3D Computer Vision (SoSe2024)

Primitives and Transformations in 1D/2D

Prof. Dr. Ulrich Schwanecke

RheinMain University of Applied Sciences

CVMR
Computer Vision & Mixed Reality Group

Last lecture

Last lecture

- Organization
- Prerequisites
- Introduction to Computer Vision
- History of Computer Vision

Today’s Lecture

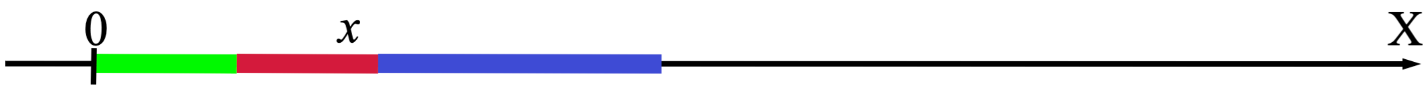
Today's Lecture

- 1D Transformations
- Primitives and Transformations
  - Homogeneous Coordinates
  - Points, Lines and Planes
  - 2D Transformations
  - Homography Estimation

1D Transformations

1D Transformations

- Original picture

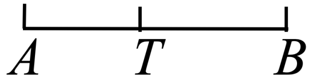


- **Euclidean Transformation**: Translation by b

  - Matrix vector representation: $x = (1, b)\begin{pmatrix} x \\ 1 \end{pmatrix}$
  - **Invariant**: Distances (Lengths)

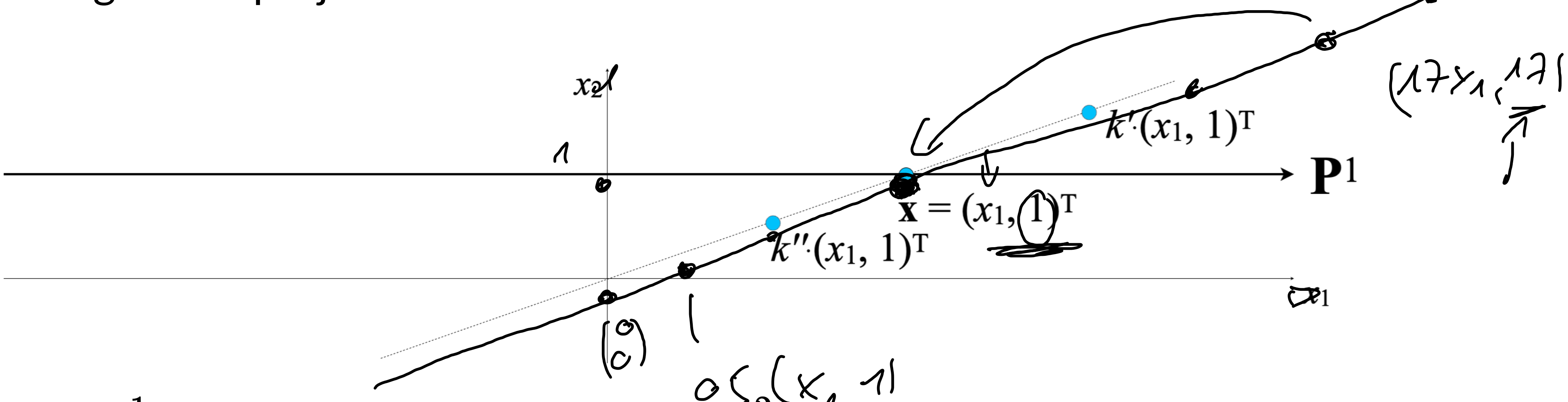
- **Similarity Tranformation**: Translation by b, Scale with a

  - Matrix vector representation: $x = (a, b)\begin{pmatrix} x \\ 1 \end{pmatrix}$

- **Invariant**: Ratio of differences $TV(ATB) = length(AT) : length(TB)$ 

The Projective Line (Projective 1D Space)

- Embedding of the projective line $\mathbb{P}^1$ in $\mathbb{R}^2$



- A point in $\mathbb{P}^1$ is represented by a vector in $\mathbb{R}^2$
- Each vector in $\mathbb{R}^2$ is a representative of an equivalence class that defines a point in $\mathbb{P}^1$: $\tilde{\mathbf{x}} = (x_1, 1)^T \sim k(x_1, 1)^T, \quad \forall k \in \mathbb{R} \setminus \{0\}$
- $(k, 0)^\top$ describes a point at infinity (**vanishing point**)

Homogeneous Representation of 1D Transformations

- **Euclidean transformation**: Translation by b

  non homogeneous: $x' = (1, b)\begin{pmatrix} x \\ 1 \end{pmatrix}$, homogeneous: $\begin{pmatrix} x' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ 1 \end{pmatrix}$

- **Similarity transformation**: Translation by b, Scale with a

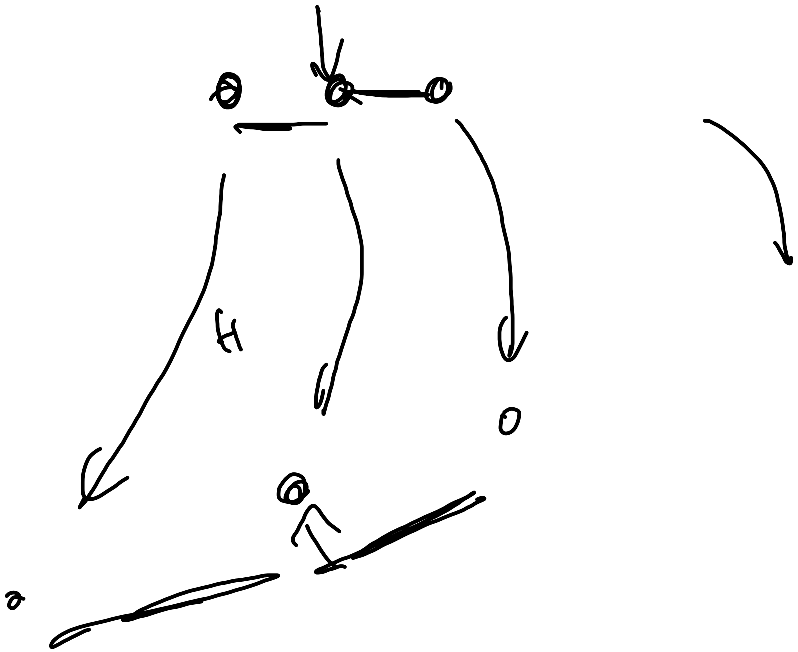
  non homogeneous: $x' = (a, b)\begin{pmatrix} x \\ 1 \end{pmatrix}$, homogeneous: $\begin{pmatrix} x' \\ 1 \end{pmatrix} = \begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ 1 \end{pmatrix}$

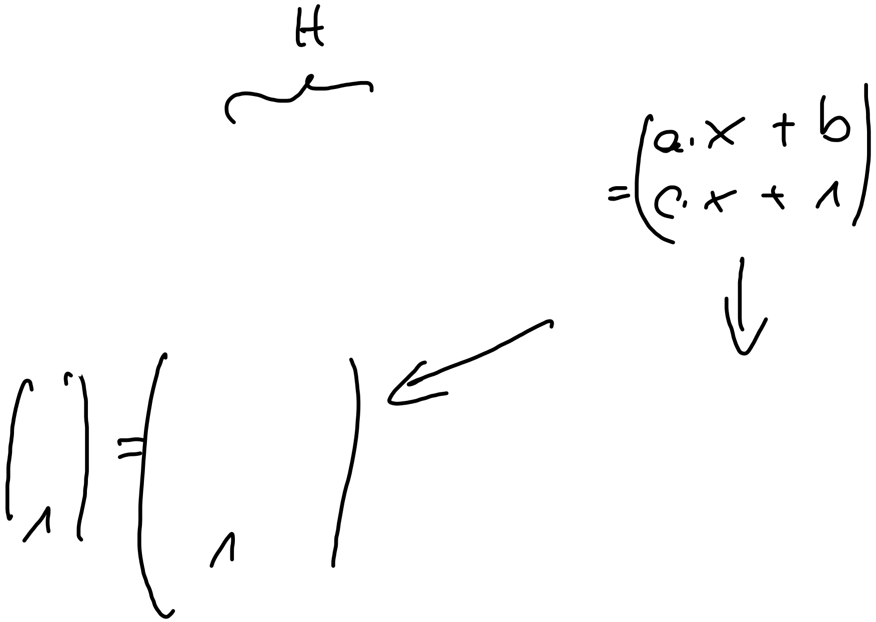
- **General form** of homogeneous (projective) transformation matrices $\mathbf{H}$ in **1D**:

