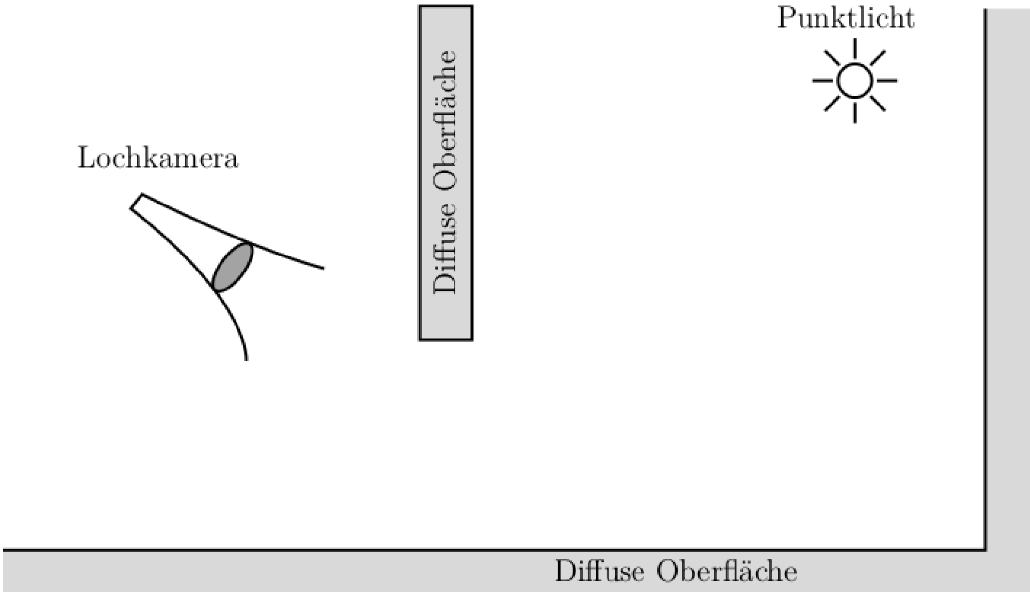
Aufgabe 1: Raytracing

Teilaufgabe 1a

Zeichnen Sie in der folgenden Szene mit Liniensegmenten einen möglichst einfachen Lichtpfad von der Kamera zur Punktlichtquelle ein, der zwar in der geometrischen Optik auftreten kann, jedoch nicht mit Whitted-Style Raytracing erzeugt werden kann!



TODO

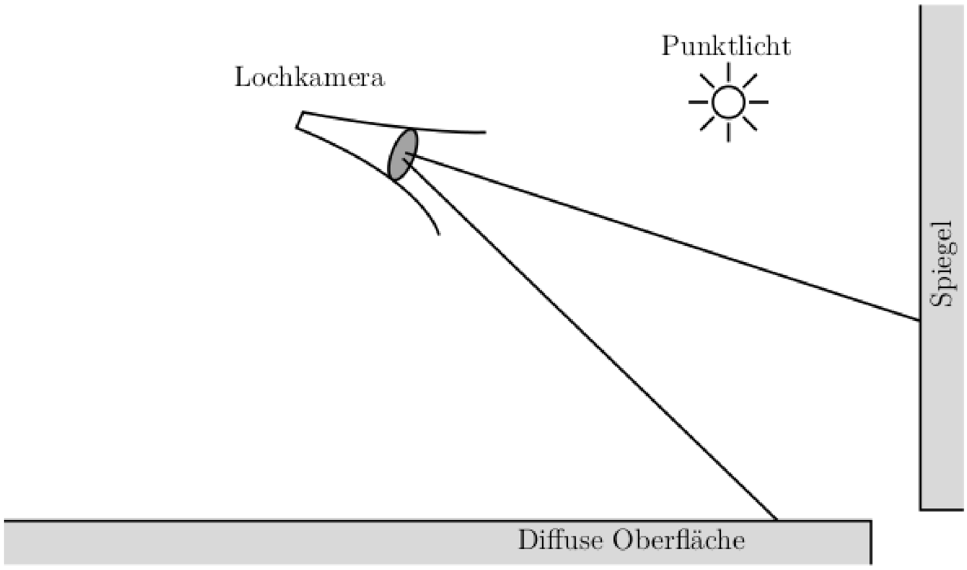
Teilaufgabe 1b

Begründen Sie kurz, warum der Pfad nicht mit Whitted-Style Raytracing erzeugt werden kann!

