3D Computer Vision

Radim Šára Martin Matoušek

Center for Machine Perception
Department of Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague

https://cw.fel.cvut.cz/wiki/courses/tdv/start

http://cmp.felk.cvut.cz
mailto:sara@cmp.felk.cvut.cz
phone ext. 7203

rev. November 9, 2021

Open Informatics Master's Course

Module II

Perspective Camera

covered by

[H&Z] Secs: 2.1, 2.2, 3.1, 6.1, 6.2, 8.6, 2.5, Example: 2.19

▶Basic Geometric Entities, their Representation, and Notation

- entities have names and representations
- names and their components:

entity	in 2-space	in 3-space
point	$m = (u, v)$	$X = (x, y, z)$
line	n	O
plane		π, φ

- associated vector representations

$$\mathbf{m} = \begin{bmatrix} u \\ v \end{bmatrix} = [u, v]^\top, \quad \mathbf{X} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \mathbf{n}$$

will also be written in an 'in-line' form as $\mathbf{m} = (u, v)$, $\mathbf{X} = (x, y, z)$, etc.

- vectors are always meant to be columns $\mathbf{x} \in \mathbb{R}^{n \times 1}$
- associated homogeneous representations

$$\underline{\mathbf{m}} = [m_1, m_2, m_3]^\top, \quad \underline{\mathbf{X}} = [x_1, x_2, x_3, x_4]^\top, \quad \underline{\mathbf{n}}$$

'in-line' forms: $\underline{\mathbf{m}} = (m_1, m_2, m_3)$, $\underline{\mathbf{X}} = (x_1, x_2, x_3, x_4)$, etc.

- matrices are $\mathbf{Q} \in \mathbb{R}^{m \times n}$, linear map of a $\mathbb{R}^{n \times 1}$ vector is $\mathbf{y} = \mathbf{Q}\mathbf{x}$
- j-th element of vector $\mathbf{m}_i$ is $(\mathbf{m}_i)_j$; element i, j of matrix $\mathbf{P}$ is $\mathbf{P}_{ij}$

▶Image Line (in 2D)

a finite line in the 2D (u, v) plane $\qquad a\,u + b\,v + c = 0$

has a parameter (homogeneous) vector $\qquad \boxed{\underline{\mathbf{n}} \simeq (a,\, b,\, c)}, \quad \|\underline{\mathbf{n}}\| \neq 0$

and there is an equivalence class for $\lambda \in \mathbb{R}$, $\lambda \neq 0$ $\qquad (\lambda a,\, \lambda b,\, \lambda c) \simeq (a,\, b,\, c)$

'Finite' lines

- standard representative for finite $\underline{\mathbf{n}} = (n_1, n_2, n_3)$ is $\lambda\underline{\mathbf{n}}$, where $\lambda = \frac{\mathbf{1}}{\sqrt{n_1^2 + n_2^2}}$

assuming $n_1^2 + n_2^2 \neq 0$; $\mathbf{1}$ is the unit, usually $\mathbf{1} = 1$

'Infinite' line

- we augment the set of lines for a special entity called the **line at infinity** (ideal line)

$$\underline{\mathbf{n}}_\infty \simeq (0, 0, \mathbf{1}) \qquad \text{(standard representative)}$$

- the set of equivalence classes of vectors in $\mathbb{R}^3 \setminus (0, 0, 0)$ forms the projective space $\mathbb{P}^2$

a set of rays →21

- line at infinity is a proper member of $\mathbb{P}^2$
- I may sometimes wrongly use $=$ instead of $\simeq$, if you are in doubt, ask me

▶Image Point

Finite point $\mathbf{m} = (u, v)$ is incident on a finite line $\underline{\mathbf{n}} = (a, b, c)$ iff   iff = works either way!

$$a\,u + b\,v + c = 0$$

can be rewritten as (with scalar product): $(u, v, \mathbf{1}) \cdot (a, b, c) = \underline{\mathbf{m}}^\top \underline{\mathbf{n}} = 0$

'Finite' points

- a finite point is also represented by a homogeneous vector $\boxed{\underline{\mathbf{m}} \simeq (u, v, \mathbf{1})}$, $\|\underline{\mathbf{m}}\| \neq 0$
- the equivalence class for $\lambda \in \mathbb{R}$, $\lambda \neq 0$ is $(m_1, m_2, m_3) = \lambda\,\underline{\mathbf{m}} \simeq \underline{\mathbf{m}}$
- the standard representative for finite point $\underline{\mathbf{m}}$ is $\lambda\,\underline{\mathbf{m}}$, where $\lambda = \frac{\mathbf{1}}{m_3}$ assuming $m_3 \neq 0$
- when $\mathbf{1} = 1$ then units are pixels and $\lambda \underline{\mathbf{m}} = (u, v, 1)$
- when $\mathbf{1} = f$ then all elements have a similar magnitude, $f \sim$ image diagonal

use $\mathbf{1} = 1$ unless you know what you are doing;
all entities participating in a formula must be expressed in the same units

'Infinite' points

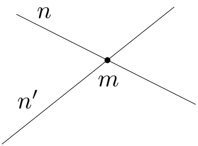
- we augment for **points at infinity** (ideal points) $\underline{\mathbf{m}}_\infty \simeq (m_1, m_2, 0)$

  proper members of $\mathbb{P}^2$
- all such points lie on the line at infinity (ideal line) $\underline{\mathbf{n}}_\infty \simeq (0, 0, 1)$, i.e. $\underline{\mathbf{m}}_\infty^\top \underline{\mathbf{n}}_\infty = 0$

▶Line Intersection and Point Join

The point of **intersection** m of image lines n and n', $n \not\simeq n'$ is

$$\underline{\mathbf{m}} \simeq \underline{\mathbf{n}} \times \underline{\mathbf{n}}'$$

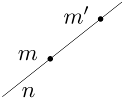


proof: If $\underline{\mathbf{m}} = \underline{\mathbf{n}} \times \underline{\mathbf{n}}'$ is the intersection point, it must be incident on both lines. Indeed, using known equivalences from vector algebra

$$\underline{\mathbf{n}}^\top \underbrace{(\underline{\mathbf{n}} \times \underline{\mathbf{n}}')}_{\underline{\mathbf{m}}} \equiv \underline{\mathbf{n}}'^\top \underbrace{(\underline{\mathbf{n}} \times \underline{\mathbf{n}}')}_{\underline{\mathbf{m}}} \equiv 0$$

The **join** n of two image points m and m', $m \not\simeq m'$ is

$$\underline{\mathbf{n}} \simeq \underline{\mathbf{m}} \times \underline{\mathbf{m}}'$$

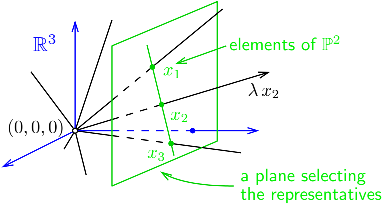


Paralel lines intersect (somewhere) on the line at infinity $\underline{\mathbf{n}}_\infty \simeq (0, 0, 1)$:

$$a\,u + b\,v + c = 0,$$
$$a\,u + b\,v + d = 0, \qquad d \neq c$$
$$(a, b, c) \times (a, b, d) \simeq (b, -a, 0)$$

- all such intersections lie on $\underline{\mathbf{n}}_\infty$
- line at infinity therefore represents the set of (unoriented) directions in the plane
- Matlab: `m = cross(n, n_prime);`

▶Homography in $\mathbb{P}^2$



Projective plane $\mathbb{P}^2$: Vector space of dimension 3 excluding the zero vector, $\mathbb{R}^3 \setminus (0,0,0)$, factorized to linear equivalence classes ('rays'), $\underline{\mathbf{x}} \simeq \lambda \underline{\mathbf{x}}$, $\lambda \neq 0$

including 'points at infinity'

we call $\underline{\mathbf{x}} \in \mathbb{P}^2$ 'points'

Homography in $\mathbb{P}^2$: Non-singular linear mapping in $\mathbb{P}^2$ — an analogic definition for $\mathbb{P}^3$

$$\underline{\mathbf{x}}' \simeq \mathbf{H}\,\underline{\mathbf{x}}, \quad \mathbf{H} \in \mathbb{R}^{3,3} \text{ non-singular}$$

Defining properties

- collinear points are mapped to collinear points

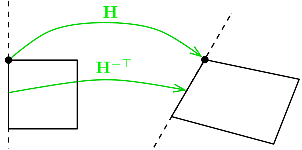
  lines of points are mapped to lines of points
- concurrent lines are mapped to concurrent lines

  concurrent = intersecting at a point
- and point-line incidence is preserved

  e.g. line intersection points mapped to line intersection points

- $\mathbf{H}$ is a 3×3 non-singular matrix, $\lambda\,\mathbf{H} \simeq \mathbf{H}$ equivalence class, 8 degrees of freedom
- homogeneous matrix representative: $\det \mathbf{H} = 1$ — $\mathbf{H} \in \mathrm{SL}(3)$
- what we call homography here is often called 'projective collineation' in mathematics

▶Mapping 2D Points and Lines by Homography



$$\underline{\mathbf{m}}' \simeq \mathbf{H}\,\underline{\mathbf{m}} \qquad \text{(image) point}$$

$$\underline{\mathbf{n}}' \simeq \mathbf{H}^{-\top}\underline{\mathbf{n}} \qquad \text{(image) line} \qquad \mathbf{H}^{-\top} = (\mathbf{H}^{-1})^{\top} = (\mathbf{H}^{\top})^{-1}$$

- incidence is preserved: $(\underline{\mathbf{m}}')^{\top}\underline{\mathbf{n}}' \simeq \underline{\mathbf{m}}^{\top}\mathbf{H}^{\top}\mathbf{H}^{-\top}\underline{\mathbf{n}} = \underline{\mathbf{m}}^{\top}\underline{\mathbf{n}} = 0$

Mapping a finite 2D point $\mathbf{m} = (u, v)$ to $\underline{\mathbf{m}} = (u', v')$

1. extend the Cartesian (pixel) coordinates to homogeneous coordinates, $\underline{\mathbf{m}} = (u, v, \mathbf{1})$
2. map by homography, $\underline{\mathbf{m}}' = \mathbf{H}\,\underline{\mathbf{m}}$
3. if $m_3' \neq 0$ convert the result $\underline{\mathbf{m}}' = (m_1', m_2', m_3')$ back to Cartesian coordinates (pixels),

$$u' = \frac{m_1'}{m_3'}\,\mathbf{1}, \qquad v' = \frac{m_2'}{m_3'}\,\mathbf{1}$$

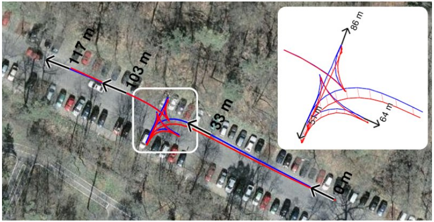
- note that, typically, $m_3' \neq 1$ — $m_3' = 1$ when $\mathbf{H}$ is affine
- an infinite point $\underline{\mathbf{m}} = (u, v, 0)$ maps the same way

Some Homographic Tasters

Rectification of camera rotation: →59 (geometry), →129 (homography estimation)

$\mathbf{H} \simeq \mathbf{K}\mathbf{R}^{\top}\mathbf{K}^{-1}$ maps from image plane to facade plane

Homographic Mouse for Visual Odometry: [Mallis 2007]



illustrations courtesy of AMSL Racing Team, Meiji University and LIBVISO: Library for VISual Odometry

$$\mathbf{H} \simeq \mathbf{K}\left(\mathbf{R} - \frac{\mathbf{t}\mathbf{n}^{\top}}{d}\right)\mathbf{K}^{-1}$$

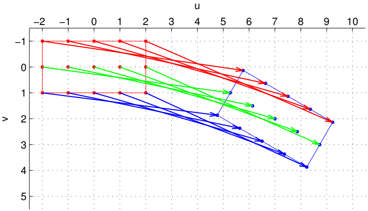
maps from plane to translated plane [H&Z, p. 327]

▶Homography Subgroups: Euclidean Mapping (aka Rigid Motion)

- Euclidean mapping (EM): rotation, translation and their combination

$$\mathbf{H} = \begin{bmatrix} \cos\phi & -\sin\phi & t_x \\ \sin\phi & \cos\phi & t_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \in \mathrm{SE}(3)$$

- note: action $H(\mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{t} \colon \mathbb{R}^2 \to \mathbb{R}^2$



rotation by 30°, then translation by $(7,\ 2)$

EM = The most general homography preserving

1. **lengths**: Let $\mathbf{x}'_i = H(\mathbf{x}_i)$. Then

$$\|\mathbf{x}'_2 - \mathbf{x}'_1\| = \|H(\mathbf{x}_2) - H(\mathbf{x}_1)\| \overset{\circledast\ \text{P1; 1pt}}{=} \cdots = \|\mathbf{x}_2 - \mathbf{x}_1\|$$

2. **angles** check the dot-product of normalized differences from a point $(\mathbf{x}-\mathbf{z})^\top(\mathbf{y}-\mathbf{z})$ (Cartesian(!))
3. **areas**: $\det \mathbf{H} = 1 \ \Rightarrow$ unit Jacobian; follows from 1. and 2.

- eigenvalues $(1,\, e^{-i\phi},\, e^{i\phi})$
- eigenvectors when $\phi \neq k\pi,\ k = 0, 1, \ldots$ (columnwise)

$$\mathbf{e}_1 \simeq \begin{bmatrix} t_x + t_y \cot\frac{\phi}{2} \\ t_y - t_x \cot\frac{\phi}{2} \\ 2 \end{bmatrix}, \quad \mathbf{e}_2 \simeq \begin{bmatrix} i \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{e}_3 \simeq \begin{bmatrix} -i \\ 1 \\ 0 \end{bmatrix}$$

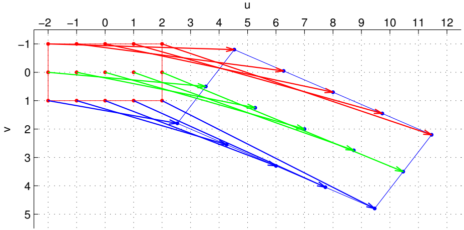
$\mathbf{e}_2$, $\mathbf{e}_3$ – circular points, i – imaginary unit

4. **circular points**: points at infinity $(i, 1, 0)$, $(-i, 1, 0)$ (preserved even by similarity)

- **similarity**: scaled Euclidean mapping (does not preserve lengths, areas)

▶Homography Subgroups: Affine Mapping

$$\mathbf{H} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$



rotation by 30°
then scaling by $\mathrm{diag}(1,\ 1.5,\ 1)$
then translation by $(7,\ 2)$

AM = The most general homography preserving

- parallelism
- ratio of areas
- ratio of lengths on <u>parallel lines</u>
- linear combinations of vectors (e.g. midpoints)
- convex hull
- line at infinity $\underline{\mathbf{n}}_\infty$ (not pointwise)

observe $\mathbf{H}^\top \underline{\mathbf{n}}_\infty \simeq \begin{bmatrix} a_{11} & a_{21} & 0 \\ a_{12} & a_{22} & 0 \\ t_x & t_y & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \underline{\mathbf{n}}_\infty \quad \Rightarrow \quad \underline{\mathbf{n}}_\infty \simeq \mathbf{H}^{-\top} \underline{\mathbf{n}}_\infty$

does not preserve

- lengths
- angles
- areas
- circular points

Euclidean mappings preserve all properties affine mappings preserve, of course

▶Homography Subgroups: General Homography

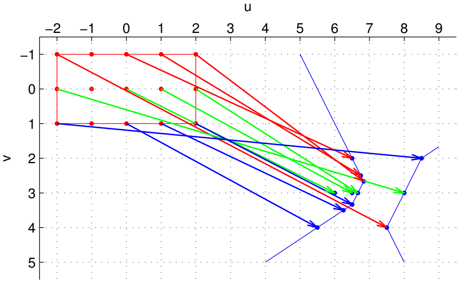
$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \qquad \mathbf{H} \in \mathrm{SL}(3)$$

preserves only

- incidence and concurrency
- collinearity
- cross-ratio on the line →46

does not preserve

- lengths
- areas
- parallelism
- ratio of areas
- ratio of lengths
- linear combinations of vectors (midpoints, etc.)
- convex hull

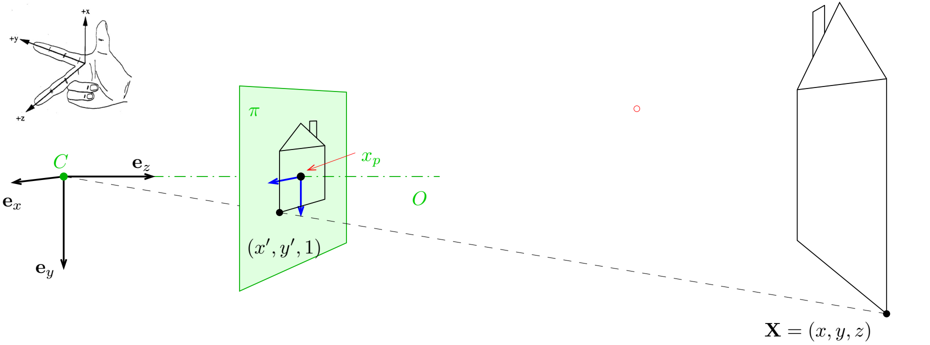


$$\mathbf{H} = \begin{bmatrix} 7 & -0.5 & 6 \\ 3 & 1 & 3 \\ 1 & 0 & 1 \end{bmatrix}$$

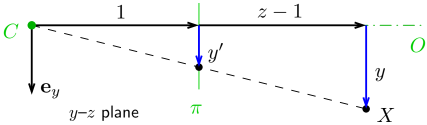
line $\underline{\mathbf{n}} = (1, 0, 1)$ is mapped to $\underline{\mathbf{n}}_\infty$: $\mathbf{H}^{-\top}\underline{\mathbf{n}} \simeq \underline{\mathbf{n}}_\infty$

(where in the picture is the line n?)

