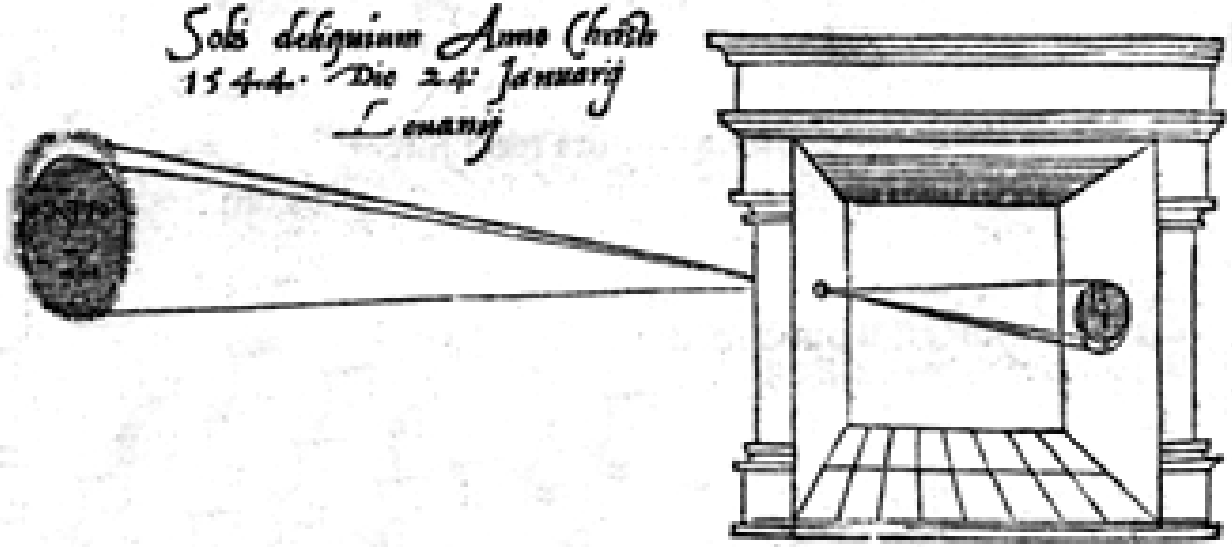
Camera Geometry and single-view Geometry

Camera Geometry

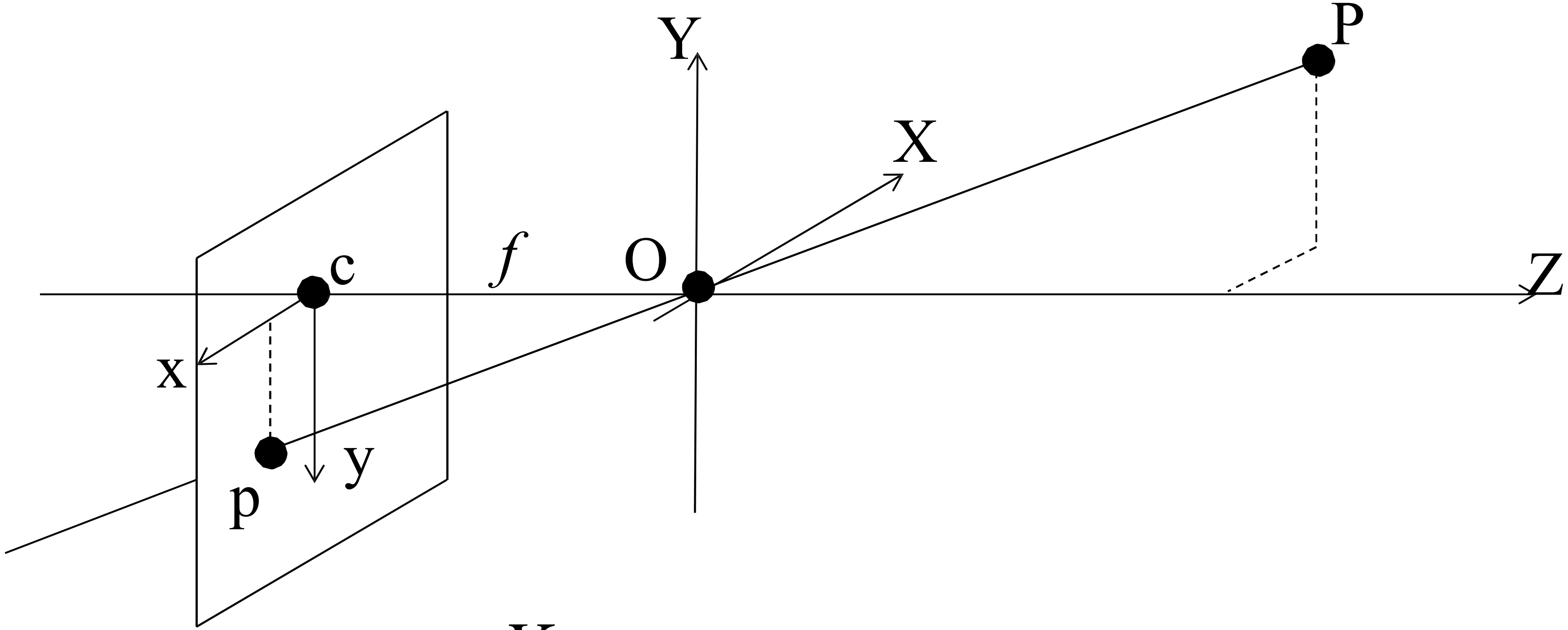
camera projective model

Pinhole camera

illum in tabula per radios Solis, quàm in cœlo contin-
git: hoc eſt, ſi in cœlo ſuperior pars deliquiũ patiatur, in
radiis apparebit inferior deficere, vt ratio exigit optica.



Sic nos exactè Anno .1544. Louanii eclipſim Solis
obſeruauimus, inuenimusq; deficere paulò plus q̃ dex-



$$x = f\,\frac{X}{Z}$$

perspective projection

$$y = f\,\frac{Y}{Z}$$

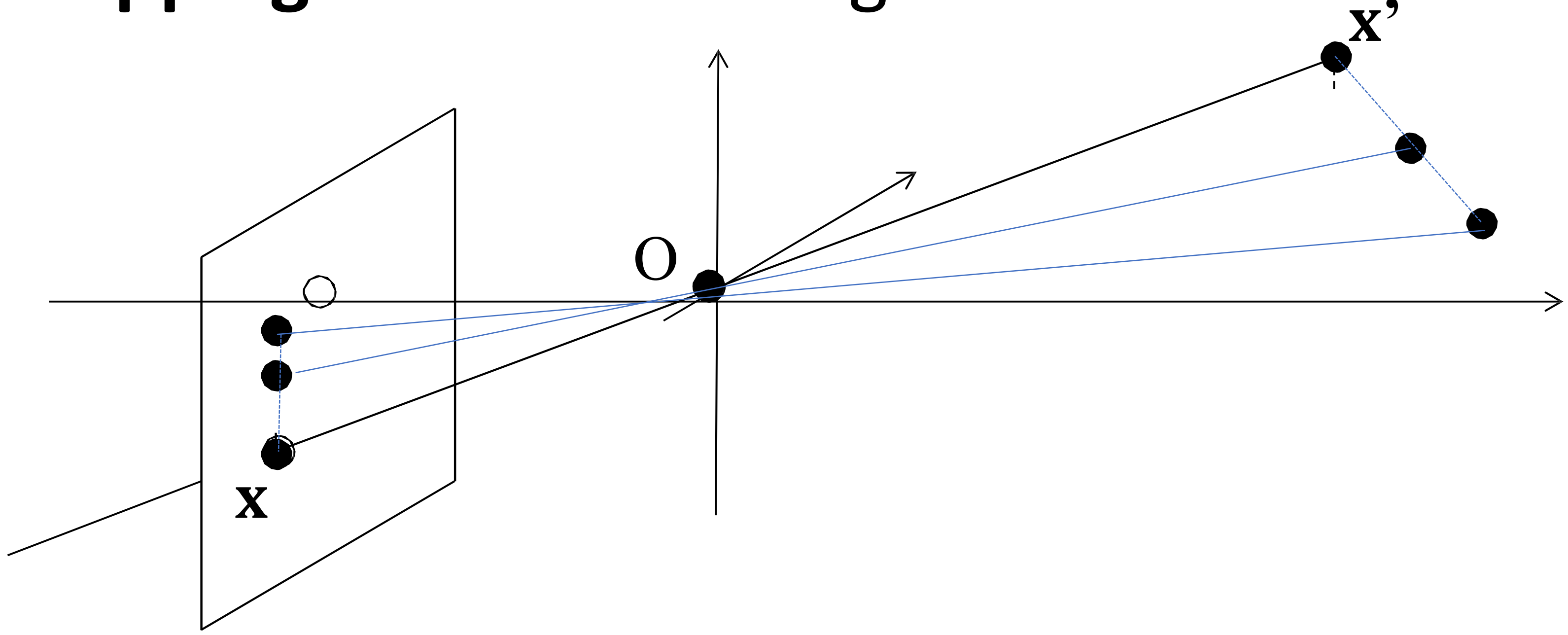
-nonlinear
-not shape-preserving
-not length-ratio preserving

Scene-to-image projection

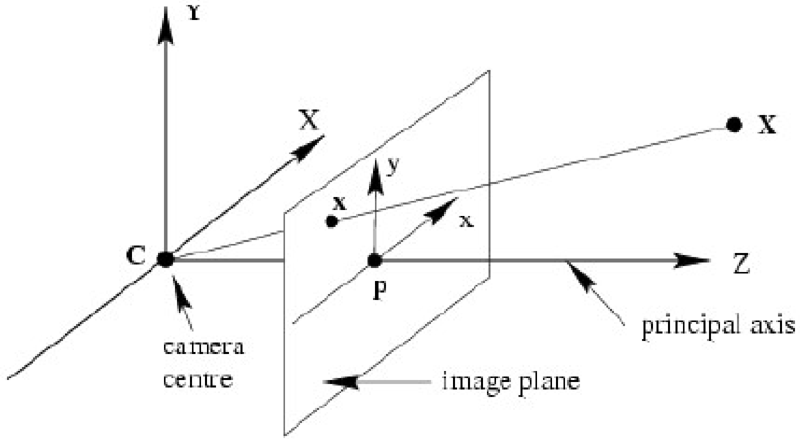
Colinear points are projected onto colinear image points

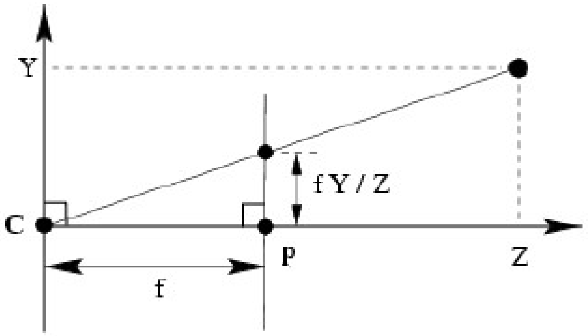
→ colinearity is preserved

→ **linear mapping** between homogeneous coordinates



CAMERA GEOMETRY





colinearity is preserved → linear relation among homogeneous coords

camera projection matrix

$$\mathbf{X} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \mathbf{P}_{3x4} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \mathbf{P}_{3x4}\mathbf{X} = \left| \mathbf{M}_{3x3} \quad \mathbf{m}_{3x1} \right| \mathbf{X}$$

invertible

3D space

image

- **SCENE**
- CAMERA

SCENE: viewing ray from image point

null-space of camera projection matrix **O** $\mathrm{P}\mathbf{O} = 0$

a point **Y** on the line **X**, **O** $\mathbf{Y} = \alpha\mathbf{X} + \beta\mathbf{O}$

its image $\mathbf{u} = \mathbf{PY} = \alpha\mathbf{PX} + \beta\mathrm{P}\mathbf{O} = \mathbf{PX}$

all points Y on (X,O) project on image of X,

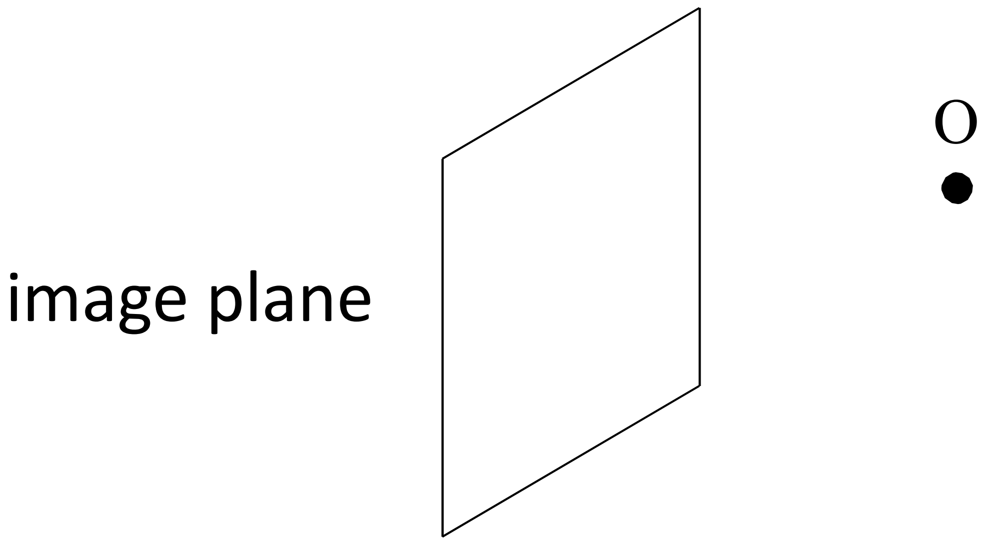
→ $\boxed{\textbf{O is camera center}}$

Image of camera center is $(0,0,0)^T$, i.e. undefined

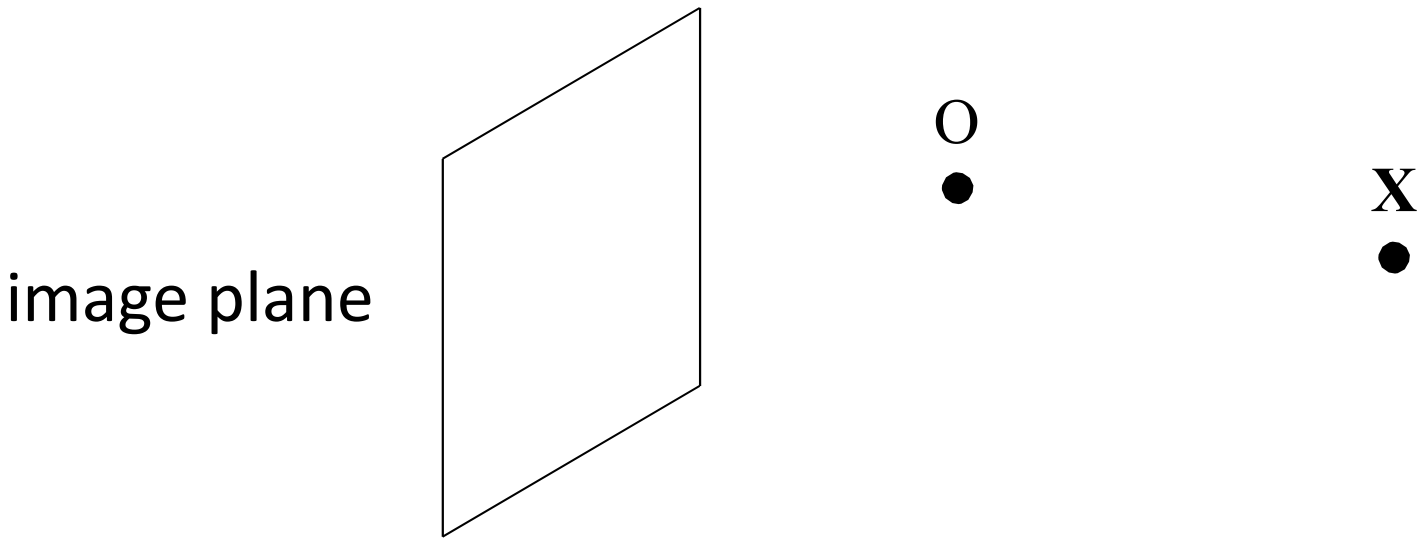
Finite cameras: $\mathbf{O} = \begin{pmatrix} \mathbf{o} \\ 1 \end{pmatrix} = \begin{pmatrix} -\mathrm{M}^{-1}\mathbf{m} \\ 1 \end{pmatrix}$

in fact …

$$O = RNS(P):\ PO = 0$$



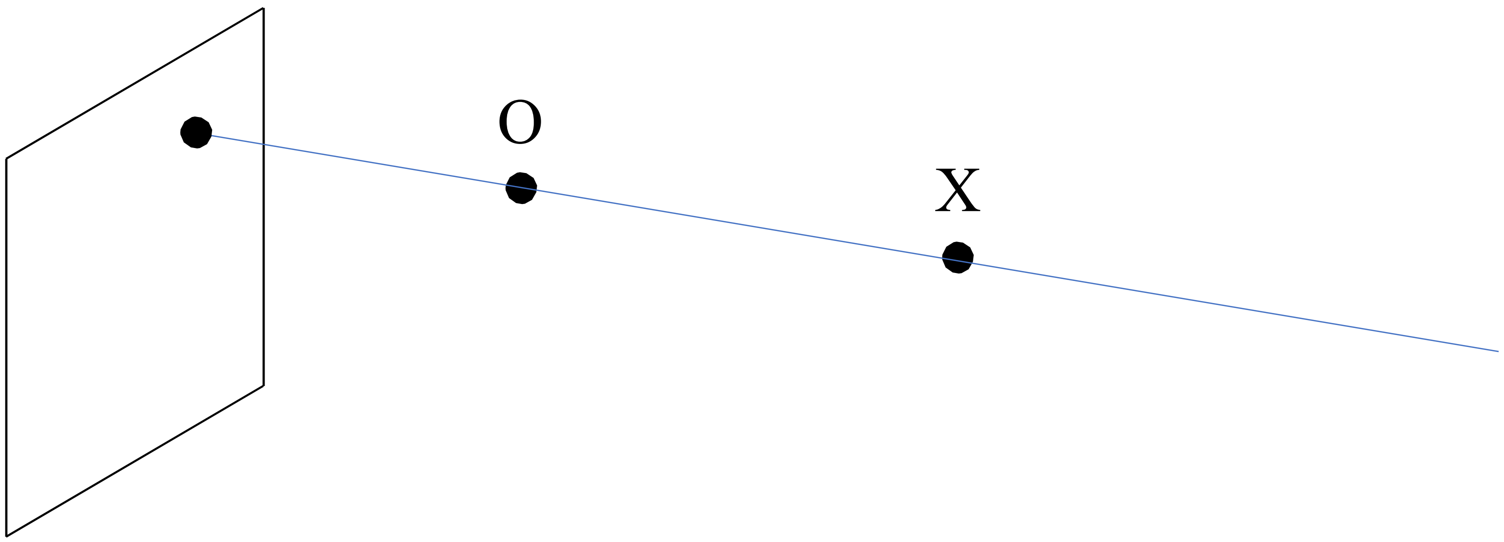
$$\mathrm{O} = RNS(P): \ P\mathrm{O} = 0$$



A point $\mathbf{X}$

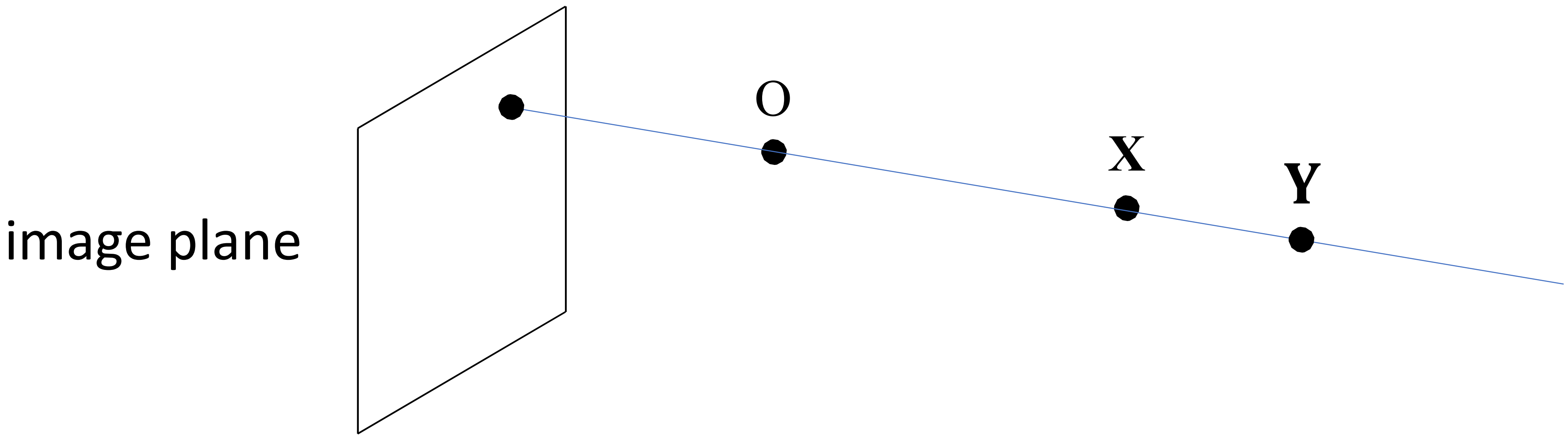
$$\mathrm{O} = RNS(P):\ P\mathrm{O} = 0$$

image plane



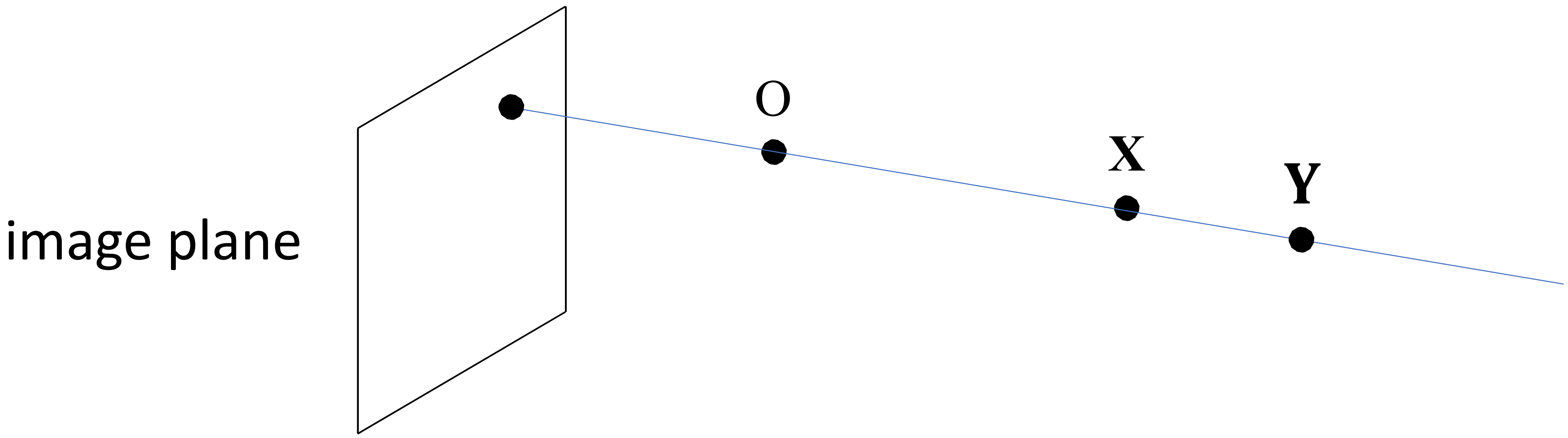
A point $\mathbf{X}$ and its image

$$\mathbf{O} = RNS(P):\ P\mathbf{O} = 0$$



A point $\mathbf{X}$ and its image;
any point $\mathbf{Y}$ on line$(\mathbf{O}, \mathbf{X})$:

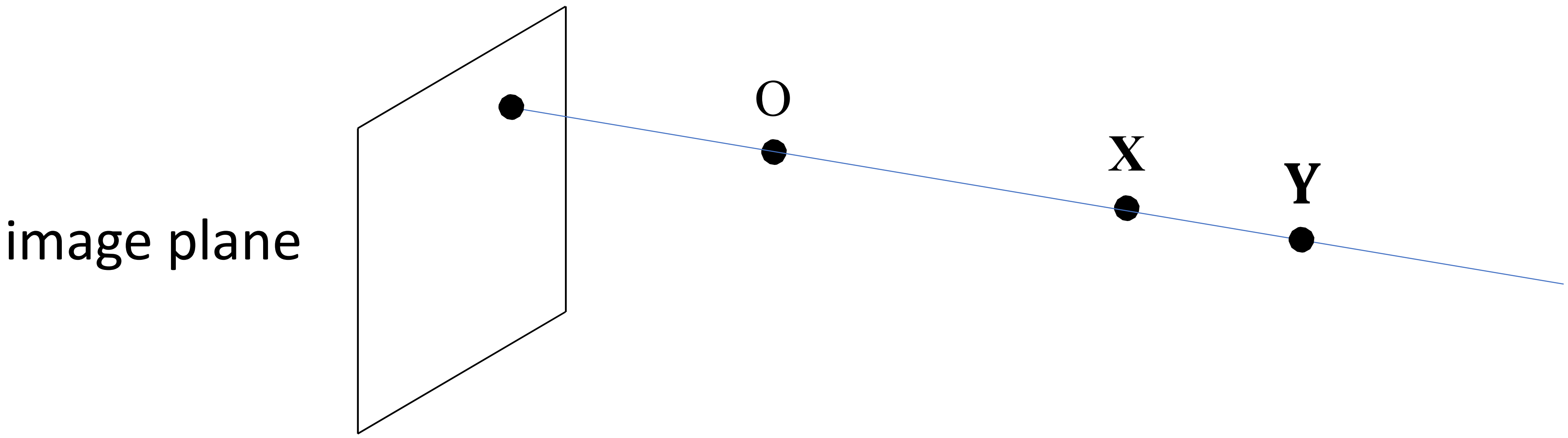
$$O = RNS(P): \; PO = 0$$



A point $\mathbf{X}$ and its image;
any point $\mathbf{Y}$ on line$(\mathbf{O}, \mathbf{X})$:

$$\mathbf{Y} = \boldsymbol{\alpha}\mathbf{X} + \boldsymbol{\beta}\mathbf{O}$$

$$\mathrm{O} = RNS(P):\ P\mathrm{O} = 0$$



A point $\mathbf{X}$ and its image;

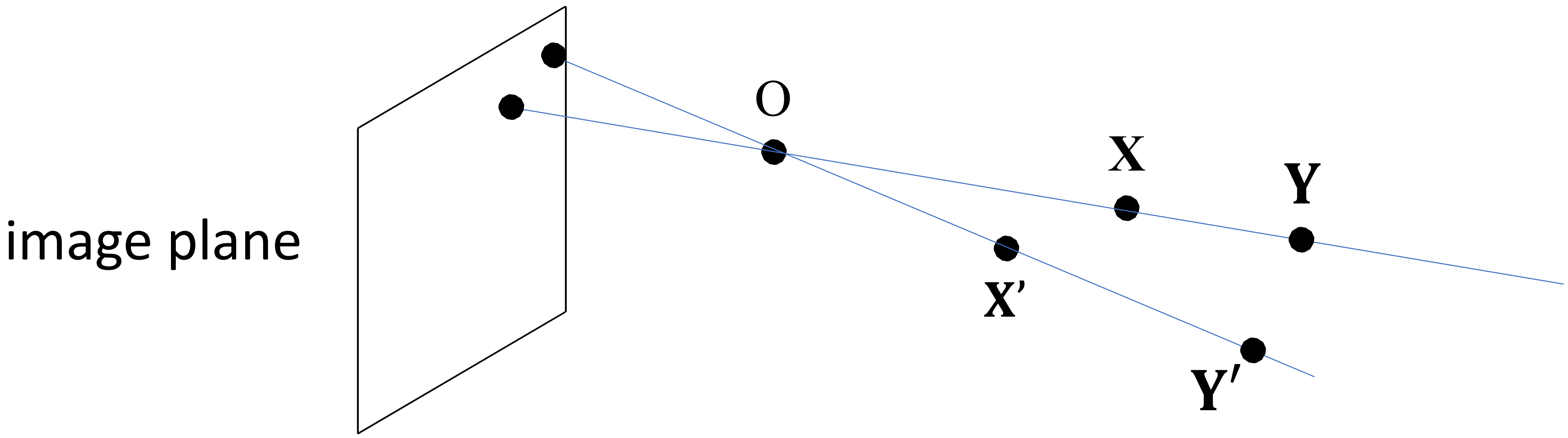
any point $\mathbf{Y}$ on line$(\mathbf{O}, \mathbf{X})$:

$$\mathbf{Y} = \boldsymbol{\alpha}\mathbf{X} + \boldsymbol{\beta}\mathbf{O}$$

also projects onto the image of $\mathbf{X}$

$$P\mathbf{Y} = \boldsymbol{\alpha}P\mathbf{X} + \boldsymbol{\beta}P\mathrm{O} = \boldsymbol{\alpha}P\mathbf{X} \approx P\boldsymbol{X}$$

the same for any other point



A point $\mathbf{X}$ and its image;
any point $\mathbf{Y}$ on line$(\mathbf{O}, \mathbf{X})$:

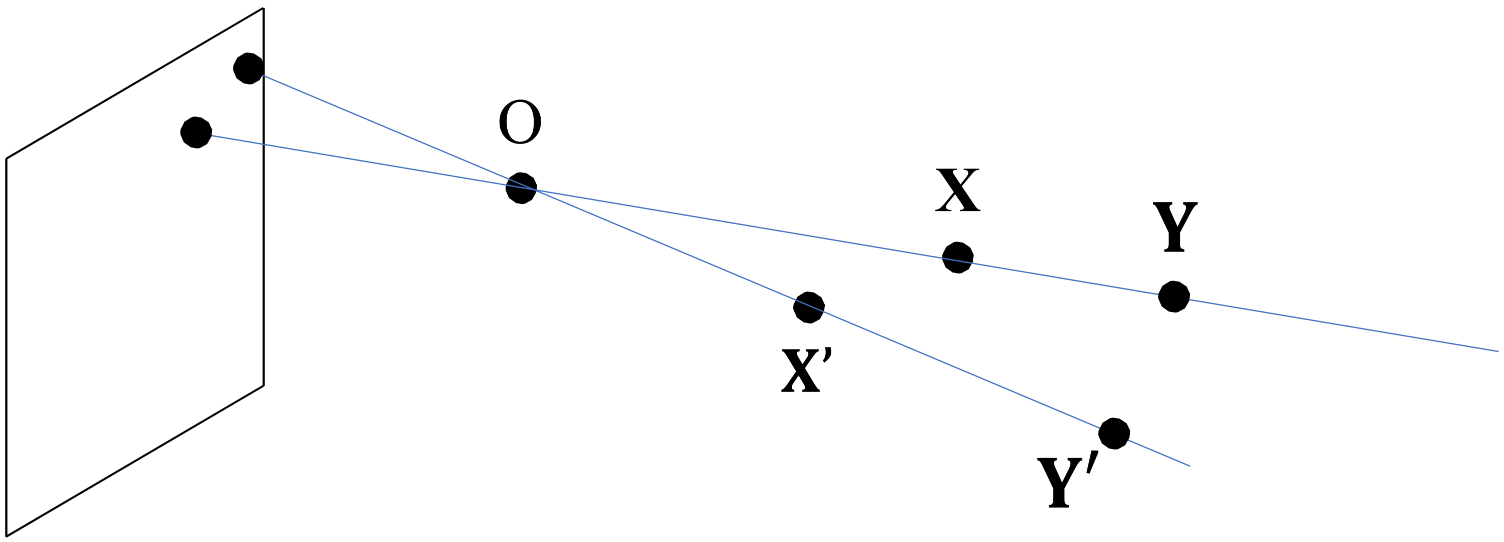
$$\mathbf{Y} = \boldsymbol{\alpha}\mathbf{X} + \boldsymbol{\beta}\mathbf{O}$$

also projects onto the image of $\mathbf{X}$

$$P\mathbf{Y} = \boldsymbol{\alpha}P\mathbf{X} + \boldsymbol{\beta}PO = \boldsymbol{\alpha}P\mathbf{X} \approx P\boldsymbol{X}$$

the same for any other point

image plane



A point $\mathbf{X}$ and its image;
any point $\mathbf{Y}$ on line$(\mathbf{O},\mathbf{X})$:

$$\mathbf{Y} = \boldsymbol{\alpha}\mathbf{X} + \boldsymbol{\beta}\mathbf{O}$$

also projects onto the image of $\mathbf{X}$

$$P\mathbf{Y} = \boldsymbol{\alpha}P\mathbf{X} + \boldsymbol{\beta}P\mathrm{O} = \boldsymbol{\alpha}P\mathbf{X} \approx P\boldsymbol{X}$$

all viewing rays go through $\mathbf{O}$

the same for any other point

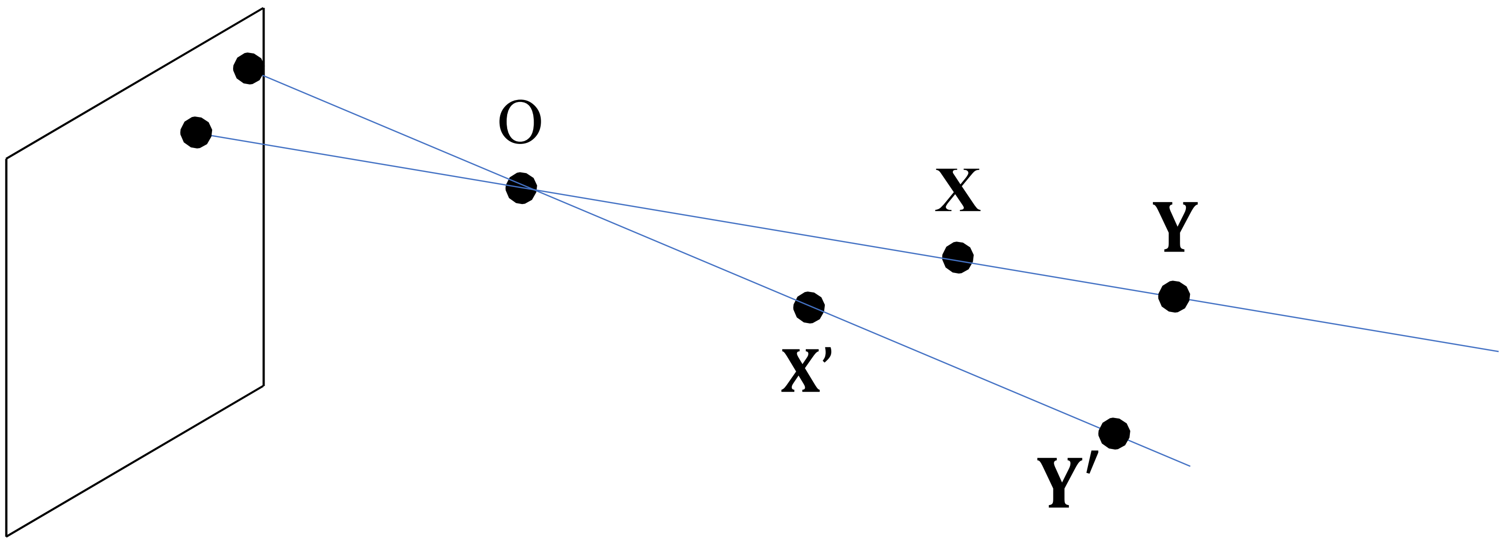


image plane

A point $\mathbf{X}$ and its image;
any point $\mathbf{Y}$ on line$(\mathbf{O}, \mathbf{X})$:

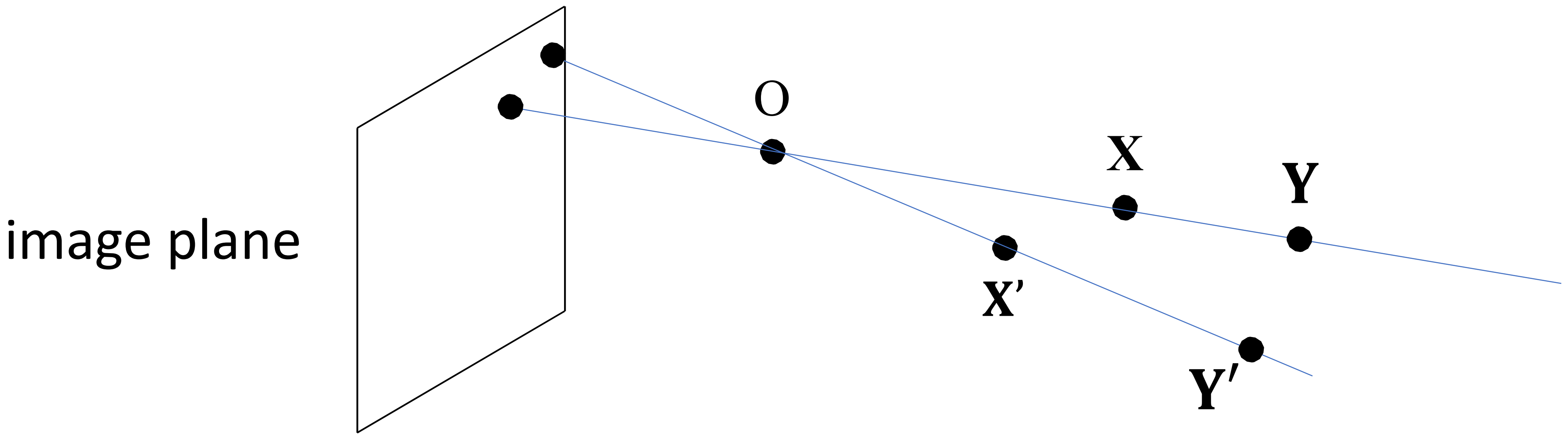
$$\mathbf{Y} = \boldsymbol{\alpha}\mathbf{X} + \boldsymbol{\beta}\mathbf{O}$$

also projects onto the image of $\mathbf{X}$

$$P\mathbf{Y} = \boldsymbol{\alpha}P\mathbf{X} + \boldsymbol{\beta}PO = \boldsymbol{\alpha}P\mathbf{X} \approx P\boldsymbol{X}$$

all viewing rays go through $\mathbf{O}$
$\mathbf{O}$ is the camera lens center

the same for any other point



A point $\mathbf{X}$ and its image;
any point $\mathbf{Y}$ on line$(\mathbf{O}, \mathbf{X})$:

$$\mathbf{Y} = \boldsymbol{\alpha}\mathbf{X} + \boldsymbol{\beta}\mathbf{O}$$

also projects onto the image of $\mathbf{X}$

$$P\mathbf{Y} = \boldsymbol{\alpha}P\mathbf{X} + \boldsymbol{\beta}PO = \boldsymbol{\alpha}P\mathbf{X} \approx PX$$

all viewing rays go through $\mathbf{O}$

$\mathbf{O}$ is the camera lens center

$$PO = [\mathbf{M} \quad \mathbf{m}]\begin{bmatrix}\mathbf{o}\\ 1\end{bmatrix} = 0$$

$\rightarrow \mathbf{o} = -\mathbf{M}^{-1}\mathbf{m}$

cartesian coordinates of $\mathbf{O}$

SCENE: viewing ray from image point

null-space of camera projection matrix **O** $\quad \mathrm{P}\mathbf{O} = 0$

a point **Y** on the line **X**, **O** $\quad \mathbf{Y} = \alpha\mathbf{X} + \beta\mathbf{O}$

its image $\quad \mathbf{u} = \mathbf{PY} = \alpha\mathbf{PX} + \beta\mathrm{P}\mathbf{O} = \mathbf{PX}$

all points Y on (X,O) project on image of X, $\rightarrow$ **all points on a line through O share the same image**

$\rightarrow$ $\boxed{\textbf{O} \text{ is camera center}}$

Image of camera center is $(0,0,0)^T$, i.e. undefined

Finite cameras: $\quad \mathbf{O} = \begin{pmatrix} \mathbf{o} \\ 1 \end{pmatrix} = \begin{pmatrix} -\mathrm{M}^{-1}\mathbf{m} \\ 1 \end{pmatrix}$

Viewing ray associated to image point **u**:

image point **u** is image of **X** if **X** is on a certain line through **O**

viewing ray associated to u

Consider the point at the infinity along this line $\mathbf{X} = \begin{vmatrix} \mathbf{d} \\ 0 \end{vmatrix}$

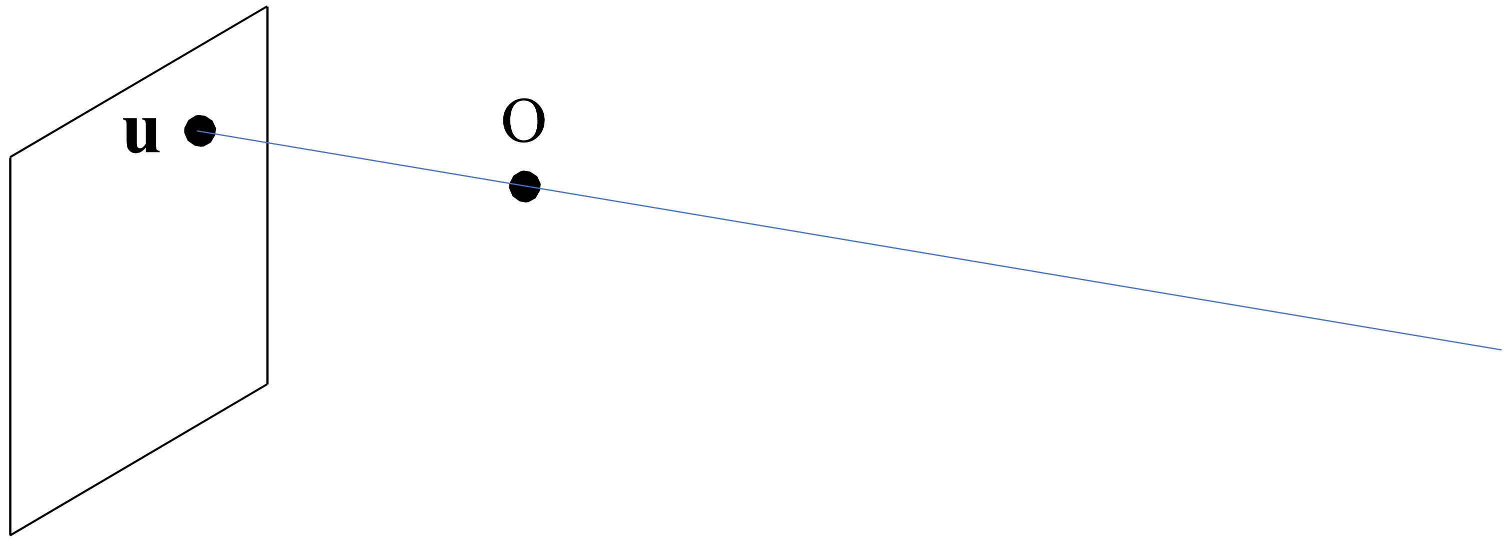
$$\mathbf{u} = \begin{vmatrix} \mathbf{M} & \mathbf{m} \end{vmatrix} \begin{vmatrix} \mathbf{d} \\ 0 \end{vmatrix} = \mathbf{M} \cdot \mathbf{d}$$

The locus of the points **x** *whose image is* **u**
is a straight line through **o** *having direction* $\mathbf{d} = \mathbf{M}^{-1} \cdot \mathbf{u}$

o is the camera viewpoint (perspective projection center)

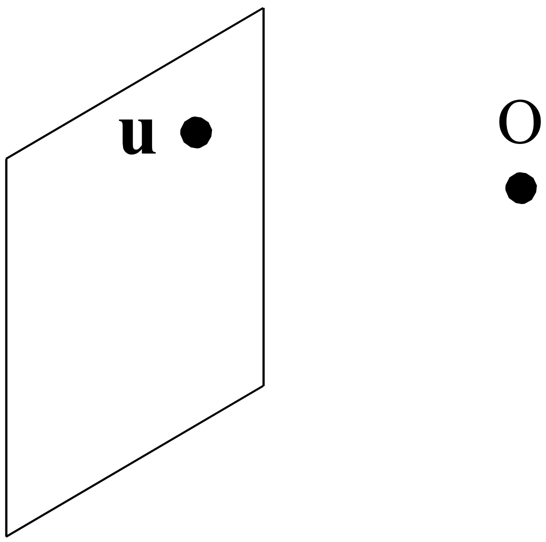
line(**o**, **d**) = *viewing ray associated to image point* **u**

