# Digital Image Processing
# Projective Geometry

MECH-M-1-SEA-DBV-ILV

(Version September 2023)

© Kurt Niel
University of Applied Sciences Upper Austria
Faculty of Engineering and Environmental Sciences
September 1st , 2023

## Literature

(1) Burger W., Burge M., Digital Image Processing, Springer, 2010, ISBN 978-1-84628-379-6 (Chap. 16)

**(2)** Schreer O., Stereoanalyse und Bildsynthese, Springer, 2005, ISBN 3-540-23439-X (Chap. 2)

(3) Klette R., Zamperoni, Handbuch der Operatoren für die Bildverarbeitung, Vieweg, 1995, ISBN 3-528-16431-X (Chap. 4)

(4) Faugeras O., Three-Dimensional Computer Vision, The MIT Press, 1993, ISBN 978-0-262-06158-2 (Chap. 2, 3)

**(5)** Tratnig M., et.al., Camera Calibration, Data Segmentation, and Fitting Approaches for a Visual Edge Inspection System, IS&T/SPIE, 2004

(6) Hartley R., Zisserman A., Mulitple View Geometry in computer vision, Cambridge, 2006, ISBN 978-0-521-54051-3 (Part 0)

(7) Kovesi P., Matlab Library MatlabFns, The University of Western Australia, 2010 (Folder Projective)

Matlab Code:

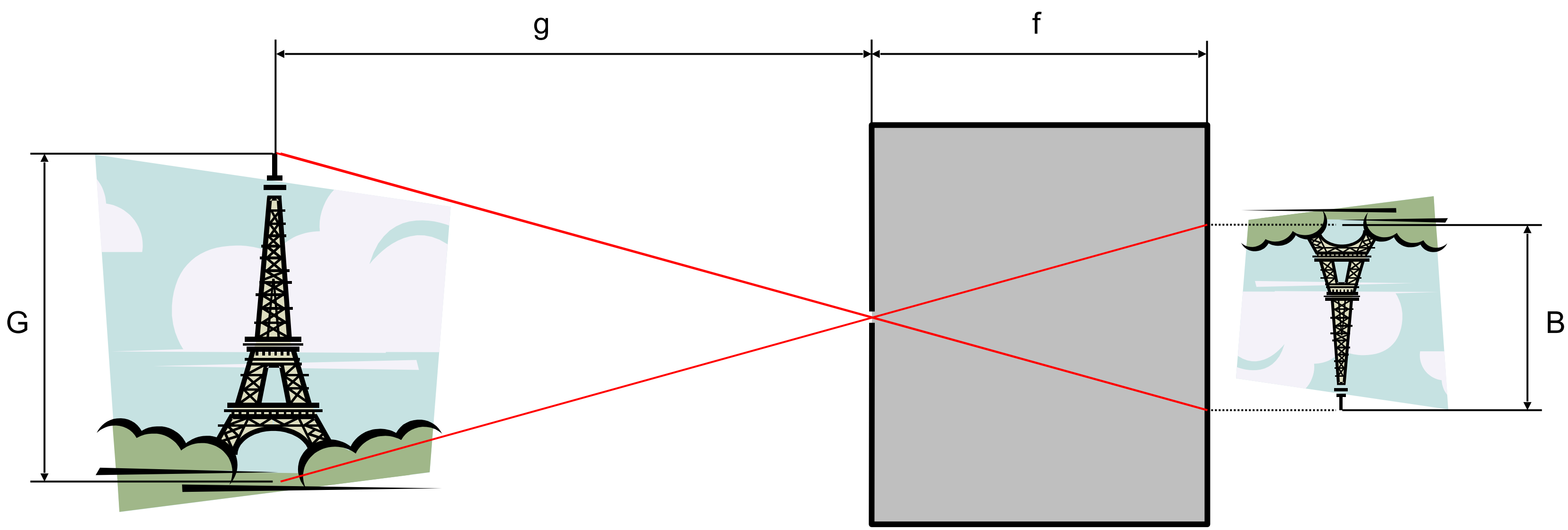
**(8)** Niel K., ProjectiveGeometry2022.zip

## PROJECTIVE GEOMETRY

- Pinhole camera model
- 2D image plane → 2D image plane via a single point
  - Euclidian, Affine, Projective mapping
- 3D object space → 2D image plane via a single point
- Define and evaluate Homography parameter set = camera calibration

Albrecht Dürer – Draughtsman Drawing a Recumbend Woman, ca. 1527

## Pinhole Camera Model

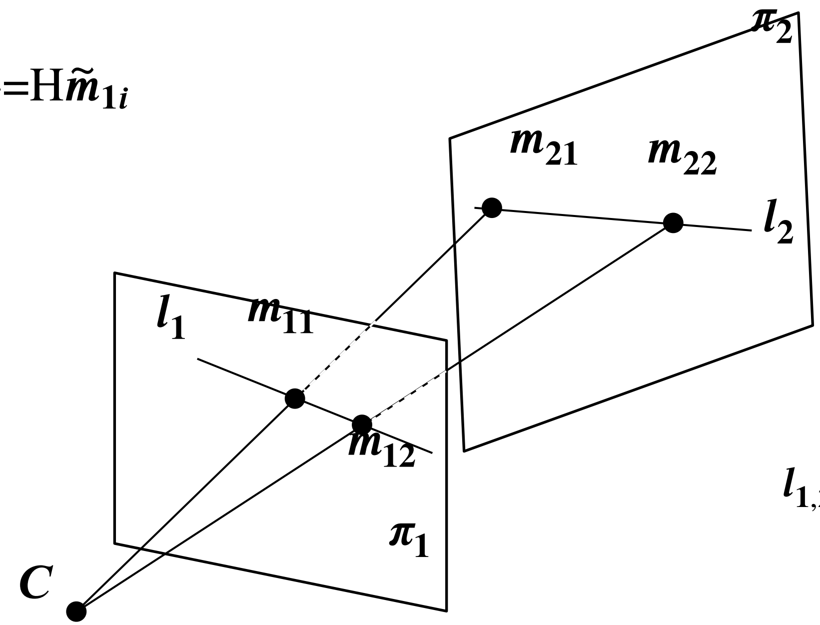



- Pinhole – all rays are crossing a certain point

  → central projection from the object space to the image space (plane!)