TODO

Teilaufgabe 1c

In der folgenden Abbildung sind zwei Primärstrahlen gegeben. Führen Sie mit diesen zeichnerisch Whitted-Style Raytracing durch! Kennzeichnen Sie Reflexionsstrahlen mit „R“, Transmissionsstrahlen mit „T“ und Schattenstrahlen mit „S“!



TODO

Teilaufgabe 1d

1. **Ray generation**: Erzeuge Sichtstrahlen durch jeden Pixel.
2. **Ray intersection**: Schnittberechnung; also: Finde Objekt welches den Strahl schneidet und am nahesten zur Kamera ist.
3. **Shading**: Schattierung / Beleuchtungsberechnung.

Aufgabe 2: Farben

Teilaufgabe 2a

Wie nennt man die Funktionen, mit denen man Tristimulus-Werte zu einem gegebenen Spektrum berechnen kann?

Color Matching Funktionen

Teilaufgabe 2b

- *Es gibt eine lineare Abbildung zwischen den Farbräumen XYZ und xyY.*

$\rightarrow$ Falsch, da bei der Umrechnung durch $X + Y + Z$ dividiert wird ($x = \frac{X}{X+Y+Z}$, $y = \frac{y}{X+Y+Z}$)

- *Es gibt eine lineare Abbildung zwischen den Farbräumen RGB und XYZ.*

→ Korrekt, der Farbraum wurden konstruiert, um eine möglichst einfache Umrechnung zu ermöglichen.

- *Die subjektiv empfundene Stärke von Sinneseindrücken ist proportional zur Intensität des physikalischen Reizes.*

→ Falsch, es besteht ein logarithmischer Zusammenhang.

Teilaufgabe 2c

Welche Information beinhalten die x- und y-Komponenten einer Farbdarstellung im CIE-xyY-Farbraum zusammengenommen?

Sie beschreiben die Farbe (Chromatizität) unabhängig von der Intensität.

Teilaufgabe 2d

Ein RGB-Eingabebild wird in den CIE-xyY-Farbraum transformiert. Um Speicher zu sparen werden die x- und y-Komponenten in geringerer Auflösung gespeichert, während die Auflösung der Y-Komponente beibehalten wird.

Warum ist dieses Vorgehen im Hinblick auf einen menschlichen Betrachter deutlich besser, als die Auflösung der RGB-Komponenten zu reduzieren?

Die Kontrastsentitivität für Chrominanz (Farbe) ist vor allem im hochfrequenten Bereich deutlich geringer als für Luminanz, daher kann man den Chrominanz-Anteil (xy-Werte) in geringerer Auflösung speichern (was dem Weglassen hoher Frequenzen entspricht), ohne dass mit den Augen ein großer Unterschied erkennbar ist.

Aufgabe 3: Homogene Koordinaten

Beim Raytracing soll ein in Weltkoordinaten definierter Strahl mit einem Objekt in dessen lokalen Koordinatensystem geschnitten werden. Die affine Transformation von lokalen in Weltkoordinaten ist durch die homogene Transformationsmatrix $M \in \mathbb{R}^{4\times4}$ gegeben.

Teilaufgabe 3a

Mit welcher Matrix kann der Strahl von Weltkoordinaten in lokale Koordinaten transformiert werden?

Mit der Matrix M^{-1}, da diese die inverse Transformation von Punkten aus *lokalen* in *Weltkoordinaten* darstellt.

