UNIVERSITÀ DI PARMA

Single View Reconstruction

Summary

UNIVERSITÀ DI PARMA

- Calibration math recap
- Vanishing points and lines
- Calibration using a single view
- Estimation of geometry using a single view

Credits

UNIVERSITÀ DI PARMA

- [FP] D. A. Forsyth and J. Ponce. **Computer Vision: A Modern Approach (2nd Edition)**. Prentice Hall, 2011.
- [HZ] R. Hartley and A. Zisserman. **Multiple View Geometry in Computer Vision. Cambridge University Press**, 2003.
- CS231A · **Computer Vision: from 3D reconstruction to recognition, Prof. Silvio Savarese** – Stanford University

UNIVERSITÀ DI PARMA

From pixels to world...

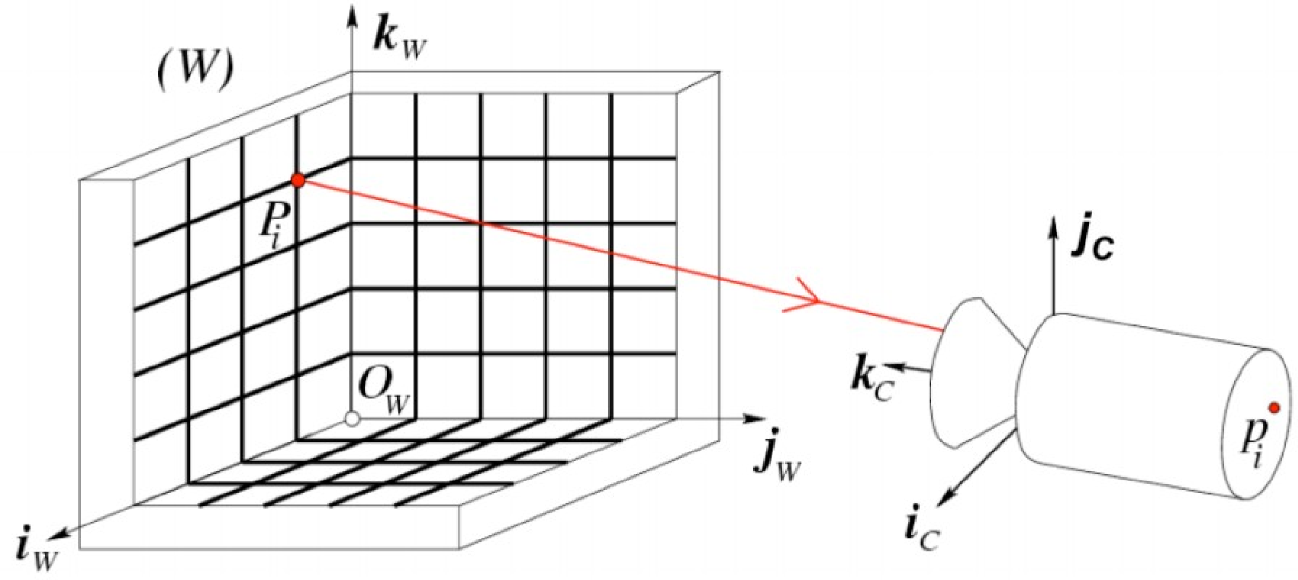
Introduction

UNIVERSITÀ DI PARMA

- We now should know how the 3D points are mapped inside the image plane
- We computed a M matrix (K + RT) that helps the mapping
- Let's try to do the opposite
  - Given a single image can we compute the 3D geometry?

Camera model

UNIVERSITÀ DI PARMA



$$p_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = M \; P_i$$

In pixels

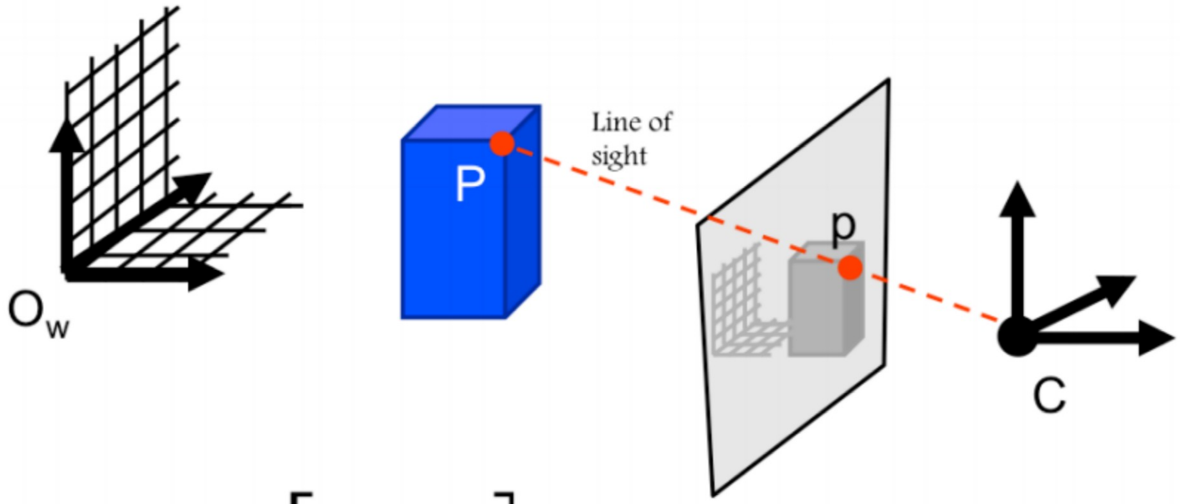
World ref. system

$$M = K\begin{bmatrix} R & T \end{bmatrix}$$

11 unknowns

Need at least 6 correspondences

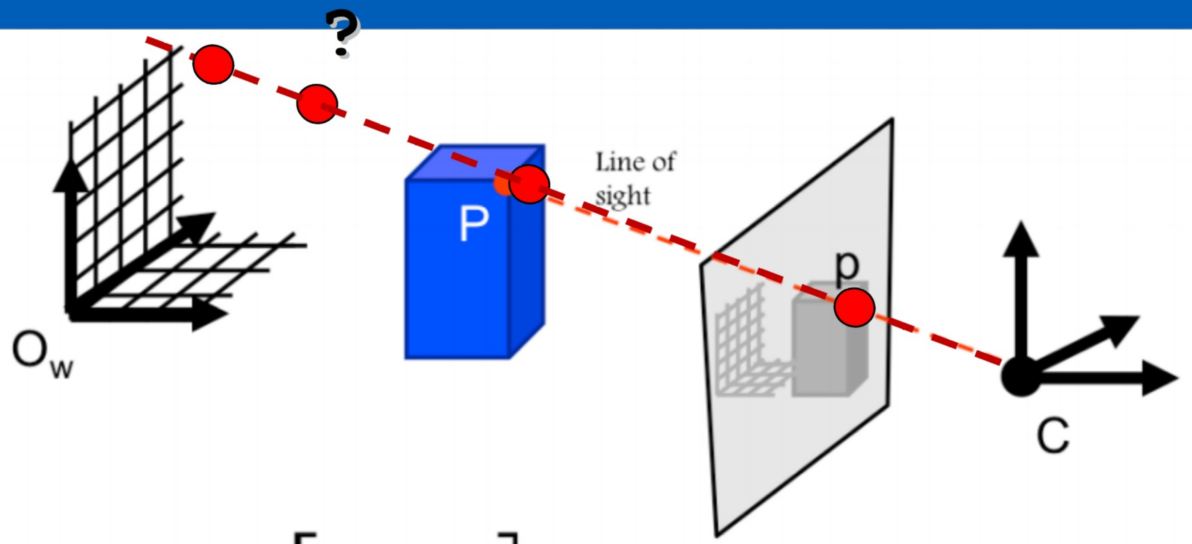
Camera model



$$M = K[R \quad T]$$

- After calibration we know the transformation world $\longrightarrow$ camera
  - We can discover p if we know P

Camera model

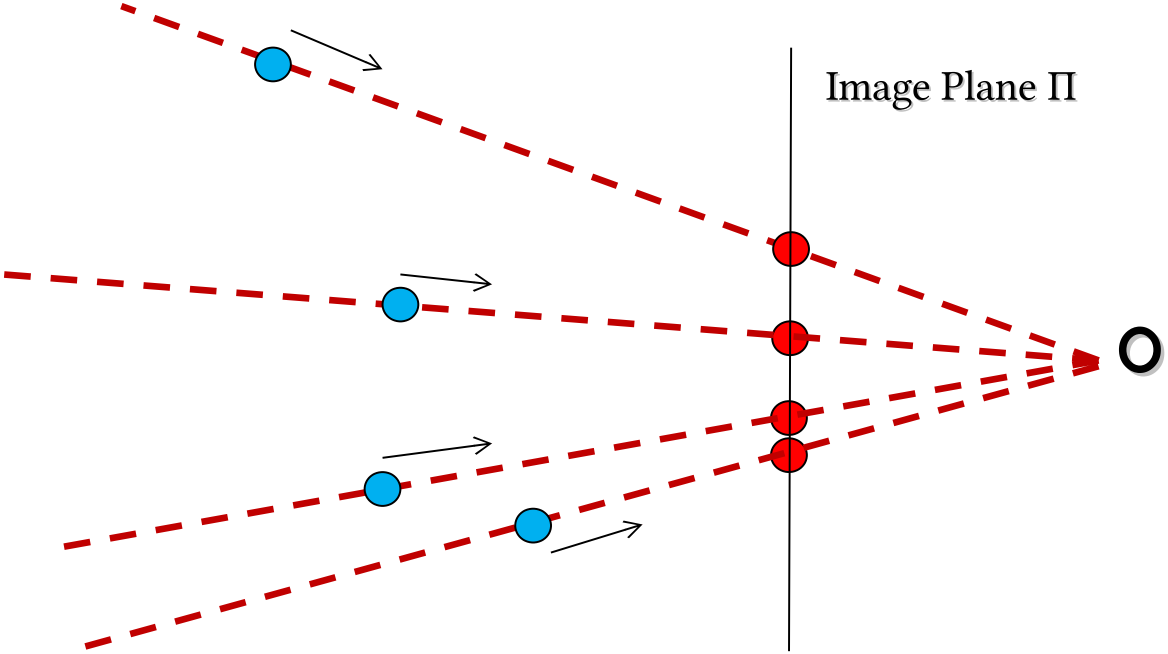


$$M = K\begin{bmatrix} R & T \end{bmatrix}$$

- Can we say where is P when we know only p?
  - NO: any P on the line of sight can generate p…
  - M can not be inverted (3×4)

