KIT Computergrafik, WS 15/16

Paul Jungeblut

11. Februar 2016

Inhaltsverzeichnis

1 Bilder, Farbe und Perzeption **1**
- 1.1 Transferfunktion . . . 1
  - 1.1.1 Quantisierung . . . 1
  - 1.1.2 γ-Korrektur . . . 2
- 1.2 α-Kanal . . . 2
- 1.3 Licht . . . 3
  - 1.3.1 Wahrnehmung von Licht . . . 3
- 1.4 Farbräume . . . 3
  - 1.4.1 Graßmannsche Gesetze . . . 3
  - 1.4.2 RGB-Farbraum . . . 4
  - 1.4.3 CMY(K)-Farbraum . . . 4
  - 1.4.4 HSV-Farbraum . . . 4
  - 1.4.5 CIE Color Matching Funktionen . . . 5
  - 1.4.6 Der XYZ-Farbraum . . . 5
  - 1.4.7 Der xyY-Farbraum . . . 5
- 1.5 Chromazitätsdiagramme . . . 6

2 Analytische Geometrie **7**
- 2.1 Vektoren und Punkte . . . 7
- 2.2 Parameterdarstellungen . . . 8
- 2.3 Koordinatensysteme . . . 8
- 2.4 Baryzentrische Koordinaten . . . 8

3 Ray Tracing **10**
- 3.1 Abtastung . . . 10
- 3.2 Lochkamera . . . 10
- 3.3 Whitted-Style Ray Tracing . . . 11
- 3.4 Ray Generation . . . 11
- 3.5 Ray Intersection . . . 11
  - 3.5.1 Kugelschnitt . . . 12
  - 3.5.2 Ebenenschnitt . . . 12
  - 3.5.3 Dreiecksschnitt . . . 13
- 3.6 Beleuchtungsberechnung . . . 13
  - 3.6.1 Bidirectional Reflectance Distribution Function - BRDF . . . 13
  - 3.6.2 Phong-Beleuchtungsmodell . . . 13
  - 3.6.3 Beleuchtung von Dreiecksnetzen . . . 14

3.7 Lichtquellen . . . 14
3.8 Schatten, Reflexion, Brechung . . . 15
3.8.1 Snellsches Brechungsgesetz . . . 15
3.9 Anti-Aliasing . . . 16
3.10 Distributed Ray Tracing . . . 16

4 Transformationen und homogene Koordinaten 17
4.1 Transformationsgruppen . . . 17
4.2 2D Transformationen . . . 17
4.2.1 Skalierung . . . 17
4.2.2 Scherung (Transvektion) . . . 17
4.2.3 Spiegelung . . . 18
4.2.4 Rotation . . . 18
4.3 3D Transformationen . . . 18
4.3.1 Rotation . . . 18
4.4 Inverse Transformationen . . . 19
4.5 Affine Abbildungen . . . 20
4.5.1 Homogene Koordinaten . . . 20
4.5.2 Affine Abbildungen mit homogenen Koordinaten . . . 20
4.6 3D Transformationen in homogenen Koordinaten . . . 21
4.7 Koordinatensysteme in der Computergrafik . . . 21
4.7.1 Wechsel zwischen Koordinatensystemen . . . 21
4.7.2 Kameratransformation . . . 22
4.8 Hierarchisches Modellieren . . . 22
4.9 Transformation von Normalen . . . 22

Dieses Skript ist inoffiziell zur Vorlesung Computergrafik am KIT im Wintersemester 2015/2016 entstanden. Es erhebt keinen Anspruch auf Vollständigkeit und Korrektheit.

Die Vorlesung wurde von Prof. Dr. Carsten Dachsbacher gehalten und das Skript orientiert sich stark an seinen Folien.

1 Bilder, Farbe und Perzeption

In der Computergrafik geht es um die Erzeugung und Manipulation von Bildern. Diese Bilder sind meiste 2D Arrays aus farbigen Pixeln. Der Speicher, in dem die Farbe mit drei Werten für rot, grün und blau gespeichert wird, heißt Framebuffer. Heutzutage sind 8 Bit pro Farbe in einem Framebuffer üblich. Steht weniger zur Verfügung, müssen fehlende Farben durch Anordnung verfügbarer Farben nachgebildet werden. Sind die Pixel ausreichend klein, werden so Mischfarben wahrgenommen.

1.1 Transferfunktion

Höhere RGB-Werte bedeuten eine hellere Farbe. Wie hell genau eine Farbe erscheint, wird durch eine Transferfunktion f beschrieben. Man kann die Transferfunktion für ein Graustufenbild oder per Farbkanal betrachten.

$$f : [0, N] \to [I_{min}, I_{max}]$$

Dabei bildet f vom Wert des Pixels auf eine Helligkeit zwischen der minimalen und maximalen Displayhelligkeit I_{min} und I_{max} ab. Sie hängt von den physikalischen Eigenschaften des Displays ab.

Die maximale Helligkeit I_{max} gibt an, wie hell ein Pixel sein kann. Bei LCDs beträgt sie meist weniger als 10% der Hintergrundbeleuchtung des Displays. Die minimale Helliglkeit I_{min} ist die Menge Licht, die für ein schwarzes Pixel emmitiert wird.

Neben dem vom Display emmitierten Licht, reflektiert auch noch Umgebungslicht mit Intensität k an der Oberfläche. Dieses hat einen großen Einfluss auf den Kontrast, der am Bildschirm wahrgenommen werden kann. Der Dynamikumfang

$$R_d := \frac{I_{max} + k}{I_{min} + k}$$

beschreibt den maximalen Kontrast des Displays.

Die Transferfunktion sollte so ausfallen, dass aufeinander folgende Pixelwerte keinen sichtbaren Helligkeitsunterschied haben. Ist diese Forderung nicht erfüllt, können Bänder auf glatten Bildbereichen erscheinen. Menschen können einen Helligkeitsunterschied von etwa 2% wahrnehmen. In dunklen Bereichen werden also kleinere Schritte der Transferfunktion benötigt.

1.1.1 Quantisierung

Die Transferfunktion kann verschieden quantisiert sein. Die verschiedenen Möglichkeiten unterscheiden sich dabei in der größe der Helligkeitsschritte zwischen aufeinander folgenden Farbwerten.

Bei einer linearen Quantisierung (gleich große Helligkeitsschritte), muss jeder Schritt kleiner als 2% von I_{min} betragen. Um Helligkeiten bis I_{max} darzustellen werden

$$\frac{I_{max} - I_{min}}{0.02 \cdot I_{min}}$$

Schritte benötigt. Bei LCDs mit Dynamikumfang 100:1 sind dies etwa 5000 Schritte. Dies würde 12-13 Bit pro Farbkanal erfordern. Vorteil der linearen Quantisierung ist jedoch die einfache Arithmetik mit Pixelwerten.

Alternativ könnte die Transferfunktion exponentiell quantisiert sein, mit genau 2% zwischen zwei Pixelwerten. Bei einer exponentiellen Quantisierung ist $0 \mapsto I_{min}$, $1 \mapsto 1.02 \cdot I_{min}$, $2 \mapsto 1.02^2 \cdot I_{min}$, usw. Da $\log_{10} 1.02 \approx \frac{1}{120}$, werden ca. 120 Schritte für eine Verzehnfachung der Helligkeit benötigt. In diesem Fall reichen also 8 Bit gerade aus, um die 240 Schritte zu ermöglichen, die ein LCD mit Dynamikumfang 100:1 bräuchte.

