Computer Graphics Zusammenfassung

Lucien Zürcher

January 1, 2019

Contents

1 Farbe

1.1 Was ist Farbe?

- **Physikalisch**, Lichtzusammensetzung, Elektromagnetischestrahlen
- **Physologisch**, Warnehmung und Interpretation

Farbe besteht aus:

- Farbton/Farbe
- Farbstich/Sättigung
- Helligkeit

1.2 Farbe eines Objektes

Ein Objekt nimmt Farbe auf und strahlt Farbe ab. Die Farbe des Objektes ist definiert durch die abgestrahlte Farbe.

- Beleuchtung (Illumination)
- Reflektion (Reflection)
- Farbsignal (Color Signal)

1.3 Licht besteht aus?

Licht besitzt verschiedene Wellenlängen, Kombinationen dieser Frequenzen ergeben eine Farbe.

- Sichtbares Licht (380mn - 780mn)
- Infrarot (780mn+)
- Ultraviolet (-380mn)

1nm = 10Å(Ångström)
1Å= øAtom

1.4 Das Auge

*Das Auge besteht aus; **Iris** (Muskel und Lichteinschränken), **Linse**, **Pupille** (Kontrolliert Iris) und **Retina** (Farb- und Lichtaufnahme am Rand des Auges)*

Die Retina besteht aus 75-100 10^6 Stäbchen (Lichtintensität) und 6-7 10^6 Zäpfchen (Farbe). Die Forea ist der dichteste Platz.

1.5 Wie sehen wir Farbe?

Durch die 3 Arten von Zäpfchen:

Kurz (S)		Mittel (M)		Lang (L)
Blau		Grün		Rot
440mn		530mn		560mn
1	:	5	:	10

1.6 Wahrnehmung

Grün 530mn wird am intensivsten wargenommen
Die Helligkeitswahrnehmung zwischen Stäbchen und Zäpfchen ist unterschiedlich

1.7 Farbsysteme

- **RGB** (Monitor, Spotligths,Pointilismus), additiv, C = (Rot, Grün, Blau)
- **CMY** (Drucken), subtraktiv, C = (Cyan, Magenta, Yellow)
- **CMYK**, CMY Mit Schwarz erweitert, K = min(Cyan, Magenta, Yellow) $C = C - K, M = M - K, Y = Y - K$
- **HSV**, Farbton (Hue) / Reinheit,Sättigung (Saturation) / Intensität (Value)
- **YUV** (Alte Fernseher, UV = 1/4 Auflösung Farbkorrektur)
  Y = $0.229 * R + 0.587G + 0.114 * B$,
  U = $0.436(B - Y)/(1 - 0.114)$,
  V = $0.615(R - Y)/(1 - 0.299)$
- **CIE-Lab**, absolutes Farbsystem
  Achsensystem mit Helligkeit als Y-Achse und X/Z-Achse definieren Farbunterschiede

1.8 Additives Farbsystem

Farben additeren (1,1,1) = Weiss, (0,0,0) = Schwarz

1.9 Subtraktives Farbsystem

Farben absorbieren (0,0,0) = Weiss, (1,1,1) = Schwarz

1.10 Farben Konvertieren

Zu Grau: $I = 0.229 * R + 0.587G + 0.114 * B$

RGB <> CMY: $\begin{pmatrix} C \\ M \\ Y \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} R \\ G \\ B \end{pmatrix}$

HSV <> RGB:

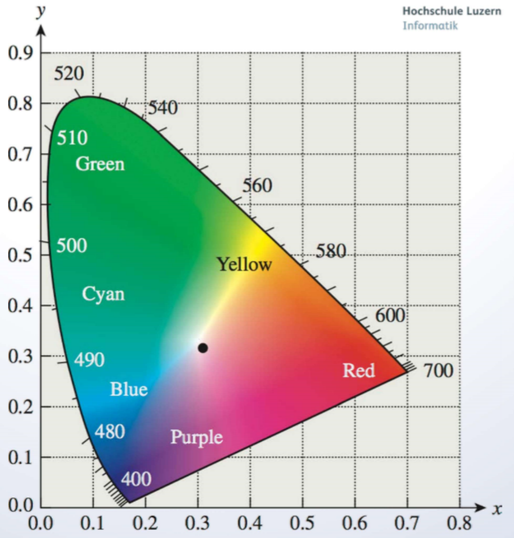
Farbe	H	S	V	R	G	B
Schwarz	–	–	0 %	0 %	0 %	0 %
Rot	0°	100 %	100 %	100 %	0 %	0 %
Gelb	60°	100 %	100 %	100 %	100 %	0 %
Braun	24,3°	75 %	36,1 %	36 %	20 %	9 %
Weiß	–	0 %	100 %	100 %	100 %	100 %
Grün	120°	100 %	100 %	0 %	100 %	0 %
Dunkelgrün	120°	100 %	50 %	0 %	50 %	0 %
Cyan	180°	100 %	100 %	0 %	100 %	100 %
Blau	240°	100 %	100 %	0 %	0 %	100 %
Magenta	300°	100 %	100 %	100 %	0 %	100 %
Orange	30°	100 %	100 %	100 %	50 %	0 %
Violett	270°	100 %	100 %	50 %	0 %	100 %

1.11 Gamma Korrektur

Erreichen von gleichmässiger Verteilung der Helligkeit / Kontrast. Das Empfinden der Helligkeit ist nicht linear.

Korrektur der Helligkeit des Bildes mit Gamme Wert. Wichtig für Bildschirme einstellen. Beim einstellen der Monitore Grauwerte mit echten Werten vergleichen (Gamma Test Pattern).

1.12 Normfarbtafel



1.13 Helligkeitswahrnehmung

Helligkeit wird logarithmisch wahrgenommen, Webers Law
$\frac{\Delta I}{I} = C$
$\log(I + \Delta I) - \log(I) = Const$

1.14 Nibs (Lichtdichte)

Gibt Helligkeitsdichte für Auge an. 10nits werden stärker wargenommen denn 100nits. Heisst, weniger Licht wird stärker wargenommen.

1.15 Mach bending

Optische Illusion, bei zwei verschiedenen Grauwerten nebeneinander unterschieden sich diese vermeitlich stärker.

1.16 Farbtäschung

Farbe wird abhängig durch Umgebung anderst wargenommen (Dunkler, Heller). Optische Illusionen

1.17 HD,UHD,UK

Unterscheiden sich durch Pixelauflösung.

1.18 Was ist HDR?

***High Dynamic Range**, speichert zusätzlichen Wert um Helligkeitsunterschiede besser unterschieden zu können (RGB-Pixelwerte propertianal zum Licht). Detailreichere dunkel und helle Spots, weniger Verlust durch Farben mit weniger Helligkeitsunterschiede.*

1.19 Begriffe

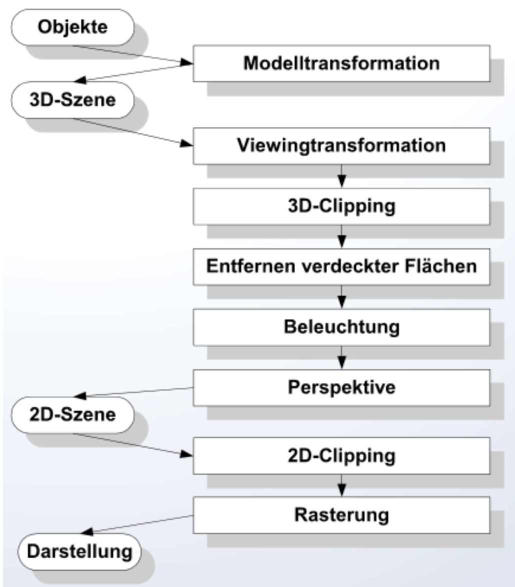
Natürliches Licht	Gemisch aus verschiedenen Lichtwellen / Frequenzen
Spektralfarben	reine Farbfrequenz; Alle Farben am Rand des CIE-Farbsystems
Spektrum	Alle Frequenzen und deren Verteilung
Spektralverteilung	Charakterisiert die Farbe, definiert durch Frequenzen (Bsp. Verschiedenes Weiss)
Komplementärfarben	Addieren ergeben Grau, gegenüberligende Farben im CIE-Farbsystem

2 WebGL

2.1 OpenGL Merkmale

- Low Level Graphics API
- Verschiedene Platformen
- 1.0/2.0 Fixe Funktionspipeline
- Vorlage für WebGL

2.2 Grafikpipeline



2.3 Programmierbare Shaders

Shaders werden für die Berechnung der zu zeichnenden Objekte verwendet. Das Programm wird direkt auf der Grafikkarte ausgeführt.

