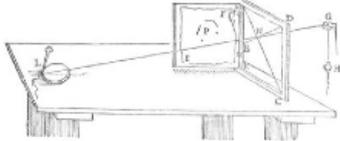





Computer-Graphik I

Projektionen & Perspektive

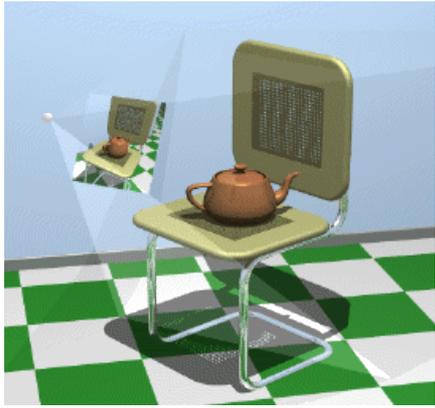



G. Zachmann
 Clausthal University, Germany
zach@in.tu-clausthal.de



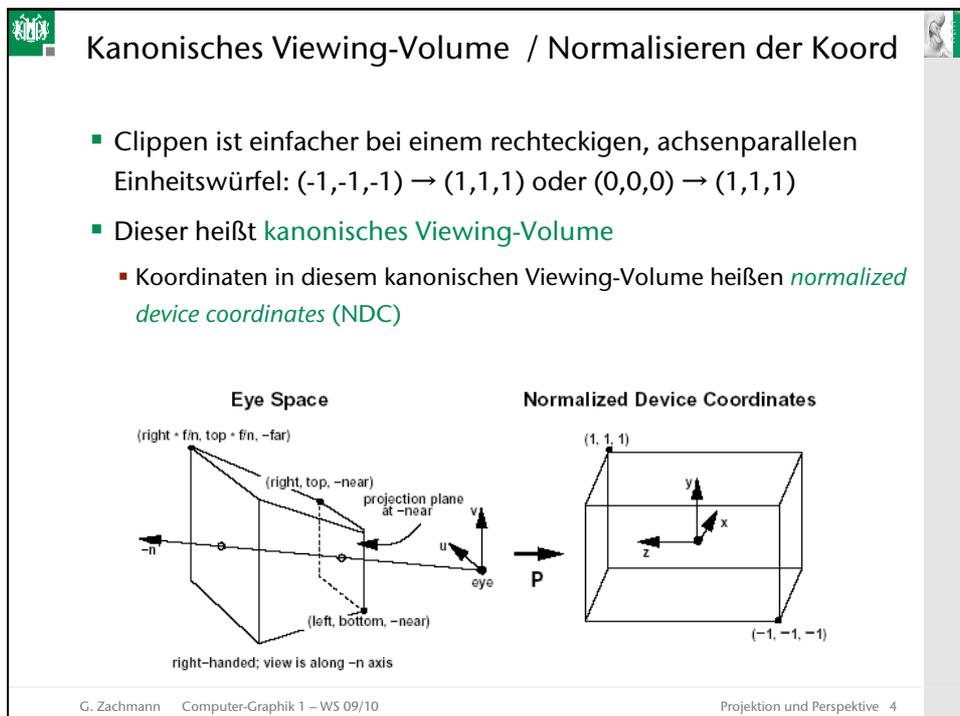
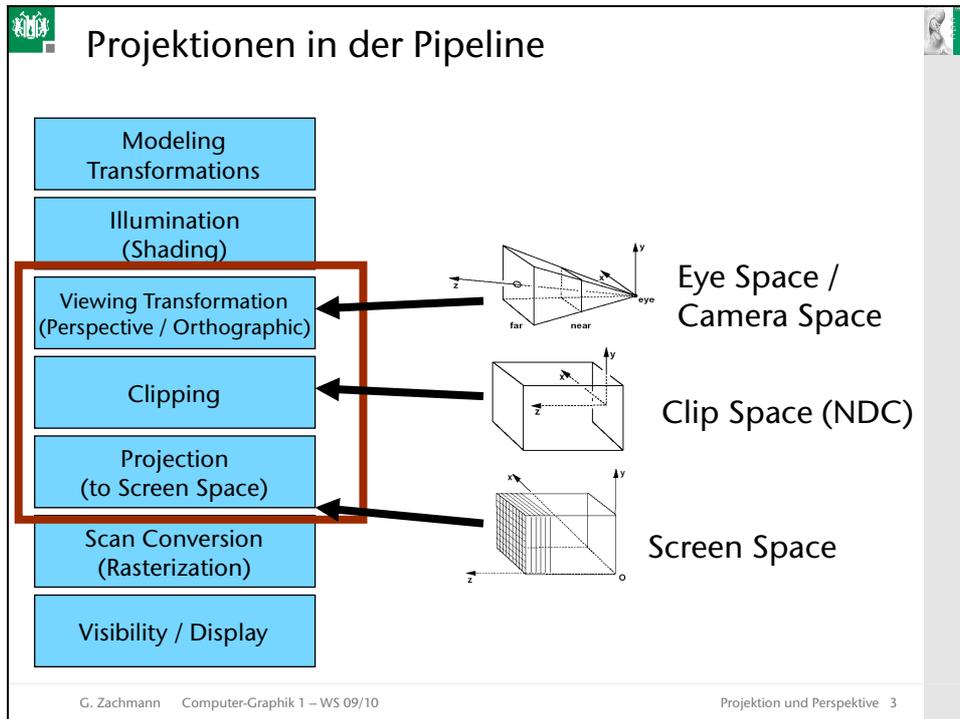