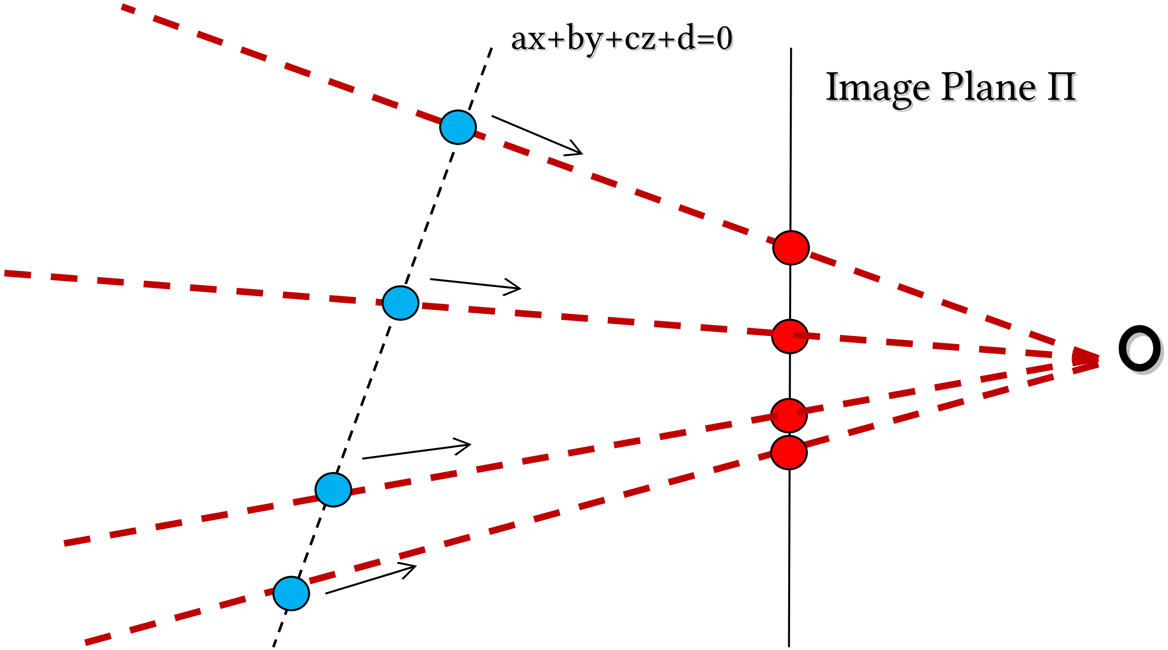
Ill posed problem

UNIVERSITÀ DI PARMA



- Projective space ⟶ we lost one dimension
- 3D [x,y,z] are projected in [x/z,y/z,1]

Add a constraint



- Just image that the (red) image points are projection of points that lie on the same plane
- In such a case for each image point there is only one world point (cyan)

Add a constraint

- Given a plane aX + bY + cZ + d = 0

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \overset{\boxed{3\times 4}}{M} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} u \\ v \\ w \\ 0 \end{bmatrix} = \overset{\boxed{4\times 4}}{\begin{bmatrix} M \\ a\,b\,c\,d \end{bmatrix}} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} = \begin{bmatrix} M \\ a\,b\,c\,d \end{bmatrix}^{-1} \begin{bmatrix} u \\ v \\ 1 \\ 0 \end{bmatrix} \xrightarrow[euclidean]{} \begin{bmatrix} X/W \\ Y/W \\ Z/W \end{bmatrix}$$

- We can obtain euclidean world coordinates of pixel (u,v)

Specific case, plane Z=0

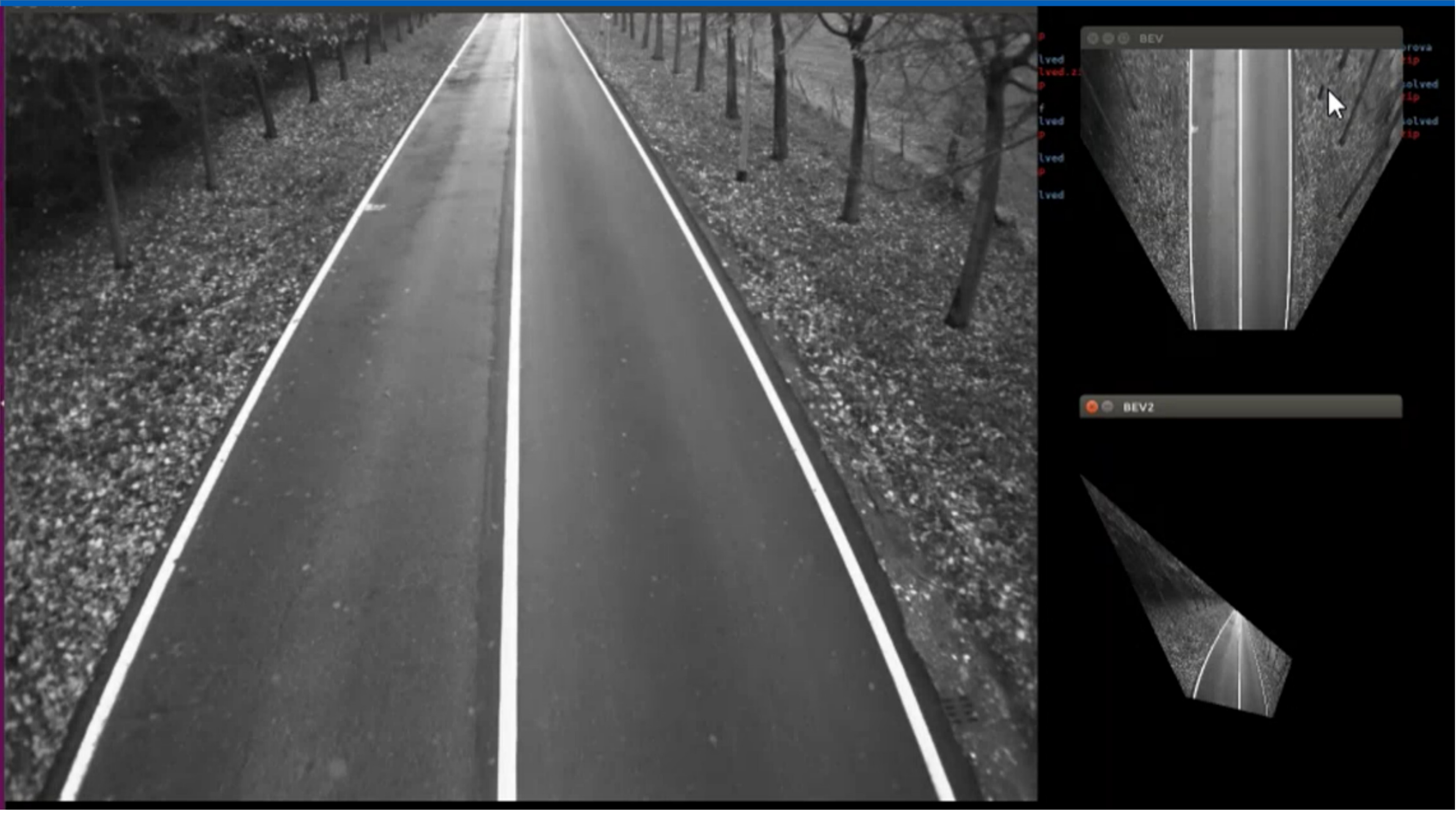
UNIVERSITÀ DI PARMA

- Consider a very specific plane ⟶ Z=0

$$\begin{bmatrix} X \\ Y \\ 0 \\ W \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} X \\ Y \\ W \end{bmatrix} = H \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

- H is 3×3
- This actually works for every possible plane!


BEV
BEV2

Single View Reconstruction

http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/hut/hutme.wrl

Single View Reconstruction

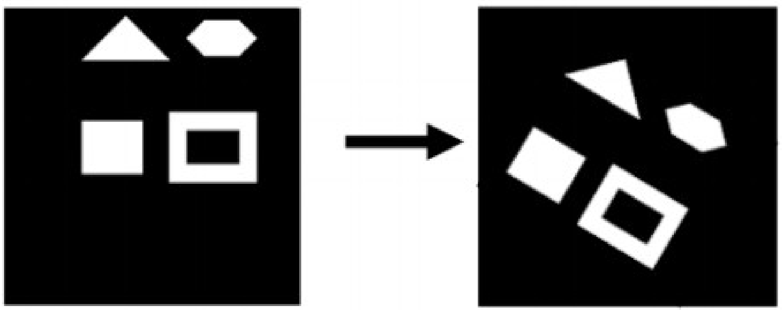
http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/hut/hutme.wrl

Transformations in 2D

- Let us introduce some transformations
- For sake of simplicity in 2D
  - Isometric transformations
  - Similarity transformations
  - Affine transformations
  - Projective transformations or **homographies**

Isometric Transformation

- Concatenation of rotation and translation
- 3 degrees of freedom
  - 2 for translation and 1 for rotation
- **It preserves distance**
- Motion of rigid object



$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_i \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

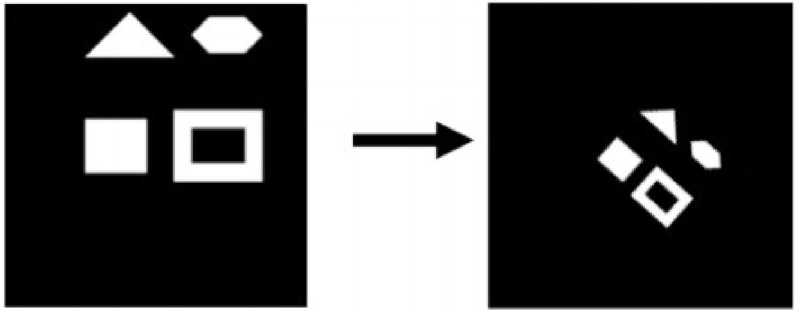
$$H_i = \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

R is a 2×2 matrix

Similarity Transformation

UNIVERSITÀ DI PARMA

- Concatenation of rotation, translation, and scale
- 4 degrees of freedom
  - 2 for translation, 1 for rotation, 1 for scale
- **It preserves shape**
- Angles among lines & Lenghts ratio



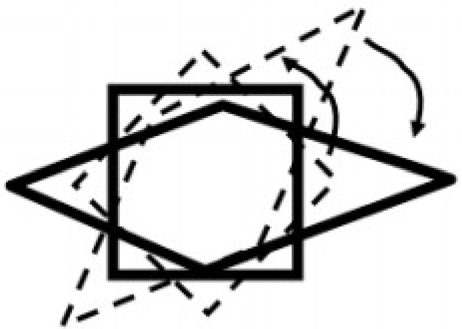
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} SR & t \\ 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad S = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$$

