

2024-Test2_A

⚠ Disclaimer

Alles was hier drinnen steht kann Fehler enthalten!, Falls dir etwas auffällt melde dich gerne auf Discord bei mir ([@xmozz](#))

1-4

Angabe: Für eine Viewport Transformation sei ein Clipspace als ein achsenparalleler Würfel mit den Eckpunkten $(-1, -1, -1)$ und $(1, 1, 1)$ gegeben. Der Screenspace ist ein handelsüblicher 4k Monitor mit $n_x = 3840$ und $n_y = 2160$ Pixeln. Der Clipspace soll flächenfüllend auf den Screenspace abgebildet werden.

Tipp: Randwerte aus dem Clipspace werden Randpunkte im Screenspace. Die z-Werte sollen bei der Transformation erhalten bleiben.

Bestimmen Sie die Viewpoint Transformationsmatrix

$$M_{vp} = \begin{pmatrix} m_{0,0} & m_{0,1} & m_{0,2} & m_{0,3} \\ m_{1,0} & m_{1,1} & m_{1,2} & m_{1,3} \\ m_{2,0} & m_{2,1} & m_{2,2} & m_{2,3} \\ m_{3,0} & m_{3,1} & m_{3,2} & m_{3,3} \end{pmatrix}.$$

[Siehe hier](#)

1. Transformation der X-Koordinate:

Der X-Bereich im Clipspace ist $[-1, 1]$. Dieser soll auf den X-Bereich des Bildschirms abgebildet werden, der üblicherweise von 0 bis n_x geht (wobei $n_x = 3840$).

Die allgemeine Formel für die lineare Abbildung eines Intervalls $[a, b]$ auf $[c, d]$ ist:

$$x_{neu} = \frac{x_{alt} - a}{b - a} \cdot (d - c) + c$$

Für die X-Koordinate:

$$a = -1, b = 1, c = 0, d = n_x = 3840$$

$$x_{screen} = \frac{x_{clip} - (-1)}{1 - (-1)} \cdot (3840 - 0) + 0$$

$$x_{screen} = \frac{x_{clip} + 1}{2} \cdot 3840$$

$$x_{screen} = 1920 \cdot (x_{clip} + 1)$$

$$x_{screen} = 1920 \cdot x_{clip} + 1920$$

2. Transformation der Y-Koordinate:

Der Y-Bereich im Clipspace ist $[-1, 1]$. Dieser soll auf den Y-Bereich des Bildschirms abgebildet werden, der üblicherweise von 0 bis n_y geht (wobei $n_y = 2160$).

Für die Y-Koordinate:

$$a = -1, b = 1, c = 0, d = n_y = 2160$$

$$y_{screen} = \frac{y_{clip} - (-1)}{1 - (-1)} \cdot (2160 - 0) + 0$$

$$y_{screen} = \frac{y_{clip}+1}{2} \cdot 2160$$

$$y_{screen} = 1080 \cdot (y_{clip} + 1)$$

$$y_{screen} = 1080 \cdot y_{clip} + 1080$$

3. Transformation der Z-Koordinate:

Laut Aufgabenstellung sollen die Z-Werte erhalten bleiben. Das bedeutet:

$$z_{screen} = z_{clip}$$

4. Homogene Koordinaten und Matrixdarstellung:

Um diese Transformationen in einer 4x4-Matrix darzustellen, verwenden wir homogene Koordinaten. Ein Punkt (x, y, z) wird als $(x, y, z, 1)$ dargestellt. Die Transformation hat die Form:

$$\begin{pmatrix} x_{screen} \\ y_{screen} \\ z_{screen} \\ 1 \end{pmatrix} = M_{vp} \begin{pmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ 1 \end{pmatrix}$$

Vergleichen wir die hergeleiteten Gleichungen mit der Matrixmultiplikation:

- $x_{screen} = m_{00}x_{clip} + m_{01}y_{clip} + m_{02}z_{clip} + m_{03}$
Daraus folgt: $m_{00} = 1920, m_{01} = 0, m_{02} = 0, m_{03} = 1920$
- $y_{screen} = m_{10}x_{clip} + m_{11}y_{clip} + m_{12}z_{clip} + m_{13}$
Daraus folgt: $m_{10} = 0, m_{11} = 1080, m_{12} = 0, m_{13} = 1080$
- $z_{screen} = m_{20}x_{clip} + m_{21}y_{clip} + m_{22}z_{clip} + m_{23}$
Daraus folgt: $m_{20} = 0, m_{21} = 0, m_{22} = 1, m_{23} = 0$
- Für die homogene Komponente muss gelten $1 = m_{30}x_{clip} + m_{31}y_{clip} + m_{32}z_{clip} + m_{33}$.
Daraus folgt: $m_{30} = 0, m_{31} = 0, m_{32} = 0, m_{33} = 1$

Ergebnis

Die Viewpoint-Transformationsmatrix M_{vp} ist:

$$M_{vp} = \begin{pmatrix} 1920 & 0 & 0 & 1920 \\ 0 & 1080 & 0 & 1080 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Frage: (1) Welcher Wert befindet sich an der Stelle $m_{0,3}$? (2 Punkte, SC)

- | | |
|-----------------|---------|
| (A) 3840 | (E) 720 |
| (B) 2160 | (F) 540 |
| (C) 1920 | (G) 1 |
| (D) 1080 | (H) 0 |

Frage: (2) Welcher Wert befindet sich an der Stelle $m_{1,1}$? (2 Punkte, SC)

- | | |
|-----------------|---------|
| (A) 3840 | (E) 720 |
| (B) 2160 | (F) 540 |
| (C) 1920 | (G) 1 |
| (D) 1080 | (H) 0 |

Frage: (3) Welcher Wert befindet sich an der Stelle $m_{2,2}$? (2 Punkte, SC)

- | | |
|----------|--------------|
| (A) 3840 | (E) 720 |
| (B) 2160 | (F) 540 |
| (C) 1920 | (G) 1 |
| (D) 1080 | (H) 0 |

Frage: (4) Welcher Wert befindet sich an der Stelle $m_{3,3}$? (2 Punkte, SC)

- | | |
|----------|--------------|
| (A) 3840 | (E) 720 |
| (B) 2160 | (F) 540 |
| (C) 1920 | (G) 1 |
| (D) 1080 | (H) 0 |

5

Frage: (5) Welche der folgenden Aussagen zu Projektionstransformationen treffen zu? (2 Punkte, MC)

- | | |
|--|--|
| (A) Je weiter das Projektionszentrum von der Abbildungsebene entfernt ist, desto mehr ähnelt das Resultat einer orthographischen Projektion. | (C) Nach der Projektionstransformation entspricht das Bild einer orthographischen Transformation dem Bild einer perspektivischen Projektion im View Frustum. |
| (B) Nach Projektionstransformationen sind alle parallelen Linien wieder parallel. | (D) Bei Projektionstransformationen bleiben die z-Werte erhalten. |

B:

- Das ist **nur bei orthographischer Projektion** korrekt.
- Bei **perspektivischer Projektion** können parallele Linien in einem Fluchtpunkt zusammenlaufen (z. B. Bahngleise), d. h. sie bleiben **nicht** parallel.
- Da die Aussage absolut formuliert ist („alle parallelen Linien“), ist sie **falsch**.

D:

- Das ist **falsch**, denn bei der perspektivischen Projektion werden die **z-Werte verändert** (z.B. zur Tiefenberechnung im z-Buffer), insbesondere durch die Division durch w .
- Auch bei der orthographischen Projektion werden die z-Werte skaliert und transformiert, um ins canonical view volume zu passen.

6

Frage: (6) Welche der folgenden Aussagen zum Abtasttheorem treffen zu? (4 Punkte, MC)

- (A) Die Fouriertransformation beschreibt ein Signal im Bildraum als Summe von Sinusschwingungen im Frequenzraum.
- (B) Eine Faltung im Frequenzraum entspricht einer Faltung im Bildraum, eine Multiplikation im Bildraum entspricht einer Multiplikation im Frequenzraum.
- (C) Die Abstände der Zacken der Kammfunktion im Bildraum entsprechen der Abtastfrequenz.
- (D) Beim Abtasten mit Frequenzen über Nyquist-Limit kommt es zu Aliasing.
- (E) Das diskretisierte Signal im Bildraum ist die Frequenz und Amplitude des Signals an den Abtastpunkten.
- (F) Die Zahnabstände der Kammfunktionen im Bildraum und im Frequenzraum sind direkt proportional.
- (G) Im Frequenzraum führt ein Überlappen der Schattenspektra zu Aliasing.
- (H) Ein durch Multiplikation mit der Kammfunktion diskretisiertes Signal kann durch Faltung mit dem sinc-Filter ($\text{sinc} = \sin(x)/x$) rekonstruiert werden.

B - Falsch weil:

- Eine Faltung im Bild-/Zeitraum entspricht einer Multiplikation im Frequenzraum.
- Eine Multiplikation im Bild-/Zeitraum entspricht einer Faltung im Frequenzraum. Die Aussage verdreht diese Beziehungen.

D - Nein wenns darunter liegt

E - Falsch - Das diskretisierte Signal im Bildraum besteht aus den **Amplitudenwerten** des Signals an den einzelnen Abtastpunkten. Es enthält nicht direkt die Frequenzinformation an jedem Punkt; Frequenzen werden durch die Fouriertransformation des gesamten Signals ermittelt.

F - Falsch, sind **indirekt (reziprok) proportional**, nicht direkt proportional ($\frac{1}{T}$)

7

Frage: (7) Welche der folgenden Sichtbarkeitsverfahren finden im Bildraum statt? (3 Punkte, MC)

- (A) Backface Detection
- (B) Octree Methode
- (C) Ray Casting
- (D) Z-Puffer
- (E) Depth Sorting
- (F) Area Subdivision

- **Objektraumverfahren** arbeiten mit den geometrischen Daten der Objekte in der Szene und bestimmen, welche Teile der Objekte sichtbar sind, bevor sie auf die Bildebene projiziert werden.
- **Bildraumverfahren** arbeiten für jeden einzelnen Bildpunkt (Pixel) und bestimmen, welches Objekt an dieser Stelle sichtbar ist, nachdem die Objekte bereits auf die Bildebene projiziert wurden.

- **(A) Backface Detection (Backface Culling)**: Dies ist ein Objektraumverfahren. Es eliminiert Polygone, deren Rückseite zum Betrachter zeigt, noch bevor die eigentliche Sichtbarkeitsprüfung im Bildraum beginnt. Es ist eine Vorselektion.
- **(B) Octree Methode**: Die Octree-Methode ist ein Raumpartitionierungsverfahren, das sowohl im Objektraum als auch im Bildraum angewendet werden kann, aber primär zur Organisation von Objekten im **Objektraum** dient, um Kollisionen oder Sichtbarkeiten effizienter zu bestimmen. Für die reine Sichtbarkeitsprüfung ist es eher ein unterstützendes Verfahren.
- **(C) Ray Casting**: Ray Casting ist ein Bildraumverfahren. Für jeden Pixel auf der Bildebene wird ein Strahl in die Szene geschickt, und es wird der Schnittpunkt mit dem nächstgelegenen Objekt bestimmt.
- **(D) Z-Puffer (Depth Buffer)**: Der Z-Puffer ist ein klassisches Bildraumverfahren. Für jeden Pixel wird die Tiefeninformation des vordersten Objekts gespeichert. Beim Zeichnen weiterer Objekte wird deren Tiefe mit der im Z-Puffer gespeicherten Tiefe verglichen, um zu entscheiden, ob das neue Objekt sichtbar ist oder nicht.
- **(E) Depth Sorting (Painter's Algorithm)**: Dies ist ein Objektraumverfahren. Objekte werden nach ihrer Tiefe sortiert und dann von hinten nach vorne gezeichnet. Dies kann Probleme bei überlappenden Objekten haben, aber es ist eine Operation, die auf den Objekten selbst basiert, bevor sie endgültig auf die Bildebene gerendert werden.
- **(F) Area Subdivision**: Die Area Subdivision (z.B. Warnock-Algorithmus) ist ein Bildraumverfahren. Sie teilt den Bildschirm rekursiv in kleinere Bereiche auf, bis die Sichtbarkeit in jedem Bereich trivial ist.