## Homography in $P^2$

$\tilde{m}_{2i}$=H$\tilde{m}_{1i}$




$\tilde{m}_{21}$=H$\tilde{m}_{11}$

$\tilde{m}_{22}$=H$\tilde{m}_{12}$

$l_{1,2}$ …Lines defined by Points $m_{1i,2i}$

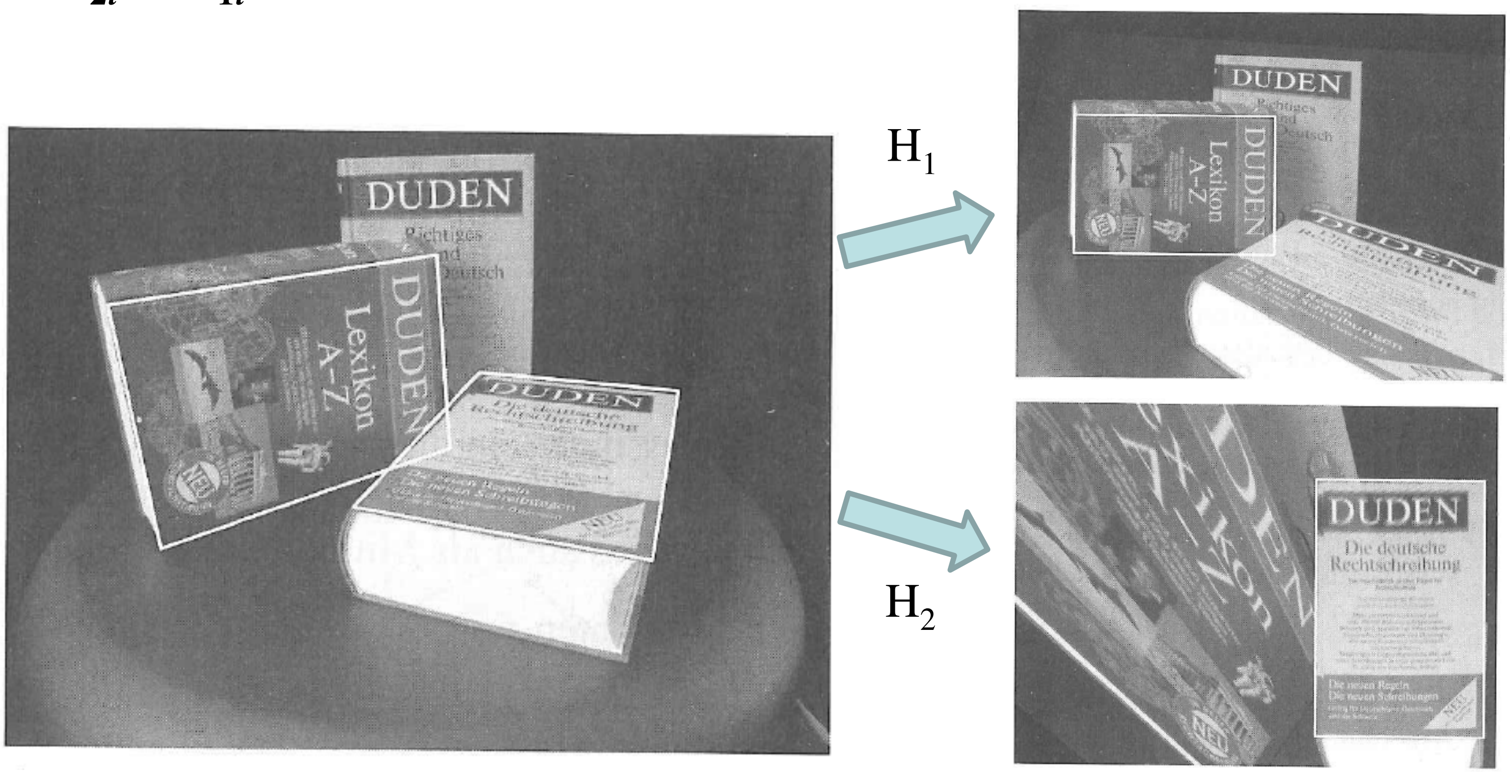
C … center point

$$\tilde{l}_2=\tilde{m}_{21}\times\tilde{m}_{22} = H\tilde{m}_{11}\times H\tilde{m}_{12}=H^*(\tilde{m}_{11}\times\tilde{m}_{12})=\det(H)\ H^{-T}\tilde{l}_1$$

$$\tilde{l}_2=H^{-T}\tilde{l}_1$$

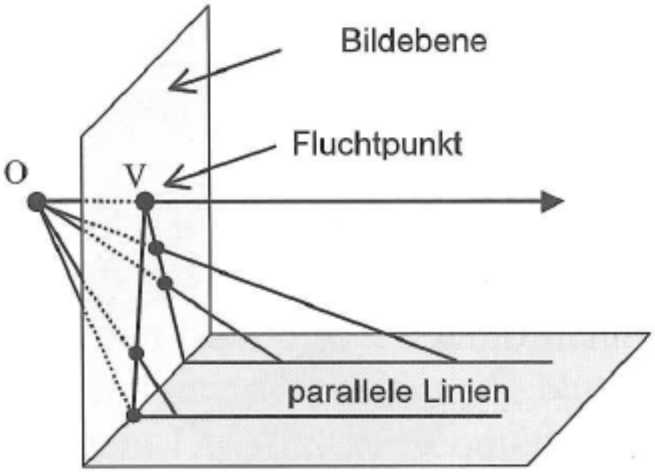
## Homography in $\mathbb{P}^2$

$$\tilde{\boldsymbol{m}}_{2i} = \mathrm{H}\tilde{\boldsymbol{m}}_{1i}$$



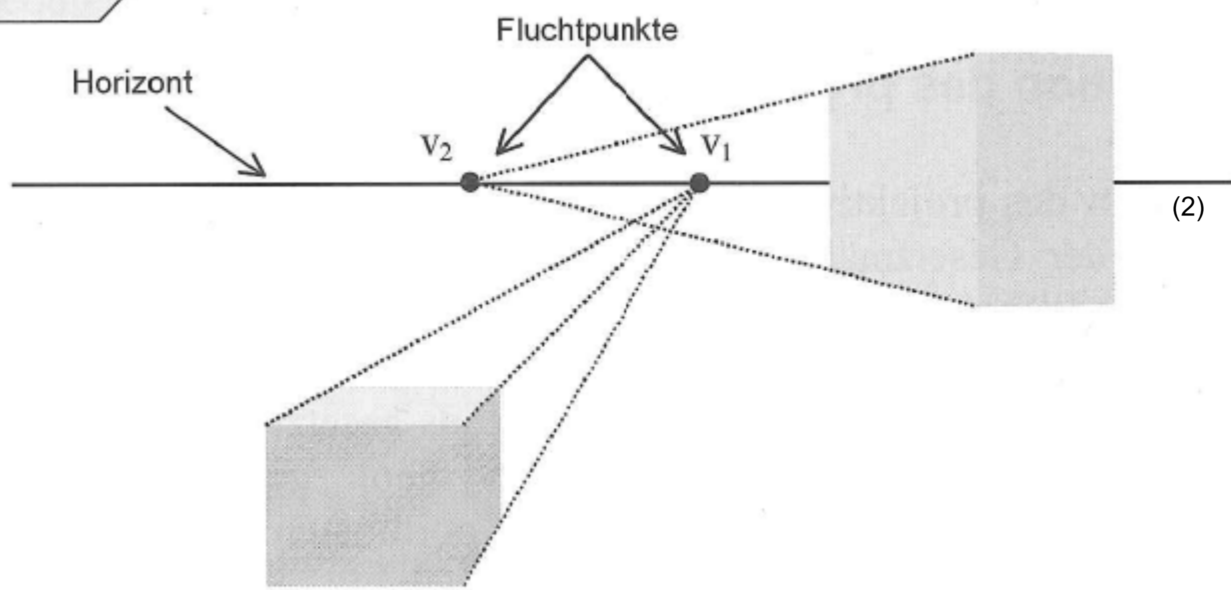

(2)

