DHBW Stuttgart
Duale Hochschule Baden-Württemberg

Computergrafik **Fachrichtung Informatik**

Dr. rer. nat. Johannes Riesterer **TINF16IN**

Formalien

Die Klausur dauert 90 Minuten. Als Hilfsmittel sind Stift und Papier zugelassen.

Vermerken Sie bitte auf jedem Ihrer Blätter Ihre Matrikelnummer.

Aufgabe 1. **(4 Punkte)**

a) Gegeben sei das Kamera-Koordinatensystem (A, B) mit

$$A := \begin{pmatrix} -1 \\ -1 \\ 0 \end{pmatrix}, \ B := \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} \right\}.$$

Der Augenabstand zwischen Augenpunkt und Bildebene betrage $d = 1$. Berechnen Sie die Projektion des Punktes $Q := \begin{pmatrix} 1 \\ -3 \\ 2 \end{pmatrix}$ auf die Bildebene in Bildkoordinaten.

Lösung.

Wir drücken Q im Kamera-Koordinatensystem aus. Da B offensichtlich eine ONB ist, ist

$$M_B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}^t = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}.$$

Damit erhalten wir

$$\theta_{(A,B)}(Q) = M_B(Q - A) = M_B \cdot \left(\begin{pmatrix} 1 \\ -3 \\ 2 \end{pmatrix} - \begin{pmatrix} -1 \\ -1 \\ 0 \end{pmatrix} \right) = M_B \cdot \begin{pmatrix} 2 \\ -2 \\ 2 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ -2 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix}.$$

Die Projektion ist damit in homogenen Koordinaten gegeben durch

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 2 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}.$$

Die zugehörigen reellen Koordinaten sind

$$\begin{pmatrix} \frac{2}{2} \\ \frac{2}{2} \\ \frac{2}{2} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

und damit sind die Bildkoordinaten

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Aufgabe 2. **(4 Punkte)**

a) Erklären Sie die Definition der Strahldichte $L(x, \omega)$ für einen Punkt x in Richtung ω und gehen Sie dabei auf das Lambertsche Cosinus-Gesetz ein.

b) Erläutern Sie die Rendergleichung und erklären Sie die Funktionsweise eines Raytracing Algorithmus.

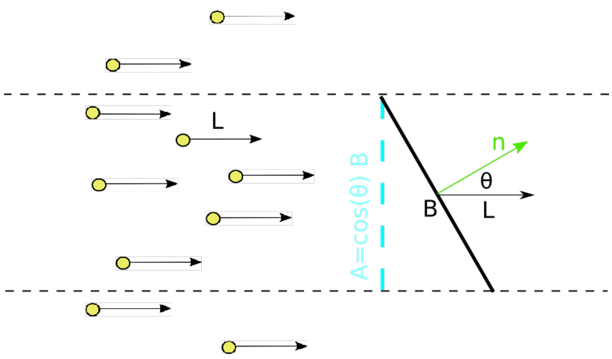
Lösung

a)

Die Strahldichte ist über den folgenden Grenzprozess definiert. Betrachtet man einen Teilchenstorm von Photonen mit Dichtefunktion η und einheitlicher Flugrichtung L, so ist die Anzahl der Photonen, welche nach einer Zeit t eine Fläche B mit Normale n passieren gegeben durch

$$D_B([0,t]) := \eta * ||L|| * \text{Flächeninhalt}(B) * t * \cos(\theta)$$

wobei θ der Winkel zwischen n und L ist.



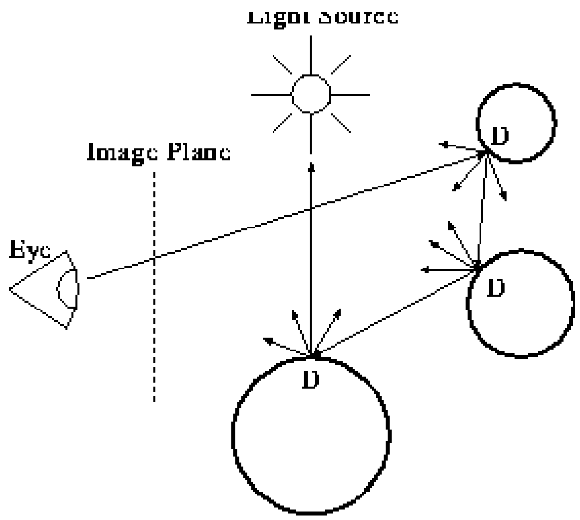
Bezeichnen wir mit $L(B) = \frac{d}{dt \cdot \cos(\theta)} D_B([0,t]) = \eta * ||L|| * \text{Flächeninhalt}(B)$, so erhalten wir die Strahldichte als Grenzwert