Motivation

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



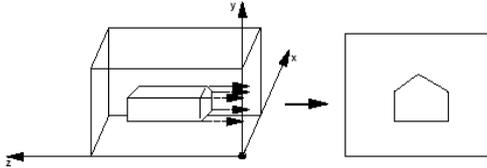
- Viewing** = Setup der richtigen Transformationen, um die 3D-Welt auf den 2D-Bildschirm zu projizieren

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 2

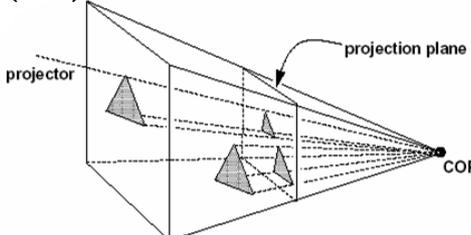


Orthographisch vs. Perspektivisch

- **Orthographisch** = parallele Projektionsstrahlen



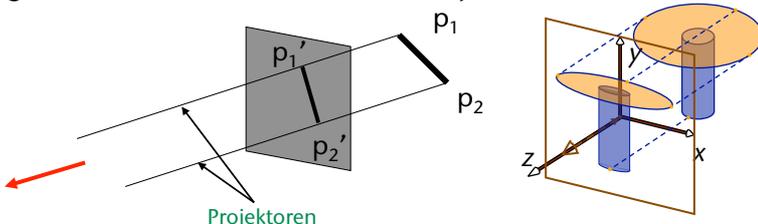
- **Perspektivisch** = alle Projektionsstrahlen laufen im Projektionszentrum (COP) zusammen



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 5

Orthographische Projektion

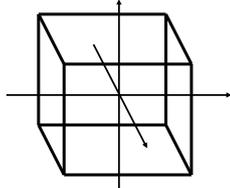
- Punkte werden orthogonal auf die Projektionsebene (*viewing plane*) projiziert
 - Projektionslinien verlaufen parallel zur Normalen der Projektionsebenen
- Eigenschaften:
 - Parallele Linien bleiben parallel
 - Winkelverhältnisse bleiben erhalten, aufgrund der parallelen Verschiebung zu Projektionsebene
- Es gibt noch andere (schiefe) Parallel-Projektionen



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 6

Die Projektionsmatrix

- Annahme: die komplette "virtuelle Welt" befindet sich im kanonischen Viewing-Volume $(-1, 1, -1, 1, -1, 1)$
- Die x- und y-Komponente bleiben unverändert
- Projektionsmatrix: setze z-Komponente auf 0

$$P_O = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$


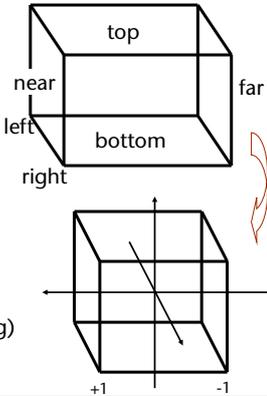
G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 7