## Features



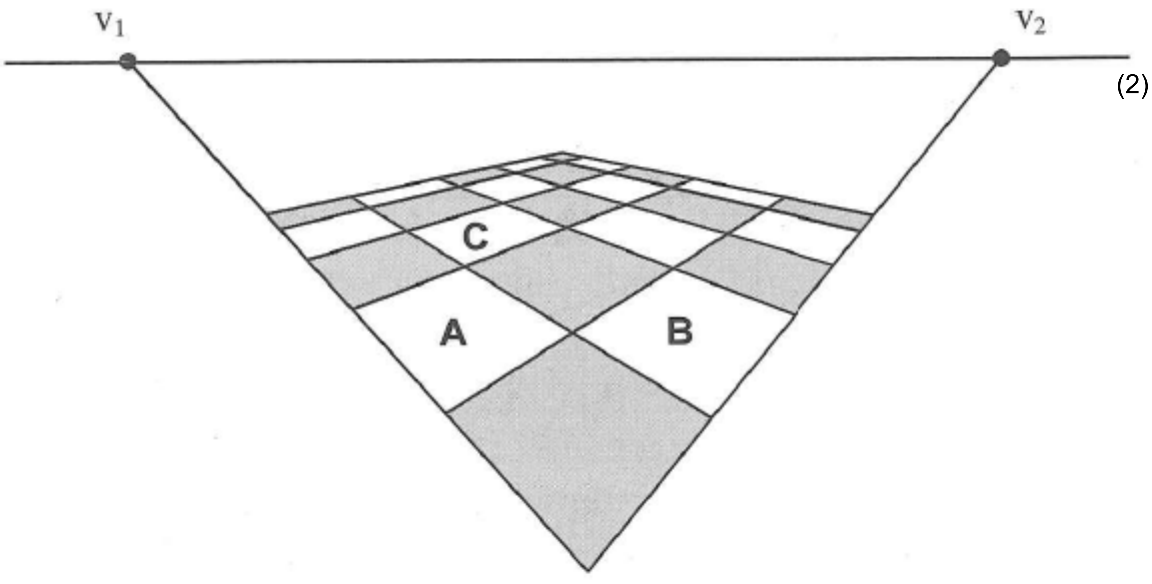

Vanishing point *(Fluchtpunkt)*

Absolut Conic *(hier Horizont)*




(2)

## Features




(2)

Perspective projection leads to variant geometrical features (lengths, areas) – in the original plane the areas A, B, and C are equal in size.

## Features

## Hierarchy of Transformations:

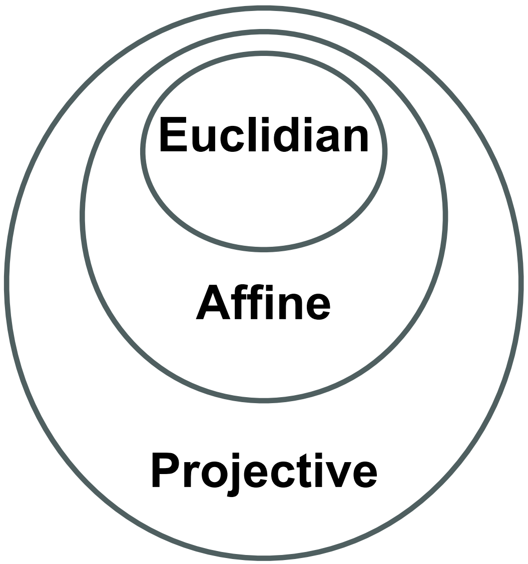



**Tabelle 2.3.** Hierarchie der Transformationen

| Transformationen | Matrix | Freiheitsgrade 2-D | Freiheitsgrade 3-D |
|---|---|---|---|
| Euklidisch | $\begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix}$ | 3 | 6 |
| Affin | $\begin{bmatrix} A & b \\ 0^T & 1 \end{bmatrix}$ | 6 | 12 |
| Projektiv | $\begin{bmatrix} A & b \\ v^T & v \end{bmatrix}$ | 8 | 15 |

**Tabelle 2.4.** Transformationen und Invarianten im euklidischen, affinen und projektiven Raum

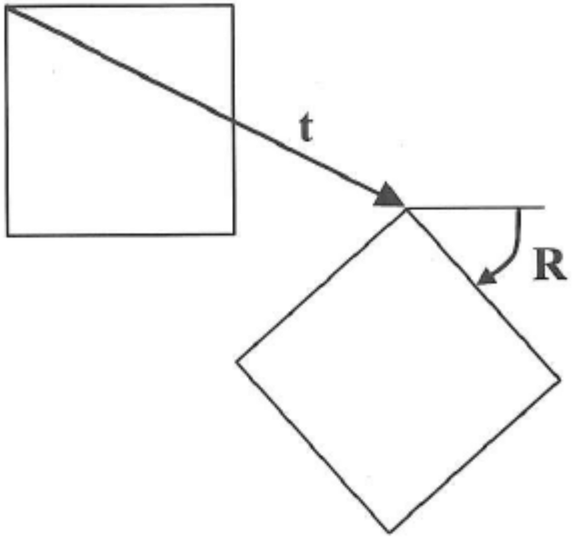
| Geometrie | euklidisch | affin | projektiv |
|---|---|---|---|
| Anzahl der Komponenten | n | n | n+1 |
| bis auf Skalierungsfaktor | nein | nein | ja |
| **Transformationen** | | | |
| Rotation, Translation | ✓ | ✓ | ✓ |
| Skalierung, Scherung | | ✓ | ✓ |
| Perspekt. Projektion | | | ✓ |
| **Invarianten** | | | |
| Länge, Winkel | ✓ | | |
| Verhältnisse, Parallelität | ✓ | ✓ | |
| Incidence, Kreuzverhältnis | ✓ | ✓ | ✓ |

(2)