►Canonical Perspective Camera (Pinhole Camera, Camera Obscura)



1. in this picture we are looking 'down the street'
2. right-handed canonical coordinate system (x, y, z) with unit vectors $\mathbf{e}_x$, $\mathbf{e}_y$, $\mathbf{e}_z$
3. origin = center of projection C
4. image plane π at unit distance from C
5. optical axis O is perpendicular to π
6. principal point x_p: intersection of O and π
7. perspective camera is given by C and π



projected point in the natural image coordinate system:

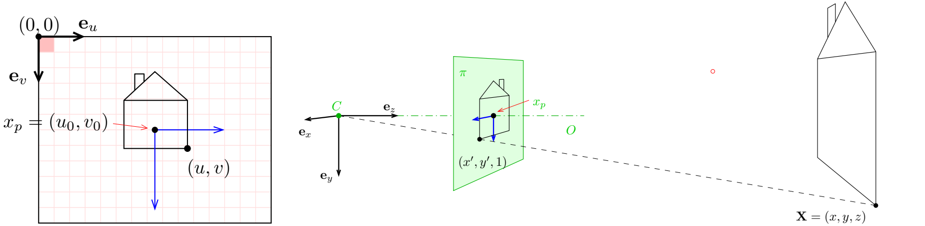
$$\frac{y'}{1} = y' = \frac{y}{1+z-1} = \frac{y}{z}, \qquad x' = \frac{x}{z}$$

►Natural and Canonical Image Coordinate Systems

projected point **in canonical camera** $(z \neq 0)$

$$(x', y', 1) = \left(\frac{x}{z}, \frac{y}{z}, 1\right) = \frac{1}{z}(x, y, z) \simeq (x, y, z) = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathbf{P}_0 = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \mathbf{P}_0 \, \underline{\mathbf{X}}$$

projected point **in scanned image** scale by f and translate origin to image corner



$$u = f\,\frac{x}{z} + u_0 \qquad v = f\,\frac{y}{z} + v_0$$

$$\frac{1}{z}\begin{bmatrix} f\,x + z\,u_0 \\ f\,y + z\,v_0 \\ z \end{bmatrix} \simeq \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{P}_0 \, \underline{\mathbf{X}} = \mathbf{P} \, \underline{\mathbf{X}}$$

- 'calibration' matrix $\mathbf{K}$ transforms canonical $\mathbf{P}_0$ to standard perspective camera $\mathbf{P}$

►Computing with Perspective Camera Projection Matrix

Projection from world to image in standard camera $\mathbf{P}$:

$$\underbrace{\begin{bmatrix} f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathbf{P}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} fx + u_0 z \\ fy + v_0 z \\ z \end{bmatrix} \simeq \underbrace{\begin{bmatrix} x + \frac{z}{f} u_0 \\ y + \frac{z}{f} v_0 \\ \frac{z}{f} \end{bmatrix}}_{\text{(a)}} \simeq \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \underline{\mathbf{m}}$$

$$\frac{m_1}{m_3} = \frac{f\,x}{z} + u_0 = u, \qquad \frac{m_2}{m_3} = \frac{f\,y}{z} + v_0 = v \quad \text{when} \quad m_3 \neq 0$$

f – 'focal length' – converts length ratios to pixels, $[f] = \mathrm{px}$, $f > 0$

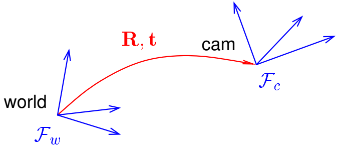
(u_0, v_0) – principal point in pixels

Perspective Camera:

1. dimension reduction — since $\mathbf{P} \in \mathbb{R}^{3,4}$
2. nonlinear unit change $\mathbf{1} \mapsto \mathbf{1} \cdot z/f$, see (a)
   for convenience we use $P_{11} = P_{22} = f$ rather than $P_{33} = 1/f$ and the u_0, v_0 in relative units
3. $m_3 = 0$ represents points at infinity in image plane π — i.e. points with $z = 0$

►Changing The Outer (World) Reference Frame

A transformation of a point from the world to camera coordinate system:



$$\mathbf{X}_c = \mathbf{R}\,\mathbf{X}_w + \mathbf{t}$$

$\mathbf{R}$ – camera rotation matrix — world orientation in the camera coordinate frame $\mathcal{F}_c$

$\mathbf{t}$ – camera translation vector — world origin in the camera coordinate frame $\mathcal{F}_c$

$$\mathbf{P}\,\underline{\mathbf{X}}_c = \mathbf{K}\mathbf{P}_0 \begin{bmatrix} \mathbf{X}_c \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{P}_0 \begin{bmatrix} \mathbf{R}\mathbf{X}_w + \mathbf{t} \\ 1 \end{bmatrix} = \mathbf{K}\begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix} \underbrace{\begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}}_{\mathbf{T}} \begin{bmatrix} \mathbf{X}_w \\ 1 \end{bmatrix} = \mathbf{K}\begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \underline{\mathbf{X}}_w$$

$\mathbf{P}_0$ (a 3×4 mtx) discards the last row of $\mathbf{T}$

- $\mathbf{R}$ is rotation, $\mathbf{R}^\top\mathbf{R} = \mathbf{I}$, $\det\mathbf{R} = +1$ — $\mathbf{I} \in \mathbb{R}^{3,3}$ identity matrix
- 6 **extrinsic parameters**: 3 rotation angles (Euler theorem), 3 translation components
- alternative, often used, camera representations

$$\mathbf{P} = \mathbf{K}\begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} = \mathbf{K}\mathbf{R}\begin{bmatrix} \mathbf{I} & -\mathbf{C} \end{bmatrix}$$

$\mathbf{C}$ – camera position in the world reference frame $\mathcal{F}_w$ — $\mathbf{t} = -\mathbf{R}\mathbf{C}$

$\mathbf{r}_3^\top$ – optical axis in the world reference frame $\mathcal{F}_w$ — third row of $\mathbf{R}$: $\mathbf{r}_3 = \mathbf{R}^{-1}[0, 0, 1]^\top$

- we can save some conversion and computation by noting that $\mathbf{K}\mathbf{R}\begin{bmatrix} \mathbf{I} & -\mathbf{C} \end{bmatrix}\underline{\mathbf{X}} = \mathbf{K}\mathbf{R}(\mathbf{X} - \mathbf{C})$

►Changing the Inner (Image) Reference Frame

The general form of calibration matrix $\mathbf{K}$ includes

- skew angle θ of the digitization raster
- pixel aspect ratio a

$$\mathbf{K} = \begin{bmatrix} a\,f & -a\,f\,\cot\theta & u_0 \\ 0 & f/\sin\theta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

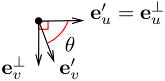
units: $[f] = \mathrm{px}$, $[u_0] = \mathrm{px}$, $[v_0] = \mathrm{px}$, $[a] = 1$

⊛ H1; 2pt: Decompose $\mathbf{K}$ to simple maps and give f, a, θ, u_0, v_0 a precise meaning; deadline LD+2 wk

Hints:

1. image projects to orthogonal system $F^{\perp}$, then it maps by skew to F', then by scale $a\,f$, f to F'', then by translation by u_0, v_0 to F'''
2. Skew: Do not confuse it with the shear mapping. Express point $\mathbf{x}$ as

$$\mathbf{x} = u'\mathbf{e}_{u'} + v'\mathbf{e}_{v'} = u^{\perp}\mathbf{e}_u^{\perp} + v^{\perp}\mathbf{e}_v^{\perp}$$



   $\mathbf{e}_{:}$ are unit-length basis vectors; consider their four pairwise dot-products.

3. $\mathbf{K}$ maps from $F^{\perp}$ to F''' as

$$w'''\,[u''', v''', 1]^{\top} = \mathbf{K}[u^{\perp}, v^{\perp}, 1]^{\top}$$

►Summary: Projection Matrix of a General Finite Perspective Camera

$$\underline{\mathbf{m}} \simeq \mathbf{P}\underline{\mathbf{X}}, \qquad \mathbf{P} = \begin{bmatrix}\mathbf{Q} & \mathbf{q}\end{bmatrix} \simeq \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} = \mathbf{K}\mathbf{R}\begin{bmatrix}\mathbf{I} & -\mathbf{C}\end{bmatrix}$$

a recipe for filling $\mathbf{P}$

general finite perspective camera has 11 parameters:

- 5 intrinsic parameters: f, u_0, v_0, a, θ
- 6 extrinsic parameters: $\mathbf{t}$, $\mathbf{R}(\alpha, \beta, \gamma)$

finite camera: $\det \mathbf{K} \neq 0$

Representation Theorem: The set of projection matrices $\mathbf{P}$ of finite perspective cameras is isomorphic to the set of homogeneous 3×4 matrices with the left 3×3 submatrix $\mathbf{Q}$ non-singular.

random finite camera: `Q = rand(3,3); while det(Q)==0, Q = rand(3,3); end, P = [Q, rand(3,1)];`

▶Projection Matrix Decomposition

$$\mathbf{P} = \begin{bmatrix}\mathbf{Q} & \mathbf{q}\end{bmatrix} \quad \longrightarrow \quad \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}$$

$\mathbf{Q} \in \mathbb{R}^{3,3}$ full rank (if finite perspective camera; see [H&Z, Sec. 6.3] for cameras at infinity)
$\mathbf{K} \in \mathbb{R}^{3,3}$ upper triangular with positive diagonal elements
$\mathbf{R} \in \mathbb{R}^{3,3}$ rotation mtx: $\mathbf{R}^\top\mathbf{R} = \mathbf{I}$ and $\det \mathbf{R} = +1$

1. $\begin{bmatrix}\mathbf{Q} & \mathbf{q}\end{bmatrix} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} = \begin{bmatrix}\mathbf{KR} & \mathbf{Kt}\end{bmatrix}$ also →35
2. RQ decomposition of $\mathbf{Q} = \mathbf{KR}$ using three Givens rotations [H&Z, p. 579]

$$\mathbf{K} = \mathbf{Q}\,\underbrace{\mathbf{R}_{32}\mathbf{R}_{31}\mathbf{R}_{21}}_{\mathbf{R}^{-1}} \qquad \mathbf{Q}\mathbf{R}_{32} = \begin{bmatrix}\cdot & \cdot & \cdot\\ \cdot & \cdot & \cdot\\ \cdot & 0 & \cdot\end{bmatrix},\ \mathbf{Q}\mathbf{R}_{32}\mathbf{R}_{31} = \begin{bmatrix}\cdot & \cdot & \cdot\\ \cdot & \cdot & \cdot\\ 0 & 0 & \cdot\end{bmatrix},\ \mathbf{Q}\mathbf{R}_{32}\mathbf{R}_{31}\mathbf{R}_{21} = \begin{bmatrix}\cdot & \cdot & \cdot\\ 0 & \cdot & \cdot\\ 0 & 0 & \cdot\end{bmatrix}$$

$\mathbf{R}_{ij}$ zeroes element ij in $\mathbf{Q}$ affecting only columns i and j and the sequence preserves previously zeroed elements, e.g. (see next slide for derivation details)

$$\mathbf{R}_{32} = \begin{bmatrix}1 & 0 & 0\\ 0 & c & -s\\ 0 & s & c\end{bmatrix} \text{ gives } \begin{matrix} c^2+s^2=1 \\ 0 = k_{32} = c\,q_{32} + s\,q_{33}\end{matrix} \Rightarrow c = \frac{q_{33}}{\sqrt{q_{32}^2+q_{33}^2}} \quad s = \frac{-q_{32}}{\sqrt{q_{32}^2+q_{33}^2}}$$

⊛ P1; 1pt: Multiply known matrices $\mathbf{K}$, $\mathbf{R}$ and then decompose back; discuss numerical errors

- RQ decomposition nonuniqueness: $\mathbf{KR} = \mathbf{KT}^{-1}\mathbf{TR}$, where $\mathbf{T} = \mathrm{diag}(-1,-1,1)$ is also a rotation, we must correct the result so that the diagonal elements of $\mathbf{K}$ are all positive
  'thin' RQ decomposition
- care must be taken to avoid overflow, see [Golub & van Loan 2013, sec. 5.2]

RQ Decomposition Step

```
Q = Array [q#1,#2 &, {3, 3}];
R32 = {{1, 0, 0}, {0, c, -s}, {0, s, c}}; R32 // MatrixForm
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & c & -s \\ 0 & s & c \end{pmatrix}$$

```
Q1 = Q.R32 ; Q1 // MatrixForm
```

$$\begin{pmatrix} q_{1,1} & c\, q_{1,2} + s\, q_{1,3} & -s\, q_{1,2} + c\, q_{1,3} \\ q_{2,1} & c\, q_{2,2} + s\, q_{2,3} & -s\, q_{2,2} + c\, q_{2,3} \\ q_{3,1} & c\, q_{3,2} + s\, q_{3,3} & -s\, q_{3,2} + c\, q_{3,3} \end{pmatrix}$$

```
s1 = Solve [{Q1[[3]][[2]] == 0, c^2 + s^2 == 1}, {c, s}][[2]]
```

$$\left\{ c \to \frac{q_{3,3}}{\sqrt{q_{3,2}^2 + q_{3,3}^2}},\ s \to -\frac{q_{3,2}}{\sqrt{q_{3,2}^2 + q_{3,3}^2}} \right\}$$