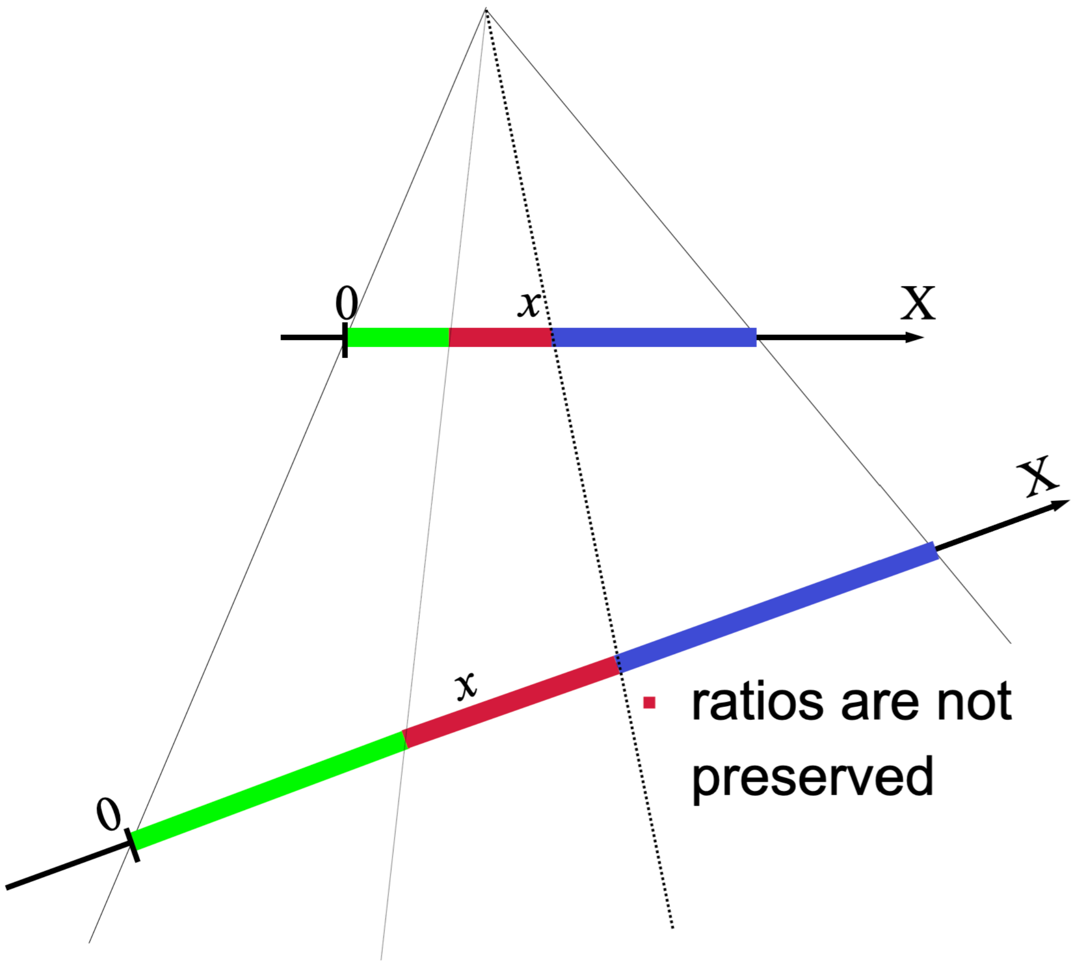
$$\mathbf{H} = \begin{pmatrix} a & b \\ c & 1 \end{pmatrix}$$

1D Projective transformations

- In $\mathbb{P}^1$ it holds







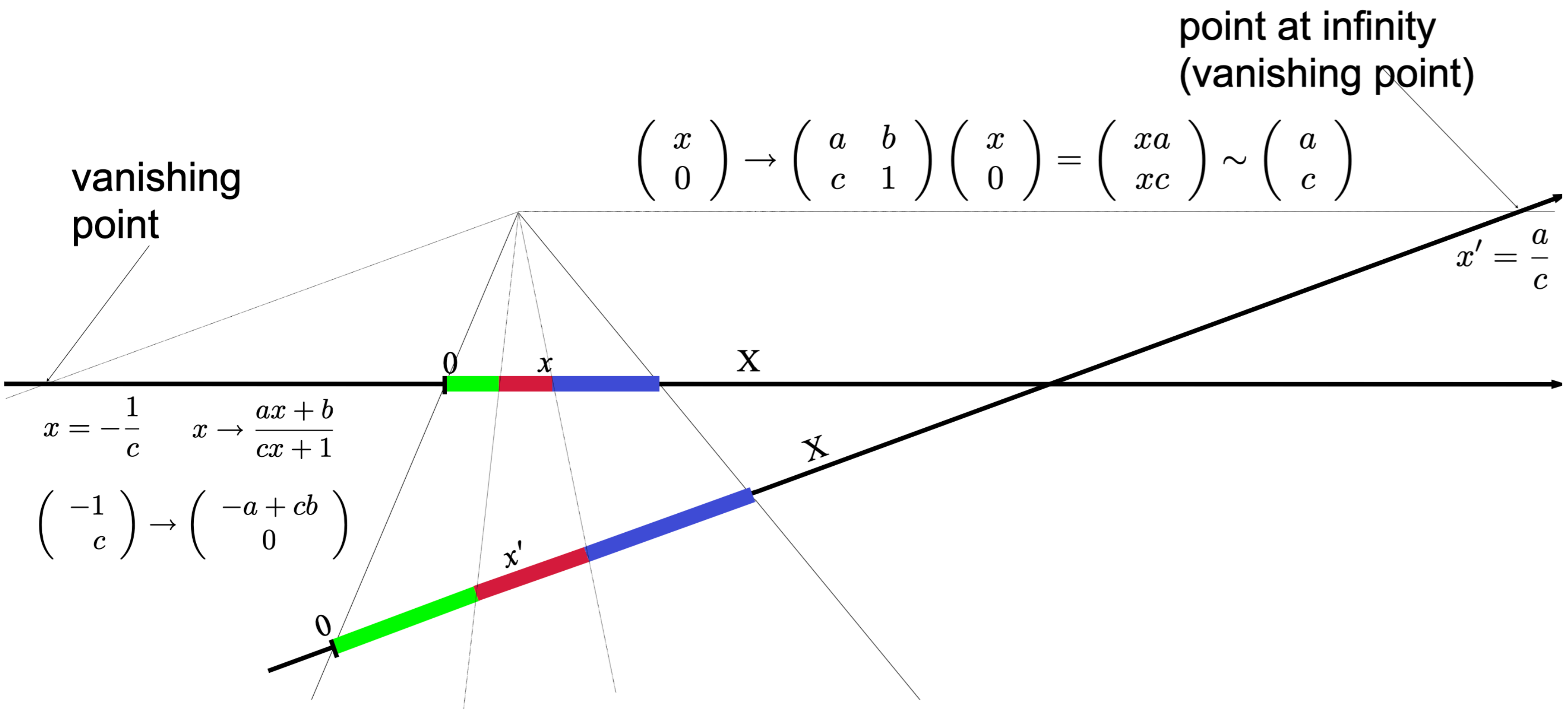
$$\begin{pmatrix} x \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} a & b \\ c & 1 \end{pmatrix} \begin{pmatrix} x \\ 1 \end{pmatrix}$$

$$x \rightarrow \frac{ax+b}{cx+1}$$

- ratios are not preserved
- $\mathbf{H}$ can be determined using three corresponding pairs of points (in general position)

Point at Infinity (Vanishing Point)

- In $\mathbb{P}^1$ it holds



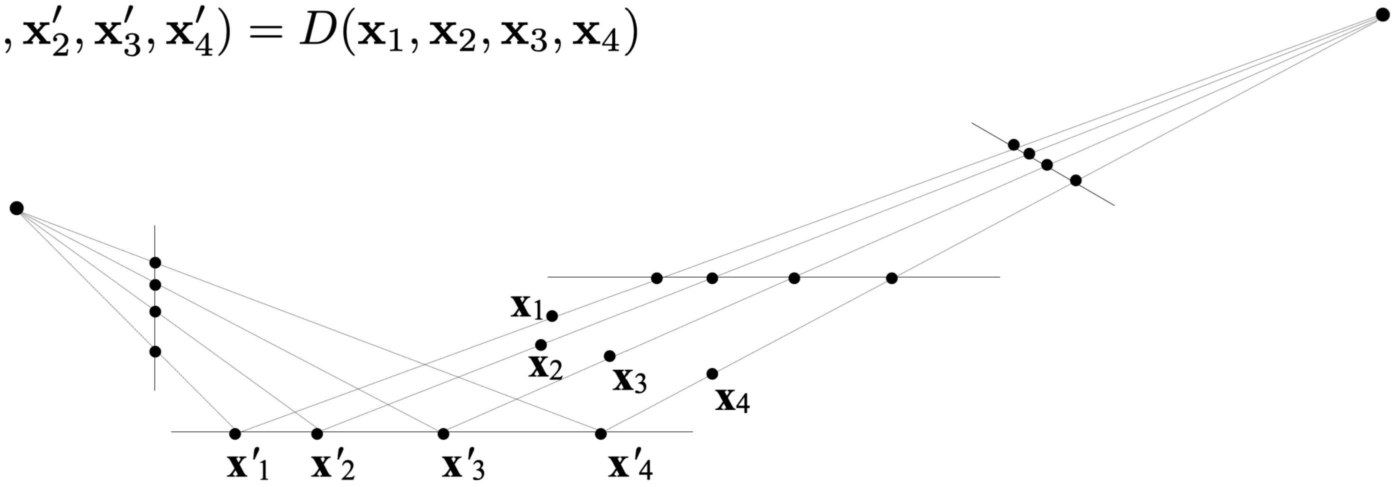
Projective 1D Geometry

- Homogeneous points in 1D: $\mathbf{x} = (x_1, x_2)^\top$
- Point transformations in 1D: $\mathbf{x}' = \mathbf{H}\mathbf{x}$ (3 DOF)
- The **cross-ratio**

$$D(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4) = \frac{|\mathbf{x}_1, \mathbf{x}_2|}{|\mathbf{x}_1, \mathbf{x}_3|} : \frac{|\mathbf{x}_2, \mathbf{x}_4|}{|\mathbf{x}_3, \mathbf{x}_4|} \quad \text{with} \quad |\mathbf{x}_i, \mathbf{x}_j| = \det \begin{pmatrix} x_{i1} & x_{j1} \\ x_{i2} & x_{j2} \end{pmatrix}$$

  remains **invariant** under projective mappings:

$$D(\mathbf{x}'_1, \mathbf{x}'_2, \mathbf{x}'_3, \mathbf{x}'_4) = D(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4)$$