Teilaufgabe 3b

Der Strahl $\mathbf{R}(t) = \mathbf{E} + t\mathbf{d}, (\mathbf{E}, \mathbf{d} \in \mathbb{R}^3)$ *soll mit der Matrix aus der Teilaufgabe a) in das lokale Koordinatensystem transformiert werden. Geben Sie dazu den transformierten Punkt* $\mathbf{E}'$ *und die transformierte Richtung* ***d'*** *an!*

Zur Vereinfachung geht man zu erweiterten Koordinaten $\begin{pmatrix}\mathbf{E}\\1\end{pmatrix}$ und $\begin{pmatrix}\mathbf{d}\\0\end{pmatrix}$ über (1 für Punkte, 0 für Richtungen, da diese Differenzen zweier Punkte darstellen).

Damit erhält man $\begin{pmatrix}\mathbf{E}'\\1\end{pmatrix} = M^{-1}\begin{pmatrix}\mathbf{E}\\1\end{pmatrix}$ und $\begin{pmatrix}\mathbf{d}'\\0\end{pmatrix} = M^{-1}\begin{pmatrix}\mathbf{d}\\0\end{pmatrix}$

Teilaufgabe 3c

Gegeben sei nun

$$M = \begin{pmatrix} 0 & 0 & -1 & 5 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Berechnen Sie für diese Werte die Matrix aus Aufgabe a)!

Invertiere die Matrix:

$$\left(\begin{array}{cccc|cccc} 0 & 0 & -1 & 5 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right) \longrightarrow \cdots \longrightarrow \left(\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 & 5 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right)$$

Allgemein gilt für eine Matrix der Form $M = \begin{pmatrix} A & t \\ 0 & 1 \end{pmatrix}$: $M^{-1} = \begin{pmatrix} A^{-1} & -A^{-1}t \\ 0 & 1 \end{pmatrix}$

Aufgabe 4: Transformationen

Teilaufgabe 4a

Gegeben sei folgendes Liniensegment $(\mathbf{p}_0, \mathbf{p}_1)$ *im Clip-Space:*

$$\mathbf{p}_0 = \begin{pmatrix} -1 \\ 1 \\ 5 \\ 10 \end{pmatrix}, \quad \mathbf{p}_1 = \begin{pmatrix} 0.5 \\ -0.5 \\ 15 \\ 5 \end{pmatrix}$$

Teilaufgabe 4a (i)

Wie lauten die Normalized Device Coordinates der Punkte $\mathbf{p}_0$ *und* $\mathbf{p}_0$*?*

TODO

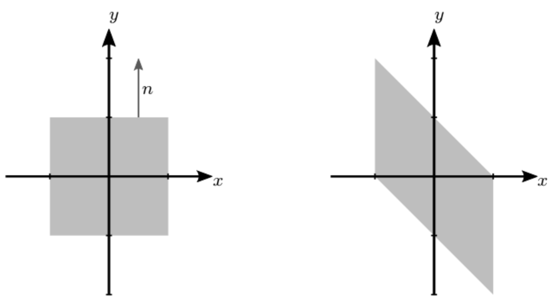
Teilaufgabe 4a (ii)

Warum sollte man dieses Liniensegment vor der Normalisierungstransformation (Dehomogenisierung) clippen?

TODO

Teilaufgabe 4b

Die folgende Abbildung zeigt ein Quadrat in 2D vor (links) und nach einer Scherung (rechts).



i) *Geben Sie die Transformationsmatrix* $M \in \mathbb{R}^{2\times 2}$ *an, die die Scherung beschreibt!*

Die Scherung hat zur Folge, dass beim Erhöhen der x-Koordinate um eine Einheit die y-Koordinate um eine Einheit verringert wird. Damit hat die Matrix die Form

$$M = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix}$$

ii) *Wie berechnet man allgemein die Matrix $N \in \mathbb{R}^{2\times 2}$ an, die zur Transformation der Normalen verwendet werden muss? Bestimmen Sie diese Matrix!*