Transformation in NDC

- Gegeben: eine achsenparallele Bounding Box (AABB) mit (links l , rechts r , unten b , oben t , near n , far f)
- Gesucht: Transformation, so daß die Box (l, r, b, t, n, f) in das kanonische Viewing-Volume $(-1, 1, -1, 1, -1, 1)$ übergeht
- Die Transformationsmatrix

$$P_O = \begin{pmatrix} \frac{2}{r-l} & 0 & 0 & 0 \\ 0 & \frac{2}{t-b} & 0 & 0 \\ 0 & 0 & \frac{2}{f-n} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -\frac{l+r}{2} \\ 0 & 1 & 0 & -\frac{t+b}{2} \\ 0 & 0 & 1 & -\frac{f+n}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Skalierung Translation (Zentrierung)



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 8

Perspektivische Projektion

- Am häufigsten verwendet in der Computergraphik & Malerei
- Unser Auge führt eine Zentralperspektive durch ("Lochkamera")
- Punkte werden entlang der Gerade zum Zentrum (z.B. Mittelpunkt der Augenlinse) auf die **View-Ebene** projiziert
- Zentrum der Projektion (COP)** ist nicht länger im unendlichen

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 9

Eigenschaften

- Entfernte Objekte sind kleiner (**perspektivische Verzerrung**)
- Parallele Linien werden **nicht auf parallele** Linien abgebildet, sondern laufen scheinbar in einem gemeinsamen Punkt zusammen
 - Solch ein Punkt heißt **Fluchtpunkt**

Zentrum der Projektion (Kamera / Auge)

Projektionsebene

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 10

Perspektive in der Geschichte der Malerei

- Erste Ansätze:

Giotto, 1305: Jesus vor Kaiphas

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 11

Brunelleschi's "Peep show" in Florenz, ca. 1410-1420

The Baptistry, San Giovanni, Florence

Duomo and Piazza del Duomo, Firenze

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 12

Schachbrettmuster wurden sehr beliebt



Christ Handing the Keys to St. Peter
 Pietro Perugino (1481-82), Fresco, Cappella Sistina, Vatican

Es dauerte eine Weile bis das Know-How europaweit bekannt war ...



Reconstruction of the temple of Jerusalem.
 From William of Tyre: *Histoire d'Outremer*.
 France, Rouen, XVe siècle
 Artiste: Maître de l'Échevinage

Erste perspektivische "Rätsel"



Die Gesandten
Hans Holbein
der Jüngere
(1533)

Eine Satire
über Perspektive

"Satire on
False Perspective"
by William
Hogarth,
1753

Bildunterschrift:

Whoever makes a
DESIGN without the
Knowledge of
PERSPECTIVE will be
liable to such
Absurdities as are
shewn in this
Frontispiece.



Eine etwas komplexere Konstruktion

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 17

Die Projektionsmatrix

- Ann.: Kamera im Ursprung, schaut in Richtung negative z-Achse
- Projektion auf eine Ebene $z = -d$, $d > 0$

$$q = \begin{pmatrix} -p_x \frac{d}{p_z} \\ -p_y \frac{d}{p_z} \\ -d \end{pmatrix} \cong \begin{pmatrix} -p_x \frac{d}{p_z} \\ -p_y \frac{d}{p_z} \\ -d \\ 1 \end{pmatrix} \cong \begin{pmatrix} p_x \\ p_y \\ p_z \\ -\frac{p_z}{d} \end{pmatrix}$$

$$Q = M \cdot P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{d} & 0 \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix}$$

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 18

- Mit anschließender Projektion auf Ebene $z = 0$

$$P_P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{d} & 0 \end{pmatrix}$$

- Beachte:
 - Wenn $d \rightarrow \infty$, dann entspricht P_P der orthographischen Projektion
 - Wenn $d \rightarrow 0$, dann wird

$$Q = \begin{pmatrix} p_x \\ p_y \\ p_z \\ -\frac{p_z}{d} \end{pmatrix}$$

instabil

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 19

Perspektivische Projektion in 2 Schritten

- Wie bei orthographischer Projektion kann man, wegen Clipping, die perspektivische Projektion in 2 Schritten machen:
 1. Perspektivische Abbildung (nicht Projektion!)
 2. Projektion

The diagram illustrates the two-step process of perspective projection. It shows a 3D coordinate system with x, y, and z axes. In the first step, a 3D object is shown in a perspective view, bounded by a 'Back plane' and a 'Front plane'. In the second step, the object is projected onto the 'Front plane', resulting in a 2D image.

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 20

■ Die Matrix für Schritt 1 (o.Bew.):

$$P_1 = \begin{pmatrix} \frac{2n}{r-l} & 0 & -\frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & -\frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

■ Achtung: der z-Wert in NDC hängt **nicht linear** vom z-Wert in Weltkoordinaten ab!