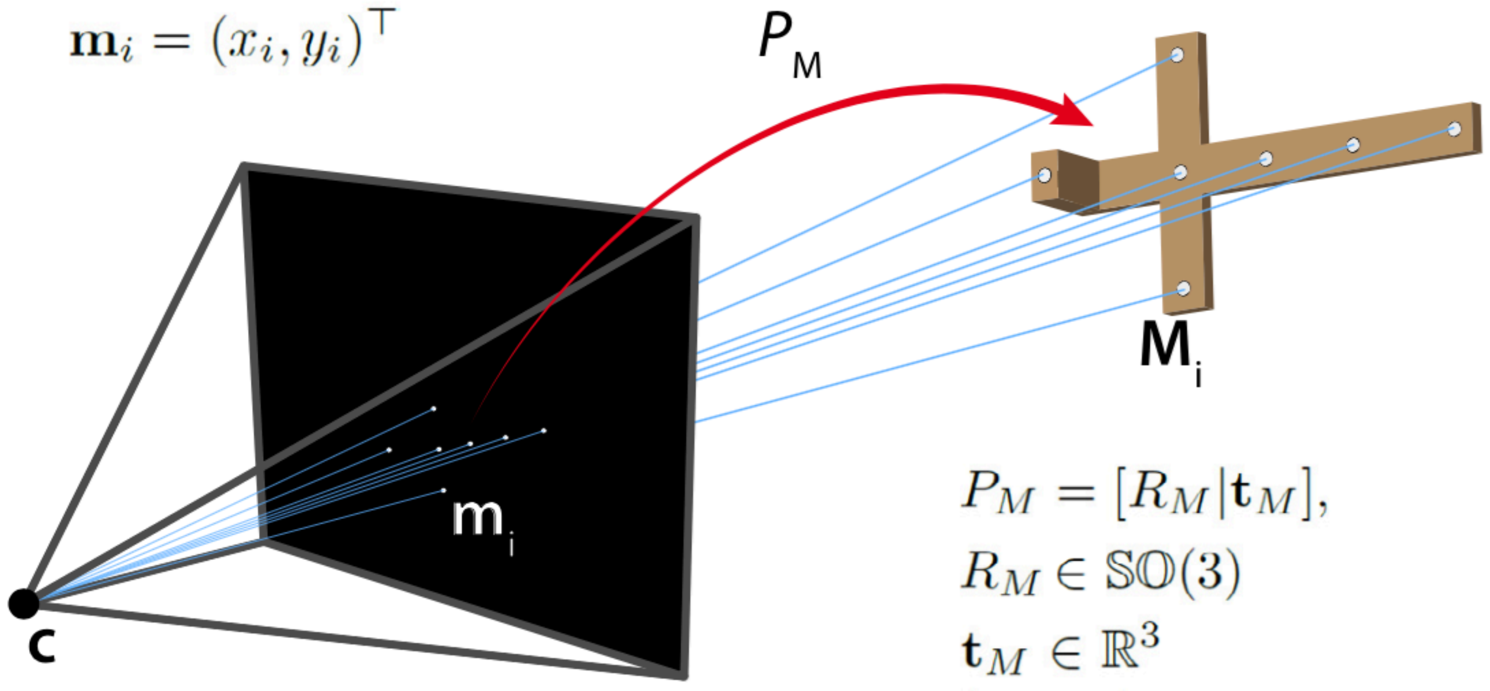


Cross Ratio Application Example

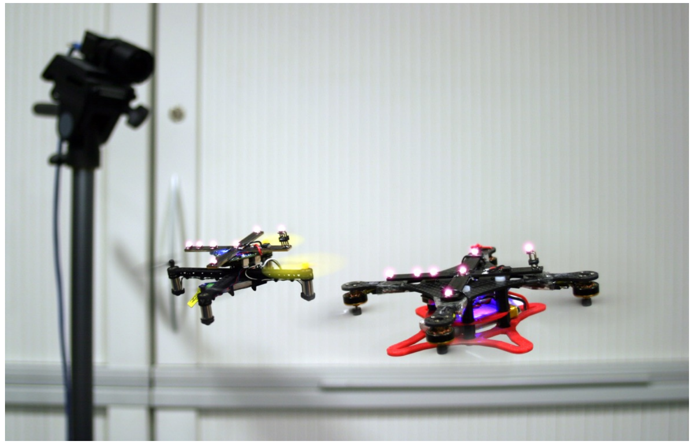
6DOF marker pose is estimated from 2D/3D-correspondences

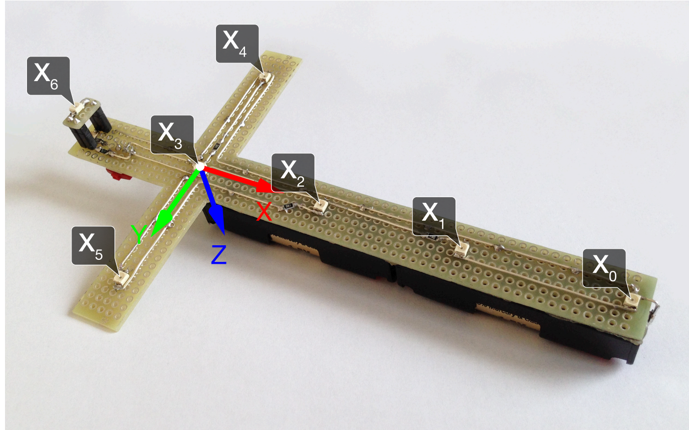
Offline pre-calibration of camera intrinsics including distortion

$$\mathbf{M}_i = (x_i, y_i, z_i)^\top$$
$$\mathbf{m}_i = (x_i, y_i)^\top$$



$$P_M = [R_M | \mathbf{t}_M],$$
$$R_M \in \mathbb{SO}(3)$$
$$\mathbf{t}_M \in \mathbb{R}^3$$





Tjaden, H.; Schwanecke, U.; Stein, F. A.; Schömer; E: High-Speed and Robust Monocular Tracking; 10th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAP), March 11-14, 2015

Primitives and Transformations

Points

Points in 1D/2D/3D can be written in **inhomogeneous coordinates** as

$$x \in \mathbb{R} \quad \text{or} \quad \mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2 \quad \text{or} \quad \mathbf{x} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \in \mathbb{R}^3$$

or in **homogenous coordinates** as

$$\tilde{\mathbf{x}} = \begin{pmatrix} \tilde{x} \\ \tilde{w} \end{pmatrix} \in \mathbb{P}^1 \quad \text{or} \quad \tilde{\mathbf{x}} = \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{w} \end{pmatrix} \in \mathbb{P}^2 \quad \text{or} \quad \tilde{\mathbf{x}} = \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ \tilde{w} \end{pmatrix} \in \mathbb{P}^3$$

where $\mathbb{P}^n = \mathbb{R}^{n+1} \setminus \{\mathbf{0}\}$ is called **projective space**. Homogeneous vectors that differ only by scale are considered equivalent and define an equivalence class, thus homogeneous vectors are **defined only up to scale**.

Points

An **inhomogeneous vector** $\mathbf{x}$ is converted to a **homogeneous vector** $\tilde{\mathbf{x}}$ as follows

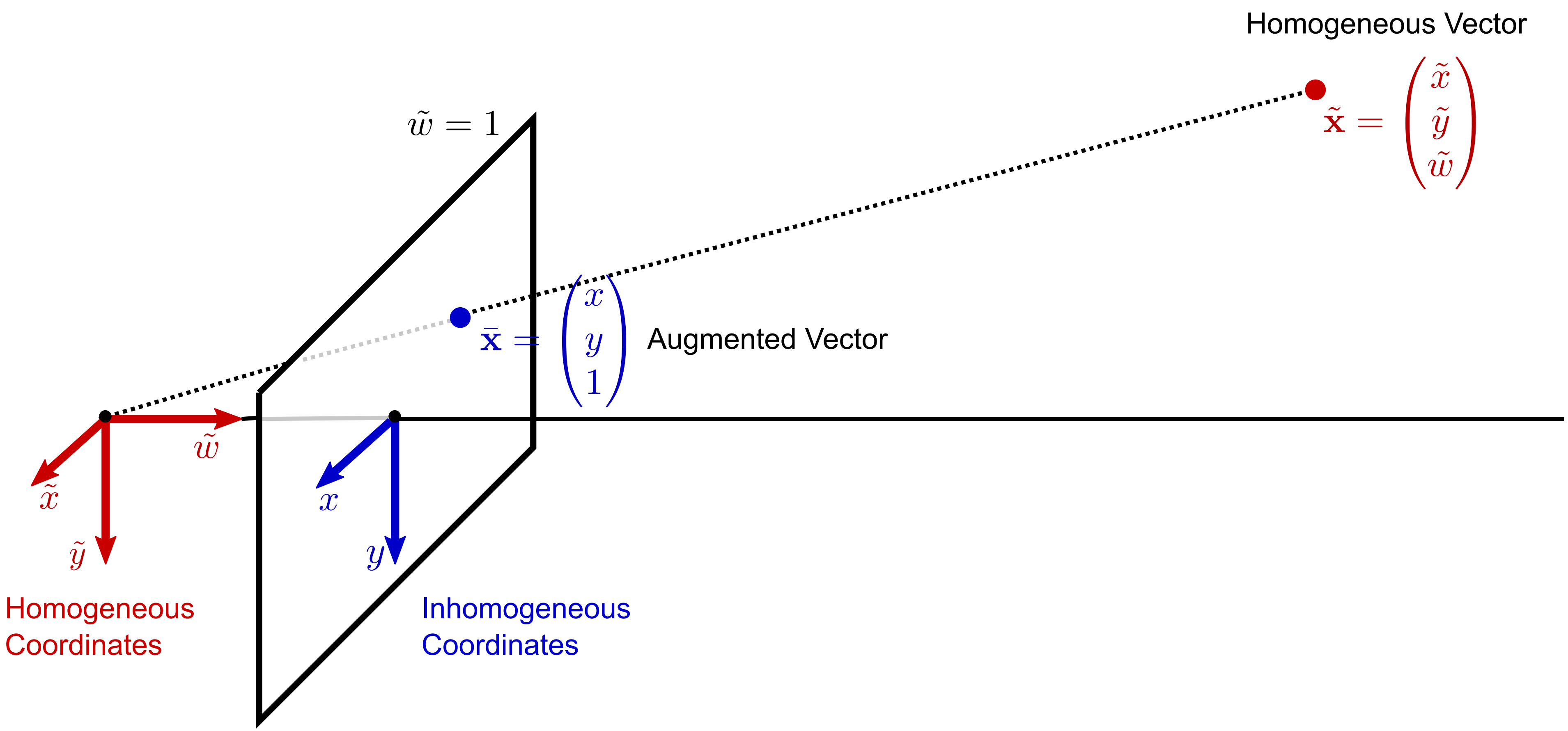
$$\tilde{\mathbf{x}} = \begin{pmatrix} \tilde{x} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} x \\ 1 \end{pmatrix} = \bar{x} \quad \text{or} \quad \tilde{\mathbf{x}} = \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \bar{\mathbf{x}} \quad \text{or} \quad \tilde{\mathbf{x}} = \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \bar{\mathbf{x}}$$

with the **augmented vector** $\bar{\mathbf{x}}$. To convert in opposite direction one has to divide by $\tilde{w}$:

$$\bar{\mathbf{x}} = \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \frac{1}{\tilde{w}} \tilde{\mathbf{x}} = \frac{1}{\tilde{w}} \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} \tilde{x}/\tilde{w} \\ \tilde{y}/\tilde{w} \\ \tilde{z}/\tilde{w} \\ 1 \end{pmatrix}$$

Homogeneous points whose last element is $\tilde{w} = 0$ can't be represented with inhomogeneous coordinates. They are called **ideal points** or **points at infinity**.