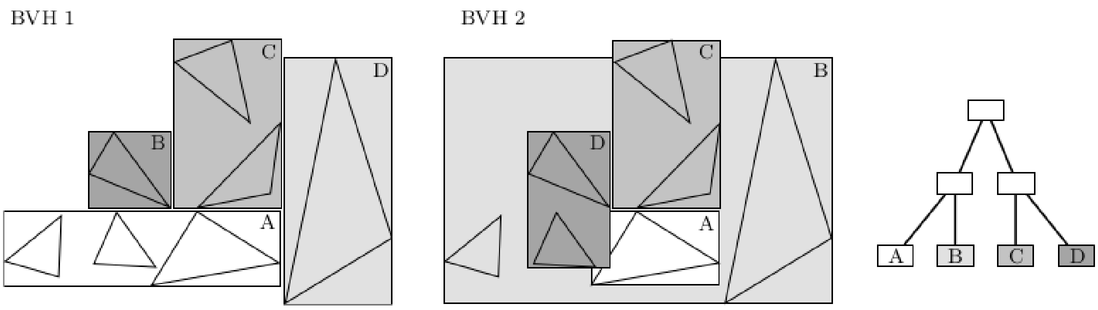
$$N = (M^{-1})^\top = \left(\begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix}^{-1} \right)^\top = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}^\top = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

iii) *Berechnen Sie mithilfe von N die transformierte Normale $\mathbf{n}'$ zur in der Abbildung eingezeichneten Normalen $\mathbf{n}$!*

$$\mathbf{n}' = \frac{N\mathbf{n}}{||N\mathbf{n}||} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Aufgabe 5: Beschleunigungsstrukturen und Hüllkörper

Teilaufgabe 5a



Welche der beiden Hierarchien kann von einem Raytracer effizienter durchlaufen werden? Begründen Sie dies in Stichpunkten!

TODO

Teilaufgabe 5b

#	Aussage	Wahr	Falsch	Kommentar
1	Ein kD-Baum kann für N Primitive den Aufwand für Strahlschnitte von $O(N)$ auf $O(\log(N))$ reduzieren.	☑	☐	average case
2	Die Surface Area Heuristic sorgt dafür, dass beiden Kindknoten gleich viele Primitive zugeteilt werden.	☐	☑	minimale Kosten
3	Für die Suche der Trennebene (Split Plane) mit Objektmittel (Object Median) ist immer eine vollständige Sortierung der Primitive notwendig.	☐	☑	Sortiert werden die Objektmittelpunkte
4	Teilen eines Knotens am Objektmittel (Object Median) führt stets zu den effizientesten Hüllkörperhierarchien.	☐	☑	SAH
5	Die Surface Area Heuristic macht die Annahme, dass Strahlen stets außerhalb des zu unterteilenden Hüllkörpers starten.	TODO	TODO	
6	Es gibt Szenen, in denen ein kD-Baum keinen Vorteil bringt.	☑	☐	Identische Dreiecke

Teilaufgabe 5c

TODO: Bitte kontrollieren!

#	Aussage	AABB	OBB	Kugel
1	Der Hüllkörper kann einen beliebig orientierten Würfel optimal, also ohne freien Raum umschliessen.	☑	☑	☐
2	Es gibt orthonormale Transformationen des eingeschlossenen Objektes, die das Volumen des optimalen Hüllkörpers verändern.	☑	☐	☐
3	Es gibt affine Transformationen des eingeschlossenen Objektes, die das Volumen des Hüllkörpers verändern.	☑	☐	☐
4	Gegegeben sind zwei Objektmengen und zu jeder Menge ihr optimaler Hüllkörper. Der Aufwand, den optimalen Hüllkörper für alle Objekte zu bestimmen, ist unabhängig von der Anzahl der Objekte.	☑	☐	☑