## Euclidean Mapping – 2D

$$\mathbf{m}_2 = \mathbf{R}\mathbf{m}_1 + \mathbf{t} \qquad \tilde{\mathbf{m}}_2^3 = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \tilde{\mathbf{m}}_1^3$$

$$\begin{bmatrix} X_2 \\ Y_2 \end{bmatrix} = \mathbf{R} \begin{bmatrix} X_1 \\ Y_1 \end{bmatrix} + \mathbf{t}, \quad \text{mit} \quad \mathrm{R} = \begin{bmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{bmatrix} \quad \text{und} \quad \mathbf{t} = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$



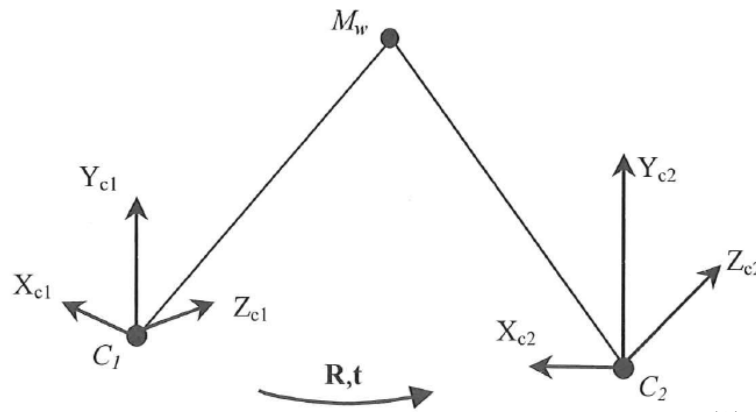

(2)

## Euclidean Mapping – 3D – transformation of coordinate systems

$$\tilde{\mathbf{m}}_2^3 = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \tilde{\mathbf{m}}_1^3$$

$$\tilde{\mathbf{m}}_i^3 = \begin{bmatrix} X_i & Y_i & Z_i & 1 \end{bmatrix}^T$$

$$M_{c1} = \mathbf{I} \cdot M_w \qquad M_{c2} = \mathbf{R} M_w + \mathbf{t}$$




(2)

$$M_w = \mathbf{R}^T \left( M_{c2} - \mathbf{t} \right) \Rightarrow M_{c1} = \mathbf{R}^T \left( M_{c2} - \mathbf{t} \right)$$

W … world coordinate
C … center point

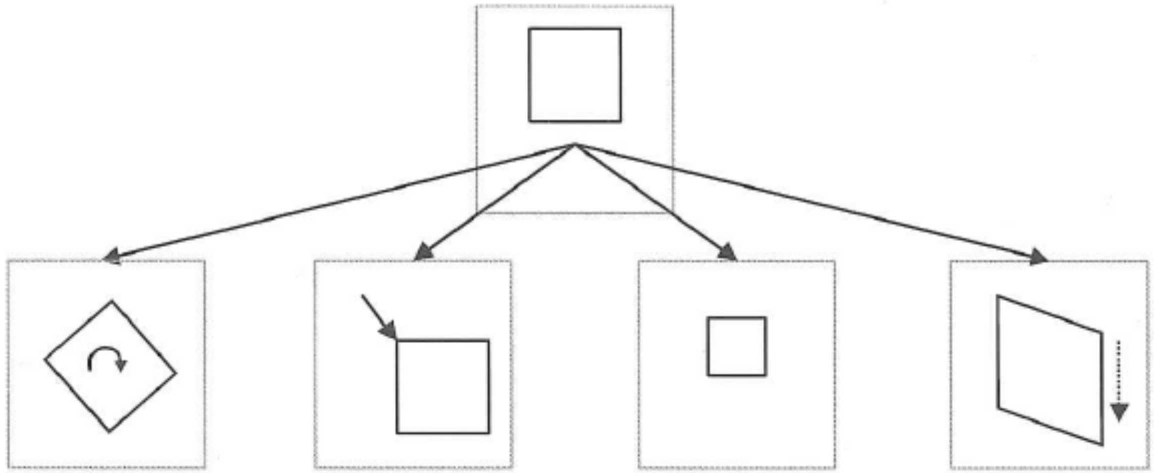
## Affine Mapping

$$\mathbf{m}_2^n = \mathbf{A}^n \, \mathbf{m}_1^n + \mathbf{b}^n \qquad\qquad \tilde{\mathbf{m}}_2^n = \begin{bmatrix} \mathbf{A}^n & \mathbf{b}^n \\ \mathbf{0}_n^T & 1 \end{bmatrix} \tilde{\mathbf{m}}_1^n$$

$$\begin{pmatrix} x_2 \\ y_2 \\ 0 \end{pmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 0 \end{pmatrix}$$

## Affine Mapping as a subgroup of Projective Mapping

$$\tilde{\boldsymbol{m}}_{2i} = \mathrm{H}\tilde{\boldsymbol{m}}_{1i}$$



(2)

**Rotation** **Translation** **Scaling** **Shear**

$$\underbrace{\underbrace{\underbrace{\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathrm{H}_{\mathrm{ro}}} \quad \underbrace{\begin{bmatrix} 1 & 0 & d_x \\ 0 & 1 & d_y \\ 0 & 0 & 1 \end{bmatrix}}_{\mathrm{H}_{\mathrm{tr}}}}_{\text{Euclidian mapping}} \quad \underbrace{\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathrm{H}_{\mathrm{sc}}} \quad \underbrace{\begin{bmatrix} 1 & s_x & 0 \\ s_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathrm{H}_{\mathrm{sh}}}}_{\text{Affine mapping}}$$