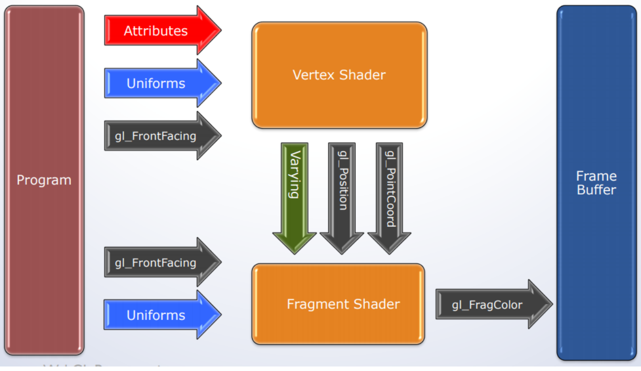
2.4 Vertex Processing

Berechnen der Positionen der Vertexe (Punkte) und Werte für den folgenden Fragmentshader.

2.5 Fragment Processing

Berechnet die Farbe der einzelnen Pixel.

2.6 Datenfluss



2.7 Attribut Variablen und Buffer definieren

Erzeugen

1. Buffer erzeugen (gl.createBuffer())
2. Array Buffer auf Buffer setzen (gl.bindBuffer(...))
3. Daten füllen (gl.BufferData(..))

Zeichnen

1. Buffer binden
2. Attribut und/oder uniform setzen (gl.vertexAttribPointer(..))
3. Attribut als Array setzen (gl.enableVertexAttribArray(..))
4. Zeichnen (gl.drawArrays(..))

3 Halbtontechnik

3.1 Verfahren der Halbtontechnik

Da nur Schwarz und Weiss gedruckt werden kann, werden die verschiedenen Stufen durch Intänsitätsstufen dargestellt. Dafür gibt es drei Verfahren:

- Quantisierung
- Dithering
- Error Diffusion

3.2 Quantisierung

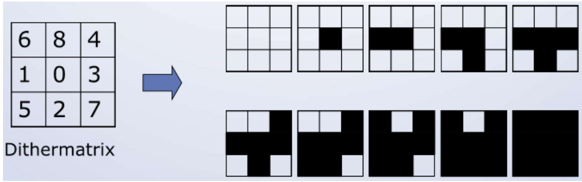
Höhere Auflösung auf tiefere Auflösung durch Runden der Pixelfarbwerte. Bsp. 16Bit -> 8Bit (Runden der Werte)

3.3 Dithering

Wenn der Drucker eine grössere Auflösung besitzt, jedoch weniger Farbstufen kann Dithering verfahren verwendet werden.

3.4 Dithermatrizen

Kann als Matrix dargestellt werden. Matrix gibt an, auf welcher stufe welche Pixel gesetzt werden



*Es gibt zwei Regeln; Gesetzter **Pixel bleibt gesetzt** und **Strukturen** in der Ditheringmatrix **vermeiden**. Es soll möglichst ein Kreis approximiert werden.*

3.5 Dithering bei gleich bleibender Auflösung

Handhabung, wenn die Auflösung gleichbleibt

- Mittelwert von n x n Region mit Ditheringmatrix ersetzen.
- Dispersed Dot Dithering

3.6 Dispersed Dot Dithering

Bayer Matrizen können hierfür verwendet werden, wodurch die Methode Bayer Dithering genannt wird.

2 x 2 Bayer Matrix

0	2
3	1

4 x 4 Bayer Matrix

0	8	2	10
12	4	14	6
3	11	1	9
15	7	13	5

$k = \frac{W_{max}}{n*n+1}$

W_{max}*: Maximalwert des Pixels (255 bei 8Bit)*
n*: Grösse der Matrix (2 x 2 =>* $n = 2$*)*
k*: Faktor für Umrechnung*

$I_{new} = \frac{I_{old}}{k}$
Für jeden Pixel den neuen Wert ausrechnen, danach mit Bayermatrix den Wert vergleichen. Pixel setzen wenn $I(x, y)_{new} > D_{ij}$*)*

