3 Die Hardware-unterstützte Beleuchtung entspricht dem sogenannten Blinn-Phong-Modell. Im Unterschied zur Phong-Berechnung wird anstelle des reflektierten View-Vektors $\hat{\mathbf{r}}_v$ der Half-Way-Vektor

$$\hat{\mathbf{h}} = \frac{\hat{\mathbf{v}} + \hat{\mathbf{l}}}{\|\hat{\mathbf{v}} + \hat{\mathbf{l}}\|}$$

verwendet. Anstelle des Terms $(\hat{\mathbf{l}} \cdot \hat{\mathbf{r}}_v)$ in der Formel für den spekularen Lichtanteil wird das innere Produkt $(\hat{\mathbf{n}} \cdot \hat{\mathbf{h}})$ gebildet. Implementieren Sie das Blinn-Phong-Modell und vergleichen Sie das Ergebnis mit der Hardware-Beleuchtung.