## Affine Mapping as a subgroup of Projective Mapping

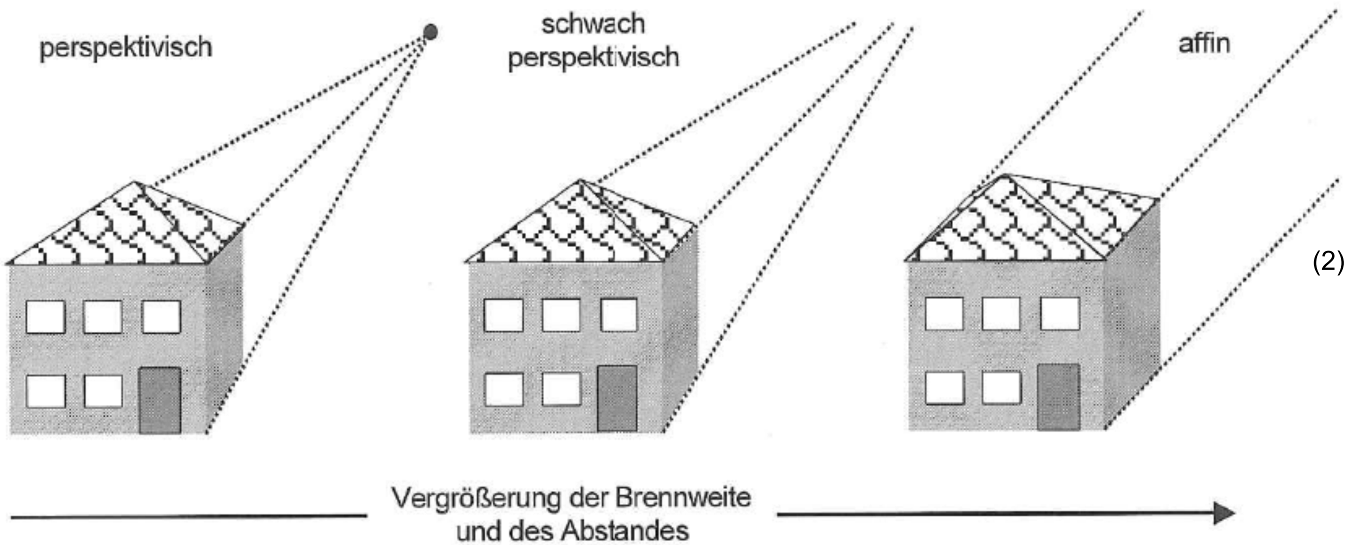
$$\tilde{\boldsymbol{m}}_{\mathbf{2}i} = \mathrm{H}\tilde{\boldsymbol{m}}_{\mathbf{1}i}$$

$$\mathbf{H} = \mathbf{H_{ro}} \,.\, \mathbf{H_{tr}} \,.\, \mathbf{H_{sc}} \,.\, \mathbf{H_{sh}}$$

$$\mathbf{H} = \begin{bmatrix} cos\alpha\; S_x - sin\alpha\; S_y s_y & cos\alpha\; S_x s_x - sin\alpha\; S_y & cos\alpha\; d_x - sin\alpha\; d_y \\ sin\alpha\; S_x + cos\alpha\; S_y s_y & sin\alpha\; S_x s_x + cos\alpha\; S_y & cos\alpha\; d_y + sin\alpha\; d_x \\ 0 & 0 & 1 \end{bmatrix}$$

## Homography in $\mathrm{P}^2$

$$\tilde{\boldsymbol{m}}_{2i} = \mathrm{H}\tilde{\boldsymbol{m}}_{1i}$$




(2)

**Camera Calibration** – get $\mathbf{H}$ values for the specific camera setup

**Pinhole camera model**

- External transformation
- Perspective transformation
- Internal transformation

**Normalized coordinates**

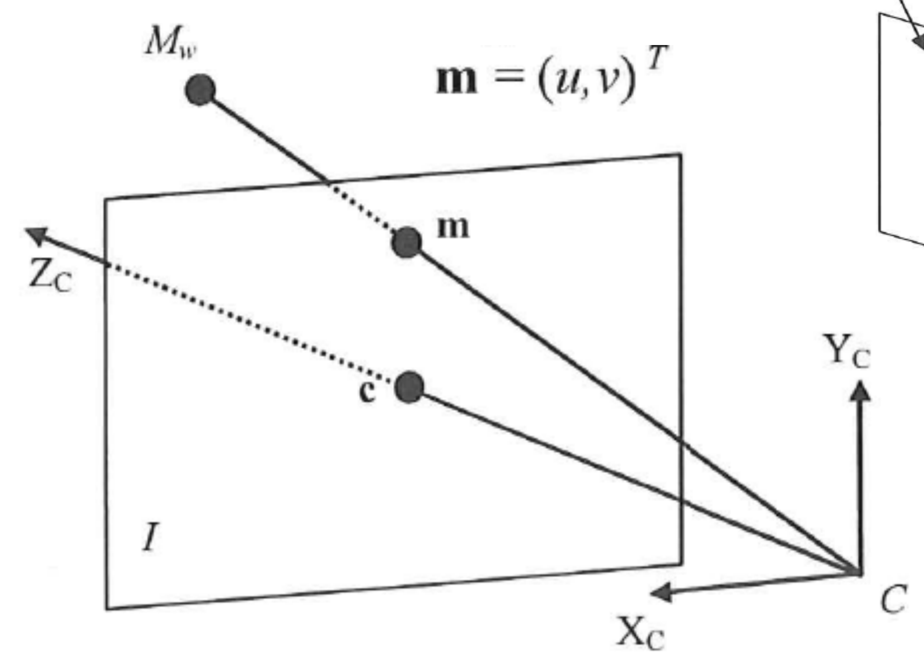
**Approximation**

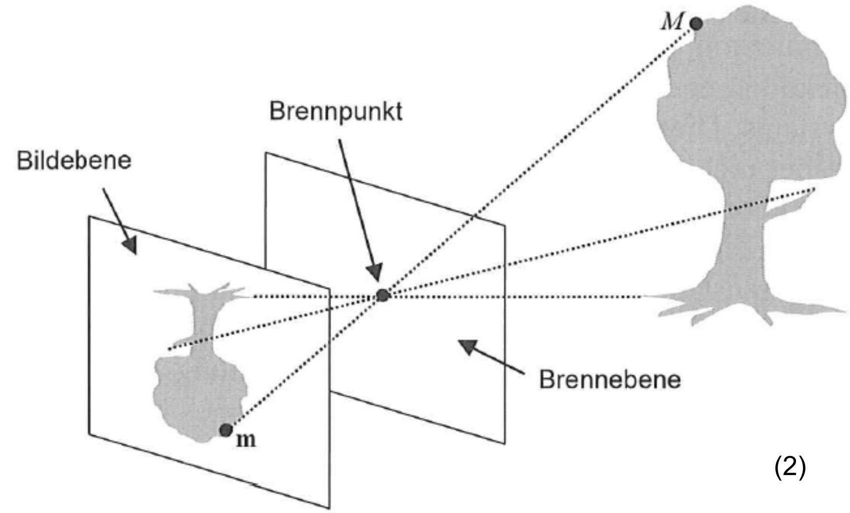
**Calibration by**

- planes
- vanishing points

## Pinhole Camera Model

$$M_w = (X_w, Y_w\ Z_w)^T$$




(2)