8

Frage: (8) Diese Formel beschreibt ein einfaches komplettes Beleuchtungsmodell:

$$L = k_a * I_a + \sum_{i=1,\dots,N} (k_d * I_i * (n \cdot l) + k_s * I_i * (n \cdot h_i)^P)$$

Welche der folgenden Aussagen treffen zu? (4 Punkte, MC)

- | | |
|--|--|
| (A) $k_a * I_a$ beschreibt Glanzpunkte. | (E) $k_d * I_i * (n \cdot l)$ beschreibt Hintergrundlicht. |
| (B) $k_d * I_i * (n \cdot l)$ beschreibt Lambert'sche Beleuchtung. | (F) $k_s * I_i * (n \cdot h_i)^P$ beschreibt Glanzpunkte. |
| (C) $k_a * I_a$ beschreibt Hintergrundlicht. | (G) $k_s * I_i * (n \cdot h_i)^P$ beschreibt Lambert'sche Beleuchtung. |
| (D) $k_a * I_a$ beschreibt Lambert'sche Beleuchtung. | (H) $k_d * I_i * (n \cdot l)$ beschreibt Glanzpunkte. |

9

Frage: (9) Welche der folgenden Aussagen zu Shading treffen zu? (3 Punkte, MC)

- | | |
|--|--|
| (A) Phong-Shading interpoliert Normalen. | (D) Gouraud-Shading interpoliert Helligkeiten. |
| (B) Flat-Shading interpoliert Normalen. | (E) Flat-Shading interpoliert Helligkeiten. |
| (C) Phong-Shading interpoliert Helligkeiten. | (F) Gouraud-Shading interpoliert Normalen. |

1. Flat-Shading (Konstante Schattierung):

- Bei Flat-Shading wird die Farbe und Helligkeit für jedes Polygon nur einmal berechnet (z.B. im Zentrum oder an einem Eckpunkt) und dann über das gesamte Polygon **konstant** gehalten. Es gibt keine Interpolation von Normalen oder Helligkeiten über das Polygon hinweg. Dies führt zu einem "facettenreichen" Aussehen mit deutlichen Kanten.
 - (B) Flat-Shading interpoliert Normalen. (Falsch)
 - (E) Flat-Shading interpoliert Helligkeiten. (Falsch)

2. Gouraud-Shading (Intensitäts-Interpolation):

- Gouraud-Shading berechnet die Helligkeit an den **Eckpunkten** eines Polygons. Dafür werden die Normalen der Eckpunkte (oft gemittelt aus den Normalen der angrenzenden Polygone) verwendet, um eine Helligkeit für jeden Eckpunkt zu bestimmen. Anschließend werden diese **Helligkeiten über das Polygon interpoliert**. Dies führt zu einer glatteren Erscheinung als Flat-Shading, kann aber Glanzpunkte "verschmieren", da die Helligkeit interpoliert wird und nicht die Normalen selbst.
 - (D) Gouraud-Shading interpoliert Helligkeiten. (Richtig)
 - (F) Gouraud-Shading interpoliert Normalen. (Falsch – es verwendet Normalen an den Eckpunkten zur Berechnung der Helligkeit, interpoliert aber dann die Helligkeit, nicht die Normalen selbst über das Polygon.)

3. Phong-Shading (Normalen-Interpolation):

- Phong-Shading berechnet die Helligkeit für **jeden einzelnen Pixel** innerhalb eines Polygons. Dafür werden die Normalen der Eckpunkte über das Polygon **interpoliert**. Anschließend wird mit der interpolierten Normalen für jeden Pixel die Beleuchtung (einschließlich diffuser und spekularer Anteile) berechnet. Dies führt zu den realistischsten Glanzpunkten und einer sehr glatten Erscheinung.
 - (A) Phong-Shading interpoliert Normalen. (Richtig)
 - (C) Phong-Shading interpoliert Helligkeiten. (Falsch – es berechnet die Helligkeiten pro Pixel basierend auf den interpolierten Normalen, interpoliert aber nicht die Helligkeiten direkt über das Polygon wie Gouraud.)

Angabe: Gegeben sei eine Szene bestehend aus zwei perfekt spiegelnden Kugeln mit den Zentren $C_1 = (0, 0, 0)^T$ und $C_2 = (50, 0, 0)^T$, beide mit Radius $R_1 = R_2 = 1$. Diese Szene ist komplett von einer weiteren, emittierenden Kugel mit Radius $R_0 = 100$ und Zentrum $C_0 = (0, 0, 0)^T$ umgeben, deren Normalvektoren in Richtung des Ursprungs zeigen. Sei θ mit $0 \leq \theta \leq \pi$ der vertikale Zenitwinkel. Die untere Hälfte ($\theta \in [\pi/2, \pi]$) der emittierende Kugel ist blau. Für $\theta \in [0, \pi/4]$ hat die emittierende Kugel Farbe A (pink) und für $\theta \in [\pi/4, \pi/2]$ hat sie die Farbe B (orange).

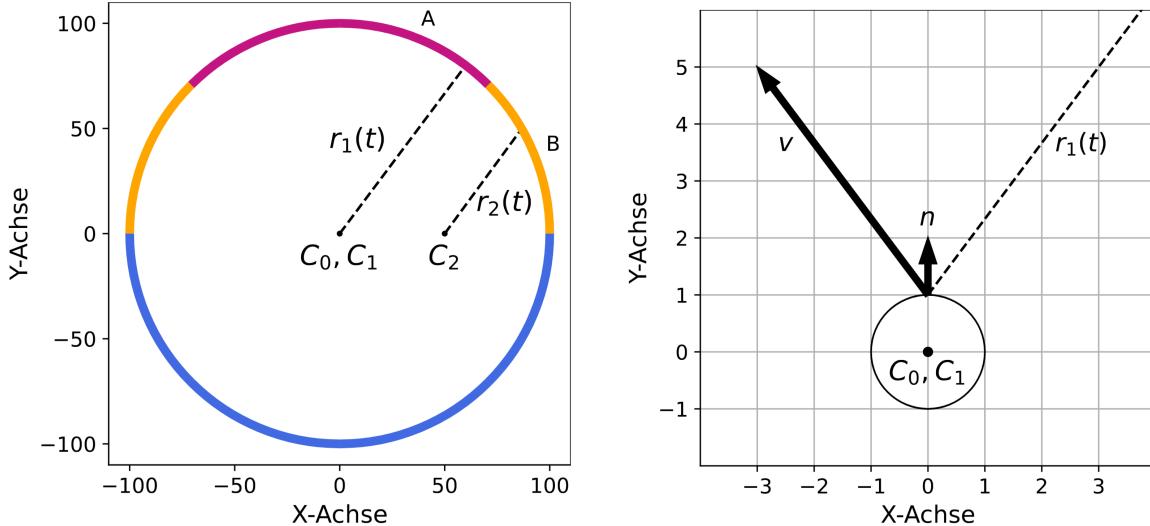


Abbildung 2: Szenenaufbau

Die Szene wird mit zwei verschiedenen Kameras gerendert, die jeweils eine der beiden kleinen Kugeln betrachten. Dabei sieht die erste Kamera den Punkt $P_1 = (0, 1, 0)$ auf der Kugel mit Zentrum C_1 und die zweite Kamera den Punkt $P_2 = (50, 1, 0)$ auf der Kugel mit Zentrum C_2 . Aus Sicht beider Punkte auf den Kugeln befindet sich die dazugehörige Kamera in Richtung $v = (-3, 4, 0)^T$ und beide Punkte haben die Richtung $(0, 1, 0)$ als Normalenvektoren. Abbildung 2 zeigt zwei 2D Querschnitte entlang der XY-Ebene der beschriebenen Szene.

Hinweise: Reflektionsrichtung: $r = (2n \cdot v)n - v$, Kugelgleichung: $\|p - c\|^2 - R^2 = 0$, Reflektionsstrahl: $r(t) = p + t \cdot r$

Frage: (10) Welche Aussagen treffen bezüglich des Renderings der oben beschriebenen Szene zu? (3 Punkte, MC)

- (A) Rendert man die Szene mittels Ray Tracing, würde Punkt P_2 Farbe A erhalten.
- (B) Rendert man die Szene mittels Reflektionen per Environment Mapping, würde Punkt P_2 Farbe B erhalten.
- (C) Rendert man die Szene mittels Reflektionen per Environment Mapping, würde Punkt P_2 Farbe A erhalten.
- (D) Rendert man die Szene mittels Reflektionen per Environment Mapping, würde Punkt P_1 Farbe A erhalten.
- (E) Rendert man die Szene mittels Reflektionen per Environment Mapping, würde Punkt P_1 Farbe B erhalten.
- (F) Rendert man die Szene mittels Ray Tracing, würde Punkt P_2 Farbe B erhalten.

Analyse der Szene und Kameras:

- **Szene:**

- Zwei perfekt spiegelnde Kugeln mit Zentren $C_1 = (0, 0, 0)^T$ und $C_2 = (50, 0, 0)^T$, beide mit Radius $R_1 = R_2 = 1$.
- Umfassende emittierende Kugel: Radius $R_0 = 100$, Zentrum $C_0 = (0, 0, 0)^T$.
 - Farbbereiche basierend auf dem vertikalen Zenitwinkel θ (Winkel von der positiven Y-Achse, da Normalenvektoren zum Ursprung zeigen):
 - Untere Hälfte ($\theta \in [\pi/2, \pi]$): Blau.

- $\theta \in [0, \pi/4]$: Farbe A (Pink).
 - $\theta \in [\pi/4, \pi/2]$: Farbe B (Orange).
- **Kameras:**
- **Kamera 1:** Sieht vom Punkt $P_1 = (0, 1, 0)^T$ auf Kugel C_1 .
 - **Kamera 2:** Sieht vom Punkt $P_2 = (50, 1, 0)^T$ auf Kugel C_2 .
 - Blickrichtung für beide Kameras: $v = (-3, 4, 0)^T$. (Dieser Vektor ist der Sehstrahl, der auf den Punkt trifft).

Berechnung der reflektierten Farbe:

Die kleinen Kugeln sind perfekt spiegelnd, daher ist ihre sichtbare Farbe die Farbe, die sie von der emittierenden Umgebungskugel reflektieren. Wir müssen den reflektierten Strahl r bestimmen und dessen Zenitwinkel θ ermitteln.