Example: 2D Points



2D Lines

2D lines can also be expressed using homogeneous coordinates $\tilde{\mathbf{l}} = (a, b, c)^\top$

$$\{\tilde{\mathbf{x}} \mid \tilde{\mathbf{l}}\tilde{\mathbf{x}} = 0\} \quad \Leftrightarrow \quad \{x, y \mid ax + by + c = 0\}$$

- We can normalize $\tilde{\mathbf{l}}$ so that $\tilde{\mathbf{l}} = (n_x, n_y, d)^\top = (\mathbf{n}, d)^\top$ with $\|\mathbf{n}\|_2 = 1$. In this case, $\mathbf{n}$ is the normal vector perpendicular to the line and d is its distance to the origin.
- An exception it the **line at infinity** $\tilde{\mathbf{l}}_\infty = (0, 0, 1)^\top$ which passes through all ideal points.

Cross Product

The **cross product** can be writen as the product of a **skew-symmetric matrix** and a vector:

$$\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times \mathbf{b} = \underbrace{\begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}}_{\text{skew-symmetric matrix } [\mathbf{a}]_\times, \text{ i.e. } [\mathbf{a}]_\times^\top = -[\mathbf{a}]_\times} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{pmatrix}$$

with

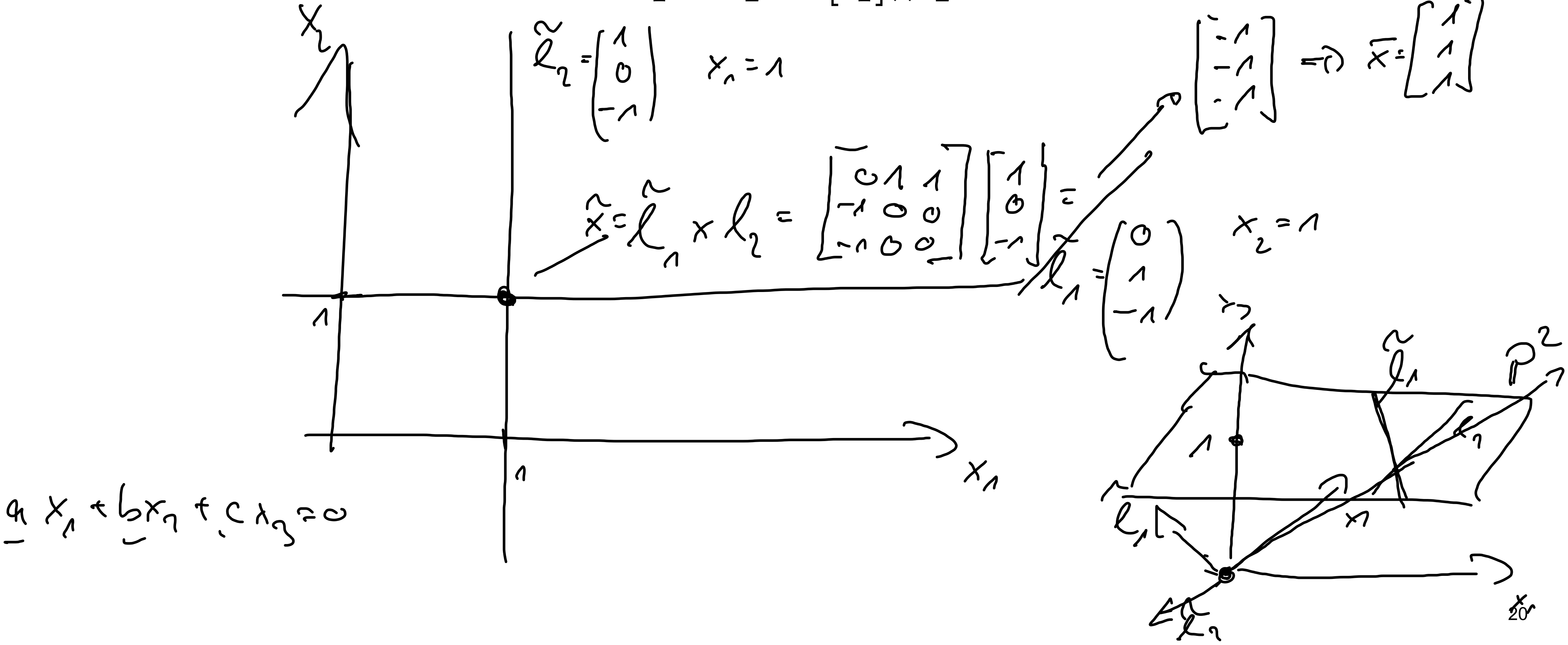
- $Rank([\mathbf{a}]_\times) = 2$
  - $[\mathbf{a}]_\times$ maps to the subspace that is perpendicular to $\mathbf{a}$
  - The null vector of $[\mathbf{a}]_\times$ is $\mathbf{a}$ itself, i.e. $[\mathbf{a}]_\times \cdot \mathbf{a} = \mathbf{0}$ and $\mathbf{a}^\top \cdot [\mathbf{a}]_\times = \mathbf{0}^\top$
- Repeated vector products: $[\mathbf{a}]_\times^2 = \mathbf{a} \cdot \mathbf{a}^\top - (\mathbf{a}^\top \cdot \mathbf{a}) \cdot \mathbf{I}$ and $[\mathbf{a}]_\times^3 = -(\mathbf{a}^\top \cdot \mathbf{a}) \cdot [\mathbf{a}]_\times$

- For $\|\mathbf{a}\| = 1$ it holds $[\mathbf{a}]_\times^3 = -[\mathbf{a}]_\times$

Intersection of two 2D Lines

In homogeneous coordinates, the **intersection** of two lines is given by

$$\tilde{\mathbf{x}} = \tilde{\mathbf{l}}_1 \times \tilde{\mathbf{l}}_2 = [\tilde{\mathbf{l}}_1]_\times \tilde{\mathbf{l}}_2$$

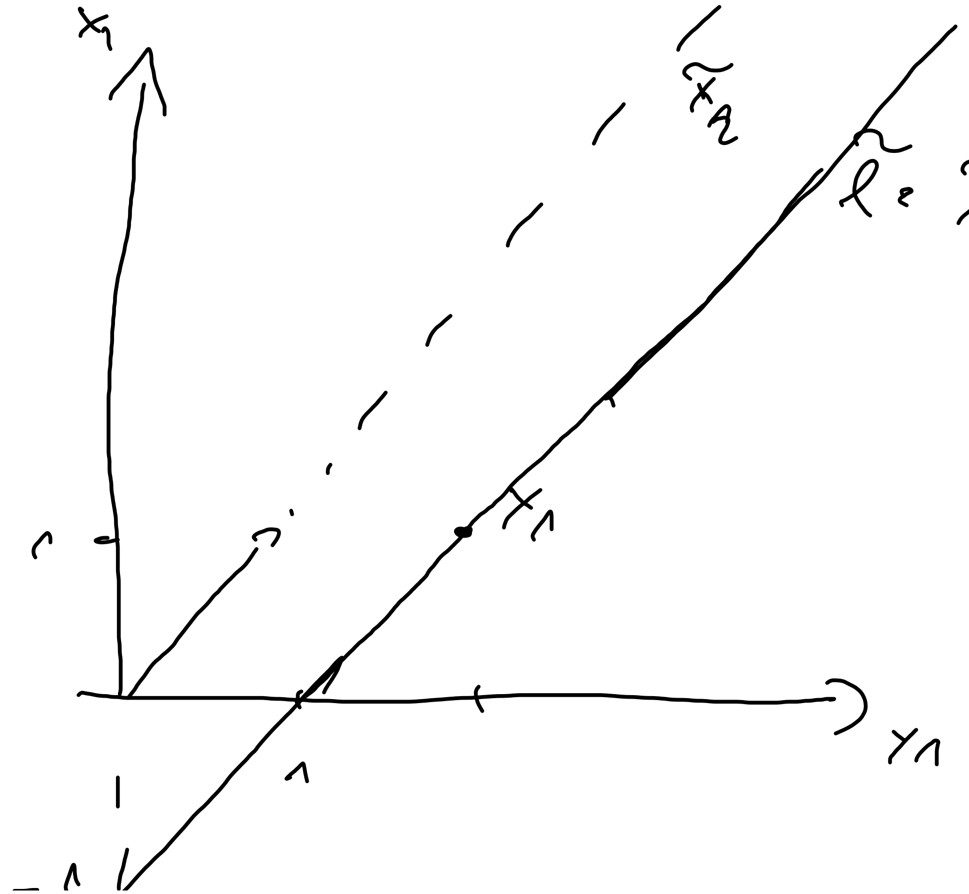


Line joining two points

In homogeneous coordinates, the **line joining two points** is also given by

$$\tilde{\mathbf{l}} = \tilde{\mathbf{x}}_1 \times \tilde{\mathbf{x}}_2 = [\tilde{\mathbf{x}}_1]_\times \tilde{\mathbf{x}}_2$$

$\tilde{x}_1 = \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} \quad \tilde{x}_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \cdot \lambda$



$\tilde{l} = \tilde{x}_1 \times x_2 = \begin{bmatrix} 0 & -1 & 1 \\ 1 & 0 & -2 \\ -1 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \\ 1 \end{bmatrix} \cdot \lambda$

$-1 x_1 + 1 \cdot x_2 + 1 = 0$

Vanishing Points and the Vanishing Line

- The intersection of parallel lines $\mathbf{l} = (a, b, c)^\top$ and $\mathbf{l}' = (a, b, c')^\top$ are points at infinity (vanishing points)

$$[\mathbf{l}]_\times \mathbf{l}' = \begin{pmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c' \end{pmatrix} = \begin{pmatrix} b(c' - c) \\ a(c - c') \\ 0 \end{pmatrix} \sim \begin{pmatrix} b \\ -a \\ 0 \end{pmatrix}$$

