Prof. G. Zachmann
J. Boeckers
N. M. Jadid
A. Mühlenbrock

Universität Bremen
Fachbereich 3
Arbeitsgruppe CGVR
10. Januar 2025

Wintersemester 2024/25

Übungen zu Computergrafik - Blatt 9

Abgabe am 19.01.2025, 23:59 Uhr

In diesem Aufgabenblatt beschäftigen wir uns im praktischen Teil mit der Beleuchtung von 3D-Modellen und im theoretischen Teil mit Farben. Laden Sie sich dazu das *LightingFramework* herunter, welches auf der CGVR-Seite zum Download bereitsteht.

Aufgabe 1 (Blinn-Phong-Modell (C++), *4+1 Punkte*)

a) Implementieren Sie das Blinn-Phong Beleuchtungsmodell in die Shader-Datei `lighting.frag`. Wir nehmen dabei vereinfacht lediglich eine Farbe I_j für das Licht j und eine Farbe k für das Material an:

$$I_{out} = k \odot I_{\text{amb}} + \sum_{j=0}^{n-1} k \odot I_j(\mathbf{n} \cdot \mathbf{l_j}) + k \odot I_j(\mathbf{n} \cdot \mathbf{h_j})^p$$

Hier gibt es zwei unterschiedliche Multipilkationsoperatoren:

- '$\mathbf{v} \odot \mathbf{u}$' ist die elementweise Multiplikation von den Farben $\mathbf{v}$ und $\mathbf{u}$ — in GLSL: `v*u`.
- '$\mathbf{v} \cdot \mathbf{u}$' ist das Skalarprodukt von den Vektoren $\mathbf{v}$ und $\mathbf{u}$ — in GLSL: `dot(v,u)`.

Zudem gibt es folgende Variablen:

- k: Die Farbe des Materials (siehe Variable `k`).
- I_{amb}: Die ambiente Farbe des globalen Lichtes (siehe uniform-Variable `ambientColor`).
- I_j: Die Farbe des Lichtes j (siehe uniform-Variable `lights[j].color`).
- $\mathbf{n}$: der Normalenvektor der Oberfläche (siehe in-Variable `normal_from_vs`).
- $\mathbf{l_j}$: Der Lichtvektor in Richtung Licht (nutze `lights[j].position` und `position_from_vs`).
- $\mathbf{h_j}$: Der Half-Vektor (auch durch `lights[j].position` und `position_from_vs` berechenbar).
- p: Die Uniformvariable `materialShininess`.
- n: Anzahl der Lichter (siehe uniform-Variable `lightNum`).

Achten Sie darauf, dass die Länge 1 haben.

Stelle bei der Implementierung sicher, dass die entsprechenden Vektoren normiert sind $(\mathbf{n}, \mathbf{l_j}, \mathbf{h_j})$ und das Fragment nur beleuchtet wird, wenn der Winkel zum Licht nicht größer als 90° ist.

b) Erweiteren Sie den Code aus Ausgabenteil (a), sodass das Licht in Abhängigkeit von der Distanz d abgeschwächt wird. Die hier zu implementierende benutzerdefinierte Abschwächungsfunktion sei wie folgt definiert:

$$i(d) = \begin{cases} (1 - \frac{d}{d_{\text{max}}})^2 & d < d_{\text{max}} \\ 0 & \text{sonst} \end{cases}$$

Bei der Distanz d soll es sich um die Distanz zwischen Lichtquelle und Fragment handeln. Für d_{max} soll das `range`-Attribut der Light-Struct des jeweiligen Lichtes benutzt werden (Zugriff über `lights[j].range`).

Aufgabe 2 (CIE-xy, *0,5+0,5+3+1 Punkte*)

Betrachten Sie das CIE-Chromatizitätsdiagramm in Abbildung 1 (weiß sei $(\frac{1}{3}, \frac{1}{3})$):

a) Welche Bedeutung hat die Verbindungslinie zwischen 770nm und 380nm in diesem Diagramm? Warum ist es eine Gerade? (der übrige Rand des Diagramms ist gekrümmt)

b) Wie verlaufen Isolinien gleicher Sättigung in dem Diagramm? Skizzieren Sie eine solche Isolinie.

c) Gegeben seien die drei Farben im sRGB-Format:

- **Orange**: $a = (1.0, 0.75, 0)$
- **Dunkelgrün**: $b = (0, 0.51, 0.14)$
- **Anthrazit**: $c = (0.16, 0.19, 0.2)$

Zeichnen Sie die Farben in das Chromatizitätsdiagramm ein, indem Sie vorher die xy-Werte berechnen (nutzen Sie die Umrechnung wie sie in der Vorlesung vorgestellt wurde). Geben Sie den Rechenweg an. Schätzen Sie außerdem die dominante Wellenlänge der Farben ab und zeichnen Sie diese ein.

d) Warum liegt die für die Farbe c berechnete Position im Diagramm *scheinbar* an der falschen Stelle? D.h. warum ist das Diagramm an dieser Stelle nicht so dunkel, wie es die Farbe ist?

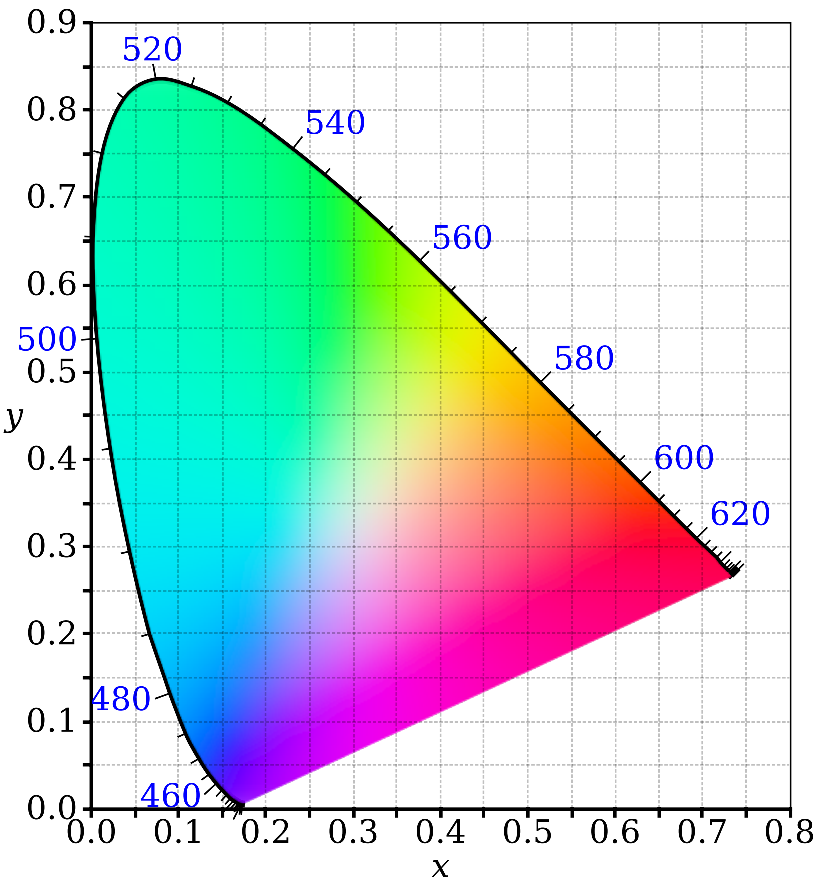

Abbildung 1: CIE-Chromatizitätsdiagramm

Erinnerung: Geben Sie die Rechnung immer mit an!