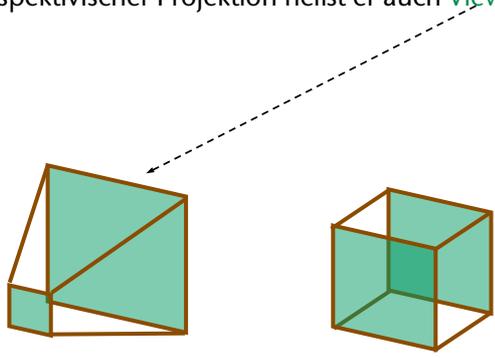
G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 21

Demo

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 22

Terminologie

- Der Bereich des 3D-Raumes, der auf den Bildschirm projiziert wird, heißt **View Volume**, oder **Viewing Volume**
- Bei perspektivischer Projektion heißt er auch **View Frustum**

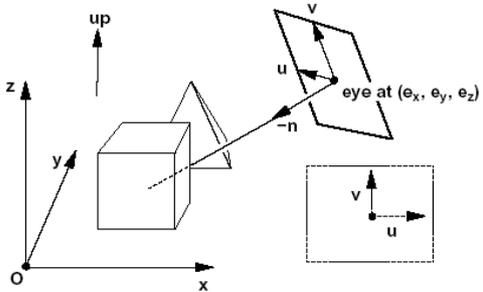


- Lat. "frustum" = (abgebrochener) Brocken

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 23

Transformation von World Space \rightarrow Eye Space

- Heißt **Viewing Transformation** oder **Camera Transformation**
- Positionieren der Kamera:
 - Augpunkt E**
 - "Up"-Vektor in Weltkoordinaten: dieser Vektor soll senkrecht auf dem Bildschirm erscheinen, also parallel zu v
 - Punkt C in Weltkoordinaten, der in der Mitte des Bildes erscheinen soll (heißt auch *Look-At*)
- Aufgabe: daraus das **Kamerakordinatensystem u, v, n** berechnen (*eye space*)



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 24

$$\mathbf{n} = \frac{E - C}{\|E - C\|}$$

$$\mathbf{u} = \frac{\mathbf{up} \times \mathbf{n}}{|\mathbf{up} \times \mathbf{n}|}$$

$$\mathbf{v} = \mathbf{n} \times \mathbf{u}$$

G. Zachmann Computer-Graphik 1 – WS 09/10
Projektion und Perspektive 25

- Transformation von Weltkoord. in Kamerakoord. = Translation + Wechsel der Orthonormalbasis
 - Gegeben: Koord.achsen x, y, z & u, v, n und der Punkt $P = (x, y, z)$
 - Bestimme P in u, v, n -Koord., also $P = (u, v, n)$
- Wechsel der Orthonormalbasis:

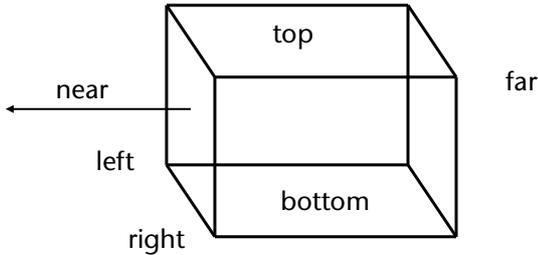
$$\begin{pmatrix} u \\ v \\ n \end{pmatrix} = \begin{pmatrix} u_x & u_y & u_z \\ v_x & v_y & v_z \\ n_x & n_y & n_z \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

mit $\mathbf{u} = (u_x, u_y, u_z)$ etc.

G. Zachmann Computer-Graphik 1 – WS 09/10
Projektion und Perspektive 26

Projektion in OpenGL

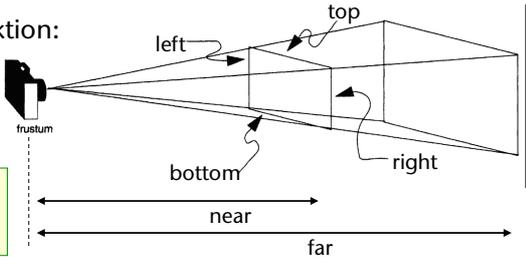
- Orthographische Projektion



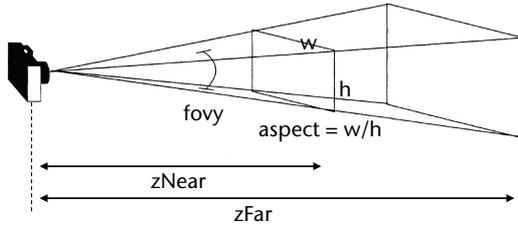
```
glOrtho(left, right, bottom, top, near, far);
```