Für Punkt $P_1 = (0, 1, 0)^T$ auf Kugel C_1 :

1. **Normalenvektor n an P_1 :** Da $C_1 = (0, 0, 0)$, ist der Normalenvektor am Punkt $(0, 1, 0)$ gleich $(0, 1, 0)$.
2. **Blickvektor v (normiert):** $v_{unnormiert} = (-3, 4, 0)^T$. Länge $= \sqrt{(-3)^2 + 4^2 + 0^2} = 5$.
 $v = (-3/5, 4/5, 0)^T$.
3. **Berechnung des Reflexionsstrahls $r = (2n \cdot v)n - v$:**
 - $n \cdot v = (0, 1, 0) \cdot (-3/5, 4/5, 0) = 4/5$.
 - $r = (2 \cdot 4/5) \cdot (0, 1, 0) - (-3/5, 4/5, 0)$
 - $r = (8/5) \cdot (0, 1, 0) - (-3/5, 4/5, 0)$
 - $r = (0, 8/5, 0) - (-3/5, 4/5, 0)$
 - $r = (3/5, 4/5, 0)$.

4. **Bestimmung des Zenitwinkels θ des Reflexionsstrahls:**

Der Zenitwinkel θ ist der Winkel zur positiven Y-Achse. Für einen Vektor (x, y, z) ist $\cos \theta = y / \text{Länge}(Vektor)$. Da r bereits ein Einheitsvektor ist (Länge $\sqrt{(3/5)^2 + (4/5)^2 + 0^2} = \sqrt{9/25 + 16/25} = \sqrt{25/25} = 1$), ist $\cos \theta = 4/5$.
 $\theta = \arccos(4/5) \approx 36.87^\circ \approx 0.6435$ Radian.

5. **Zuordnung der Farbe:**

Der Winkel 0.6435 Radian liegt im Bereich $[0, \pi/4]$ ($[0, 0.785]$ Radian).

Daher erhält Punkt P_1 die **Farbe A (Pink)**.

Für Punkt $P_2 = (50, 1, 0)^T$ auf Kugel C_2 :

- Der Normalenvektor an P_2 auf Kugel C_2 ist ebenfalls $(0, 1, 0)$ (relativ zu C_2 , oder absolut $(0, 1, 0)$ in einer lokalen Koordinate der Kugel).
- Die Blickrichtung $v = (-3, 4, 0)^T$ ist dieselbe.
- Die Berechnung des Reflexionsstrahls r und des Zenitwinkels θ führt zu den **gleichen Ergebnissen** wie für P_1 .
- Daher erhält Punkt P_2 ebenfalls die **Farbe A (Pink)**.

Bewertung der Optionen:

- **Ray Tracing:** Simuliert das Verhalten von Lichtstrahlen. Ein Strahl vom Auge trifft auf die spiegelnde Oberfläche, wird reflektiert, und der reflektierte Strahl trifft auf die Umgebungskugel, deren Farbe dann sichtbar wird. Die obige Berechnung ist direkt anwendbar.
 - (A) Rendert man die Szene mittels Ray Tracing, würde Punkt P_2 Farbe A erhalten.
(Richtig)
 - (F) Rendert man die Szene mittels Ray Tracing, würde Punkt P_2 Farbe B erhalten.
(Falsch)
- **Reflexionen per Environment Mapping:** Eine Technik, bei der die Umgebung als Textur (Environment Map) auf die Oberfläche projiziert wird, basierend auf der Reflexionsrichtung. Da die Umgebungskugel die primäre Lichtquelle für Reflexionen ist, ist dies eine geeignete Methode, die ähnliche Ergebnisse wie Ray Tracing liefern sollte.
 - (B) Rendert man die Szene mittels Reflexionen per Environment Mapping, würde Punkt P_2 Farbe B erhalten.
(Falsch)
 - (C) Rendert man die Szene mittels Reflexionen per Environment Mapping, würde Punkt P_2 Farbe A erhalten.
(Richtig)
 - (D) Rendert man die Szene mittels Reflexionen per Environment Mapping, würde Punkt P_1 Farbe A erhalten.
(Richtig)
 - (E) Rendert man die Szene mittels Reflexionen per Environment Mapping, würde Punkt P_1 Farbe B erhalten.
(Falsch)

Frage: (11) Welche Aussagen treffen generell bezüglich der oben beschriebenen Szene zu? (2 Punkte, MC)

- | | |
|---|---|
| (A) Die beiden Punkten P_1 und P_2 haben unterschiedliche Reflektionsrichtungen.
(B) An beiden Punkten P_1 und P_2 ist die Reflektionsrichtung $r = (3/5, 4/5)$. | (C) Ein vom Ursprung ausgehender Strahl mit Richtung $r = (3/5, 4/5)$ schneidet die emittierende Kugel an der Position $(50\sqrt{2}, 50\sqrt{2})$.
(D) Ein vom Ursprung ausgehender Strahl mit Richtung $r = (3/5, 4/5)$ schneidet die emittierende Kugel an der Position $(60, 80)$. |
|---|---|

Begründung:

1. Reflexionsrichtung für P1 und P2:

- Die Szene beschreibt zwei spiegelnde Kugeln. Die Reflexionsrichtung r wird durch die Formel $r = (2n \cdot v)n - v$ bestimmt, wobei n der Normalenvektor der Oberfläche am Punkt und v der Einfallvektor des Blickstrahls ist.
- **Für Punkt P1 = (0,1,0) auf Kugel C1 (Zentrum (0,0,0)):**
 - Der Normalenvektor n am Punkt (0,1,0) auf einer Kugel mit Zentrum (0,0,0) ist der normierte Vektor vom Zentrum zum Punkt, also $n = (0, 1, 0)$.
 - Die Blickrichtung (Einfallvektor des Augestrahl) ist $v = (-3, 4, 0)^T$. Normiert ist dies $v = (-3/5, 4/5, 0)^T$.
 - Berechnung: $n \cdot v = (0, 1, 0) \cdot (-3/5, 4/5, 0) = 4/5$.

- $r = (2 \cdot 4/5) \cdot (0, 1, 0) - (-3/5, 4/5, 0) = (8/5) \cdot (0, 1, 0) + (3/5, -4/5, 0) = (0, 8/5, 0) +$

• Für Punkt P2 = (50,1,0) auf Kugel C2 (Zentrum (50,0,0)):

- Der Punkt P2 ist (50,1,0) und das Zentrum C2 ist (50,0,0). Der Vektor von C2 nach P2 ist (0,1,0). Daher ist der Normalenvektor n an P2 ebenfalls (0, 1, 0).
- Die Blickrichtung v ist dieselbe: $v = (-3/5, 4/5, 0)^T$.
- Da n und v für P1 und P2 identisch sind, ist auch die resultierende Reflexionsrichtung $r = (3/5, 4/5, 0)$ für P2 identisch.
- Daher sind die Reflexionsrichtungen an beiden Punkten identisch. Dies bestätigt Aussage (B) und widerlegt Aussage (A).

2. Schnittpunkt mit der emittierenden Kugel:

- Die emittierende Kugel hat Zentrum $C_0 = (0, 0, 0)^T$ und Radius $R_0 = 100$.
- Ein Strahl, der vom Ursprung ausgeht (d.h. vom Zentrum der Kugel), trifft die Oberfläche der Kugel in Richtung des Strahls.
- Wenn der Strahl die Richtung $r = (3/5, 4/5, 0)$ hat, ist der Schnittpunkt einfach der Punkt auf der Oberfläche im Abstand des Radius in dieser Richtung.
- Schnittpunkt = $R_0 \cdot r = 100 \cdot (3/5, 4/5, 0) = (100 \cdot 3/5, 100 \cdot 4/5, 100 \cdot 0) = (60, 80, 0)$.
- Aussage (C) gibt $(50\sqrt{2}, 50\sqrt{2})$ an, was ungefähr $(70.7, 70.7)$ ist und nicht $(60, 80)$. Daher ist (C) falsch.
- Aussage (D) gibt $(60, 80)$ an, was mit unserer Berechnung übereinstimmt (die Z-Koordinate ist 0). Daher ist (D) richtig.

12

Frage: (12) Welche der folgenden Aussagen zum Radiosity-Verfahren treffen zu? (4 Punkte, MC)

- | | |
|--|---|
| (A) Radiosity muss für jeden Blickwinkel neu berechnet werden. | (C) Radiosity erzeugt KEINE Glanzpunkte. |
| (B) Radiosity kann indirekte Beleuchtung berücksichtigen. | (D) Radiosity kann Lichtbrechung berücksichtigen. |

Begründung:

1. View-Independence vs. View-Dependence (A):

- Das Radiosity-Verfahren berechnet den Energietransport zwischen Oberflächen in einer Szene. Es resultiert in einer Verteilung der Farbhelligkeit (Radiosity-Werte) für jede Oberfläche (oder jedes Patch einer Oberfläche). Diese Berechnung ist unabhängig vom Standpunkt des Betrachters. Sobald die Radiosity-Werte berechnet sind, kann die Szene von jedem beliebigen Blickwinkel aus gerendert werden, ohne die Radiosity-Berechnung zu wiederholen. Nur die finale Projektion und Schattierung (auf Basis der bereits berechneten Radiosity-Werte) muss für den neuen Blickwinkel durchgeführt werden. Daher muss Radiosity *nicht* für jeden Blickwinkel neu berechnet werden. Aussage (A) ist falsch.

2. Indirekte Beleuchtung (B):

- Radiosity modelliert, wie Licht, das von einer Lichtquelle emittiert wird, von Oberflächen diffus reflektiert wird und dann andere Oberflächen beleuchtet. Dies ist der Kern der indirekten Beleuchtung, d.h., Licht, das nicht direkt von einer Lichtquelle, sondern über Reflexionen von anderen Oberflächen kommt. Dies ist eine der Hauptstärken von Radiosity. Aussage (B) ist richtig.

3. Glanzpunkte (C):

- Das Standard-Radiosity-Verfahren basiert auf einem Modell von perfekt diffusen (Lambert'schen) Oberflächen. Solche Oberflächen streuen das Licht gleichmäßig in alle Richtungen und erzeugen keine spiegelnden Reflexionen oder Glanzpunkte. Um Glanzpunkte zu modellieren, müssten hybride Render-Techniken (z.B. Radiosity für diffuse Komponenten und Ray Tracing für spiegelnde Komponenten) verwendet werden. Im reinen Radiosity-Modell sind Glanzpunkte nicht enthalten. Aussage (C) ist richtig.

4. Lichtbrechung (D):

- Radiosity ist primär für die Modellierung des Energietransports bei diffusen Reflexionen konzipiert. Es berücksichtigt nicht Phänomene wie Lichtbrechung (Refraktion) oder die Transmission von Licht durch transparente Materialien. Diese Effekte werden typischerweise durch Ray Tracing oder Path Tracing gehandhabt. Aussage (D) ist falsch.

13

Frage: (13) Welche der folgenden Aussagen zum Texture Mapping treffen zu? (4 Punkte, MC)

- | | |
|--|---|
| (A) Multitexturing kombiniert verschiedene Texturen. | (C) Für die perspektivische Korrektur muss die Interpolation NACH der Homogenisierung erfolgen. |
| (B) Texturen geben 3D-Objekten mehr Details. | (D) Das Mapping verbindet Textur-Raum, Objekt-Raum und Bild-Raum. |

14-18

Angabe: Eine animierte Kamera soll ein am Ursprung positioniertes Objekt betrachten und dabei ihre Position entlang eines Pfades ändern, der durch die folgenden drei Punkte geht: $P_0 = (-2, 0)^T$, $P_1 = (-1, 3)^T$, $P_2 = (2, -1)^T$. Dieser Szenenaufbau ist in Abbildung 3 dargestellt. Zum Berechnen des Pfades werden Hermite Splines verwendet, wobei die Kurve in zwei Segmente S_1 und S_2 aufgeteilt wird. Für die Tangentensteigung an den drei Punkten sollen folgende Werte gelten: $T_0 = (-1, 0)^T$, $T_1 = (3, 0)^T$, $T_2 = (-3, 0)^T$.