Als Approximation der exponentiallen Quantisierung wird in der Praxis häufig eine potenzfunktion basierte Quantisierung eingesetzt.

$$I(n) = \left(\frac{n}{N}\right)^{\gamma} \cdot I_{max}$$

Der Exponent γ muss in diesem Fall immer mit angegeben werden. Ist $\gamma = 1$ hat man eine lineare Quantisierung.

1.1.2 γ-Korrektur

In diesem Abschnitt gelten vereinfachend die Idealwerte $I_{min} = k = 0$ und $I_{max} = 1$. Mit insgesamt N Schritten ($N = 256$ bei 8 Bit) wird ein Pixelwert n auf die Intensität $I(n)$ abgebildet.

$$I(n) \propto \left(\frac{n}{N}\right)^{\gamma}$$

Der γ-Wert charakterisiert das Display. In der Computergrafik wird ein Pixelwert α aber üblicherweise in einem linearen Raum berechnet. Bei der Darstellung will man, dass ein doppelter Wert doppelte Helligkeit bedeutet. Pixelwerte werden daher *direkt vor der Darstellung* einer γ-Korrektur unterzogen. Damit $I(n) \propto \alpha$ ist, wird $\alpha \propto \alpha^{\frac{1}{\gamma}}$ verwendet. Diese Korrektur wird unabhängig für jede Primärfarbe durchgeführt

1.2 α-Kanal

Oft werden Bilder 32 Bit pro Pixel kodiert. Ein Beispiel ist das RGBA-Format, wo neben den Primärfarben rot, grün und blau zusätzlich 8 Bit für einen α-Kanal zur Verfügung stehen. Der α-Wert gibt die Opazität (Gegenteil von Transparenz) an. Die Verwendung eines Alphakanals erlaubt es, Details von der Geometrie in die Textur zu verlagern und so das Rendern der Szene zu beschleunigen.

1.3 Licht

Licht ist elektoromagnetische Strahlung. Eine elektormagnetische Welle hat eine Wellenlänge λ und eine Frequenz $\nu = \frac{c}{\lambda}$. Jede solche Wellenlänge repräsentiert eine Spektralfarbe. Sichtbares Licht hat Wellenlängen zwischen 380 nm-700 nm. Licht ist in der Regel zusammengesetzt aus vielen verschiedenen Wellenlängen, jede mit einer bestimmten Intensität. $P(\lambda)$ ist die Strahlungsleistung der Wellenlänge λ.

Das menschliche Auge kann die spektrale Zusammensetzung von Licht nicht erfassen. Es passt sich zudem den äußeren (physikalischen) Umständen an. Als Rezeptoren dienen Stäbchen und Zapfen. Die ca. 120 Millionen Stäbchen sind sehr lichtempfindlich und eignen sich für monochromatisches Nachtsehen. Mit ca. 6-7 Millionen Zapfen ist trichromatisches Tagsehen möglich. Es gibt drei Arten von Zapfen, die sich in ihrer Empfindlichkeit bezüglich verschiedener Lichtsprektren unterscheiden: S-Zapfen (7%) entsprechen dem blauen Licht, M-Zapfen (37%)dem (gelb-)grünen und L-Zapfen (56%) dem (orange-)roten Licht.

1.3.1 Wahrnehmung von Licht

Die Wahrnehmung von Licht erfolgt anhand des Tripels (s, m, l) mit

$$s = \int S(\lambda)P(\lambda)\ \mathrm{d}\lambda, \qquad m = \int M(\lambda)P(\lambda)\ \mathrm{d}\lambda, \qquad l = \int L(\lambda)P(\lambda)\ \mathrm{d}\lambda.$$

Daraus folgt, dass es unterschiedliche Spektren mit unterschiedlichen Wellenlängen und Intensitäten gibt, die zur gleichen Wahrnehmung führen. Diesen Effekt bezeichnet man als Metamerismus. Der Metamerismus ist von elementarer Bedeutung in der Computergrafik, denn so kann ein Monitor mit drei Primärfarben den gleichen Eindruck vermitteln wie ein komplexes Spektrum.

1.4 Farbräume

Grundsätzlich kann zwischen additiver und subtraktiver Farbmischung unterschieden werden.

Bei additiver Farbmischung sind Rot, Grün und Blau die drei Primärfarben. Die Farkombination entsteht durch *Addition* der Spektren.

Bei der subtraktiven Farbmischung sind die Primärfarben Cyan, Gelb und Magenta. Die Farbkombination entsteht durch *Multiplikation* der Spektren.

Ein Farbmodell ist ein mathematisches Modell, in dem Farben durch Wertetupel beschrieben werden (z. B. 3-Tupel bei RGB oder 4-Tupel bei CMYK). Ein Farbraum ist die Menge aller Farben, die mit einem bestimmten Modell beschrieben werden können. Die Tristimuluswerte beschreiben eine Farbe in einem bestimmten Farbraum eines Farbmodells. Ohne Angabe des Farbmodells sind diese Werte nichtssagend.

1.4.1 Graßmannsche Gesetze

- Farbe ist eine dreidimensionale Größe (z. B. rot/grün/blau oder Farbton/Sättigung/Helligkeit).

- Superpositionsprinzip: Die Intensität einer additiv gemischten Farbe entspricht der Summe der Intensitäten der Ausgangsfarben.
- Der Farbton einer additiven Mischfarbe hängt nur vom Farbeindruck der Ausgangsfarben ab und nicht von deren Spektren. Auf die spektrale Zusammensetzung kann nicht rückgeschlossen werden. Beim Addieren von Spektren können einzelne Spektren also durch Metamere ersetzt werden, ohne dass sich der Farbeindruck ändert.

1.4.2 RGB-Farbraum

Im RGB-Farbraum dienen Rot, Grün und Blau als Primärfarben. Die genaue Definition der Primärfarben häng vom jeweiligen RGB-Raum ab. Es handelt sich um einen dreidimensionalen Farbraum mit

$$C = rR + gG + bB \in [0,1]^3.$$

Die Koeffizienten r, g, b werden Tristimuluswerte genannt. Zur Bestimmung Helligkeit kann die Luminanzapproximation

$$Y = 0.3r + 0.59g + 0.11b$$

1.4.3 CMY(K)-Farbraum

Der CMK-Farbraum ist ein subtraktiver Farbraum mit den Primärfarben Cyan, Magenta und Gelb. Er ist dual zum RGB-Farbraum:

$$\begin{pmatrix} C \\ M \\ Y \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

verwendet werden. Beim Drucken wird oft noch Schwarz als vierte Primärfarbe verwendet. Man spricht dann vom CMYK-Farbraum.

$$K = \min\{C, M, Y\}, \qquad \begin{pmatrix} C' \\ M' \\ Y' \\ K \end{pmatrix} = \begin{pmatrix} C - K \\ M - K \\ Y - K \\ K \end{pmatrix}$$

1.4.4 HSV-Farbraum

Der HSV-Farbraum besteht aus Farbton (engl. hue), Sättigung (engl. saturation) und Helligkeit (engl. value). Er ist weder additiv noch subtraktiv aber recht intuitiv und findet deshalb oft Anwendung in Benutzerschnittstellen.