Viewing ray associated to an image point **u**: the «backprojection» of **u**



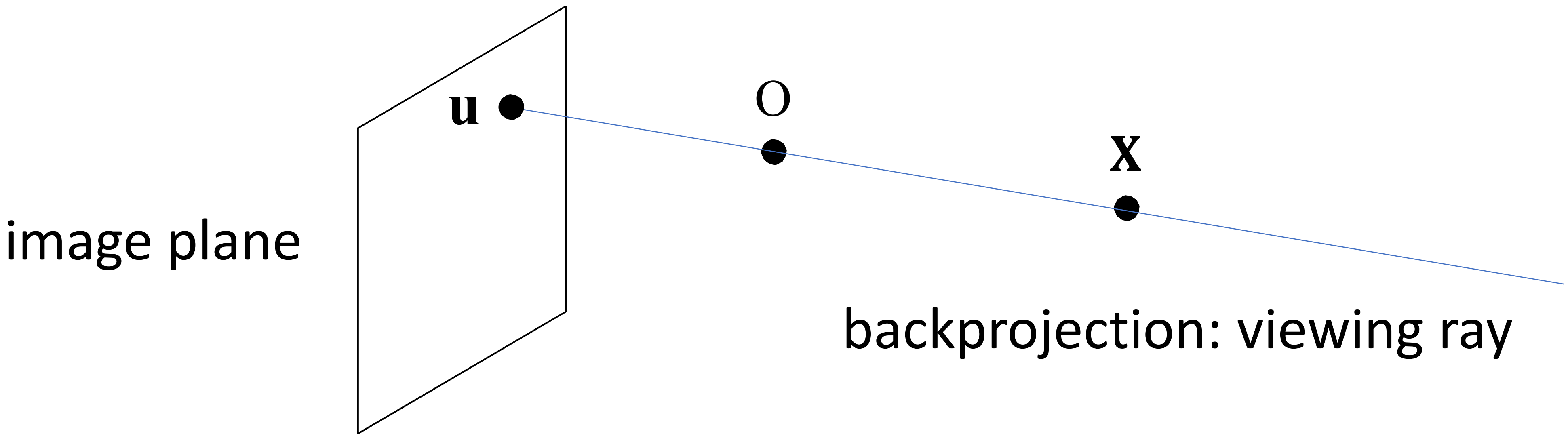
$$\mathrm{O} = RNS(P):\ P\mathrm{O} = 0$$

image plane



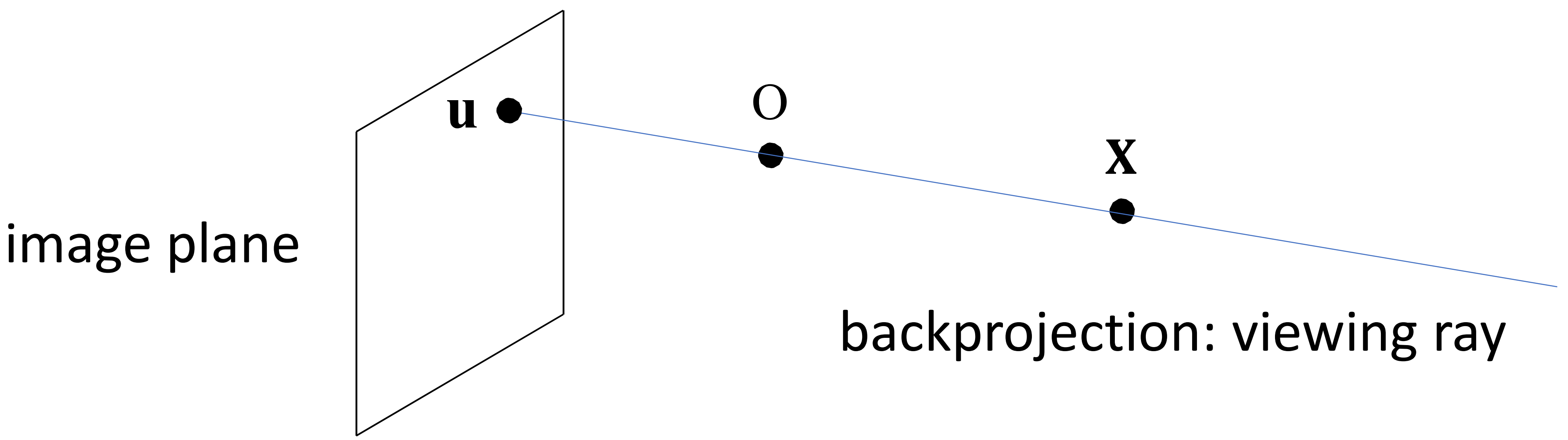
An image point **u**

$$\mathrm{O} = RNS(P): \; P\mathrm{O} = 0$$



An image point $\mathbf{u}$ and its backprojection: set of 3D points $\mathbf{X}$ projecting onto $\mathbf{u}$

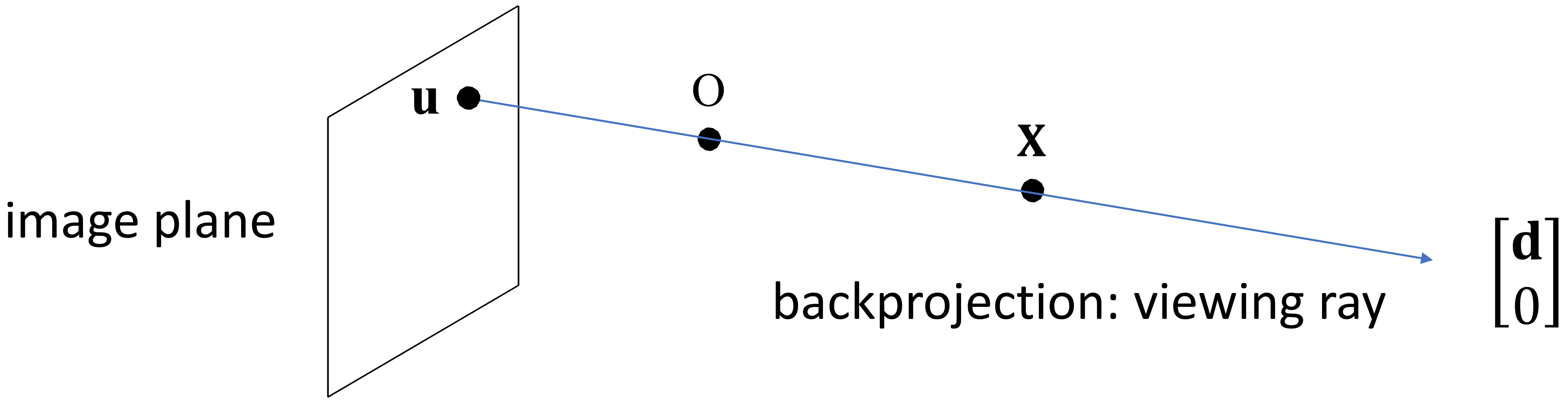
$$O = RNS(P):\ PO = 0$$



An image point $\mathbf{u}$ and its backprojection: set of 3D points $\mathbf{X}$ projecting onto $\mathbf{u}$

(i) a line through O

$$\mathrm{O} = RNS(P):\ P\mathrm{O} = 0$$

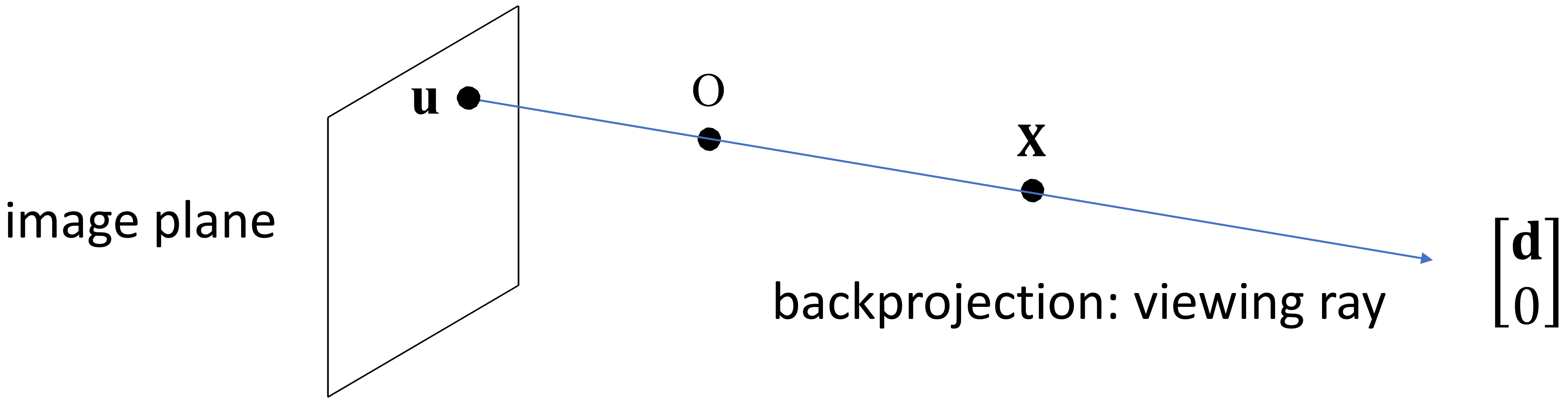


An image point $\mathbf{u}$ and its backprojection: set of 3D points $\mathbf{X}$ projecting onto $\mathbf{u}$

(i) a line through O

(ii) whose direction $\mathbf{d}$ …

$$O = RNS(P): \; PO = 0$$

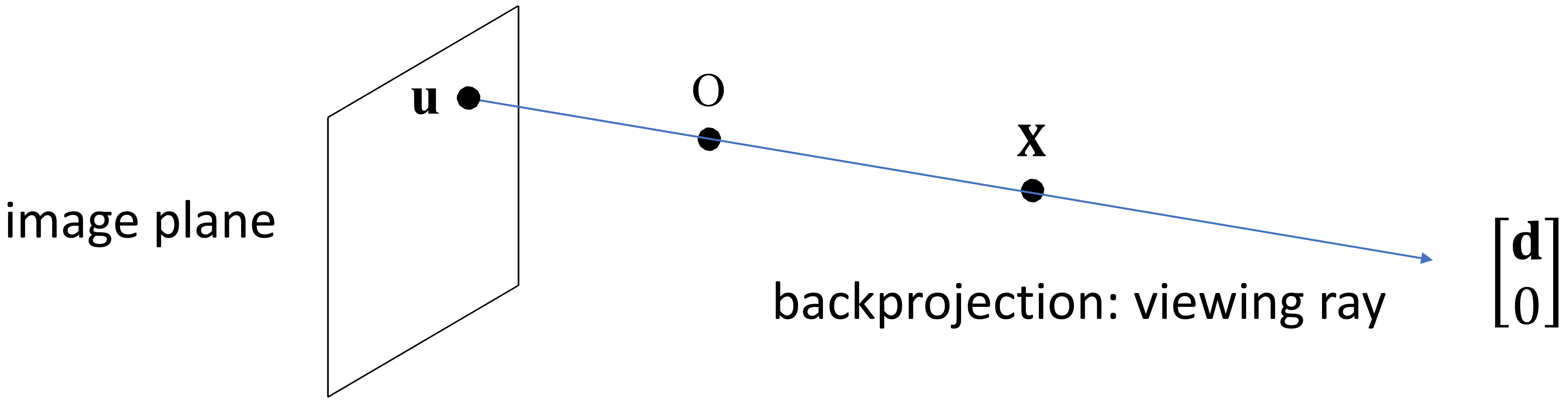


An image point $\mathbf{u}$ and its backprojection: set of 3D points $\mathbf{X}$ projecting onto $\mathbf{u}$

(i) a line through O

(ii) whose direction $\mathbf{d}$ satisfies $\mathbf{u} = [\mathbf{M} \quad \mathbf{m}]\begin{bmatrix}\mathbf{d}\\0\end{bmatrix} = \mathbf{M}\mathbf{d}$

$$O = RNS(P): \; PO = 0$$



An image point $\mathbf{u}$ and its backprojection: set of 3D points $\mathbf{X}$ projecting onto $\mathbf{u}$

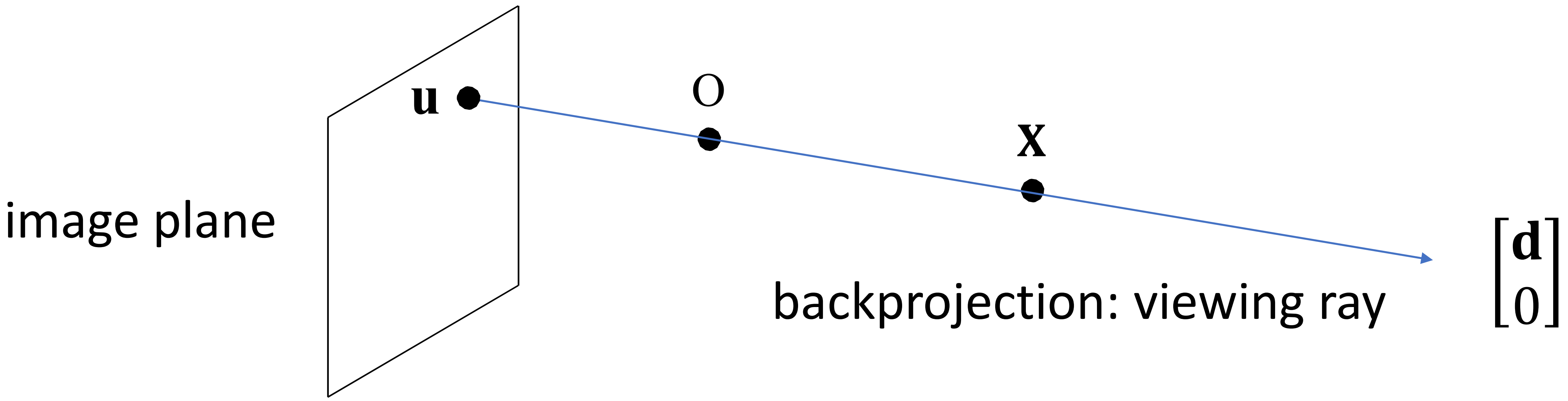
(i) a line through O

(ii) whose direction $\mathbf{d}$ satisfies $\mathbf{u} = [\mathbf{M} \quad \mathbf{m}]\begin{bmatrix}\mathbf{d}\\0\end{bmatrix} = \mathbf{Md}$

$\rightarrow$

$$\mathbf{d} = \mathbf{M}^{-1}\mathbf{u}$$

the viewing ray associated to an image point $\mathbf{u}$



The backprojection of image point $\mathbf{u}$, i.e., the viewing ray associated to $\mathbf{u}$, for the camera $P = [\mathbf{M} \quad \mathbf{m}]$, is the straight line

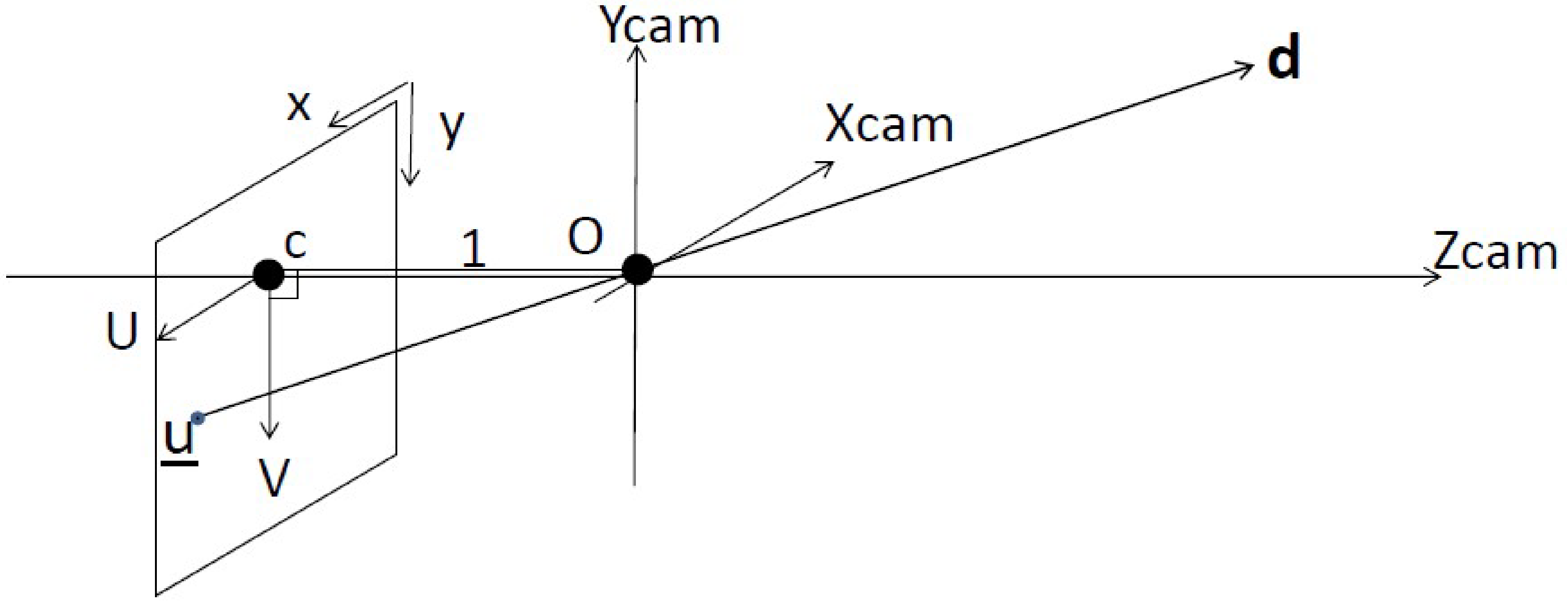
(i) through $\mathrm{O} = RNS(P)$

(ii) whose direction $\mathbf{d}$ is $\mathbf{d} = \mathbf{M}^{-1}\mathbf{u}$

- SCENE
- **CAMERA**

CAMERA

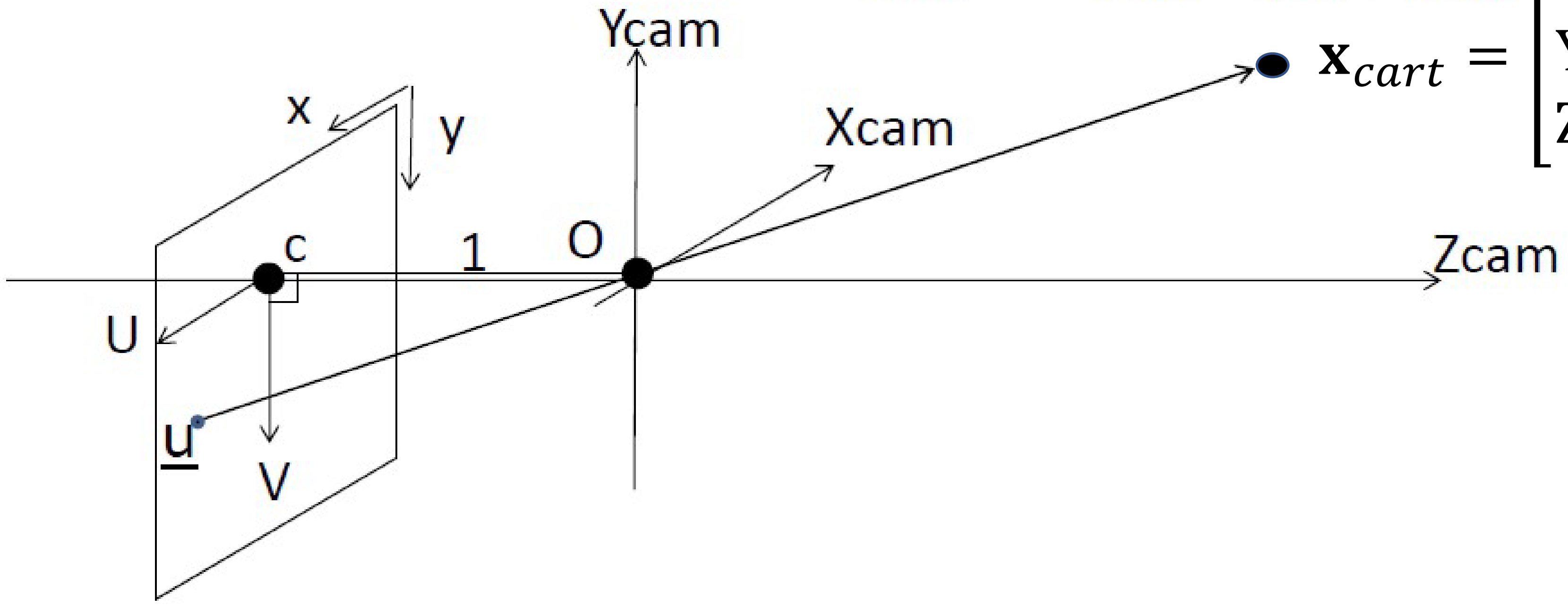
$$\mathbf{d}_{cam} = \mathbf{R}_{cam->world}\mathbf{d}_{world}$$



geometric coordinates

$$\begin{vmatrix} U \\ V \\ 1 \end{vmatrix} = \begin{vmatrix} X \\ Y \\ -1 \end{vmatrix} \propto \mathbf{d}_{cam}$$

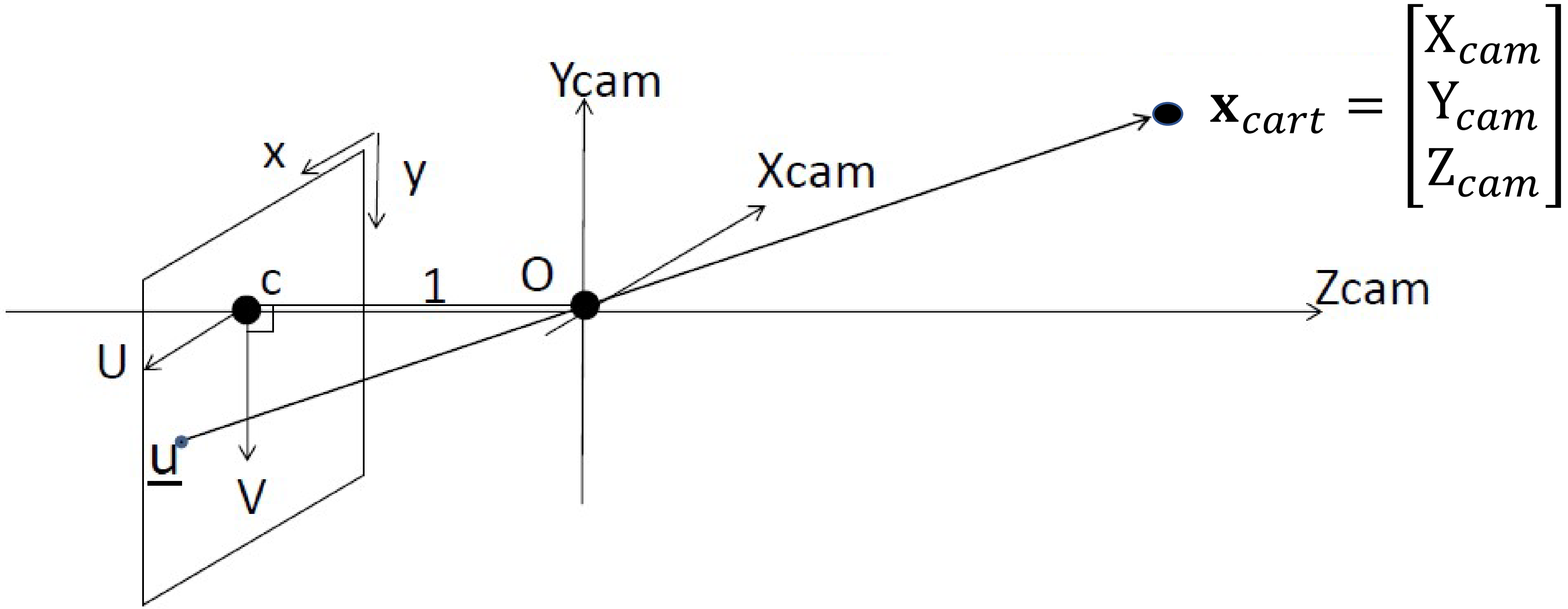
camera



$$\mathbf{x}_{cart} = \begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \end{bmatrix}$$

geometric coordinates $\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} \propto \begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \end{bmatrix} = \mathbf{x}_{cart}$