  - Vanishing points: $(x_1, x_2, 0)^\top$
  - Vanishing line: $\mathbf{l}_\infty = (0, 0, 1)^\top$
- An alternative definition of $\mathbb{P}^2$ is $\mathbb{P}^2 = \mathbb{R}^2 \cup \mathbf{l}_\infty$
  - Note that $\mathbb{P}^2$ does not distinguish between vanishing points and other points

Duality

The **duality principle**:

To every proposition of two-dimensional projective geometry exists a ***dual proposition*** *which is obtained by interchanging the role of points and lines in the original proposition.*

- For example, dual are

$$\mathbf{x} \longleftrightarrow \mathbf{l}$$

$$\mathbf{x}^T\mathbf{l} = 0 \longleftrightarrow \mathbf{l}^T\mathbf{x} = 0$$

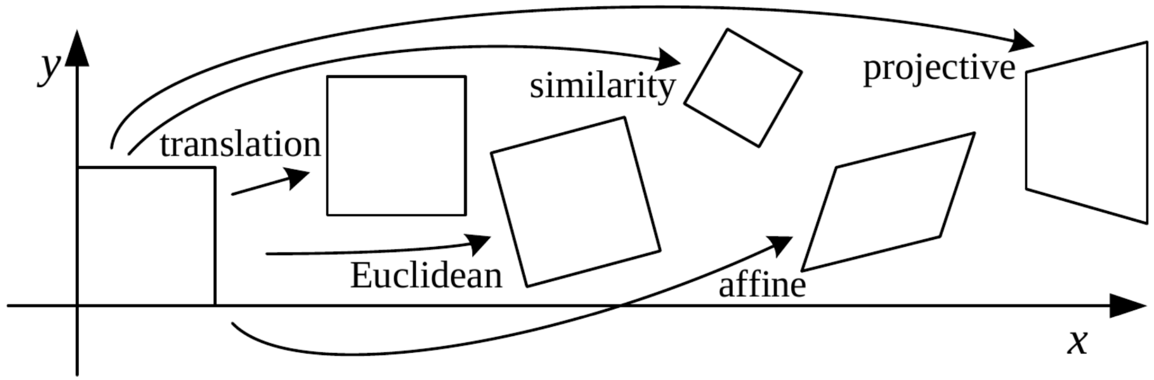
$$\mathbf{x} = \mathbf{l} \times \mathbf{l}' \longleftrightarrow \mathbf{l} = \mathbf{x} \times \mathbf{x}'$$

2D Transformations

Translation (2 DoF):

- 2D Translation of the Input

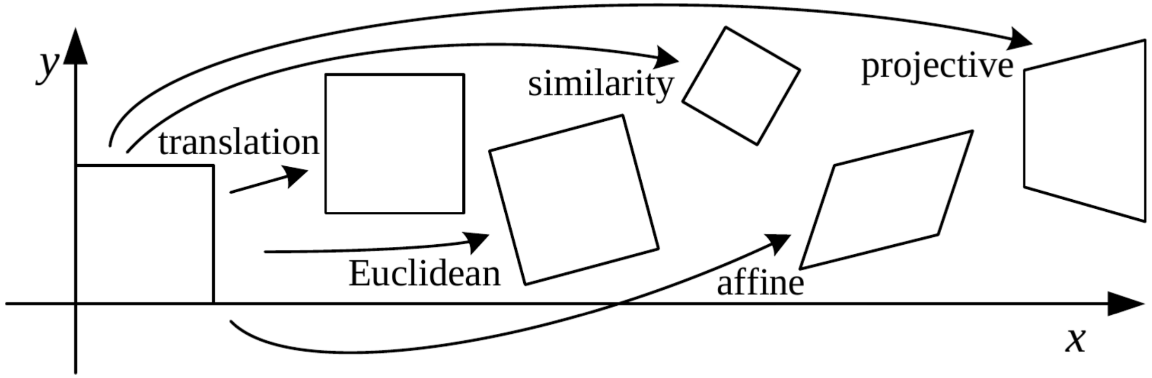
$$\mathbf{x}' = \mathbf{x} + \mathbf{t} \quad \Leftrightarrow \quad \bar{\mathbf{x}}' = \begin{bmatrix} \mathbf{I} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \bar{\mathbf{x}}$$



- Using homogeneous representations allows to easily chain/invert transformations
- Augmented vectors $\bar{\mathbf{x}}$ can always be replaced by general homogeneous vectors $\tilde{\mathbf{x}}$

2D Transformations

Euclidean (Ridig, 3 DoF):



- 2D Translation + 2D Rotation

$$\mathbf{x}' = \mathbf{R}\mathbf{x} + \mathbf{t} \quad \Leftrightarrow \quad \bar{\mathbf{x}}' = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \bar{\mathbf{x}}$$

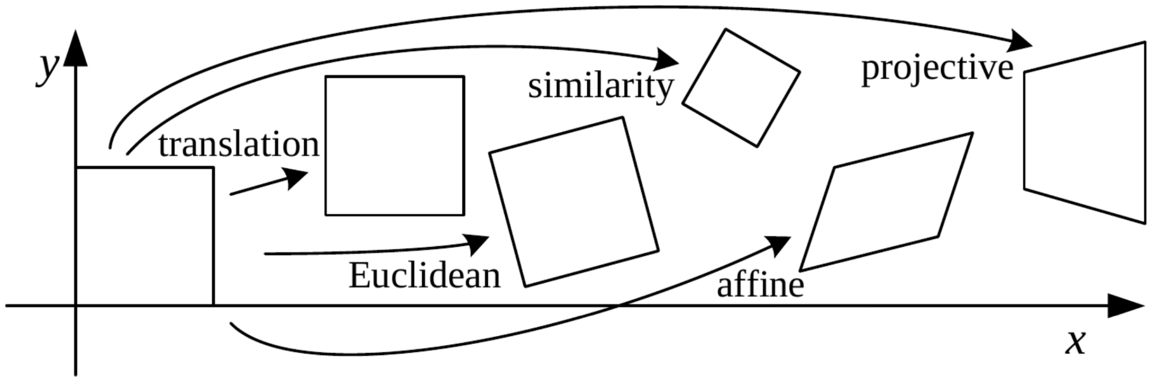
- $\mathbf{R} \in SO(2)$ is an orthonormal matrix with $\mathbf{R}\mathbf{R}^\top = \mathbf{I}$ and $\det(\mathbf{R}) = 1$ (rotation matrix)
- Euclidean transformations preserve Euclidean distances and angles

2D Transformations

Similarity (4 DoF):

- 2D Translation + Scaled 2D Rotation

$$\mathbf{x}' = s\mathbf{R}\mathbf{x} + \mathbf{t} \quad \Leftrightarrow \quad \bar{\mathbf{x}}' = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \bar{\mathbf{x}}$$



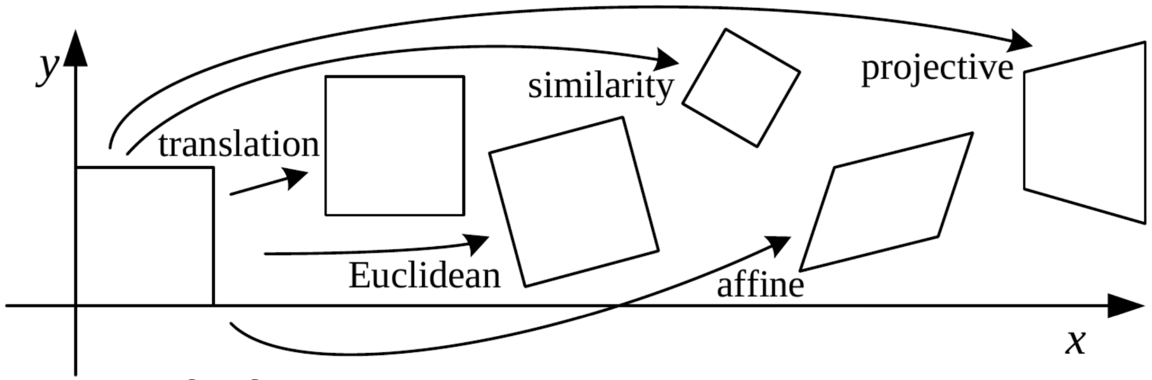
- $\mathbf{R} \in SO(2)$ is rotation matrix and s an arbitrary scale factor
- Similarity transformations preserve angles

2D Transformations

Affine (6 DoF):

- 2D Linear Transformation

$$\mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{t} \quad \Leftrightarrow \quad \bar{\mathbf{x}}' = \begin{bmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \bar{\mathbf{x}}$$



- $\mathbf{A} \in \mathbb{R}^{2\times 2}$ is an arbitrary 2×2 matrix

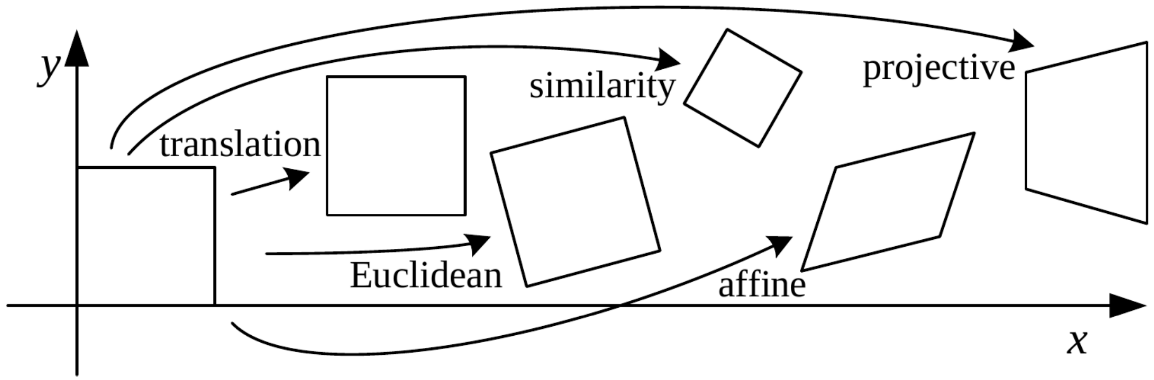
- Parallel lines remain parallel under affine transformations