1.4.5 CIE Color Matching Funktionen

Eine Color Matching Funktion gibt an, wie die Primärfarben addiert werden müssen, um eine Spekralfarbe zu reprodizieren. Nicht jede wahrnehmbare Farbe lässt sich durch Addition dreier Primärfarben darstellen. In diesem Fall muss eine der Primärfarben zur Referenzfarbe addiert werden. Mit den restlichen Primärfarben kann die Farbe dann nachgebildet werden.

Zur Berechnung einer metameren Farbe im selben RGB-Farbraum, müssen die Trsitimuluswerte r, g, b der folgenden Color Matching Funktionen berechnet werden.

$$r = \int \bar{r}(\lambda)P(\lambda)\ \mathrm{d}\lambda, \qquad g = \int \bar{g}(\lambda)P(\lambda)\ \mathrm{d}\lambda, \qquad b = \int \bar{b}(\lambda)P(\lambda)\ \mathrm{d}\lambda$$

Dabei stellen $\bar{r}, \bar{g}, \bar{b}$ die Spektren der Primärfarben dar. Dabei können wegen oben genanntem Effekt jedoch negative Vergleichswerte entstehen. Die Konsequenz: Einige Spektralfarben sind nicht realisierbar. RGB ist also kein perfekter Farbraum, dafür jedoch realisierbar.

1.4.6 Der XYZ-Farbraum

Ziel des XYZ-Farbraums ist es, alle wahrnehmaren Farben beschreiben zu können. Es ist demanch ein Farbraum mit rein positive Color Matching Funktionen. Die Y-Komponente des XYZ-Farbraums entspricht der Luminanz. Die Konvertierung zum RGB-Farbraum ist eine lineare Abbildung.

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = M \cdot \begin{pmatrix} R \\ G \\ B \end{pmatrix}, \qquad \begin{pmatrix} R \\ G \\ B \end{pmatrix} = M^{-1} \cdot \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}, \qquad M = \begin{pmatrix} 0.49 & 0.31 & 0.20 \\ 0.18 & 0.81 & 0.01 \\ 0.00 & 0.01 & 0.99 \end{pmatrix}$$

1.4.7 Der xyY-Farbraum

Für alle $k > 0$ repräsentiert $k(X, Y, Z)$ die gleiche Farbe, nur mit unterschiedlicher Intensität. Von daher können die Werte auf die $X + Y + Z = 1$ Ebene normalisiert werden. Eine anschließende Projektion auf die XY-Ebene (z = 0 setzen) enthält nach wie vor alle Farbtöne und Sättigungen. Es gilt

$$x = \frac{X}{X+Y+Z}, \qquad y = \frac{Y}{X+Y+Z}, \qquad z = \frac{Z}{X+Y+Z} = 1 - x - y.$$

Im xyY-Farbraum wird so die Information in Helligkeit und Chromazität (Farbe) aufgeteilt. Es müssen die Werte x und y sowie die Helligkeit Y angegeben werden. Der Wert z kann wie oben gezeigt, durch x und y berechnet werden und muss nicht mit gespeichert werden. X und Z können dann wie folgt aus x, y und Y berechnet werden:

$$X = \frac{Y}{y}x, \qquad Z = \frac{Y}{y}(1 - x - y)$$

1.5 Chromazitätsdiagramme

Ein Chromazitätsdiagramm enthält alle sichtbaren Farben, dem Gamut der menschlichen Wahrnehmung. Der Weißpunkt ($x = y = z = \frac{1}{3}$) entspricht dabei in etwa dem Sonnenlicht. Die Spektralfarben befinden sich entlang der Randkurve und entsprechen dem monochromatischem Licht. Die Purpurlinie ist die Menge von gesättigten Farben, die ein Mensch wahrnehmen kann. Es handelt sich dabei aber nicht um Spektralfarben.

Farben auf der Strecke zwischen zwei Punkten können durch Addition der Farben an den Endpunkten der Strecke gebildet werden. Die reine Farbe C_p zu einer Farbe C findet man auf dem Rand durch Verlängern der Strecke vom Weißpunkt durch C. Die Komplementärfarbe C_c liegt auf der Linie durch den Weißpunkt auf dem gegenüberliegenden Rand.

Alle Farben innerhalb eines Dreiecks lassen sich durch Addition der Farben an den Eckpunkten des Dreiecks bilden. Die darstellbaren Farben eines Ausgabegeräts werden durch dessen Gamut beschrieben. Es sind alle Farben innerhalb des von den Primärfarben aufgespannten Dreiecks (bzw. Polygons).

Gamut-Mapping ist eine Abbildung zwischen zwei Gamuts mit dem Ziel Farbverschiebungen gering zu halten.

2 Analytische Geometrie

Dieses Kapitel gibt einen sehr groben Überlick über einige Konzepte aus der anlytischen Geometrie, die in der Computergrafik wichtig sind. Alles ist sehr informell und nur als Wiederholung bekannten Inhalts gedacht.

2.1 Vektoren und Punkte

Ein Vektor besteht aus einer Richtung und einer Länge. Vektoren werden genutzt, um Verschiebungen oder Richtungen anzugeben. Ein Ortsvektor ist der Vektor vom Ursprung zu einem Punkt P.

Vektoraddition und -skalierung funktioniert komponentenweise. Die Länge eines Vektors ergibt sich durch $|a| = \sqrt{\sum_{i=1}^{n} a_i}$. Vektoren der Länge 1 heißen Einheitsvektoren.

Definition 2.1 (Skalarprodukt)
Seien $a = (a_1 \dots a_n)^T$ *und* $b = (b_1 \dots b_n)^T$ *Vektoren.* $\langle a, b\rangle = a_1 b_1 + \dots + a_n b_n$ *ist das Skalarprodukt von* a *und* b.

Das Skalarprodukt kann auch als Matrixmultiplikation $a^T b$ aufgefasst werden. Es gelten die folgenden Eigenschaften:

(i) $\langle a, b\rangle = |a||b| \cdot \cos\phi$. Dabei ist ϕ der von a und b eingeschlossene Winkel.

(ii) $\langle a, a\rangle = a^2$

(iii) $\langle a, b\rangle = \langle b, a\rangle$ (Symmetrie)

(iv) $\langle \lambda a + b, c\rangle = \lambda\langle a, c\rangle + \langle b, c\rangle$ (Linearität)

Definition 2.2 (Kreuzprodukt)
Seien $a = (a_1\ a_2\ a_3)^T$ *und* $b = (b_1\ b_2\ b_3)^T$ *Vektoren.*

$$a \times b = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{pmatrix} = n|a||b| \sin\phi$$

heißt das Kreuzprodukt von a *und* b. $|n|$ *ist* 1 *und* n *steht senkrecht zu der von* a *und* b *aufgespannten Ebene.* ϕ *ist wieder der von* a *und* b *eingeschlossene Winkel.*

Für das Kreuzprodukt gelten die folgenden Identitäten:

(i) $\langle a, a \times b\rangle = \langle b, a \times b\rangle = 0$

(ii) $\langle a, b \times c\rangle = \langle a \times b, c\rangle$

(iii) $a \times (\lambda b + c) = \lambda(a \times b) + a \times c$

(iv) $a \times (b \times c) = \langle\langle a, c\rangle, b\rangle - \langle\langle a, b\rangle, c\rangle$

(v) $\langle a \times b, c \times d\rangle = \langle a, c\rangle\langle b, d\rangle - \langle b, c\rangle\langle a, d\rangle$

Beispiel 2.3 (Anwendung des Kreuzprodukts)
Seien ab, c die Eckpunkte eines Dreiecks und $n' = (b-a) \times (c-a)$. Dann ist $n = \frac{n'}{|n'|}$ die Oberflächennormale und der $\frac{1}{2}|n'|$ der orientierte Flächeninhalt des Dreiecks.

