Klausur

WP Computergrafik Sommersemester 2016	29.06.2016	Prof. Dr. Philipp Jenke Seite 1 von 13

Name:	
Matrikelnummer:	

Aufgabe	**Maximale Punktzahl**	**Erreichte Punktzahl**
Verschiedenes	18	
Polygonale Netze	7	
Licht und Texturen	8	
Datenstrukturen	10	
Kurven und Flächen	9	
Gesamt	**52**	

Hilfsmittel:

- Erlaubtes Material: 1 Blatt handschriftliche Notizen (mit Vor- und Rückseite)
- Nicht erlaubt: Elektronische Geräte in irgendeiner Form, also kein Taschenrechner, Notebook, Handy, usw.
- Dauer: 90 Minuten

Klausur

WP Computergrafik Sommersemester 2016	29.06.2016	Prof. Dr. Philipp Jenke Seite 2 von 13

1 Verschiedenes

1.1 Octree

Wie viele Blattknoten hat ein voll besetzter Octree der Tiefe 2 (Tiefe 0 = Würfel um gesamte Szene)?

1.2 Schatten

Skizzieren Sie den durch die Lichtquelle L und das abschattende Objekt O auf die beschattete Ebene E projizierten Schatten direkt in der Abbildung 1.

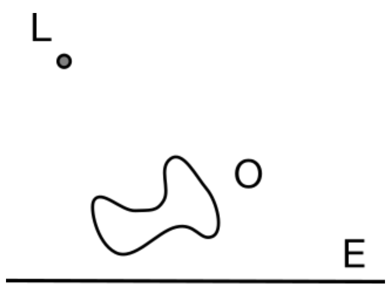

Abbildung 1: Das Objekt O wirft relativ zur Lichtquelle L einen Schatten auf die beschattete Ebene E.

1.3 Schattenvolumen

Zeichnen Sie in folgende Szene (Abbildung 2) die Schattenpolygone ein und schreiben Sie in die Zwischenräume jeweils den Zählerstand für die Schattenvolumen.

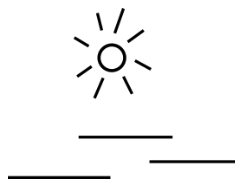

Abbildung 2: Schattenvolumen.

1.4 Normalenvektor

Geben Sie einen Vector $\vec{v}$ an, der im 2D senkrecht auf dem Vektor $\begin{pmatrix} 1 \\ 2 \end{pmatrix}$ steht.

1.5 Implizite Function

Gegeben sind die implizite Funktion $f(\begin{pmatrix} x \\ y \end{pmatrix}) = x^2 + (y-2)^2$ und der Isowert $\lambda = 1$. Liegt der Punkt $\begin{pmatrix} 0.5 \\ 2 \end{pmatrix}$ innen, auf dem Rand oder außen?

1.6 Gradient

Wie lautet der Gradient $\nabla f(\begin{pmatrix} x \\ y \end{pmatrix})$ für die Funktion aus der vorherigen Frage?

1.7 Vereinfachung

Welche Kante $e_0 \ldots e_4$ in Abbildung 3 wird bei der Vereinfachung nach Garland/Heckbert als erste zur Kollabierung ausgewählt? Falls mehrere Kanten ähnlich gut geeinet sind, geben Sie diese alle an.

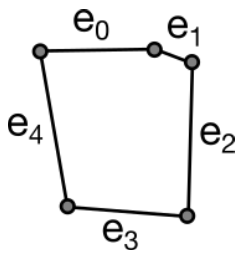

Abbildung 3: Auswahl einer Kante bei der Vereinfachung von Netzen.

1.8 Baryzentrische Koordinaten

Geben Sie eine Größensortierung der Baryzentrischen Koordinaten des Punktes p relativ zu den Eckpunkten a, b, c aus Abbildung 4 an (z.B.: $\alpha < \beta < \gamma$).