2D Transformations

Perspective (Homography, 8 DoF):

- 2D Linear Transformation

$$\tilde{\mathbf{x}}' = \tilde{\mathbf{H}}\tilde{\mathbf{x}} \qquad \left(\bar{\mathbf{x}} = \frac{1}{\tilde{w}}\tilde{\mathbf{x}}\right)$$



- $\tilde{\mathbf{H}} \in \mathbb{R}^{3\times 3}$ is an arbitrary homogeneous 3×3 matrix (i.e. specified only up to scale)
- Perspective transformations preserve straight lines

Example: Mapping between planes

- A central projection of on plane onto another can be represented by a homography, i.e. $\tilde{\mathbf{x}}' = \tilde{\mathbf{H}}\tilde{\mathbf{x}}$

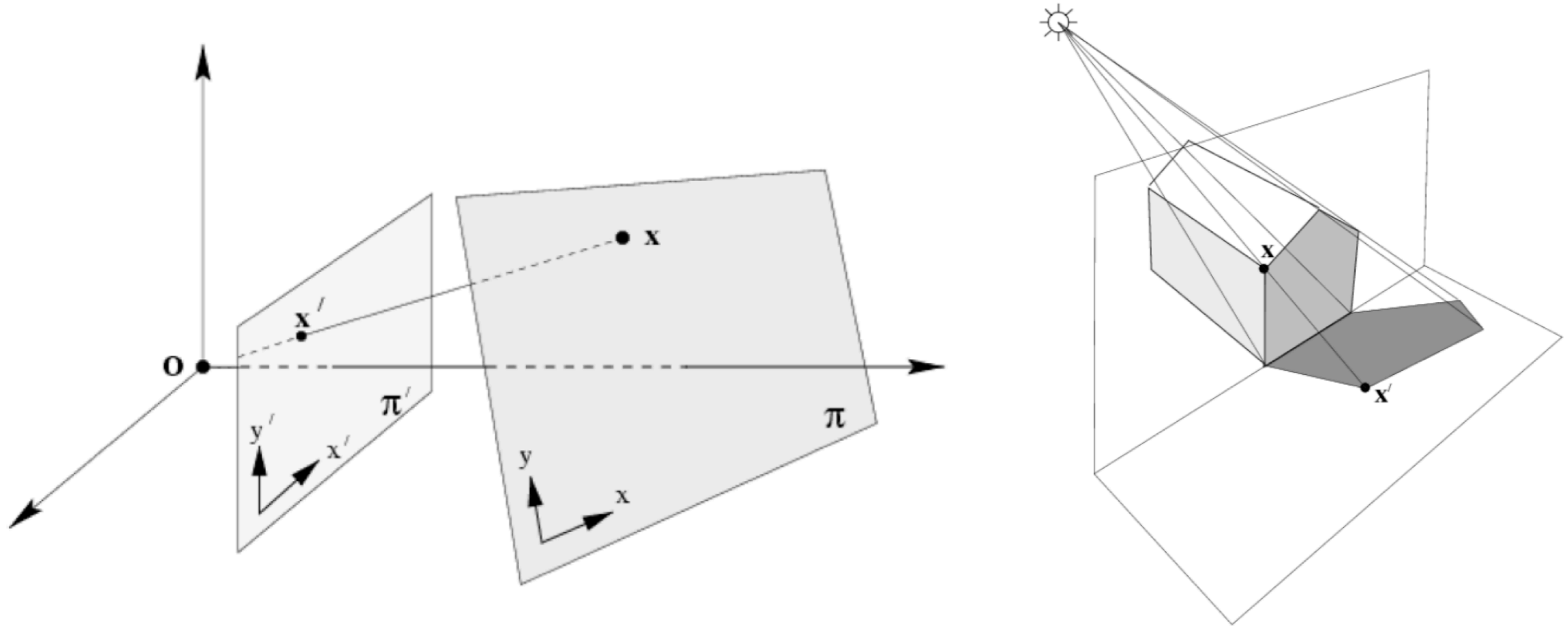
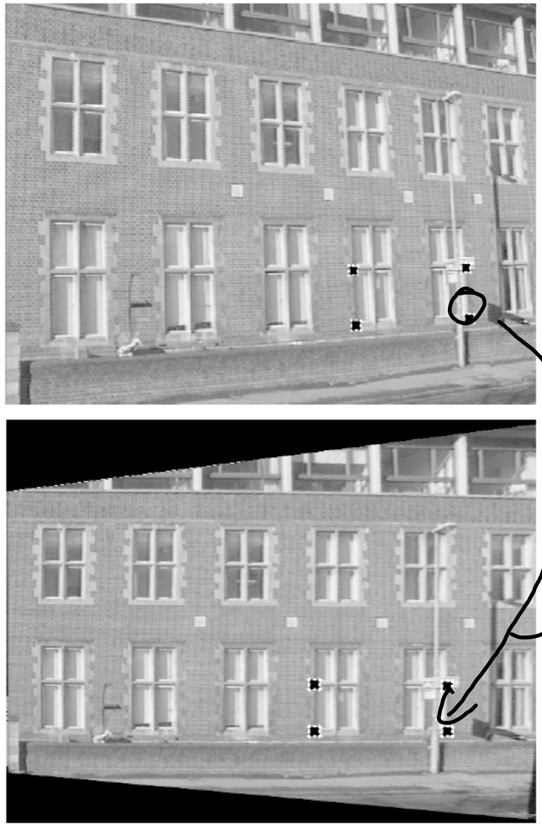


image: Hartley & Zisserman, Multiple View Geometry

Example: Removal of Projective Distortions

- The transformation that maps points from a known plane to their current (true) coordinates removes the projective distortion for all points of the same plane



images: Hartley & Zisserman, Multiple View Geometry

$$x' = \frac{x'_1}{x'_3} = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}, \quad y' = \frac{x'_2}{x'_3} = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}$$

$$x'(h_{31}x + h_{32}y + h_{33}) = h_{11}x + h_{12}y + h_{13}$$

$$y'(h_{31}x + h_{32}y + h_{33}) = h_{21}x + h_{22}y + h_{23}$$

(linear in h_{ij})

- ➡ 2 conditions for each pair of points
- ➡ 8DOF

⇨ You need **four** points

$$p \overset{H}{\longmapsto} p'$$

$$H \begin{pmatrix} p_x \\ p_y \\ 1 \end{pmatrix} = \begin{pmatrix} p'_x \\ p'_y \\ p'_w \end{pmatrix}$$

2

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix} = \begin{bmatrix} p'_x \\ p'_y \\ p'_w \end{bmatrix}$$

$$p'_x = h_{11} p_x + h_{12} p_y + h_{13}$$

$$p'_y =$$

Direct Linear Transformation for Homography Estimation

Estimate a homography from a set of 2D correspondences:

Let $\mathcal{X} = \{\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}'_i\}_{i=1}^N$ denote a set of N 2D-to-2D correspondences related by $\tilde{\mathbf{x}}'_i = \tilde{\mathbf{H}}\tilde{\mathbf{x}}_i$

- As the correspondence vectors are homogeneous, they have the same direction but may differ in magnitude. Thus the correspondences are related by $\tilde{\mathbf{x}}'_i \times \tilde{\mathbf{H}}\tilde{\mathbf{x}}_i = \mathbf{0}$
- Using $\tilde{\mathbf{h}}_k^\top$ to denote the k'th row of $\tilde{\mathbf{H}}$, this can be rewritten as a linear equation in $\tilde{\mathbf{h}}$:

$$\underbrace{\begin{bmatrix} \mathbf{0}^\top & -\tilde{w}'_i\tilde{\mathbf{x}}_i^\top & \tilde{y}'_i\tilde{\mathbf{x}}_i^\top \\ \tilde{w}'_i\tilde{\mathbf{x}}_i^\top & \mathbf{0}^\top & -\tilde{x}'_i\tilde{\mathbf{x}}_i^\top \\ -\tilde{y}'_i\tilde{\mathbf{x}}_i^\top & \tilde{x}'_i\tilde{\mathbf{x}}_i^\top & \mathbf{0}^\top \end{bmatrix}}_{\mathbf{A}_i} \underbrace{\begin{bmatrix} \tilde{\mathbf{h}}_1 \\ \tilde{\mathbf{h}}_2 \\ \tilde{\mathbf{h}}_3 \end{bmatrix}}_{\tilde{\mathbf{h}}} = \mathbf{0}$$

  - The last row is linearly dependent (up to scale) on the first two and can be dropped.