2.2 Parameterdarstellungen

Eine Gerade ist durch zwei Punkte $P \neq Q$ definiert.

$$g(t) = P + t \cdot (Q - P), \quad t \in \mathbb{R}$$

Wählt man t aus $\mathbb{R}_+$, erhält man eine Halbgerade von Punkt P durch den Punkt Q. Wird t dagegen auf $[0, 1]$ eingeschränkt, erhält man die Strecke zwischen P und Q.

Analog ist eine Ebene durch drei nicht kollineare Punkte P, Q, R definiert.

$$g(s, t) = P + s \cdot (Q - P) + t \cdot (R - P), \quad s, t \in \mathbb{R}$$

Die Vektoren $(Q - P)$ und $(R - P)$ spannen die Ebene auf. Wählt man s und t aus $[0, 1]$, erhält man das Parallelogramm zwischen $(Q - P)$ und $(R - P)$.

2.3 Koordinatensysteme

Ein n-dimensionales Koordinatensystem wird durch eine Menge von n linear unabhängigen Basisvektoren $B = \{b_1, \ldots, b_n\}$ definiert. Die Basisvektoren müssen nicht gleich lang sein. B ist orthogonal, wenn alle Basisvektoren senkrecht zueinander stehen. Haben zusätzlich alle Basisvektoren Länge 1, heißt B orthonormal. Die Einheitsvektoren $\{e_1, \ldots, e_n\}$ bilden eine solche Orthonormalbasis.

2.4 Baryzentrische Koordinaten

Definition 2.4
Seien $P_1, \ldots, P_k$ Punkte des $\mathbb{R}^n$ und $k \leq n + 1$. Wenn ein Punkt Q als Affinkombination $(\lambda_1 + \ldots \lambda_k = 1)$

$$Q = \lambda_1 P_1 + \ldots + \lambda_k P_k$$

geschrieben werden kann, so bezeichnet man $(\lambda_1, \ldots, \lambda_k)$ als baryzentrische Koordinaten von Q bezüglich $P_1, \ldots, P_k$.

Alle Konvexkombinationen $(\lambda_1 P_1 + \ldots + \lambda_k P_k)$ einer Punktmenge $P_1, \ldots, P_k$, bilden die konvexe Hülle der Punktmenge. Da ein Dreieck konvex ist, ist ein Punkt genau dann innerhalb des Dreiecks, wenn all seine baryzentrischen Koordinaten bezüglich der Eckpunkte des Dreiecks nicht negativ sind.

Beispiel 2.5
Sei P_1, P_2, P_3 ein Dreieck und Q ein Punkt. Dann gilt für die baryzentrischen Koordinaten $\lambda_1, \lambda_2, \lambda_3$:

$$\lambda_1 = \frac{\Delta(Q, P_2, P_3)}{\Delta(P_1, P_2, P_3)} \qquad \lambda_2 = \frac{\Delta(P_1, Q, P_3)}{\Delta(P_1, P_2, P_3)} \qquad \lambda_3 = \frac{\Delta(P_1, P_2, Q)}{\Delta(P_1, P_2, P_3)}$$

Dabei ist $\Delta(A, B, C)$ der orientierte Flächeninhalt des Dreiecks A, B, C.

3 Ray Tracing

Die Idee beim Ray Tracing ist es, für jeden Pixel, alle Objekte zu finden, die diesen Pixel beeinflussen. Anhand all dieser Objekte wird die Pixelfarbe bestimmt. Dazu wird vom Rückwärtslichttransport ausgegangen. Man startet an der Kamera und sucht alle Pfade, auf denen das Licht dort hin gelangt. Dabei wird angenommen, dass der Lichttransport den Gesetzen der geometrischen Optik folgt.

3.1 Abtastung

Ein Rasterbild ist eine äquidistante Abtastung eines Bildsignals. Das Bildsignal wird also vereinfachend als stückweise konstante Funktion aufgefasst. Die bringt Probleme wie Aliasing oder den Moiré-Effekt mit sich.

Satz 3.1 (Nyquist-Shannon-Abtasttheorem)
Ein kontinuierliches, bandbegrenztes Signal mit einer maximalen Frequenz f_{max} muss mit einer Frequenz echt größer als $2f_{max}$ abgetastet werden, damit aus dem diskreten Signal das Ursprungssignal exakt rekostruiert werden kann.

Ist die Abtastfrequenz zu gering, kommt es zu Aliasing. Ein möglicher Lösungsansatz ist eine Vorfilterung des Signals, bei der hohe Frequenzen entfernt werden. Dies ist jedoch im allgemeinen Fall nicht möglich. Eine andere Möglichkeit ist eine Überabtastung mit anschließender Filterung.

3.2 Lochkamera

Am einfachsten zur Bildsynthese ist das Modell der Lochkamera. Sie ist definiert durch die Position ihrer Öffnung und der Bildebene. Da keine Linse verwendet wird, hat sie unbeschränkte Schärfentiefe.

Eine virtuelle Kamera ist definiert durch ihre Position und Blickrichtung, sowie die Orientierung der Vertikalen Achse. Dazu die Breite und Höhe der Bildebene und ihr Abstand *vor* der Kamera.

Bei der Bildsynthese kann objektbasiert oder bildbasiert vorgegangen werden.

Beim objektbasierten Rendern werden für jedes Objekt alle Pixel bestimmt, die es überdeckt. Dann wird die Farbe dieser Pixel ermittelt.

Beim bildbasierten Rendern werden für jeden Pixel alle an dieser Stelle sichtbaren Objekte bestimmt. Daraus wird die Pixelfarbe ermittelt.

3.3 Whitted-Style Ray Tracing

Ray Tracing besteht aus drei Schritten, die in dieser Reihenfolge ausgeführt werden.

1. Ray Generation Für jeden Pixel wird ein Strahl von der Kamera durch diesen Pixel erzeugt.

2. Ray Intersection Für jeden Strahl wird das Objekt gefunden, das die Kamera an diesem Pixel sieht. Es ist das Objekt, das diesen Strahl schneidet und dessen Schnittpunkt am nächsten an der Kamera liegt.

3. Beleuchtungsberechnung Farbe und Schattierung dieses Objekts an dieser Stelle wird berechnet. Dazu können rekursiv weitere Strahlen erzeugt werden, um z. B. reflektierende Oberflächen darzustellen.

3.4 Ray Generation

Die virtuelle Kamera ist definiert durch ihr Projektionszentrum e (engl. eye) und einen up-Vektor mit $|up| = 1$. Sei z der Zielpunkt eines Strahls. Definiere dann

$$w = \frac{(e - z)}{|e - z|}, \qquad u = up \times w, \qquad v = w \times u.$$

Dabei ist w die negative Blickrichtung.

Die Bildebene ist gegeben durch ihren Abstand d zur Kamera, ihren linken und rechten Rand l und r sowie ihren oberen und unteren Rand b und t. Strahlen von e aus zu einem Punkt s auf der Bildebene sind nun beschrieben durch:

$$s = \lambda_1 u + \lambda_2 v - dw \qquad \lambda_1 \in [l, r] \quad \lambda_2 \in [b, t]$$

Typischerweise ist das Sichtfeld symmetrisch, es gilt also $l = -r$ und $t = -b$. Das Verhältnis aus der Breite zur Höhe des Bildschirms heißt Aspect Ratio.