$$L(x, n) := \lim_{B->x} L(B)$$

bei dem die Fläche B zu einem Punkt x zusammengezogen wird. Die Strahlungsleistung aus einer Richtung θ am Punkt x berechnet sich demnach durch $I(x, \theta) = L(x, \theta) \cdot \cos(\theta)$, was auch als Lambertsches Cosinusgesetz bezeichnet wird.

b)

Beim Raytracing werden von einem Augenpunkt A Strahlen durch die Pixel P_i der Bildebene verfolgt, mit dem Ziel die spezifische Lichtenergie $L(A, \frac{\overline{AP_i}}{||\overline{AP_i}||})$ für diese Richtung zu berechnen.



Die Berechnung erfolgt rekursiv. An einem Schnittpunkt eines Strahls mit einem Objekt wird die Rendergleichung gelöst. Dafür werden ausgehend von dem Schnittpunkt weitere Strahlen in der Szene verfolgt. Dieser Vorgang wird wiederholt, bis eine vorgegebene Rekursionstiefe erreicht wird.

Die Rendergleichung lautet

$$L_o(x, \omega_o) = L_e(x, \omega_o) + \int_{H^2} f_r(x, \omega, \omega_0) \cdot L_i(x, \omega) \cdot \cos(\theta) d\omega \ , \tag{1}$$

und beschreibt die ausgehende Strahlungsleistung am Punkt x in Richtung ω. $L_e(x, \omega_o)$ ist die von der Oberfläche emittierte Strahlungsleistung. Das Integral

$$\int_{H^2} f_r(x, \omega, \omega_0) \cdot L_i(x, \omega) \cdot \cos(\theta) d\omega$$

beschreibt die an dem Punkt x reflektierte Strahlungsleistung. Dabei wird eine eingehende Strahlungsleistung $L_i(x, \omega)$ mit dem Reflektionsfaktor der BRDF-Funktion f_r multipliziert und über alle eingehenden Richtungen unter Berücksichtigung des Lambertschen Kosinusgetzes aufsummiert.

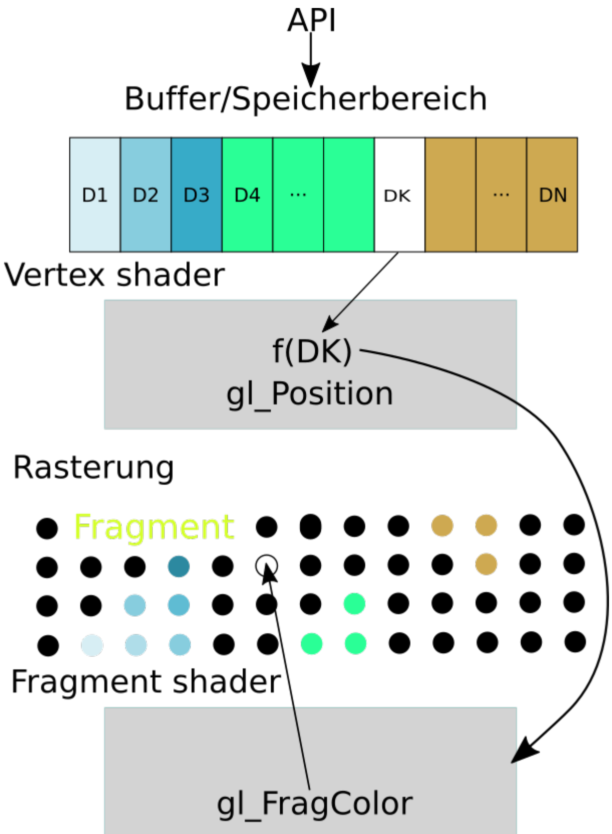
Aufgabe 3. **(4 Punkte)**

a) Erläutern Sie schematisch die Funktionsweise und den Ablauf eines GLSL Shaderprogrammes.

b) Erklären Sie die Funktionsweise des Bresenham Algorithmus.