```
Q1 /. s1 // Simplify // MatrixForm
```

$$\begin{pmatrix} q_{1,1} & \frac{-q_{1,3}\, q_{3,2} + q_{1,2}\, q_{3,3}}{\sqrt{q_{3,2}^2 + q_{3,3}^2}} & \frac{q_{1,2}\, q_{3,2} + q_{1,3}\, q_{3,3}}{\sqrt{q_{3,2}^2 + q_{3,3}^2}} \\ q_{2,1} & \frac{-q_{2,3}\, q_{3,2} + q_{2,2}\, q_{3,3}}{\sqrt{q_{3,2}^2 + q_{3,3}^2}} & \frac{q_{2,2}\, q_{3,2} + q_{2,3}\, q_{3,3}}{\sqrt{q_{3,2}^2 + q_{3,3}^2}} \\ q_{3,1} & 0 & \sqrt{q_{3,2}^2 + q_{3,3}^2} \end{pmatrix}$$

▶Center of Projection (Optical Center)

Observation: finite $\mathbf{P}$ has a non-trivial right null-space — rank 3 but 4 columns

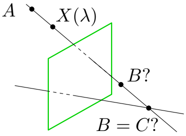
Theorem

Let $\mathbf{P}$ be a camera and let there be $\underline{\mathbf{B}} \neq \mathbf{0}$ s.t. $\mathbf{P}\,\underline{\mathbf{B}} = \mathbf{0}$. Then $\underline{\mathbf{B}}$ is equivalent to the projection center $\underline{\mathbf{C}}$ (homogeneous, in world coordinate frame).

Proof.

1. Let AB be a spatial line (B given from $\mathbf{P}\underline{\mathbf{B}} = \mathbf{0}$, $A \neq B$). Then

$$\underline{\mathbf{X}}(\lambda) \simeq \lambda\,\underline{\mathbf{A}} + (1-\lambda)\,\underline{\mathbf{B}}, \qquad \lambda \in \mathbb{R}$$



2. It projects to

$$\mathbf{P}\underline{\mathbf{X}}(\lambda) \simeq \lambda\,\mathbf{P}\,\underline{\mathbf{A}} + (1-\lambda)\,\mathbf{P}\,\underline{\mathbf{B}} \simeq \mathbf{P}\,\underline{\mathbf{A}}$$

- the entire line projects to a single point $\Rightarrow$ it must pass through the projection center of $\mathbf{P}$
- this holds for any choice of $A \neq B$ $\Rightarrow$ the only common point of the lines is the C, i.e. $\underline{\mathbf{B}} \simeq \underline{\mathbf{C}}$

□

Hence

$$\mathbf{0} = \mathbf{P}\,\underline{\mathbf{C}} = \begin{bmatrix}\mathbf{Q} & \mathbf{q}\end{bmatrix}\begin{bmatrix}\mathbf{C}\\ 1\end{bmatrix} = \mathbf{Q}\,\mathbf{C} + \mathbf{q} \Rightarrow \boxed{\mathbf{C} = -\mathbf{Q}^{-1}\mathbf{q}}$$

$\underline{\mathbf{C}} = (c_j)$, where $c_j = (-1)^j \det \mathbf{P}^{(j)}$, in which $\mathbf{P}^{(j)}$ is $\mathbf{P}$ with column j dropped

Matlab: `C_homo = null(P);` or `C = -Q\q;`

▶Optical Ray

Optical ray: Spatial line that projects to a single image point.

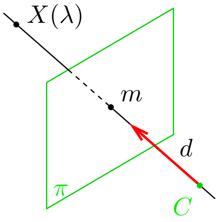
1. Consider the following spatial line
   $\mathbf{d} \in \mathbb{R}^3$ line direction vector, $\|\mathbf{d}\| = 1$, $\lambda \in \mathbb{R}$, Cartesian representation

$$\mathbf{X}(\lambda) = \mathbf{C} + \lambda\,\mathbf{d}$$

2. The projection of the (finite) point $X(\lambda)$ is

$$\underline{\mathbf{m}} \simeq \begin{bmatrix}\mathbf{Q} & \mathbf{q}\end{bmatrix} \begin{bmatrix}\mathbf{X}(\lambda) \\ 1\end{bmatrix} = \mathbf{Q}(\mathbf{C} + \lambda\mathbf{d}) + \mathbf{q} = \lambda\,\mathbf{Q}\,\mathbf{d} =$$

$$= \lambda \begin{bmatrix}\mathbf{Q} & \mathbf{q}\end{bmatrix} \begin{bmatrix}\mathbf{d} \\ 0\end{bmatrix}$$



…which is also the image of a point at infinity in $\mathbb{P}^3$

- optical ray line corresponding to image point m is the set

$$\boxed{\mathbf{X}(\mu) = \mathbf{C} + \mu\,\mathbf{Q}^{-1}\underline{\mathbf{m}}, \qquad \mu \in \mathbb{R} \qquad (\mu = 1/\lambda)}$$

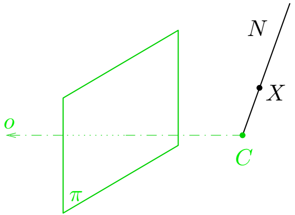
- optical ray direction may be represented by a point at infinity $(\mathbf{d}, 0)$ in $\mathbb{P}^3$
- optical ray is expressed in world coordinate frame

▶Optical Axis

Optical axis: Optical ray that is perpendicular to image plane π

1. points X on a given line N parallel to π project to a point at infinity $(u, v, 0)$ in π:

$$\begin{bmatrix} u \\ v \\ 0 \end{bmatrix} \simeq \mathbf{P}\underline{\mathbf{X}} = \begin{bmatrix} \mathbf{q}_1^\top & q_{14} \\ \mathbf{q}_2^\top & q_{24} \\ \mathbf{q}_3^\top & q_{34} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}$$



2. therefore the set of points X is parallel to π iff

$$\mathbf{q}_3^\top \mathbf{X} + q_{34} = 0$$

3. this is a plane with $\pm\mathbf{q}_3$ as the normal vector
4. optical axis direction: substitution $\mathbf{P} \mapsto \lambda\mathbf{P}$ must not change the direction
5. we select (assuming $\det(\mathbf{R}) > 0$)

$$\boxed{\mathbf{o} = \det(\mathbf{Q})\,\mathbf{q}_3}$$

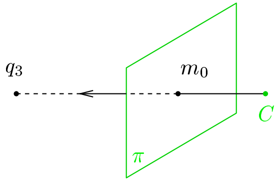
if $\mathbf{P} \mapsto \lambda\mathbf{P}$ then $\det(\mathbf{Q}) \mapsto \lambda^3 \det(\mathbf{Q})$ and $\mathbf{q}_3 \mapsto \lambda\,\mathbf{q}_3$ [H&Z, p. 161]

- the axis is expressed in world coordinate frame

▶Principal Point

Principal point: The intersection of image plane and the optical axis

1. as we saw, $\mathbf{q}_3$ is the directional vector of optical axis
2. we take point at infinity on the optical axis that must project to the principal point m_0



3. then

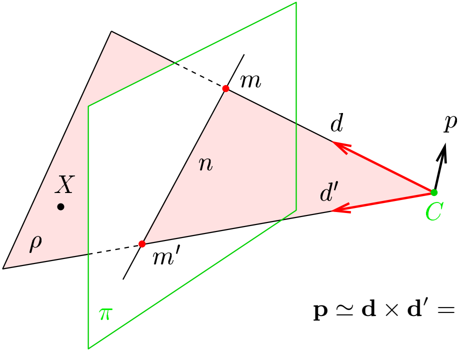
$$\underline{\mathbf{m}}_0 \simeq \begin{bmatrix} \mathbf{Q} & \mathbf{q} \end{bmatrix} \begin{bmatrix} \mathbf{q}_3 \\ 0 \end{bmatrix} = \mathbf{Q}\,\mathbf{q}_3$$

principal point: $\boxed{\underline{\mathbf{m}}_0 \simeq \mathbf{Q}\,\mathbf{q}_3}$

- principal point is also the center of radial distortion

▶Optical Plane

A spatial plane with normal p containing the projection center C and a given image line n.



optical ray given by m $\quad \mathbf{d} \simeq \mathbf{Q}^{-1}\underline{\mathbf{m}}$

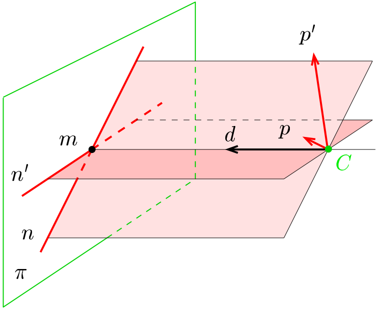
optical ray given by m' $\quad \mathbf{d}' \simeq \mathbf{Q}^{-1}\underline{\mathbf{m}}'$

$$\mathbf{p} \simeq \mathbf{d} \times \mathbf{d}' = (\mathbf{Q}^{-1}\underline{\mathbf{m}}) \times (\mathbf{Q}^{-1}\underline{\mathbf{m}}') = \mathbf{Q}^{\top}(\underline{\mathbf{m}} \times \underline{\mathbf{m}}') = \mathbf{Q}^{\top}\underline{\mathbf{n}}$$

- note the way $\mathbf{Q}$ factors out!

hence, $0 = \mathbf{p}^{\top}(\mathbf{X} - \mathbf{C}) = \underline{\mathbf{n}}^{\top}\underbrace{\mathbf{Q}(\mathbf{X} - \mathbf{C})}_{\rightarrow 30} = \underline{\mathbf{n}}^{\top}\mathbf{P}\underline{\mathbf{X}} = (\mathbf{P}^{\top}\underline{\mathbf{n}})^{\top}\underline{\mathbf{X}}$ for every X in plane ρ

optical plane is given by n: $\quad \boxed{\underline{\boldsymbol{\rho}} \simeq \mathbf{P}^{\top}\underline{\mathbf{n}}}$ $\qquad \rho_1\, x + \rho_2\, y + \rho_3\, z + \rho_4 = 0$



optical plane normal given by n $$\mathbf{p} = \mathbf{Q}^\top \underline{\mathbf{n}}$$

optical plane normal given by n' $$\mathbf{p}' = \mathbf{Q}^\top \underline{\mathbf{n}}'$$

$$\mathbf{d} = \mathbf{p} \times \mathbf{p}' = (\mathbf{Q}^\top \underline{\mathbf{n}}) \times (\mathbf{Q}^\top \underline{\mathbf{n}}') = \mathbf{Q}^{-1}(\underline{\mathbf{n}} \times \underline{\mathbf{n}}') = \mathbf{Q}^{-1}\underline{\underline{\mathbf{m}}}$$

▶Summary: Projection Center; Optical Ray, Axis, Plane

General (finite) camera

$$\mathbf{P} = \begin{bmatrix}\mathbf{Q} & \mathbf{q}\end{bmatrix} = \begin{bmatrix}\mathbf{q}_1^\top & q_{14} \\ \mathbf{q}_2^\top & q_{24} \\ \mathbf{q}_3^\top & q_{34}\end{bmatrix} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix} = \mathbf{K}\,\mathbf{R}\begin{bmatrix}\mathbf{I} & -\mathbf{C}\end{bmatrix}$$

$\underline{\mathbf{C}} \simeq \operatorname{rnull}(\mathbf{P}), \quad \mathbf{C} = -\mathbf{Q}^{-1}\mathbf{q}$ — projection center (world coords.) →35

$\mathbf{d} = \mathbf{Q}^{-1}\,\underline{\mathbf{m}}$ — optical ray direction (world coords.) →36

$\mathbf{o} = \det(\mathbf{Q})\,\mathbf{q}_3$ — outward optical axis (world coords.) →37

$\underline{\mathbf{m}}_0 \simeq \mathbf{Q}\,\mathbf{q}_3$ — principal point (in image plane) →38

$\underline{\boldsymbol{\rho}} = \mathbf{P}^\top\,\underline{\mathbf{n}}$ — optical plane (world coords.) →39

$$\mathbf{K} = \begin{bmatrix} a\,f & -a\,f\,\cot\theta & u_0 \\ 0 & f/\sin\theta & v_0 \\ 0 & 0 & 1\end{bmatrix}$$ camera (calibration) matrix (f, u_0, v_0 in pixels) →31

$\mathbf{R}$ — camera rotation matrix (cam coords.) →30

$\mathbf{t}$ — camera translation vector (cam coords.) →30

What Can We Do with An 'Uncalibrated' Perspective Camera?

How far is the engine?

distance between sleepers (ties) 0.806m but we cannot count them, the image resolution is too low

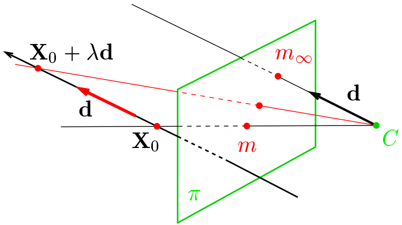
We will review some life-saving theory...
...and build a bit of geometric intuition...

In fact

- 'uncalibrated' = the image contains a 'calibrating object' that suffices for the task at hand

▶Vanishing Point

Vanishing point: the limit of the projection of a point that moves along a space line infinitely in one direction. the image of the point at infinity on the line



$$\underline{\mathbf{m}}_\infty \simeq \lim_{\lambda \to \pm\infty} \mathbf{P} \begin{bmatrix} \mathbf{X}_0 + \lambda\mathbf{d} \\ 1 \end{bmatrix} = \cdots \simeq \mathbf{Q}\,\mathbf{d}$$

⊛ P1; 1pt: Prove (use Cartesian coordinates and L'Hôpital's rule)

- the V.P. of a spatial line with directional vector $\mathbf{d}$ is $\underline{\mathbf{m}}_\infty \simeq \mathbf{Q}\,\mathbf{d}$
- V.P. is independent on line position $\mathbf{X}_0$, it depends on its directional vector only
- all parallel (world) lines share the same (image) V.P., including the optical ray defined by m_∞

Some Vanishing Point "Applications"

where is the sun?

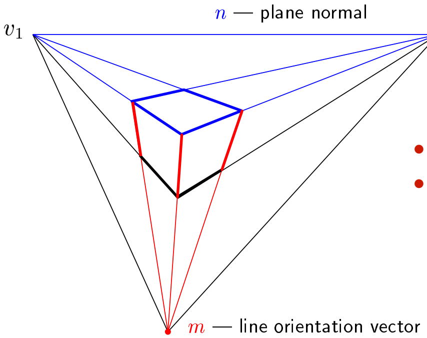
what is the wind direction?
(must have video)

fly above the lane,
at constant altitude!

►Vanishing Line

Vanishing line: The set of vanishing points of all lines in a plane

the image of the line at infinity in the plane
and in all parallel planes (!)



- any box with parallel edges
- top (blue) and bottom (black) box planes are parallel, hence they share V.L. n

- V.L. n corresponds to spatial plane of normal vector $\mathbf{p} = \mathbf{Q}^\top \underline{\mathbf{n}}$

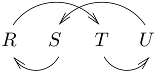
  because this is the normal vector of a parallel optical plane (!) →39

- a spatial plane of normal vector $\mathbf{p}$ has a V.L. represented by $\underline{\mathbf{n}} = \mathbf{Q}^{-\top} \mathbf{p}$.

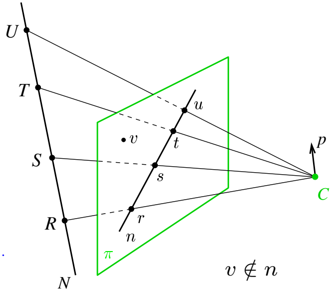
▶Cross Ratio

Four distinct collinear spatial points R, S, T, U define cross-ratio

$$[RSTU] = \frac{|\overrightarrow{RT}|}{|\overrightarrow{SR}|}\,\frac{|\overrightarrow{US}|}{|\overrightarrow{TU}|}$$



a mnemonic (∞)



- $|\overrightarrow{RT}|$ – signed distance from R to T in the arrow direction
- each point X is once in numerator and once in denominator
- if X is 1st in a numerator term, it is 2nd in the denominator term
- six cross-ratios from four points:

$$[SRUT] = [RSTU],\ [RSUT] = \frac{1}{[RSTU]},\ [RTSU] = 1 - [RSTU],\ \cdots$$

Obs: $[RSTU] = \dfrac{|\underline{\mathbf{r}}\ \ \underline{\mathbf{t}}\ \ \underline{\mathbf{v}}|}{|\underline{\mathbf{s}}\ \ \underline{\mathbf{r}}\ \ \underline{\mathbf{v}}|}\cdot\dfrac{|\underline{\mathbf{u}}\ \ \underline{\mathbf{s}}\ \ \underline{\mathbf{v}}|}{|\underline{\mathbf{t}}\ \ \underline{\mathbf{u}}\ \ \underline{\mathbf{v}}|},\quad |\underline{\mathbf{r}}\ \ \underline{\mathbf{t}}\ \ \underline{\mathbf{v}}| = \det\left[\underline{\mathbf{r}}\ \ \underline{\mathbf{t}}\ \ \underline{\mathbf{v}}\right] = (\underline{\mathbf{r}} \times \underline{\mathbf{t}})^\top \underline{\mathbf{v}}$ (1)

Corollaries:

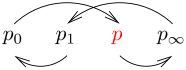
- cross ratio is invariant under homographies $\underline{\mathbf{x}}' \simeq \mathbf{H}\underline{\mathbf{x}}$ plug $\mathbf{H}\underline{\mathbf{x}}$ in (1): $(\mathbf{H}^{-\top}(\underline{\mathbf{r}} \times \underline{\mathbf{t}}))^\top \mathbf{H}\underline{\mathbf{v}}$
- cross ratio is invariant under perspective projection: $[RSTU] = [r\,s\,t\,u]$

- 4 collinear points: any perspective camera will "see" the same cross-ratio of their images
- we measure the same cross-ratio in image as on the world line
- one of the points R, S, T, U may be at infinity (we take the limit, in effect $\frac{\infty}{\infty} = 1$)

▶1D Projective Coordinates

The 1-D projective coordinate of a point P is defined by the following cross-ratio:

$$[P] = [P_0\, P_1\, P\, P_\infty] = [p_0\, p_1\, p\, p_\infty] = \frac{|\overrightarrow{p_0\, p}|}{|\overrightarrow{p_1\, p_0}|}\, \frac{|\overrightarrow{p_\infty\, p_1}|}{|\overrightarrow{p\, p_\infty}|} = [p]$$

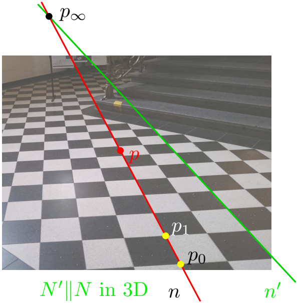


naming convention:

P_0 – the origin	$[P_0] = 0$
P_1 – the unit point	$[P_1] = 1$
P_∞ – the supporting point	$[P_\infty] = \pm\infty$

$$[P] = [p]$$

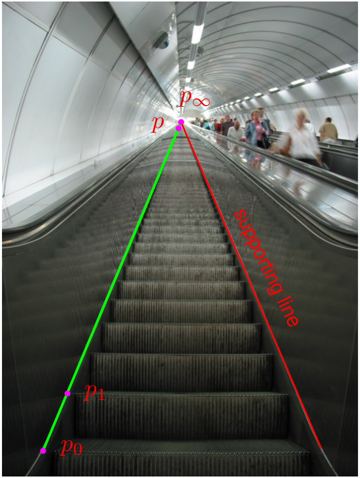
$[P]$ is equal to Euclidean coordinate along N
$[p]$ is its measurement in the image plane



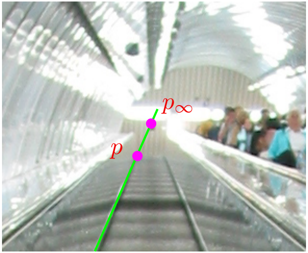
Applications

- Given the image of a 3D line N, the origin, the unit point, and the vanishing point, then the Euclidean coordinate of any point $P \in N$ can be determined →48
- Finding v.p. of a line through a regular object →49

Application: Counting Steps



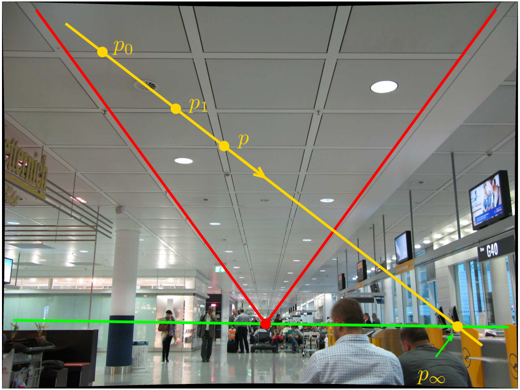
- Namesti Miru underground station in Prague

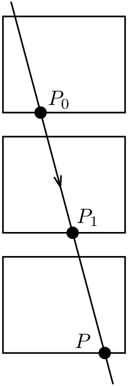


detail around the vanishing point

Result: $[P] = 214$ steps (correct answer is 216 steps) 4Mpx camera

Application: Finding the Horizon from Repetitions





in 3D: $|P_0 P| = 2|P_0 P_1|$ then [H&Z, p. 218]

$$[P_0 P_1 P P_\infty] = \frac{|P_0 P|}{|P_1 P_0|} = 2 \quad \Rightarrow \quad x_\infty = \frac{x_0\,(2\,x - x_1) - x\,x_1}{x + x_0 - 2\,x_1}$$

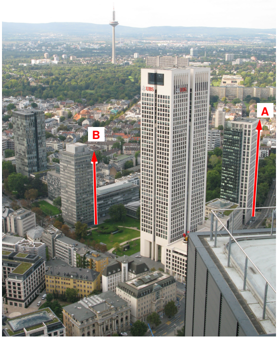
- x – 1D coordinate along the yellow line, positive in the arrow direction
- could be applied to counting steps (→48) if there was no supporting line

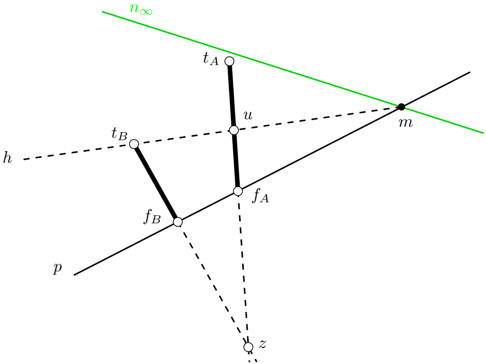
⊛ P1; 1pt: How high is the camera above the floor?

Homework Problem

⊛ H2; 3pt: What is the ratio of heights of Building A to Building B?

- expected: conceptual solution; use notation from this figure
- deadline: LD+2 weeks

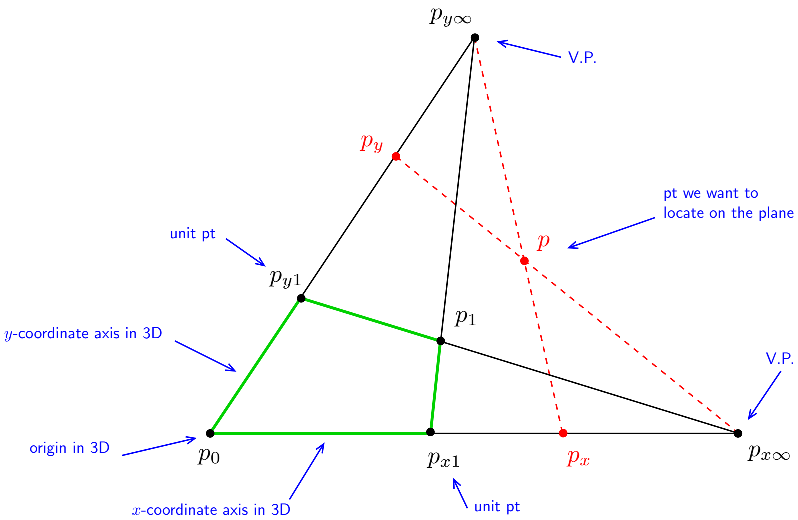




Hints

1. What are the interesting properties of line h connecting the top t_B of Buiding B with the point m at which the horizon intersects the line p joining the foots f_A, f_B of both buildings? [1 point]
2. How do we actually get the horizon n_∞? (we do not see it directly, there are some hills there...) [1 point]
3. Give the formula for measuring the length ratio. Make sure you distinguish points in 3D from their images [formula = 1 point]

2D Projective Coordinates



$$[P_x] = [P_0\ P_{x1}\ P_x\ P_{x\infty}] \qquad [P_y] = [P_0\ P_{y1}\ P_y\ P_{y\infty}]$$

Application: Measuring on the Floor (Wall, etc)

San Giovanni in Laterano, Rome

- measuring distances on the floor in terms of tile units
- what are the dimensions of the seal? Is it circular (assuming square tiles)?
- needs no explicit camera calibration

because we can see the calibrating object (vanishing points)

Module III

Computing with a Single Camera

3.1 Calibration: Internal Camera Parameters from Vanishing Points and Lines

3.2 Camera Resection: Projection Matrix from 6 Known Points

3.3 Exterior Orientation: Camera Rotation and Translation from 3 Known Points

3.4 Relative Orientation Problem: Rotation and Translation between Two Point Sets

covered by

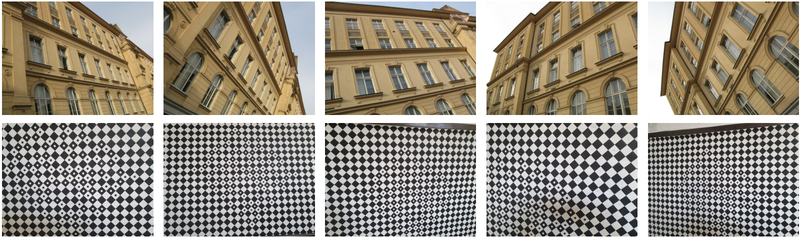
[1] [H&Z] Secs: 8.6, 7.1, 22.1

[2] Fischler, M.A. and Bolles, R.C . Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM* 24(6):381–395, 1981

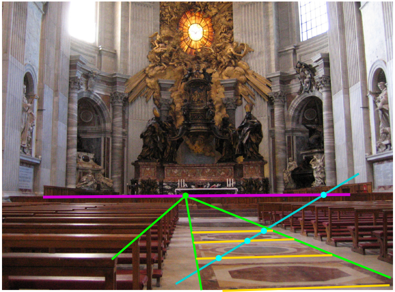
[3] [Golub & van Loan 2013, Sec. 2.5]

Obtaining Vanishing Points and Lines

- orthogonal direction pairs can be collected from multiple images by camera rotation

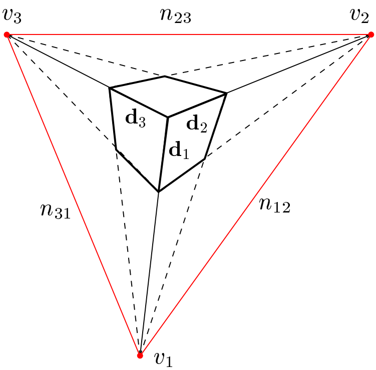


- vanishing line can be obtained from vanishing points and/or regularities (→49)



▶Camera Calibration from Vanishing Points and Lines

Problem: Given finite vanishing points and/or vanishing lines, compute $\mathbf{K}$



$$\mathbf{d}_i = \lambda_i \mathbf{Q}^{-1} \underline{\mathbf{v}}_i, \qquad i = 1,2,3 \quad \rightarrow 43$$

$$\mathbf{p}_{ij} = \mu_{ij} \mathbf{Q}^\top \underline{\mathbf{n}}_{ij}, \quad i,j = 1,2,3,\ i \neq j \quad \rightarrow 39 \tag{2}$$

- simple method: solve (2) after eliminating λ_i, μ_{ij}.

Special Configurations

1. orthogonal rays $\mathbf{d}_1 \perp \mathbf{d}_2$ in space then

$$0 = \mathbf{d}_1^\top \mathbf{d}_2 = \underline{\mathbf{v}}_1^\top \mathbf{Q}^{-\top} \mathbf{Q}^{-1} \underline{\mathbf{v}}_2 = \underline{\mathbf{v}}_1^\top \underbrace{(\mathbf{K}\mathbf{K}^\top)^{-1}}_{\boldsymbol{\omega}\ \text{(IAC)}} \underline{\mathbf{v}}_2$$