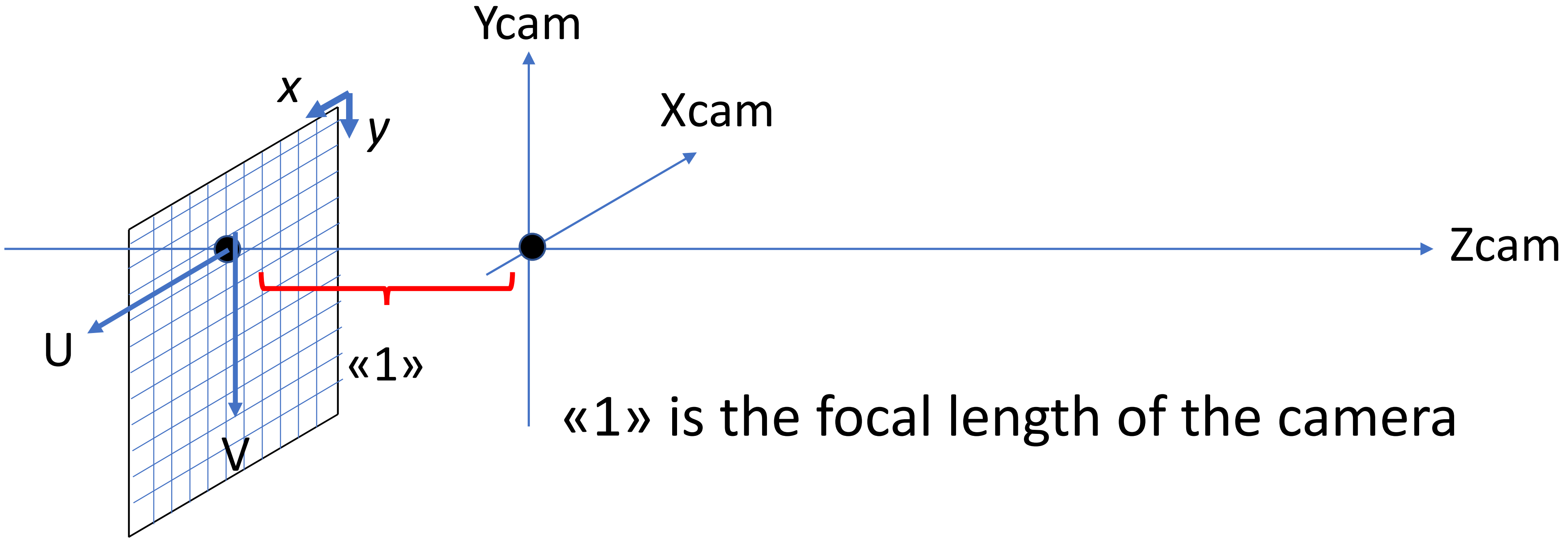
camera



geometric coordinates $\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} \propto \begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \end{bmatrix} = \mathbf{x}_{cart}$

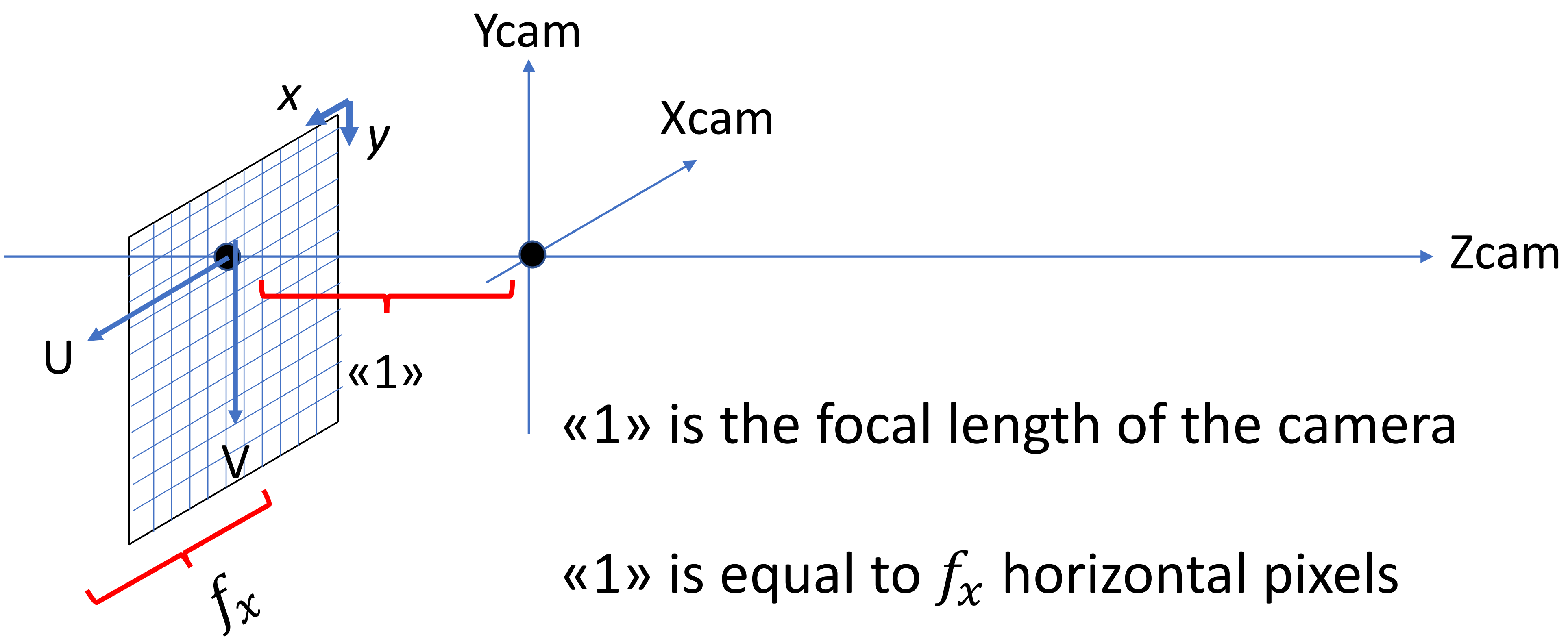
geometric coordinates vs pixel coordinates ??

pixel coordinates



«1» is the focal length of the camera

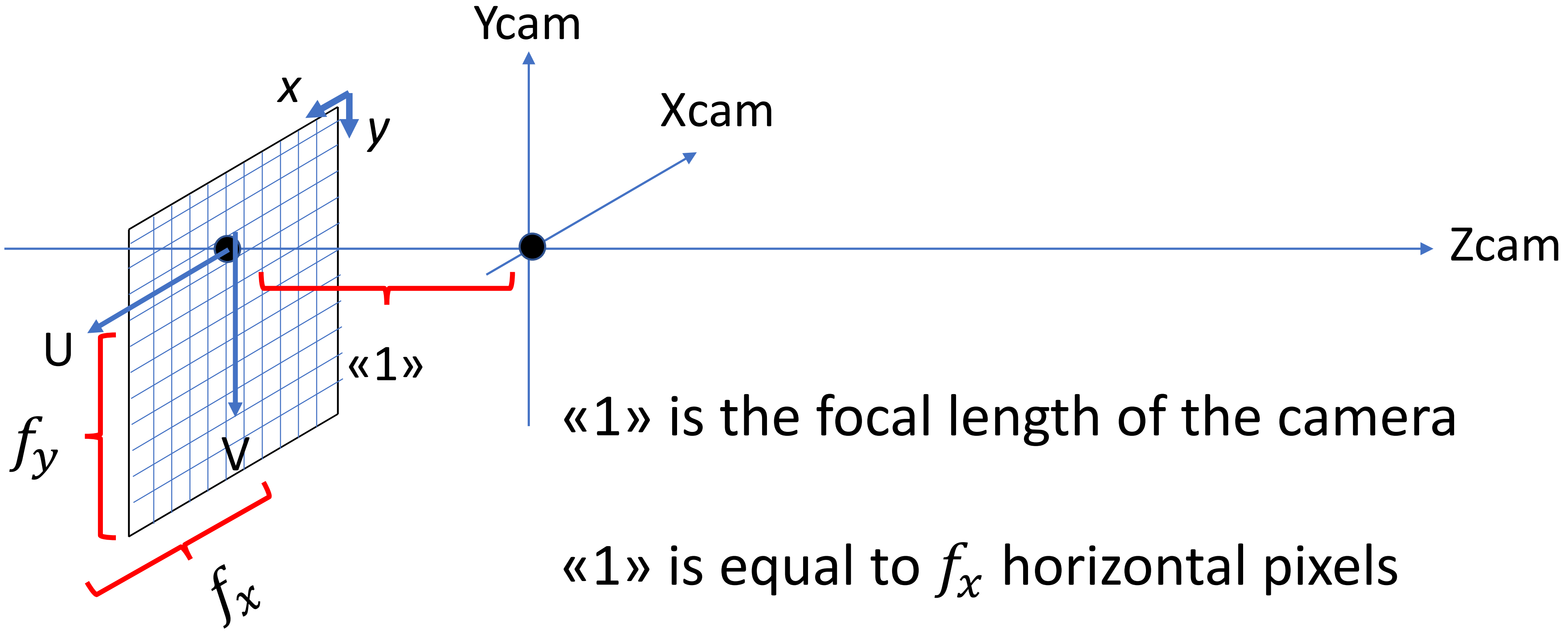
pixel coordinates



«1» is the focal length of the camera

«1» is equal to f_x horizontal pixels

pixel coordinates

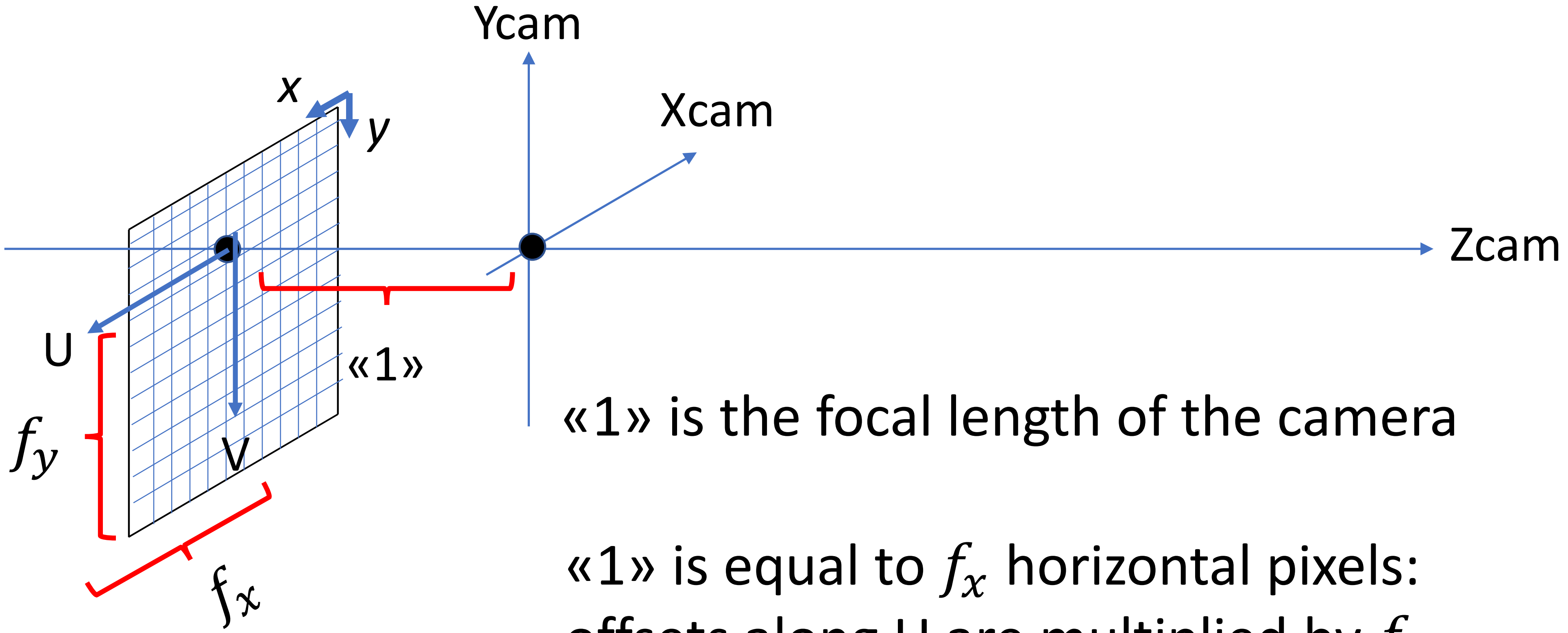


«1» is the focal length of the camera

«1» is equal to f_x horizontal pixels

«1» is equal to f_y vertical pixels

pixel coordinates

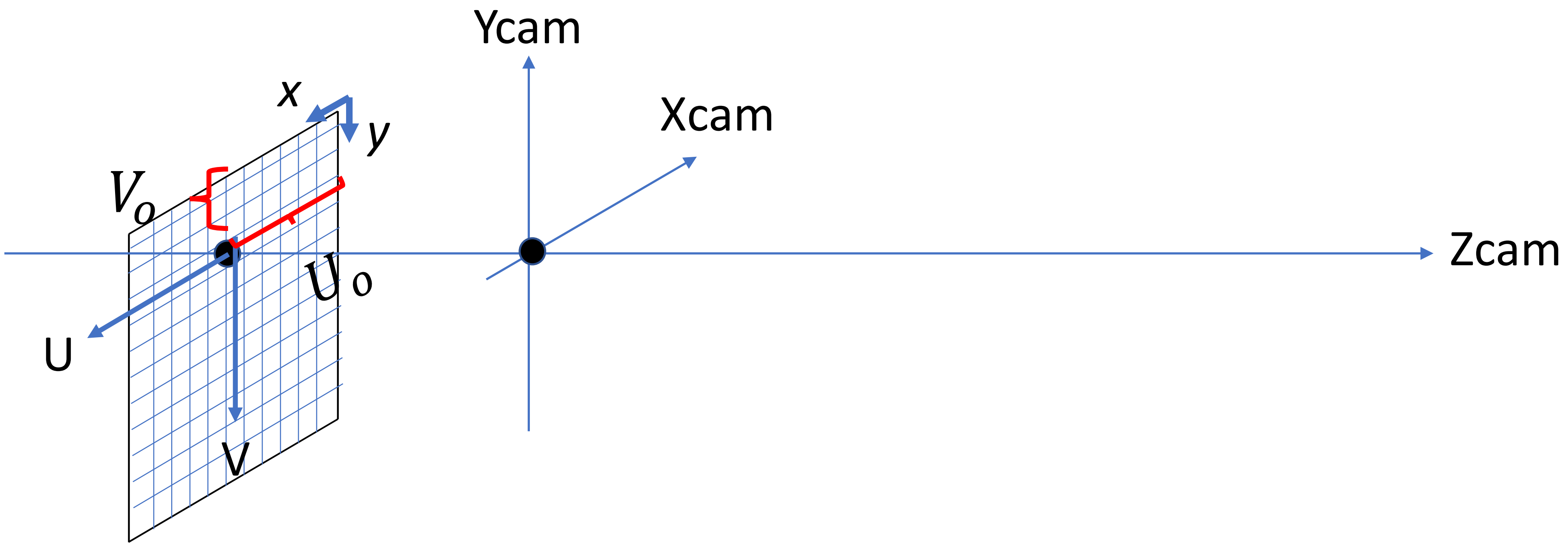


«1» is the focal length of the camera

«1» is equal to f_x horizontal pixels:
offsets along U are multiplied by f_x

«1» is equal to f_y vertical pixels:
offsest along V are mutiplied by f_y

pixel coordinates

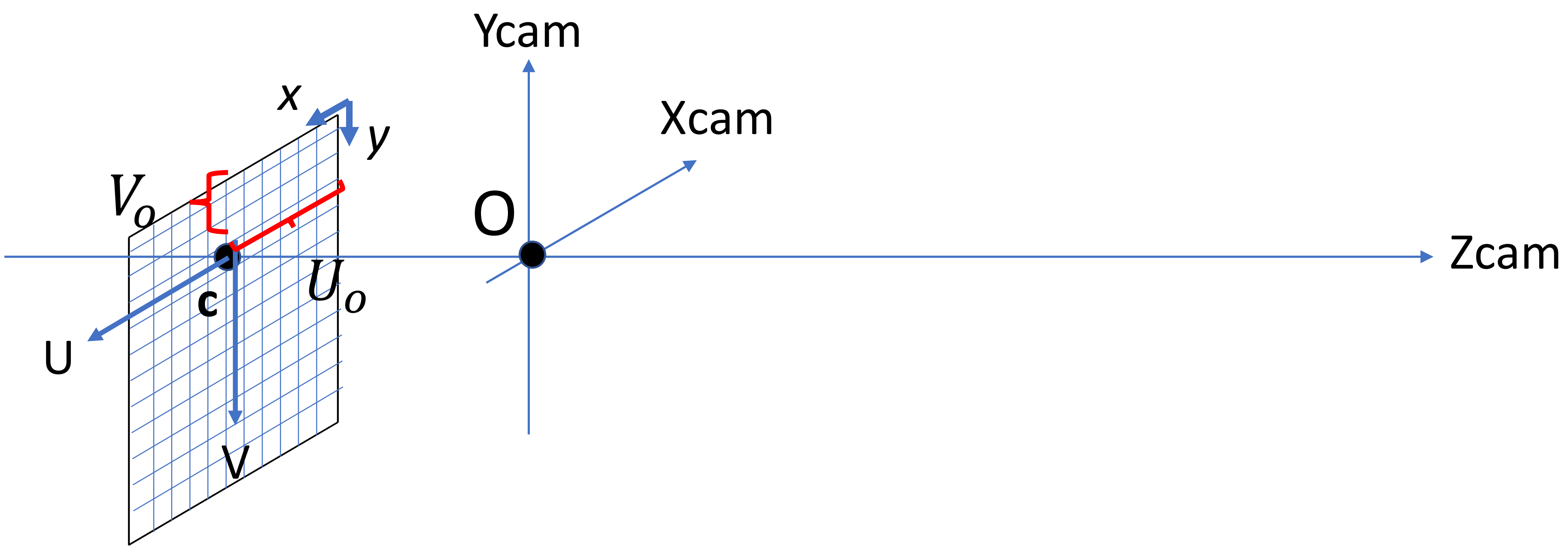


Camera **principal point**: projection of the camera center onto image plane

Pixel coordinates of the principal point:

$$(U_o, V_o)$$

pixel coordinates

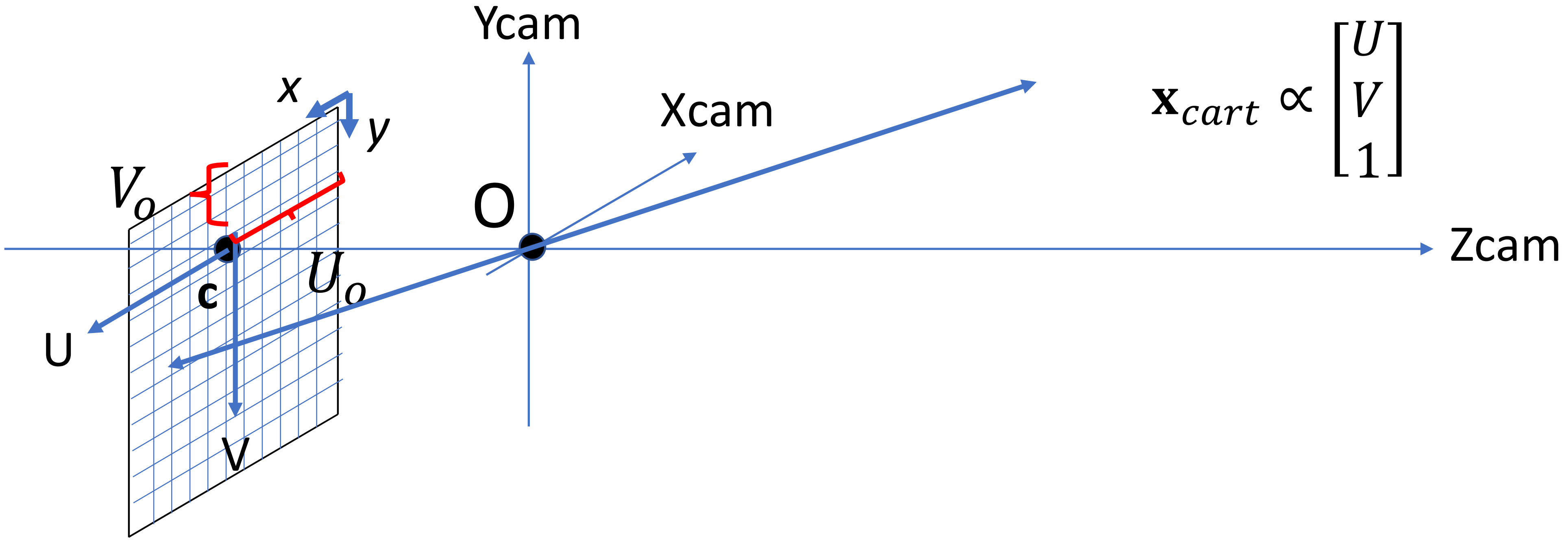


Camera **principal point**: projection of the camera center O onto image plane

Pixel coordinates of the principal point:

$$\mathbf{c} = (U_o, V_o)$$

components of $\mathbf{d}$

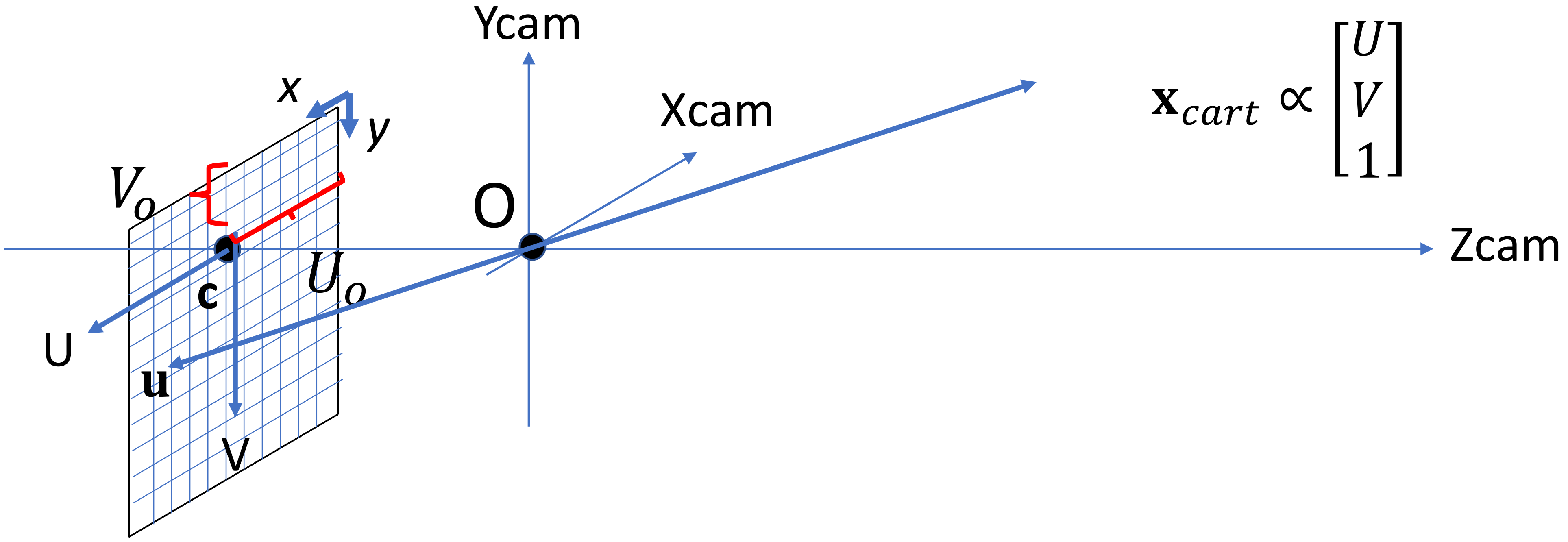


Camera **principal point**: projection of the camera center O onto image plane

Pixel coordinates of the principal point:

$$\mathbf{c} = (U_o, V_o)$$

geometric coordinates of the image point: (U, V)