Lösung

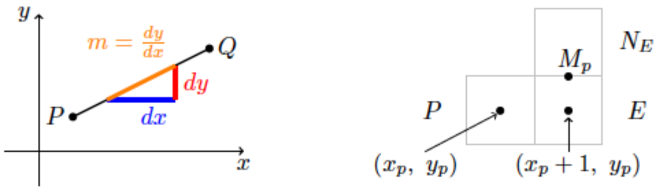
a)



Eine Applikation beschreibt den Speicherbereich des Vertexshaders mit Werten. Jeweils drei hintereinander folgende Werte definieren hierbei die Werte an den Eckpunkten eines Dreiecks. Der Vertexshader wird parallel auf diesen Werten aufgerufen. Er schreibt für jeden dieser Werte einen vierdimensionalen Vektor in Clipping-Koordinaten in den Speicherbereich des Fragmentshaders so wie etwaige weitere varying Variablen. Nun werden die Koordinaten projiziert und gerastert. Die Werte der Varyings werden interpoliert. Der Fragmentshader wird auf allen sichtbaren Pixeln aufgerufen und hat dabei Zugriff auf die Interpolierten Werte der varyings. Für jedes Pixel gibt er einen vierdimensionalen Vektor aus, der einen Farbwert im RGBA Format kodiert.

b)

Der Bresenham Algorithmus verläuft rekursiv. Angenommen man hat eine Linie $\overline{PQ}$ mit Steigung $0 < m < 1$ bereits bis zum Pixel P gerastert. Dann wird das folgende Entscheidungsproblem gelöst.



(a) Steigungsdreieck **(b)** Pixelraster

$$M_p \begin{cases} \text{oberhalb } \overline{PQ} & \Rightarrow \quad \text{wähle } E \\ \text{unterhalb } \overline{PQ} & \Rightarrow \quad \text{wähle } N_E \end{cases}$$

Diese Problem lässt sich mit Hilfe der Geradengleichung in das arithmetisches Problem

$$(x,\ y) \in y = mx + b \quad \Leftrightarrow \quad F(x,\ y) = 0$$

bzw.

$$F(x,\ y) \quad \begin{cases} = 0 & (x,\ y) \text{ liegt auf } S \\ > 0 & (x,\ y) \text{ liegt unterhalb von } S \\ < 0 & (x,\ y) \text{ liegt oberhalb von } S \end{cases}$$

überführen. Durch wiederholtes Einsetzen der Bedingung ist leicht zu sehen, dass dieses Problem inkrementell mit den Inkrementen

$$\triangle_E := 2 \cdot dy$$
$$\triangle_{NE} := 2dy - 2dx$$

gelöst werden kann.

Aufgabe 4. **(4 Punkte)**

a) Gegeben ist ein Oberflächenpunkt $(1,0,0)^t$ mit Normale $(0,1,0)^t$, eine Lichtquelle im Punkt $(0,2,0)^t$ und der Augenpunkt in $(2,1,0)^t$. Berechnen Sie die Helligkeit des Oberflächenpunktes nach dem Phongschen Beleuchtungsmodell mit dem Hardness-Faktor $h = 2$ und dem Specularity-Faktor $s = \frac{1}{4}$. Setzen Sie den Ambiente Factor $I_a = 0$ sowie Material und Licht-Faktoren auf $I_l = I_m = 1$.

Hinweis:
Die Reflektion r eines normierten Vektors l an einer Ebene mit Normalenvektor n ist gegeben durch $r = -l + 2 < n, l > n$.

b) Erklären Sie die Unterschiede zwischen Gouraud-Shading und Phong-Shading.

Lösung

a)

Nach dem Phongschen Beleuchtungsmodell ist die Helligkeit gegeben durch $I = I_d + I_s + I_a$. Der diffuse Anteil berechnet sich durch $I_d = max(0, \cos(\theta))$, wobei θ der Winkel zwischen n und $\overline{OL}$ ist. Wir berechnen

$$\cos(\theta) = \langle \frac{\overline{OL}}{||\overline{OL}||}, \frac{n}{||n||} \rangle = \langle \frac{1}{\sqrt{5}} \begin{pmatrix} -1 \\ 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \rangle = \frac{2}{\sqrt{5}}$$

und damit $I_d = \frac{2}{\sqrt{5}}$.

