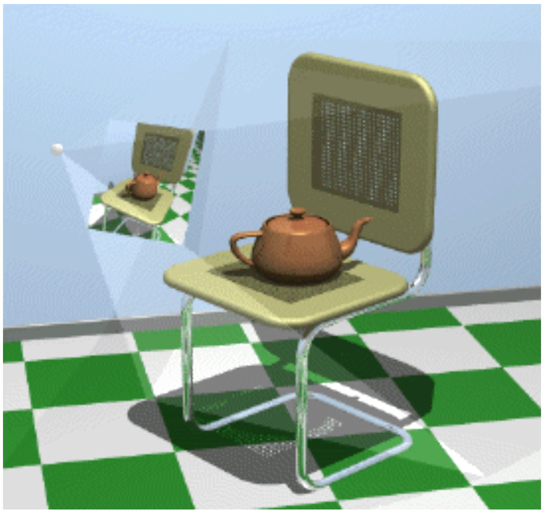
Computer-Graphik I
Transformationen & Viewing

$$\begin{bmatrix}\cos 90^\circ & \sin 90^\circ \\ -\sin 90^\circ & \cos 90^\circ\end{bmatrix}\begin{bmatrix}a_1\\a_2\end{bmatrix} =$$

G. Zachmann
Clausthal University, Germany
cg.in.tu-clausthal.de

Motivation

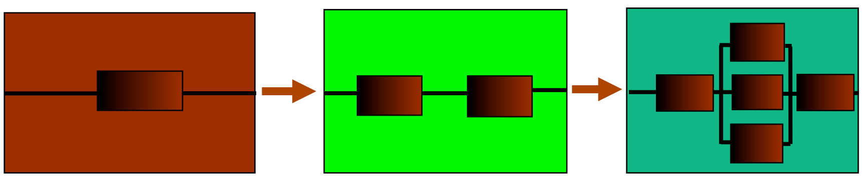
- Man möchte die virtuelle 3D Welt auf einem 2D Display darstellen



Motivation

- Transformationen werden benötigt, um ...
  - Objekte, Beleuchtung und Kamera zu positionieren und animieren;
  - alle Berechnungen im selben Koordinatensystem durchzuführen;
  - Objekte zu projizieren
- OpenGL verwendet 4x4-Matrizen zur Spezifikation von Transormationen
- Viewing = welche Transformationen muß man verwenden, um die 3D-Welt auf den 2D-Bildschirm zu projizieren

Die Graphik-Pipeline (stark vereinfacht)



Anwendung

Geometrie-Stufe

Raster-Stufe

Alle Berechnungen, die 1x pro Polygon oder pro Vertex (Ecke) durchgeführt werden
Z.B.:
- Modell- und Viewing-Transformation
- Projektion
- Beleuchtung
- Clipping

Arbeitet im 3D

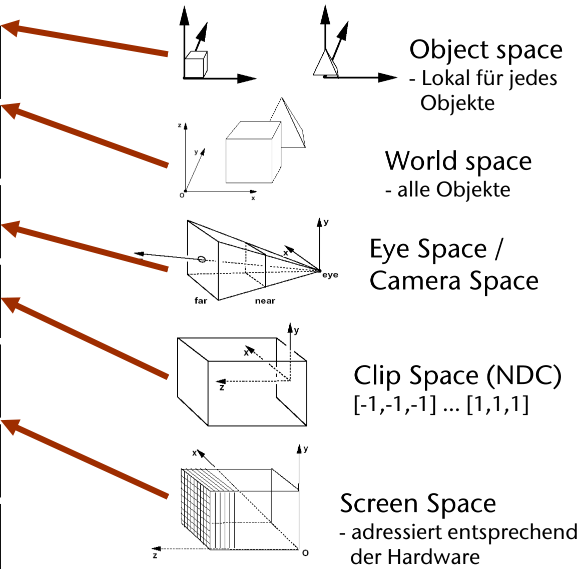
Im folgenden diese Tasks

Kennen wir (teilweise) schon
Z.B.:
- Scan Conversion

Arbeitet im 2D

Koordinatensysteme in der Pipeline

- Modeling Transformations
- Illumination (Shading)
- Viewing Transformation (Perspective / Orthographic)
- Clipping
- Projection (to Screen Space)
- Scan Conversion (Rasterization)
- Visibility / Display



Lineare und affine Abbildung

- Lineare Transformationen (z.B. Rotation, Skalierung und Scherung) können durch eine 3x3 Matrix dargestellt werden

