



Computergrafik (SoSe 2023)

Blatt 10

fällig Sonntag 2. Juli, 24:00

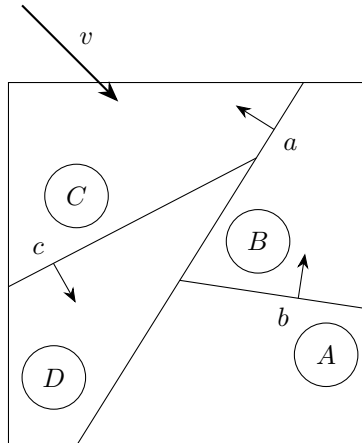
Verspätet eingereichte Abgaben können nicht berücksichtigt werden.

Aufgabe 1 (Theorie: Splines, Raytracing, Radiosity)

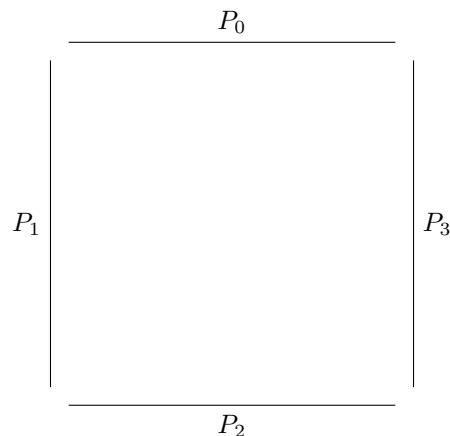
- (a) (2P) Eine kubische Bézierkurve ist gegeben durch die Kontrollpunkte a_0 , a_1 , a_2 und a_3 , wie unten abgebildet. An das Ende dieser Kurve soll eine quartische Bézierkurve C^2 -stetig angeschlossen werden. Deren Kontrollpunkte b_3 und b_4 sind bereits festgelegt. Bestimmen Sie b_0 , b_1 , und b_2 . Dies können Sie approximativ grafisch (nicht rechnerisch) erledigen.



- (b) (1P) Eine lineare Bézierkurve A ist gegeben durch die Kontrollpunkte $a_0 = (3, 7)$ und $a_1 = (4, 4)$. An ihr Ende soll eine lineare Bézierkurve B angeschlossen werden, und an das Ende von B eine weitere lineare Bézierkurve C . Geben Sie Kontrollpunkte für B und C an (rechnerisch, in Koordinaten), so dass beide Übergänge C^0 -stetig sind. Wenn für einen Kontrollpunkt mehrere Wahlmöglichkeiten bestehen, wählen Sie beliebig. Geben Sie an, welche Kontrollpunkte eindeutig bestimmt waren und welche Sie wählen konnten. Wenn Sie der Meinung sind, dass diese Aufgabe nicht lösbar ist, begründen Sie dies.
- (c) (1P) Verfahren Sie genau so wie in der vorigen Teilaufgabe, mit dem Unterschied, dass nun beide Übergänge C^1 -stetig sein sollen.
- (d) (1P) Verfahren Sie genau so wie in der vorigen Teilaufgabe, mit dem Unterschied, dass nun beide Übergänge C^2 -stetig sein sollen.



- (e) (2P) Für den in der abgebildeten 2D-Szene oben links angegebenen Sichtstrahl v soll (wie dies im Kontext von Raytracing von zentraler Bedeutung ist) der erste Schnittpunkt mit einem Szenenobjekt, also hier einem Kreis, bestimmt werden. Nutzen Sie dabei zur Beschleunigung den BSP-Traversierungsalgorithmus aus der Vorlesung. Der BSP-Baum ist bereits in Form der Trennebenen a , b und c mit ihren Normalen in die Szene eingezeichnet. Listen Sie alle Methodenaufrufe auf. Sie brauchen keine konkreten numerischen Rechnungen anzustellen (alle nötigen Entscheidungen lassen sich durch gutes Augenmaß treffen) und als Ergebnis auch keinen konkreten Schnittpunkt anzugeben, sondern lediglich auf welchem Kreis dieser Schnittpunkt liegt.
- (f) (1P) Nehmen Sie an, der Strahl hätte die Richtung $\left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right)^T$ und die Normale des Kreises am Schnittpunkt betrage $\left(-\frac{1}{\sqrt{5}}, \frac{2}{\sqrt{5}}\right)^T$. Berechnen Sie den reflektierten Vektor (welcher beim Raytracing zur Einbeziehung von Spiegelungen benötigt wird).
- (g) (2P) Gegeben ist die unten abgebildete Anordnung vier Flächen P_i . Diese Anordnung ist rotationssymmetrisch. Für die Emission der drei Flächen gilt $E_0 = 1$, $E_1 = E_2 = E_3 = 0$ und für ihre Reflektivität gilt $\rho_0 = \rho_3 = 0$, $\rho_1 = 0.8$, $\rho_2 = 0.5$. Hinsichtlich der Formfaktoren ist gegeben, dass $F_{0,1} = 0.2$ und $F_{0,2} = 0.3$. Stellen Sie das zugehörige Radiosity-Gleichungssystem in Matrixschreibweise (mit konkreten Werten) auf.