2. orthogonal planes $\mathbf{p}_{ij} \perp \mathbf{p}_{ik}$ in space

$$0 = \mathbf{p}_{ij}^\top \mathbf{p}_{ik} = \underline{\mathbf{n}}_{ij}^\top \mathbf{Q}\mathbf{Q}^\top \underline{\mathbf{n}}_{ik} = \underline{\mathbf{n}}_{ij}^\top \boldsymbol{\omega}^{-1} \underline{\mathbf{n}}_{ik}$$

3. orthogonal ray and plane $\mathbf{d}_k \parallel \mathbf{p}_{ij}$, $k \neq i,j$ — normal parallel to optical ray

$$\mathbf{p}_{ij} \simeq \mathbf{d}_k \quad \Rightarrow \quad \mathbf{Q}^\top \underline{\mathbf{n}}_{ij} = \frac{\lambda_i}{\mu_{ij}} \mathbf{Q}^{-1} \underline{\mathbf{v}}_k \quad \Rightarrow \quad \underline{\mathbf{n}}_{ij} = \varkappa\, \mathbf{Q}^{-\top} \mathbf{Q}^{-1} \underline{\mathbf{v}}_k = \varkappa\, \boldsymbol{\omega}\, \underline{\mathbf{v}}_k, \quad \varkappa \neq 0$$

- n_{ij} may be constructed from non-orthogonal v_i and v_j, e.g. using the cross-ratio
- $\boldsymbol{\omega}$ is a symmetric, positive definite 3×3 matrix — IAC = Image of Absolute Conic
- equations are quadratic in $\mathbf{K}$ but linear in $\boldsymbol{\omega}$

►cont'd

	configuration	equation	# constraints
(3)	orthogonal v.p.	$\underline{\mathbf{v}}_i^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_j = 0$	1
(4)	orthogonal v.l.	$\underline{\mathbf{n}}_{ij}^\top \boldsymbol{\omega}^{-1} \underline{\mathbf{n}}_{ik} = 0$	1
(5)	v.p. orthogonal to v.l.	$\underline{\mathbf{n}}_{ij} = \varkappa\, \boldsymbol{\omega}\, \underline{\mathbf{v}}_k$	2
(6)	orthogonal image raster $\theta = \pi/2$	$\omega_{12} = \omega_{21} = 0$	1
(7)	unit aspect $a = 1$ when $\theta = \pi/2$	$\omega_{11} - \omega_{22} = 0$	1
(8)	known principal point $u_0 = v_0 = 0$	$\omega_{13} = \omega_{31} = \omega_{23} = \omega_{32} = 0$	2

- these are homogeneous linear equations for the 5 parameters in $\boldsymbol{\omega}$ in the form $\mathbf{D}\mathbf{w} = \mathbf{0}$

  $\varkappa$ can be eliminated from (5)
- we need at least 5 constraints for full $\boldsymbol{\omega}$ — symmetric 3×3
- we get $\mathbf{K}$ from $\boldsymbol{\omega}^{-1} = \mathbf{K}\mathbf{K}^\top$ by Choleski decomposition

  the decomposition returns a positive definite upper triangular matrix

  one avoids solving an explicit set of quadratic equations for the parameters in $\mathbf{K}$

Examples

Assuming orthogonal raster, unit aspect (ORUA): $\theta = \pi/2$, $a = 1$

$$\boldsymbol{\omega} \simeq \begin{bmatrix} 1 & 0 & -u_0 \\ 0 & 1 & -v_0 \\ -u_0 & -v_0 & f^2 + u_0^2 + v_0^2 \end{bmatrix}$$

Ex 1:

Assuming ORUA and known $m_0 = (u_0, v_0)$, two <u>finite</u> orthogonal vanishing points give f

$$\underline{\mathbf{v}}_1^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_2 = 0 \quad \Rightarrow \quad f^2 = \left|(\mathbf{v}_1 - \mathbf{m}_0)^\top (\mathbf{v}_2 - \mathbf{m}_0)\right|$$

in this formula, $\mathbf{v}_i$, $\mathbf{m}_0$ are Cartesian (not homogeneous)!

Ex 2:

Non-orthogonal vanishing points $\mathbf{v}_i$, $\mathbf{v}_j$, known angle ϕ: $\cos\phi = \dfrac{\underline{\mathbf{v}}_i^\top \boldsymbol{\omega} \underline{\mathbf{v}}_j}{\sqrt{\underline{\mathbf{v}}_i^\top \boldsymbol{\omega} \underline{\mathbf{v}}_i}\sqrt{\underline{\mathbf{v}}_j^\top \boldsymbol{\omega} \underline{\mathbf{v}}_j}}$

- leads to polynomial equations
- e.g. ORUA and $u_0 = v_0 = 0$ gives

$$(f^2 + \mathbf{v}_i^\top \mathbf{v}_j)^2 = (f^2 + \|\mathbf{v}_i\|^2) \cdot (f^2 + \|\mathbf{v}_j\|^2) \cdot \cos^2\phi$$

▶Camera Orientation from Two Finite Vanishing Points

Problem: Given $\mathbf{K}$ and two vanishing points corresponding to two known orthogonal directions $\mathbf{d}_1$, $\mathbf{d}_2$, compute camera orientation $\mathbf{R}$ with respect to the plane.

- 3D coordinate system choice, e.g.:

$$\mathbf{d}_1 = (1, 0, 0), \quad \mathbf{d}_2 = (0, 1, 0)$$

- we know that

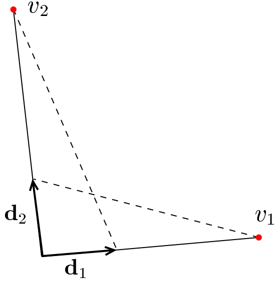
$$\mathbf{d}_i \simeq \mathbf{Q}^{-1}\underline{\mathbf{v}}_i = (\mathbf{K}\mathbf{R})^{-1}\underline{\mathbf{v}}_i = \mathbf{R}^{-1}\underbrace{\mathbf{K}^{-1}\underline{\mathbf{v}}_i}_{\underline{\mathbf{w}}_i}$$

$$\mathbf{R}\mathbf{d}_i \simeq \underline{\mathbf{w}}_i$$

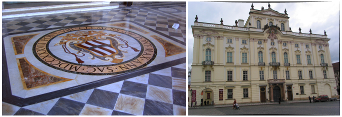
- knowing $\mathbf{d}_{1,2}$ we conclude that $\underline{\mathbf{w}}_i / \|\underline{\mathbf{w}}_i\|$ is the i-th column $\mathbf{r}_i$ of $\mathbf{R}$
- the third column is orthogonal: $\mathbf{r}_3 \simeq \mathbf{r}_1 \times \mathbf{r}_2$

$$\mathbf{R} = \begin{bmatrix} \frac{\underline{\mathbf{w}}_1}{\|\underline{\mathbf{w}}_1\|} & \frac{\underline{\mathbf{w}}_2}{\|\underline{\mathbf{w}}_2\|} & \frac{\underline{\mathbf{w}}_1 \times \underline{\mathbf{w}}_2}{\|\underline{\mathbf{w}}_1 \times \underline{\mathbf{w}}_2\|} \end{bmatrix}$$

- in general we have to care about the signs $\pm\underline{\mathbf{w}}_i$ (such that $\det \mathbf{R} = 1$)



some suitable scenes



Application: Planar Rectification

Principle: Rotate camera (image plane) parallel to the plane of interest.

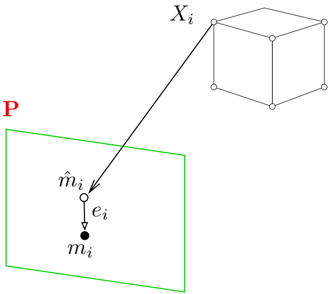
$$\underline{\mathbf{m}} \simeq \mathbf{K}\mathbf{R}\begin{bmatrix}\mathbf{I} & -\mathbf{C}\end{bmatrix}\underline{\mathbf{X}} \qquad \underline{\mathbf{m}}' \simeq \mathbf{K}\begin{bmatrix}\mathbf{I} & -\mathbf{C}\end{bmatrix}\underline{\mathbf{X}}$$

$$\underline{\mathbf{m}}' \simeq \mathbf{K}(\mathbf{K}\mathbf{R})^{-1}\,\underline{\mathbf{m}} = \mathbf{K}\mathbf{R}^\top\mathbf{K}^{-1}\,\underline{\mathbf{m}} = \mathbf{H}\,\underline{\mathbf{m}}$$

- $\mathbf{H}$ is the rectifying homography
- both $\mathbf{K}$ and $\mathbf{R}$ can be calibrated from two finite vanishing points assuming ORUA →57
- not possible when one of them is (or both are) infinite
- without ORUA we would need 4 additional views to calibrate $\mathbf{K}$ as on →54

▶Camera Resection

Camera calibration and orientation from a known set of $k \geq 6$ reference points and their images $\{(X_i, m_i)\}_{i=1}^{6}$.

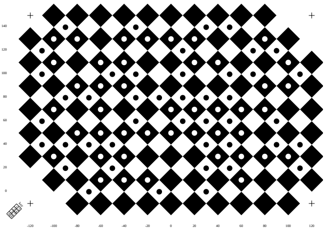


- X_i are considered exact
- m_i is a measurement subject to detection error

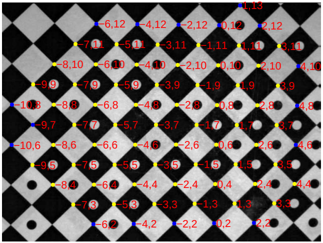
$$\mathbf{m}_i = \hat{\mathbf{m}}_i + \mathbf{e}_i \qquad \text{Cartesian}$$

- where $\lambda_i \, \underline{\hat{\mathbf{m}}}_i = \mathbf{P}\underline{\mathbf{X}}_i$

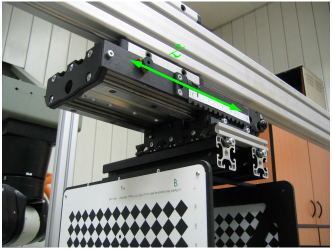
Resection Targets



calibration chart



automatic calibration point detection



resection target with translation stage

- target translated at least once
- by a calibrated (known) translation
- X_i point locations looked up in a table based on their code

▶The Minimal Problem for Camera Resection

Problem: Given $k = 6$ corresponding pairs $\{(X_i, m_i)\}_{i=1}^{k}$, find $\mathbf{P}$

$$\lambda_i \underline{\mathbf{m}}_i = \mathbf{P}\underline{\mathbf{X}}_i, \qquad \mathbf{P} = \begin{bmatrix} \mathbf{q}_1^\top & q_{14} \\ \mathbf{q}_2^\top & q_{24} \\ \mathbf{q}_3^\top & q_{34} \end{bmatrix} \qquad \begin{aligned} \underline{\mathbf{X}}_i &= (x_i, y_i, z_i, 1), \quad i = 1, 2, \ldots, k,\ k = 6 \\ \underline{\mathbf{m}}_i &= (u_i, v_i, 1), \quad \lambda_i \in \mathbb{R},\ \lambda_i \neq 0,\ |\lambda_i| < \infty \end{aligned}$$

easily modifiable for infinite points X_i but be aware of →64

expanded: $\lambda_i u_i = \mathbf{q}_1^\top \mathbf{X}_i + q_{14}, \quad \lambda_i v_i = \mathbf{q}_2^\top \mathbf{X}_i + q_{24}, \quad \lambda_i = \mathbf{q}_3^\top \mathbf{X}_i + q_{34}$

after elimination of λ_i: $(\mathbf{q}_3^\top \mathbf{X}_i + q_{34})u_i = \mathbf{q}_1^\top \mathbf{X}_i + q_{14}, \quad (\mathbf{q}_3^\top \mathbf{X}_i + q_{34})v_i = \mathbf{q}_2^\top \mathbf{X}_i + q_{24}$

Then

$$\mathbf{A}\,\mathbf{q} = \begin{bmatrix} \mathbf{X}_1^\top & 1 & \mathbf{0}^\top & 0 & -u_1\mathbf{X}_1^\top & -u_1 \\ \mathbf{0}^\top & 0 & \mathbf{X}_1^\top & 1 & -v_1\mathbf{X}_1^\top & -v_1 \\ \vdots & & & & & \vdots \\ \mathbf{X}_k^\top & 1 & \mathbf{0}^\top & 0 & -u_k\mathbf{X}_k^\top & -u_k \\ \mathbf{0}^\top & 0 & \mathbf{X}_k^\top & 1 & -v_k\mathbf{X}_k^\top & -v_k \end{bmatrix} \cdot \begin{bmatrix} \mathbf{q}_1 \\ q_{14} \\ \mathbf{q}_2 \\ q_{24} \\ \mathbf{q}_3 \\ q_{34} \end{bmatrix} = \mathbf{0} \tag{9}$$

- we need 11 indepedent parameters for $\mathbf{P}$
- $\mathbf{A} \in \mathbb{R}^{2k,12}$, $\mathbf{q} \in \mathbb{R}^{12}$
- 6 points in a general position give $\operatorname{rank} \mathbf{A} = 12$ and there is no (non-trivial) null space
- drop one row to get rank-11 matrix, then the basis vector of the null space of $\mathbf{A}$ gives $\mathbf{q}$

▶The Jack-Knife Solution for $k = 6$

- given the 6 correspondences, we have 12 equations for the 11 parameters
- can we use all the information present in the 6 points?

Jack-knife estimation

1. $n := 0$
2. for $i = 1, 2, \ldots, 2k$ do
   a) delete i-th row from $\mathbf{A}$, this gives $\mathbf{A}_i$
   b) if $\dim \operatorname{null} \mathbf{A}_i > 1$ continue with the next i
   c) $n := n + 1$
   d) compute the right null-space $\mathbf{q}_i$ of $\mathbf{A}_i$ — e.g. by 'economy-size' SVD
   e) $\hat{\mathbf{q}}_i := \mathbf{q}_i$ normalized to $q_{34} = 1$ and dimension-reduced — assuming finite cam. with $P_{3,4} = 1$
3. from all n vectors $\hat{\mathbf{q}}_i$ collected in Step 1d compute

$$\mathbf{q} = \frac{1}{n}\sum_{i=1}^{n} \hat{\mathbf{q}}_i, \qquad \operatorname{var}[\mathbf{q}] = \frac{n-1}{n} \operatorname{diag} \sum_{i=1}^{n} (\hat{\mathbf{q}}_i - \mathbf{q})(\hat{\mathbf{q}}_i - \mathbf{q})^\top$$

regular for $n \geq 11$

variance of the sample mean

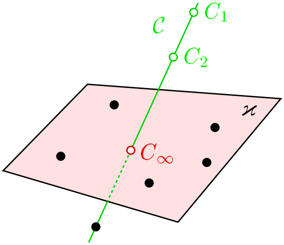
- have a solution + an error estimate, per individual elements of $\mathbf{P}$ (except P_{34})
- at least 5 points must be in a general position (→64)
- large error indicates near degeneracy
- computation not efficient with $k > 6$ points, needs $\binom{2k}{11}$ draws, e.g. $k = 7 \Rightarrow 364$ draws
- better error estimation method: decompose $\mathbf{P}_i$ to $\mathbf{K}_i$, $\mathbf{R}_i$, $\mathbf{t}_i$ (→33), represent $\mathbf{R}_i$ with 3 parameters (e.g. Euler angles, or in exponential map representation →143) and compute the errors for the parameters
- even better: use the $\mathrm{SE}(3)$ Lie group for $(\mathbf{R}_i, \mathbf{t}_i)$ and average its Lie-algebraic representations

▶Degenerate (Critical) Configurations for Camera Resection

Let $\mathcal{X} = \{X_i;\ i = 1, \ldots\}$ be a set of points and $\mathbf{P}_1 \not\simeq \mathbf{P}_j$ be two regular (rank-3) cameras. Then two configurations $(\mathbf{P}_1, \mathcal{X})$ and $(\mathbf{P}_j, \mathcal{X})$ are image-equivalent if

$$\mathbf{P}_1 \underline{\mathbf{X}}_i \simeq \mathbf{P}_j \underline{\mathbf{X}}_i \quad \text{for all} \quad X_i \in \mathcal{X}$$

there is a non-trivial set of other cameras that see the same image



Case 4

Results

- importantly: If all calibration points $X_i \in \mathcal{X}$ lie on a plane $\varkappa$ then camera resection is non-unique and all image-equivalent camera centers lie on a spatial line $\mathcal{C}$ with the $C_\infty = \varkappa \cap \mathcal{C}$ excluded

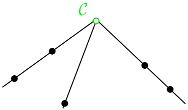
  this also means we cannot resect if all X_i are infinite
- and more: by adding points $X_i \in \mathcal{X}$ to $\mathcal{C}$ we gain nothing
- there are additional image-equivalent configurations, see next

proof sketch in [H&Z, Sec. 22.1.2]

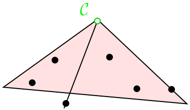
Note that if $\mathbf{Q}$, $\mathbf{T}$ are suitable homographies then $\mathbf{P}_1 \simeq \mathbf{Q}\mathbf{P}_0\mathbf{T}$, where $\mathbf{P}_0$ is canonical and the analysis can be made with $\hat{\mathbf{P}}_j \simeq \mathbf{Q}^{-1}\mathbf{P}_j$

$$\mathbf{P}_0 \underbrace{\mathbf{T}\underline{\mathbf{X}}_i}_{\underline{\mathbf{Y}}_i} \simeq \hat{\mathbf{P}}_j \underbrace{\mathbf{T}\underline{\mathbf{X}}_i}_{\underline{\mathbf{Y}}_i} \quad \text{for all} \quad Y_i \in \mathcal{Y}$$

cont'd (all cases)

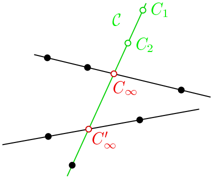


Case 5

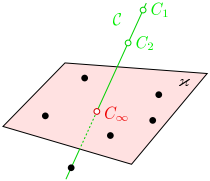


Case 6

- points lie on three optical rays or one optical ray and one optical plane
- cameras C_1, C_2 co-located at point $\mathcal{C}$
- Case 5: camera sees 3 isolated point images
- Case 6: cam. sees a line of points and an isolated point



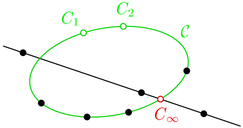
Case 3



Case 4

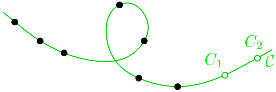
- points lie on a <u>line</u> $\mathcal{C}$ and
  1. on two lines meeting $\mathcal{C}$ at C_∞, C'_∞
  2. or on a plane meeting $\mathcal{C}$ at C_∞
- cameras lie on a line $\mathcal{C} \setminus \{C_\infty, C'_\infty\}$
- Case 3: camera sees 2 lines of points
- Case 4: **dangerous!**

Case 2



- points lie on a <u>planar conic</u> $\mathcal{C}$ and an additional line meeting $\mathcal{C}$ at C_∞
- cameras lie on $\mathcal{C} \setminus \{C_\infty\}$ not necessarily an ellipse
- Case 2: camera sees 2 lines of points

Case 1



- points and cameras all lie on a <u>twisted cubic</u> $\mathcal{C}$
- Case 1: camera sees points on a conic dangerous but unlikely to occur

▶Three-Point Exterior Orientation Problem (P3P)

Calibrated camera rotation and translation from Perspective images of 3 reference Points.

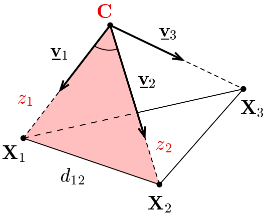
Problem: Given $\mathbf{K}$ and three corresponding pairs $\{(m_i, X_i)\}_{i=1}^{3}$, find $\mathbf{R}$, $\mathbf{C}$ by solving

$$\lambda_i \underline{\mathbf{m}}_i = \mathbf{K}\mathbf{R}\,(\mathbf{X}_i - \mathbf{C}), \qquad i = 1, 2, 3 \qquad\qquad \mathbf{X}_i \text{ Cartesian}$$

1. Transform $\underline{\mathbf{v}}_i \stackrel{\text{def}}{=} \mathbf{K}^{-1}\underline{\mathbf{m}}_i$. Then

$$\lambda_i \underline{\mathbf{v}}_i = \mathbf{R}\,(\mathbf{X}_i - \mathbf{C}). \tag{10}$$

2. If there was no rotation in (10), the situation would look like this



3. and we could shoot 3 lines from the given points $\mathbf{X}_i$ in given directions $\underline{\mathbf{v}}_i$ to get $\mathbf{C}$
4. given $\mathbf{C}$ we solve (10) for λ_i, $\mathbf{R}$

►P3P cont'd

If there is rotation $\mathbf{R}$

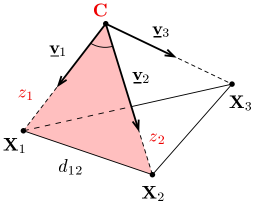
1. Eliminate $\mathbf{R}$ by taking rotation preserves length: $\|\mathbf{Rx}\| = \|\mathbf{x}\|$

$$|\lambda_i| \cdot \|\underline{\mathbf{v}}_i\| = \|\mathbf{X}_i - \mathbf{C}\| \stackrel{\text{def}}{=} z_i \tag{11}$$

2. Consider only angles among $\underline{\mathbf{v}}_i$ and apply Cosine Law per triangle $(\mathbf{C}, \mathbf{X}_i, \mathbf{X}_j)$ $i, j = 1, 2, 3,\ i \neq j$

$$d_{ij}^2 = z_i^2 + z_j^2 - 2\, z_i\, z_j\, c_{ij},$$

$$z_i = \|\mathbf{X}_i - \mathbf{C}\|,\ \ d_{ij} = \|\mathbf{X}_j - \mathbf{X}_i\|,\ \ c_{ij} = \cos(\angle \underline{\mathbf{v}}_i\, \underline{\mathbf{v}}_j)$$



4. Solve the system of 3 quadratic eqs in 3 unknowns z_i

[Fischler & Bolles, 1981]

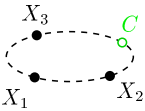
there may be no real root; there are up to 4 solutions that cannot be ignored (verify on additional points)

5. Compute $\mathbf{C}$ by trilateration (3-sphere intersection) from $\mathbf{X}_i$ and z_i; then λ_i from (11)

6. Compute $\mathbf{R}$ from (10) we will solve this problem next →70

Similar problems (P4P with unknown f) at `http://aag.ciirc.cvut.cz/minimal/` (papers, code)

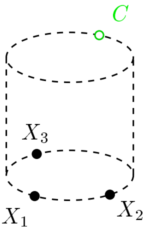
Degenerate (Critical) Configurations for Exterior Orientation



no solution

1. C cocyclic with (X_1, X_2, X_3) camera sees points on a line

---



unstable solution

- center of projection C located on the orthogonal circular cylinder with base circumscribing the three points X_i

unstable: a small change of X_i results in a large change of C
can be detected by error propagation

degenerate

- camera C is coplanar with points (X_1, X_2, X_3) but is not on the circumscribed circle of (X_1, X_2, X_3)

camera sees points on a line

- additional critical configurations depend on the quadratic equations solver

[Haralick et al. IJCV 1994]

►Populating A Little ZOO of Minimal Geometric Problems in CV

problem	given	unknown	slide
camera resection	6 world–img correspondences $\left\{(X_i,\, m_i)\right\}_{i=1}^{6}$	$\mathbf{P}$	→62
exterior orientation	$\mathbf{K}$, 3 world–img correspondences $\left\{(X_i,\, m_i)\right\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{C}$	→66
relative orientation	3 world-world correspondences $\left\{(X_i,\, Y_i)\right\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{t}$	→70

- camera resection and exterior orientation are similar problems in a sense:
  - we do resectioning when our camera is uncalibrated
  - we do orientation when our camera is calibrated
- relative orientation involves no camera (see next) it is a recurring problem in 3D vision
- more problems to come

▶The Relative Orientation Problem

Problem: Given point triples (X_1, X_2, X_3) and (Y_1, Y_2, Y_3) in a general position in $\mathbf{R}^3$ such that the correspondence $X_i \leftrightarrow Y_i$ is known, determine the relative orientation $(\mathbf{R}, \mathbf{t})$ that maps $\mathbf{X}_i$ to $\mathbf{Y}_i$, i.e.

$$\mathbf{Y}_i = \mathbf{R}\mathbf{X}_i + \mathbf{t}, \quad i = 1, 2, 3\,.$$

Applies to:

- 3D scanners
- merging partial reconstructions from different viewpoints
- generalization of the last step of P3P

Obs: Let the centroid be $\bar{\mathbf{X}} = \frac{1}{3}\sum_i \mathbf{X}_i$ and analogically for $\bar{\mathbf{Y}}$. Then

$$\bar{\mathbf{Y}} = \mathbf{R}\bar{\mathbf{X}} + \mathbf{t}.$$

Therefore

$$\mathbf{Z}_i \stackrel{\text{def}}{=} (\mathbf{Y}_i - \bar{\mathbf{Y}}) = \mathbf{R}(\mathbf{X}_i - \bar{\mathbf{X}}) \stackrel{\text{def}}{=} \mathbf{R}\mathbf{W}_i$$

If all dot products are equal, $\mathbf{Z}_i^\top \mathbf{Z}_j = \mathbf{W}_i^\top \mathbf{W}_j$ for $i, j = 1, 2, 3$, we have

$$\mathbf{R}^* = \begin{bmatrix}\mathbf{W}_1 & \mathbf{W}_2 & \mathbf{W}_3\end{bmatrix}^{-1} \begin{bmatrix}\mathbf{Z}_1 & \mathbf{Z}_2 & \mathbf{Z}_3\end{bmatrix}$$

Poor man's solver:

- normalize $\mathbf{W}_i$, $\mathbf{Z}_i$ to unit length and then use the above formula
- but this is equivalent to a non-optimal objective — it ignores errors in vector lengths

An Optimal Algorithm for Relative Orientation

We setup a minimization problem

$$\mathbf{R}^* = \arg\min_{\mathbf{R}} \sum_{i=1}^{3} \|\mathbf{Z}_i - \mathbf{R}\mathbf{W}_i\|^2 \quad \text{s.t.} \quad \mathbf{R}^\top\mathbf{R} = \mathbf{I}, \quad \det\mathbf{R} = 1$$

$$\arg\min_{\mathbf{R}} \sum_i \|\mathbf{Z}_i - \mathbf{R}\mathbf{W}_i\|^2 = \arg\min_{\mathbf{R}} \sum_i \left(\|\mathbf{Z}_i\|^2 - 2\mathbf{Z}_i^\top\mathbf{R}\mathbf{W}_i + \|\mathbf{W}_i\|^2\right) = \cdots$$

$$\cdots = \arg\max_{\mathbf{R}} \sum_i \mathbf{Z}_i^\top\mathbf{R}\mathbf{W}_i$$

Obs 1: Let $\mathbf{A} : \mathbf{B} = \sum_{i,j} a_{ij}b_{ij}$ be the dot-product (Frobenius inner product) over real matrices. Then

$$\mathbf{A} : \mathbf{B} = \mathbf{B} : \mathbf{A} = \operatorname{tr}(\mathbf{A}^\top\mathbf{B})$$

Obs 2: (cyclic property for matrix trace)

$$\operatorname{tr}(\mathbf{ABC}) = \operatorname{tr}(\mathbf{CAB})$$

Obs 3: ($\mathbf{Z}_i$, $\mathbf{W}_i$ are vectors)

$$\mathbf{Z}_i^\top\mathbf{R}\mathbf{W}_i = \operatorname{tr}(\mathbf{Z}_i^\top\mathbf{R}\mathbf{W}_i) \overset{\text{O2}}{=} \operatorname{tr}(\mathbf{W}_i\mathbf{Z}_i^\top\mathbf{R}) \overset{\text{O1}}{=} (\mathbf{Z}_i\mathbf{W}_i^\top) : \mathbf{R} = \mathbf{R} : (\mathbf{Z}_i\mathbf{W}_i^\top)$$

Let there be SVD of

$$\sum_i \mathbf{Z}_i\mathbf{W}_i^\top \overset{\text{def}}{=} \mathbf{M} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$$

Then

$$\mathbf{R} : \mathbf{M} = \mathbf{R} : (\mathbf{U}\mathbf{D}\mathbf{V}^\top) \overset{\text{O1}}{=} \operatorname{tr}(\mathbf{R}^\top\mathbf{U}\mathbf{D}\mathbf{V}^\top) \overset{\text{O2}}{=} \operatorname{tr}(\mathbf{V}^\top\mathbf{R}^\top\mathbf{U}\mathbf{D}) \overset{\text{O1}}{=} (\mathbf{U}^\top\mathbf{R}\mathbf{V}) : \mathbf{D}$$

cont'd: The Algorithm

We are solving

$$\mathbf{R}^* = \arg\max_{\mathbf{R}} \sum_i \mathbf{Z}_i^\top \mathbf{R} \mathbf{W}_i = \arg\max_{\mathbf{R}} \left(\mathbf{U}^\top \mathbf{R} \mathbf{V}\right) : \mathbf{D}$$

A particular solution is found as follows:

- $\mathbf{U}^\top \mathbf{R} \mathbf{V}$ must be (1) orthogonal, and most similar to (2) diagonal, (3) positive definite
- Since $\mathbf{U}$, $\mathbf{V}$ are orthogonal matrices then the solution to the problem is among $\mathbf{R}^* = \mathbf{U}\mathbf{S}\mathbf{V}^\top$, where $\mathbf{S}$ is diagonal and orthogonal, i.e. one of

$$\pm\,\mathrm{diag}(1,1,1), \quad \pm\,\mathrm{diag}(1,-1,-1), \quad \pm\,\mathrm{diag}(-1,1,-1), \quad \pm\,\mathrm{diag}(-1,-1,1)$$

- $\mathbf{U}^\top \mathbf{V}$ is not necessarily positive definite
- We choose $\mathbf{S}$ so that $(\mathbf{R}^*)^\top \mathbf{R}^* = \mathbf{I}$

Alg:

1. Compute matrix $\mathbf{M} = \sum_i \mathbf{Z}_i \mathbf{W}_i^\top$.
2. Compute SVD $\mathbf{M} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$.
3. Compute all $\mathbf{R}_k = \mathbf{U}\mathbf{S}_k\mathbf{V}^\top$ that give $\mathbf{R}_k^\top \mathbf{R}_k = \mathbf{I}$.
4. Compute $\mathbf{t}_k = \bar{\mathbf{Y}} - \mathbf{R}_k \bar{\mathbf{X}}$.

- The algorithm can be used for more than 3 points
- Triple pairs can be pre-filtered based on motion invariants (lengths, angles)
- Can be used for the last step of the exterior orientation (P3P) problem →66

Module IV

Computing with a Camera Pair

4.1 Camera Motions Inducing Epipolar Geometry

4.2 Estimating Fundamental Matrix from 7 Correspondences

4.3 Estimating Essential Matrix from 5 Correspondences

4.4 Triangulation: 3D Point Position from a Pair of Corresponding Points

covered by

[1] [H&Z] Secs: 9.1, 9.2, 9.6, 11.1, 11.2, 11.9, 12.2, 12.3, 12.5.1

[2] H. Li and R. Hartley. Five-point motion estimation made easy. In *Proc ICPR* 2006, pp. 630–633

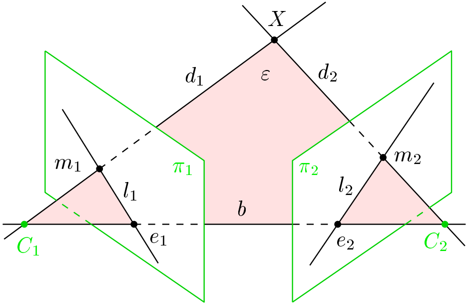
additional references

H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293 (5828):133–135, 1981.

▶Geometric Model of a Camera Stereo Pair

Epipolar geometry:

- brings constraints necessary for inter-image matching
- its parametric form encapsulates information about the relative pose of two cameras



two-camera setup

Description

- baseline b joins projection centers C_1, C_2
$$\mathbf{b} = \mathbf{C}_2 - \mathbf{C}_1$$
- epipole $e_i \in \pi_i$ is the image of C_j:
$$\underline{\mathbf{e}}_1 \simeq \mathbf{P}_1 \underline{\mathbf{C}}_2, \quad \underline{\mathbf{e}}_2 \simeq \mathbf{P}_2 \underline{\mathbf{C}}_1$$
- $l_i \in \pi_i$ is the image of epipolar plane
$$\varepsilon = (C_2, X, C_1)$$
- l_j is the epipolar line ('epipolar') in image π_j induced by m_i in image π_i

Epipolar constraint: corresponding d_2, b, d_1 are coplanar

a necessary condition →87

$$\mathbf{P}_i = \begin{bmatrix}\mathbf{Q}_i & \mathbf{q}_i\end{bmatrix} = \mathbf{K}_i \begin{bmatrix}\mathbf{R}_i & \mathbf{t}_i\end{bmatrix} = \mathbf{K}_i \mathbf{R}_i \begin{bmatrix}\mathbf{I} & -\mathbf{C}_i\end{bmatrix} \quad i = 1, 2$$ →31

Epipolar Geometry Example: Forward Motion

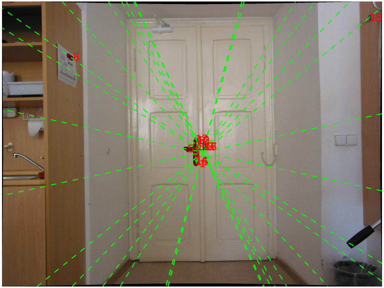


image 1

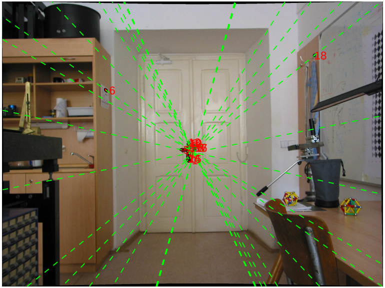
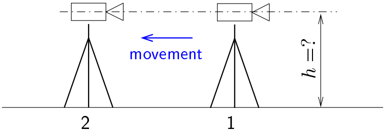


image 2

- red: correspondences
- green: epipolar line pairs per correspondence

click on the image to see their IDs
same ID in both images

How high was the camera above the floor?



▶Cross Products and Maps by Skew-Symmetric 3×3 Matrices

- There is an equivalence $\mathbf{b} \times \mathbf{m} = [\mathbf{b}]_\times \mathbf{m}$, where $[\mathbf{b}]_\times$ is a 3×3 skew-symmetric matrix

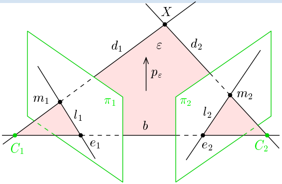
$$[\mathbf{b}]_\times = \begin{bmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{bmatrix}, \qquad \text{assuming} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

Some properties

1. $[\mathbf{b}]_\times^\top = -[\mathbf{b}]_\times$ — the general antisymmetry property
2. $\mathbf{A}$ is skew-symmetric iff $\mathbf{x}^\top \mathbf{A} \mathbf{x} = 0$ for all $\mathbf{x}$ — skew-sym mtx generalizes cross products
3. $[\mathbf{b}]_\times^3 = -\|\mathbf{b}\|^2 \cdot [\mathbf{b}]_\times$
4. $\big\|[\mathbf{b}]_\times\big\|_F = \sqrt{2}\,\|\mathbf{b}\|$ — Frobenius norm ($\|\mathbf{A}\|_F = \sqrt{\operatorname{tr}(\mathbf{A}^\top \mathbf{A})} = \sqrt{\sum_{i,j} |a_{ij}|^2}$)
5. $\operatorname{rank}[\mathbf{b}]_\times = 2$ iff $\|\mathbf{b}\| > 0$ — check minors of $[\mathbf{b}]_\times$
6. $[\mathbf{b}]_\times \mathbf{b} = \mathbf{0}$
7. eigenvalues of $[\mathbf{b}]_\times$ are $(0, \lambda, -\lambda)$
8. for any 3×3 regular $\mathbf{B}$: $\mathbf{B}^\top [\mathbf{B}\mathbf{z}]_\times \mathbf{B} = \det \mathbf{B}\, [\mathbf{z}]_\times$ — follows from the factoring on →39
9. in particular: if $\mathbf{R}\mathbf{R}^\top = \mathbf{I}$ then $[\mathbf{R}\mathbf{b}]_\times = \mathbf{R}[\mathbf{b}]_\times \mathbf{R}^\top$

- note that if $\mathbf{R}_b$ is rotation about $\mathbf{b}$ then $\mathbf{R}_b \mathbf{b} = \mathbf{b}$
- note $[\mathbf{b}]_\times$ is not a homography; it is not a rotation matrix — it is the logarithm of a rotation mtx

►Expressing Epipolar Constraint Algebraically



$$\mathbf{P}_i = \begin{bmatrix}\mathbf{Q}_i & \mathbf{q}_i\end{bmatrix} = \mathbf{K}_i \begin{bmatrix}\mathbf{R}_i & \mathbf{t}_i\end{bmatrix},\ i = 1, 2$$

$\mathbf{R}_{21}$ – relative camera rotation, $\mathbf{R}_{21} = \mathbf{R}_2\mathbf{R}_1^\top$

$\mathbf{t}_{21}$ – relative camera translation, $\mathbf{t}_{21} = \mathbf{t}_2 - \mathbf{R}_{21}\mathbf{t}_1 = -\mathbf{R}_2\mathbf{b}$ →74

$\mathbf{b}$ – baseline vector (world coordinate system)

remember: $\mathbf{C} = -\mathbf{Q}^{-1}\mathbf{q} = -\mathbf{R}^\top\mathbf{t}$ →33 and 35

$$\boxed{0 = \mathbf{d}_2^\top \underbrace{\mathbf{p}_\varepsilon}_{\text{normal of } \varepsilon} \simeq \underbrace{(\mathbf{Q}_2^{-1}\underline{\mathbf{m}}_2)^\top}_{\text{optical ray}} \underbrace{\mathbf{Q}_1^\top \underline{\mathbf{l}}_1}_{\text{optical plane}} = \underline{\mathbf{m}}_2^\top \underbrace{\mathbf{Q}_2^{-\top}\mathbf{Q}_1^\top(\underline{\mathbf{e}}_1 \times \underline{\mathbf{m}}_1)}_{\text{image of } \varepsilon \text{ in } \pi_2} = \underline{\mathbf{m}}_2^\top \underbrace{\left(\mathbf{Q}_2^{-\top}\mathbf{Q}_1^\top [\underline{\mathbf{e}}_1]_\times\right)}_{\text{fundamental matrix } \mathbf{F}} \underline{\mathbf{m}}_1}$$

Epipolar constraint $\underline{\mathbf{m}}_2^\top \mathbf{F}\, \underline{\mathbf{m}}_1 = 0$ is a point-line incidence constraint

- point $\underline{\mathbf{m}}_2$ is incident on epipolar line $\underline{\mathbf{l}}_2 \simeq \mathbf{F}\underline{\mathbf{m}}_1$
- point $\underline{\mathbf{m}}_1$ is incident on epipolar line $\underline{\mathbf{l}}_1 \simeq \mathbf{F}^\top\underline{\mathbf{m}}_2$
- $\mathbf{F}\underline{\mathbf{e}}_1 = \mathbf{F}^\top\underline{\mathbf{e}}_2 = \mathbf{0}$ (non-trivially)
- all epipolars meet at the epipole

$$\underline{\mathbf{e}}_1 \simeq \mathbf{Q}_1\mathbf{C}_2 + \mathbf{q}_1 = \mathbf{Q}_1\mathbf{C}_2 - \mathbf{Q}_1\mathbf{C}_1 = \mathbf{K}_1\mathbf{R}_1\mathbf{b} = -\mathbf{K}_1\mathbf{R}_1\mathbf{R}_2^\top\mathbf{t}_{21} = -\mathbf{K}_1\mathbf{R}_{21}^\top\mathbf{t}_{21}$$

$$\mathbf{F} = \mathbf{Q}_2^{-\top}\mathbf{Q}_1^\top[\underline{\mathbf{e}}_1]_\times = \mathbf{Q}_2^{-\top}\mathbf{Q}_1^\top[-\mathbf{K}_1\mathbf{R}_{21}^\top\mathbf{t}_{21}]_\times \overset{\circledast 1}{=} \cdots \simeq \mathbf{K}_2^{-\top}[-\mathbf{t}_{21}]_\times\mathbf{R}_{21}\mathbf{K}_1^{-1}$$ fundamental

$$\mathbf{E} = [-\mathbf{t}_{21}]_\times\mathbf{R}_{21} = \underbrace{[\mathbf{R}_2\mathbf{b}]_\times}_{\text{baseline in Cam 2}}\mathbf{R}_{21} = \mathbf{R}_{21}\underbrace{[\mathbf{R}_1\mathbf{b}]_\times}_{\text{baseline in Cam 1}} = \mathbf{R}_{21}[-\mathbf{R}_{21}^\top\mathbf{t}_{21}]_\times$$ essential

▶The Structure and the Key Properties of the Fundamental Matrix

$$\mathbf{F} = (\underbrace{\mathbf{Q}_2\mathbf{Q}_1^{-1}}_{\text{epipolar homography } \mathbf{H}_e})^{-\top}[\underline{\mathbf{e}}_1]_\times = \underbrace{\mathbf{K}_2^{-\top}\mathbf{R}_{21}\mathbf{K}_1^{\top}}_{\mathbf{H}_e^{-\top}}\overbrace{[\underline{\mathbf{e}}_1]_\times}^{\text{left epipole}} \overset{\to 76}{\simeq} \overbrace{[\mathbf{H}_e\underline{\mathbf{e}}_1]_\times}^{\text{right epipole}}\mathbf{H}_e = \mathbf{K}_2^{-\top}\underbrace{[-\mathbf{t}_{21}]_\times\mathbf{R}_{21}}_{\text{essential matrix } \mathbf{E}}\mathbf{K}_1^{-1}$$

1. $\mathbf{E}$ captures relative camera pose only [Longuet-Higgins 1981]

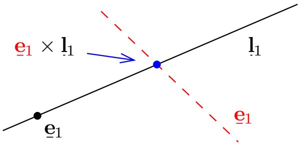
(the change of the world coordinate system does not change $\mathbf{E}$)

$$\begin{bmatrix}\mathbf{R}_i' & \mathbf{t}_i'\end{bmatrix} = \begin{bmatrix}\mathbf{R}_i & \mathbf{t}_i\end{bmatrix} \cdot \begin{bmatrix}\mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1\end{bmatrix} = \begin{bmatrix}\mathbf{R}_i\mathbf{R} & \mathbf{R}_i\mathbf{t} + \mathbf{t}_i\end{bmatrix},$$

then

$$\mathbf{R}_{21}' = \mathbf{R}_2'\mathbf{R}_1'^{\top} = \cdots = \mathbf{R}_{21} \qquad \mathbf{t}_{21}' = \mathbf{t}_2' - \mathbf{R}_{21}'\mathbf{t}_1' = \cdots = \mathbf{t}_{21}$$

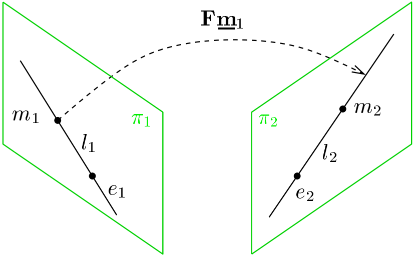
2. the translation length $\mathbf{t}_{21}$ is lost since $\mathbf{E}$ is homogeneous
3. $\mathbf{F}$ maps points to lines and it is not a homography
4. $\mathbf{H}_e$ maps epipoles to epipoles, $\mathbf{H}_e^{-\top}$ epipolar lines to epipolar lines: $\underline{\mathbf{l}}_2 \simeq \mathbf{H}_e^{-\top}\underline{\mathbf{l}}_1$



another epipolar line map: $\underline{\mathbf{l}}_2 \simeq \mathbf{F}[\underline{\mathbf{e}}_1]_\times\underline{\mathbf{l}}_1$

- proof by point/line 'transmutation' (left)
- point $\underline{\mathbf{e}}_1$ does not lie on line $\underline{\mathbf{e}}_1$ (dashed): $\underline{\mathbf{e}}_1^\top\underline{\mathbf{e}}_1 \neq 0$
- $\mathbf{F}[\underline{\mathbf{e}}_1]_\times$ is not a homography, unlike $\mathbf{H}_e^{-\top}$ but it does the same job for epipolar line mapping
- no need to decompose $\mathbf{F}$ to obtain $\mathbf{H}_e$

►Summary: Relations and Mappings Involving Fundamental Matrix



$$0 = \underline{\mathbf{m}}_2^\top \mathbf{F}\,\underline{\mathbf{m}}_1$$

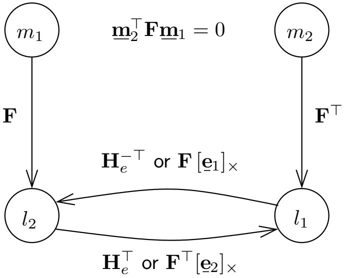
$$\underline{\mathbf{e}}_1 \simeq \operatorname{null}(\mathbf{F}), \qquad \underline{\mathbf{e}}_2 \simeq \operatorname{null}(\mathbf{F}^\top)$$

$$\underline{\mathbf{e}}_1 \simeq \mathbf{H}_e^{-1}\underline{\mathbf{e}}_2 \qquad \underline{\mathbf{e}}_2 \simeq \mathbf{H}_e\underline{\mathbf{e}}_1$$

$$\underline{\mathbf{l}}_1 \simeq \mathbf{F}^\top\underline{\mathbf{m}}_2 \qquad \underline{\mathbf{l}}_2 \simeq \mathbf{F}\underline{\mathbf{m}}_1$$

$$\underline{\mathbf{l}}_1 \simeq \mathbf{H}_e^\top\underline{\mathbf{l}}_2 \qquad \underline{\mathbf{l}}_2 \simeq \mathbf{H}_e^{-\top}\underline{\mathbf{l}}_1$$

$$\underline{\mathbf{l}}_1 \simeq \mathbf{F}^\top[\underline{\mathbf{e}}_2]_\times\underline{\mathbf{l}}_2 \qquad \underline{\mathbf{l}}_2 \simeq \mathbf{F}[\underline{\mathbf{e}}_1]_\times\underline{\mathbf{l}}_1$$



- $\mathbf{F}[\underline{\mathbf{e}}_1]_\times$ maps epipolar lines to epipolar lines but it is not a homography
- $\mathbf{H}_e = \mathbf{Q}_2\mathbf{Q}_1^{-1}$ is the epipolar homography→78
  $\mathbf{H}_e^{-\top}$ maps epipolar lines to epipolar lines, where

$$\mathbf{H}_e = \mathbf{Q}_2\mathbf{Q}_1^{-1} = \mathbf{K}_2\mathbf{R}_{21}\mathbf{K}_1^{-1}$$

you have seen this →59

▶Representation Theorem for Fundamental Matrices

Def: $\mathbf{F}$ is fundamental when $\mathbf{F} \simeq \mathbf{H}^{-\top}[\underline{\mathbf{e}}_1]_\times$, where $\mathbf{H}$ is regular and $\underline{\mathbf{e}}_1 \simeq \operatorname{null} \mathbf{F} \neq \mathbf{0}$.

Theorem: A 3×3 matrix $\mathbf{A}$ is fundamental iff it is of rank 2.

Proof.

Direct: By the geometry, $\mathbf{H}$ is full-rank, $\underline{\mathbf{e}}_1 \neq \mathbf{0}$, hence $\mathbf{H}^{-\top}[\underline{\mathbf{e}}_1]_\times$ is a 3×3 matrix of rank 2.

Converse:

1. let $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ be the SVD of $\mathbf{A}$ of rank 2; then $\mathbf{D} = \operatorname{diag}(\lambda_1, \lambda_2, 0)$, $\lambda_1 \geq \lambda_2 > 0$
2. we write $\mathbf{D} = \mathbf{B}\mathbf{C}$, where $\mathbf{B} = \operatorname{diag}(\lambda_1, \lambda_2, \lambda_3)$, $\mathbf{C} = \operatorname{diag}(1, 1, 0)$
3. then $\mathbf{A} = \mathbf{U}\mathbf{B}\mathbf{C}\mathbf{V}^\top = \mathbf{U}\mathbf{B}\mathbf{C}\underbrace{\mathbf{W}\mathbf{W}^\top}_{\mathbf{I}}\mathbf{V}^\top$ with $\mathbf{W}$ rotation
4. we look for a rotation $\mathbf{W}$ that maps $\mathbf{C}$ to a skew-symmetric $\mathbf{S}$, i.e. $\mathbf{S} = \mathbf{C}\mathbf{W}$