The image cannot be displayed. Your computer may not have enough memory to open the image, or the image may have been corrupted. Restart your computer,

- Affine Transformationen (z.B. Translation), können **nicht** als 3x3 Matrix dargestellt werden

The image cannot be

- Merke die Konvention

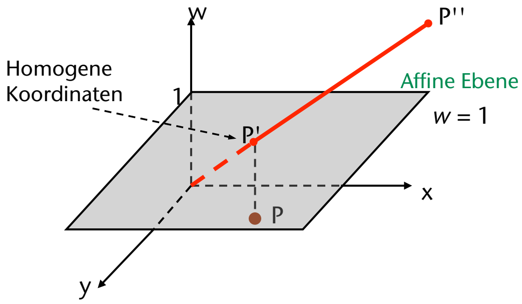
"Matrix mal Vektor"

Homogene Koordinaten im 3D

- Homogene Darstellung ist nützlich für Transformationen von Punkten und Vektoren
- Erweitert 3D Punkte und Vektoren zu 4D Punkte und Vektoren
- Homogener Punkt $P = (p_x, p_y, p_z, p_w) \quad p_w = 1$
- Homogener Vektor $\mathbf{p} = (p_x, p_y, p_z, p_w) \quad p_w = 0$

Veranschaulichung im 2D

- Erweitere Punkt P = (x,y) zu P' = $(x,y,1)$
- Assoziiere Linie $w \cdot (x,y,1) = (wx, wy, w)$ mit P'



- M.a.W.: ein 3D-Vektor (x, y, w) beschreibt ...
  - ... den 2D-Punkt $(x/w, y/w)$ für $w \neq 0$
  - ... den 2D-Vektor (x, y) für $w = 0$

Homogenisierung im 3D

- Der homogene Punkt

$$P = (x, y, z, w) \quad w \neq 0$$

beschreibt den Punkt an der Stelle

$$P = \left(\frac{x}{w}, \frac{y}{w}, \frac{z}{w}\right)$$

Operationen auf Punkten und Vektoren in homogenen Koord.

- Punkt + Vektor = Punkt

$$\begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} + \begin{pmatrix} v_x \\ v_y \\ v_z \\ 0 \end{pmatrix} = \begin{pmatrix} p_x + v_x \\ p_y + v_y \\ p_z + v_z \\ 1 \end{pmatrix}$$

- Vektor + Vektor = Vektor

$$\begin{pmatrix} u_x \\ u_y \\ u_z \\ 0 \end{pmatrix} + \begin{pmatrix} v_x \\ v_y \\ v_z \\ 0 \end{pmatrix} = \begin{pmatrix} u_x + v_x \\ u_y + v_y \\ u_z + v_z \\ 0 \end{pmatrix}$$

- Punkt – Punkt = Vektor

$$\begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} - \begin{pmatrix} q_x \\ q_y \\ q_z \\ 1 \end{pmatrix} = \begin{pmatrix} p_x - q_x \\ p_y - q_y \\ p_z - q_z \\ 0 \end{pmatrix}$$

Homogene Matrizen für Transformationen in 3D

- Matrix

$$M = \begin{pmatrix} m_{00} & m_{01} & m_{02} \\ m_{10} & m_{11} & m_{12} \\ m_{20} & m_{21} & m_{22} \end{pmatrix}$$

- Homogene Form

$$M_{4x4} = \begin{pmatrix} m_{00} & m_{01} & m_{02} & 0 \\ m_{10} & m_{11} & m_{12} & 0 \\ m_{20} & m_{21} & m_{22} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Lineare Abb. (Matrix-Vektor-Multiplikation)

- 3x3-Form

$$M \cdot \mathbf{v} = \begin{pmatrix} m_{00} & m_{01} & m_{02} \\ m_{10} & m_{11} & m_{12} \\ m_{20} & m_{21} & m_{22} \end{pmatrix} \cdot \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$$

- Homogene Form

$$M_{4x4} \cdot \mathbf{v}' = \begin{pmatrix} m_{00} & m_{01} & m_{02} & 0 \\ m_{10} & m_{11} & m_{12} & 0 \\ m_{20} & m_{21} & m_{22} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} v_x \\ v_y \\ v_z \\ 0 \end{pmatrix}$$

Affine Abbildungen im 3D

- 3x3-Form

$$M \cdot \mathbf{p} + t = \begin{pmatrix} m_{00} & m_{01} & m_{02} \\ m_{10} & m_{11} & m_{12} \\ m_{20} & m_{21} & m_{22} \end{pmatrix} \cdot \begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

- Homogene Form

$$M_{4x4} \cdot \mathbf{p}_4 = \begin{pmatrix} m_{00} & m_{01} & m_{02} & t_x \\ m_{10} & m_{11} & m_{12} & t_y \\ m_{20} & m_{21} & m_{22} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix}$$

- In homogenen Koordinaten lassen sich sogar affine Abbildungen als einfache Matrix-Vektor-Multiplikation darstellen!