Aufgabe 2 (Praxis: Raytracing)

- (a) (3P) Ergänzen Sie die Funktion `update` derart, dass für das Pixel (i, j) der Bildebene ein Strahl von der Kameraposition im Ursprung durch dieses Pixel erzeugt wird. Die Öffnungswinkel sowohl in x - als auch in y -Richtung sollen 90° betragen. In einer Bildebene, die $\frac{\text{size}}{2}$ von der Kameraposition entfernt liegt, sind die Raumkoordinaten des Pixels dann $(j - \frac{\text{size}}{2}, -(i - \frac{\text{size}}{2}), -\frac{\text{size}}{2})$.

Rufen Sie dann mit diesem (von der Kameraposition startenden) Strahl die Methode `trace` auf. Die zurückgegebene Farbe soll mittels der Methode `setPixel` auf das entsprechende Pixel des Canvas gezeichnet werden.

Hinweis: Stellen Sie die Strahlrichtung durch einen Vektor der Länge 1 dar. Dies vereinfacht spätere Schnittberechnungen (und wird von bereits implementierten Schnittberechnungen in der Klasse `Sphere` auch so erwartet).

Hinweis: Wenn Sie die Zeile `new Plane(...)` bei der Initialisierung der Objekte auskommentieren, sollten Sie nach Bearbeitung dieser Teilaufgabe schon die vier Kugeln im Bild erkennen können.

- (b) (3P) Ergänzen Sie den Code, analog zur Klasse `Sphere`, um eine Klasse `Plane` zum Verwalten von Ebenen. Eine Ebene wird über drei enthaltene Punkte a , b und c und ein Material definiert, und soll mit dem Aufruf `new Plane(a, b, c, material)` erzeugt werden können. Die Klasse soll zudem die folgenden Methoden bieten:

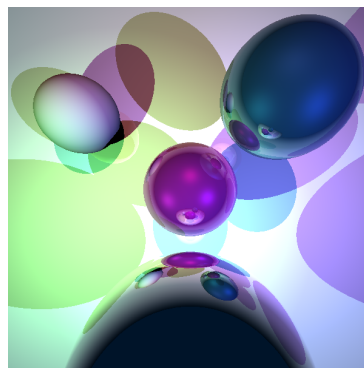
`normal()` Die Methode soll, analog zur Klasse `Sphere`, die Normale der Ebene zurückgeben.

`intersect(o, v)` Die Methode soll, analog zur Klasse `Sphere`, den ersten Schnittpunkt des Strahls mit Startpunkt o und Richtungsvektor v mit der Ebene, sowie die Entfernung des Schnittpunkts vom Startpunkt bestimmen. Gibt es keinen Schnittpunkt, so soll die Entfernung `Infinity` (mit beliebigem Punkt) zurückgegeben werden.

Hinweis: Fügen Sie die Zeile `new Plane(...)` nun wieder hinzu, falls Sie sie auskommentiert hatten.

- (c) (3P) Ergänzen Sie die Methode `trace` derart, dass sie rekursiv aufgerufen wird und somit den Strahl nach Reflexion weiterverfolgt um Spiegelungseffekte zu berechnen. Dazu muss der Reflexionsvektor berechnet werden. Nutzen Sie die Variablen `depth` und `maxDepth`, die durch den Slider eingestellt werden kann, um die Rekursionstiefe zu beschränken. Berechnen Sie die finale Farbe des Pixels aus der direkten Farbe, die durch die Methode `phong` berechnet wird, und der vom reflektierten Strahl zurückgelieferten Farbe. Die Reflektivität eines Materials ist in der Variablen `reflectivity` $\in [0, 1]$ gegeben.
- (d) (1P) Fügen Sie Code hinzu, der `visibilityFactor` auf 0 setzt, wenn sich ein Punkt bezüglich einer Lichtquelle im Schatten befindet (Stichwort "Shadow Feelers"). Dadurch sollten sich dann im Bild auch Schatten ergeben, wenn die Checkbox aktiviert ist.

Hinweis: Nach Bearbeitung aller Teilaufgaben sollte für eine maximale Rekursionstiefe von 3 das folgende Bild entstehen:



Hinweis: Die einzelnen Kugeln lassen sich per Drag-and-Drop in der Szene verschieben.



Aufgabe 3 (Bonus: Soft-Schatten & Super-Sampling)

- (a) Fügen Sie der Szene weitere Objekte mit neuen Typen hinzu, z.B. Tetraeder, Würfel, Zylinder oder Dreiecksnetze. Die Klassen der Objekte müssen dafür ebenfalls die Methoden `normal` und `intersect` implementieren.
- (b) Machen Sie die Punktlichtquellen zu sphärischen Lichtquellen, indem Sie Ihnen einen kleinen Radius zuweisen. Testen Sie dann (statt mit *einem* Shadow Feeler) mit mehreren Shadow Feelers zu unterschiedlichen auf jeder *Kugel*lichtquelle verteilten Punkten, um so auch anteilige Teilschatten ("Soft Shadows"), wie Sie von realen Lichtquellen erzeugt werden, realisieren zu können. Mit der Checkbox soll zwischen Punkt- und Kugellichtquellen gewechselt werden.
- (c) Fügen Sie Super-Sampling hinzu, um die Treppenstufenartefakte an den Rändern der Kugeln zu beheben. Der Faktor, der mit dem Slider eingestellt werden kann, befindet sich in der Variable `ssFactor`.