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 27

Perspektivische Projektion:



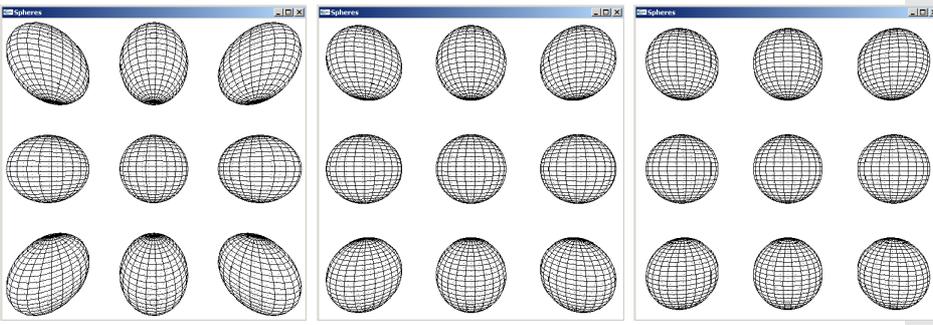
```
glFrustum( left, right,
           bottom, top,
           near, far );
```



```
gluPerspective( fovy,
                aspect,
                zNear,
                zFar );
```

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 28

■ Vorsicht bei Perspektive: Öffnungswinkel nicht zu groß wählen!

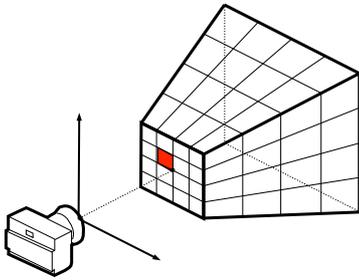


90° 60° 45°

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 29

Asymmetrisches Frustum

- Kommt manchmal vor, z.B.
 - Stereo-Projektion
 - Rendern eines Posters mit 10000 x 10000 Pixel (Framebuffer zu klein)
 - Mit `gluPerspective()` nicht möglich
- Poster: Zerlegen in viele Teilbilder mit voller Auflösung

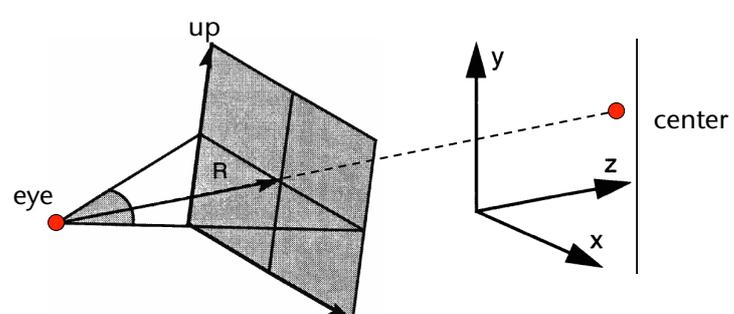


```
glFrustum( left, right,
           bottom, top,
           near, far );
```

Hier ist $left \neq right$, $top \neq bottom$!

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 30

Festlegen des Viewpoints mittels GL Utility



```
gluLookAt( eyeX, eyeY, eyeZ, centerX, centerY,
           centerZ, upX, upY, upZ );
```

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 31

Die Projektionsmatrix in OpenGL

- Projektionen werden auch durch Matrizen realisiert
- Achtung: für diese existiert eine eigene "globale" Matrix!
- Alle Transformations-Operationen (**glTranslate**, ..., **glFrustum**, ...) multiplizieren immer mit der aktuell "eingeschalteten" Matrix!!
- Umschalten mittels


```
glMatrixMode( { GL_MODELVIEW, GL_PROJECTION } );
```

 - Nach dem Umschalten beziehen sich alle Matrixbefehle auf die entsprechende Matrix
 - Zurückschalten auf **GL_MODELVIEW** nicht vergessen!