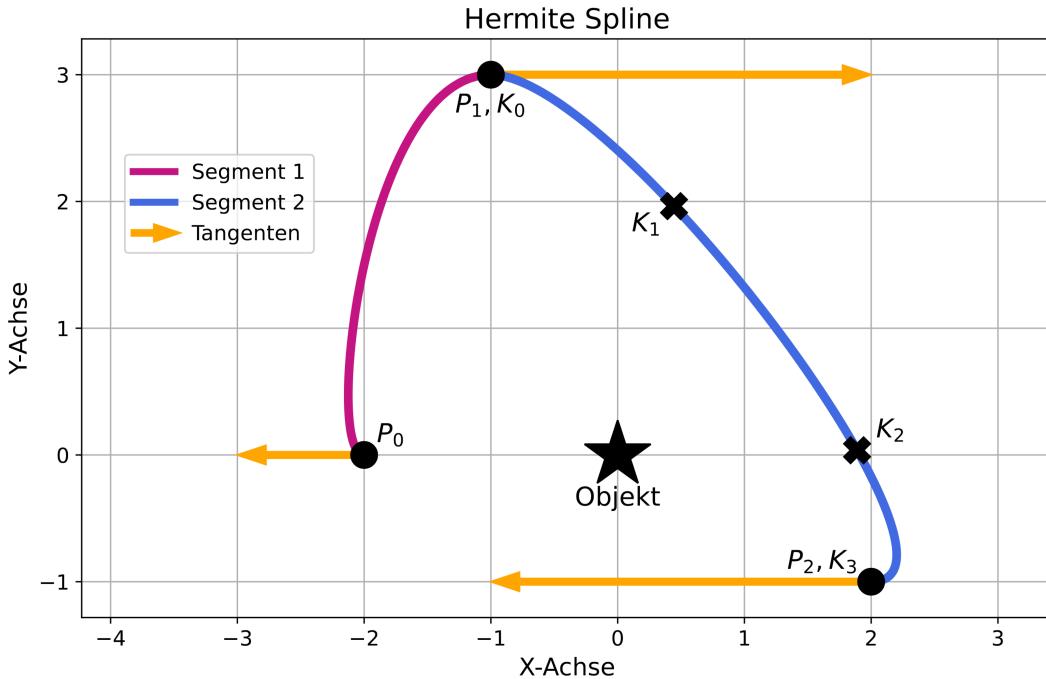


Abbildung 3: Szenenaufbau

Die parametrische Form eines Hermite Spline Segmentes S kann als Vektorgleichung der Form $P_S(t) = at^3 + bt^2 + ct + d$, $0 \leq t \leq 1$ geschrieben werden, wobei a, b, c und d die vier Koeffizientenvektoren des Segmentes sind.

Hinweis: Die Hermite-Matrix M_H sei gegeben als

$$M_H = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Frage: (14) Welchen Wert hat der Koeffizientenvektor a in der parametrische Form des ersten Hermite Spline Segmentes S_1 zwischen P_0 und P_1 ? (2 Punkte, SC)

- | | |
|----------------------|----------------------|
| (A) $a = (-2, -5)^T$ | (E) $a = (-2, -6)^T$ |
| (B) $a = (0, -5)^T$ | (F) $a = (1, 6)^T$ |
| (C) $a = (0, 6)^T$ | (G) $a = (0, -6)^T$ |
| (D) $a = (-2, 6)^T$ | (H) $a = (1, -6)^T$ |

Um den Koeffizientenvektor a für das erste Hermite Spline Segment S_1 zwischen P_0 und P_1 zu bestimmen, nutzen wir die Beziehung zwischen den Koeffizientenvektoren und den Kontrollpunkten/Tangentenvektoren über die Hermite-Matrix M_H .

Die allgemeine Formel für die Koeffizientenvektoren (a, b, c, d) eines Hermite Spline Segments ist:

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = M_H \begin{bmatrix} P_0 \\ P_1 \\ T_0 \\ T_1 \end{bmatrix}$$

Für das erste Segment S_1 zwischen P_0 und P_1 sind die gegebenen Werte:

- Endpunkt 1: $P_0 = (-2, 0)^T$
- Endpunkt 2: $P_1 = (-1, 3)^T$
- Tangente an Endpunkt 1: $T_0 = (-1, 0)^T$
- Tangente an Endpunkt 2: $T_1 = (3, 0)^T$

Die Hermite-Matrix M_H ist gegeben als:

$$M_H = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Der Koeffizientenvektor a ist der erste Zeile des Ergebnisses der Matrixmultiplikation. Das bedeutet, a ergibt sich aus der ersten Zeile von M_H multipliziert mit dem Spaltenvektor der Kontrollpunkte und Tangenten:

$$a = 2 \cdot P_0 - 2 \cdot P_1 + 1 \cdot T_0 + 1 \cdot T_1$$

Setzen wir die gegebenen Vektoren ein:

$$a = 2 \cdot \begin{pmatrix} -2 \\ 0 \end{pmatrix} - 2 \cdot \begin{pmatrix} -1 \\ 3 \end{pmatrix} + 1 \cdot \begin{pmatrix} -1 \\ 0 \end{pmatrix} + 1 \cdot \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

Berechne die X- und Y-Komponenten separat:

X-Komponente von a (a_x):

$$a_x = (2 \cdot -2) - (2 \cdot -1) + (1 \cdot -1) + (1 \cdot 3)$$

$$a_x = -4 - (-2) + (-1) + 3$$

$$a_x = -4 + 2 - 1 + 3$$

$$a_x = 0$$

Y-Komponente von a (a_y):

$$a_y = (2 \cdot 0) - (2 \cdot 3) + (1 \cdot 0) + (1 \cdot 0)$$

$$a_y = 0 - 6 + 0 + 0$$

$$a_y = -6$$

Zusammengesetzt ergibt sich der Koeffizientenvektor:

$$a = \begin{pmatrix} -2 \\ -6 \end{pmatrix}$$

Vergleich mit den Optionen:

Die berechnete Lösung $a = (-2, -6)^T$ entspricht Option (E).

Frage: (15) Welchen Wert hat der Koeffizientenvektor b in der parametrische Form des ersten Hermite Spline Segmente S_1 zwischen P_0 und P_1 ? (2 Punkte, SC)

- | | |
|--------------------|--------------------|
| (A) $b = (2, 6)^T$ | (E) $b = (8, 9)^T$ |
| (B) $b = (2, 9)^T$ | (F) $b = (0, 6)^T$ |
| (C) $b = (5, 6)^T$ | (G) $b = (0, 9)^T$ |
| (D) $b = (5, 9)^T$ | (H) $b = (8, 6)^T$ |

Der Koeffizientenvektor b ist die zweite Zeile des Ergebnisses der Matrixmultiplikation. Das bedeutet, b ergibt sich aus der zweiten Zeile von M_H multipliziert mit dem Spaltenvektor der Kontrollpunkte und Tangenten:

$$b = -3 \cdot P_0 + 3 \cdot P_1 - 2 \cdot T_0 - 1 \cdot T_1$$

Setzen wir die gegebenen Vektoren ein:

$$b = -3 \cdot \begin{pmatrix} -2 \\ 0 \end{pmatrix} + 3 \cdot \begin{pmatrix} -1 \\ 3 \end{pmatrix} - 2 \cdot \begin{pmatrix} -1 \\ 0 \end{pmatrix} - 1 \cdot \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

Berechne die X- und Y-Komponenten separat:

X-Komponente von b (b_x):

$$\begin{aligned} b_x &= (-3 \cdot -2) + (3 \cdot -1) - (2 \cdot -1) - (1 \cdot 3) \\ b_x &= 6 + (-3) - (-2) - 3 \\ b_x &= 6 - 3 + 2 - 3 \\ b_x &= 2 \end{aligned}$$

Y-Komponente von b (b_y):

$$\begin{aligned} b_y &= (-3 \cdot 0) + (3 \cdot 3) - (2 \cdot 0) - (1 \cdot 0) \\ b_y &= 0 + 9 - 0 - 0 \\ b_y &= 9 \end{aligned}$$

Zusammengesetzt ergibt sich der Koeffizientenvektor:

$$b = \begin{pmatrix} 2 \\ 9 \end{pmatrix}$$

Vergleich mit den Optionen:

Die berechnete Lösung $b = (2, 9)^T$ entspricht Option (B).

Frage: (16) Welchen Wert hat der Koeffizientenvektor c in der parametrische Form des ersten Hermite Spline Segmentes S_1 zwischen P_0 und P_1 ? (2 Punkte, SC)

- | | |
|---------------------|---------------------|
| (A) $c = (3, 0)^T$ | (E) $c = (1, 3)^T$ |
| (B) $c = (-3, 3)^T$ | (F) $c = (1, 0)^T$ |
| (C) $c = (-1, 0)^T$ | (G) $c = (3, 3)^T$ |
| (D) $c = (-3, 0)^T$ | (H) $c = (-1, 3)^T$ |

Der Koeffizientenvektor c ist die dritte Zeile des Ergebnisses der Matrixmultiplikation. Das bedeutet, c ergibt sich aus der dritten Zeile von M_H multipliziert mit dem Spaltenvektor der Kontrollpunkte und Tangenten:

$$c = 0 \cdot P_0 + 0 \cdot P_1 + 1 \cdot T_0 + 0 \cdot T_1$$

Setzen wir die gegebenen Vektoren ein:

$$c = 0 \cdot \begin{pmatrix} -2 \\ 0 \end{pmatrix} + 0 \cdot \begin{pmatrix} -1 \\ 3 \end{pmatrix} + 1 \cdot \begin{pmatrix} -1 \\ 0 \end{pmatrix} + 0 \cdot \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

Berechne die X- und Y-Komponenten separat:

X-Komponente von c (c_x):

$$c_x = (0 \cdot -2) + (0 \cdot -1) + (1 \cdot -1) + (0 \cdot 3)$$

$$c_x = 0 + 0 - 1 + 0$$

$$c_x = -1$$

Y-Komponente von c (c_y):

$$c_y = (0 \cdot 0) + (0 \cdot 3) + (1 \cdot 0) + (0 \cdot 0)$$

$$c_y = 0 + 0 + 0 + 0$$

$$c_y = 0$$

Zusammengesetzt ergibt sich der Koeffizientenvektor:

$$c = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

Vergleich mit den Optionen:

Die berechnete Lösung $c = (-1, 0)^T$ entspricht Option (C).