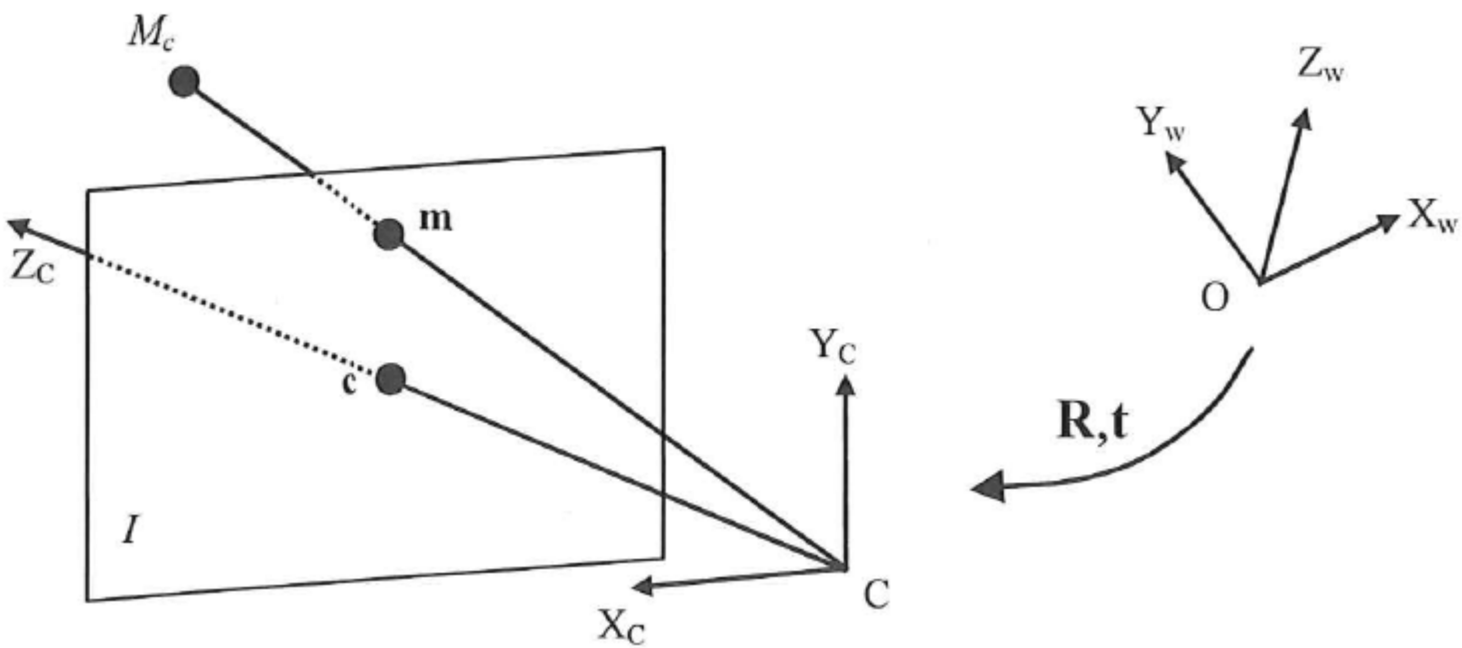
**Transformation $M_w$ to m**

1. Extern – *camera positioning*
2. Perspective – *lens*
3. Intern – *sensor chip*

W … world coordinate index
M … point in real world
C … center point
u, v … coordinate within plane I (sensor plane)
c … center coordinate = principal camera point
m … point in sensor plane

## **1/3 External Transformation** – *camera positioning*

$$M_c = \mathbf{R}\, M_w + \mathbf{t}$$

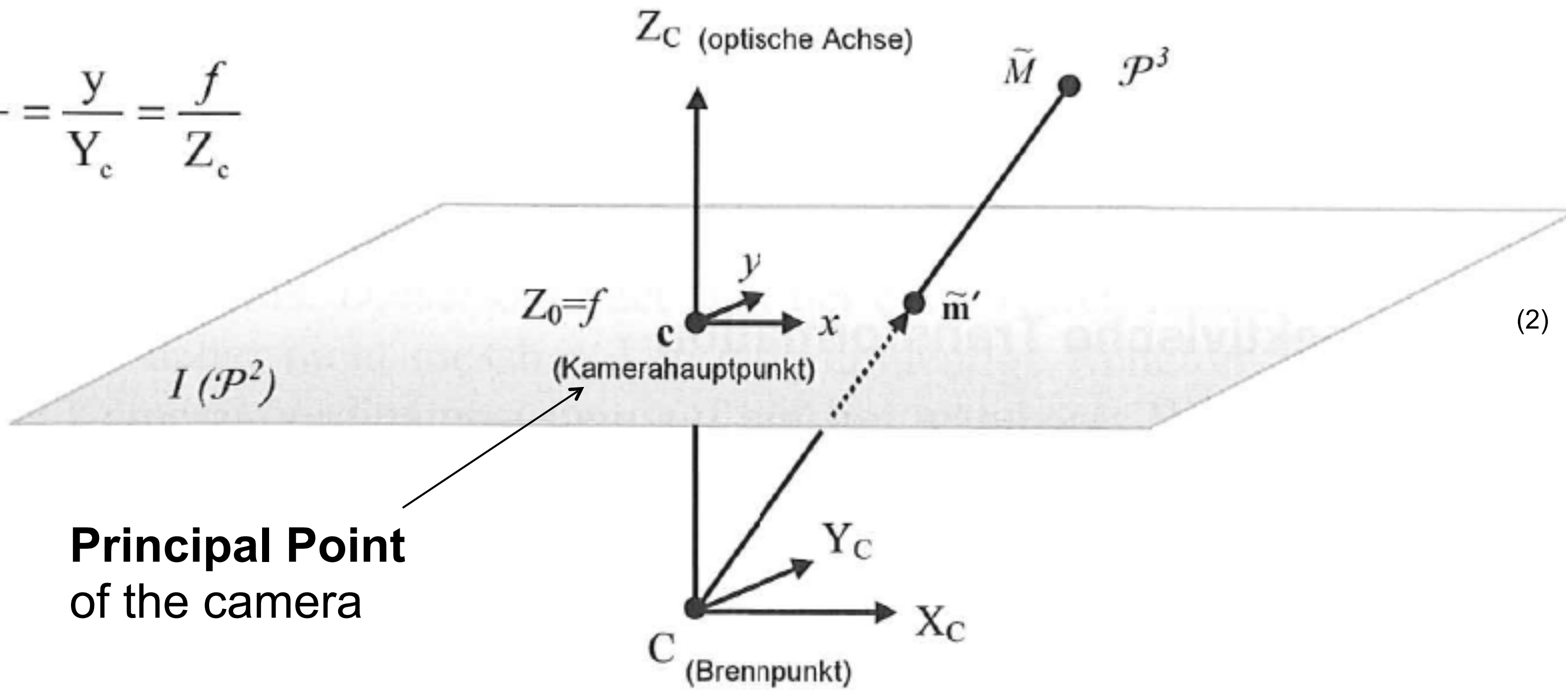



(2)

$$\tilde{M}_c = \mathbf{D}\,\tilde{M}_w \quad \text{mit} \quad \underbrace{\mathbf{D} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0_3^T & 1 \end{bmatrix}}_{\textbf{Extrinsic Matrix}}$$