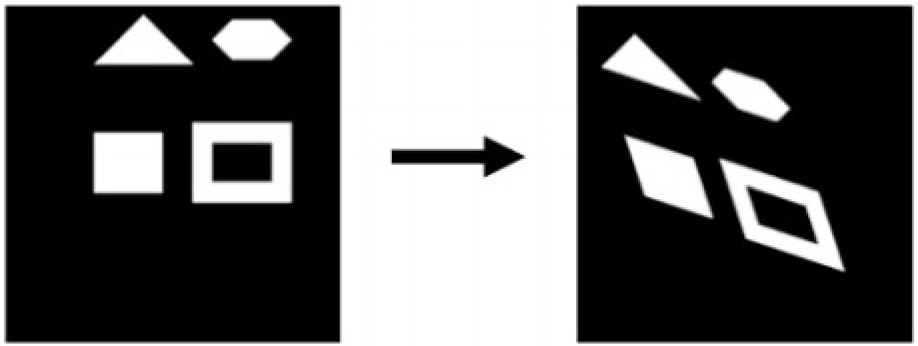
SR is still a 2×2 matrix

Affine Transformation

UNIVERSITÀ DI PARMA

- **It preserves ??**





$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_a \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = R(\theta) \cdot R(-\phi) \cdot D \cdot R(\phi) \qquad D = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

Affine Transformation

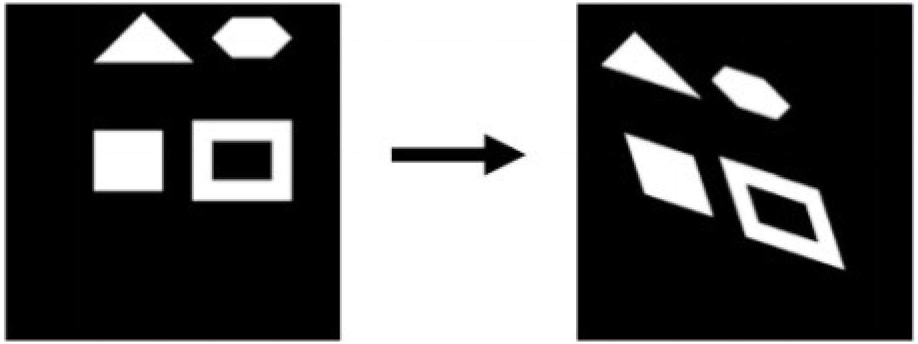
- Matrix A can be seen as a Single Value Decomposition of 2 ortogonal matrices and a diagonal one

$$A = UDV^T = (UV^T)(VDV^T) = R(\theta)\cdot R(-\phi)\cdot D\cdot R(\phi)$$

$$UV^T = R(\theta), \quad V^T = R(\phi), \quad D = D$$

Affine Transformation

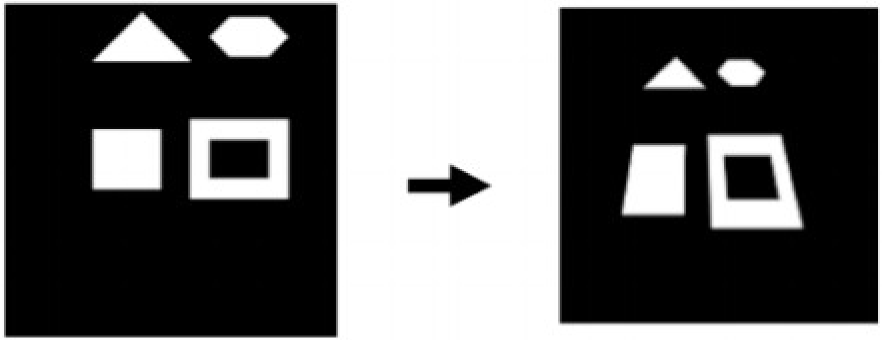
- **It preserves**
  - **Parallel lines**
  - **Areas ratio**
  - **Collinear segment ratio**
- 6 degrees of freedom



$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_a \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = R(\theta) \cdot R(-\phi) \cdot D \cdot R(\phi) \qquad D = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

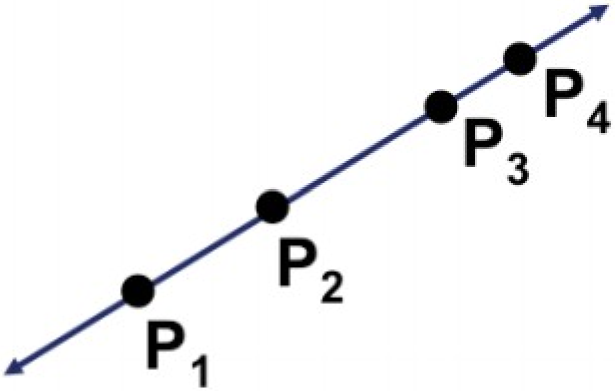
Projective Transformation

- **It preserves**
  - **Collinearity**
  - **Cross-ratio for 4 collinear points**
- 8 degrees of freedom



$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} A & t \\ v & c \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_p \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Cross-ratio for 4 collinear points



[Eq. 9]

$$\frac{\|\mathbf{P}_3 - \mathbf{P}_1\| \, \|\mathbf{P}_4 - \mathbf{P}_2\|}{\|\mathbf{P}_3 - \mathbf{P}_2\| \, \|\mathbf{P}_4 - \mathbf{P}_1\|}$$

$$\mathbf{P}_i = \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

Projective Transformation

UNIVERSITÀ DI PARMA

- 8 degrees of freedom
  - Consider a scale factor
- The previous equation can be written as

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = k \begin{bmatrix} A & t \\ v & c \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = k' \begin{bmatrix} \frac{A}{c} & \frac{t}{c} \\ \frac{v}{c} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Affine transformation vs Homograpy

UNIVERSITÀ DI PARMA

- It can be noticed that “up” to the affine transformation z is left unchanged

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_a \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Omographies also affect z!

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} A & t \\ v & c \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = H_p \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Affine transformation vs Homograpy

- Affine transformations:
  - move the (image) point
  - originally the projection of a (world) point
  - in another point
  - that it is the projection of another (world) point at the same distance as the previous one
  - (x,y,z) $\longrightarrow$ (x',y',z)
- Omographies do not have that constraint!
  - Let's consider the last H_P line as [a b c]
    - (x,y,z) $\longrightarrow$ (x',y',ax+by+cz)

Planar Homograpy

- A plane in the $\mathbb{R}^3$ space is
  - $ax+by+cz+d=0 \longrightarrow ax+by+cz=-d$
- Projection of points that lies on such plane are (x', y', -d)
- Homographies transform points that lie on a plane on another plane: plane ⟶ another plane with a different point of view

Planar Homograpy

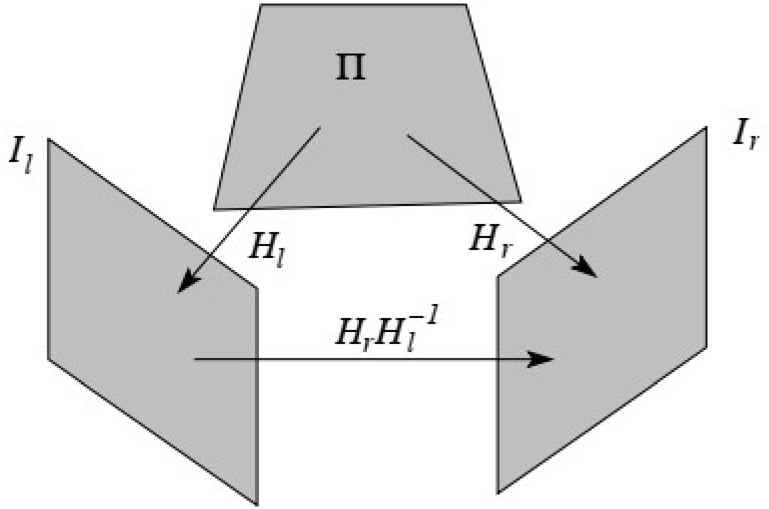
UNIVERSITÀ DI PARMA

- Consider 2 planes
  - $ax+by+cz+d=0$ and $a'x+b'y+c'z+d'=0$
- It can be easily demonstrated that we can compute an homograpy that bring from one plane to the other
- We can compute H and H' homographies
- We can then combine them to obtain another projection
  - $H_{\Pi} = (H)\cdot(H')^{-1}(d/d')$

Planar Homograpy

UNIVERSITÀ DI PARMA

- We can see this in a different fashion
- The projection of a plane on another 2 planes can be computed using a homographic transformation



Planar Homography

- Assuming the 2 planes are the sensor planes of 2 cameras
- The induced homography can be computed using intrinsic camera parameters and their relative position

$$H_{\Pi} = K'\left(R + \frac{tn^T}{d}\right)K^{-1}$$

- K and K' are intrinsic camera matrices
- R and t are relative rototranslation matrices
- n·p=d is plane equation

UNIVERSITÀ DI PARMA

Points and Lines at Infinity

2D Lines

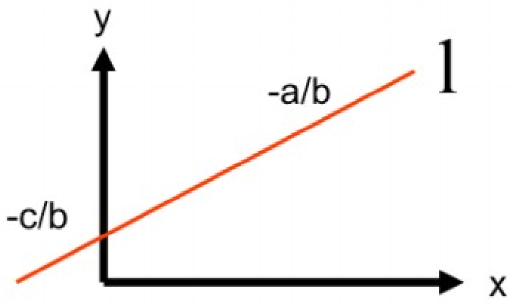
- Let us consider a straight line
  - How we can write it
  - When a point lies on that line

2D Lines

UNIVERSITÀ DI PARMA

$$ax + by + c = 0$$

$$\mathbf{l} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$



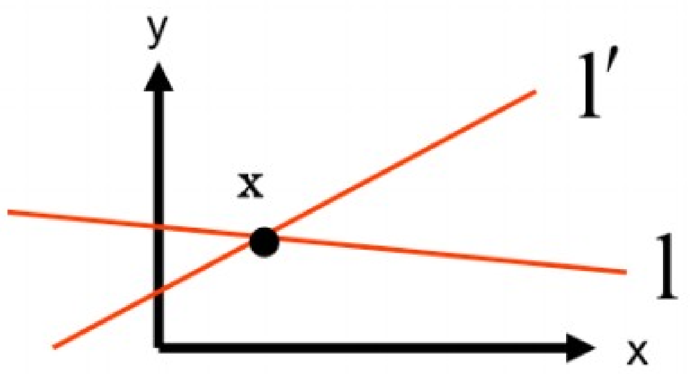
$$\text{if } x = [x, y] \in l \quad \text{or} \quad \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \in l \quad \rightarrow \quad [x\ y\ 1] \begin{bmatrix} a \\ b \\ c \end{bmatrix} = x^T l = 0$$

2D Lines Intersection

- Not paralle lines l and l' intersect at <u>unique</u> point x
- Where is x?
  - x lies on both lines, therefore: $x^T l = 0$ and $x^T l' = 0$ (1)
  - Let x be $l \times l'$
  - Does this satisfies (1)?
    - Hint: how is the result of a cross product?
    - Solution: it is a vector perpendicular to both l and l'
    - And what is the result of a scalar product between two perpendicular vectors?