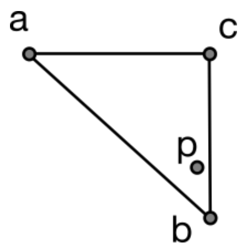

Abbildung 4: Baryzentrische Koordinaten des Punktes p bezüglich a, b, c.

1.9 Schnitt Strahl-Ebene

Berechnen Sie den Schnittpunkt x des Strahls

$$S : \begin{pmatrix} 1 \\ 5 \end{pmatrix} + \lambda \begin{pmatrix} \frac{1}{3} \\ -1 \end{pmatrix}$$

mit der Ebenen

$$E : \begin{pmatrix} 0 \\ 1 \end{pmatrix} \cdot x - 2 = 0.$$

2 Polygonale Netze

2.1 Anliegende Facetten finden

In dieser Teilaufgabe entwickeln Sie einen Algorithmus zum Finden aller Facetten f die inzident oder adjazent zu einer gegebenen Halbkante h sind.

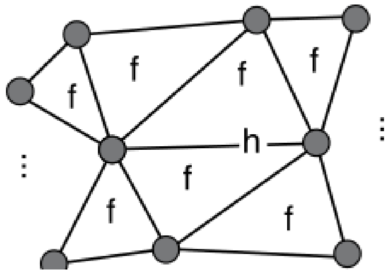

Abbildung 5: An Halbkante h anliegende Facetten f.

Schreiben Sie eine Hilfsmethode `List<Facette> getInzidenteFacetten(HalbkantenVertex v)` (Pseudocode), die alle inzidenten Facetten zu einem Vertex v liefert.

```
List<Facette> getInzidenteFacetten(HalbkantenVertex v){

}
```

Verwenden Sie die in der vorherigen Teilaufgabe geschriebene Hilfsmethode, um eine Methode zum Finden aller an einer Halbkante anliegenden inzidenten und adjazenten Facetten zu schreiben (Pseudocode):

```
List<Facette> getFacetten(Halbkante h){

}
```

2.2 Corner-Cutting-Schema

In der Vorlesung haben Sie das Corner-Cutting-Schema kennengelernt. Im Schritt zur Berechnung des Durchschnitts haben Sie für jeden Vertex v_i^{neu} die gemittelte Position mit

$$v_i^{neu} = \frac{1}{2}v_{i-1} + \frac{1}{2}v_i$$

berechnet.
Betrachten wir nun eine neue Berechnungsvorschrift mit

$$v_i^{neu} = \frac{1}{4}v_{i-1} + \frac{1}{2}v_i + \frac{1}{4}v_{i+1}.$$

Zeichnen Sie die neuen Vertices nach dem Split-Schnitt direkt in Abbildung 6 ein und berechnen Sie die neue Position für den Vertex a nach der Durchschnittsbildung.

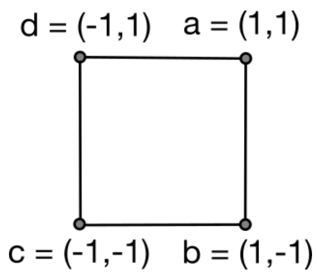

Abbildung 6: Führen Sie für dieses Ausgangspolygon einen Split-Schritt im Corner-Cutting-Verfahren durch.

$a^{neu} =$

3 Licht und Texturen

3.1 Strahl-Segment-Schnitt

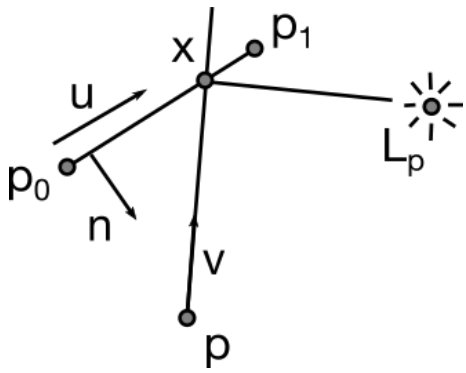

Abbildung 7: Der Strahl $S : p + \lambda v$ trifft auf das Segment (p_0, p_1). Die Szene wird von einer Lichtquelle an der Position L_p beleuchtet.