5. then $\mathbf{W} = \begin{bmatrix} 0 & \alpha & 0 \\ -\alpha & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $|\alpha| = 1$, and $\mathbf{S} = [\mathbf{s}]_\times$, $\mathbf{s} = (0, 0, 1)$
6. we write

   $\mathbf{v}_3$ – 3rd column of $\mathbf{V}$, $\mathbf{u}_3$ – 3rd column of $\mathbf{U}$

$$\mathbf{A} = \mathbf{U}\mathbf{B}[\mathbf{s}]_\times \mathbf{W}^\top \mathbf{V}^\top = \overset{\circledast\,1}{\cdots} = \underbrace{\mathbf{U}\mathbf{B}(\mathbf{V}\mathbf{W})^\top}_{\simeq \mathbf{H}^{-\top}} [\mathbf{v}_3]_\times \simeq \underbrace{[\mathbf{H}\mathbf{v}_3]_\times}_{\simeq [\mathbf{u}_3]_\times} \mathbf{H}, \tag{12}$$

7. $\mathbf{H}$ regular, $\mathbf{A}\mathbf{v}_3 = \mathbf{0}$, $\mathbf{u}_3\mathbf{A} = \mathbf{0}$ for $\mathbf{v}_3 \neq \mathbf{0}$, $\mathbf{u}_3 \neq \mathbf{0}$ □

- we also got a (non-unique: α, λ_3) decomposition formula for fundamental matrices
- it follows there is no constraint on $\mathbf{F}$ except for the rank

►Representation Theorem for Essential Matrices

Theorem

Let $\mathbf{E}$ *be a* 3×3 *matrix with SVD* $\mathbf{E} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$*. Then* $\mathbf{E}$ *is essential iff* $\mathbf{D} \simeq \mathrm{diag}(1, 1, 0)$*.*

Proof.

Direct:

If $\mathbf{E}$ is an essential matrix, then the epipolar homography matrix is a rotation matrix (→78), hence $\mathbf{H}^{-\top} \simeq \mathbf{U}\mathbf{B}(\mathbf{V}\mathbf{W})^\top$ in (12) must be (λ-scaled) orthogonal, therefore $\mathbf{B} = \lambda\mathbf{I}$.

we have fixed the missing λ_3 in (12)

Then

$$\mathbf{R}_{21} = \mathbf{H}^{-\top} \simeq \mathbf{U}\mathbf{W}^\top\mathbf{V}^\top \simeq \mathbf{U}\mathbf{W}\mathbf{V}^\top$$

Converse:

$\mathbf{E}$ is fundamental with

$$\mathbf{D} = \mathrm{diag}(\lambda, \lambda, 0) = \underbrace{\lambda\mathbf{I}}_{\mathbf{B}}\,\underbrace{\mathrm{diag}(1, 1, 0)}_{\mathbf{D}}$$

then $\mathbf{B} = \lambda\mathbf{I}$ in (12) and $\mathbf{U}(\mathbf{V}\mathbf{W})^\top$ is orthogonal, as required. □

▶Essential Matrix Decomposition

We are decomposing $\mathbf{E}$ to $\mathbf{E} \simeq [-\mathbf{t}_{21}]_\times \mathbf{R}_{21} = \mathbf{R}_{21}[-\mathbf{R}_{21}^\top \mathbf{t}_{21}]_\times$ [H&Z, sec. 9.6]

1. compute SVD of $\mathbf{E} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ and verify $\mathbf{D} = \lambda\,\mathrm{diag}(1,1,0)$
2. ensure $\mathbf{U}$, $\mathbf{V}$ are rotation matrices by $\mathbf{U} \mapsto \det(\mathbf{U})\mathbf{U}$, $\mathbf{V} \mapsto \det(\mathbf{V})\mathbf{V}$
3. compute

$$\mathbf{R}_{21} = \mathbf{U} \underbrace{\begin{bmatrix} 0 & \alpha & 0 \\ -\alpha & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{W}} \mathbf{V}^\top, \quad \mathbf{t}_{21} = -\beta\,\mathbf{u}_3, \qquad |\alpha| = 1, \quad \beta \neq 0 \tag{13}$$

Notes

- $\mathbf{v}_3 \simeq \mathbf{R}_{21}^\top \mathbf{t}_{21}$ by (12), hence $\mathbf{R}_{21}\mathbf{v}_3 \simeq \mathbf{t}_{21} \simeq \mathbf{u}_3$ since it must fall in left null space by $\mathbf{E} \simeq [\mathbf{u}_3]_\times \mathbf{R}_{21}$
- $\mathbf{t}_{21}$ is recoverable up to scale β and direction $\operatorname{sign}\beta$
- the result for $\mathbf{R}_{21}$ is unique up to $\alpha = \pm 1$ despite non-uniqueness of SVD
- the change of sign in α rotates the solution by 180° about $\mathbf{t}_{21}$

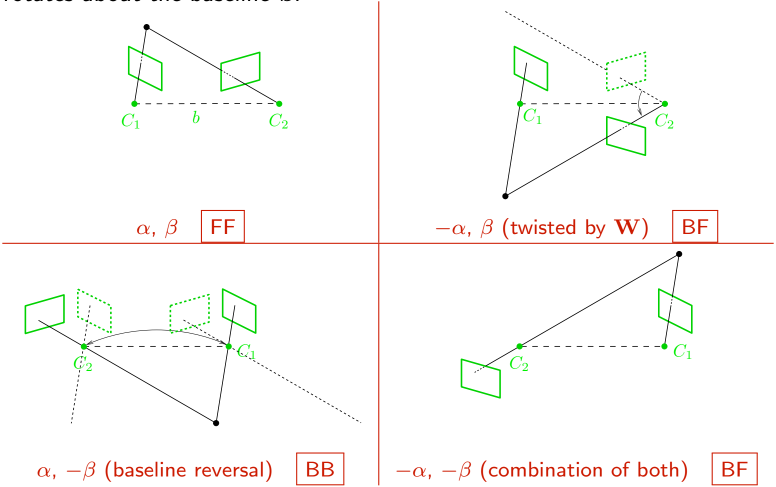
  $\mathbf{R}(\alpha) = \mathbf{U}\mathbf{W}\mathbf{V}^\top$, $\mathbf{R}(-\alpha) = \mathbf{U}\mathbf{W}^\top\mathbf{V}^\top \Rightarrow \mathbf{T} = \mathbf{R}(-\alpha)\mathbf{R}^\top(\alpha) = \cdots = \mathbf{U}\,\mathrm{diag}(-1,-1,1)\mathbf{U}^\top$
  which is a rotation by 180° about $\mathbf{u}_3 \simeq \mathbf{t}_{21}$: show that $\mathbf{u}_3$ is the rotation axis

$$\mathbf{U}\,\mathrm{diag}(-1,-1,1)\mathbf{U}^\top \mathbf{u}_3 = \mathbf{U} \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \mathbf{u}_3$$

- 4 solution sets for 4 sign combinations of α, β see next for geometric interpretation

▶Four Solutions to Essential Matrix Decomposition

Transform the world coordinate system so that the origin is in Camera 2. Then $\mathbf{t}_{21} = -\mathbf{b}$ and $\mathbf{W}$ rotates about the baseline $\mathbf{b}$. →77



- chirality constraint: all 3D points are in front of both cameras
- this singles-out the upper left case [H&Z, Sec. 9.6.3]

Problem: Given a set $\{(x_i, y_i)\}_{i=1}^{k}$ of $k = 7$ finite correspondences, estimate f. m. $\mathbf{F}$.

$$\underline{\mathbf{y}}_i^\top \mathbf{F}\, \underline{\mathbf{x}}_i = 0, \quad i = 1, \dots, k, \qquad \underline{\text{known}}: \quad \underline{\mathbf{x}}_i = (u_i^1, v_i^1, 1), \quad \underline{\mathbf{y}}_i = (u_i^2, v_i^2, 1)$$

terminology: correspondence = truth, later: match = algorithm's result; hypothesized corresp.

Solution:

$$\underline{\mathbf{y}}_i^\top \mathbf{F}\, \underline{\mathbf{x}}_i = (\underline{\mathbf{y}}_i \underline{\mathbf{x}}_i^\top) : \mathbf{F} = \left(\operatorname{vec}(\underline{\mathbf{y}}_i \underline{\mathbf{x}}_i^\top)\right)^\top \operatorname{vec}(\mathbf{F}),$$

rotation property of matrix trace →71

$$\operatorname{vec}(\mathbf{F}) = \begin{bmatrix} f_{11} & f_{21} & f_{31} & \dots & f_{33} \end{bmatrix}^\top \in \mathbb{R}^9$$

column vector from matrix

$$\mathbf{D} = \begin{bmatrix} \left(\operatorname{vec}(\mathbf{y}_1 \mathbf{x}_1^\top)\right)^\top \\ \left(\operatorname{vec}(\mathbf{y}_2 \mathbf{x}_2^\top)\right)^\top \\ \left(\operatorname{vec}(\mathbf{y}_3 \mathbf{x}_3^\top)\right)^\top \\ \vdots \\ \left(\operatorname{vec}(\mathbf{y}_k \mathbf{x}_k^\top)\right)^\top \end{bmatrix} = \begin{bmatrix} u_1^1 u_1^2 & u_1^1 v_1^2 & u_1^1 & u_1^2 v_1^1 & v_1^1 v_1^2 & v_1^1 & u_1^2 & v_1^2 & 1 \\ u_2^1 u_2^2 & u_2^1 v_2^2 & u_2^1 & u_2^2 v_2^1 & v_2^1 v_2^2 & v_2^1 & u_2^2 & v_2^2 & 1 \\ u_3^1 u_3^2 & u_3^1 v_3^2 & u_3^1 & u_3^2 v_3^1 & v_3^1 v_3^2 & v_3^1 & u_3^2 & v_3^2 & 1 \\ \vdots & & & & & & & & \vdots \\ u_k^1 u_k^2 & u_k^1 v_k^2 & u_k^1 & u_k^2 v_k^1 & v_k^1 v_k^2 & v_k^1 & u_k^2 & v_k^2 & 1 \end{bmatrix} \in \mathbb{R}^{k,9}$$

$$\boxed{\mathbf{D}\operatorname{vec}(\mathbf{F}) = \mathbf{0}}$$

▶7-Point Algorithm Continued

$$\mathbf{D}\,\mathrm{vec}(\mathbf{F}) = \mathbf{0}, \quad \mathbf{D} \in \mathbb{R}^{k,9}$$

- for $k = 7$ we have a rank-deficient system, the null-space of $\mathbf{D}$ is 2-dimensional
- but we know that $\det \mathbf{F} = 0$, hence
  1. find a basis of the null space of $\mathbf{D}$: $\mathbf{F}_1$, $\mathbf{F}_2$ — by SVD or QR factorization
  2. get up to 3 real solutions for α from

     $$\det(\alpha \mathbf{F}_1 + (1-\alpha)\mathbf{F}_2) = 0 \qquad \text{cubic equation in } \alpha$$
  3. get up to 3 fundamental matrices $\mathbf{F}_i = \alpha_i \mathbf{F}_1 + (1-\alpha_i)\mathbf{F}_2$
  4. if $\operatorname{rank} \mathbf{F}_i < 2$ for all $i = 1, 2, 3$ then fail

- the result may depend on image (domain) transformations
- normalization improves conditioning →92
- this gives a good starting point for the full algorithm →111
- dealing with mismatches need not be a part of the 7-point algorithm →112

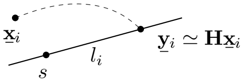
▶Degenerate Configurations for Fundamental Matrix Estimation

When is $\mathbf{F}$ not uniquely determined from any number of correspondences? [H&Z, Sec. 11.9]

1. when images are related by homography
   a) camera centers coincide $\mathbf{t}_{21} = 0$: $\mathbf{H} = \mathbf{K}_2 \mathbf{R}_{21} \mathbf{K}_1^{-1}$ $\mathbf{H}$ – as in epipolar homography
   b) camera moves but all 3D points lie in a plane $(\mathbf{n}, d)$: $\mathbf{H} = \mathbf{K}_2(\mathbf{R}_{21} - \mathbf{t}_{21}\mathbf{n}^\top/d)\mathbf{K}_1^{-1}$

   - in both cases: epipolar geometry is not defined
   - we get an arbitrary solution from the 7-point algorithm, in the form of $\mathbf{F} = [\underline{\mathbf{s}}]_\times \mathbf{H}$

   note that $[\underline{\mathbf{s}}]_\times \mathbf{H} \simeq \mathbf{H}'[\underline{\mathbf{s}}']_\times$ →76



   - given (arbitrary, fixed) $\underline{\mathbf{s}}$
   - and correspondence $x_i \leftrightarrow y_i$
   - y_i is the image of x_i: $\underline{\mathbf{y}}_i \simeq \mathbf{H}\underline{\mathbf{x}}_i$
   - a necessary condition: $y_i \in l_i$, $\quad \underline{\mathbf{l}}_i \simeq \underline{\mathbf{s}} \times \mathbf{H}\underline{\mathbf{x}}_i$

$$0 = \underline{\mathbf{y}}_i^\top(\underline{\mathbf{s}} \times \mathbf{H}\underline{\mathbf{x}}_i) = \underline{\mathbf{y}}_i^\top[\underline{\mathbf{s}}]_\times \mathbf{H}\underline{\mathbf{x}}_i \quad \text{for any } \underline{\mathbf{x}}_i, \underline{\mathbf{y}}_i, \underline{\mathbf{s}} \ (!)$$

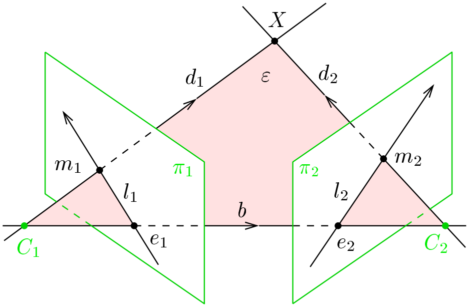
2. both camera centers and all 3D points lie on a ruled quadric
   hyperboloid of one sheet, cones, cylinders, two planes
   - there are 3 solutions for $\mathbf{F}$

notes

- estimation of $\mathbf{E}$ can deal with planes: $[\underline{\mathbf{s}}]_\times \mathbf{H}$ is essential, then $\mathbf{H} = \mathbf{R} - \mathbf{t}\mathbf{n}^\top/d$, and $\underline{\mathbf{s}} \simeq \mathbf{t}$ not arbitrary
- a complete treatment with additional degenerate configurations in [H&Z, sec. 22.2]
- a stronger epipolar constraint could reject some configurations

A Note on Oriented Epipolar Constraint

- a tighter epipolar constraint preserves orientations
- requires all points and cameras be on the same side of the plane at infinity



$$(\underline{\mathbf{e}}_2 \times \underline{\mathbf{m}}_2) \overset{+}{\simeq} \mathbf{F}\,\underline{\mathbf{m}}_1$$

notation: $\underline{\mathbf{m}} \overset{+}{\simeq} \underline{\mathbf{n}}$ means $\underline{\mathbf{m}} = \lambda \underline{\mathbf{n}}$, $\lambda > 0$

- we can read the constraint as $(\underline{\mathbf{e}}_2 \times \underline{\mathbf{m}}_2) \overset{+}{\simeq} \mathbf{H}_e^{-\top}\,(\mathbf{e}_1 \times \underline{\mathbf{m}}_1)$
- note that the constraint is not invariant to the change of either sign of $\underline{\mathbf{m}}_i$
- all 7 correspondence in 7-point alg. must have the same sign — see later
- this may help reject some wrong matches, see $\rightarrow$112 — [Chum et al. 2004]
- an even more tight constraint: scene points in front of both cameras — expensive
  this is called chirality constraint

▶5-Point Algorithm for Relative Camera Orientation

Problem: Given $\{m_i, m_i'\}_{i=1}^{5}$ corresponding image points and calibration matrix $\mathbf{K}$, recover the camera motion $\mathbf{R}$, $\mathbf{t}$.

Obs:

1. $\mathbf{E}$ – homogeneous 3×3 matrix; 9 numbers up to scale
2. $\mathbf{R}$ – 3 DOF, $\mathbf{t}$ – 2 DOF only, in total 5 DOF $\rightarrow$ we need $9 - 1 - 5 = 3$ constraints on $\mathbf{E}$
3. idea: $\mathbf{E}$ essential iff it has two equal singular values and the third is zero $\rightarrow$81

This gives an equation system:

$$\underline{\mathbf{v}}_i^\top \mathbf{E}\, \underline{\mathbf{v}}_i' = 0 \qquad \text{5 linear constraints } (\underline{\mathbf{v}} \simeq \mathbf{K}^{-1}\underline{\mathbf{m}})$$

$$\det \mathbf{E} = 0 \qquad \text{1 cubic constraint}$$

$$\mathbf{E}\mathbf{E}^\top\mathbf{E} - \frac{1}{2}\operatorname{tr}(\mathbf{E}\mathbf{E}^\top)\mathbf{E} = \mathbf{0} \qquad \text{9 cubic constraints, 2 independent}$$

⊛ P1; 1pt: verify this equation from $\mathbf{E} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$, $\mathbf{D} = \lambda\, \mathrm{diag}(1, 1, 0)$

1. estimate $\mathbf{E}$ by SVD from $\underline{\mathbf{v}}_i^\top \mathbf{E}\, \underline{\mathbf{v}}_i' = 0$ by the null-space method — 4D null space
2. this gives $\mathbf{E} \simeq x\mathbf{E}_1 + y\mathbf{E}_2 + z\mathbf{E}_3 + \mathbf{E}_4$
3. at most 10 (complex) solutions for x, y, z from the cubic constraints

- when all 3D points lie on a plane: at most 2 real solutions (twisted-pair) — can be disambiguated in 3 views or by chirality constraint ($\rightarrow$83) unless all 3D points are closer to one camera
- 6-point problem for unknown f — [Kukelova et al. BMVC 2008]
- resources at `http://aag.ciirc.cvut.cz/minimal/`

►The Triangulation Problem

Problem: Given cameras $\mathbf{P}_1$, $\mathbf{P}_2$ and a correspondence $x \leftrightarrow y$ compute a 3D point $\underline{\mathbf{X}}$ projecting to x and y

$$\lambda_1\, \underline{\mathbf{x}} = \mathbf{P}_1 \underline{\mathbf{X}}, \qquad \lambda_2\, \underline{\mathbf{y}} = \mathbf{P}_2 \underline{\mathbf{X}}, \qquad \underline{\mathbf{x}} = \begin{bmatrix} u^1 \\ v^1 \\ 1 \end{bmatrix}, \qquad \underline{\mathbf{y}} = \begin{bmatrix} u^2 \\ v^2 \\ 1 \end{bmatrix}, \qquad \mathbf{P}_i = \begin{bmatrix} (\mathbf{p}_1^i)^\top \\ (\mathbf{p}_2^i)^\top \\ (\mathbf{p}_3^i)^\top \end{bmatrix}$$

Linear triangulation method after eliminating λ_1, λ_2

$$u^1\,(\mathbf{p}_3^1)^\top \underline{\mathbf{X}} = (\mathbf{p}_1^1)^\top \underline{\mathbf{X}}, \qquad u^2\,(\mathbf{p}_3^2)^\top \underline{\mathbf{X}} = (\mathbf{p}_1^2)^\top \underline{\mathbf{X}},$$

$$v^1\,(\mathbf{p}_3^1)^\top \underline{\mathbf{X}} = (\mathbf{p}_2^1)^\top \underline{\mathbf{X}}, \qquad v^2\,(\mathbf{p}_3^2)^\top \underline{\mathbf{X}} = (\mathbf{p}_2^2)^\top \underline{\mathbf{X}}$$

Gives

$$\mathbf{D}\underline{\mathbf{X}} = \mathbf{0}, \qquad \mathbf{D} = \begin{bmatrix} u^1\,(\mathbf{p}_3^1)^\top - (\mathbf{p}_1^1)^\top \\ v^1\,(\mathbf{p}_3^1)^\top - (\mathbf{p}_2^1)^\top \\ u^2\,(\mathbf{p}_3^2)^\top - (\mathbf{p}_1^2)^\top \\ v^2\,(\mathbf{p}_3^2)^\top - (\mathbf{p}_2^2)^\top \end{bmatrix}, \qquad \mathbf{D} \in \mathbb{R}^{4,4}, \quad \underline{\mathbf{X}} \in \mathbb{R}^4 \tag{14}$$

- typically, $\mathbf{D}$ has full rank (!)
- what else: back-projected rays will generally not intersect due to image error, see next
- what else: using Jack-knife (→63) not recommended sensitive to small error
- idea: we will step back and use SVD (→90)
- but the result will not be invariant to projective frame
  replacing $\mathbf{P}_1 \mapsto \mathbf{P}_1\mathbf{H}$, $\mathbf{P}_2 \mapsto \mathbf{P}_2\mathbf{H}$ does not always result in $\underline{\mathbf{X}} \mapsto \mathbf{H}^{-1}\underline{\mathbf{X}}$
- note the homogeneous form in (14) can represent points $\underline{\mathbf{X}}$ at infinity

▶The Least-Squares Triangulation by SVD

- if $\mathbf{D}$ is full-rank we may minimize the algebraic least-squares error

$$\varepsilon^2(\underline{\mathbf{X}}) = \|\mathbf{D}\underline{\mathbf{X}}\|^2 \quad \text{s.t.} \quad \|\underline{\mathbf{X}}\| = 1, \qquad \underline{\mathbf{X}} \in \mathbb{R}^4$$

- let $\mathbf{d}_i$ be the i-th row of $\mathbf{D}$ taken as a column vector, then

$$\|\mathbf{D}\underline{\mathbf{X}}\|^2 = \sum_{i=1}^{4} (\mathbf{d}_i^\top \underline{\mathbf{X}})^2 = \sum_{i=1}^{4} \underline{\mathbf{X}}^\top \mathbf{d}_i \mathbf{d}_i^\top \underline{\mathbf{X}} = \underline{\mathbf{X}}^\top \mathbf{Q}\, \underline{\mathbf{X}}, \text{ where } \mathbf{Q} = \sum_{i=1}^{4} \mathbf{d}_i \mathbf{d}_i^\top = \mathbf{D}^\top \mathbf{D} \in \mathbb{R}^{4,4}$$

- we write the SVD of $\mathbf{Q}$ as $\mathbf{Q} = \sum_{j=1}^{4} \sigma_j^2\, \mathbf{u}_j \mathbf{u}_j^\top$, in which [Golub & van Loan 2013, Sec. 2.5]

$$\sigma_1^2 \geq \cdots \geq \sigma_4^2 \geq 0 \quad \text{and} \quad \mathbf{u}_l^\top \mathbf{u}_m = \begin{cases} 0 & \text{if } l \neq m \\ 1 & \text{otherwise} \end{cases}$$

- then $\underline{\mathbf{X}} = \arg \min_{\mathbf{q}, \|\mathbf{q}\|=1} \mathbf{q}^\top \mathbf{Q}\, \mathbf{q} = \mathbf{u}_4$ the last column of the $\mathbf{U}$ matrix from $\mathrm{SVD}(\mathbf{D}^\top \mathbf{D})$

Proof (by contradiction).

Let $\bar{\mathbf{q}} = \sum_{i=1}^{4} a_i \mathbf{u}_i$ s.t. $\sum_{i=1}^{4} a_i^2 = 1$, then $\|\bar{\mathbf{q}}\| = 1$, as desired, and

$$\bar{\mathbf{q}}^\top \mathbf{Q}\, \bar{\mathbf{q}} = \sum_{j=1}^{4} \sigma_j^2\, \bar{\mathbf{q}}^\top \mathbf{u}_j\, \mathbf{u}_j^\top \bar{\mathbf{q}} = \sum_{j=1}^{4} \sigma_j^2\, (\mathbf{u}_j^\top \bar{\mathbf{q}})^2 = \cdots = \sum_{j=1}^{4} a_j^2 \sigma_j^2 \;\geq\; \sum_{j=1}^{4} a_j^2 \sigma_4^2 = \sigma_4^2$$

since $\sigma_j \geq \sigma_4$ □

- if $\sigma_4 \ll \sigma_3$, there is a unique solution $\underline{\mathbf{X}} = \mathbf{u}_4$ with residual error $(\mathbf{D}\,\underline{\mathbf{X}})^2 = \sigma_4^2$
  the quality (conditioning) of the solution may be expressed as $q = \sigma_3/\sigma_4$ (greater is better)

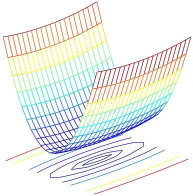
Matlab code for the least-squares solver:

```
[U,O,V] = svd(D);
X = V(:,end);
q = sqrt(O(end-1,end-1)/O(end,end));
```

⊛ P1; 1pt: Why did we decompose $\mathbf{D}$ here, and not $\mathbf{Q} = \mathbf{D}^\top\mathbf{D}$?

►Numerical Conditioning

- The equation $\mathbf{D}\underline{\mathbf{X}} = \mathbf{0}$ in (14) may be ill-conditioned for numerical computation, which results in a poor estimate for $\underline{\mathbf{X}}$.



Why: on a row of $\mathbf{D}$ there are big entries together with small entries, e.g. of orders projection centers in mm, image points in px

$$\begin{bmatrix} 10^3 & 0 & 10^3 & 10^6 \\ 0 & 10^3 & 10^3 & 10^6 \\ 10^3 & 0 & 10^3 & 10^6 \\ 0 & 10^3 & 10^3 & 10^6 \end{bmatrix}$$

Quick fix:

1. re-scale the problem by a regular diagonal conditioning matrix $\mathbf{S} \in \mathbb{R}^{4,4}$

$$\mathbf{0} = \mathbf{D}\,\underline{\mathrm{m}X} = \mathbf{D}\,\mathbf{S}\,\mathbf{S}^{-1}\underline{\mathrm{m}X} = \bar{\mathbf{D}}\,\underline{\mathrm{m}\bar{X}}$$

   choose $\mathbf{S}$ to make the entries in $\hat{\mathbf{D}}$ all smaller than unity in absolute value:

$$\mathbf{S} = \mathrm{diag}(10^{-3}, 10^{-3}, 10^{-3}, 10^{-6})$$