Grundtransformationen im 3D

- Translation
- Rotation
- Skalierung
- Scherung (kommt in der Praxis fast nie vor)

- Verkettung
- Starrkörpertransformation (*rigid body transformation*)
- Gewöhnliche Transformation

Translation

- Eines Punktes

$$T_t \cdot P = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} = \begin{pmatrix} p_x + t_x \\ p_y + t_y \\ p_z + t_z \\ 1 \end{pmatrix}$$

- Eines Vektors

$$T_t \cdot \mathbf{v} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} v_x \\ v_y \\ v_z \\ 0 \end{pmatrix} = \begin{pmatrix} v_x \\ v_y \\ v_z \\ 0 \end{pmatrix}$$

- Inverse

$$T_t^{-1} = T_{-t}$$

Rotation

- Rotation um x-, y-, z-Achse um Winkel ϕ

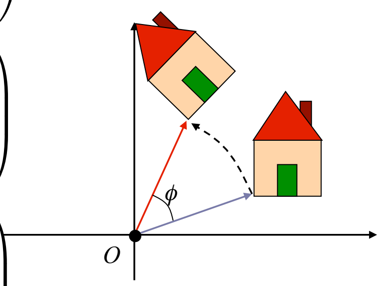
$$R_x(\phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{?}\cos\phi & \text{?}\sin\phi & 0 \\ 0 & \text{?}\sin\phi & \text{?}\cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

X-Koord. bleibt unverändert

Vorzeichentest: ϕ=90 →
y geht nach z, z geht nach -y.

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

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



Orthogonalität

- Falls R Rotationsmatrix ist ⇒ R orthogonal:

$$RR^T = R^T R = I$$

- Folgen:

$$\det(R) = \pm 1$$

$$R^{-1} = R^{\mathrm{T}}$$

R^{T} ist orthogonal

$$\|Rv\| = \|v\| \qquad \text{(Längenerhaltung)}$$

$$(Ru)\cdot(Rv) = u\cdot v \qquad \text{(Winkelerhaltung)}$$

$$R_1, R_2 \text{ sind orthogonal } \Rightarrow R_1 R_2 \text{ sind orthogonal}$$

Die Spalten von R sind zueinander *orthonormal* (nicht nur orthogonal!)

Skalierung

- Kann zum Vergrößern oder Verkleinern verwendet werden

$$S(s_x, s_y, s_z) = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- s_x, s_y, s_z beschreiben Längenänderung in x-, y-, z-Richtung
- Uniforme (isotrope) Skalierung: $s_x = s_y = s_z$
- Nicht-uniforme (anisotrope)
- Inverse

$$S^{-1}(s_x, s_y, s_z) = S\left(\frac{1}{s_x}, \frac{1}{s_y}, \frac{1}{s_z}\right)$$

- Alternative, homogene Skalierungs-Matrix:

$$S(s,s,s) = \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cong \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{s} \end{pmatrix}$$

- Aber besser die "normale" Skalierungsmatrix verwenden

Scherung

- Verschiebt z.B. die x-Koordinate abhängig von der Entfernung zur Ebene z=0 (d.h., xy-Ebene)
- Zum Beispiel: $H_{xz}(s)$ schert den x-Wert gemäß dem z-Wert

$$H_{xz}(s) \cdot \mathbf{p} = \begin{pmatrix} 1 & 0 & s & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} = \begin{pmatrix} p_x + sp_z \\ p_y \\ p_z \\ 1 \end{pmatrix}$$

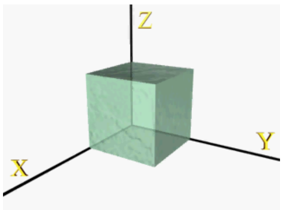
- Inverse:

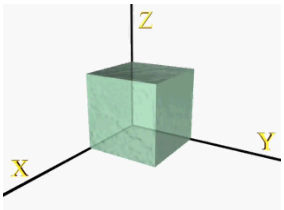
$$H_{xz}^{-1}(s) = H_{xz}(-s)$$

- Achtung: Determinante = 1
  → Volumen bleibt erhalten ...
  aber Winkel werden hier nicht erhalten!