Frage: (17) Welchen Wert hat der Koeffizientenvektor d in der parametrische Form des ersten Hermite Spline Segmentes S_1 zwischen P_0 und P_1 ? (2 Punkte, SC)

- | | |
|---------------------|---------------------|
| (A) $d = (1, 0)^T$ | (E) $d = (1, 3)^T$ |
| (B) $d = (2, 0)^T$ | (F) $d = (-2, 0)^T$ |
| (C) $d = (-2, 3)^T$ | (G) $d = (2, 3)^T$ |
| (D) $d = (-1, 3)^T$ | (H) $d = (-1, 0)^T$ |

Der Koeffizientenvektor d ist die vierte Zeile des Ergebnisses der Matrixmultiplikation. Das bedeutet, d ergibt sich aus der vierten Zeile von M_H multipliziert mit dem Spaltenvektor der

Kontrollpunkte und Tangenten:

$$d = 1 \cdot P_0 + 0 \cdot P_1 + 0 \cdot T_0 + 0 \cdot T_1$$

Setzen wir die gegebenen Vektoren ein:

$$d = 1 \cdot \begin{pmatrix} -2 \\ 0 \end{pmatrix} + 0 \cdot \begin{pmatrix} -1 \\ 3 \end{pmatrix} + 0 \cdot \begin{pmatrix} -1 \\ 0 \end{pmatrix} + 0 \cdot \begin{pmatrix} 3 \\ 0 \end{pmatrix}$$

Berechne die X- und Y-Komponenten separat:

X-Komponente von d (d_x):

$$d_x = (1 \cdot -2) + (0 \cdot -1) + (0 \cdot -1) + (0 \cdot 3)$$

$$d_x = -2 + 0 + 0 + 0$$

$$d_x = -2$$

Y-Komponente von d (d_y):

$$d_y = (1 \cdot 0) + (0 \cdot 3) + (0 \cdot 0) + (0 \cdot 0)$$

$$d_y = 0 + 0 + 0 + 0$$

$$d_y = 0$$

Zusammengesetzt ergibt sich der Koeffizientenvektor:

$$d = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

Vergleich mit den Optionen:

Die berechnete Lösung $d = (-2, 0)^T$ entspricht Option (F).

Frage: (18) Welche Eigenschaften besitzt die oben beschriebene und in Abbildung 3 dargestellte Hermit Spline Kurve? (3 Punkte, MC)

- | | |
|--------------------------------------|---------------------------------------|
| (A) Lokaler Einfluss der Stützpunkte | (D) C^2 -stetig |
| (B) Approximierend | (E) C^1 -stetig |
| (C) Interpolierend | (F) Globaler Einfluss der Stützpunkte |

Begründung:

1. Interpolierend vs. Approximierend (B und C):

- Eine Kurve ist **interpolierend**, wenn sie durch alle angegebenen Kontrollpunkte (Stützpunkte) verläuft.
- Eine Kurve ist **approximierend**, wenn sie sich den Kontrollpunkten annähert, aber nicht unbedingt durch sie hindurchgeht (z.B. Bézier-Kurven, die nur die Endpunkte interpolieren).
- Die Angabe für Hermite Splines besagt klar, dass der Pfad **durch** die Punkte P_0, P_1, P_2 geht. Die Definition des Hermite Splines ist explizit so, dass der Spline an den

Endpunkten des Segments genau die Positionen der Kontrollpunkte annimmt und die dort definierten Tangenten hat.

- Daher ist die Hermite Spline Kurve **interpolierend**. Dies macht Aussage (C) richtig und Aussage (B) falsch.

2. Lokaler vs. Globaler Einfluss der Stützpunkte (A und F):

- **Lokaler Einfluss:** Eine Änderung an einem Kontrollpunkt oder einer Tangente beeinflusst nur das Segment(e) in unmittelbarer Nähe dieses Punktes.
- **Globaler Einfluss:** Eine Änderung an einem Kontrollpunkt oder einer Tangente beeinflusst die gesamte Kurve.
- Hermite Splines sind segmentweise definiert. Jedes Segment S_i hängt nur von seinen beiden Endpunkten und deren Tangenten ab. Wenn Sie beispielsweise P_1 oder T_1 ändern, betrifft dies nur die Segmente S_1 und S_2 , nicht jedoch andere Segmente, die weiter entfernt liegen würden (wären es mehr als zwei Segmente). Dies ist ein Kennzeichen von Splines im Allgemeinen.
- Daher besitzt die Hermite Spline Kurve **lokalen Einfluss der Stützpunkte**. Dies macht Aussage (A) richtig und Aussage (F) falsch.

3. Stetigkeit (D und E):

- **C^0 -Stetigkeit (Positionsstetigkeit):** Die Kurve ist durchgehend und es gibt keine Sprünge. Das bedeutet, $P_S(1)$ des ersten Segments ist gleich $P_S(0)$ des nächsten Segments. Bei Hermite Splines ist dies per Definition gegeben, da P_1 (Endpunkt von S_1) der Startpunkt von S_2 ist.
- **C^1 -Stetigkeit (Tangentenstetigkeit / erste Ableitungsstetigkeit):** Die erste Ableitung (die Tangente) ist an den Verbindungsstellen der Segmente stetig. Das bedeutet, die Tangente am Ende des ersten Segments ist gleich der Tangente am Anfang des nächsten Segments. In unserer Angabe sind explizit Tangenten T_0, T_1, T_2 für die Punkte P_0, P_1, P_2 gegeben. An den Übergangspunkten wie P_1 wird T_1 für das Ende von S_1 und den Anfang von S_2 verwendet, was eine automatische C^1 -Stetigkeit am Übergangspunkt garantiert, sofern die Tangenten identisch sind (was hier der Fall ist, da ein einziger T_1 Wert gegeben ist).
- **C^2 -Stetigkeit (Krümmungsstetigkeit / zweite Ableitungsstetigkeit):** Die zweite Ableitung (die Krümmung) ist an den Verbindungsstellen der Segmente stetig. Dies ist bei einfachen Hermite Splines (die nur durch Endpunkte und Tangenten kontrolliert werden) **nicht** automatisch gegeben. Um C^2 -Stetigkeit zu gewährleisten, müssten zusätzliche Bedingungen an die Tangenten gestellt oder ein komplexeres Spline-Modell (z.B. Cubic B-Splines oder Catmull-Rom Splines, die C^2 -Stetigkeit durch Mittelung benachbarter Punkte implizieren) verwendet werden.
- Die Hermite Spline Kurve ist **C^1 -stetig**. Dies macht Aussage (E) richtig und Aussage (D) falsch.

Angabe: Das zweite Hermite Spline Segment S_2 durch die Punkte $P_1 = (-1, 3)^T$ und $P_2 = (2, -1)^T$ hat die parametrische Form:

$$P_{S_2}(t) = \begin{pmatrix} -6 \\ 8 \end{pmatrix} t^3 + \begin{pmatrix} 6 \\ -12 \end{pmatrix} t^2 + \begin{pmatrix} 3 \\ 0 \end{pmatrix} t + \begin{pmatrix} -1 \\ 3 \end{pmatrix}, 0 \leq t \leq 1$$

Um vier Kamerapositionen K_0, K_1, K_2 und K_3 zu bestimmen, tasten wir den Kurvenparameter t gleichmäßig an den Stellen $t = 0$ für K_0 , $t = 1/3$ für K_1 , $t = 2/3$ für K_2 , und $t = 1$ für K_3 ab.

Frage: (19) Was ist der exakte x-Wert der Kameraposition K_1 entlang des zweiten Segmentes S_2 ? **(2 Punkte, SC)**

- | | |
|----------------------|-----------------------|
| (A) $K_{1,x} = 2/5$ | (D) $K_{1,x} = 4/9$ |
| (B) $K_{1,x} = 9/20$ | (E) $K_{1,x} = 5/12$ |
| (C) $K_{1,x} = 1/2$ | (F) $K_{1,x} = 21/50$ |

Berechnung des x-Wertes für K_1 :

1. Extrahiere die x-Komponente der parametrischen Gleichung:

Die x-Komponente von $P_{S_2}(t)$ ist:

$$P_{S_2,x}(t) = -6t^3 + 6t^2 + 3t - 1$$

2. Setze $t = 1/3$ in die x-Komponente ein:

$$P_{S_2,x}(1/3) = -6 \cdot \left(\frac{1}{3}\right)^3 + 6 \cdot \left(\frac{1}{3}\right)^2 + 3 \cdot \left(\frac{1}{3}\right) - 1$$

3. Berechne die Potenzen von $1/3$:

- $\left(\frac{1}{3}\right)^3 = \frac{1^3}{3^3} = \frac{1}{27}$
- $\left(\frac{1}{3}\right)^2 = \frac{1^2}{3^2} = \frac{1}{9}$

4. Setze die berechneten Potenzen in die Gleichung ein:

$$P_{S_2,x}(1/3) = -6 \cdot \left(\frac{1}{27}\right) + 6 \cdot \left(\frac{1}{9}\right) + 3 \cdot \left(\frac{1}{3}\right) - 1$$

5. Multipliziere die Terme:

$$P_{S_2,x}(1/3) = -\frac{6}{27} + \frac{6}{9} + \frac{3}{3} - 1$$

6. Vereinfache die Brüche:

- $-\frac{6}{27} = -\frac{2}{9}$ (durch 3 gekürzt)
- $\frac{6}{9} = \frac{2}{3}$ (durch 3 gekürzt)
- $\frac{3}{3} = 1$

7. Setze die vereinfachten Brüche ein:

$$P_{S_2,x}(1/3) = -\frac{2}{9} + \frac{2}{3} + 1 - 1$$

8. Addiere/Subtrahiere die Terme:

$$P_{S_2,x}(1/3) = -\frac{2}{9} + \frac{2}{3}$$

Um die Brüche zu addieren, bringen wir sie auf einen gemeinsamen Nenner (9):

$$P_{S_2,x}(1/3) = -\frac{2}{9} + \frac{2 \cdot 3}{3 \cdot 3}$$

$$P_{S_2,x}(1/3) = -\frac{2}{9} + \frac{6}{9}$$

$$P_{S_2,x}(1/3) = \frac{-2 + 6}{9}$$

$$P_{S_2,x}(1/3) = \frac{4}{9}$$

Ergebnis:

Der exakte x-Wert der Kameraposition K_1 ist $4/9$.