```
S = diag(1./max(abs(D), 1))
```

2. solve for $\underline{\mathrm{m}\bar{X}}$ as before
3. get the final solution as $\underline{\mathrm{m}X} = \mathbf{S}\,\underline{\mathrm{m}\bar{X}}$

- when SVD is used in camera resection, conditioning is essential for success →62

Algebraic Error vs Reprojection Error

- algebraic error (c – camera index, (u^c, v^c) – image coordinates) from SVD →91

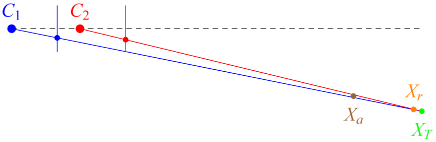
$$\varepsilon^2(\underline{\mathbf{X}}) = \sigma_4^2 = \sum_{c=1}^{2} \left[\left(u^c (\mathbf{p}_3^c)^\top \underline{\mathbf{X}} - (\mathbf{p}_1^c)^\top \underline{\mathbf{X}} \right)^2 + \left(v^c (\mathbf{p}_3^c)^\top \underline{\mathbf{X}} - (\mathbf{p}_2^c)^\top \underline{\mathbf{X}} \right)^2 \right]$$

- reprojection error

$$e^2(\underline{\mathbf{X}}) = \sum_{c=1}^{2} \left[\left(u^c - \frac{(\mathbf{p}_1^c)^\top \underline{\mathbf{X}}}{(\mathbf{p}_3^c)^\top \underline{\mathbf{X}}} \right)^2 + \left(v^c - \frac{(\mathbf{p}_2^c)^\top \underline{\mathbf{X}}}{(\mathbf{p}_3^c)^\top \underline{\mathbf{X}}} \right)^2 \right]$$

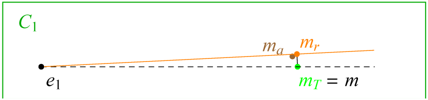
- algebraic error zero ⇔ reprojection error zero $\qquad \sigma_4 = 0 \Rightarrow$ non-trivial null space
- epipolar constraint satisfied ⇒ equivalent results
- in general: minimizing algebraic error is cheap but it gives inferior results
- minimizing reprojection error is expensive but it gives good results
- the midpoint of the common perpendicular to both optical rays gives about 50% greater error in 3D
- the golden standard method – deferred to →106

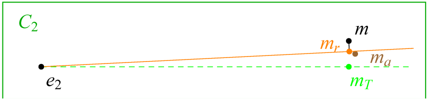
Ex:



- forward camera motion
- error $f/50$ in image 2, orthogonal to epipolar plane

X_T – noiseless ground truth position
X_r – reprojection error minimizer
X_a – algebraic error minimizer
m – measurement (m_T with noise in v^2)





▶We Have Added to The ZOO (cont'd from →69)

problem	given	unknown	slide
camera resection	6 world–img correspondences $\left\{(X_i, m_i)\right\}_{i=1}^{6}$	$\mathbf{P}$	62
exterior orientation	$\mathbf{K}$, 3 world–img correspondences $\left\{(X_i, m_i)\right\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{t}$	66
relative pointcloud orientation	3 world-world correspondences $\left\{(X_i, Y_i)\right\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{t}$	70
fundamental matrix	7 img–img correspondences $\left\{(m_i, m_i')\right\}_{i=1}^{7}$	$\mathbf{F}$	84
relative camera orientation	$\mathbf{K}$, 5 img–img correspondences $\left\{(m_i, m_i')\right\}_{i=1}^{5}$	$\mathbf{R}$, $\mathbf{t}$	88
triangulation	$\mathbf{P}_1$, $\mathbf{P}_2$, 1 img–img correspondence (m_i, m_i')	X	89

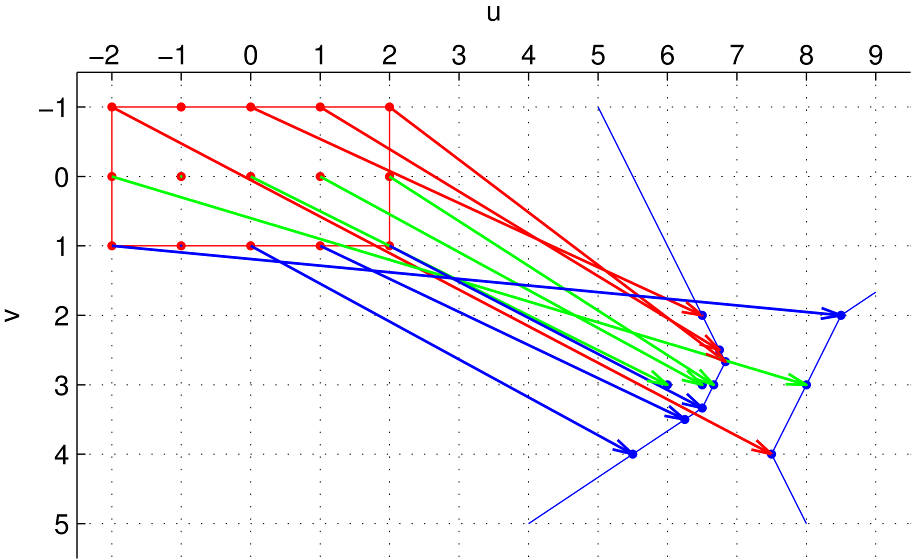
A bigger ZOO at `http://cmp.felk.cvut.cz/minimal/`

calibrated problems

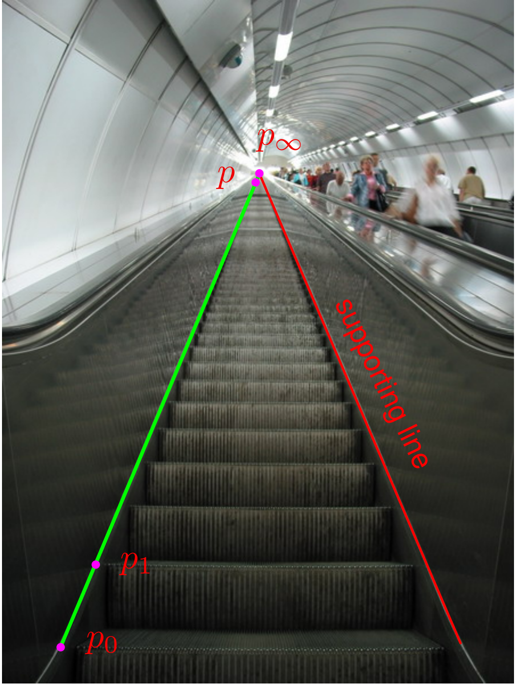
- have fewer degenerate configurations
- can do with fewer points (good for geometry proposal generators →119)

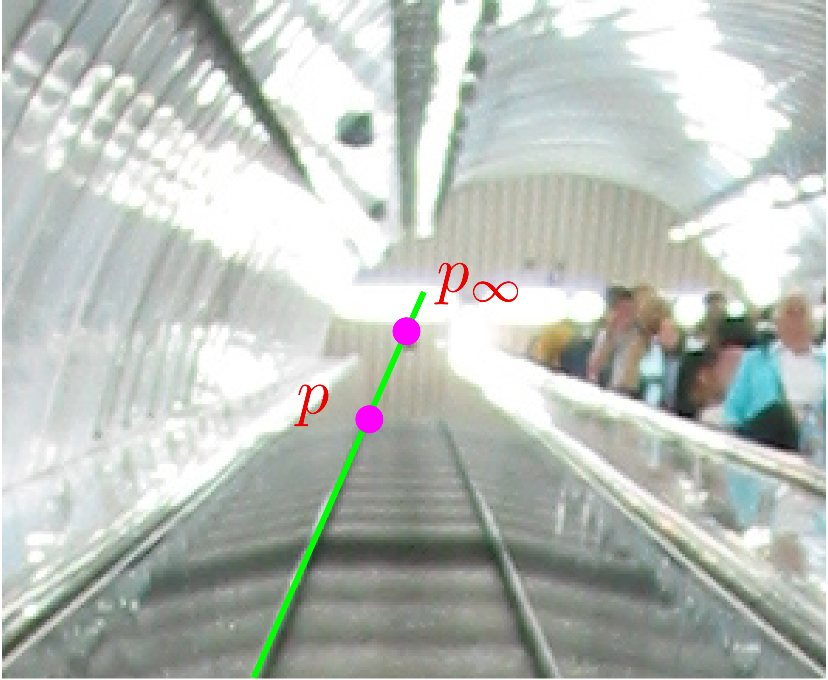
- algebraic error optimization (SVD) makes sense in camera resection and triangulation only
- but it is not the best method; we will now focus on 'optimizing optimally'

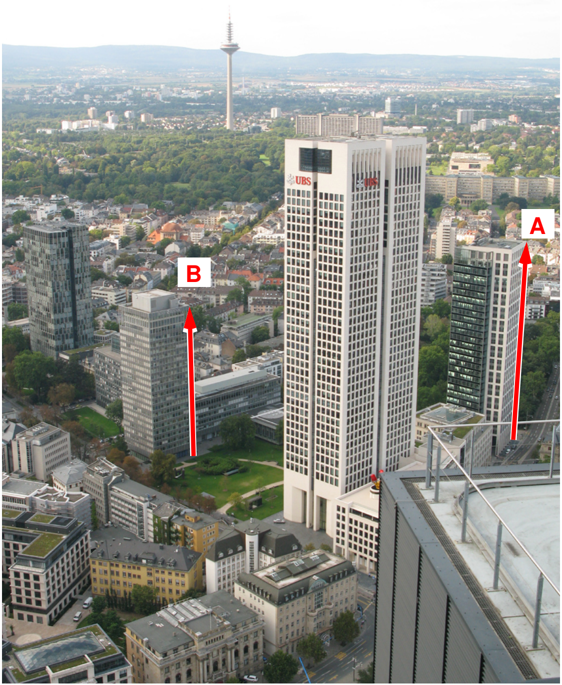
Thank You

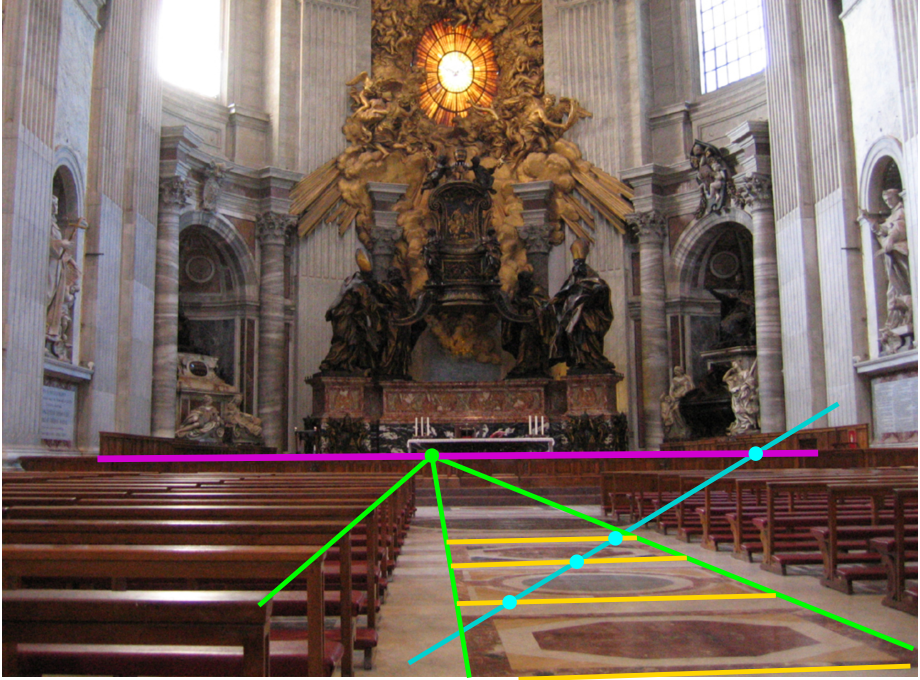


u
−2
−1
0
1
2
3
4
5
6
7
8
9
v
−1
0
1
2
3
4
5


p_∞
p
supporting line
p_1
p_0

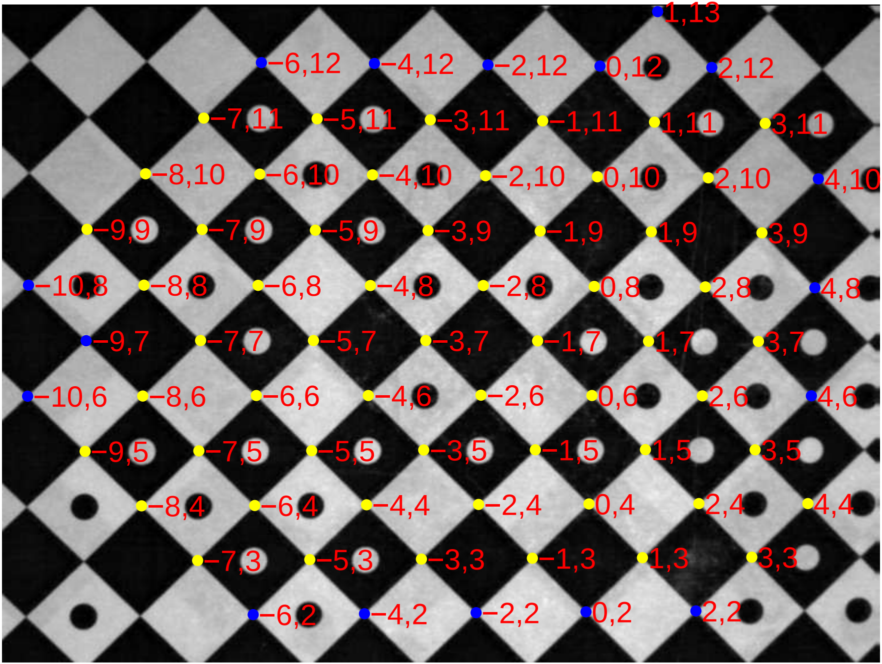

p_∞
p

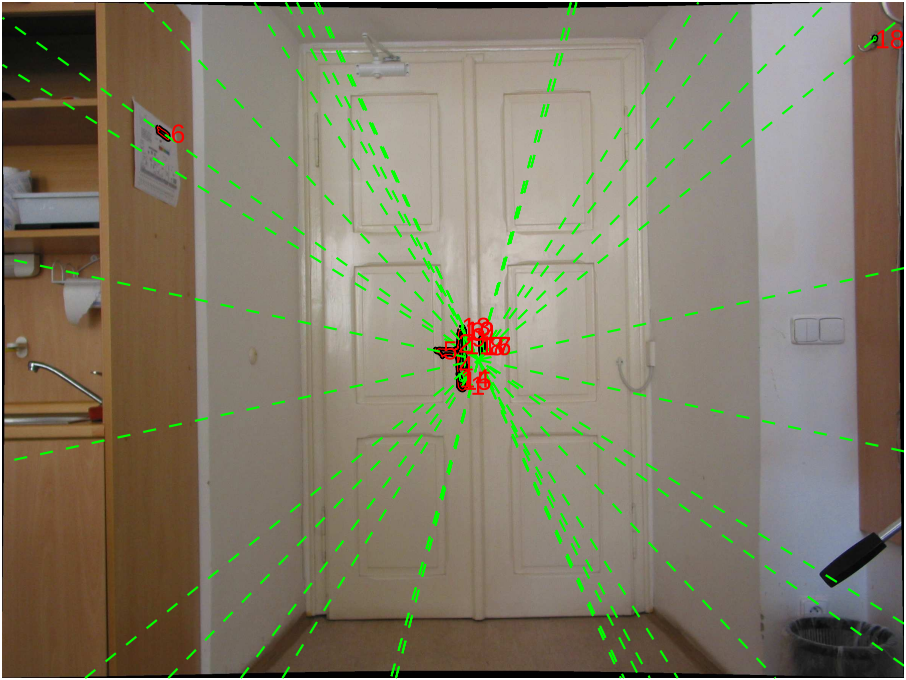

UBS
A
B

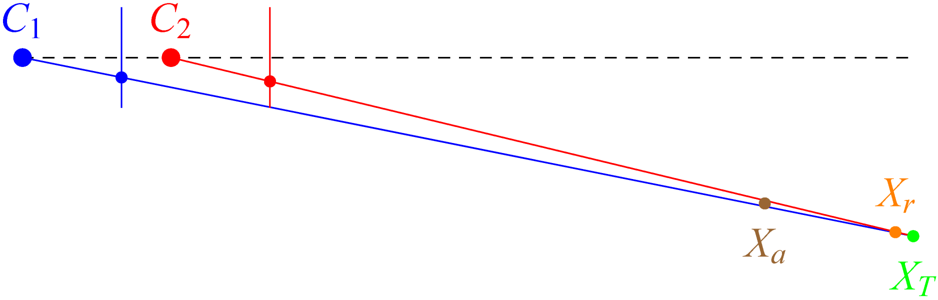


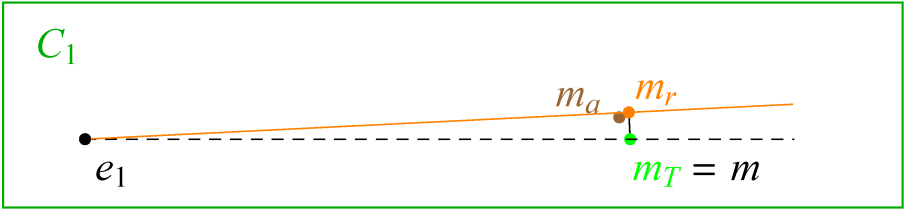

CLEMENS·XII·PONT·MAX·ANN·SAC·MDCCXXXVII·PONT·VIII

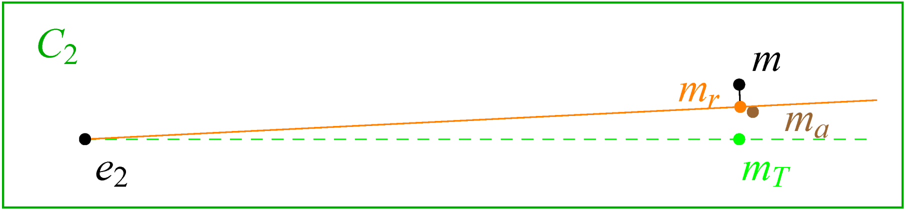

BOOKS
FILMS
TICKETS
BOOKS


1,13
−6,12 −4,12 −2,12 0,12 2,12
−7,11 −5,11 −3,11 −1,11 1,11 3,11
−8,10 −6,10 −4,10 −2,10 0,10 2,10 4,10
−9,9 −7,9 −5,9 −3,9 −1,9 1,9 3,9
−10,8 −8,8 −6,8 −4,8 −2,8 0,8 2,8 4,8
−9,7 −7,7 −5,7 −3,7 −1,7 1,7 3,7
−10,6 −8,6 −6,6 −4,6 −2,6 0,6 2,6 4,6
−9,5 −7,5 −5,5 −3,5 −1,5 1,5 3,5
−8,4 −6,4 −4,4 −2,4 0,4 2,4 4,4
−7,3 −5,3 −3,3 −1,3 1,3 3,3
−6,2 −4,2 −2,2 0,2 2,2


6
18


C_1
C_2
X_r
X_a
X_T


C_1
m_a
m_r
e_1
$m_T = m$


C_2
m
m_r
m_a
e_2
m_T