2D Lines Intersection

$$x = l \times l'$$ Eq. 11

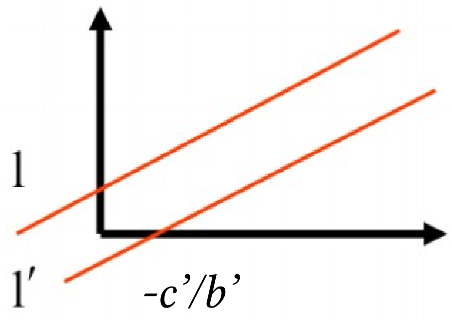


$$\mathbf{l} \times \mathbf{l}' \perp \mathbf{l} \;\rightarrow (\mathbf{l} \times \mathbf{l}') \cdot \mathbf{l} = 0 \;\rightarrow x \in l \quad \text{[Eq. 12]}$$

$$\mathbf{l} \times \mathbf{l}' \perp \mathbf{l}' \;\rightarrow \underbrace{(\mathbf{l} \times \mathbf{l}')}_{\mathbf{x}} \cdot \mathbf{l}' = 0 \;\rightarrow x \in l' \quad \text{[Eq. 13]}$$

2D Parallel Lines Intersection

$$x=\begin{bmatrix}x\\y\\w\end{bmatrix}, w\neq 0$$



$$l=\begin{bmatrix}a\\b\\c\end{bmatrix} \qquad l'=\begin{bmatrix}a'\\b'\\c'\end{bmatrix}=\begin{bmatrix}a\\b\\c''\end{bmatrix}$$

$$x_\infty=\begin{bmatrix}x\\y\\0\end{bmatrix}$$

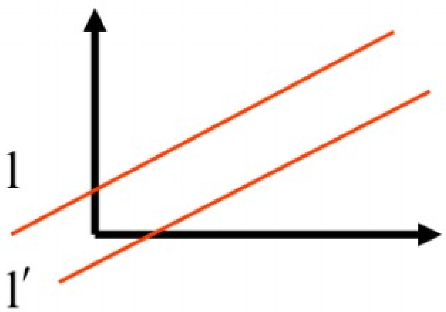
$$l\times l'=\begin{bmatrix}bc''-bc\\ac-ac''\\ab-ab\end{bmatrix}=\begin{bmatrix}b(c''-c)\\-a(c''-c)\\0(c''-c)\end{bmatrix}\propto\begin{bmatrix}b\\-a\\0\end{bmatrix}=x_\infty \quad eq.13$$

- The result is a point to infinity
  - **Ideal point**

2D Ideal Points

UNIVERSITÀ DI PARMA

$$x_\infty = \begin{bmatrix} x \\ y \\ 0 \end{bmatrix} = \begin{bmatrix} b \\ -a \\ 0 \end{bmatrix}$$



$$l = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \qquad l' = \begin{bmatrix} a' \\ b' \\ c' \end{bmatrix}$$

Note: the line l = $[a\ b\ c]^T$ pass trough the ideal point x_∞

$$x_\infty^T l = \begin{bmatrix} b & -a & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = 0$$

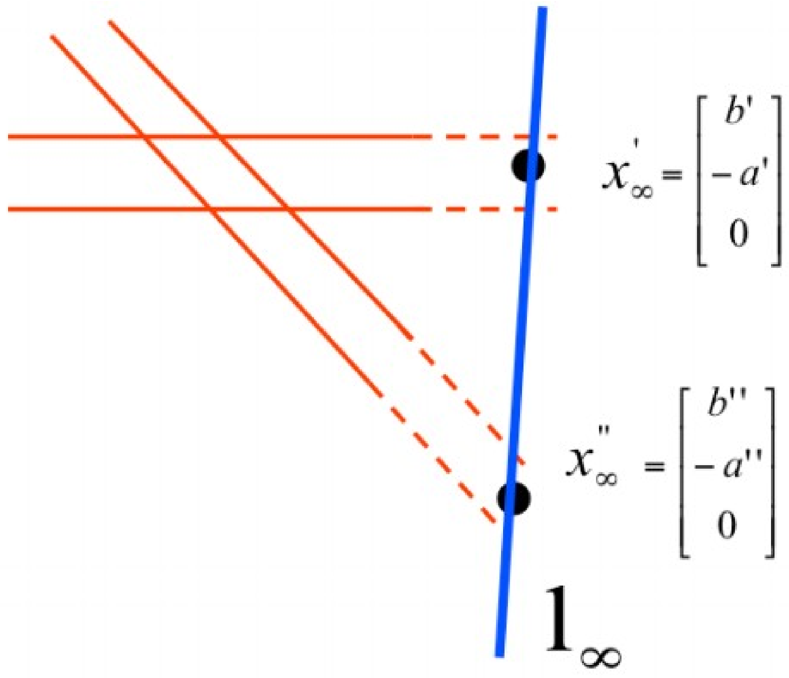
- This is true for every straight lines with the same slope

Ideal line

- All ideal points lie on the same line → ideal line or vanishing line
  - set of "directions" of lines in the plane

$$l_\infty = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$[x\ y\ 0] \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = 0$$



Ideal points transformations

UNIVERSITÀ DI PARMA

$$H = \begin{bmatrix} A & t \\ v & b \end{bmatrix}$$

$$p' = H\ p$$

is it a point at infinity?

$$H\ p_\infty = ? \ = \begin{bmatrix} A & t \\ v & b \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

[Eq. 17]

Ideal line transformation

$$H = \begin{bmatrix} A & t \\ v & b \end{bmatrix}$$

$$l' = H^{-T}\, l$$

[Eq. 19]

is it a line at infinity?

$$H^{-T}\, l_\infty = ? = \begin{bmatrix} A & t \\ v & b \end{bmatrix}^{-T} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ b \end{bmatrix}$$

…no!

[Eq. 20]

$$H_A^{-T}\, l_\infty = ? \quad = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix}^{-T} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} A^{-T} & 0 \\ -t^T A^{-T} & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

[Eq. 21]

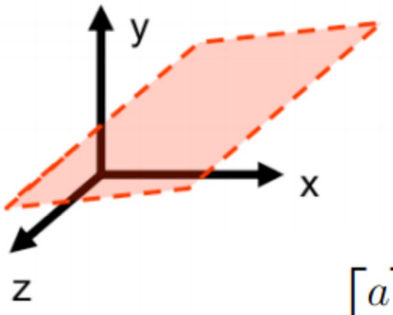
UNIVERSITÀ DI PARMA

Back to 3D

3D planes e lines

[HZ] Ch 3.2.2

$$\mathrm{x} = \begin{bmatrix} \mathrm{x}_1 \\ \mathrm{x}_2 \\ \mathrm{x}_3 \\ 1 \end{bmatrix} \qquad \Pi = \begin{bmatrix} \mathrm{a} \\ \mathrm{b} \\ \mathrm{c} \\ \mathrm{d} \end{bmatrix}$$



$$x^T \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = ax_1 + bx_2 + cx_3 + d = 0$$

$$x \in \Pi \leftrightarrow x^T \Pi = 0$$

[Eq. 22]

$$\mathrm{ax} + \mathrm{by} + \mathrm{cz} + \mathrm{d} = 0$$

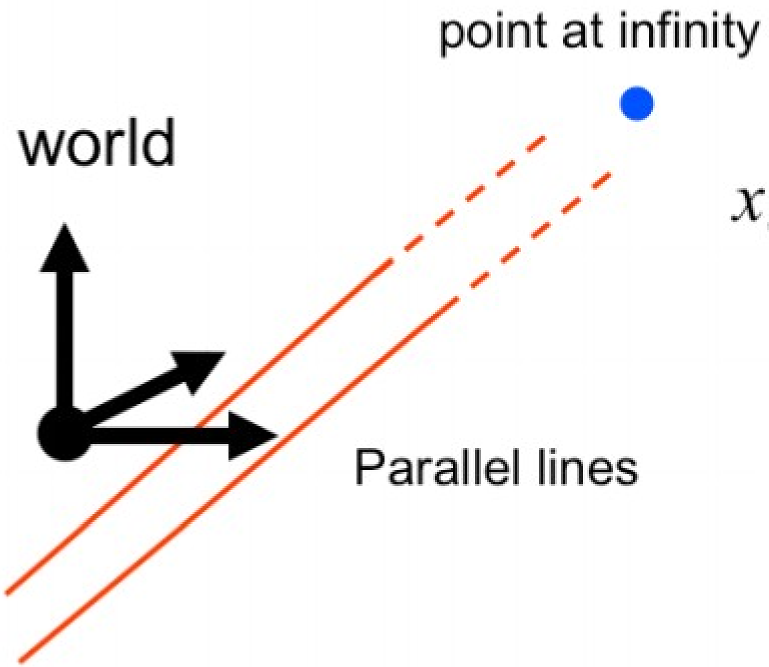
[Eq. 23]

$$l_{3D} = \begin{bmatrix} \Pi_1^T \\ \Pi_2^T \end{bmatrix}_{2\times 4}$$

- Lines have 4 degree of freedom
  - They can be seen as intersections of 2 planes