## **2/3 Perspective Transformation** - *lens*

$$\frac{x}{X_c} = \frac{y}{Y_c} = \frac{f}{Z_c}$$




$$\begin{bmatrix} U \\ V \\ S \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}, \quad \text{mit } x = U/S, \quad y = V/S \quad \text{für } S \neq 0$$

## 2/3 Perspective Transformation - *lens*

$$\begin{bmatrix} U \\ V \\ S \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}, \quad \text{mit } \underbrace{x = U/S, \quad y = V/S}_{\text{Points at the image plane of the camera}} \quad \text{für } S \neq 0$$

$$s\tilde{\mathbf{m}}' = \mathbf{P}'\tilde{M}_c \quad \text{mit } \underbrace{\mathbf{P}' = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\textbf{Perspective Projection Matrix}} \quad \text{und } s = S$$

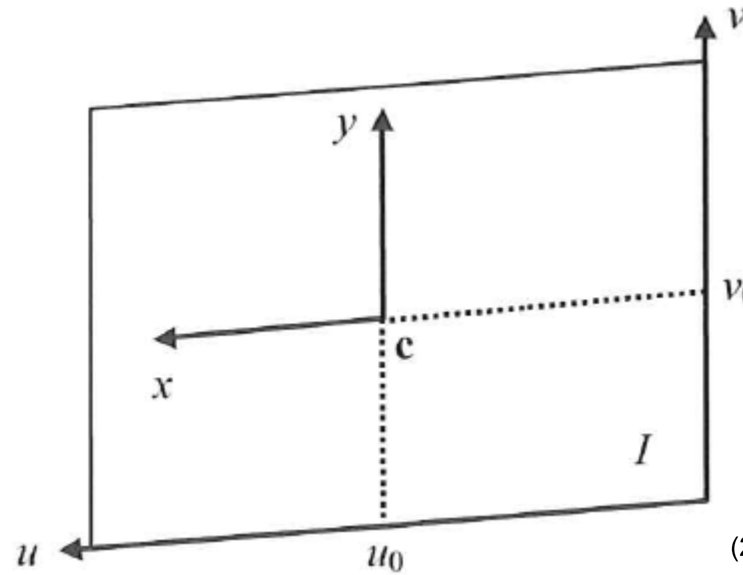
$$s\tilde{\mathbf{m}}' = \mathbf{P}'\,\mathrm{D}\,\tilde{M}_w$$

D .. Extrinsic Matrix

Additional correction of lens distortion (barrel/pillow) necessary!

## 3/3 Internal Transformation – *sensor chip*

$$\tilde{\mathbf{m}} = \mathbf{H}\tilde{\mathbf{m}}' \quad \text{mit} \quad \mathbf{H} = \begin{bmatrix} k_u & 0 & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$




(2)

$$s\tilde{\mathbf{m}} = \mathbf{HP}'\tilde{M}_c \quad \text{mit} \quad \mathbf{P}_{\text{neu}} = \mathbf{HP}' = \begin{bmatrix} fk_u & 0 & u_0 & 0 \\ 0 & fk_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

## Normalized Coordinates

Normalized coordinates
→ image plane has the distance f = 1 to the origin coordinate system

$$\mathbf{P}_N = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad s\tilde{\mathbf{m}} = \mathbf{HP}'\widetilde{M}_c \quad \text{mit} \quad \mathbf{P}_{\text{neu}} = \mathbf{HP}' = \begin{bmatrix} fk_u & 0 & u_0 & 0 \\ 0 & fk_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{P}_{\text{neu}} = \mathbf{AP}_N \quad \text{with} \quad \underbrace{\mathbf{A} = \begin{bmatrix} a_u & 0 & u_0 \\ 0 & a_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{\textbf{Intrinsic Matrix}} \quad \text{and} \quad a_u = fk_u, a_v = fk_v$$

$$\tilde{\mathbf{m}} = \mathbf{A}\tilde{\mathbf{m}}'$$

## Normalized Coordinates

$$s\tilde{\mathbf{m}} = \underbrace{\mathbf{A}\tilde{\mathbf{m}}'}_{\text{intern}} = \mathbf{A}\cdot\underbrace{\mathbf{P}_{\mathrm{N}}\,\tilde{M}_c}_{\substack{\text{perspek-}\\ \text{tivisch}}} = \mathbf{A}\mathbf{P}_{\mathrm{N}}\cdot\underbrace{\mathbf{D}\,\tilde{M}_w}_{\text{extern}} = \mathbf{A}\left[\mathbf{R}\ \mathbf{t}\right]\tilde{M}_w = \mathbf{P}\,\tilde{M}_w$$

with $\mathbf{P} = \mathbf{A}\left[\mathbf{R}\ \mathbf{t}\right]$

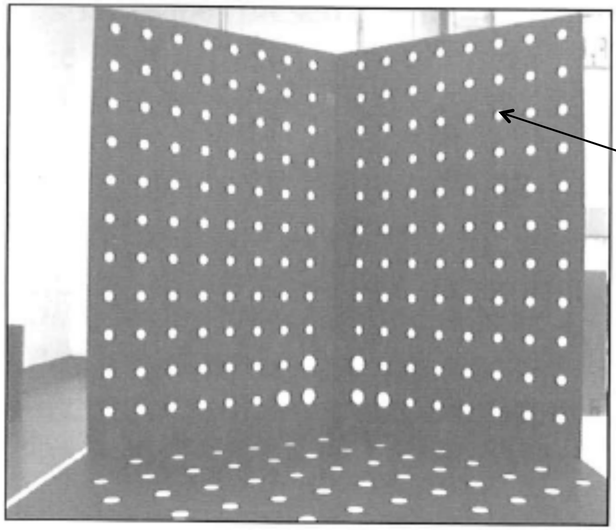
$$u = \frac{q_{11}\mathrm{X_w} + q_{12}\mathrm{Y_w} + q_{13}\mathrm{Z_w} + q_{14}}{q_{31}\mathrm{X_w} + q_{32}\mathrm{Y_w} + q_{33}\mathrm{Z_w} + q_{34}}$$

$$v = \frac{q_{21}\mathrm{X_w} + q_{22}\mathrm{Y_w} + q_{23}\mathrm{Z_w} + q_{24}}{q_{31}\mathrm{X_w} + q_{32}\mathrm{Y_w} + q_{33}\mathrm{Z_w} + q_{34}}$$

## Approximation

$$s\tilde{\mathbf{m}} = \mathbf{P}\tilde{M}_w, \quad \text{mit} \quad \mathbf{P} = \mathbf{A}\left[\mathbf{R}\ \mathbf{t}\right] = \begin{bmatrix} a_u \mathbf{r}_1^T + u_0 \mathbf{r}_3^T & a_u t_x + u_0 t_z \\ a_v \mathbf{r}_2^T + v_0 \mathbf{r}_3^T & a_v t_y + v_0 t_z \\ \mathbf{r}_3^T & t_z \end{bmatrix}$$

e.g., by using a calibration piece with exactly known 3D measuring points:



(2)

The camera plane/pixel positions of the points must be determined (e.g., COG – center of gravity – of the white circle areas)

## Approximation

Ignoring lens distortion and taking a linear model a proper accuracy can be worked out by approximation of the parameters:

$$s\tilde{\mathbf{m}} = s\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{P}\widetilde{M}_w = \left(\begin{array}{c|c} \mathbf{q}_1^T & q_{14} \\ \mathbf{q}_2^T & q_{24} \\ \mathbf{q}_3^T & q_{34} \end{array}\right)\widetilde{M}_w$$

with $\mathbf{m} = (u, v)^T$

$$(\mathbf{q}_1 - u \cdot \mathbf{q}_3)^T \cdot M + q_{14} - u \cdot q_{34} = 0$$
$$(\mathbf{q}_2 - v \cdot \mathbf{q}_3)^T \cdot M + q_{24} - v \cdot q_{34} = 0$$

using N measuring points → 2N homogenous linear equations $\mathbf{A} \cdot \mathbf{x} = 0$

**A** is 2Nx12-matrix; **x** has the dimension 12x1: $\mathbf{x} = \left[\mathbf{q}_1^T, q_{14}, \mathbf{q}_2^T, q_{24}, \mathbf{q}_3^T, q_{34}\right]^T$

## Approximation

Because $\mathbf{q}_3$ contains the last line of the rotation matrix, the following additional condition appears for excluding trivial solutions:

$$\left\|\mathbf{q}_3\right\| = 1$$

$$\boxed{\left\|\mathbf{A} \cdot \mathbf{x}\right\| \rightarrow \min}$$

→ Determining Eigenvektoren of a 3x3 matrix
→ Inverting a 9x9 matrix

The results are:

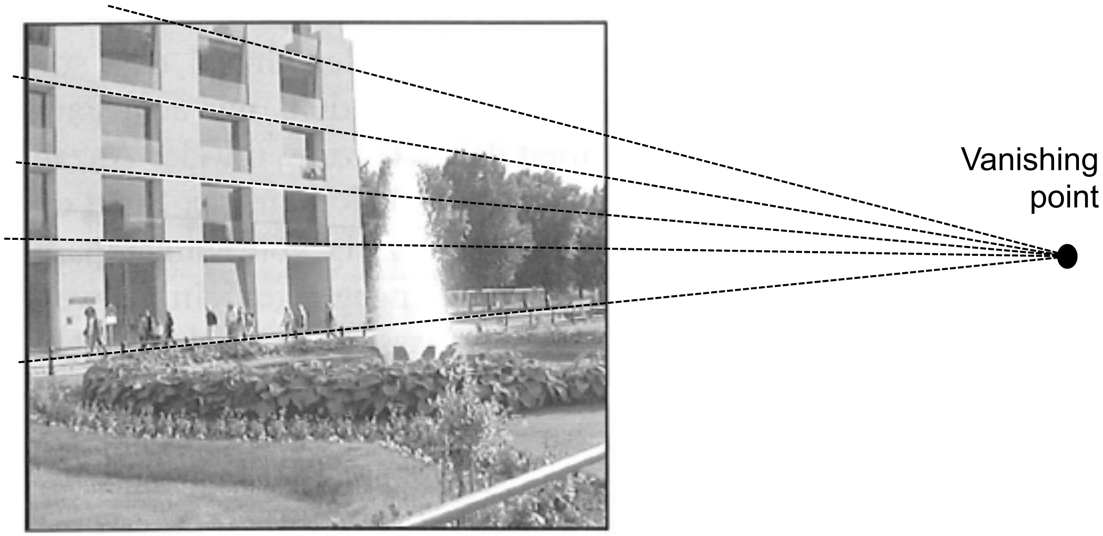
- ✓ Components of the projective matrix implicit including
  - ✓ Intrinsic and
  - ✓ Extrinsic parameters.

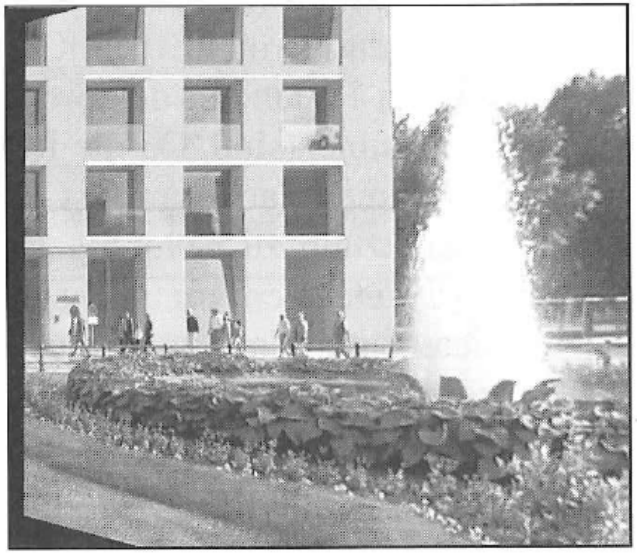
## Calibration by Planes / by Vanishing Points

### Planes
→ At least three plane squares at planes which are not parallel are needed for calculating the essential homograph parameters

### Vanishing points
→ at least five conjugated pairs of vanishing points (= five vanishing lines) are needed for calculating the essential homograph parameters

(2)



## Conference Paper – SPIE 2004

[5]

# Camera Calibration, Data Segmentation, and Fitting Approaches for a Visual Edge Inspection System

Mark Tratnig[a], Paul O'Leary[b], Helmut Hlobil[c], Johann Reisinger[d]

[a] Christian Doppler Labor für Sensorische Messtechnik, University of Leoben, Peter Tunner Strasse 27, A-8700 Leoben, Austria

[b] Institute for Automation, University of Leoben, Peter Tunner Strasse 27, A-8700 Leoben, Austria

[c, d] Voest-Alpine Mechatronics Ges.m.b.H., A-4031 Linz, Austria

## ABSTRACT

The design of edges is very important for many components. In this paper we therefore present a light-sectioning based measurement head, which is suitable for the edge inspection of different workpieces. Beyond the design we also present a new calibration technique for its camera. The calibration is mainly based on several perspective projections, which are successively executed. In each step, the linear system of homogeneous equations is solved by using singular value decomposition. Each mapping is therefore obtained in the least squares sense. Because of the novel design of the calibration device, a high number of reference points can be used for the description of these mappings.

The inspection of a workpiece detail implicates a large amount of data, some of which is useless. To extract the data essential for the fitting routines, a special correlation/regression based template matching is proposed.