3.5 Ray Intersection

Geometrische Objekte können auf drei verschiedene Arten beschrieben werden:

- Parameterdarstellung Einsetzen aller gültigen Parameterwerte liefert alle Punkte des Objekts.

- Explizite Darstellung Es ist eine Funktion gegeben, die an jeder Position beschreibt, ob das Objekt an dieser Position ist.

- Implizite Darstellung Alle Punkte des Objekts bilden die Lösungsmenge eines Systems von Gleichungen.

3.5.1 Kugelschnitt

Alle Punkte auf der Kugeloberfläche K haben Abstand r vom Mittelpunkt $c = (c_x, c_y, c_z)$. Die implizite Darstellung der Kugel ist

$$K = \{(x, y, z) \mid |(x - c_x, y - c_y, z - c_z)| = r\}.$$

Sei $r(t) = e + td$ ein mit $t \in \mathbb{R}_+$ parametriesierter Strahl. Für den Schnittpunkt aus Kugel und Strahl ergibt sich:

$$\begin{aligned} 0 &= |r(t) - c|^2 - r^2 \\ &= |e + td - c|^2 - r^2 \\ &= \langle e + td - c, e + td - c\rangle - r^2 \\ &= \underbrace{\langle e - c, e - c\rangle - r^2}_{c} + \underbrace{2\langle td, e - c\rangle}_{b \cdot t} + \underbrace{t^2\langle d, d\rangle}_{a \cdot t^2} \end{aligned}$$

Mit der Mitternachtsformel

$$t_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

lassen sich nun die Parameter t_1 und t_2 bestimmen.

$$D = b^2 - 4ac$$

heißt ist die Diskriminante. Ist $D < 0$, gibt es keinen Schnittpunkt. Ist sie gleich 0, gibt es genau einen Schnittpunkt bei $r(t_1) = r(t_2)$. Ist D positiv, gibt es bei $r(t_1)$ und $r(t_2)$ jeweils einen Schnittpunkt. Die Parameter t_1 und t_2 können kleiner als 0 sein. In diesem Fall liegt der Schnittpunkt hinter der Kamera und sollte nicht betrachtet werden.

3.5.2 Ebenenschnitt

Eine Ebene im $\mathbb{R}^3$ hat die implizite Darstellung

$$E = \{(x, y, z) \mid ax + by + cz + d = 0, \quad a, b, c, d \in \mathbb{R}, \quad a, b, c, \neq 0\}.$$

Mit zwei nicht kollinearen Vektoren in der Ebene lässt sich die Normale n berechnen. Sei $r(t) = e + td$ ein Strahl mit $|d| = 1$. Dazu sei $\langle x, n\rangle - d = 0$ mit $|n|$ die Ebene in Hesse-Normalform.

$$\begin{aligned} 0 &= \langle e + td, n\rangle - d \\ &= \langle e, n\rangle + t\langle d, n\rangle - d \end{aligned}$$

Damit folgt für den Parameter t:

$$t = \frac{d - \langle e, n\rangle}{\langle d, n\rangle}$$

Fall $\langle d, n\rangle = 0$ gilt, sind Strahl und Ebene parallel und es exitiert kein Schnittpunkt. Andernfalls schneiden sich Strahl und Ebene im Punkt $r(t)$. Wenn $t < 0$ ist, liegt der Schnittpunkt hinter der Kamera und sollte ignoriert werden.

3.5.3 Dreiecksschnitt

Um einen Schintt zwischen einem Strahl und einem Dreieck zu berechnen, muss zuerst ein Schnittpunkt des Strahl mit der vom Dreieck aufgespannten Ebene gefunden werden. Die Koordianten des Schnittpunktes können dann in baryzentrische Koordinaten überführt und auf Positivität überprüft werden.

3.6 Beleuchtungsberechnung

Beleuchtung ist essentiell für einen dreidimensionalen Eindruck. Ein wichtiger Teil der Beleuchtungsberechungen ist die Reflexion. Es gibt zwei Extreme. Bei der spekularen Reflexion wird das Licht nur anhand eines Strahls reflektiert, wobei Einfallswinkel gleich Ausfallswinkel gilt. Dagegen wird das Licht bei der diffusen/lambterschen Reflexion zu gleichen Teilen in alle Richtungen gestreut.

Im Folgenden wird nur Reflexion an der Oberfläche von Objekten behandelt.

3.6.1 Bidirectional Reflectance Distribution Function - BRDF

Eine BRDF (Bidirectional Reflectance Distribution Function) ist ein radiometrisches Konzept, um die Reflexion an einem Oberflächenpunkt zu beschreiben. Sie gibt das Verhältnis von ausgehendem zu einfallendem Licht an einem Oberflächenpunkt an. Um Materialien abzubilden, muss die BRDF erst aufwendig an diesem Material gemessen werden.

3.6.2 Phong-Beleuchtungsmodell

Das Phong-Beleuchtungsmodell ist ein phänomenologisches (also physikalische nicht korrektes) Modell zur Darstellung der Reflexion, anhand von drei Komponenten, die von den Materialparametern k_a, k_d und k_s sowie dem Phong-Exponenten n abhängen:

- Ambient Die indirekte Beleuchtung, also Licht, das von anderen Oberflächen reflektiert wird. Es ergibt sich der Anteil $I_a = k_a \cdot I_L$.
- Diffus Der Anteil der lambertschen Reflexion. Für den diffusen Anteil ergibt sich $I_d = k_d \cdot I_L \cdot \cos\theta = k_d \cdot I_L \cdot \langle N, L\rangle$. Dabei ist I_L die Intensität der Lichtquelle, N die normierte Normale am Oberflächenpunkt und L der normierte Vektor zur Lichtquelle.
- Spekular Spekulare Reflexion bzw. perfekte Spiegelung. Die spekulare Reflexion findet ausschließlich in Richtung R_L statt. Der Vektor R_L ist die Spiegelung des Vektors L zur Lichtquelle an der Oberflächennormalen N. Sind alle Vektoren normiert, ergibt sich $R_L = 2N \cdot \langle N, L\rangle - L$.

  Durch gerichtete Reflexion entstehen Glanzlichter. Die Stärke der Spiegelung fällt für von R_L verschiedene Richtungen stark ab. Der Abfall wird durch $\cos^n \alpha$ modelliert. Der spekulare Anteil ergibt sich damit zu $I_s = k_s \cdot I_L \cdot \cos^n \alpha = k_s \cdot I_L \cdot \langle R_L, V\rangle^n$.

Die Gesamtbeleuchtung ergibt sich durch

$$\begin{aligned} I &= I_a + I_d + I_s \\ &= k_a \cdot I_L + k_d \cdot I_L \cdot \langle N, L \rangle + k_s \cdot I_L \cdot \langle R_L, V \rangle^n. \end{aligned}$$

Die Reflexionskoeffizienten k_a, k_d und k_s sind theoretisch Wellenlängenabhängig und werden deshalb oft für drei Wellenlängen (rot, grün, blau) angegeben.