3D vanishing points

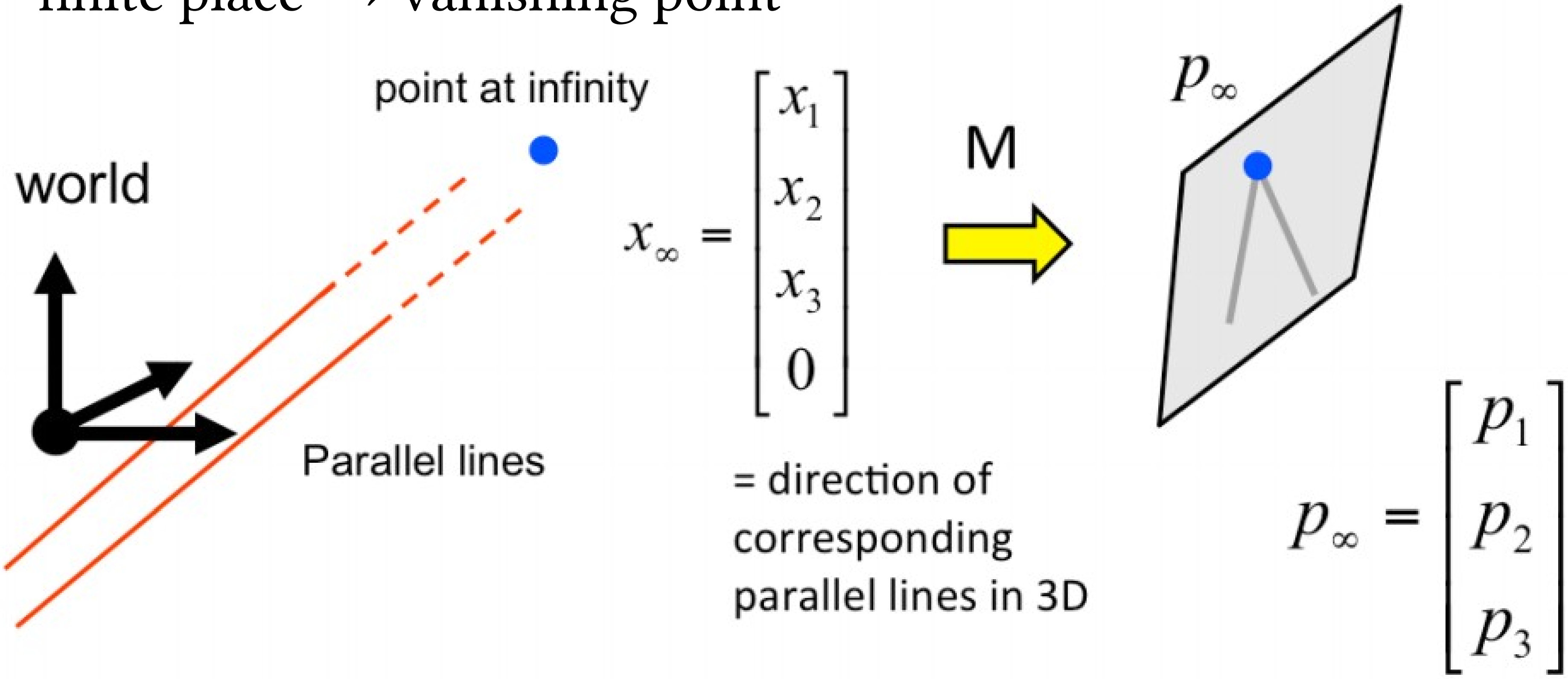
- Similar to 2D case



$$x_\infty = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \end{bmatrix}$$

3D vanishing points

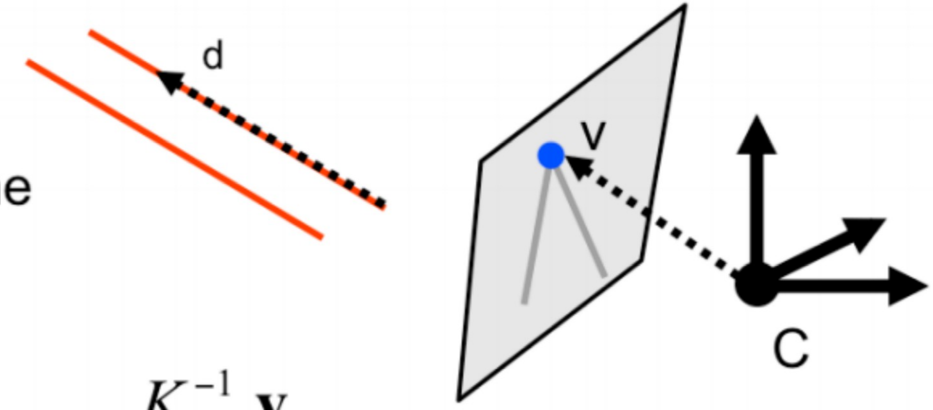
- A projective transformation M can bring point to infinity in a finite place → vanishing point



Vanishing points vs Camera parameters

UNIVERSITÀ DI PARMA

Let's consider a pencil of lines



d = direction of the line
= [a, b, c]^T

$$\mathbf{v} = K\,\mathbf{d}$$

[Eq. 24]

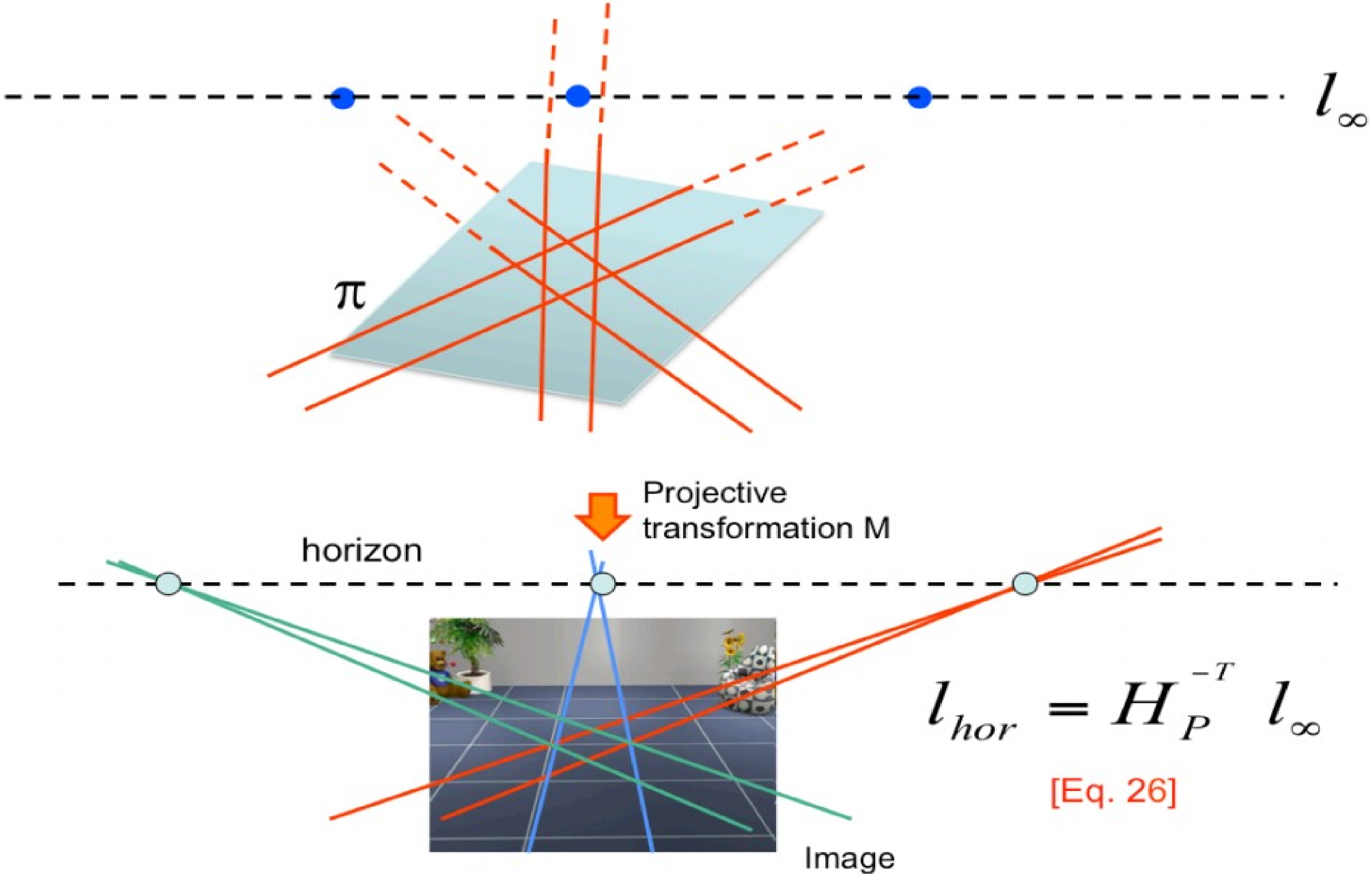
$$\mathbf{d} = \frac{K^{-1}\,\mathbf{v}}{\left\| K^{-1}\,\mathbf{v} \right\|}$$

[Eq. 25]

Proof:

$$X_\infty = \begin{bmatrix} a \\ b \\ c \\ 0 \end{bmatrix} \xrightarrow{\mathrm{M}} \mathbf{v} = \mathbf{M}\,\mathbf{X}_\infty = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix} a \\ b \\ c \\ 0 \end{bmatrix} = \mathbf{K}\begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

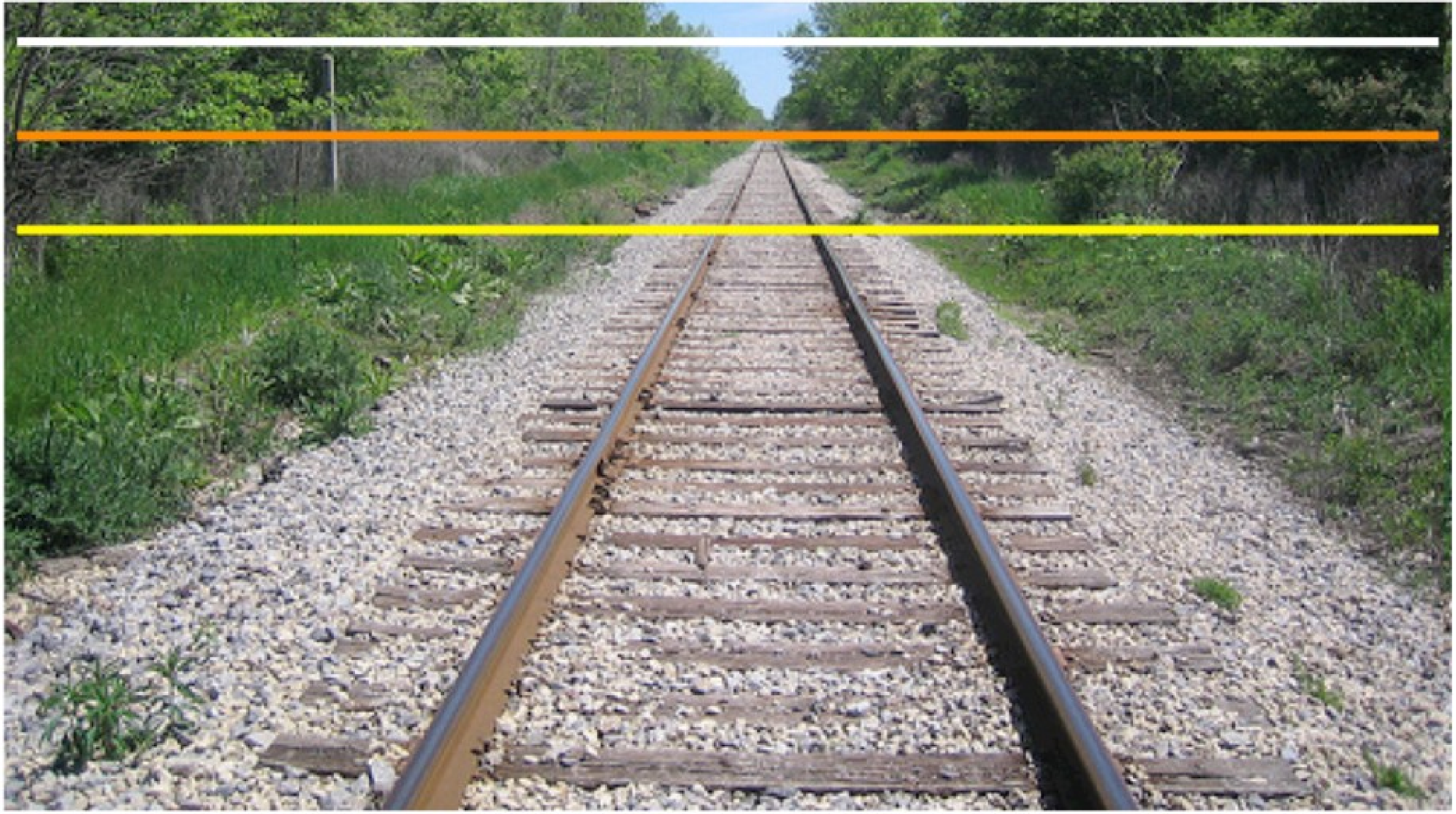
Horizon line



$$l_{hor} = H_P^{-T}\ l_\infty$$

[Eq. 26]