Aufgabe 6: Texturen

Teilaufgabe 6a

TODO

Teilaufgabe 6b

TODO

Aufgabe 7: Beleuchtung

Teilaufgabe 7a

$\mathbf{P} = \lambda_1 \mathbf{P}_1 + \lambda_2 \mathbf{P}_2 + \lambda_3 \mathbf{P}_3$, wobei $\lambda_1 + \lambda_2 + \lambda_3 = 1$ und $\lambda_1, \lambda_2, \lambda_3 \geq 0$

Teilaufgabe 7b

- berechne $I_i = k_d I_L(\mathbf{n}_i \cdot normalize(\mathbf{L} - \mathbf{P}_i))$ für $i = 1, 2, 3$
- interpoliere $I = \lambda_1 I_1 + \lambda_2 I_2 + \lambda_3 I_3$
- Bemerkung: $normalize(\mathbf{x}) = \frac{\mathbf{x}}{||\mathbf{x}||}$

Teilaufgabe 7c

- interpoliere $\mathbf{n} = normalize(\lambda_1 \mathbf{n}_1 + \lambda_2 \mathbf{n}_2 + \lambda_3 \mathbf{n}_3)$
- berechne $I = k_d I_L(\mathbf{n} \cdot normalize(\mathbf{L} - \mathbf{P}))$

Teilaufgabe 7d

In den Eckpunkten. Allgemein gilt im i-ten Eckpunkt: $\lambda_i = 1$ sowie $\lambda_j = 0\,(\forall j \neq i)$. Somit sind die Berechnungen in diesen Punkten äquivalent, wie man leicht aus den obrigen Formeln entnehmen kann.

Teilaufgabe 7e

Im rechten Fall wird mehr Licht zum betrachter reflektiert.

Teilaufgabe 7f

Fresnel-Effekt

Aufgabe 8: Partikeleffekte und OpenGL-Blending

Teilaufgabe 8a

TODO

Teilaufgabe 8b

TODO

Teilaufgabe 8c

TODO

Teilaufgabe 8d

TODO

Aufgabe 9: OpenGL

TODO

Aufgabe 10: Reflexionen in OpenGL

shader.frag

```
// ...
uniform float h; // Y-Koordinate der Wasserebene im World Space
in vec3 P; // World-Space-Position des aktuellen Fragments
// ...

void main(void)
{
    // Implementieren Sie hier das Clipping:
    if (P.y < h) {
```

```
        discard;
    }

    // Es folgt das spezifische Shading des jeweiligen Objekts
    // ...
}
```

Aufgabe 11: GLSL-Hatching

Teilaufgabe 11a

$$\left(1 - \frac{x - x_1}{x_2 - x_1}\right) y_1 + \frac{x - x_1}{x_2 - x_1} y_2$$

oder

$$y_1 + \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$$

Teilaufgabe 11b

shader.frag

```
uniform sampler2D dunkel; // dunkle Textur
uniform sampler2D mittel; // mittlere Textur
uniform sampler2D hell; // helle Textur
in vec2 tex_coord; // Texturkoordinate

vec4 get_hatched_color(float h) // Helligkeit h liegt in [0, 1].
{
    vec4 t_d = texture(dunkel, tex_coord);
    vec4 t_m = texture(mittel, tex_coord);
    vec4 t_h = texture(hell, tex_coord);

    float weight_d = clamp(1. - 4. * (h - .25), 0., 1.);
    float weight_m = clamp(1 - 4. * abs(h - .5), 0., 1.);
    float weight_h = clamp(4. * (h - .5), 0., 1.);

    return weight_d * t_d + weight_m * t_m + weight_h * t_h;
}
```

Aufgabe 12: Bézierkurven

Teilaufgabe 12a

TODO

Teilaufgabe 12b

TODO

Teilaufgabe 12c

1. Ja
2. Ja
3. Nein, da die Kurve nicht innerhalb der konvexen Hülle der Kontrollpunkte ist.