Diffuse Reflexionen haben meist die Farbe der Oberfläche. Spekulare Reflexionen haben meist die Farbe der Oberfläche, wenn es sich um Metalle handelt. Ansonsten oft die Farbe der Lichtquelle.

Bei der Berechnung der Beleuchtungen ist man nur an den Richtungen interessiert, für die die Skalarprodukte positiv sind.

Optional kann das Phong-Beleuchtungsmodell um einen Emmisionsterm ersetzt werden.

3.6.3 Beleuchtung von Dreiecksnetzen

Bei den Beleuchtungsberechnungen im Phong-Beleuchtungsmodell wird an mehreren Stellen die Oberflächennormale verwendet. Beim sog. Flat Shading wird dazu die Dreiecksnormale verwendet. Nachteil dabei ist, dass die Kanten des Dreiecksnetzes deutlich sichtbar werden. Die Illision einer glatten, gekürmmten Oberfläche erreicht man, indem man die Vertex-Normalen betrachtet und über die Dreicksfläche interpoliert werden. Die Vertex-Normale berechnet sich als gewichtetes Mittel der Normalen der angrenzenden Dreiecke. Die Normale eines Dreiecks kann mit dem Kreuzprodukt bestimmt werden. Die so erlangte Vertex-Normale muss normalisiert werden. Es darf nicht einfach stattdessen durch die Anzahl der angrenzenden Normalen geteilt werden.

Nun wird aber jede Kante weich gezeichnet. Ist der Winkel zwischen zwei benachbarten Dreicken groß, benötigt man mehrere Normalen. Aus diesem Grund speichert man bei Dreiecksnetzen in der Regel drei Eckpunkte und drei Normalen.

Die Interpolation der Normalen wird linear und komponentenweise anhand der baryzentrischen Koordinaten durchgeführt. Beleuchtungsberechnungen mit der interpolierten Normalen nennt man Phong-Shading.

3.7 Lichtquellen

Es gibt verschiedene Lichtquellen mit unterschiedlichen Eigenschaften. Eine Punktlichtquelle ist definiert durch ihre Position und Intensität. Die Intensität fällt mit dem Abstandsquadrat.

Paralleles Licht ist definiert durch seine Richtung und die Flussdichte. Sonnenlicht ist nahezu parallel.

Bei einem Strahler/Spotlight gibt es einen Lichtkegel. Die Intensität bei Winkel θ beträgt $\cos^n \theta$.

Bei Flächenlichtquellen strahlt eine Fläche. Reale Lichtquellen sind (fast) immer Flächenlichtquellen. Sie sind der Grund für weiche Schatten.

3.8 Schatten, Reflexion, Brechung

Betrachtet man ausschließlich die Schnittpunkte mit den Primärstrahlen, wird eine Oberfläche genau dann beleuchtet, wenn sie der Kamera zugewandt ist. Dabei wird vernachlässigt, dass andere Objekte der Szene Schatten werfen können. Aus diesem Grund werden vom Schnittpunkt aus Schattenstrahlen/Sekundärstrahlen zu allen Lichtquelles ausgesendet und getestet, ob diese ein andere Objekte schneiden. Wegen der begrenzten Genauigkeit von Fließkommazahlen, kann es sein, dass der Schattenstrahl das Objekt, an dem er startet, erneut schneidet. Bei konvexen Objekten reicht es zu testen, dass nicht das gleiche Objekt erneut geschnitten wird. Ansonsten kann der Strahl z. B. mit einem kleinen Abstand zum Schnittpunkt gestartet werden.

Neben den Schattenstrahlen gibt es auch noch Reflexionsstrahlen und Transmissionsstrahlen. Ein solcher kann genauso behandelt werden, wie ein Sichtstrahl. Durch rekursive Verfolgung von Mehrfachreflexion/-transmission, können weitere Sekundärstrahlen entstehen. Die für den Strahl berechnete Farbe wird mit einem weiteren Koeffizienten k_r/k_t in die Beleuchtung mit eingerechnet.

Die Rekursion kann dabei theoretisch beliebig tief gehen. Eine Möglichkeit ist eine feste Rekursionsgrenze. Alternativ, kann abgebrochen werden, wenn Beitrag zur Farbe vernachlässigbar klein wird.

Die Reflexion folgt dabei dem Prinzip Einfallswinkel gleich Ausfallswinkel. Bei der Brechung muss das Snellsche Gesetz benutzt werden, um die Richtung des Strahls zu ermitteln.

3.8.1 Snellsches Brechungsgesetz

Das Snellsche Brechungsgesetz beschreibt die Richtungsänderung einer Welle, beim Übergang von einem Medium in eines mit anderer Brechzahl. Da sich Licht in verschiedenen Medien unterschiedlich schnell bewegt, definiert sich die (reele) Brechzahl η als

$$\eta = \frac{c_0}{c_\eta}.$$

Dabei ist c_0 die Lichtgeschwindigkeit im Vakuum und c_η die Lichtgeschwindigkeit im betrachteten Medium. Die Brechzahl ist dabei in der Regel wellenlängenabhängig. Unter Dispersion versteht man die Abhöngigkeit der Brechzahl von unterschiedlichen Wellenlängen.

Satz 3.2 (Snellsches Brechungsgesetzt)
Sei η_i die Brechzahl das Eingangsmediums und η_t die Brechzahl des Mediums in das eingedrungen wird. θ_i ist der Einfallswinkel, θ_t der Winkel zum Lot im Medium.

$$\eta_i \cdot \sin\theta_i = \eta_t \cdot \sin\theta_t$$

Die Brechung findet dabei beim Übergang ins optisch dichtere Medium zum Lot hin statt. Der Fresnel-Effekt beschreibt die Verteilung der Strahldichte zwischen Reflexion

und Transmission. Der Transmissionsvektor T ergibt sich durch

$$T = -\frac{\eta_i}{\eta_t} I + \left(\frac{\eta_i}{\eta_t} \cdot \cos\theta_i - \sqrt{1 - \left(\frac{\eta_i}{\eta_t}\right)^2 \cdot (1 - \cos^2\theta_i)} \right) N.$$

3.9 Anti-Aliasing

Unter Anti-Aliasing versteht man, Aliasingeffektie zu reduzieren. Als Technik dafür bietet sich das Überabtasten (Supersampling). Beim uniformen Supersampling werden gleichmäßig mehrere Samples pro Pixel genommen. Dabei muss der Abstand zwischen den Samples immer gleich sein.

Beim adaptiven Supersampling unterteilt man einen Pixel in Samples, wenn die Differenz der Farbe zum Nachbarpixel zu groß ist. Dieses Verfahren kann rekursiv fortgesetzt werden. Adaptives Supersampling wird heute kaum noch verwendet.

Auch stochastisches Sampling ist möglich. Dabei werden mehrere Samples pro Pixel an zufälligen Stellen gemacht. Beim Stratified Sampling wird der Pixel zuerst in ein Gitter unterteilt, dann wird in jeder Zelle an einer zufälligen Stelle abgetastet.

Nicht-uniformes Abtasten veringert das Aliasing aber erhöht das Rauschen. Rauschen wird jedoch vom menschlichen Auge als weit weniger störend empfunden.

3.10 Distributed Ray Tracing

Beim Whitted-Style ray Tracing sehen die Bilder oft zu perfekt aus:

- Perfekte Spiegelung und Reflexion.
- Keine weichen Schattenkanten.
- Unendliche Schärfentiefe.
- Keine Bewegungsunschärfe.