Horizon line



Humans intuitively deduce properties about the image

Horizon line

If lines intersect on horizon line they are parallel

Horizon line

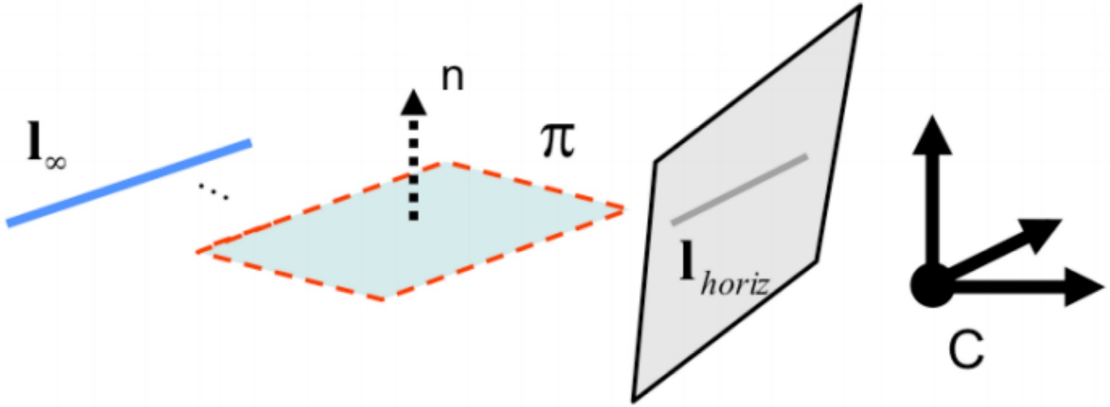
If lines intersect on horizon line they are parallel

Horizon line

If lines intersect on horizon line they are parallel

Horizon line vs Camera parameters

[HZ] Ch 8.6.2



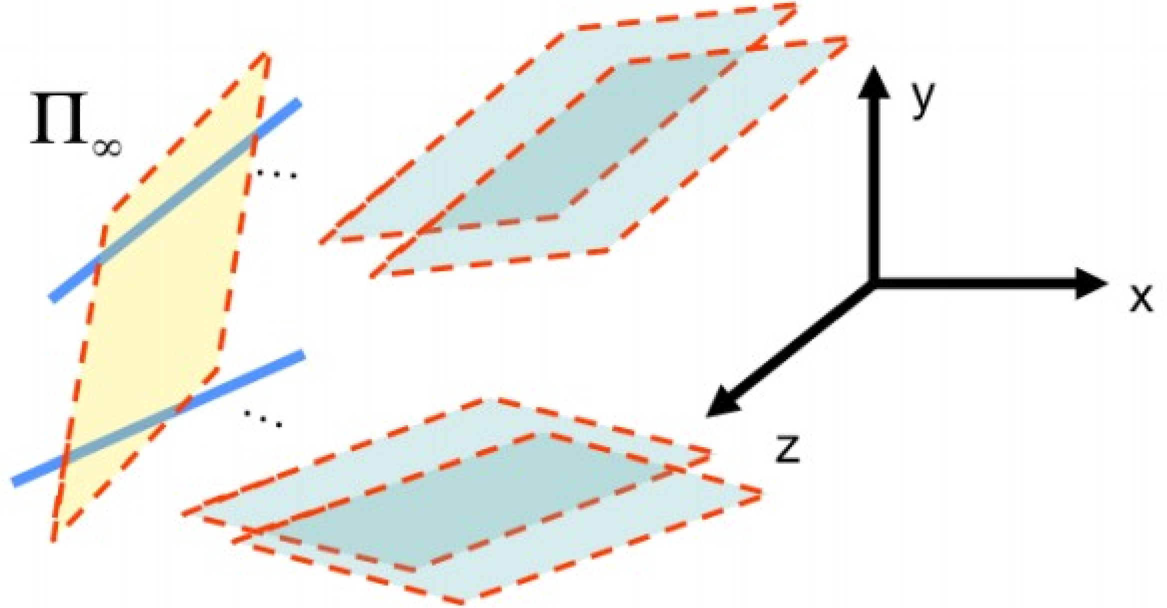
we can estimate the orientation of the ground plane!

$$\mathbf{n} = K^T \mathbf{l}_{\text{horiz}}$$

[Eq. 27]

Planes at infinity

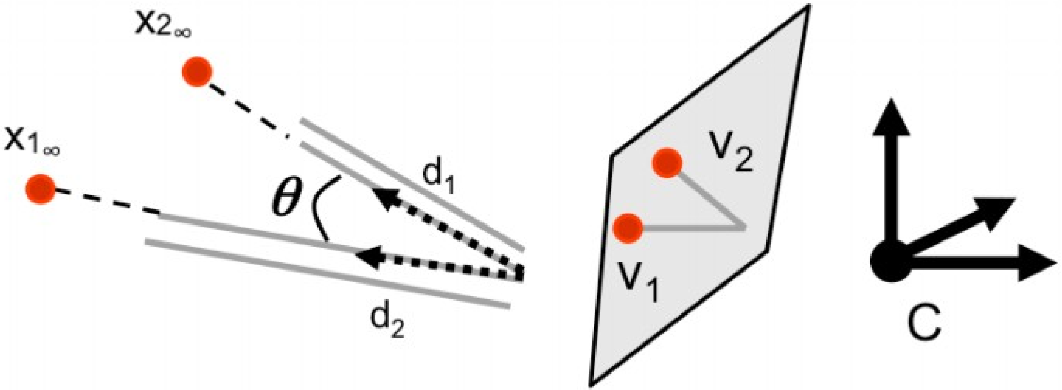
- Parallel planes intersect on a line at infinity
- 2 or more lines at infinity define a plane at infinity



$$\Pi_\infty = \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

plane at infinity

Angle between 2 pencils of straight lines



$$\cos\theta = \frac{v_1^T \omega\, v_2}{\sqrt{v_1^T \omega\, v_1}\sqrt{v_2^T \omega\, v_2}}$$

[Eq. 28]

$$\omega = (K\,K^T)^{-1}$$

If $\theta = 90 \quad \rightarrow \quad \boxed{v_1^T \omega\, v_2 = 0}$ [Eq. 29]

Scalar equation

Small recap

$$\boxed{\mathbf{v} = K\,\mathbf{d}}$$

[Eq. 24]

$$\boxed{\mathbf{n} = \mathrm{K}^{\mathrm{T}}\mathbf{l}_{\mathrm{horiz}}}$$

[Eq. 27]

$$\cos\theta = \frac{\mathrm{v}_1^{\mathrm{T}}\boldsymbol{\omega}\,\mathrm{v}_2}{\sqrt{\mathrm{v}_1^{\mathrm{T}}\boldsymbol{\omega}\,\mathrm{v}_1}\sqrt{\mathrm{v}_2^{\mathrm{T}}\boldsymbol{\omega}\,\mathrm{v}_2}}$$

[Eq. 28]

$$\xrightarrow{\boldsymbol{\theta} = 90} \boxed{\mathrm{v}_1^{\mathrm{T}}\boldsymbol{\omega}\,\mathrm{v}_2 = 0}$$

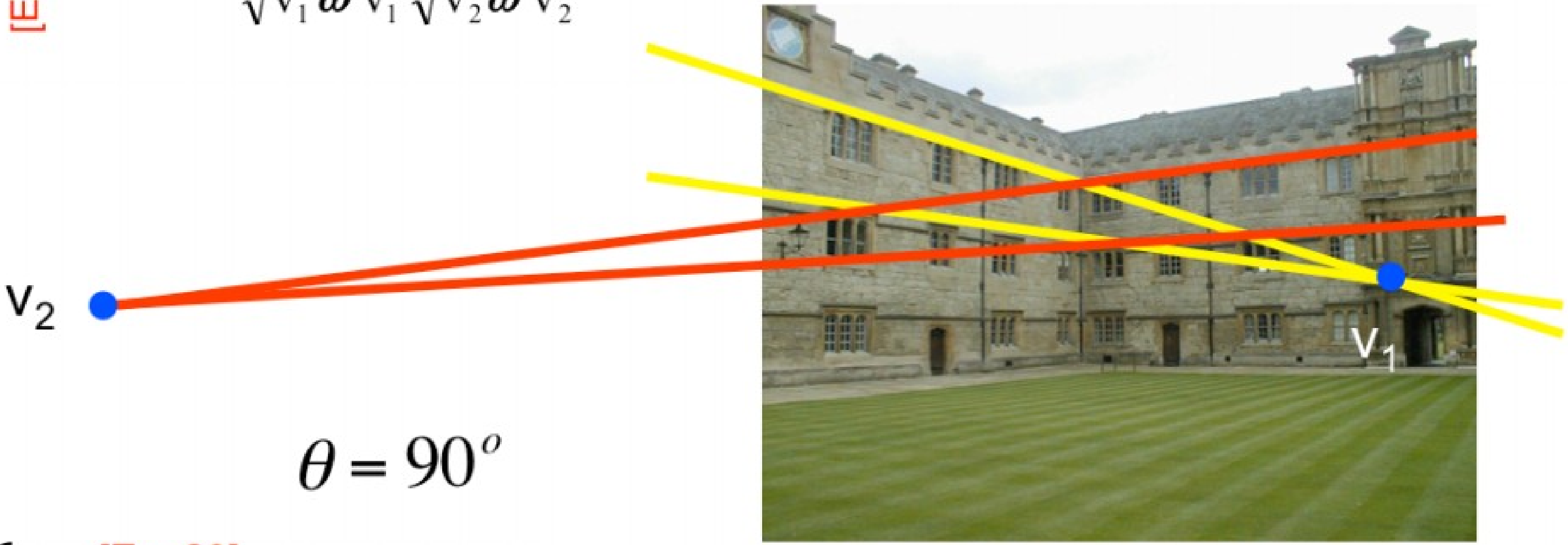
[Eq. 29]

$$\boxed{\omega = (K\,K^{T})^{-1}}$$

Estimating geometry from a single image

[Eq. 28]

$$\cos\theta = \frac{v_1^T \omega\, v_2}{\sqrt{v_1^T \omega\, v_1}\,\sqrt{v_2^T \omega\, v_2}}$$



$$\theta = 90^o$$

[Eq. 29]