Direct Linear Transformation for Homography Estimation

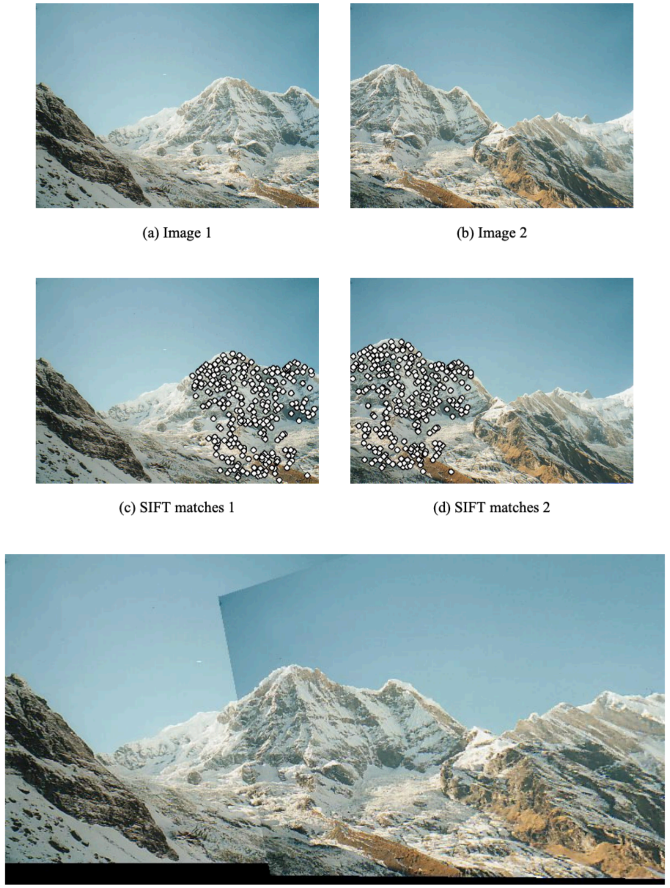
Each point correspondence yields two equations. Stacking all equations into a $2N \times 9$ dimensional matrix $\mathbf{A}$ leads to the following **constrained least squares problem**

$$\begin{aligned}\tilde{\mathbf{h}}^* &= \text{argmin}_{\tilde{\mathbf{h}}} \|\mathbf{A}\tilde{\mathbf{h}}\|_2^2 + \lambda(\|\tilde{\mathbf{h}}\|_2^2 - 1) \\ &= \text{argmin}_{\tilde{\mathbf{h}}} \tilde{\mathbf{h}}^\top \mathbf{A}^\top \mathbf{A}\tilde{\mathbf{h}} + \lambda(\tilde{\mathbf{h}}^\top \tilde{\mathbf{h}} - 1)\end{aligned}$$

- $\tilde{\mathbf{h}}$ can be fixed to $\|\tilde{\mathbf{h}}\|_2^2 = 1$ as $\tilde{\mathbf{H}}$ is homogeneous (defined up to scale) and the trivial solution $\tilde{\mathbf{h}} = 0$ is not of interest
- The solution to this optimization problem is the **singular vector** corresponding to the **smallest singular value** of $\mathbf{A}$ (the last column of $\mathbf{V}$ when decomposing $\mathbf{A} = \mathbf{UDV}^\top$)
- The resulting algorithm is called **Direct Linear Transformation**

Applications

Image Stitching



(a) Image 1

(b) Image 2

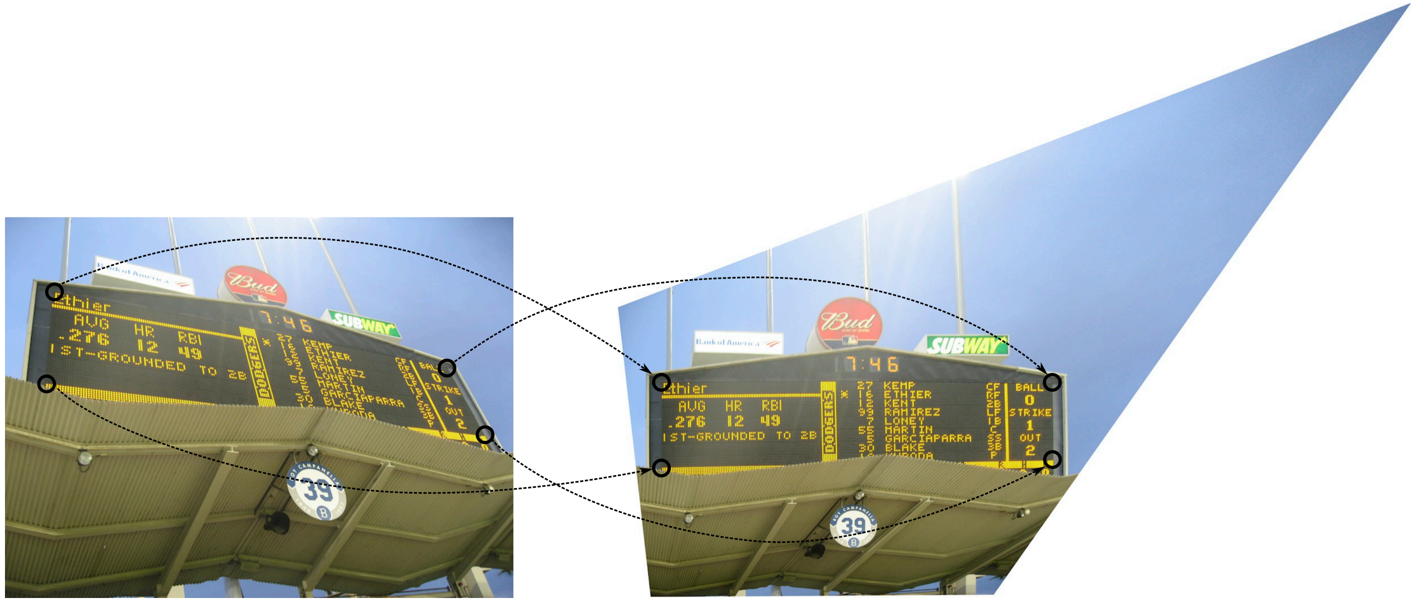
(c) SIFT matches 1

(d) SIFT matches 2

(e) Images aligned according to a homography

[M. Brown, D. Lowe, Recognising Panoramas, ICCV 2003]

Perspective Distortion Correction



2D Transformations on Co-Vectors

If a point $\tilde{\mathbf{x}}$ is transformed by a perspective 2D transformation $\tilde{\mathbf{H}}$ as

$$\tilde{\mathbf{x}}' = \tilde{\mathbf{H}}\tilde{\mathbf{x}}$$

for a transformed 2D line it must hold

$$0 = \tilde{\mathbf{l}}'^\top \tilde{\mathbf{x}}' = \tilde{\mathbf{l}}'^\top \tilde{\mathbf{H}}\tilde{\mathbf{x}} = (\tilde{\mathbf{H}}^\top \tilde{\mathbf{l}}')^\top \tilde{\mathbf{x}} = \tilde{\mathbf{l}}^\top \tilde{\mathbf{x}}$$

and therefore

$$\tilde{\mathbf{l}}' = \tilde{\mathbf{H}}^{-\top} \tilde{\mathbf{l}}$$

Thus, the **action of a projective transformation on a co-vector** such as a 2D line or 3D plane can be represented by the **transposed inverse** of the matrix.

Overview of 2D Transformations

Transformation	Matrix	# DOF	Preserves
translation	$\begin{bmatrix}\mathbf{I} & \mathbf{t}\end{bmatrix}_{2\times 3}$	2	orientation
rigid (Euclidean)	$\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}_{2\times 3}$	3	length
similarity	$\begin{bmatrix}s\mathbf{R} & \mathbf{t}\end{bmatrix}_{2\times 3}$	4	angles
affine	$\begin{bmatrix}\mathbf{A} & \mathbf{t}\end{bmatrix}_{2\times 3}$	6	parallelism
projective	$\begin{bmatrix}\mathbf{H}\end{bmatrix}_{3\times 3}$	8	straight lines

- Transformations form **nested set of groups** (closed under composition, inverse)
- 2×3 matrices are extended with a third $[\mathbf{0}^\top 1]$ row for homogeneous transforms

Effect of 2D Transformations on the Vanishing Line

- Affine transformation

$$\begin{bmatrix} \mathbf{A} & \mathbf{T} \\ \mathbf{0}^\top & v \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{A} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ 0 \end{pmatrix}$$

  - Vanishing line remains at infinity
  - Points moving on the vanishing line

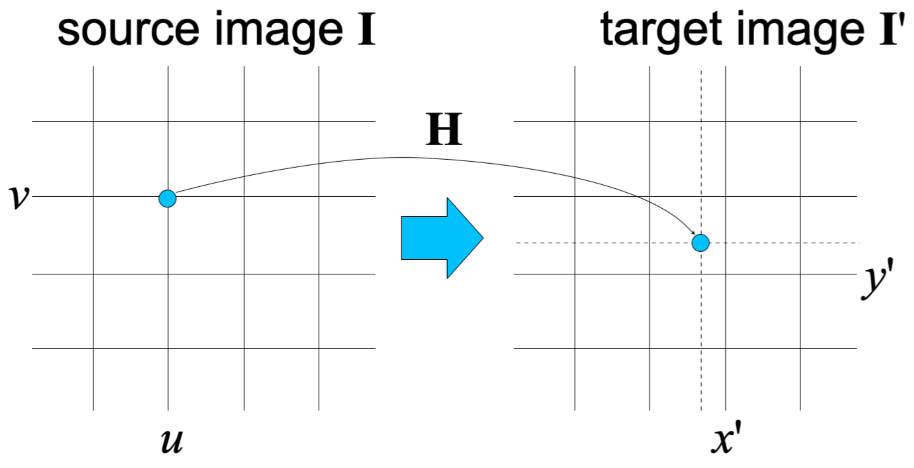
- Projective transformations

$$\begin{bmatrix} \mathbf{A} & \mathbf{T} \\ \mathbf{v}^\top & v \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{A} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \\ v_1 x_1 + v_2 x_2 \end{pmatrix}$$

  - Vanishing line becomes finite
  - Vanishing points (horizon) can be observed

Applying a (projective) Mapping $\mathbf{H}$ to an Image $\mathbf{I}$

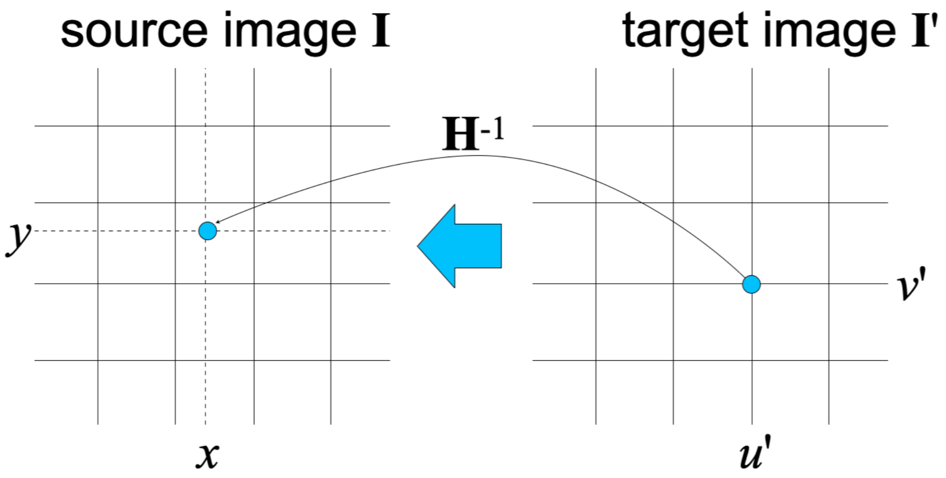
- **Source-to-Target Mapping**
  - For each point (pixel) (u, v) in the source image $\mathbf{I}$, calculate the transformed position $(x_1, x_2, x_3)^\top = \mathbf{H}(u, v, 1)^\top$ in the target image $\mathbf{I}'$
  - Transformed positions usually do not fall on grid points
  - Not all elements in the target image are hit exactly ones
    - Gaps can occur (enlargement) or information is lost (several target pixels are “hit” by the same source pixel)