Beim Distributed Ray Tracing werden n Schattenstrahlen zu zufälligen Punkten auf der Lichtquelle geschickt. Da reale Lichtquellen endliche Ausdehnung haben, führt dies zu weichen Schatten.

Für Bewegungunschärfe können die n Strahlen zu leicht unterschiedlichen Zeiten verschickt werden. Dies simuliert die Belichtungszeit bei realen Kameras.

Für Schärfentiefe müssen die n Strahlen durch eine simulerte Linse gebrochen werden.

4 Transformationen und homogene Koordinaten

4.1 Transformationsgruppen

- Euklidische Transformationen erhalten Abstände und Inhaltsgrößen (Flächeninhalt, Volumen). Zu ihnen zählen die Translation, die Identität sowie die Rotation.
- Ähnlichkeitsabbildungen erhalten Winkel. Alle euklidischen Transformationen sind Ähnlichkeitsabbildungen. Außerdem gehören isotrope Skalierungen in diese Klasse.
- Lineare Abbildungen enthalten alle Skalierungen, Rotationen, Spiegelungen und Scherungen. Sie sind additiv, $T(p+q) = T(p)+T(q)$, und homogen, $T(\lambda p) = \lambda T(p)$.
- Affine Abbildungen enthalten die linearen Abbildungen sowie Translationen. Parallele Linien bleiben bei affinen Abbildungen erhalten. Weiter sind affine Abbildungen teilverhältnistreu. Projektive Abbildungen enthalten alle affinen Abbildungen. Die einzige Forderung ist jedoch, dass Geraden auf Geraden abgebildet werden.

4.2 2D Transformationen

Transformationen lassen sich durch Matrizen darstellen. Dies erlaubt eine leichte Hintereinanderausführung. Dazu müssen nur die einzelnen Transformationsmatrizen multipliziert werden.

4.2.1 Skalierung

Eine Skalierung ändert Längen und Winkel (nur bei nicht isotropen Skalierungen, $s_x \neq s_y$).

$$\text{scale}(s_x, s_y) = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix}$$

4.2.2 Scherung (Transvektion)

Unter einer Scherung versteht man die Verschiebung entlang einer Achse. Flächeninhalte bleiben daher erhalten. Es kann entlang beliebiger Achsen geschert werden.

$$\text{shear}_x(s) = \begin{pmatrix} 1 & s \\ 0 & 1 \end{pmatrix} \qquad \text{shear}_y(s) = \begin{pmatrix} 1 & 0 \\ s & 1 \end{pmatrix}$$

4.2.3 Spiegelung

Spiegelungen an einer Koordinatenachse sind negative Skalierungen. Es kann entlang beliebiger Achsen gespiegelt werden.

$$\text{reflect}_x = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \text{reflect}_y = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

4.2.4 Rotation

Eine Rotation beschreibt eine Drehung um einen Winkel ϕ.

$$\text{rotate}(\phi) \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}$$

4.3 3D Transformationen

4.3.1 Rotation

Die folgenden drei Matrizen drehen um die x-, y- und z-Achse.

$$\mathrm{R}_x(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix} \qquad \mathrm{R}_y(\phi) = \begin{pmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{pmatrix}$$

$$\mathrm{R}_z(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Allgemein werden Rotationen durch orthogonale Matrizen beschrieben. Sie sind orientierungserhaltend und ihre Zeilen- bzw. Spaltenvektoren sind paarweise orthonormal. Eine quadratische, reele Matrix M ist orhtogonal, wenn $M^T \cdot M = M \cdot M^T = I$ gilt. Es gilt also $M^{-1} = M^T$.

Seien u, v und w die Basisvektoren eines orthonormalen Koordinatensystems. Die Rotationsmatrix

$$R_{uvw} = \begin{pmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ w_x & w_y & w_u \end{pmatrix}$$

bildet die Basisvektoren u, v und w auf die kartesischen Achsen ab. Umgekehrt bildet R^T_{uvw} die kartesischen Einheitsvektoren auf u, v und w ab.

Neben den orthogonalen Matrizen können Rotationen auch über sog. Euler-Rotationen berechnet werden. Dabei wird nacheinander um verschiedene Koordinatenachsen rotiert. Es gibt zum Beispiel die folgenden drei Möglichkeiten:

$$R_z \to R_x \to R_z \qquad R_z \to R_y \to R_z \qquad R_x \to R_y \to R_z$$

Sind die Rotationswinkel um die x-, y- und z-Achse ψ, θ und ϕ, ergibt sich die Rotationsmatrix

$$\begin{aligned} R &= R_z(\phi) \cdot R_y(\theta) \cdot R_x(\psi) \\ &= \begin{pmatrix} \cos\theta\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \cos\theta\sin\phi & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi \\ -\sin\theta & \sin\psi\cos\theta & \cos\psi\cos\theta \end{pmatrix}. \end{aligned}$$

Die Winkel ψ, θ und ϕ heißen Euler-Winkel und beschreiben zusammen mit Festlegung der Achsen und der Reihenfolge die Orientierung eines Objekts.

Oft will man um eine Achse d und einen Winkel ϕ rotieren. Dazu wird ein Orthonormalsystem mit d als eine der Achsen benötigt. Sei $d = (d_x, d_y, d_z)$ mit $|d| = 1$. Wähle nun z. B.

$$e = \frac{1}{\sqrt{d_y^2 + d_z^2}}(0, -d_z, dy) \qquad \text{und} \qquad f = d \times e.$$

Mit der Matrix M werden d, e und f auf x, y und z abgebildet.

$$M = \begin{pmatrix} d^T \\ e^T \\ f^T \end{pmatrix} = \begin{pmatrix} d_x & d_y & d_z \\ e_x & e_y & e_z \\ f_x & f_y & f_z \end{pmatrix}$$

Im Koordinatensystem (d, e, f) kann dann mit $R_x(\phi)$ rotiert werden. Mit Rücktransformation ergibt sich

$$R_{d,\phi} = M^{-1} \cdot R_x(\phi) \cdot M$$

4.4 Inverse Transformationen

Die inverse Matrix beschreibt auch die geometrisch inverse Transformation. Eine Matrix zu invertieren, ist im Allgemeinen kompliziert, sodass zusätzliche Matrixeigenschaften wenn immer möglich ausgenutzt werden sollten.

- $S^{-1}(x, y, z) = S(\frac{1}{x}, \frac{1}{y}, \frac{1}{z})$ Skalierungen lassen sich einfach umkehren, wenn kein Skalierungsfaktor 0 ist.
- $R^{-1}(\phi) = R(-\phi)$ Rotationen können umgekehrt werden, indem man den Winkel umdreht. Noch besser ist es, die Orthogonalität auszunutzen: $R^-1 = R^T$.
- $T^{-1}(x, y, z) = T(-x, -y, -z)$ Mit homogenen Koordinaten können auch Translationen als Matrix beschrieben werden.
- Bei zusammengesetzten Transformationen gilt: $(AB)^{-1} = B^{-1}A^{-1}$

Nicht alle Transformationen sind aber überhaupt invertierbar. Können obige Regeln nicht angewendet werden, muss numerisch invertiert werden. Dies ist vergleichsweise aufwendig.

4.5 Affine Abbildungen

Affine Abbildungen sind eine Kombination aus einer linearen Abbildung und einer Translation. Dabei werden Geraden auf Geraden abgebildet und Parallelen bleiben parallel. Sie sind teilverhältnistreu aber nicht winkelerhaltend.