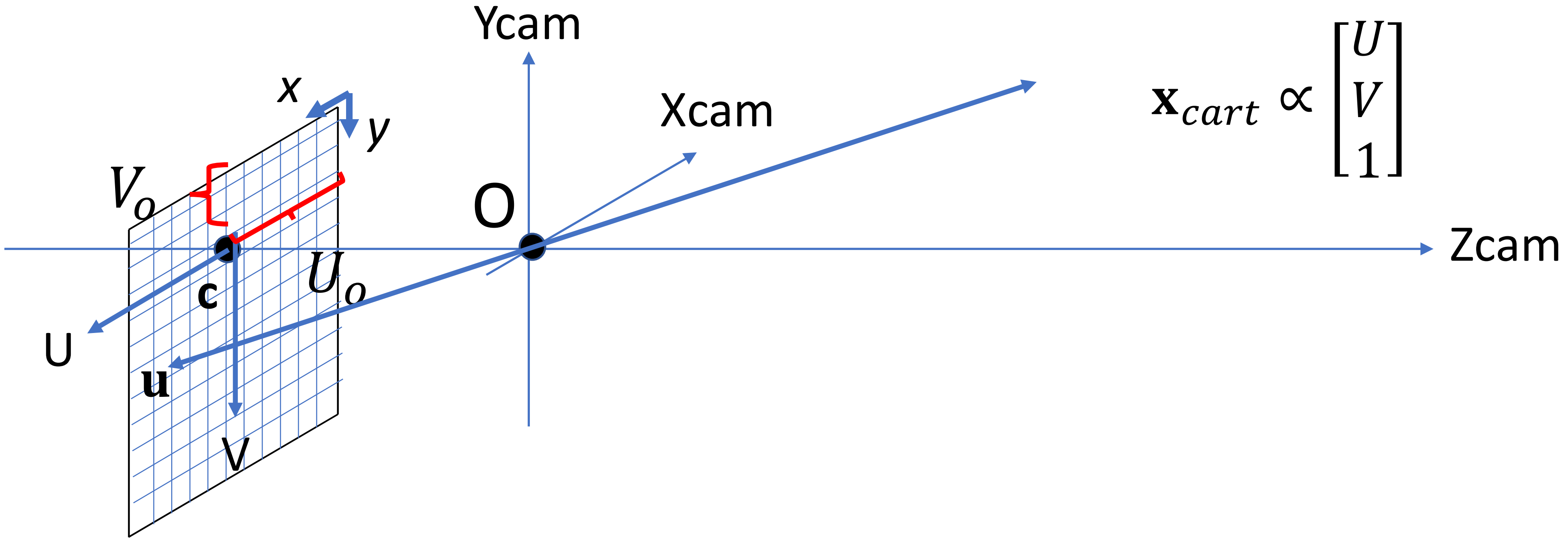
$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

Camera **principal point**: projection of the camera center O onto image plane

Pixel coordinates of the principal point:

$$\mathbf{c} = (U_o, V_o)$$

Cartesian pixel coordinates of the image point: (x, y)



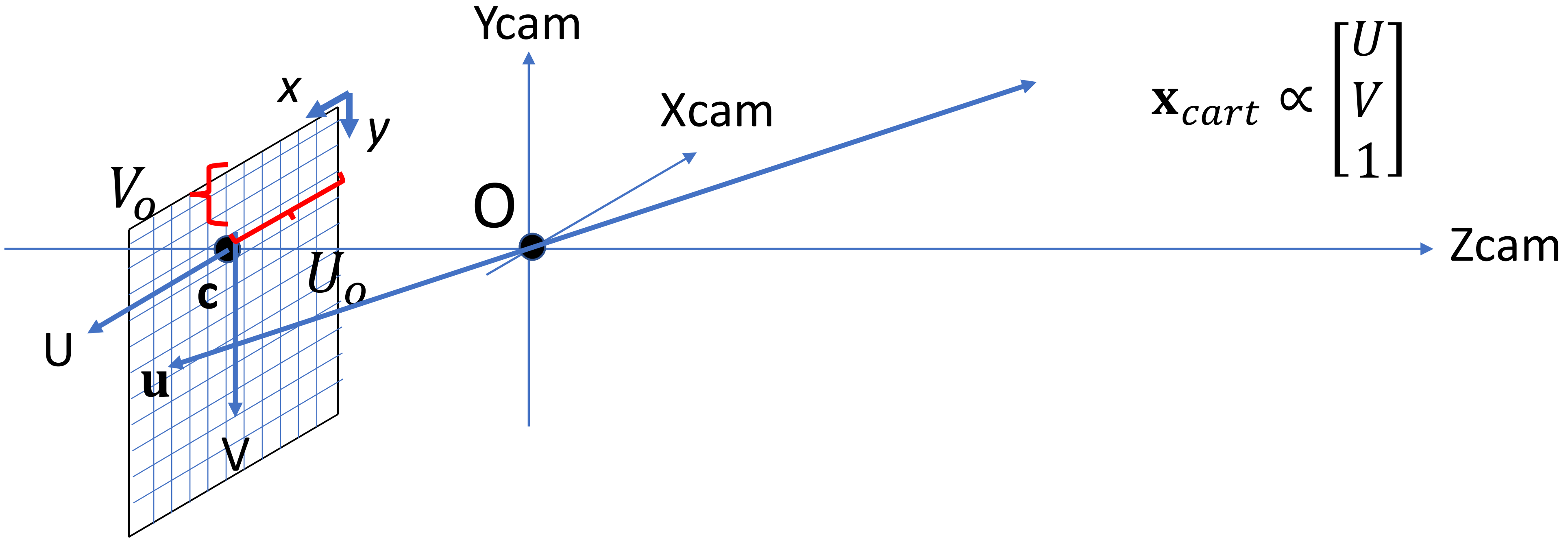
$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

Camera **principal point**: projection of the camera center O onto image plane

Pixel coordinates of the principal point:

$$\mathbf{c} = (U_o, V_o)$$

Homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$

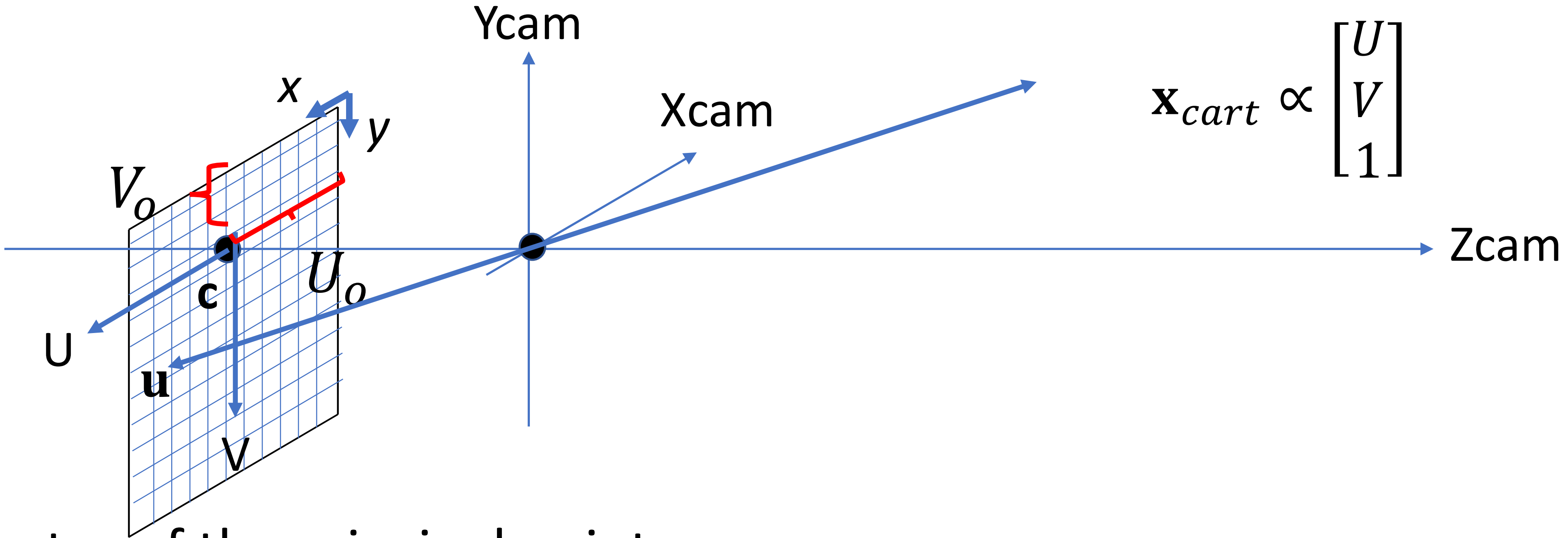


Camera **principal point**: projection of the camera center O onto image plane

Pixel coordinates of the principal point:

$$\mathbf{c} = (U_o, V_o)$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

Pixel coordinates of the principal point:

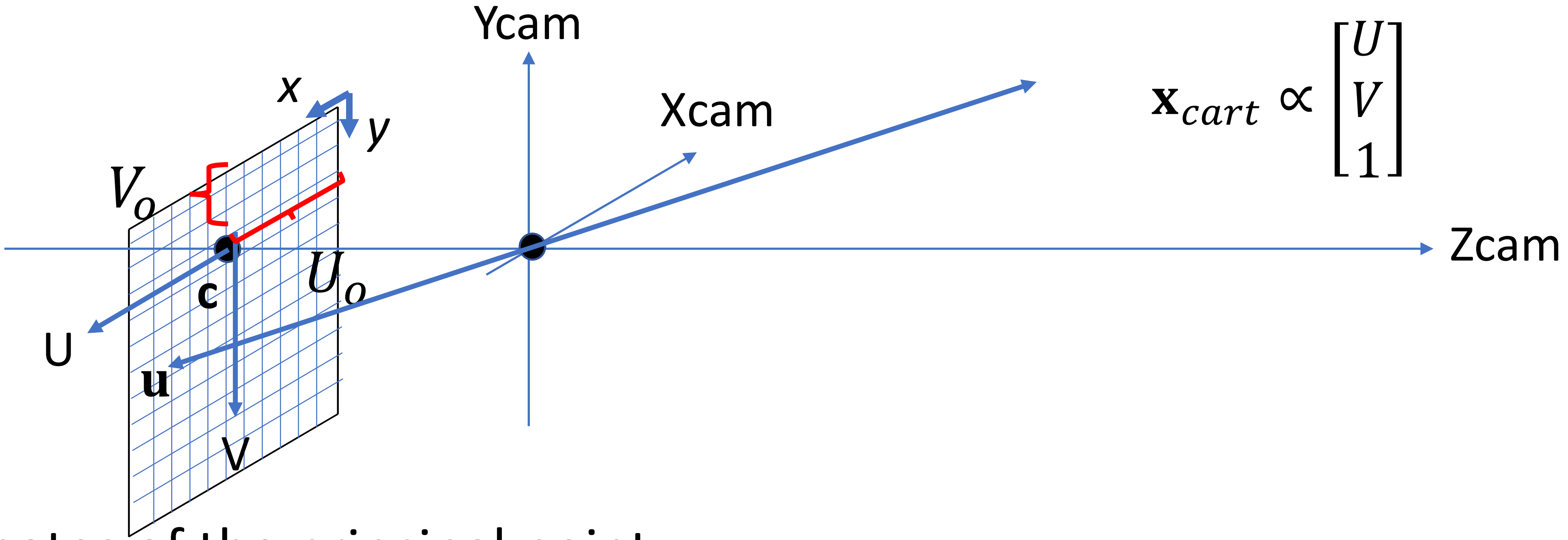
$$\mathbf{c} = (U_o, V_o)$$

«1» is equal to f_x horizontal pixels:
offsets along U are multiplied by f_x

«1» is equal to f_y vertical pixels:
offsest along V are mutiplied by f_y

$$\mathbf{u} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x U + U_o \\ f_y V + V_o \\ 1 \end{bmatrix}$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

Pixel coordinates of the principal point:

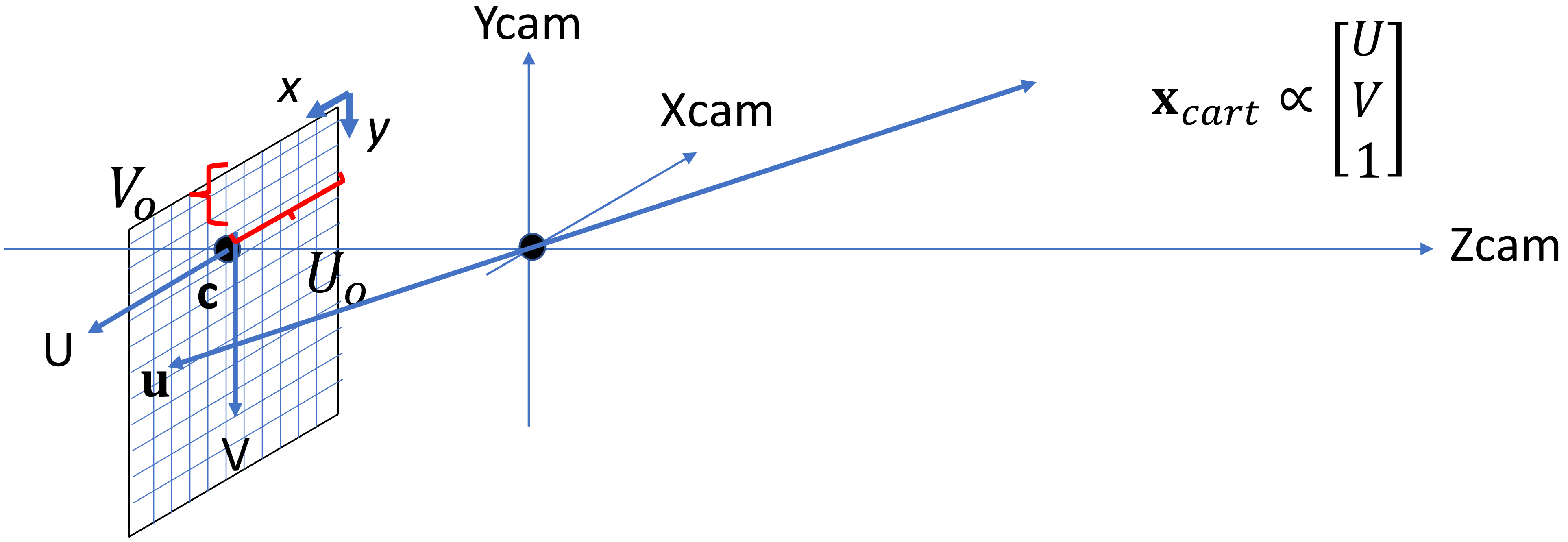
$$\mathbf{c} = (U_o, V_o)$$

«1» is equal to f_x horizontal pixels: offsets along U are multiplied by f_x

«1» is equal to f_y vertical pixels: offsest along V are mutiplied by f_y

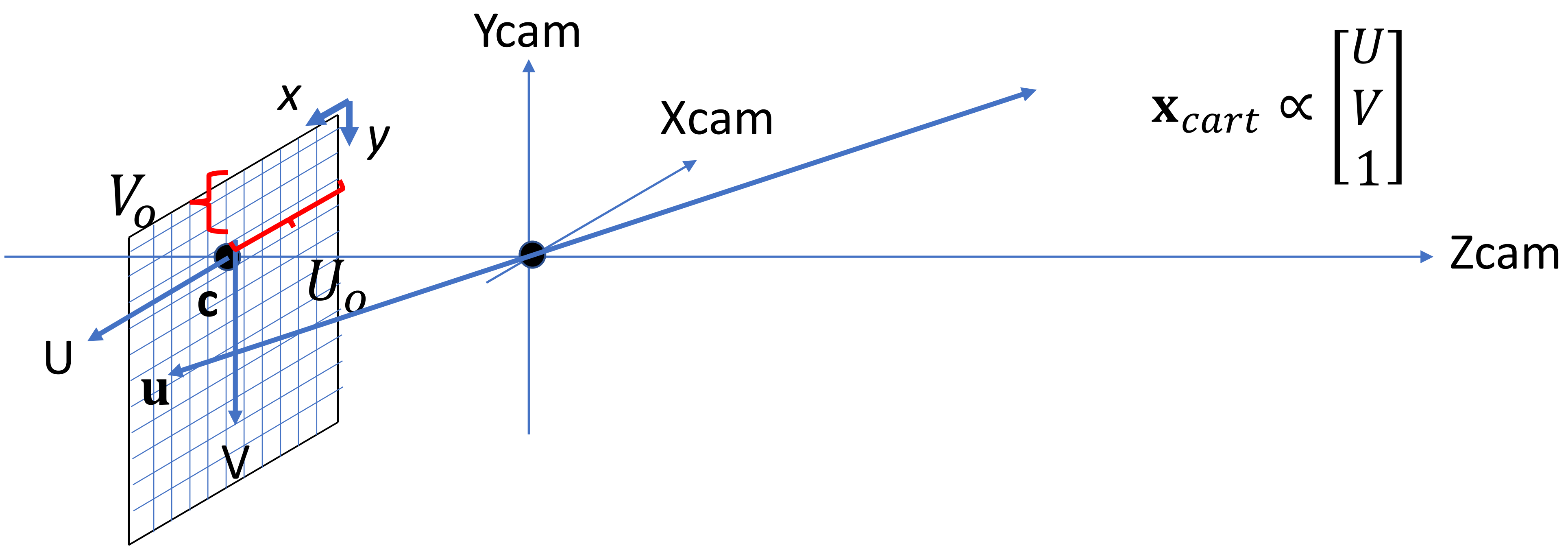
$$\mathbf{u} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x U + U_o \\ f_y V + V_o \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



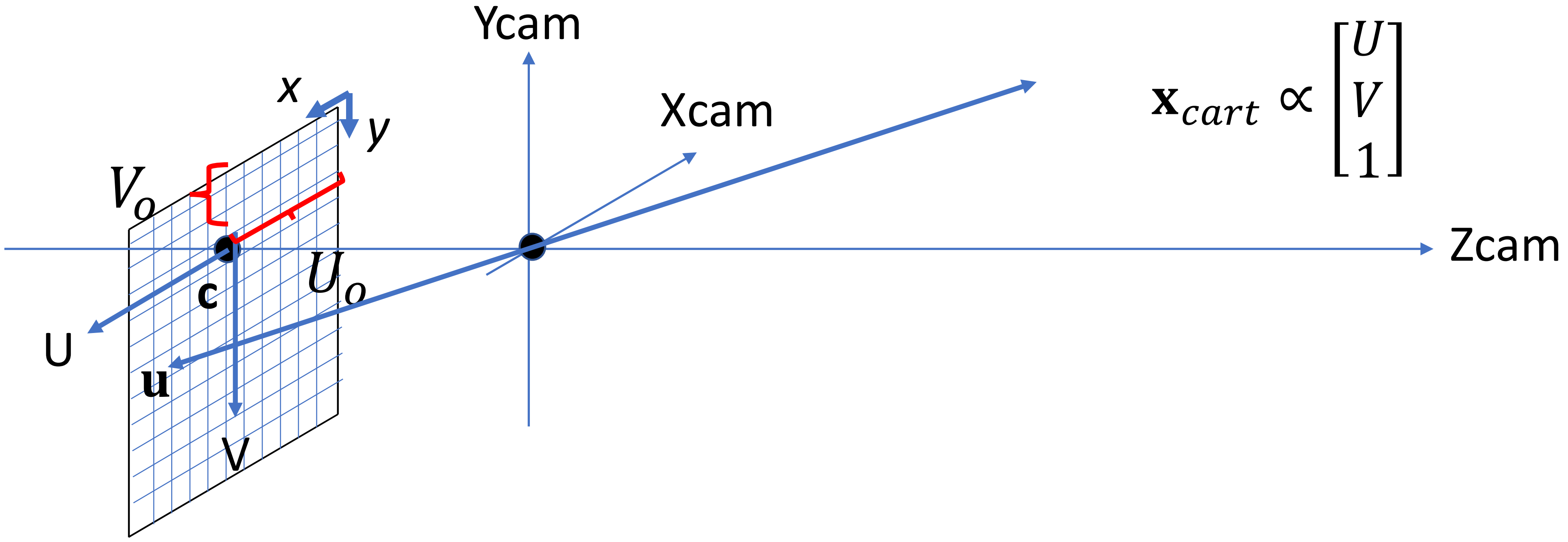
$$\mathbf{u} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x U + U_o \\ f_y V + V_o \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{cart}$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



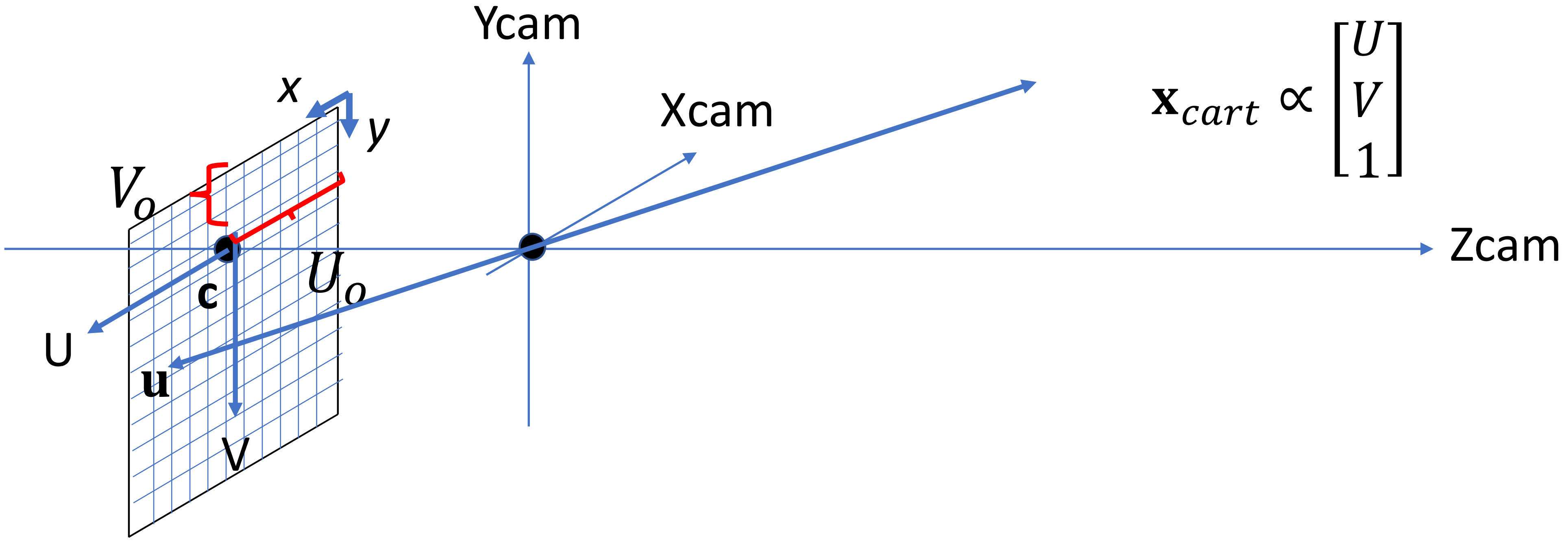
$$\mathbf{u} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x U + U_o \\ f_y V + V_o \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \mathbf{x}_{cart} = \mathbf{K}\, \mathbf{x}_{cart}$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