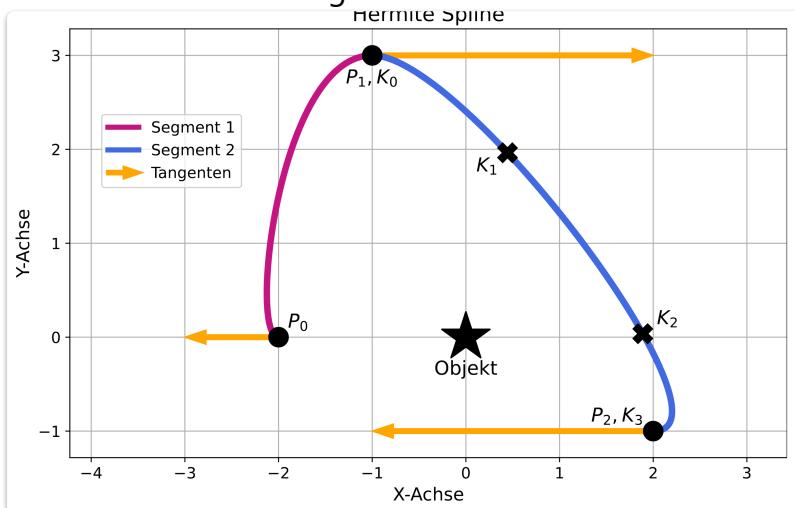
20 - 22

t	$s(t)$
0	A
1/3	1.8
2/3	B
1	5.3

Tabelle 1: Linear approximierte Distanz-Zeit-Funktion entlang des zweiten Segmentes

Angabe: Wie man in Abbildung 3 sehen kann, führt eine gleichmäßige Abtastung des Kurvenparameters t zu einer ungleichmäßigen Abtastung der zurückgelegten Strecke und somit zu einer unbeabsichtigten Geschwindigkeitsänderung der Kamera während der Animation. Nehmen Sie nun folgende, teils angenäherte Werte für die Kameraposition an: $K_0 = P_1 = (-1, 3)^T$, $K_1 = (0.5, 2)^T$, $K_2 = (2, 0)^T$ und $K_3 = P_2 = (2, -1)^T$. Die folgende Tabelle 1 enthält die Werte der Distanz-Zeit-Funktion $s(t)$ für die Kamerapositionen K entlang des zweiten Segmentes. Hierbei wird $s(t)$ über mehrere lineare Abschnitte, sprich der euklidischen Distanz zwischen den Kamerapositionen, approximiert.

hier nochmal Abbildung 3:



Frage: (20) Welchen Wert von $s(t)$ für $t = 0$ hat Lücke A in obiger Tabelle 1? (1 Punkt, SC)

- | | |
|--------------|----------|
| (A) -1 | (D) -1.8 |
| (B) 1.8 | (E) 0.5 |
| (C) 0 | (F) 1 |

Da die Tabelle laut Definition sich nur aufs 2. Segment bezieht, und wir uns im 2. Segment bei $t = 0$ beim Startpunkt befinden, haben wir hier noch nichts zurückgelegt $\Rightarrow s(0) = 0$

Frage: (21) Welchen Wert von $s(t)$ für $t = 2/3$ hat Lücke B in obiger Tabelle 1? (2 Punkte, SC)

- | | |
|----------------|---------|
| (A) 5.3 | (D) 1 |
| (B) 0 | (E) 1.8 |
| (C) 4.3 | (F) 2.5 |

1. Verständnis der Funktion $s(t)$:

Die Funktion $s(t)$ repräsentiert die akkumulierte Distanz entlang des **zweiten**

Kurvensegments. Diese Distanz wird durch die Summe der euklidischen Abstände zwischen den gegebenen Kamerapositionen (K_0, K_1, K_2, K_3) approximiert. Der Parameter t läuft von 0 bis 1 für dieses Segment.

2. Gegebene Kamerapositionen für das zweite Segment:

- $K_0 = P_1 = (-1, 3)^T$ (entspricht $t = 0$)
- $K_1 = (0.5, 2)^T$ (entspricht $t = 1/3$ laut Tabelle)
- $K_2 = (2, 0)^T$ (entspricht $t = 2/3$ laut Tabelle)
- $K_3 = P_2 = (2, -1)^T$ (entspricht $t = 1$)

3. Berechnung der euklidischen Distanzen zwischen den Punkten:

- **Abstand $d(K_0, K_1)$:**

$$\begin{aligned} d(K_0, K_1) &= \sqrt{(0.5 - (-1))^2 + (2 - 3)^2} \\ &= \sqrt{(1.5)^2 + (-1)^2} \\ &= \sqrt{2.25 + 1} = \sqrt{3.25} \approx 1.80277\dots \end{aligned}$$

Kontrolle mit Tabelle: Für $t = 1/3$ ist $s(t) = 1.8$, was dieser Distanz entspricht.

- **Abstand $d(K_1, K_2)$:**

$$\begin{aligned} d(K_1, K_2) &= \sqrt{(2 - 0.5)^2 + (0 - 2)^2} \\ &= \sqrt{(1.5)^2 + (-2)^2} \\ &= \sqrt{2.25 + 4} = \sqrt{6.25} = 2.5 \end{aligned}$$

- **Abstand $d(K_2, K_3)$:**

$$\begin{aligned} d(K_2, K_3) &= \sqrt{(2 - 2)^2 + (-1 - 0)^2} \\ &= \sqrt{(0)^2 + (-1)^2} \\ &= \sqrt{1} = 1 \end{aligned}$$

4. Bestimmung von $s(t)$ für $t = 2/3$:

Die Funktion $s(t)$ ist die kumulierte Distanz.

- $s(0) = 0$ (Startpunkt des Segments bei K_0)
- $s(1/3) = d(K_0, K_1) = 1.8$ (Distanz bis K_1)
- $s(2/3)$ ist die Distanz bis K_1 plus die Distanz von K_1 zu K_2 .

$$s(2/3) = s(1/3) + d(K_1, K_2)$$

$$s(2/3) = 1.8 + 2.5$$

$$s(2/3) = 4.3$$

Also rein theoretisch muss man sich bei dem Beispiel nur die Distanz zwischen 1/3 und 2/3 ausrechnen und zu dem Tabelleneintrag davor addieren.

Frage: (22) Welche Aussagen bezüglich des obigen Rechenbeispiels treffen zu? (3 Punkte, MC)

- (A) Würden wir bei der Distanz $s = 0.9$ anhand der Approximation der Distanz-Zeit-Funktion $s(t)$ in Tabelle 1 die Kurve abtasten, wäre der dazugehörige Kurvenparameter $t = 1/4$.
- (B) Um die Strecke entlang des zweiten Hermite Spline Segmentes gleichmäßig abzutasten würden wir das Intervall $[s(0), s(1)]$ in gleichmäßige Stücke unterteilen.
- (C) Würde man die Kameraposition nicht über eine Kurve festlegen, sondern physikalisch simulieren, sollte man am Besten das Explizite Euler Verfahren zur numerischen Integration der Zeit verwenden, da dieses Oszillationen vermeidet, weil es bedingungslos stabil und energiebewahrend ist.
- (D) Die Rotation der Kamera sollte am besten als Matrix abgespeichert werden, da Matrizen sich aufgrund des Gimbal Lock Phänomens besonders gut zur Darstellung von Rotationen eignen.
- (E) Quaternionen wären eine schlechte Darstellungsmöglichkeit der Rotation der Kamera, da die sphärische Interpolation zwischen ihnen aufwändig und instabil ist.
- (F) Das obige Rechenbeispiel verwendet die Methode des Keyframings, bei der Werte wie die Position zu bestimmten Zeitpunkten festgelegt werden, wobei es möglich ist, manche Werte (wie zum Beispiel die Farbe des betrachteten Objektes am Ursprung) auch nur am Anfang und Ende der Animation festzulegen.

A:

Wenn A so gewählt ist, dass $s(0) = 0$, dann:

$$s(t) = 5.4 \cdot t \quad \text{für } t \in [0, 1/3]$$

Dann wäre bei $s = 0.9$

$$t = \frac{0.9}{5.4} = \frac{1}{6} \neq \frac{1}{4}.$$

C:

Das explizite Euler-Verfahren ist nicht bedingungslos stabil und nicht energiebewahrend. Ganz im Gegenteil: Es ist instabil bei steifen Systemen und kann Oszillationen verstärken.

D:

Das ist falsch. Gimbal Lock ist ein Problem bei Matrizen. Sie sind nicht speziell wegen Gimbal Lock vorzuziehen.

E:

Ganz im Gegenteil: Quaternionen sind hervorragend für die Darstellung und Interpolation von Rotationen geeignet (insbesondere wegen slerp = „spherical linear interpolation“).

23

Frage: (23) Welche der folgenden Aussagen zum Latenten Raum treffen zu? (4 Punkte, MC)

- | | |
|--|---|
| (A) Er hat meist viele Dimensionen. | (C) Er ist definiert durch latente Variablen. |
| (B) Er ist leicht verständlich durch Variablenverschränkung. | (D) Er kann ohne Verluste in 2D dargestellt werden. |

24 - 27

Angabe: In Abbildung 4 ist ein Quadrat mit den vier Eckpunkten $\mathcal{Q} = \{(0,0), (0,4), (4,0), (4,4)\}$ gegeben. Die Werte der Punkte werden mittels der Funktion $f(x,y) = 2x^2 + y^2 - 2$ bestimmt. Beantworten Sie die folgenden Fragen.

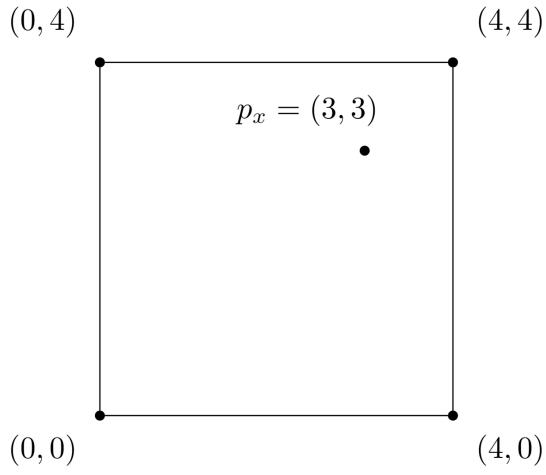


Abbildung 4: Quadrat

Frage: (24) Bestimmen Sie den Wert für den Punkt $(0, 4)$. (3 Punkte, SC)

- | | |
|--------|--------|
| (A) 0 | (D) 22 |
| (B) 12 | (E) 30 |
| (C) 14 | (F) 32 |

$$f(0, 4) = 2 \cdot 0^2 + 4^2 - 2 = 14$$

Frage: (25) Bestimmen Sie den Wert für den Punkt p_x . (3 Punkte, SC)

- | | |
|--------|--------|
| (A) 8 | (D) 25 |
| (B) 14 | (E) 32 |
| (C) 19 | (F) 35 |

$$f(p_x) = f(3, 3) = 2 \cdot 3^2 + 3^2 - 2 = 18 + 7 = 25$$

Frage: (26) Bestimmen Sie den Wert für den Punkt p_x mit Hilfe der Nearest Neighbour Interpolation und den vier Eckpunkten des Quadrats \mathcal{Q} . (4 Punkte, SC)

- | | |
|--------|--------|
| (A) 14 | (D) 30 |
| (B) 18 | (E) 36 |
| (C) 25 | (F) 46 |

1. Identifiziere den nächsten Nachbarn:

Der Punkt $(4, 4)$ hat den kleinsten Abstand ($\sqrt{2}$) zu $p_x = (3, 3)$.

2. Weise den Wert des nächsten Nachbarn zu:

Der Wert des nächsten Nachbarn $(4, 4)$ ist

$$f(4, 4) = 2(4)^2 + (4)^2 - 2 = 46$$

Frage: (27) Bestimmen Sie den Wert für den Punkt p_x mit Hilfe der bilinearen Interpolation und den vier Eckpunkten des Quadrats \mathcal{Q} . (4 Punkte, SC)

- | | |
|--------|--------|
| (A) 18 | (D) 34 |
| (B) 25 | (E) 40 |
| (C) 32 | (F) 46 |

Die **Bilineare Interpolation** ist eine Methode, um den Wert einer Funktion an einem beliebigen Punkt innerhalb eines Rechtecks zu schätzen, wenn die Funktionswerte an den vier Eckpunkten des Rechtecks bekannt sind. Sie ist eine Erweiterung der linearen Interpolation auf zwei Dimensionen.