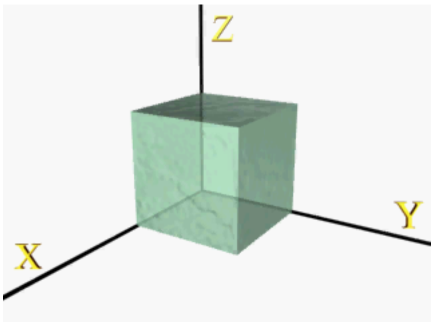
Movies

H_{XY} = X mit Y





H_{XZ} = X mit Z



H_{XZ} dann H_{ZY}

Spiegelung

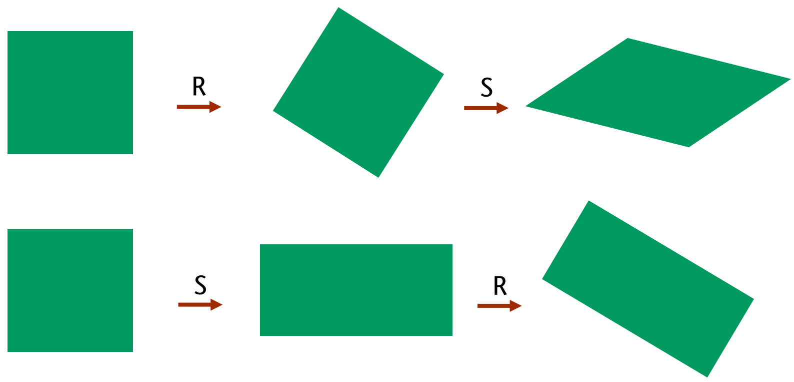
- Spiegelung entlang der x-Achse, m.a.W., Spiegelung bzgl. der yz-Ebene:

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

  - Analog die anderen beiden Spiegelungen
- Achtung: $$\det(M_x) < 0 \quad !$$
  - Bei allen anderen Transformationen T bisher war $$\det(T) > 0$$
- Spiegelungen sind in der CG eigtl. immer ausgeschlossen
  - U.a., weil der Umlaufsinn der Polygone umgedreht wird

Verknüpfung / *Concatenation*

- Nützlich zur Steigerung der Effizienz
- Achtung: Multiplikation von Matrizen ist nicht kommutativ → Reihenfolge der Transformation spielt eine Rolle!
- Beispiel:

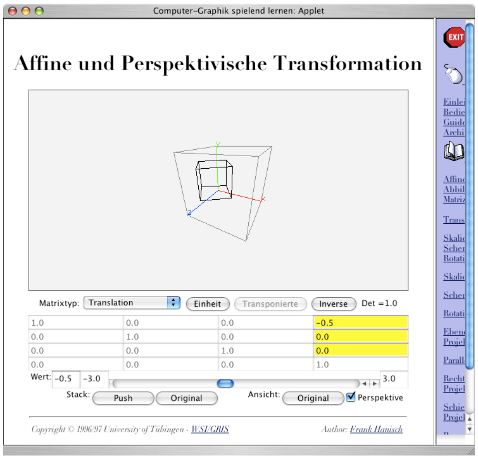


- Reihenfolge in einer Matrixkette:

$$p' = M_n \cdot \ldots \cdot M_2 \cdot M_1 \cdot p$$

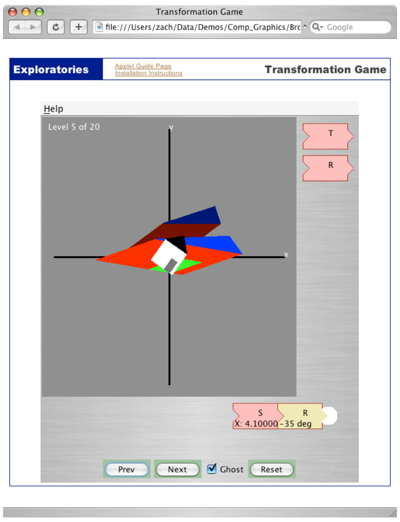
Reihenfolge der Ausführung

Demo zur Konkatenation von Transformationen



Quelle: http://www.gris.uni-tuebingen.de/gris/GDV/java/doc/html/etc/AppletIndex.html

und noch eine ...



http://www.cs.brown.edu/exploratories → Transformation Game

Wieviele Freiheitsgrade haben Rotationen?

- Antwort: 3 DOFs (*degrees of freedom*)
  - 2 DOFs für die Achse + 1 DOF für den Winkel; oder
  - 3 Euler-Winkel