$i = x$ modulo n
$j = y$ modulo n

3.7 Error Diffusion

Anstatt Kreise, Punkte verschiedener Dichte anordnen. Das Bild wird dabei sequenziell durchlaufen; links -> rechts, oben -> unten

Error Diffusion verteilt den Fehler auf die umliegenden Pixel

		7/16
1/16	5/16	3/16

Gewichtungsmatrix
Beispiel:

X	191	140	113
244	221	105	100

$191 - 255 = -64$, *da Pixel Schwarz (255), Fehler:* -64

X	X	140 + (7/16 * -64)	113
244 + (1/16 * -64)	221 + (5/16 * -64)	105 + (3/16 * -64)	100

Wenn Wert > 128 = 255, ansonten Wert <= 128 = 0

4 Vektoren

● *Skalarprodukt*
· *Matrixprodukt*

4.1 Addition

$$\vec{a} + \vec{b} = \begin{bmatrix} a_1 \\ a_2 \\ a_n \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_n \end{bmatrix} = \begin{bmatrix} a_1 + b_1 \\ a_2 + b_2 \\ a_n + b_n \end{bmatrix}$$

4.2 Multiplikation mit Skalar

$$\lambda\vec{a} = \lambda \begin{bmatrix} a_1 \\ a_2 \\ a_n \end{bmatrix} = \begin{bmatrix} \lambda a_1 \\ \lambda a_2 \\ \lambda a_n \end{bmatrix}$$

$\lambda \in$ *Skalar*

4.3 Skalarprodukt

$\vec{a} \bullet \vec{b} = |\vec{a}| \cdot |\vec{b}| \cdot \cos\pi$

4.4 Orthogonal

$\vec{e}_x \bullet \vec{e}_y = 0$
Senkrecht zueinander, wenn Skalarprodukt zweier Einheitsvektoren 0 ergibt.

4.5 Länge des Vektors

$||v|| = \sqrt{v \cdot v}$

4.6 Einheitsvektor

$e_v = \frac{1}{||v||} \bullet v$
$(i = e_1, j = e_2, k = e_3)$

$\vec{e}_x = [1, 0, 0]^T$
$\vec{e}_y = [0, 1, 0]^T$
$\vec{e}_z = [0, 0, 1]^T$

4.7 Euklidische Distanz

$\bar{AB} = \sqrt{(b_1 - a_1)^2 + (b_2 - a_2)^2 + \cdots + (b_n - a_n)^2}$

4.8 Achsenabschnitt

Gegeben sind 3 Punkte $p_x = x$*,*$p_y = y$*,*$p_z = z$ *ergibt Ebenegleichung:*

$\frac{x}{p_x} + \frac{y}{p_y} + \frac{z}{1} = 1$, HNF =

4.9 Hessische Normalform

TODO

4.10 Begriffe

Ortsvektor	Vom Ursprung zum Punkt
Richtungsvektor	Eine Richtung im Raum
Einheitsvektor	Eine Einheit in eine beliebige Richtung
Linearkombination	Ein Vektor, der ein vielfaches eines Einheitvektors ist
Skalar	Ist ein reelle oder komplexe Zahl
Rechtssystem	Koordinatensystem aufgebaut wie die rechte Hand wobei; der Zeigfinger X-Achse ($\vec{e}_x$), Mittelfinger Y-Achse ($\vec{e}_y$) und Daumen Z-Achse ($\vec{e}_z$)

5 Transformation

5.1 Transformation des Koordinatenystems

TODO

5.2 homogene Koordinaten

jeder Punkt P(x,y,z) des Raumes $\mathbb{R}^{\nVdash}$ *besitzt eine 4-komponenten Vektor* $\vec{r}$

$$\vec{r} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}, x = \frac{x_1}{x_4}, y = \frac{x_2}{x_4}, z = \frac{x_3}{x_4}$$

$$(x, y, z) = (\frac{x_1}{x_4}, \frac{x_2}{x_4}, \frac{x_3}{x_4})$$

5.3 Ebene im Raum

Ebene ϵ *im Raum* $\mathbb{R}^3$

$\epsilon : ax + by + cz + d = 0$ *Hessische Normalform*

$$\vec{w} = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}, \text{Punkt: } \vec{r} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Ebenengleichung:

$\vec{w} \bullet \vec{r} = w^T \cdot r = ax + by + cz + d = 0$

5.4 Prokektive Transformation

Die homogene Matrix $\mathbf{H}$ *ist nur bis auf einen konstanten Faktor bestimmt, heisst, alle Vielfachen von* $\mathbf{H}$ *sind auch gültig*

$\eta : \mathbb{P}^3 \mapsto \mathbb{P}^3$ stellt eine **projektiven Transformation** dar

$$\eta(r) = \mathbf{H} \cdot r = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Euklidisch (starre Bewegung)

$$D = \begin{bmatrix} \mathbf{R} & t \\ 0^T & 1 \end{bmatrix}$$

Abstand zwischen zwei Punkten, alle Winkel

$(R^{-1} = R^T)$

Ähnlichkeit

$$S = \begin{bmatrix} k \cdot \mathbf{M} & t \\ 0^T & 1 \end{bmatrix}$$

Winkel zwischen zwei Punkten, alle Winkel

Affin

$$A = \begin{bmatrix} \mathbf{C} & t \\ 0^T & 1 \end{bmatrix}$$

Parallelität, Verhältnis zwischen Volumeninhalt

Allgemein

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix}$$

Geraden bleiben Geraden

5.5 Euklidische Transformationen

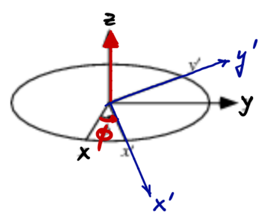
TODO Translation, Spiegelung an einer Ebene, Rotation, Zusammensetzen von

5.6 Rotation um beliebige Achse

1) Rotation um ϕ um z-Achse (Matrix D)
2) Rotation um den Winkel $\theta \in [0, \pi]$ (um frühere X-Achse) (Matrix C)
3) Eigentlich Rotation um den gegeben Winkel ψ (Matrix B)

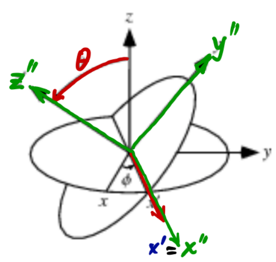
$c_\alpha = \cos\alpha$, $s_\alpha = \cos\alpha$, $\alpha \in \phi, \theta, \psi$

D

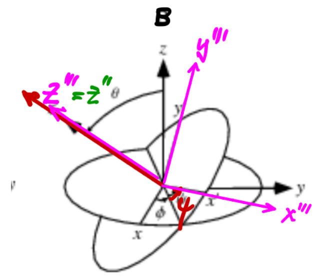


$$\mathbf{D} = \begin{bmatrix} c_\phi & s_\phi & 0 \\ -s_\phi & c_\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

C



$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\theta & s_\theta \\ 0 & -s_\theta & c_\theta \end{bmatrix}$$



$$\mathbf{D} = \begin{bmatrix} c_\psi & s_\psi & 0 \\ -s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Danach wieder zurück rotieren um ϕ und θ

5.7 Rotation um eine Achse durch den Ursprung

TODO insert T / $R_{y,x,z}$

Todo rotation around any axis

Todo altertative, rotation around origin

5.8 Parallele Projektion

Projektion auf Ebene $\epsilon : ax + by + cz + d = 0$

Die ebene ist definiert durch Normalvektor $\vec{n} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$

Normalenvektor erhalten: $|\vec{n}| = \sqrt{a^2 + b^2 + c^2} = 1$

Projektionsrichtung definiert durch $\vec{v} = (v_x, v_y, v_z)$
Normalisieren von Projektionsrichtung: $|\vec{v}|$

Ist $|\vec{n}|$ *(Ebenen Normalenvektor) und* $|\vec{v}|$ *(Projektionsrichtung) gegeben*

$\vec{x} = \vec{x}_0 + t\vec{v}$, komponentenweise $\begin{bmatrix} x = x_0 + tv_x \\ y = y_0 + tv_y \\ y = y_0 + tv_y \end{bmatrix}$
Wobei x_0 *Punkt wo auf* x *auf Ebene Projeziert wird*