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 32

- Sollte man die Transformation nach Kamerakoord. in die MODELVIEW-Matrix oder in die PROJECTION-Matrix multiplizieren?
- Antwort: allg. üblich ist es, sie in die MODELVIEW-Matrix zu stecken, aber es geht genauso die PROJECTION-Matrix
 - In manchen Fällen, z.B. in einer Cave, **muß** man es sogar in der PROJECTION-Matrix machen
- Gesamtansicht aller Matrizen:

$$p' = M_{\text{proj}} \cdot V \cdot T \cdot \dots \cdot S \cdot R \cdot p$$

$\underbrace{M_{\text{proj}}}_{\text{GL_PROJECTION}} \cdot \underbrace{V \cdot T \cdot \dots \cdot S \cdot R}_{\text{GL_MODELVIEW}} \cdot p$

Zum Schluss: Transformation ins Kamerakoordinatensystem z.B. mit gluLookAt als erstes Kommando

Beliebige Transformationen



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 33

OpenGL-Beispiel

- Ein typisches OpenGL-Programm sieht dann ungefähr so aus:

```

glColor3f( 0.0, 0.0, 0.0 );
glClear( GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT );
glMatrixMode( GL_PROJECTION ); // set up projection
glLoadIdentity();

glOrtho( -1.0,1.0, -1.0,1.0, 1.0,10.0 );
oder
glFrustum( -1.0,1.0, -1.0,1.0, 1.0,10.0 );

glMatrixMode( GL_MODELVIEW ); // set up camera trf
glLoadIdentity();
gluLookAt( 0,0,1, 0,0,0, 0,1,0 );

glTranslatef( 0,0,tz ); // set up model-view
glRotatef( alpha, 1.0, 0.0, 0.0 );

render geometry ...

```

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 34

Demo

<http://www.xmission.com/~nate/tutors.html>

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 35

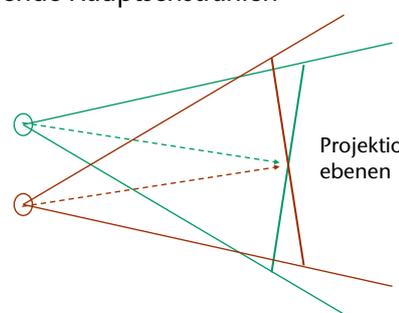
Stereo-Sehen

- Wichtiger "depth cue"
 - Nicht der wichtigste: nur bis einige Meter Entfernung
- Disparität im Auge = $\delta_2 - \delta_1 = \gamma - \alpha$
- Horopter = Punkte mit gleicher Tiefe wie fokussiertes Objekt = 0-Disparität
- Parallaxe auf dem Bildschirm:

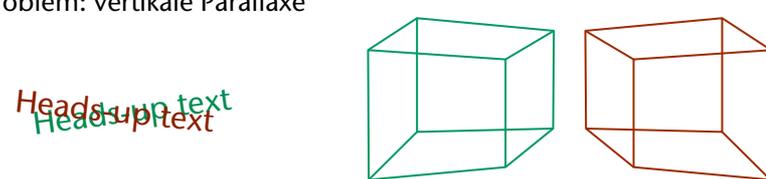
G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 36

Projektion

- Falsch: konvergierende Hauptsehstrahlen



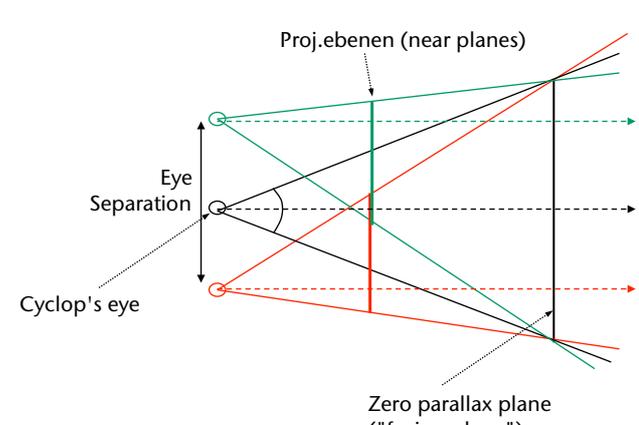
- Problem: vertikale Parallaxe



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 37

Stereoscopic Projection

- Richtig: parallele Hauptsehstrahlen
→ off-axis perspective projection



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 38

- Gegeben eye separation i , aspect ratio w/h , horizontal FOV α , near n , zero-parallax depth z_0
- Aufgabe: bestimme *left/right/top/bottom* für `glFrustum()`
- Annahme: kein Head-Tracking, d.h., Zyklopen-Auge befindet sich über der Mitte der Zero-Parallax-Plane
- *top* und *bottom* wie gehabt: $t = \frac{h}{w} l$
- Berechnung von *left* am Beispiel des linken Auges:

$$l_c = n \tan \frac{\alpha}{2}$$

$$l' - l_c = i \frac{z_0 - n}{z_0}$$

$$l = l_c + (l' - l_c) - i = l_c - i \frac{n}{z_0}$$

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 39

Stereoscopic Rendering in Qt

```

// Initialisierung
QGLFormat fmt;
fmt.setStereo(true);
QGLFormat::setDefaultFormat(fmt);
// oder
MyGLWidget* myWidget = new MyGLWidget(fmt, ...);

// Rendering
glDrawBuffer(GL_BACK_LEFT);
glClear(...);
glFrustum(...);
renderScene(...);
glDrawBuffer(GL_BACK_RIGHT);
glClear(...);
glFrustum(...);
renderScene(...);

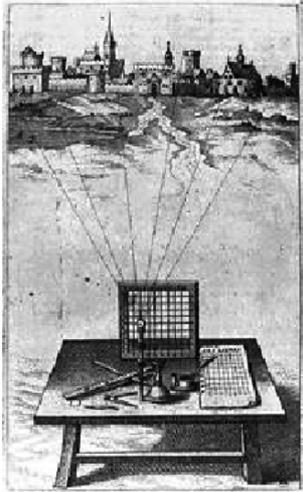
```

<http://doc.trolltech.com/4.2/qglformat.html>

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 40

■ Noch einige Beispiele aus der Kunst

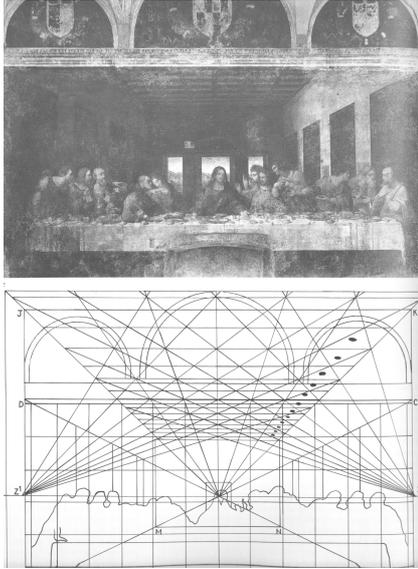
- Theoretisch wurde die Lösung des Problems der Perspektive von Leon Battista Alberti in seinem Buch *Della Pittura*, 1435-1436, beschrieben
- Brunelleschi löste es als erster praktisch 1410-1420



Alberti's *reticolato*

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 41

- Leonardo da Vinci:
There are some who look at the things produced by nature through glass, or other surfaces or transparent veils. They trace outlines on the surface of the transparent medium... But such an invention is to be condemned in those who do not know how to portray things without it, how to reason about nature with their minds... They are always poor and mean in every invention and in the composition of narratives, which is the final aim of this science



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 42

Geschichte Verwendung der Perspektive



Raffael: *Die Schule von Athen*

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 43

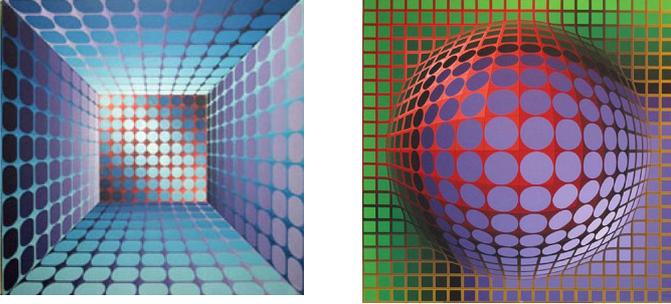
Multi-Perspektive, um den mystischen Eindruck zu erhöhen



De Chirico:

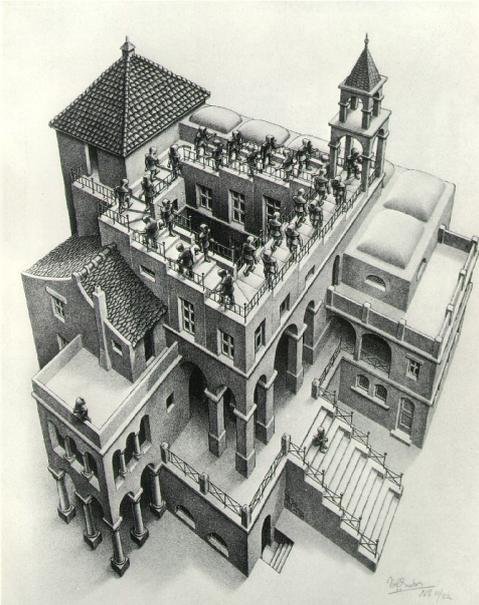
G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 44

Viktor Vasarely: Perspektive in der abstrakten Kunst



G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 45

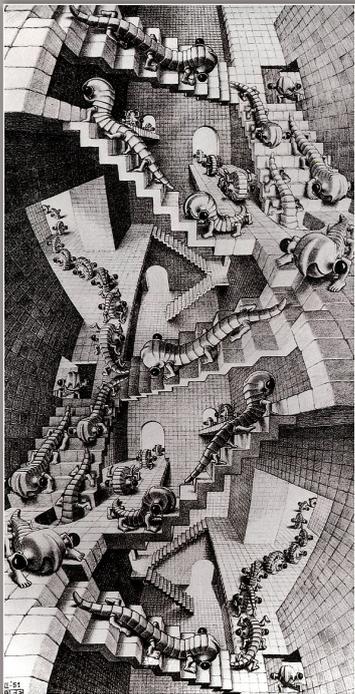
Einsatz der Perspektive zur Irritation des Betrachters



Maurits Cornelis ESCHER:
Ascending and Descending
1960
Lithograph

G. Zachmann Computer-Graphik 1 – WS 09/10 Projektion und Perspektive 46

Nicht-lineare Perspektive



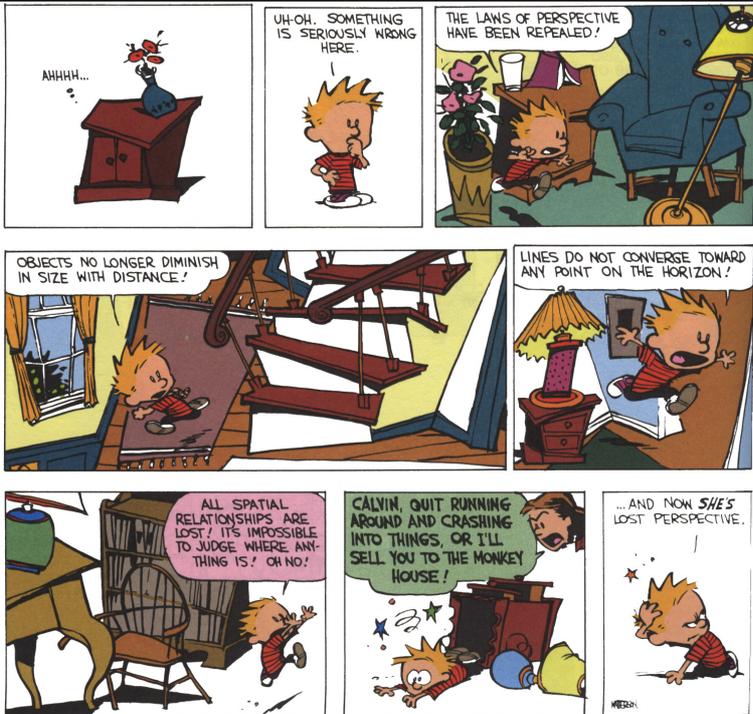
M. C. Escher

G. Zachmann Computer-Graphik 1 – WS 09/10

Art Center 47

calvin and hobbes

by WATTERSON



AHHH...

UH-OH. SOMETHING IS SERIOUSLY WRONG HERE.

THE LAWS OF PERSPECTIVE HAVE BEEN REPEALED!

OBJECTS NO LONGER DIMINISH IN SIZE WITH DISTANCE!

LINES DO NOT CONVERGE TOWARD ANY POINT ON THE HORIZON!

ALL SPATIAL RELATIONSHIPS ARE LOST! IT'S IMPOSSIBLE TO JUDGE WHERE ANYTHING IS! OH NO!

CALVIN, QUIT RUNNING AROUND AND CRASHING INTO THINGS, OR I'LL SELL YOU TO THE MONKEY HOUSE!

...AND NOW SHE'S LOST PERSPECTIVE.