4.5.1 Homogene Koordinaten

Der euklidische/affine Raum wird um sog. Fernpunkte erweitert. Parallele Geraden in einem affinen Raum schneiden sich nicht. In einer Projektion jedoch schon. Geraden besitzen aber eine Richtung, diese Richtung ergänzt man in homogenen Koordinaten.

Die Menge aller Geraden durch den Ursprung im $\mathbb{R}^3$ nennt man den reellen projektiven Raum $P(\mathbb{R}^3)$. Ein Vektor $v \in \mathbb{R}^3$ definiert eine Gerade mit Dimension 2.

Der $\mathbb{R}^2$ kann mit dem homogenen Koordinaten in den $\mathbb{R}^3$ eingebettet werden.

- Punkte:
$$(x, y)_{2D} \to (x, y, 1)_h \equiv \left\{ (x', y', w) \mid \left(\frac{x'}{w}, \frac{y'}{w} \right) = (x, y) \right\}$$
- Richtungen:
$$(x, y)_{2D} \to (x, y, 0)_h$$

Der $\mathbb{R}^3$ enthält also alle Punkte und Richtungen des $\mathbb{R}^2$ ohne zwischen diesen zu unterscheiden.

4.5.2 Affine Abbildungen mit homogenen Koordinaten

Bisher haben wir affine Abbildungen wie folgt betrachtet:

$$x \mapsto Ax + b = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

Wenn $b_1 \neq 0$ oder $_2 \neq 0$, ist sie also nicht linear. Mit homogenen Koordinaten kann die selbe Abbildung wie folgt beschrieben werden:

$$\begin{aligned} x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} &\mapsto \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}_h \\ &\mapsto \begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}_h = \begin{pmatrix} a_{11}x_1 + a_{21}x_2 + b_1 \\ a_{21}x_1 + a_{22}x_2 + b_2 \\ 1 \end{pmatrix}_h \\ &\mapsto \begin{pmatrix} a_{11}x_1 + a_{21}x_2 + b_1 \\ a_{21}x_1 + a_{22}x_2 + b_2 \end{pmatrix} = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 \\ a_{21}x_1 + a_{22}x_2 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = Ax + b \end{aligned}$$

Beispiel 4.1
Rotation in 2D um die z-Achse durch den Punkt $c = (c_x, c_y)$ um Winkel ϕ

$$\begin{pmatrix} 1 & 0 & c_x \\ 0 & 1 & c_y \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & -c_x \\ 0 & 1 & -c_y \\ 0 & 0 & 1 \end{pmatrix}$$

4.6 3D Transformationen in homogenen Koordinaten

$$\text{scale}(s_x, s_y, s_z) = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{shear}_z(d_x, d_y) = \begin{pmatrix} 1 & 0 & d_x & 0 \\ 0 & 1 & d_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\text{rotate}_x(\phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{translate}(d_x, d_y, d_z) = \begin{pmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4.7 Koordinatensysteme in der Computergrafik

Objekte in einer Szene werden in ihrem eigenen Objekt- oder Modellkoordinatensystem beschrieben. Die Positionierung im Weltkoordinatensystem erfolgt dann durch Translation, Rotation und Skalierung. Danach folgt die transformation ins Kamerakoordinatensystem. Beim Ray Tracing geschieht dies implizit.

4.7.1 Wechsel zwischen Koordinatensystemen

Das globale Weltkordinatensystem hat Ursprung o und Basisvektoren x und y. Es ist das Bezugssystem für lokale Koordinatensysteme. Diese haben einen Ursprung e und Basisvektoren u und v. Für einen Punkt P gilt:

$$P = (p_x, p_y) = e + p_u u + p_v v$$

Für den Wechsel zwischen den Koordinatensystemen in homogenen Koordinaten gilt dann:

$$\begin{pmatrix} p_x \\ p_y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & e_x \\ 0 & 1 & e_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_x & v_x & 0 \\ u_y & v_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} p_u \\ p_v \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} p_u \\ p_u \\ 1 \end{pmatrix} = \begin{pmatrix} u_x & u_y & 0 \\ v_x & v_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -e_x \\ 0 & 1 & -e_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ 1 \end{pmatrix}$$

4.7.2 Kameratransformation

Die virtuelle Kamera ist definiert durch ihre Position e, die negative Blickrichtung w und den up-Vektor up. Dann ist $u = up \times w$ und $v = w \times u$. Die Vektoren u, v, w bilden die Basis des Kamerakoordinatensystems. Eine Transformation in dieses erleichtert projektive Abbildungen.

4.8 Hierarchisches Modellieren

Modelltransformationen sind üblicherweise zusammengesetzte Transformationen. Bei komplexen Szenen macht es die Modellierung einfacher, wenn mehrere Instanzen von Objekten zu Gruppen zusammengefasst werden und dann die Gruppen transformiert werden. Durch diese Zusammenfassung entsteht ein Szenengraph. Der Szenengraph ist gerichtet und azyklisch, aber kein Baum, da ein Objekt mehrfach in einer oder mehrerer Gruppen sein kann.

4.9 Transformation von Normalen

Die Oberflächennormale ist ein Einheitsvektor lokal-senkrecht zur Oberfläche. Sie ist wichtig für die Beleuchtungsberechnung. Die Normale kann nicht wie Objekte transformiert werden. Die liegt daran, dass lineare und affine Transformationen im Allgemeinen nicht winkeltreu sind.

Statt die Normale n zu transformieren, wird die Tangentenebene in homogenen Koordinaten transformiert. Der Trick ist, einen Punkt p auf der Tangentenebene E in p' zu transformieren und aus p' die transformierte Normale n' zu berechnen (muss senkrecht auf p' stehen). Seien dazu

$$
\begin{aligned}
E &= \{(x, y, z, d) \mid n_x x + n_y y + n_z z + d = 0\} && \text{die Tangentenebene in HNF,} \\
n &= (n_x, n_y, n_z, d) && \text{die Normale und} \\
p &= (x, y, z, 1) \in E && \text{ein Punkt in } E.
\end{aligned}
$$

Weiter sein M die Transformationsmatrix. Es gilt mit dem Skalarprodukt in homogenen Koordinaten:

$$p \in E \iff n^T p = 0$$

Der Punkt $p' = Mp$ liegt in der transformierten Tangentenebene und die transformierte Normale n' soll senkrecht darauf stehen.

$$
\begin{aligned}
0 &= n^T p \\
&= n^T (M^{-1} M) p \\
&= \underbrace{(n^T M^{-1})}_{n'^T} \cdot \underbrace{(Mp)}_{p'}
\end{aligned}
$$

Aus $n'^T = (n^T M^{-1})$ folgt dann:

$$n' = (n^T M^{-1})^T = (M^{-1})^T n$$

Beachte, dass nur die Richtung von n' korrekt ist, n' muss noch normiert werden. Zur Steigerung der Effizienz können die folgenden beiden Sonderfälle betrachtet werden:

- Bei Ähnlichkeitsabbildungen (euklidischen Transformationen und Skalierungen) ist $(M^{-1})^T = \lambda M$. In diesem Fall gilt also $n' = Mn$.
- Bei einer orthogonalen Matrix M gilt $(M^{-1})^T = M$. In diesem Fall gilt also ebenfalls $n' = Mn$.