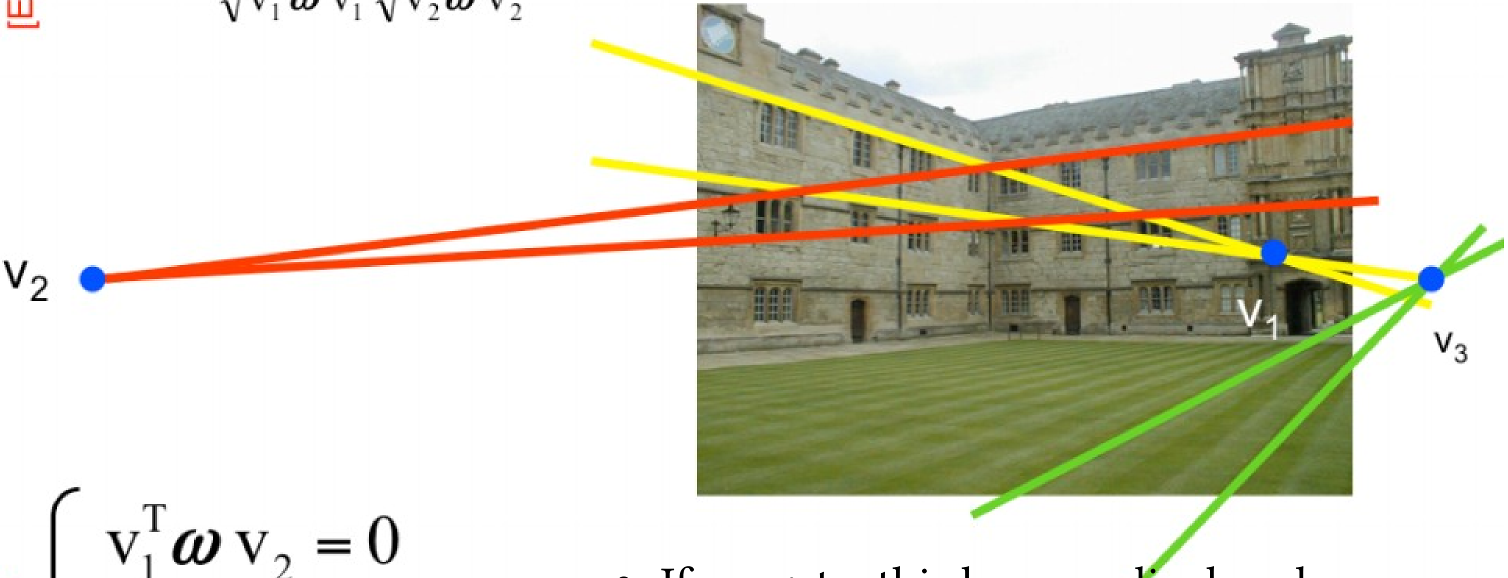
$$\begin{cases} v_1^T \omega\, v_2 = 0 \\ \omega = (K\, K^T)^{-1} \end{cases} \longrightarrow$$

- Identify two 3D planes and a pair of parallel lines for each plane → v_1, v_2 and eq. 28
- 3D perpendicular → eq. 29
- Scalar eq. 29 is still not enough (K has 5 DOF)

Estimating geometry from a single image

[Eq. 28]

$$\cos\theta = \frac{v_1^T \omega\, v_2}{\sqrt{v_1^T \omega\, v_1}\sqrt{v_2^T \omega\, v_2}}$$



[Eqs. 31]

$$\begin{cases} v_1^T \omega\, v_2 = 0 \\ v_1^T \omega\, v_3 = 0 \\ v_2^T \omega\, v_3 = 0 \end{cases}$$

- If we get a third perpendicular plane...
- Scalar eq. 29 is still not enough (3 vs 5 DOF)

Small recap about K

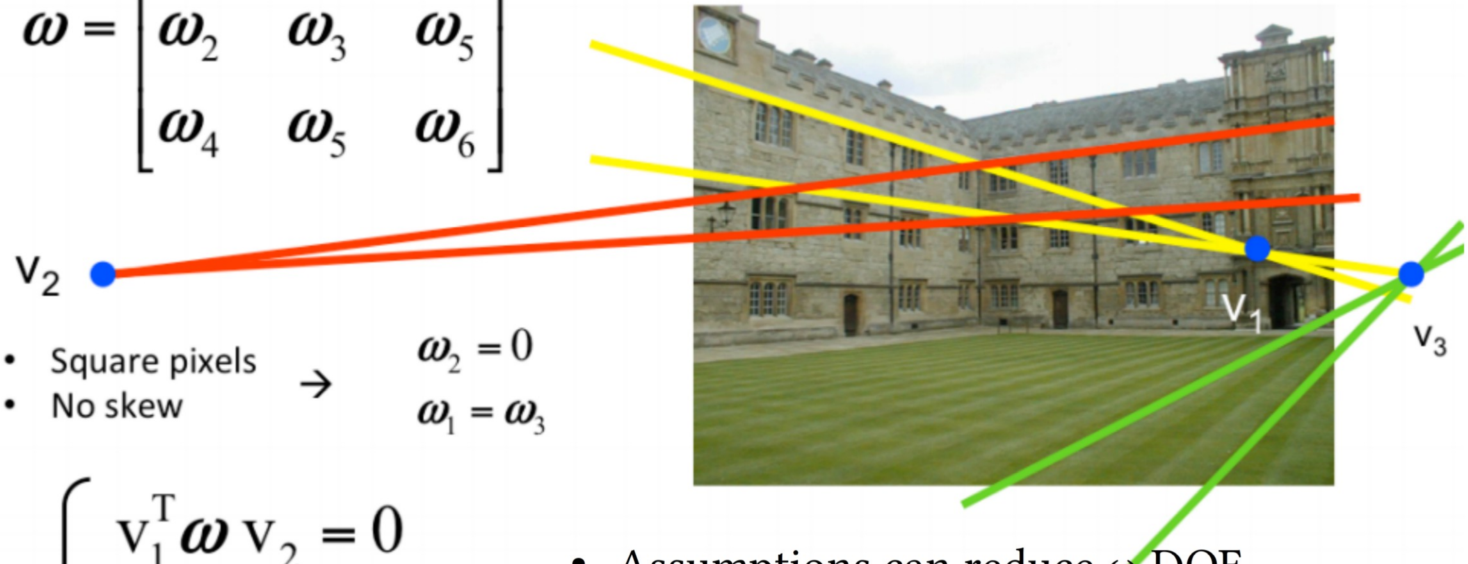
UNIVERSITÀ DI PARMA

$$K = \begin{bmatrix} \alpha & -\alpha \cot(\theta) & c_x \\ 0 & \dfrac{\beta}{\sin(\theta)} & c_y \\ 0 & 0 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Square pixels
No Skew

Estimating geometry from a single image

$$\omega = \begin{bmatrix} \omega_1 & \omega_2 & \omega_4 \\ \omega_2 & \omega_3 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix}$$



- Square pixels
- No skew

$\rightarrow$ $\omega_2 = 0$, $\omega_1 = \omega_3$

[Eqs. 31]

$$\begin{cases} v_1^T \omega \, v_2 = 0 \\ v_1^T \omega \, v_3 = 0 \\ v_2^T \omega \, v_3 = 0 \end{cases}$$

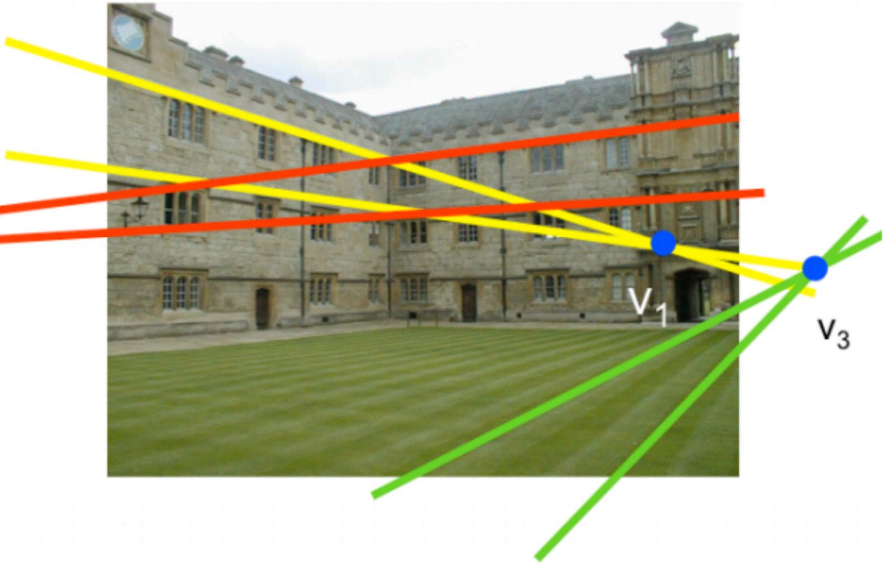
- Assumptions can reduce ω DOF

Estimating geometry from a single image

$$\omega = \begin{bmatrix} \omega_1 & 0 & \omega_4 \\ 0 & \omega_1 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix}$$

known up to scale

v_2



- Square pixels
- No skew

→ $\omega_2 = 0$, $\omega_1 = \omega_3$

[Eqs. 31]

$$\begin{cases} v_1^T \boldsymbol{\omega}\, v_2 = 0 \\ v_1^T \boldsymbol{\omega}\, v_3 = 0 \\ v_2^T \boldsymbol{\omega}\, v_3 = 0 \end{cases}$$

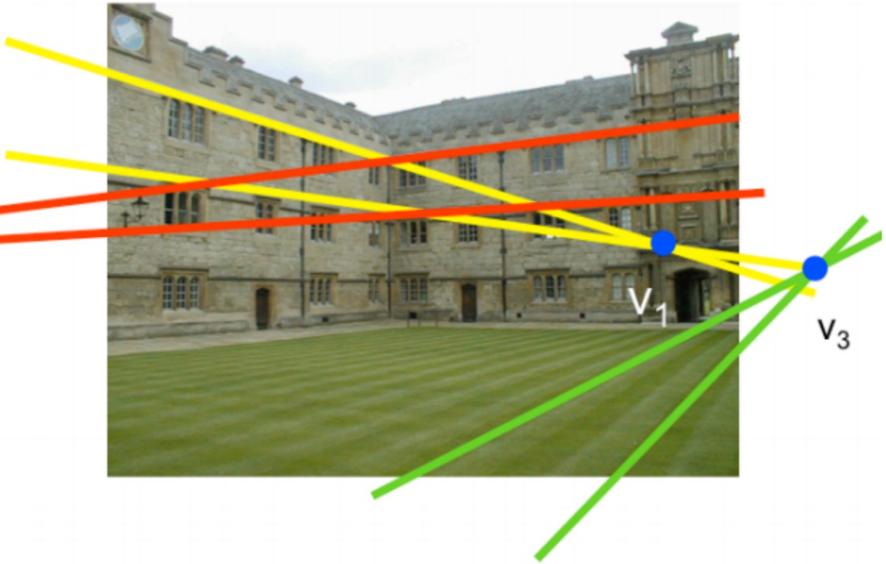
→ Compute ω !