Funktionsweise der Bilinearen Interpolation

Man kann sich die bilineare Interpolation als eine zweistufige lineare Interpolation vorstellen:

1. **Lineare Interpolation entlang der X-Achse:** Zuerst interpoliert man linear entlang der X-Achse für die beiden Y-Koordinaten der Eckpunkte. Dies ergibt zwei Zwischenwerte auf der Höhe des zu interpolierenden Punktes, aber jeweils an der linken und rechten X-Achse des Quadrats.
2. **Lineare Interpolation entlang der Y-Achse:** Dann werden diese beiden Zwischenwerte linear entlang der Y-Achse interpoliert, um den endgültigen Wert am gewünschten Punkt zu erhalten.

Gegebene Daten

- **Funktion:** $f(x, y) = 2x^2 + y^2 - 2$
- **Vier Eckpunkte des Quadrats Q:**
 - $Q_1 = (0, 0)$
 - $Q_2 = (0, 4)$
 - $Q_3 = (4, 0)$
 - $Q_4 = (4, 4)$
- **Punkt zur Interpolation:** $p_x = (3, 3)$

Berechnung der Funktionswerte an den Eckpunkten

Zuerst berechnen wir die Werte der Funktion $f(x, y)$ an den vier Eckpunkten des Quadrats. Diese Werte dienen als unsere bekannten Stützstellen für die Interpolation.

- $f(0, 0) = 2(0)^2 + (0)^2 - 2 = 0 + 0 - 2 = -2$
- $f(0, 4) = 2(0)^2 + (4)^2 - 2 = 0 + 16 - 2 = 14$
- $f(4, 0) = 2(4)^2 + (0)^2 - 2 = 2(16) + 0 - 2 = 32 - 2 = 30$
- $f(4, 4) = 2(4)^2 + (4)^2 - 2 = 2(16) + 16 - 2 = 32 + 16 - 2 = 46$

Rechenweg für $p_x = (3, 3)$ mittels Bilinearer Interpolation

Seien die Eckpunkte $(x_0, y_0), (x_1, y_0), (x_0, y_1)$ und (x_1, y_1) mit $x_0 = 0, x_1 = 4, y_0 = 0, y_1 = 4$. Der zu interpolierende Punkt ist $(x, y) = (3, 3)$.

Die Formel für die bilineare Interpolation lautet:

$$f(x, y) \approx \frac{1}{(x_1 - x_0)(y_1 - y_0)} \begin{bmatrix} f(x_0, y_0)(x_1 - x)(y_1 - y) \\ +f(x_1, y_0)(x - x_0)(y_1 - y) \\ +f(x_0, y_1)(x_1 - x)(y - y_0) \\ +f(x_1, y_1)(x - x_0)(y - y_0) \end{bmatrix}$$

Setzen wir die Werte ein:

- $x_0 = 0, y_0 = 0$
- $x_1 = 4, y_1 = 4$
- $x = 3, y = 3$

Der Nenner ist $(x_1 - x_0)(y_1 - y_0) = (4 - 0)(4 - 0) = 4 \times 4 = 16$.

Nun berechnen wir jeden Term der Summe:

1. $f(x_0, y_0)(x_1 - x)(y_1 - y) = f(0, 0)(4 - 3)(4 - 3) = (-2)(1)(1) = -2$
2. $f(x_1, y_0)(x - x_0)(y_1 - y) = f(4, 0)(3 - 0)(4 - 3) = (30)(3)(1) = 90$
3. $f(x_0, y_1)(x_1 - x)(y - y_0) = f(0, 4)(4 - 3)(3 - 0) = (14)(1)(3) = 42$
4. $f(x_1, y_1)(x - x_0)(y - y_0) = f(4, 4)(3 - 0)(3 - 0) = (46)(3)(3) = 46 \times 9 = 414$

Addieren der Terme und Division durch den Nenner:

$$f(3, 3) \approx \frac{1}{16}(-2 + 90 + 42 + 414)f(3, 3) \approx \frac{1}{16}(544)f(3, 3) \approx 34$$

Alternativer Rechenweg (zweistufige lineare Interpolation)

Schritt 1: Lineare Interpolation entlang der X-Achse

Interpoliere für $x = 3$ bei $y = 0$ (zwischen $(0, 0)$ und $(4, 0)$):

$$f(3,0) = f(0,0) \frac{4-3}{4-0} + f(4,0) \frac{3-0}{4-0} f(3,0) = (-2) \frac{1}{4} + (30) \frac{3}{4} f(3,0) = -\frac{2}{4} + \frac{90}{4} = \frac{88}{4} = 22$$

Interpoliere für $x = 3$ bei $y = 4$ (zwischen $(0,4)$ und $(4,4)$):

$$f(3,4) = f(0,4) \frac{4-3}{4-0} + f(4,4) \frac{3-0}{4-0} f(3,4) = (14) \frac{1}{4} + (46) \frac{3}{4} f(3,4) = \frac{14}{4} + \frac{138}{4} = \frac{152}{4} = 38$$

Schritt 2: Lineare Interpolation entlang der Y-Achse

Interpoliere für $y = 3$ bei $x = 3$ (zwischen den soeben berechneten $f(3,0)$ und $f(3,4)$):

$$f(3,3) = f(3,0) \frac{4-3}{4-0} + f(3,4) \frac{3-0}{4-0} f(3,3) = (22) \frac{1}{4} + (38) \frac{3}{4} f(3,3) = \frac{22}{4} + \frac{114}{4} = \frac{136}{4} = 34$$

Beide Wege führen zum selben Ergebnis.

Antwort: Der Wert für den Punkt $p_x = (3,3)$ mittels bilinearer Interpolation ist **34**.

28 - 31

Angabe: In Abbildung 5 ist ein 5×5 großer Bildausschnitt dargestellt, auf den der Moravec Ecken-detektor angewendet werden soll.

	x	0	1	2	3	4
y		5	5	5	5	5
0		5	5	5	5	5
1		10	10	8	5	5
2		10	10	9	5	5
3		10	10	10	5	5
4		5	5	5	5	5

Abbildung 5: Bildausschnitt

Frage: (28) Berechnen Sie die Veränderung der Intensitäten für die markierte Stelle $(2,2)$ und die Verschiebung $(1,0)$. Verwenden Sie dazu eine Fenstergröße von 3×3 und die Summe der quadrierten Differenzen. (3 Punkte, SC)

- | | |
|--------|--------|
| (A) 32 | (D) 49 |
| (B) 36 | (E) 55 |
| (C) 42 | (F) 71 |

- Berechnung der Summe der quadrierten Differenzen (SSD):**

Die SSD wird berechnet als $\sum_{u,v} (I_1(u, v) - I_2(u, v))^2$ über alle Pixel im Fenster.

- $I_1(1, 1) - I_2(2, 1) = (10 - 8)^2 = 2^2 = 4$
- $I_1(2, 1) - I_2(3, 1) = (8 - 5)^2 = 3^2 = 9$
- $I_1(3, 1) - I_2(4, 1) = (5 - 5)^2 = 0^2 = 0$
- $I_1(1, 2) - I_2(2, 2) = (10 - 9)^2 = 1^2 = 1$
- $I_1(2, 2) - I_2(3, 2) = (9 - 5)^2 = 4^2 = 16$
- $I_1(3, 2) - I_2(4, 2) = (5 - 5)^2 = 0^2 = 0$
- $I_1(1, 3) - I_2(2, 3) = (10 - 10)^2 = 0^2 = 0$
- $I_1(2, 3) - I_2(3, 3) = (10 - 5)^2 = 5^2 = 25$
- $I_1(3, 3) - I_2(4, 3) = (5 - 5)^2 = 0^2 = 0$

Summe der quadrierten Differenzen:

$$SSD = 4 + 9 + 0 + 1 + 16 + 0 + 0 + 25 + 0 = 55$$

Frage: (29) Angenommen die Intensitäten für die markierte Stelle (2, 2) und die Verschiebungen (1, 0), (1, 1), (0, 1) und (-1, 1) waren $E(1, 0) = 50$, $E(1, 1) = 32$, $E(0, 1) = 64$, $E(-1, 1) = 24$. Bestimmen Sie aus den Veränderungen der Intensitäten den Interest Value. (3 Punkte, SC)

- | | |
|----------|--------|
| (A) 24 | (D) 50 |
| (B) 42.5 | (E) 0 |
| (C) 32 | (F) 64 |

Hier muss man dann einfach den kleinsten Wert nehmen...

Frage: (30) Berechnen Sie für die Stelle (2, 2) aus Abbildung 5 das Ergebnis eines 3×3 -Medianfilters. (3 Punkte, SC)

- | | |
|-------|---------|
| (A) 7 | (D) 10 |
| (B) 8 | (E) 8.5 |
| (C) 9 | (F) 9.5 |

Einfach beim sortierten, das mittlere nehmen: 5, 5, 5, 8, 9, 10, 10, 10, 10

Frage: (31) Berechnen Sie für die Stelle (2, 2) aus Abbildung 5 das Ergebnis eines 3×3 -Mittelwertfilters, gerundet auf eine Nachkommastelle. (3 Punkte, SC)

- | | |
|---------|----------|
| (A) 4.8 | (D) 9.2 |
| (B) 6.0 | (E) 12.5 |
| (C) 8.0 | (F) 13.8 |

Alles zusammenaddieren und durch 9:

$$\frac{5 + 5 + 5 + 8 + 9 + 10 + 10 + 10 + 10}{9} = 8$$

32 - 34

Angabe: Die folgende Aufgabe ist vereinfacht zum Identifizieren von Keypoints anhand von SIFT. In Abbildung 6 sind die berechneten Difference of Gaussians (DoGs) eines Bildes für die Skalen s_1 - s_4 einer Oktave dargestellt (*scale space*). Im genannten *scale space* ist genau ein Keypoint $k = (x_k, y_k, s_k)$ enthalten (Wertebereich 0-3). Keypoints sind gekennzeichnet durch Extrema (Minima und Maxima) im *scale space*. Extrema in der untersten und obersten Ebene werden nicht berücksichtigt. Geben Sie den Keypoint an. Nehmen Sie eine $3 \times 3 \times 3$ -Nachbarschaft an.

x	y	
0		
0		
1		
2	$s = 0$	

1	2	0	2
0	-2	0	1
1	1	0	2
1	1	1	1
$s = 1$			

2	3	0	2
1	-3	1	1
2	0	1	1
1	1	2	1
$s = 2$			

2	5	0	4
4	-2	8	6
0	8	-2	3
-1	1	2	2
$s = 3$			

Abbildung 6: DoGs für verschiedene Skalen s

Frage: (32) Bestimmen Sie x_k . (3 Punkte, SC)

- | | |
|-----------------------|----------------|
| (A) 0
(B) 1 | (C) 2
(D) 3 |
|-----------------------|----------------|

Frage: (33) Bestimmen Sie y_k . (3 Punkte, SC)

- | | |
|-----------------------|----------------|
| (A) 0
(B) 1 | (C) 2
(D) 3 |
|-----------------------|----------------|

Frage: (34) Bestimmen Sie s_k . (3 Punkte, SC)

- | | |
|----------------|-----------------------|
| (A) 0
(B) 1 | (C) 2
(D) 3 |
|----------------|-----------------------|

Vorgehensweise:

Wir müssen jeden Punkt in den Ebenen $s = 1$ und $s = 2$ überprüfen, um festzustellen, ob er ein lokales Extremum (Minimum oder Maximum) innerhalb seiner $3 \times 3 \times 3$ -Nachbarschaft ist. Die Nachbarschaft eines Punktes (x, y, s) umfasst alle 26 umliegenden Punkte:

- Die 8 direkten Nachbarn in der gleichen Ebene (s).
- Die 9 Punkte in der Ebene darunter ($s - 1$).
- Die 9 Punkte in der Ebene darüber ($s + 1$).