Der Strahl $S : p + \lambda v$ schneidet das Segment ($p_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, p_1 = \begin{pmatrix} 7 \\ 4 \end{pmatrix}$) im Punkt $x = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$. Zeigen Sie, dass der Schnittpunkt innerhalb des Intervalls liegt.
Tip: Interpolieren Sie x zwischen p_0 und p_1 (lineare Interpolation). Wenn das Interpolationsgewicht α in [0,1] liegt, dann liegt x zwischen p_0 und p_1.

3.2 Texturkoordinaten

Bestimmen Sie die Texturkoordinate u von x bezüglich p_0 und p_1. p_0 hat die Texturkoordinate $u = 0$ und $p1$ hat die Texturkoordinate $u = 1$.

3.3 Phong: Diffuse Leuchtdichte

Berechnen Sie die diffuse Leuchtdichte an der Stelle x für die Lichtquelle an der Position $L_p = \begin{pmatrix} 5 \\ 1 \end{pmatrix}$ mit der Farbe $c = (1, 0, 0)$ und dem diffusen Reflexionsfaktor 1. Die Normale des Segments ist $n = \begin{pmatrix} 1 \\ -2 \end{pmatrix}$.

$L_{diff} =$

4 Datenstrukturen

Gegeben ist eine Datenstruktur für einen Quadtree im 2D:

```
class Zelle<T> {
  // Liste der Elemente
  List<T> elemente = {};
  // bei Blattknoten null, sonst 4
  // Zellen
  Zelle [] kindZellen;
  // Liefert wahr, wenn element
  // in der Zelle liegt
  boolean umschliesst(T element){...}
}
```

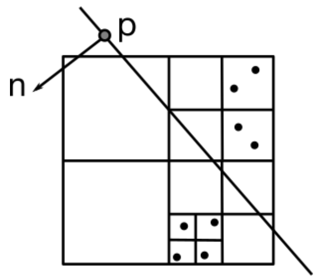

Abbildung 8: Datenstruktur für Zelle (links) und schematische Darstellung einer Szene aus Punkten (rechts).

4.1 Aufbau des Baumes

Vervollständigen Sie die rekursive Methode der Klasse `Zelle` zum Einfügen eines Elements in den Quadtree. Der Algorithmus kennt die maximale Anzahl pro Zelle `MAX_PER_ZELLE` (falls mehr Elemente in einer Zelle sind, wird die Zelle unterteilt), und die maximale Tiefe des Quadtrees `MAX_TIEFE`. Der Aufruf erfolgt mit `wurzel.einfuegen(element, 0)`. Nehmen Sie an, es gibt eine Methode `split()`, die für einen Blattknoten die 4 Kindknoten erzeugt.

```
void einfuegen(T element, int tiefe){
```

```
}
```

4.2 Sichtbarkeit

Geben Sie einen Algorithmus in Pseudocode an, der für eine Ebene $E = (n, p)$ mit Normale n und Ebenenpunkt p und einen Quadtree-Wurzelknoten `wurzel` alle sichtbaren Elemente bestimmt und als Liste zurückgibt. Als Elemente nehmen wir Punkte aus dem $\mathbf{R}^2$ an. Sichtbar sollen alle Elemente sein, die in Normalenrichtung der Ebene liegen. Der Aufruf erfolgt mit `getSichtbare(n, p, wurzel)`.

```
Liste<Vector> getSichtbare(Vector n, Vector p, Zelle wurzel){
```

```
}
```

4.3 Umschliessender Hüllkörper

Die Wurzelzelle eines Quadtrees muss alle Objekte der Szene beinhalten (siehe Abbildung 9). Schreiben Sie eine Methode, die die Koordinaten (x,y) der linken unteren Ecke und der rechten oberen Ecke des Hüllkörpers für eine Liste von Punkten im $\mathbf{R}^2$ berechnet. Hinweis: Der berechnete Hüllkörper muss nicht quadratisch sein.