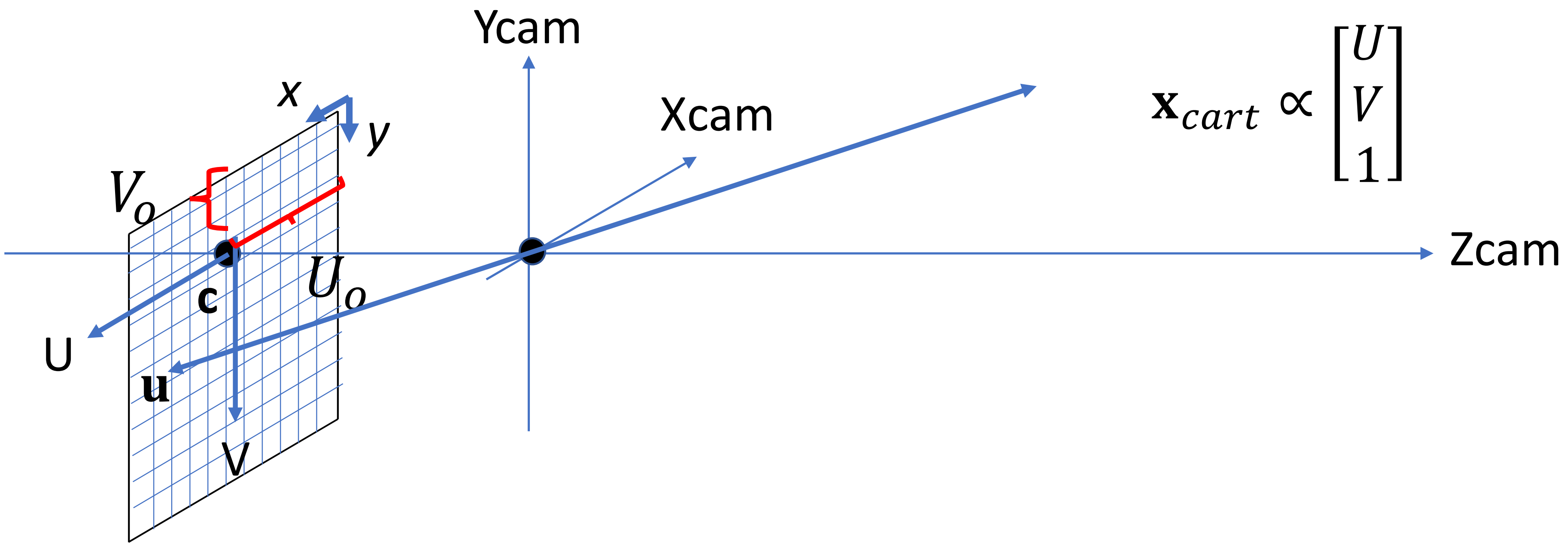
$$\mathbf{u} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x U + U_o \\ f_y V + V_o \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \mathbf{x}_{cart} = \mathbf{K} \begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \end{bmatrix}$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



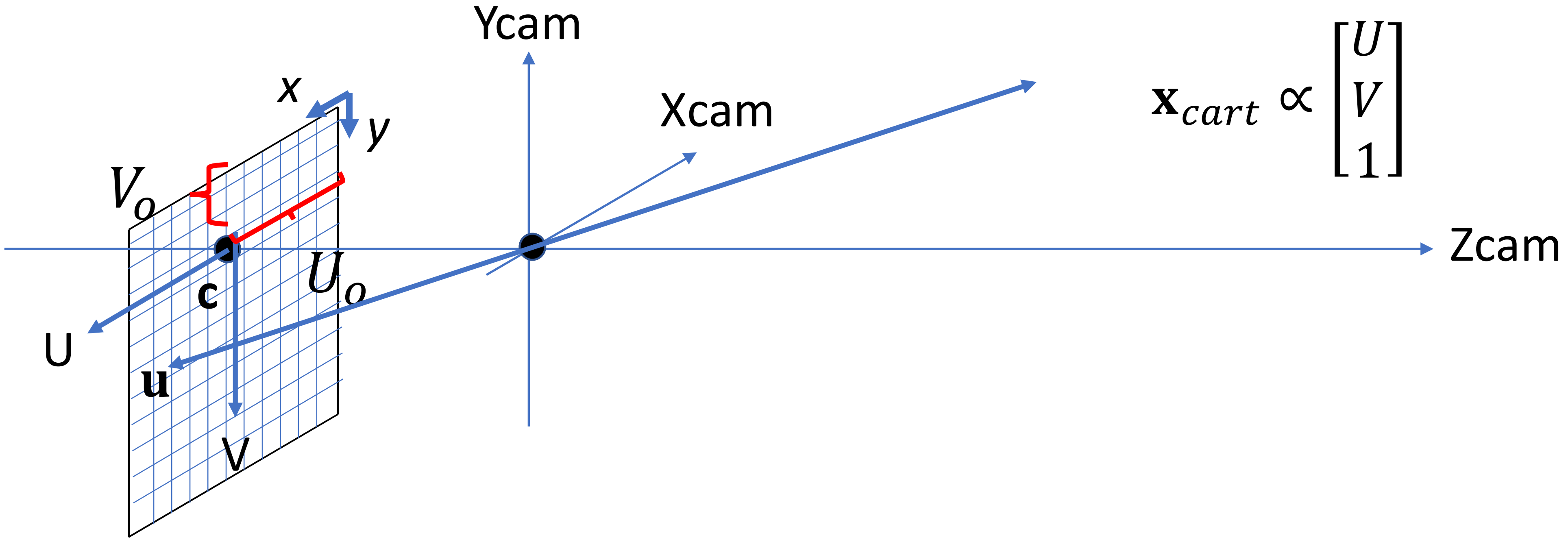
$$\mathbf{u} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \mathbf{x}_{cart} = \mathbf{K} \begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \end{bmatrix} = \begin{bmatrix} \mathbf{K} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \end{bmatrix} \begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \\ 1 \end{bmatrix}$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



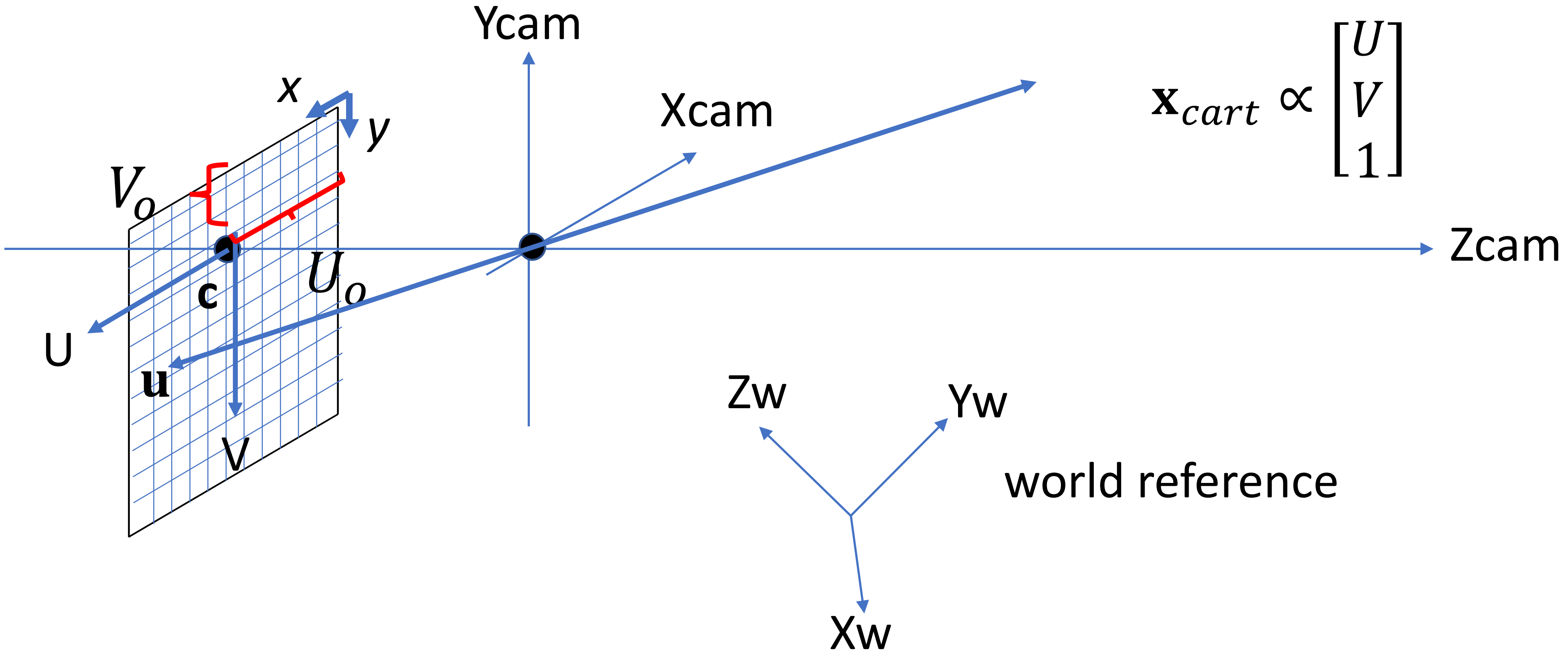
$$\mathbf{u} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} f_x & 0 & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{K}} \mathbf{x}_{cart} = \begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \end{bmatrix} = \begin{bmatrix} \mathbf{K} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \end{bmatrix} \mathbf{X}_{cam} = [\mathbf{K} \quad \mathbf{0}]\mathbf{X}_{cam}$$

homogeneous pixel coordinates of image point: $\mathbf{u} = [x, y, 1]^T$



$$\mathbf{u} = \mathbf{K}\,\mathbf{x}_{cart} = \mathbf{K}\begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \end{bmatrix} = \begin{bmatrix} \mathbf{K} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \end{bmatrix}\mathbf{X}_{cam} = [\mathbf{K} \quad \mathbf{0}]\mathbf{X}_{cam} = \mathbf{K}[\mathrm{I} \quad 0]\mathbf{X}_{cam}$$

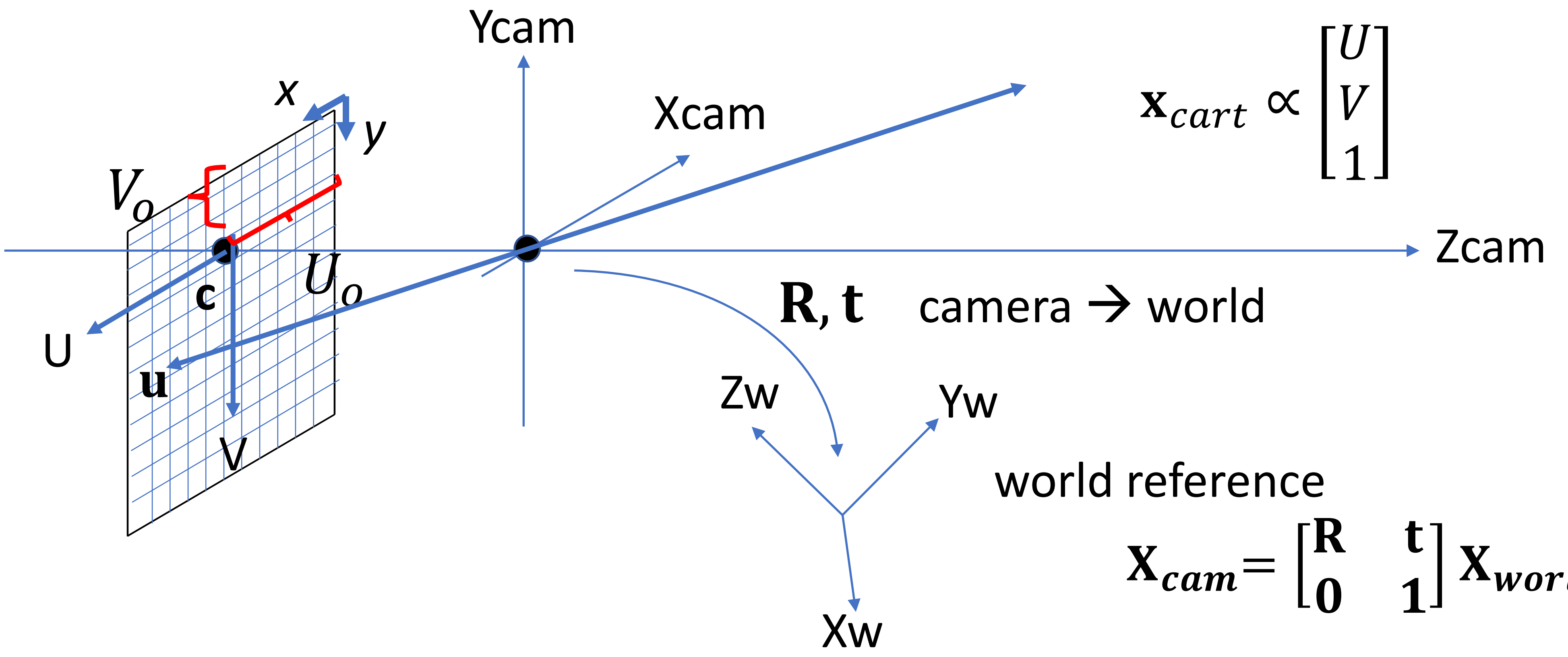
In general, world reference is ≠ camera reference



$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

$$\mathbf{u} = \mathbf{K}\,\mathbf{x}_{cart} = \mathbf{K}\begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \end{bmatrix} = \begin{bmatrix} \mathbf{K} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \end{bmatrix}\mathbf{X}_{cam} = [\mathbf{K} \quad \mathbf{0}]\mathbf{X}_{cam} = \mathbf{K}[\mathrm{I} \quad 0]\mathbf{X}_{cam}$$

In general, world reference is ≠ camera reference

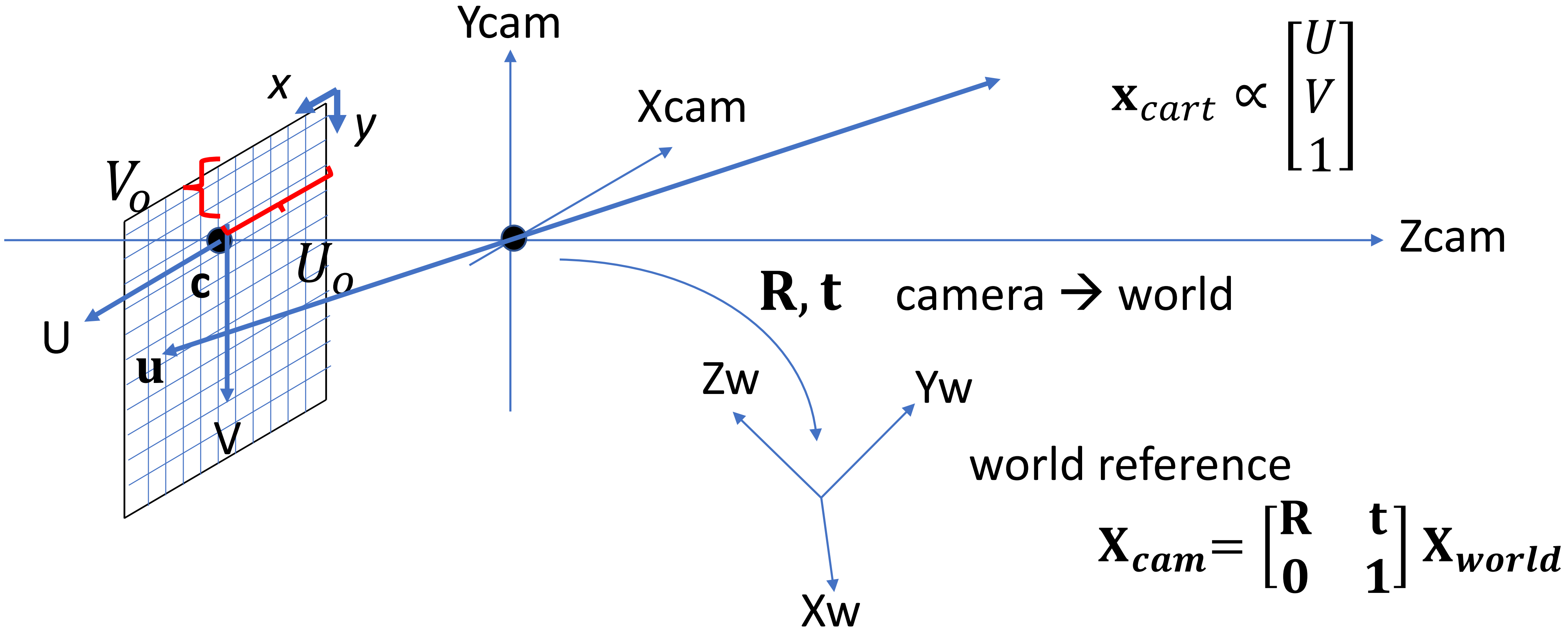


$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

$$\mathbf{X}_{cam} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \mathbf{X}_{world}$$

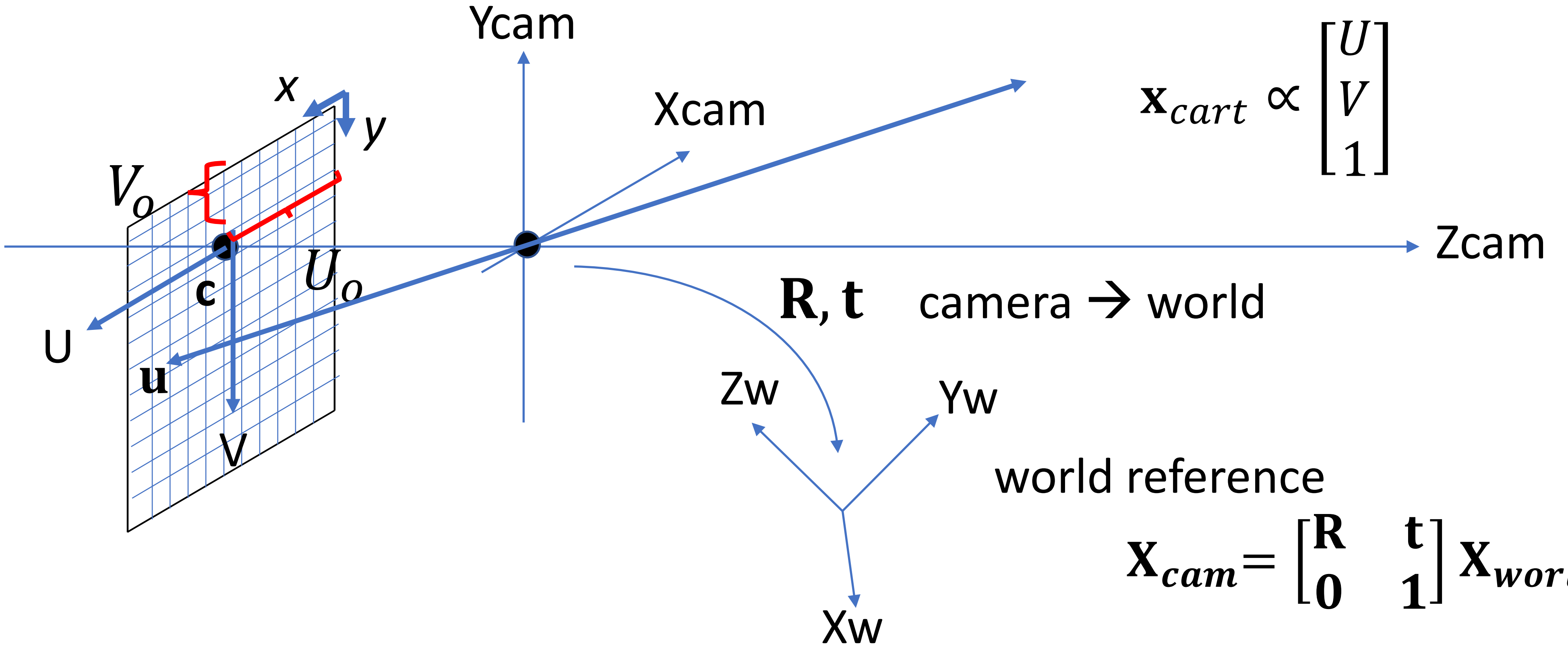
$$\mathbf{u} = \mathbf{K}\,\mathbf{x}_{cart} = \mathbf{K} \begin{bmatrix} \mathrm{X}_{cam} \\ \mathrm{Y}_{cam} \\ \mathrm{Z}_{cam} \end{bmatrix} = \begin{bmatrix} \mathbf{K} & \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \end{bmatrix} \mathbf{X}_{cam} = [\mathbf{K} \quad \mathbf{0}]\mathbf{X}_{cam} = \mathbf{K}[\mathrm{I} \quad 0]\mathbf{X}_{cam}$$

In general, world reference is ≠ camera reference



$$\mathbf{u} = \mathbf{K}[\mathrm{I} \quad 0]\mathbf{X}_{cam} = \mathbf{K}[\mathrm{I} \quad 0]\begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & \mathbf{1} \end{bmatrix}\mathbf{X}_{world}$$

In general, world reference is ≠ camera reference

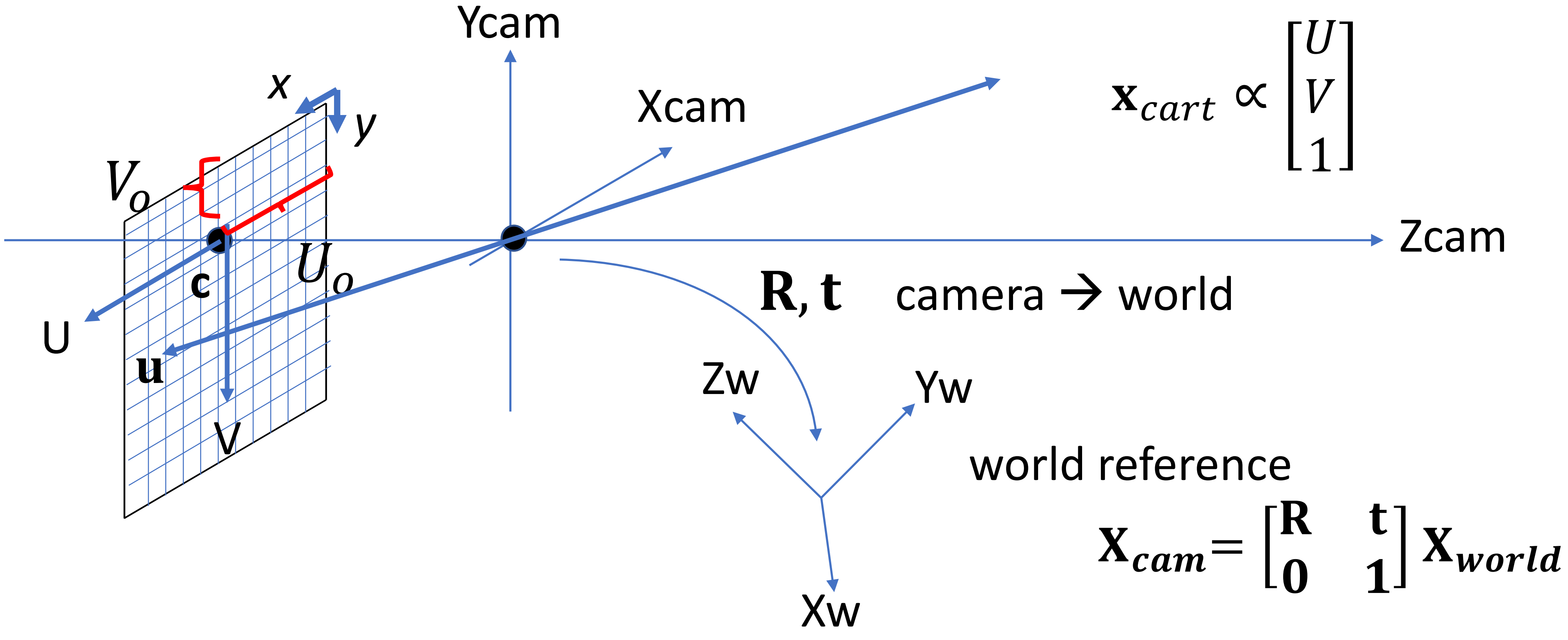


$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

$$\mathbf{X}_{cam} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \mathbf{X}_{world}$$