Die Aufgabenstellung besagt, dass "genau ein Keypoint" enthalten ist. Wir suchen also nach dem einzigen Punkt, der diese Bedingung erfüllt.

Ein auffälliger Kandidat für ein Extremum ist der Wert -3 in Skala $s = 2$ an den Koordinaten $(x = 1, y = 1)$, also $D(1, 1, 2) = -3$.

Überprüfung des Punktes $(x = 1, y = 1, s = 2)$ mit dem Wert -3 :

Wir müssen diesen Wert mit allen 26 seiner Nachbarn vergleichen. Ist er kleiner als alle seine Nachbarn, ist er ein lokales Minimum.

1. Nachbarn in Skala $s = 1$ (Ebene unter $D(1, 1, 2)$):

Die Werte der 3×3 -Nachbarschaft um $(1, 1)$ in Skala $s = 1$:

$$D(0, 0, 1) = 1, D(1, 0, 1) = 2, D(2, 0, 1) = 0$$

$$D(0, 1, 1) = 0, D(1, 1, 1) = -2, D(2, 1, 1) = 0$$

$$D(0, 2, 1) = 1, D(1, 2, 1) = 1, D(2, 2, 1) = 0$$

Alle diese Werte sind ≥ -2 , also ≥ -3 .

2. Nachbarn in Skala $s = 2$ (gleiche Ebene wie $D(1, 1, 2)$, ohne $D(1, 1, 2)$ selbst):

Die 8 Werte der 3×3 -Nachbarschaft um $(1, 1)$ in Skala $s = 2$:

$$D(0, 0, 2) = 2, D(1, 0, 2) = 3, D(2, 0, 2) = 0$$

$$D(0, 1, 2) = 1, D(2, 1, 2) = 1$$

$$D(0, 2, 2) = 2, D(1, 2, 2) = 0, D(2, 2, 2) = 1$$

Alle diese Werte sind ≥ 0 , also ≥ -3 .

3. Nachbarn in Skala $s = 3$ (Ebene über $D(1, 1, 2)$):

Die Werte der 3×3 -Nachbarschaft um $(1, 1)$ in Skala $s = 3$:

$$D(0, 0, 3) = 2, D(1, 0, 3) = 5, D(2, 0, 3) = 0$$

$$D(0, 1, 3) = 4, D(1, 1, 3) = -2, D(2, 1, 3) = 8$$

$$D(0, 2, 3) = 0, D(1, 2, 3) = 8, D(2, 2, 3) = -2$$

Alle diese Werte sind ≥ -2 , also ≥ -3 .

Fazit:

Da der Wert $D(1, 1, 2) = -3$ kleiner ist als alle seine 26 Nachbarn, ist er ein **lokales Minimum** und somit der gesuchte Keypoint.

35

Frage: (35) Wieviele Vergleiche bzw. Abfragen müssen für einen Punkt im *scale space* bei SIFT maximal durchgeführt werden, um ihn als Keypoint zu identifizieren? (3 Punkte, SC)

- | | |
|---------------------------|----------------------------|
| (A) 8
(B) 16
(C) 26 | (D) 27
(E) 28
(F) 48 |
|---------------------------|----------------------------|

Weil wir in dem Fall 26 Nachbaren haben ($9 + 8 + 9 = 26$)

36

Frage: (36) Kreuzen Sie die richtigen Antworten bezüglich dem SIFT Algorithmus an. (4 Punkte, MC)

- | | |
|---|--|
| (A) SIFT ist rotationsinvariant. | (D) SIFT ist skalierungsinvariant. |
| (B) SIFT eliminiert ungeeignete Keypoints mittels einem Eckendetektor. | (E) Ein Bild hat maximal 128 SIFT Keypoints. |
| (C) SIFT eliminiert ungeeignete Keypoints mittels der Hough-Transformation. | (F) SIFT Deskriptoren besitzen die Dimension 64. |

bin mir nicht sicher ob B stimmt

C - Falsch Hough Transformation ist für Linien und Figuren erkennung.

E - Nein kein Limit

F - nein 128

37 - 38

Frage: (37) Betrachten Sie ein Perzeptron, das mit einer Signum-Aktivierungsfunktion

$$\text{sgn}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0, \\ -1 & \text{if } x < 0 \end{cases}$$

dem Gewichtungsvektor $\mathbf{w} = [0.1 \ -0.5 \ -0.1]^T$ und einem Bias $\theta = 0$ dargestellt wird. Wenn der Eingangsvektor zum Perzeptron $\mathbf{x} = [2 \ 1 \ -1]^T$ ist, berechnen Sie den Ausgang y des Perzeptrons. (2 Punkte, SC)

- | | |
|----------|----------|
| (A) 0.2 | (E) -0.3 |
| (B) 0.3 | (F) 1 |
| (C) -0.2 | (G) 0.5 |
| (D) -1 | (H) 0 |

Ein Perzeptron berechnet seinen Ausgang in zwei Schritten:

1. Berechnung der gewichteten Summe (Netzeingabe) z :

Die Netzeingabe z wird durch das Skalarprodukt des Gewichtsvektors \mathbf{w} und des Eingangsvektors \mathbf{x} , addiert mit dem Bias b , ermittelt.

$$z = \mathbf{w}^T \mathbf{x} + b$$

Gegeben sind:

$$\mathbf{w} = [0.1 \ -0.5 \ -0.1]^T$$

$$\mathbf{x} = [2 \ 1 \ -1]^T$$

$$b = 0$$

Setzen wir die Werte ein:

$$\begin{aligned} z &= (0.1 \times 2) + (-0.5 \times 1) + (-0.1 \times -1) + 0 \\ &= 0.2 - 0.5 + 0.1 + 0 \\ &= -0.3 + 0.1 \\ &= -0.2 \end{aligned}$$

2. Anwendung der Aktivierungsfunktion auf z , um den Ausgang y zu erhalten:

Die gegebene Aktivierungsfunktion ist die Signum-Funktion:

$$y = \text{sgn}(z) = \begin{cases} 1 & \text{if } z > 0 \\ 0 & \text{if } z = 0 \\ -1 & \text{if } z < 0 \end{cases}$$

Wir haben $z = -0.2$. Da $-0.2 < 0$ ist, wenden wir die entsprechende Regel an:

$$y = \text{sgn}(-0.2) = -1$$

Frage: (38) Bestimmen Sie den Bias θ für das Perzeptron, sodass für $\mathbf{x} = [2 \ 1 \ -1]^T$ $y = 0$ gilt.
(2 Punkte, SC)

- | | |
|---|---|
| (A) 0.2
(B) 0.7
(C) -1
(D) 0.5 | (E) -0.2
(F) -0.5
(G) -0.6
(H) 1 |
|---|---|

Da $-0.2 + 0.2 = 0$

39

Frage: (39) Kreuzen Sie die richtigen Antworten bezüglich Deep Learning an. **(2 Punkte, MC)**

- | | |
|---|--|
| (A) Neuronale Netzwerke können durch den Backpropagation Algorithmus trainiert werden.
(B) Clustering ist eine überwachte (supervised) Methode bei Machine Learning. | (C) Deep Learning basiert auf klassischen Algorithmen zur Merkmalsextraktion, zum Beispiel SIFT.
(D) Autoencoder besitzen dieselbe Eingangs- und Ausgangsdimension. |
|---|--|

B - Clustering ist eine **unüberwachte (unsupervised)** Lernmethode im Machine Learning. Es versucht, Muster in unbeschrifteten Daten zu finden und Datenpunkte in Gruppen (Cluster) zu organisieren, ohne dass vorher bekannte Ausgaben existieren.

C - Ein wesentlicher Vorteil und Merkmal von Deep Learning (insbesondere Convolutional Neural Networks, CNNs) ist, dass es **automatisch Merkmale extrahiert** (Feature Learning) und sich nicht auf handgefertigte oder "klassische" Merkmalsextraktionsalgorithmen wie SIFT oder HOG verlassen muss.

40

Frage: (40) Kreuzen Sie die richtigen Aussagen bezüglich der Fourier- und Houghtransformation an. **(4 Punkte, MC)**

- (A) Die Houghtransformation kann Geraden inklusive deren Start- und Endpunkte in einem Bild erkennen.
- (B) Die Fouriertransformierte einer Gaußfunktion ist wieder eine Gaußfunktion.
- (C) Die detektierten Strukturen im Houghraum sind als Minima gekennzeichnet.
- (D) Die Fourier-Transformation kann verwendet werden, um zeitabhängige Signale in ihre Frequenzkomponenten zu zerlegen.
- (E) Die Fouriertransformation ist im Allgemeinen komplexwertig.
- (F) Der Houghraum besitzt die Dimension drei für Geraden, die mittels $y = kx + d$ parametrisiert werden.

A - Die klassische Houghtransformation erkennt **ideale unendliche Geraden**. Start- und Endpunkte erkennt sie **nicht** – dafür wären zusätzliche Verarbeitungsschritte nötig.

C - Im Houghraum entsprechen die **Maxima** (nicht Minima) den Parametern (z.B. (ρ, θ)) von häufig erkannten Geraden. Ein Maximum bedeutet: „viele Punkte stimmen für diese Gerade“

F - zweidimensional

41

Frage: (41) Kreuzen Sie die richtigen Antworten bezüglich Stereo Vision an. **(4 Punkte, MC)**

- (A) Die horizontale Differenz eines Punktes in zwei Bildern nennt man Disparität.
- (B) Die Disparität wird größer, je näher das Objekt zur Kamera ist.
- (C) Beim regionenbasiertem Matching können zum Beispiel SIFT Deskriptoren verwendet werden.
- (D) Mittels Stereo Vision kann aus zweidimensionalen Bildern die Tiefeninformation wie-dergewonnen werden.
- (E) Bei Structure-from-Motion muss die Kamerageometrie bekannt sein.
- (F) Beim merkmalbasiertem Matching werden Features von beiden Bildern miteinander verglichen.

C:

- **Regionenbasiertes Matching** nutzt **intensitätsbasierte Fenster** (z.B. SAD, SSD, NCC) und **vergleicht Pixelblöcke**.
- **SIFT** ist ein **merkmalbasiertes Verfahren**, nicht regionenbasiert.

E:

- Structure-from-Motion (**SfM**) berechnet sowohl die **Kamerapositionen** als auch die **3D-Struktur gleichzeitig** aus mehreren Bildern.
- Die Kamerageometrie (Extrinsik) **muss nicht im Voraus bekannt** sein – sie wird iterativ rekonstruiert.

42

Angabe: Gegeben ist die Transformationsmatrix

$$\mathbf{T} = \begin{pmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

für homogene Koordinaten im zweidimensionalen Raum.

Frage: (42) Kreuzen Sie die Transformationen an, die durch \mathbf{T} darstellbar sind. (4 Punkte, MC)

- (A) Translation
- (B) Spiegelung
- (C) Skalierung
- (D) Projektive Transformation
- (E) Rotation

A - **Nicht darstellbar**, da rechte Spalte (Translationsteil) = (0,0)

D - **Nein**, da kein perspektivischer Anteil in der letzten Zeile vorhanden ist