ψ *entspricht Winkel zwischen* $\vec{n}$ *und* $\vec{v}$

cos$(\psi) = \vec{v} \bullet \vec{n}$
TODO - gleichung t t*

5.9 Parallele Projektionsmatrix

$$\begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} = \mathbf{H} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} =$$

$$\frac{1}{c_\psi} \begin{bmatrix} (c_\psi - av_x) & -bv_x & -cv_x & -dv_x \\ -av_y & (c_\psi - bv_y) & -cv_y & -dv_y \\ -av_z & -bv_z & (c_\psi - cv_z) & -dv_z \\ 0 & 0 & 0 & c_\psi \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix}$$

cos$(\psi) = c_\psi$

5.10 Perspektivische Projektion

Fall wenn Zentrum O im Nullpunkt

$\epsilon : ax + by + cz + d = 0$, Ebene

Beliebigen Punkt $A_0(x_0, y_0, z_0)$ *mit Projektionspunkt* $A^*(x^*, y^*, z*)$ *in Ebene* ϵ

$$\begin{bmatrix} x^* \\ y^* \\ z^* \end{bmatrix} = \begin{bmatrix} \lambda x_0 \\ \lambda y_0 \\ \lambda z_0 \end{bmatrix}$$
$\lambda = -\frac{d}{ax_0 + by_0 + cz_0}$

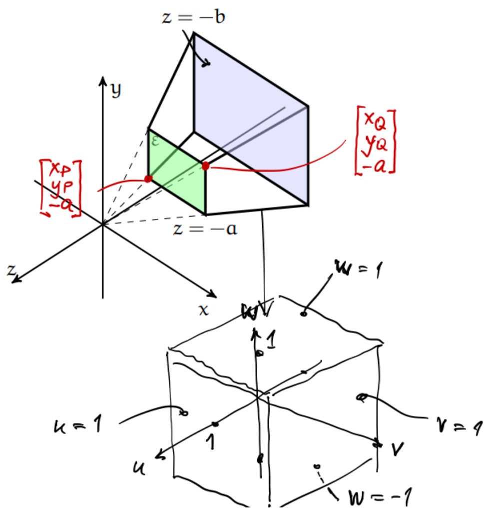
$$(ax_0 + by_0 + cz_0) \cdot \begin{bmatrix} x^* \\ y^* \\ z^* \\ 1 \end{bmatrix} = \begin{bmatrix} -dx_0 \\ -dy_0 \\ -dz_0 \\ ax_0 + by_0 + cz_0 \end{bmatrix} =$$
$$\begin{bmatrix} -d & 0 & 0 & 0 \\ 0 & -d & 0 & 0 \\ 0 & 0 & -d & 0 \\ a & b & c & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix}$$

5.11 Perspektivische Projektionmatrix

$$\mathbf{H} = \begin{bmatrix} -d & 0 & 0 & 0 \\ 0 & -d & 0 & 0 \\ 0 & 0 & -d & 0 \\ a & b & c & 0 \end{bmatrix}$$

5.12 Sichtvolumen Clipping

Das kanonische Sichtvolmen ist ein Würfel mit $P(\pm 1, \pm 1, \pm 1)$
Defür sind vorne und hinten, sowie zwei Punkte bestimmend Grösse gegeben



P links unten, Q rechts oben
z vorne $z = -a$, *z hinten* $z = -b$

$$\mathbf{T} = \begin{bmatrix} \frac{2a}{x_Q - x_P} & 0 & \frac{x_Q + x_P}{x_Q - x_P} & 0 \\ 0 & \frac{2a}{y_Q - y_P} & \frac{y_Q + y_P}{y_Q - y_P} & 0 \\ 0 & 0 & -\frac{b+a}{b-a} & -2\frac{ba}{b-a} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

6 Curves

6.1 Kurvie in der Ebene

Explizite Darstellung
$\gamma : [a, b] \to \mathbb{R}, x \mapsto y = f(x)$
Kreis:
oberer Halbkreis $\sqrt{r^2 - x^2}$
unterer Halbkreis $\sqrt{r^2 - x^2}$

Implizite Darstellung
$F(x, y) = 0$
Kreis: $x^2 + y^2 - r^2 = 0$

Parameterdarstellung
$\gamma : [a, b] \to \mathbb{R}^2, t \mapsto X(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$
Punkte miteinander verbunden, einzeln angegeben
Kreis: $\begin{bmatrix} r\cos t \\ r\sin t \end{bmatrix}$