Applying a (projective) Mapping $\mathbf{H}$ to an Image $\mathbf{I}$

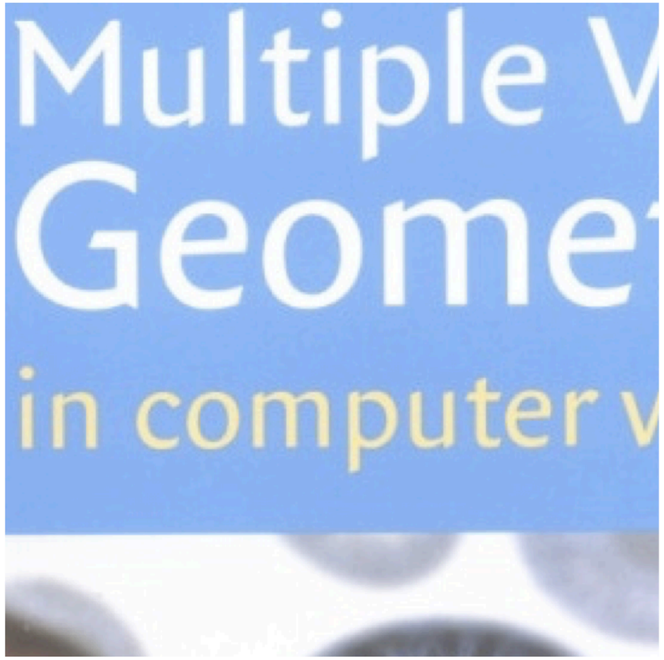
- **Target-to-Source Mapping**
  - For each point (pixel) (u', v') in the target image $\mathbf{I}'$, calculate the corresponding position $(x_1, x_2, x_3)^\top = \mathbf{H}^{-1}(u', v', 1)^\top$ in the source image $\mathbf{I}$
  - *Interpolate* the corresponding intensities $\mathbf{I}'(u', v')$
    - All pixel values of the target image are evaluated exactly once
  - To apply $\mathbf{x}' = \mathbf{H}\mathbf{x}$ on the image $\mathbf{I}$ you need $\mathbf{H}^{-1}$

```python
def transform_image(s_img, H):
    """ s_img : source image
        H     : (projective) mapping """
    h, w = s_img.shape
    t_img = np.zeros((h,w)) # make empty target image
    for t_pos in product(range(h), range(w)):
        s_pos =  np.linalg.inv(H) @ np.array(t_pos+(1,))
        t_img.item(t_pos) = interpolate_value(s_img, s_pos)
    return t_img
```

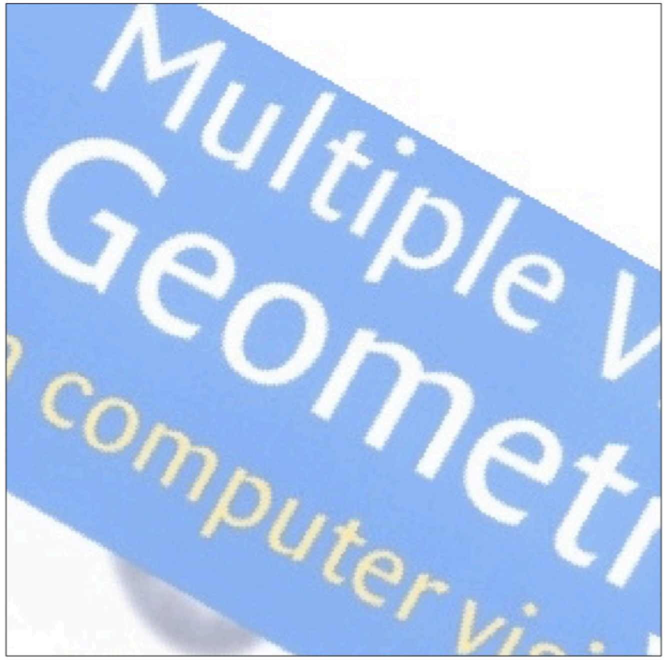


Aliasing when Applying (Projective) Mappings to Images

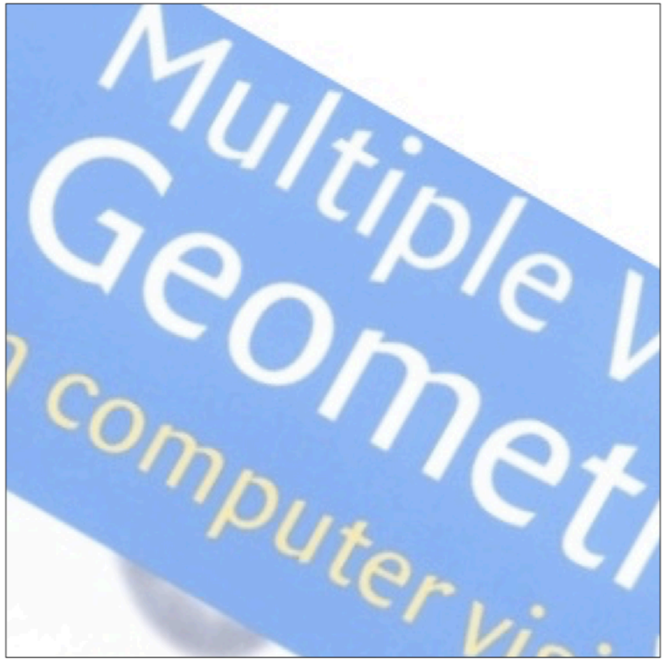
- When applying (projective) mappings to images e.g. the (u, v) **coordinates of the original image I has to be rounded**, to calculate the corresponding gray values
  - Rounding leads to (unwanted) step effects (**aliasing**)



original image



without interpolation
(or next-neighbour interpolation)



with bilinearer
interpolation

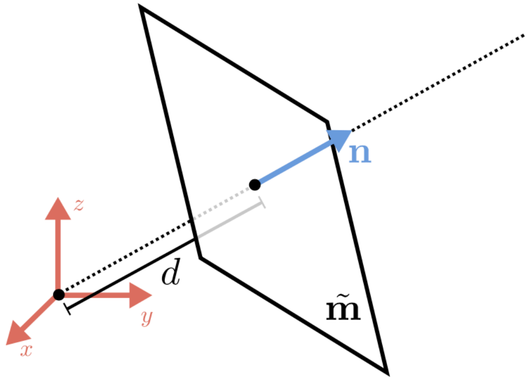
Next Lecture

3D Planes

3D planes can also be represented with homogeneous coordinates $\tilde{\mathbf{m}} = (a, b, c, d)^\top$ as it holds

$$\{\tilde{\mathbf{x}} \mid \tilde{\mathbf{m}}^\top \bar{\mathbf{x}} = 0\} \quad \Leftrightarrow \quad \{x, y, z \mid ax + by + cz + d = 0\}$$

- As for 2D lines $\tilde{\mathbf{m}}$ can be **normalized** so that $\tilde{\mathbf{m}} = (n_x, n_y, n_z, d)^\top = (\mathbf{n}, d)^\top$, $\|\mathbf{n}\|_2 = 1$
  - $\mathbf{n}$ is the normal for the plane and d its distance to the origin



$$\underbrace{\begin{pmatrix} n_x & n_y & n_z & d \end{pmatrix}}_{\tilde{\mathbf{m}}^T} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = 0$$

- An exception is the **plane at infinity** $\tilde{\mathbf{m}} = (0, 0, 0, 1)^\top$ which passes through all ideal points (=points at infinity) for which $\tilde{w} = 0$

Overview of 3D Transformations

Transformation	Matrix	# DOF	Preserves
translation	$\begin{bmatrix} \mathbf{I} & \mathbf{t} \end{bmatrix}_{3\times 4}$	3	orientation
rigid (Euclidean)	$\begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}_{3\times 4}$	6	length
similarity	$\begin{bmatrix} s\mathbf{R} & \mathbf{t} \end{bmatrix}_{3\times 4}$	7	angles
affine	$\begin{bmatrix} \mathbf{A} & \mathbf{t} \end{bmatrix}_{3\times 4}$	12	parallelism
projective	$\begin{bmatrix} \mathbf{H} \end{bmatrix}_{4\times 4}$	15	straight lines

- 3D transformations are defined analogously to 2D transformations
- 3×4 matrices are extended with a fourth $[\mathbf{0}^\top 1]$ row for homogeneous transforms

Quiz

Vector Stuff

When are two vectors $\vec{a}$ and $\vec{b}$ perpendicular?

A: If $\vec{a} \times \vec{b} = 0.$

B: If $\vec{a} \cdot \vec{b} \neq 0.$

C: If $\vec{a} \times \vec{b} \neq 0.$

D: If $\vec{a} \cdot \vec{b} = 0.$

2D Lines and Points Stuff

Give the homogenous vector for line through the points $\vec{x} = (1, 2)^\top$ and $\vec{y} = (3, 4)^\top$.

A: $(1, -1, 1)^\top$

B: $(1, 2, 3)^\top$

C: $(2, 3, 4)^\top$

D: $(-2, 2, -2)^\top$

2D Lines and Points Stuff

Give the homogenous vector for the intersection of the lines $x - y + 1 = 0$ and $x - y - 1 = 0$.

A: $(2, 2, 0)^\top$

B: $(1, 2, 3)^\top$

C: $(1, 1, 0)^\top$

D: $(-2, 2, -2)^\top$

Transformation Stuff

A point is transformed by $\begin{pmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$. What is the proper transformation for lines?

A: $\begin{pmatrix} \frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$

B: $\begin{pmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

C: $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix}$

D: $\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{3} \end{pmatrix}$