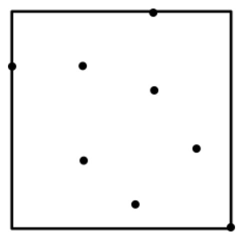

Abbildung 9: Umschliessender Hüllkörper für eine Menge von Punkten.

5 Kurven und Flächen

5.1 Konstruktion eines Kreises

Konstruieren Sie zwei Bezier-Kurven (je vom Grad 3), die aneinandergefügt zusammen in etwa die Form eines Kreises bilden. Zeichnen Sie dazu direkt in Abbildung 10 die notwendigen Kontrollpunkte ein und skizzieren Sie die sich ergebende Kurvenform.

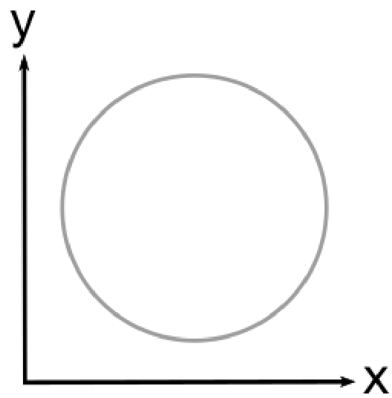

Abbildung 10: Zeichnen Sie die notwendigen Kontrollpunkte direkt in die Abbildung ein. Der hellgraue Kreis kann, muss aber nicht, als Vorlage dienen.

5.2 Kurvenkonstruktion

Skizzieren Sie den Verlauf der Kurve

$$p(t) = \sum_{i=0}^{2} c_i B_i(t)$$

für die folgenden Basisfunktionen und die bereits eingezeichneten Kontrollpunkte. Skizzieren Sie außerdem den Kurvenpunkt $x = p(0.25)$.

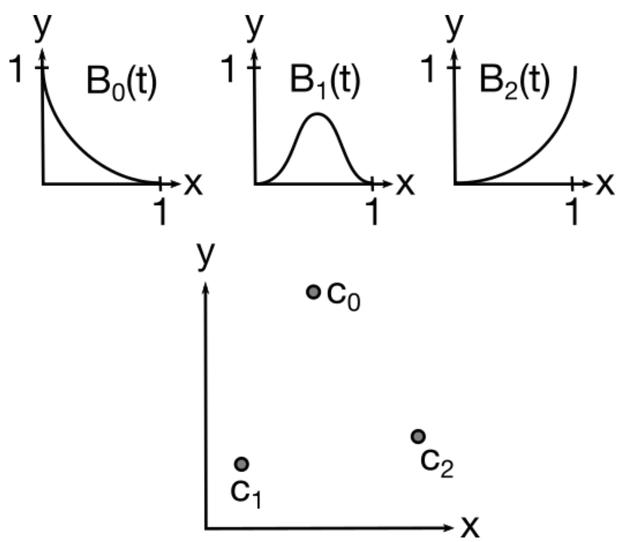

Abbildung 11: Basisfunktionen $B_0 ... B_2$ und die vorgegebenen Kontrollpunkte.

5.3 Flächen-Tesselierung

Gegeben ist eine Tensorproduktfläche $S(u, v)$, wie in Abbildung 12 abgebildet. Geben Sie einen Algorithmus in Pseudocode an, um die Fläche mit Auflösung $m \times n$ mit Quads zu tesselieren (wie angedeutet).

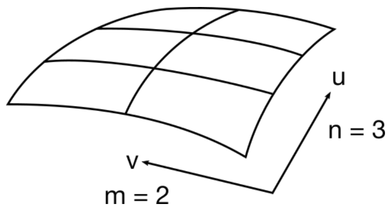

Abbildung 12: Tesselierung einerTensorproduktfläche.