Estimating geometry from a single image

Fact. Cholesky [HZ] Ch pg. 582

$$\omega = \begin{bmatrix} \omega_1 & 0 & \omega_4 \\ 0 & \omega_1 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix}$$ known up to scale

v_2



- Square pixels
- No skew

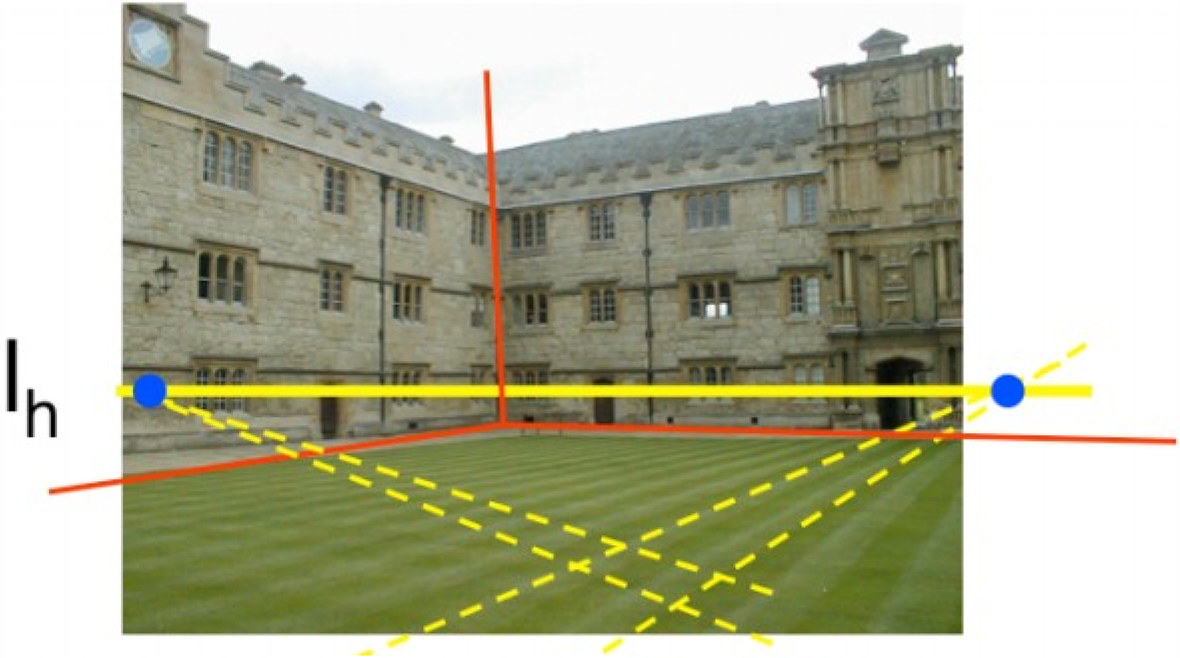
$\rightarrow$ $\omega_2 = 0$, $\omega_1 = \omega_3$

[Eqs. 31]

$$\begin{cases} v_1^T \omega \, v_2 = 0 \\ v_1^T \omega \, v_3 = 0 \\ v_2^T \omega \, v_3 = 0 \end{cases}$$

$$\omega = (K\,K^T)^{-1} \longrightarrow K$$

Estimating geometry from a single image



[Eq. 27]

$$K \text{ known} \rightarrow \mathbf{n} = K^T \mathbf{l}_{horiz}$$

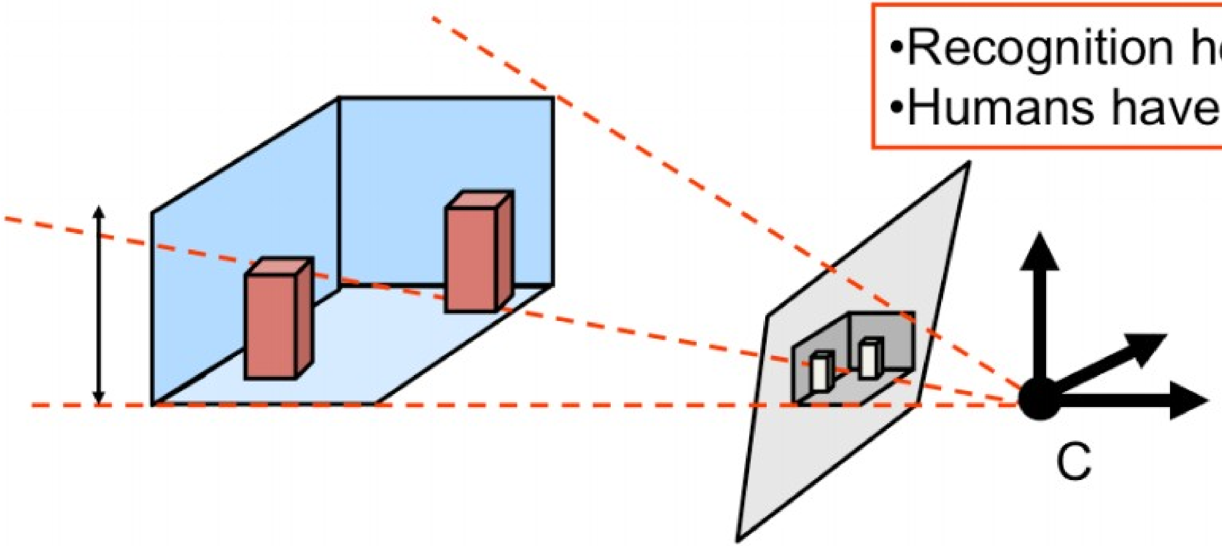
Eq. 27 allows to compute corresponding lines at infinity
We can then select orientation discontinuities (red lines)
And now we can compute the orientation of all planes

Example

http://www.robots.ox.ac.uk/~vgg/projects/SingleView/models/merton/merton.wrl

Example

Small recap



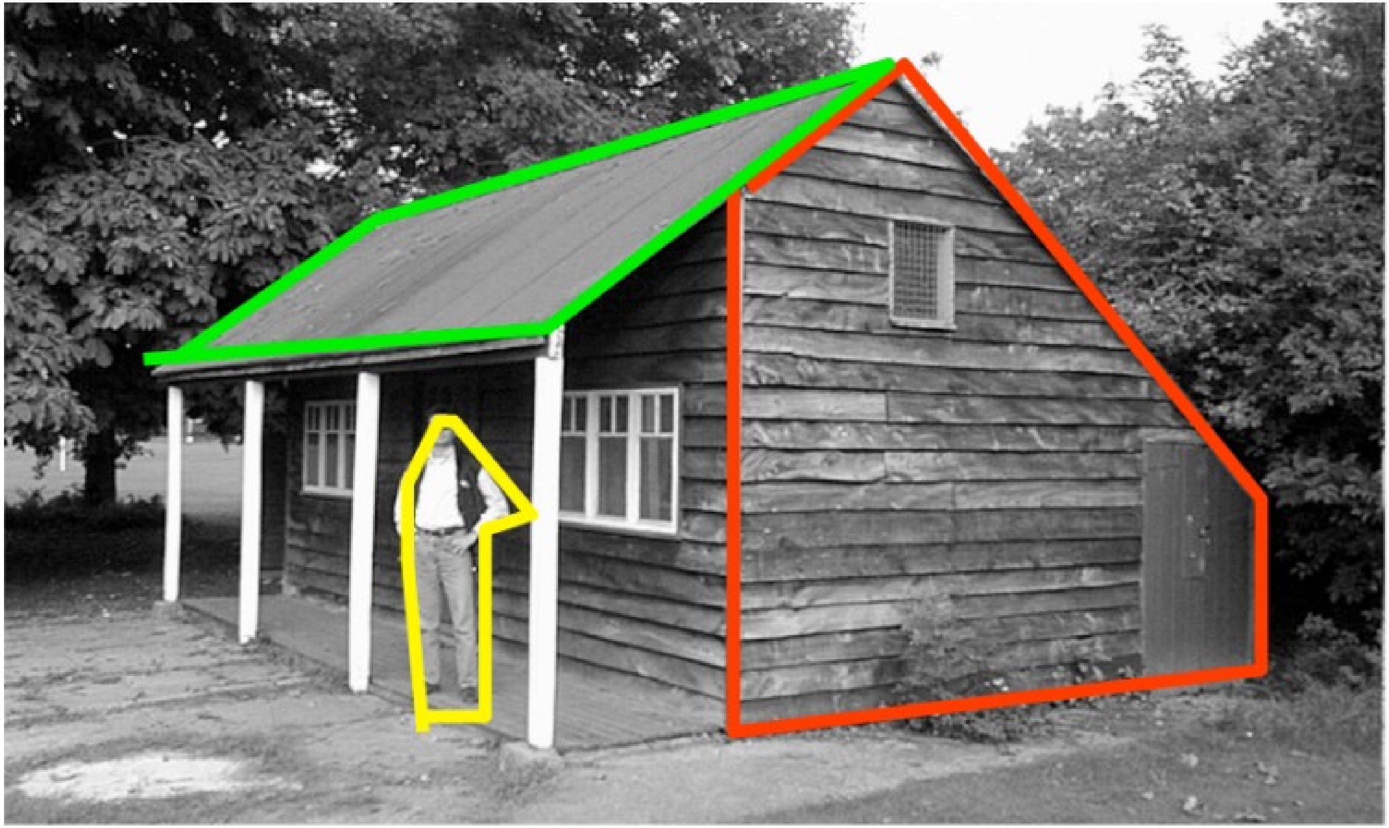
•Recognition helps reconstruction!
•Humans have learnt this

- Now we are able to compute K and some 3D world info using a single camera!
- Scale still unknown
  - We can recover having other info about what we are looking at
- Please this is not a unmanned procedure!

Small recap

UNIVERSITÀ DI PARMA

Not an automatic procedure...



We are not cheating!

UNIVERSITÀ DI PARMA

St. Jerome in his Study

H. Steenwick

We are not cheating!

Not an automatic procedure…

- If humans can do this
  - IA also!

(a) input image (b) superpixels (c) constellations (d) labeling (e) novel view

UNIVERSITÀ DI PARMA

Single View Reconstruction

Question time!