Der reflektierende Anteil berechnet sich durch $I_s = s \cdot max(0, \cos(\alpha))^h$, wobei α der Winkel zwischen dem reflektierten Strahl r und $\overline{OA}$ ist. Mit $r = \frac{1}{\sqrt{5}} \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}$ und $\overline{OA} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$ berechnen wir

$$\cos(\alpha) = \langle \frac{\overline{OA}}{||\overline{OA}||}, r \rangle = \langle \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \frac{1}{\sqrt{5}} \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix} \rangle = \frac{3}{\sqrt{2}\sqrt{5}}$$

und damit $I_s = \frac{1}{4}(\frac{3}{\sqrt{2}\sqrt{5}})^2 = \frac{9}{40}$. Da $I_a = 0$ erhalten wir $I = I_d + I_a = \frac{2}{\sqrt{5}} + \frac{9}{40}$.

b)

Sei $L(n)$ ein Beleuchtungsmodell (z.B. Phong) angewendet auf einen Normalenvektor n und $I(x_1, x_2, x_3)$ eine Interpolationsfunktio, welche zwischen den Werten x_1, x_2, x_3 interpoliert. Beim Gouraud-Shading werden im Vertex Shader die Intensitäten $L(n_1), L(n_2), L(n_3)$ für die Eckpunkte eines Dreiecks berechnet und dann im Fragmentshader zwischen diesen Werten interpoliert, also $I(L(n_1), L(n_2), L(n_3))$ berechnet. Im Gegensatz dazu wird beim Phong Shading im Fragment-Shader zwischen den Normalen interpoliert und darauf das Beleuchtungsmodell angewendet, es wird also $L(I(n_1, n_2, n_3))$ berechnet. Gouraud-Shading ist damit deutlich aufwendiger aber physikalisch korrekter.

Aufgabe 5. **(4 Punkte)**

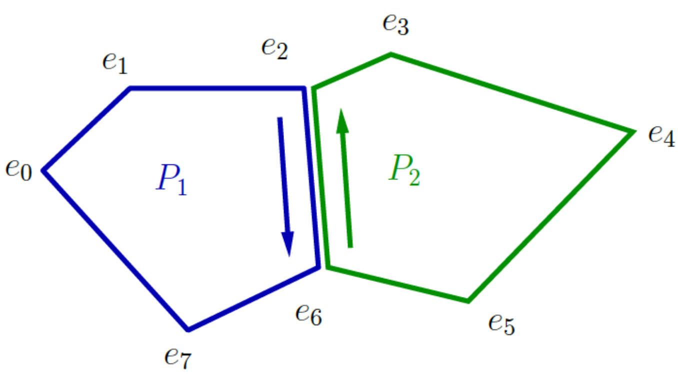
a) Wie viele Orientierungen kann ein Netz haben? Begründen Sie Ihre Antwort.

b) Gegeben sind die Punkte $b_0 := \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix}$, $b_1 := \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$, $b_2 := \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$ so wie die Bezierkurve $B(t) := \sum_{i=0}^{2} B_i^2(t) \cdot b_i$. Berechnen Sie mit Hilfe des Algorithmus von de Casteljau $B(\frac{1}{4})$.

Lösung

a)

Ein Netz ist orientierbar, wenn alle zugehörigen Polygone gleich orientiert werden können und dabei alle benachbarten Kanten gegenläufig sind.



Ist diese Bedingung mit der Wahl einer Orientierung der Polygone erfüllt, so ist diese ebenso mit der Wahl der entgegengesetzten Orientierung erfüllt. Ist diese Bedingung nicht erfüllt, gibt es keine Möglichkeit das Netz zu orientieren. Ein Netz hat also entweder genau zwei oder keine Orientierung.

b) Wir berechnen die itterierten Streckenteilungen:

$$b_1^1 = \frac{3}{4} b_0 + \frac{1}{4} b_1 = \frac{3}{4} \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix}$$

$$b_2^1 = \frac{3}{4} b_1 + \frac{1}{4} b_2 = \frac{1}{4} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

$$b_2^2 = \frac{3}{4} b_1^1 + \frac{1}{4} b_2^1 = \frac{9}{16} \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} + \frac{1}{16} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -\frac{1}{2} \\ \frac{10}{16} \\ 0 \end{pmatrix}$$

und damit $B(\frac{1}{4}) = \begin{pmatrix} -\frac{1}{2} \\ \frac{10}{16} \\ 0 \end{pmatrix}$