6.2 Kurve im Raum

$$\gamma : [a, b] \to \mathbb{R}^3, t \mapsto X(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$$

6.3 Spirale entlang des Zylinders

$x^2 + y^2 = r^2$

$$\gamma : [0, 4\pi] \to \mathbb{R}^3, t \mapsto X(t) = \begin{bmatrix} r\cos t \\ rsint \\ ht/(2\pi) \end{bmatrix}$$

Grundriss ergibt Kreis, Höhe Linear

6.4 Methode unbestimmte Koeffizienten

$P_3(x) = c_0 + c_1 x^2 + c_2 x^2 + c_3 x^3$

$$\begin{bmatrix} 1 & x_0 & x_0^2 & x_0^3 \\ 1 & x_1 & x_1^2 & x_1^3 \\ 1 & x_2 & x_2^2 & x_2^3 \\ 1 & x_3 & x_3^2 & x_3^3 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

$c_0 = c_1 = c_2 = c_3 = 1$

6.5 Lagrange Methode

$l_0(x) = (x - x_1)(x - x_2)\ldots$
$L_0(x) = \frac{l_0(x)}{l_0(x_0)} = \frac{(x-x_1)(x-x_2)\ldots}{(x_0-x_1)(x_0-x_2)\ldots}$
$P_n(x) = y_0 L_0(x) + y_1 L_1(x) + \cdots + y_n L_n()$

$l_k(x) = \prod_{i=0 i\neq k}^{n}(x - x_i)$
$L_k(x) = \frac{l_k(x)}{l_k(x_k)}$

6.6 Lineare Bézier spline

$P(t) = (1 - t)P_0 + P_1 (0 \le t \le 1)$
Gewichteter Durchschnitt der Kontrollpunkte

$P(t) = (P_1 - P_0)t + P_0$
Polynom in t

$P(t) = [P_0, P_1] \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} t \\ 1 \end{bmatrix} (0 \le t \le 1)$
Matrizform

6.7 Quadric Bézier spline

drei Kontrollpunkte P_0, P_1, P_2

$P_0^1(t) = (1 - t)P_0 + P_1$
$P_1^1(t) = (1 - t)P_0 + P_1$

$P(t) = (1 - t)^2 P_0 + 2(1 - t)tP_1 + t^2 P_2$

6.8 Qubic Bézier Spline

vier Kontrollpunkte P_0, P_1, P_2, P_3

Mit P_0^1, P_1^1 *und*
$P_2^1(t) = (1 - t)P_2 + tP_3$

$P_1^2(t) = (1 - t)P_0^1(t) + tP_1^1(t)$
$P_2^2(t) = (1 - t)P_1^1(t) + tP_2^1(t)$

$P(t) = (1-t)^3 P_0 + 3(1-t)^2 tP_1 + 3(1-t)t^2 P_2 + t^3 P_3$

6.9 Bernsteinpolynome

7 Appendix

7.1 Radians

Winkel $\alpha°$	Bogenmass	Sinus	Kosinus
0°	0	$\frac{1}{2}\sqrt{0} = 0$	$\frac{1}{2}\sqrt{4} = 1$
30°	$\frac{\pi}{6}$	$\frac{1}{2}\sqrt{1} = \frac{1}{2}$	$\frac{1}{2}\sqrt{3}$
45°	$\frac{\pi}{4}$	$\frac{1}{2}\sqrt{2} = \frac{1}{\sqrt{2}}$	$\frac{1}{2}\sqrt{2} = \frac{1}{\sqrt{2}}$
60°	$\frac{\pi}{3}$	$\frac{1}{2}\sqrt{3}$	$\frac{1}{2}\sqrt{1} = \frac{1}{2}$
90°	$\frac{\pi}{2}$	$\frac{1}{2}\sqrt{4} = 1$	$\frac{1}{2}\sqrt{0} = 0$
180°	π	0	-1
270°	$\frac{3\pi}{2}$	-1	0
360°	2π	0	1

$\cos^2(\alpha) = \frac{1}{1+\tan^2(\alpha)}, \sin^2(\alpha) = \frac{\tan^2(\alpha)}{1+\tan^2(\alpha)}$