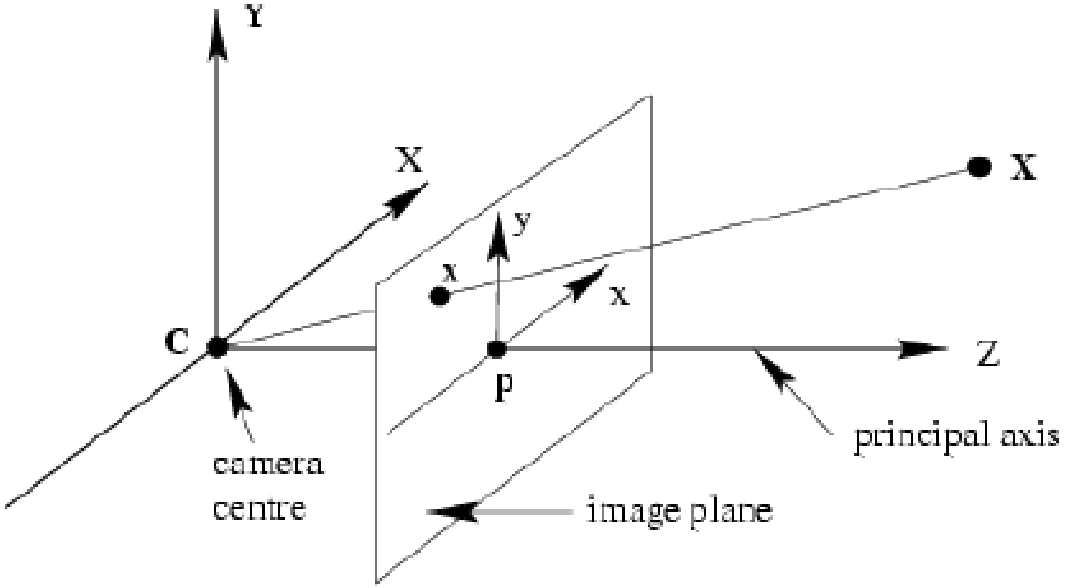
$$\mathbf{u} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\ \mathbf{X}_{world} = [\mathbf{KR} \quad \mathbf{Kt}]\ \mathbf{X}_{world}$$

In general, world reference is ≠ camera reference



$$\mathbf{u} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\ \mathbf{X}_{world} = [\mathbf{KR} \quad \mathbf{Kt}]\ \mathbf{X}_{world}$$

remember …



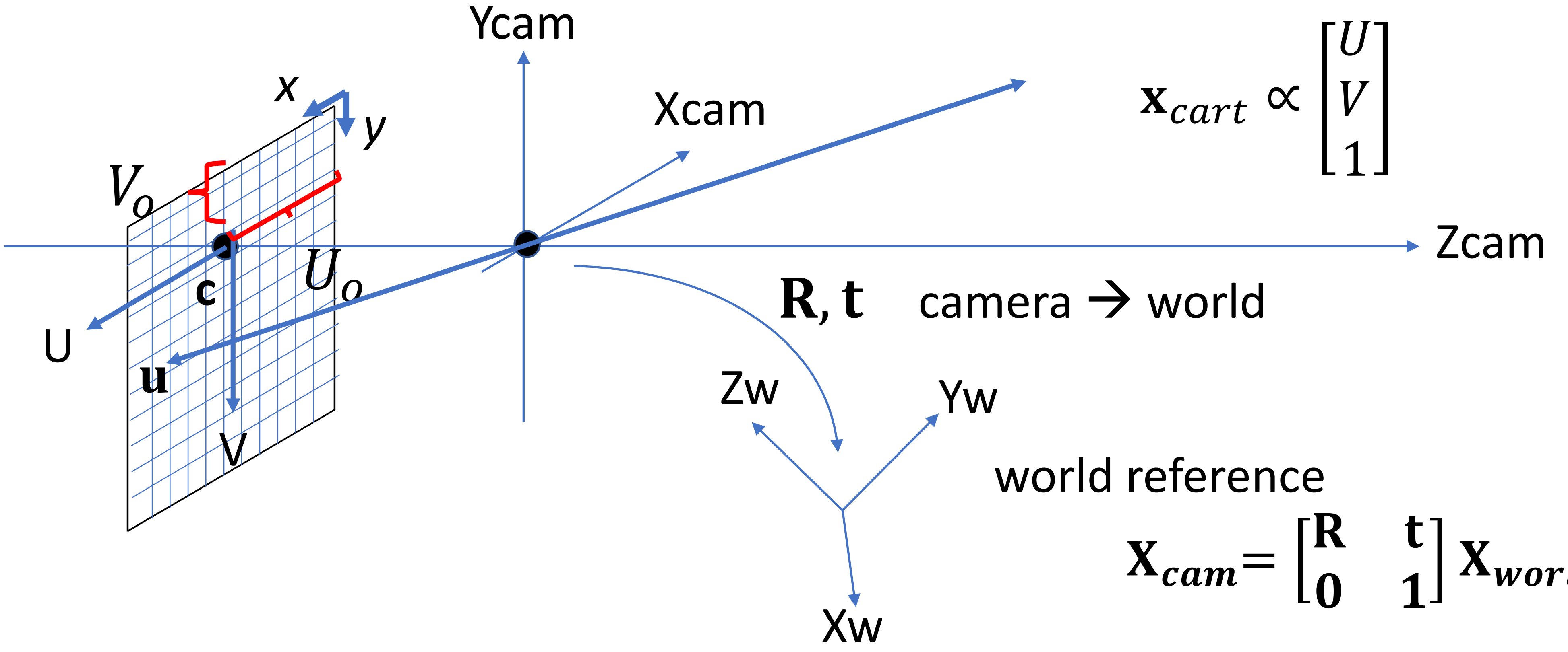
camera projection matrix

$$\mathbf{X} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \mathbf{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \mathbf{P}_{3\text{x}4} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \mathbf{P}_{3\text{x}4}\mathbf{X} = \left| \; \mathbf{M}_{3\text{x}3} \quad \mathbf{m}_{3\text{x}1} \; \right| \mathbf{X}$$

invertible

$$\mathbf{M} = \mathbf{K}\mathbf{R}_{cam->world}$$

In general, world reference is ≠ camera reference



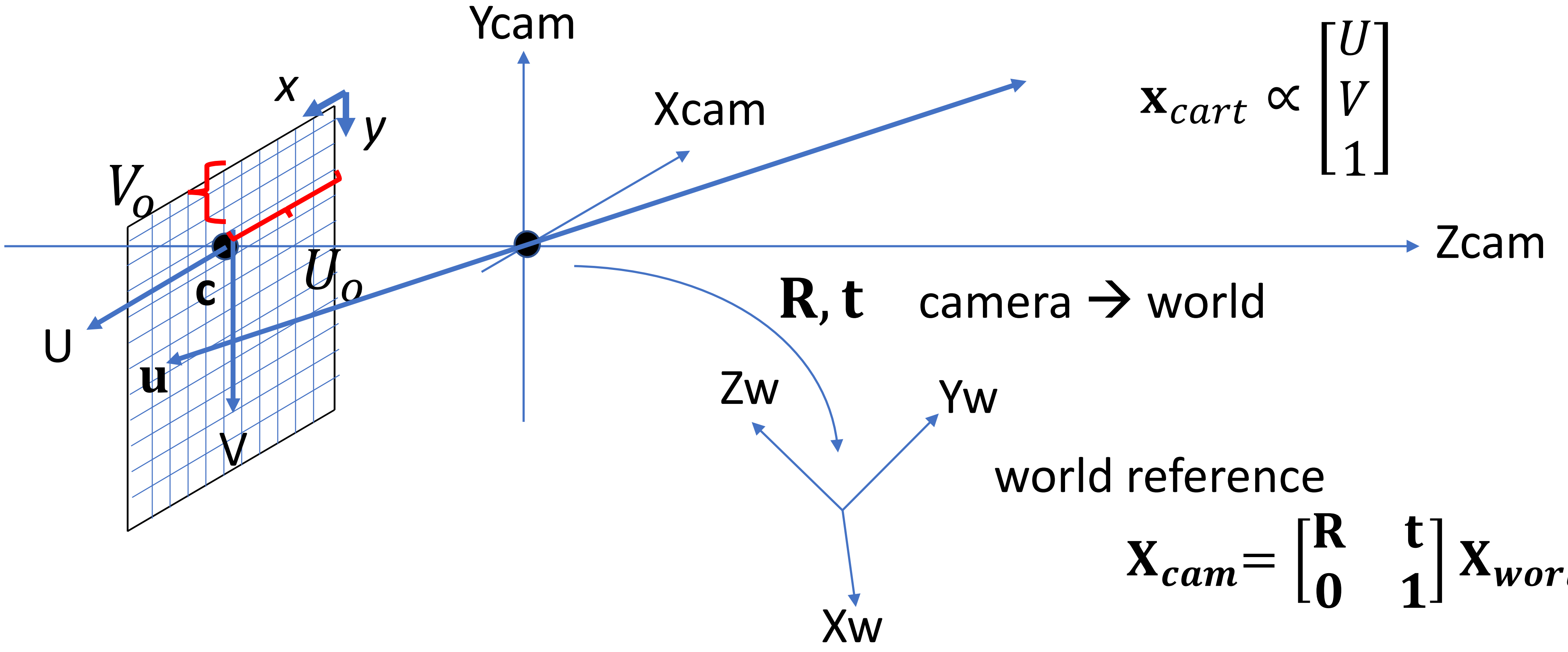
$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

$$\mathbf{X}_{cam} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \mathbf{X}_{world}$$

$$\mathbf{u} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\ \mathbf{X}_{world} = [\mathbf{KR} \quad \mathbf{Kt}]\ \mathbf{X}_{world}$$

remember $\mathbf{u} = \mathbf{P}\,\mathbf{X}_{world} = [\mathbf{M} \quad \boldsymbol{m}]\ \mathbf{X}_{world}$

In general, world reference is ≠ camera reference



$$\mathbf{x}_{cart} \propto \begin{bmatrix} U \\ V \\ 1 \end{bmatrix}$$

$$\mathbf{X}_{cam} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \mathbf{X}_{world}$$

$$\mathbf{u} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\ \mathbf{X}_{world} = [\mathbf{KR} \quad \mathbf{Kt}]\ \mathbf{X}_{world}$$

remember $\mathbf{u} = \ \mathbf{P}\,\mathbf{X}_{world} \qquad = \ [\mathbf{M} \quad \mathbf{m}]\,\mathbf{X}_{world}$

→ $\boxed{\mathbf{M} = \mathbf{KR}}$ and $\boxed{\mathbf{m} = \mathbf{Kt}}$

$$\mathbf{u} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\ \mathbf{X}_{world} = [\mathbf{KR} \quad \mathbf{Kt}]\ \mathbf{X}_{world}$$

but
$$\mathbf{u} = \mathbf{P}\,\mathbf{X}_{world} = [\mathbf{M} \quad \mathbf{m}]\ \mathbf{X}_{world}$$

$$\rightarrow \mathbf{M} = \mathbf{KR} \text{ and } \mathbf{m} = \mathbf{Kt}$$

$$\mathbf{M} = \mathbf{KR}_{cam->world}$$

given **M**, find **K** and **R** by Q-R matrix decomposition of inv(**M**)

$$\mathbf{u} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\ \mathbf{X}_{world} = [\mathbf{KR} \quad \mathbf{Kt}]\ \mathbf{X}_{world}$$

but

$$\mathbf{u} = \mathbf{P}\,\mathbf{X}_{world} = [\mathbf{M} \quad \mathbf{m}]\ \mathbf{X}_{world}$$

$$\rightarrow \mathbf{M} = \mathbf{KR} \quad \text{and} \quad \mathbf{m} = \mathbf{Kt}$$

$$\mathbf{M} = \mathbf{KR}_{cam->world}$$

given **M**, find **K** and **R** by Q-R matrix decomposition of inv(**M**)

from $\mathbf{m} = -\mathbf{Mo} \rightarrow \mathbf{u} = [\mathbf{M} \quad \mathbf{m}]\ \mathbf{X}_{world} = \mathbf{M}[\mathbf{I} \quad -\boldsymbol{o}]\ \mathbf{X}_{world}$

$$\boxed{\mathbf{u} = \mathbf{KR}[\mathbf{I} \quad -\boldsymbol{o}]\ \mathbf{X}_{world}}$$

camera calibration

Intrinsic camera **calibration**:
estimation of matrix $\mathbf{K}$
- focal distance f_X
- focal distance f_Y
- principal point (U_o, V_o)
- skew factor (in old cameras)

aspect ratio $a = f_X / f_Y$:
ratio between pixel width and height

intrinsic camera parameters don’t vary
under camera displacement

Extrinsic camera **calibration**:
estimation of matrix $\mathbf{R}$ and vector $\boldsymbol{t}$
- camera rotation $\mathbf{R}$
- camera translation $\boldsymbol{t}$

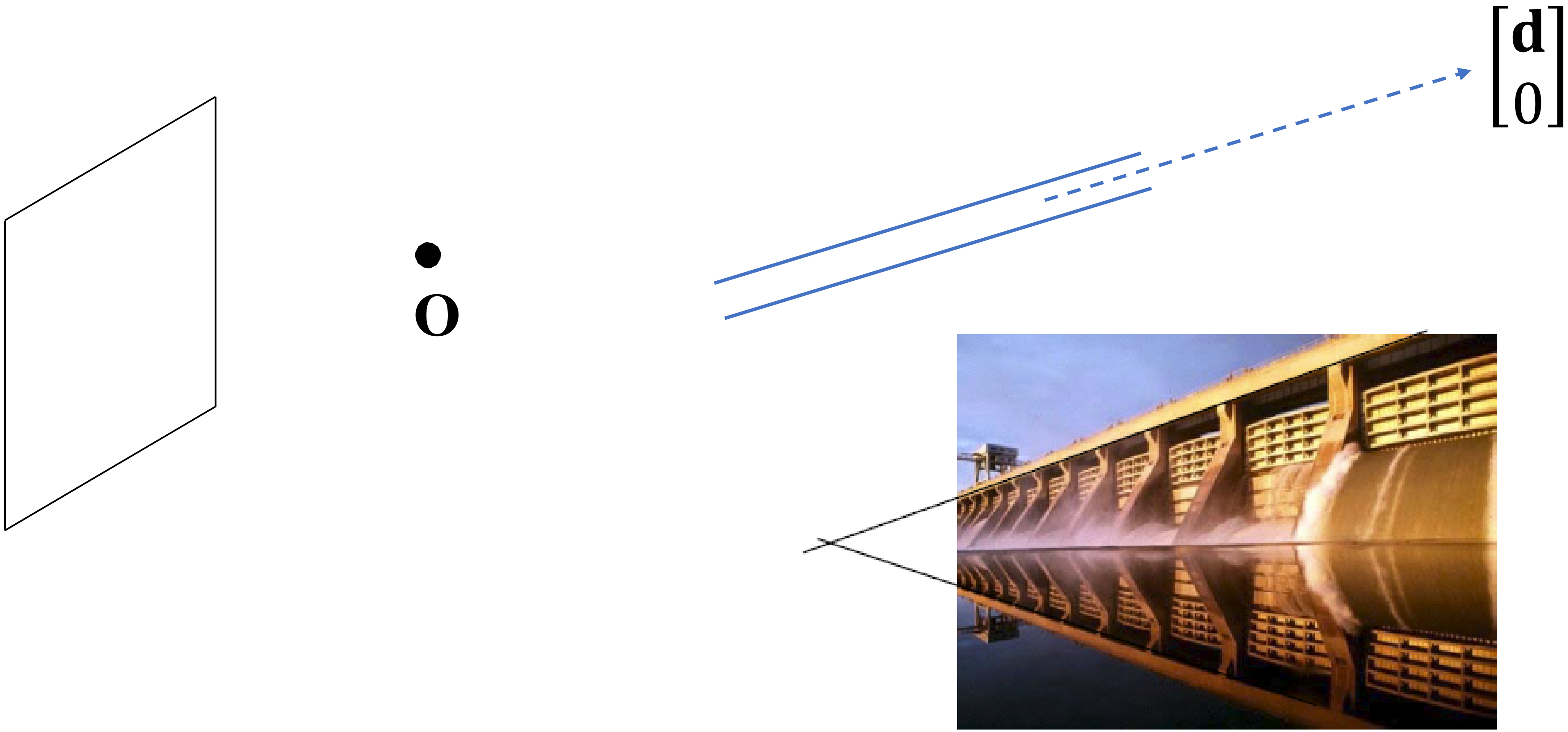
extrinsic camera parameters vary under camera displacement

$$x = x_o + (x_o - c_x)(K_1 r^2 + K_2 r^4 + \ldots)$$
$$y = y_o + (y_o - c_y)(K_1 r^2 + K_2 r^4 + \ldots)$$

$$r = (x_o - c_x)^2 + (y_o - c_y)^2 \quad .$$

vanishing point of a direction

parallel lines concur at the point at the infinity of their common direction $\mathbf{d}$



vanishing points

V image of the point at the ∞ along direction **d**

$$\mathbf{u}_V = \left| \begin{array}{c:c} \mathbf{M} & \mathbf{m} \end{array} \right| \cdot \left| \begin{array}{c} \mathbf{d} \\ \hdashline 0 \end{array} \right| = \mathbf{M} \cdot \mathbf{d}$$

vanishing points

V image of the point at the ∞ along direction **d**

$$\mathbf{u}_V = \left[\, \mathbf{M} \;\vdots\; \mathbf{m} \,\right] \cdot \left[\begin{array}{c} \mathbf{d} \\ \hdashline 0 \end{array} \right] = \mathbf{M} \cdot \mathbf{d}$$

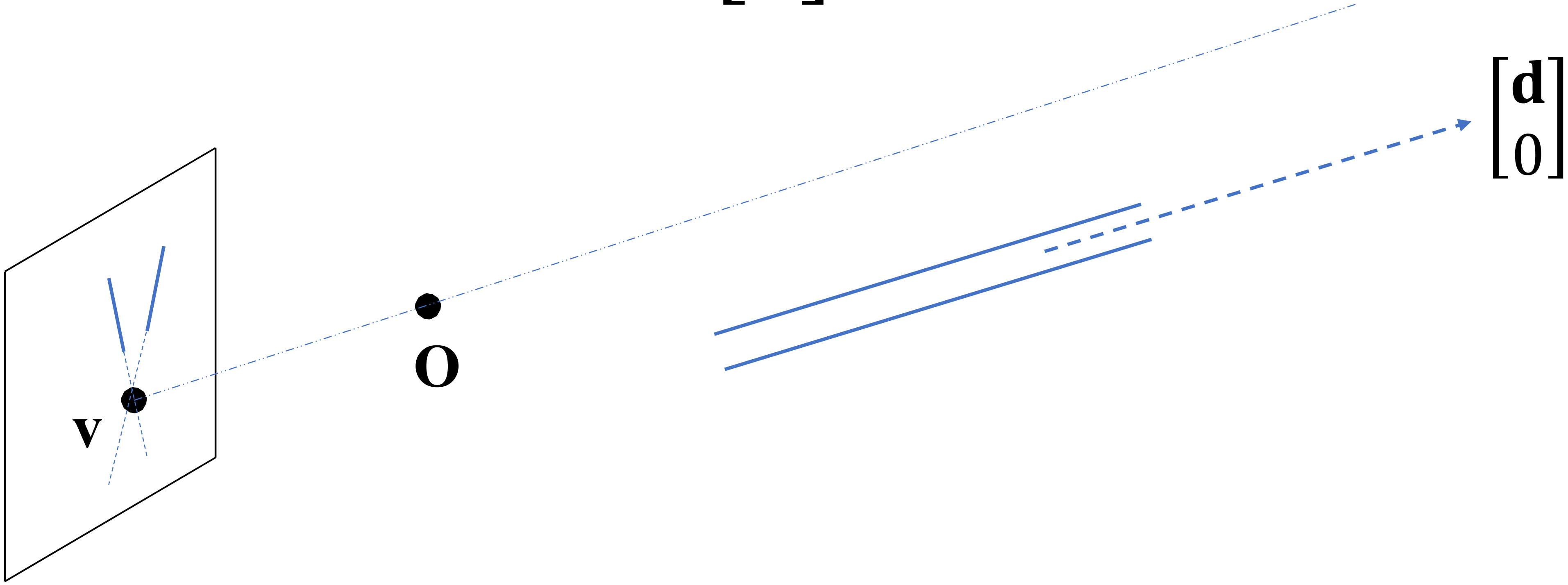
Remember: the direction of the backprojection of image point $\mathbf{V}$ (viewing ray associated to $\mathbf{V}$) is $\mathbf{M}^{-1}\mathbf{u}_V = \mathbf{M}^{-1}\,\mathbf{Md} = \mathbf{d}$

→

Vanishing Point Theorem:

The viewing ray associated to the vanishing point $\mathbf{V}$ of a direction $\mathbf{d}$ is parallel to $\mathbf{d}$

The backprojection (viewing ray) of vanishing point $\mathbf{v}$ goes through point at the infinity $\begin{bmatrix}\mathbf{d}\\0\end{bmatrix}$ (since $\mathbf{v}$ is its image)



→ the viewing ray of $\mathbf{v}$ is parallel to the direction $\mathbf{d}$

mobile robot navigation

vehicle driving

If $\mathbf{M}$ is known, find $\mathbf{V}$ and follow the direction of the corridor:

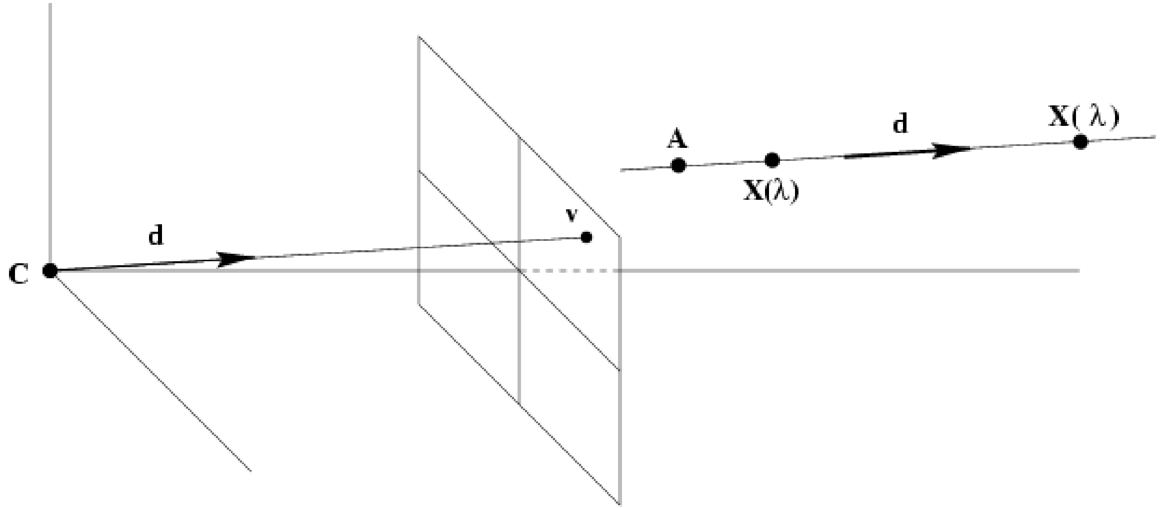
$$\mathbf{d} = \mathbf{M}^{-1}\mathbf{V}$$

Vanishing points (d measured wrt camera reference)

$$x(\lambda) = PX(\lambda) = PA + \lambda PD = a + \lambda Kd$$

$$v = \lim_{\lambda \to \infty} x(\lambda) = \lim_{\lambda \to \infty} (a + \lambda Kd) = Kd$$

$$v = PX_{\infty} = Kd$$



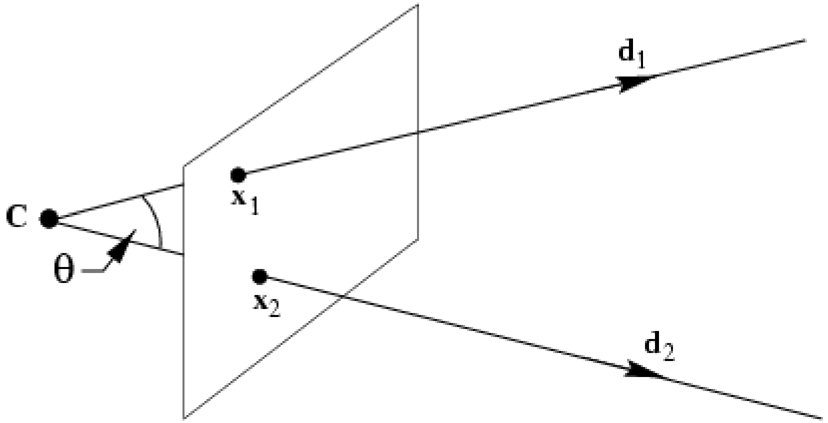
Vanishing points (d measured wrt world reference)

$$v = PX_{\infty} = KRd_w$$

What does INTRINSIC calibration give?

$$x = K[I \mid 0]\begin{bmatrix} d \\ 0 \end{bmatrix}$$

$$d = K^{-1}x$$



$$\cos\theta = \frac{d_1^{T} d_2}{\sqrt{(d_1^{T} d_1)(d_2^{T} d_2)}} = \frac{x_1^{T}(K^{-T}K^{-1})x_2}{\sqrt{(x_1^{T}(K^{-T}K^{-1})x_1)(x_2^{T}(K^{-T}K^{-1})x_2)}}$$

An image l defines a plane through the camera center with normal n=K^Tl measured in the camera's Euclidean frame

→ Relative position of viewing rays associated to different image points

The image ω of the absolute conic Ω_∞ is **independent** of $\mathbf{R}$ and **t**: it only depends on intrinsic parameters

$$x = PX_\infty = KR[I \mid -\widetilde{C}]\begin{pmatrix} d \\ 0 \end{pmatrix} = KRd$$

mapping between π_∞ to an image is given by the planar homogaphy u$= \boldsymbol{Hd}$, with $\boldsymbol{H} = \mathbf{M} = \mathbf{K}\,\mathbf{R}$

image of the absolute conic (IAC) $\boldsymbol{\omega} = \mathrm{H}^{-\mathrm{T}}\Omega_\infty \mathrm{H}^{-1} = \mathrm{K}^{-\mathrm{T}}\mathrm{R}\mathrm{I}_3\mathrm{R}^{-1}\mathrm{K}^{-1}$

$$\omega = \left(\mathrm{K}\mathrm{K}^{\mathrm{T}}\right)^{-1} = \mathrm{K}^{-\mathrm{T}}\mathrm{K}^{-1} \qquad \left(\mathrm{C} \mapsto \mathrm{H}^{-\mathrm{T}}\mathrm{C}\mathrm{H}^{-1}\right)$$

(i) IAC depends only on intrinsics

(ii) angle between two rays

(iii) DIAC=ω^*=KK^T

(iv) $\omega \Leftrightarrow$ K (cholesky factorisation)

(v) IAC contains image of circular points

$$\cos\theta = \frac{x_1^{\mathrm{T}}\omega x_2}{\sqrt{\left(x_1^{\mathrm{T}}\omega x_1\right)\left(x_2^{\mathrm{T}}\omega x_2\right)}}$$

summary

homography from $\boldsymbol{\pi}_{\boldsymbol{\infty}}$ to image plane: $\mathbf{H} = \mathbf{M} = \mathbf{K}\,\mathbf{R}$

→ vanishing points = image of points at the ∞: $\mathbf{v}_d = \mathbf{H}\,\mathbf{d} = \mathbf{KR}\,\mathbf{d}$

image of the absolute conic $\boldsymbol{\omega} = \mathbf{H}^{-\mathrm{T}}\Omega_\infty\mathbf{H}^{-1} = \mathbf{K}^{-\mathrm{T}}\mathbf{R}^{-\mathrm{T}}\mathrm{I}_3\mathbf{R}^{-1}\mathbf{K}^{-1} = \mathbf{K}^{-\mathrm{T}}\mathbf{K}^{-1}$

→ IAC $\boldsymbol{\omega}$ independent of $\mathbf{R}$ and t; it only depends on intrinsic $\mathbf{K}$

from IAC $\boldsymbol{\omega}$ to calibration matrix $\mathbf{K}$: Cholesky factorisation of DIAC $\omega^{-1} = \mathbf{KK}^{\mathrm{T}}$

absolute conic $\Omega_\infty = \bigcup_\pi \{\mathbf{I}_\pi, \mathbf{J}_\pi\}$ made of circular points

→ IAC $\boldsymbol{\omega} = \bigcup_\pi \{\mathbf{I}'_\pi, \mathbf{J}'_\pi\}$ made of imaged circular points $\mathbf{I}'_\pi = \mathbf{HI}_\pi$, $\mathbf{J}'_\pi = \mathbf{HJ}_\pi$

constraint on $\boldsymbol{\omega}$ from known angle btw. two directions, and their vanishing points

$$\cos\theta = \frac{\mathbf{v_1}^{\mathrm{T}}\omega\mathbf{v_2}}{\sqrt{(\mathbf{v_1}^{\mathrm{T}}\omega\mathbf{v_1})(\mathbf{v_2}^{\mathrm{T}}\omega\mathbf{v_2})}}$$

Scenario n. 1

Known: vanishing points and angle θ between directions

$$\cos\theta = \frac{\mathbf{d_1}^{\mathrm{T}}\mathbf{d_2}}{\sqrt{\left(\mathbf{d_1}^{\mathrm{T}}\mathbf{d_1}\right)\left(\mathbf{d_2}^{\mathrm{T}}\mathbf{d_2}\right)}} = \frac{\mathbf{v_1}^{\mathrm{T}}\omega\mathbf{v_2}}{\sqrt{(\mathbf{v_1}^{\mathrm{T}}\omega\mathbf{v_1})(\mathbf{v_2}^{\mathrm{T}}\omega\mathbf{v_2})}}$$

$\rightarrow$

constraint on the IAC ω

(linear if directions are orthogonal)

Scenario n. 2

Known: vanishing points and IAC ω

$$\cos\,\theta = \frac{\mathbf{d_1}^{\mathrm{T}}\mathbf{d_2}}{\sqrt{\left(\mathbf{d_1}^{\mathrm{T}}\mathbf{d_1}\right)\left(\mathbf{d_2}^{\mathrm{T}}\mathbf{d_2}\right)}} = \frac{\mathbf{v_1}^{\mathrm{T}}\omega\mathbf{v_2}}{\sqrt{(\mathbf{v_1}^{\mathrm{T}}\omega\mathbf{v_1})(\mathbf{v_2}^{\mathrm{T}}\omega\mathbf{v_2})}}$$

→

compute angle θ between directions

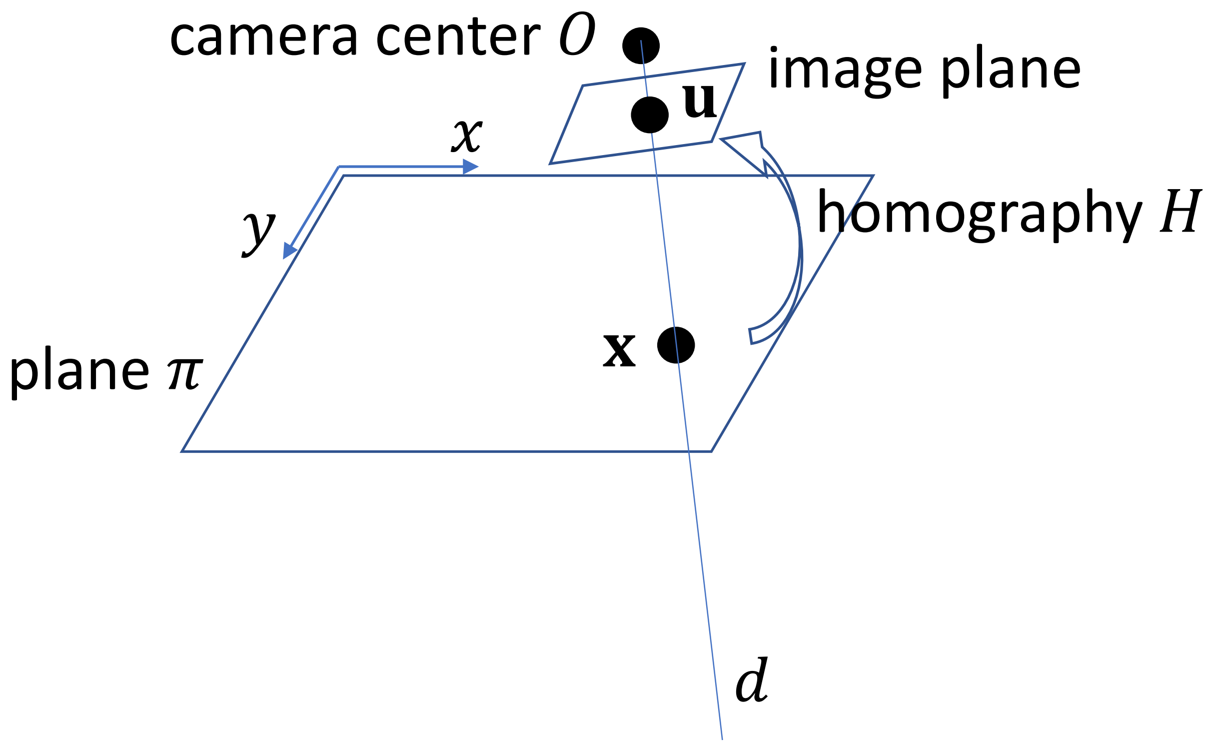
→ reconstruction of the shape

Camera calibration
from a single image of a known planar shape
and known camera center position

Calibration from known plane π and camera center O

Let us refer the coordinates to a reference attached to a knwon plane π : a generic point on this plane has homogeneous coordinates

$\mathbf{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$, while $O = \begin{bmatrix} \mathrm{x}_O \\ \mathrm{y}_O \\ \mathrm{z}_O \end{bmatrix}$ are the known cartesian coordinates of the camera center,



Calibration from known plane π and camera center O

1) Estimate homography H from known points on the plane π and their images

2) Call M_o the matrix relating any point $\mathbf{x}$ on π to the direction $\mathbf{d}$ of a ray from O to $\mathbf{x}$:

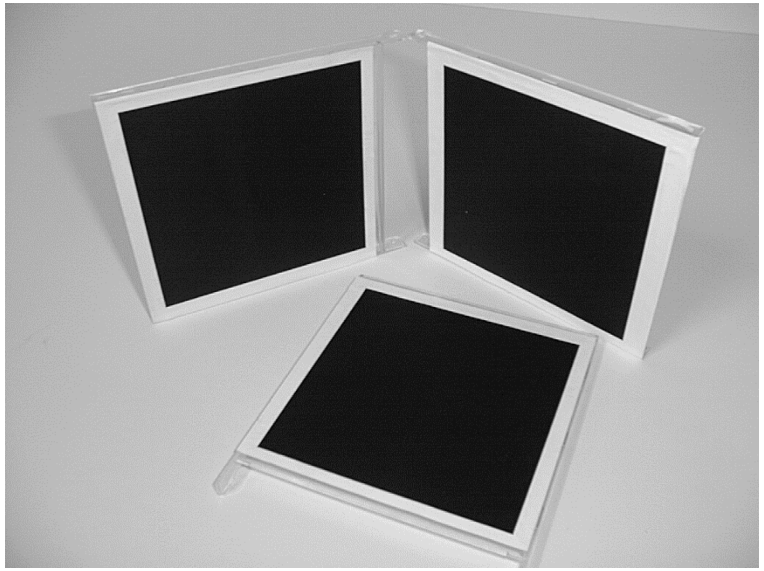
$$\mathbf{d} = \begin{bmatrix} x - \mathrm{x}_o \\ y - \mathrm{y}_o \\ -\mathrm{z}_o \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\mathrm{x}_o \\ 0 & 1 & -\mathrm{y}_o \\ 0 & 0 & -\mathrm{z}_o \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \rightarrow {M_o}^{-1} = \begin{bmatrix} 1 & 0 & -\mathrm{x}_o \\ 0 & 1 & -\mathrm{y}_o \\ 0 & 0 & -\mathrm{z}_o \end{bmatrix}$$

3) Compute the matrix M relating any image point $\mathbf{u}$ to the direction $\mathbf{d}$ of its viewing ray from $\mathbf{u} = H\mathbf{x}$, is $\mathbf{d} = {M_o}^{-1}\mathbf{x} = {M_o}^{-1}H^{-1}\mathbf{u} \rightarrow M^{-1} = {M_o}^{-1}H^{-1}$

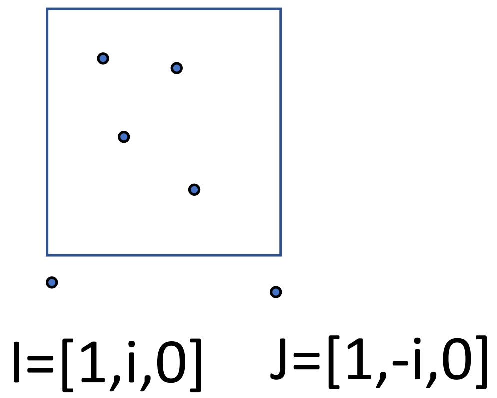
4) Q-R decompose matrix M^{-1} as $M^{-1} = R^{-1}K^{-1}$, where K is the camera intrinsic calibration matrix and R^{-1} is the rotation matrix from the world (reference attached to π) to the camera

Camera calibration
from images of known planar shapes
(Zhang method)

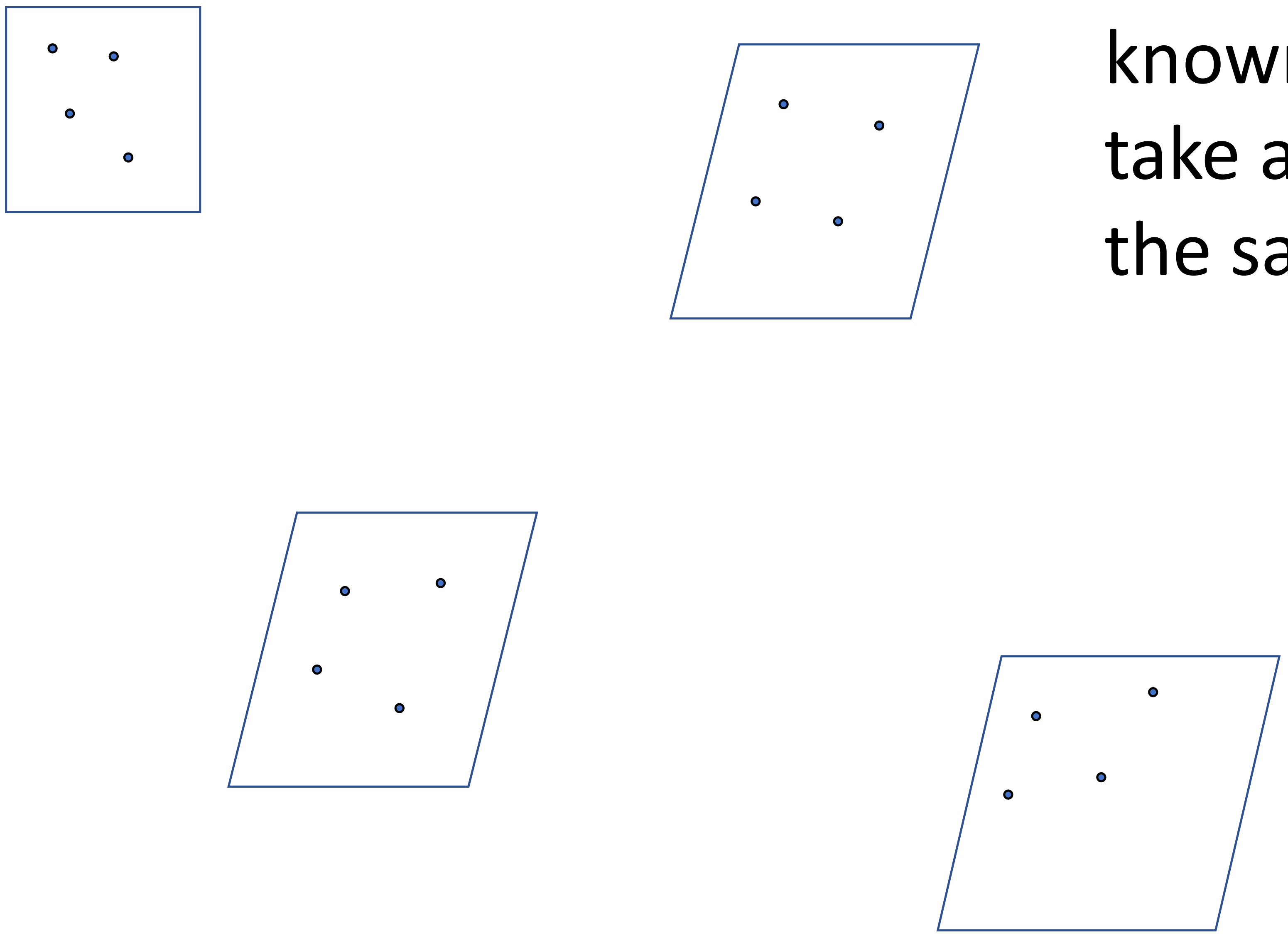
A simple calibration device (Zhang 2000)



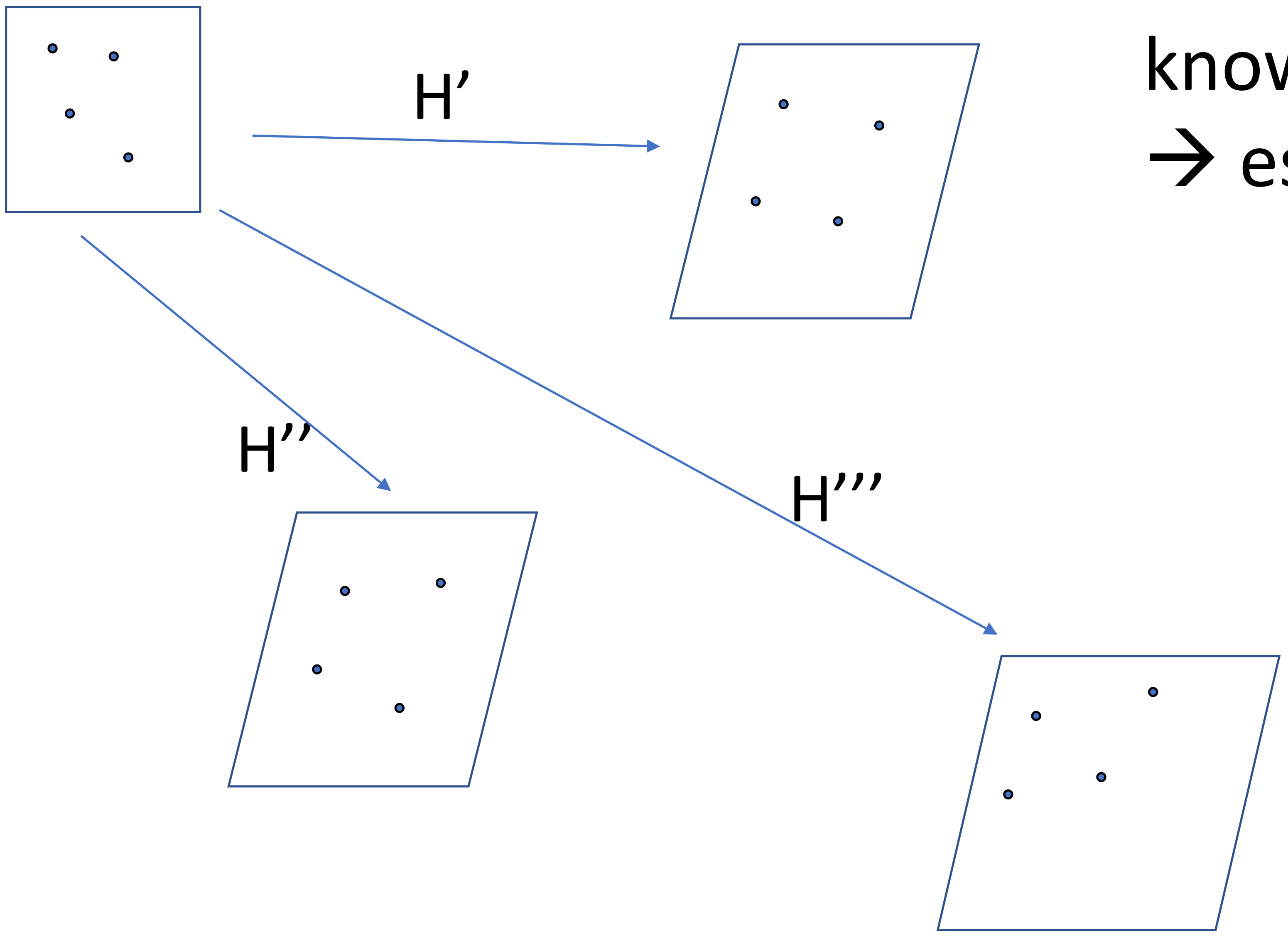
(i) compute H for each square (corners → (0,0),(1,0),(0,1),(1,1))
(ii) compute the imaged circular points $H(1,\pm i,0)^T$
(iii) fit a conic to 6 imaged circular points
(iv) compute K from ω through Cholesky factorization



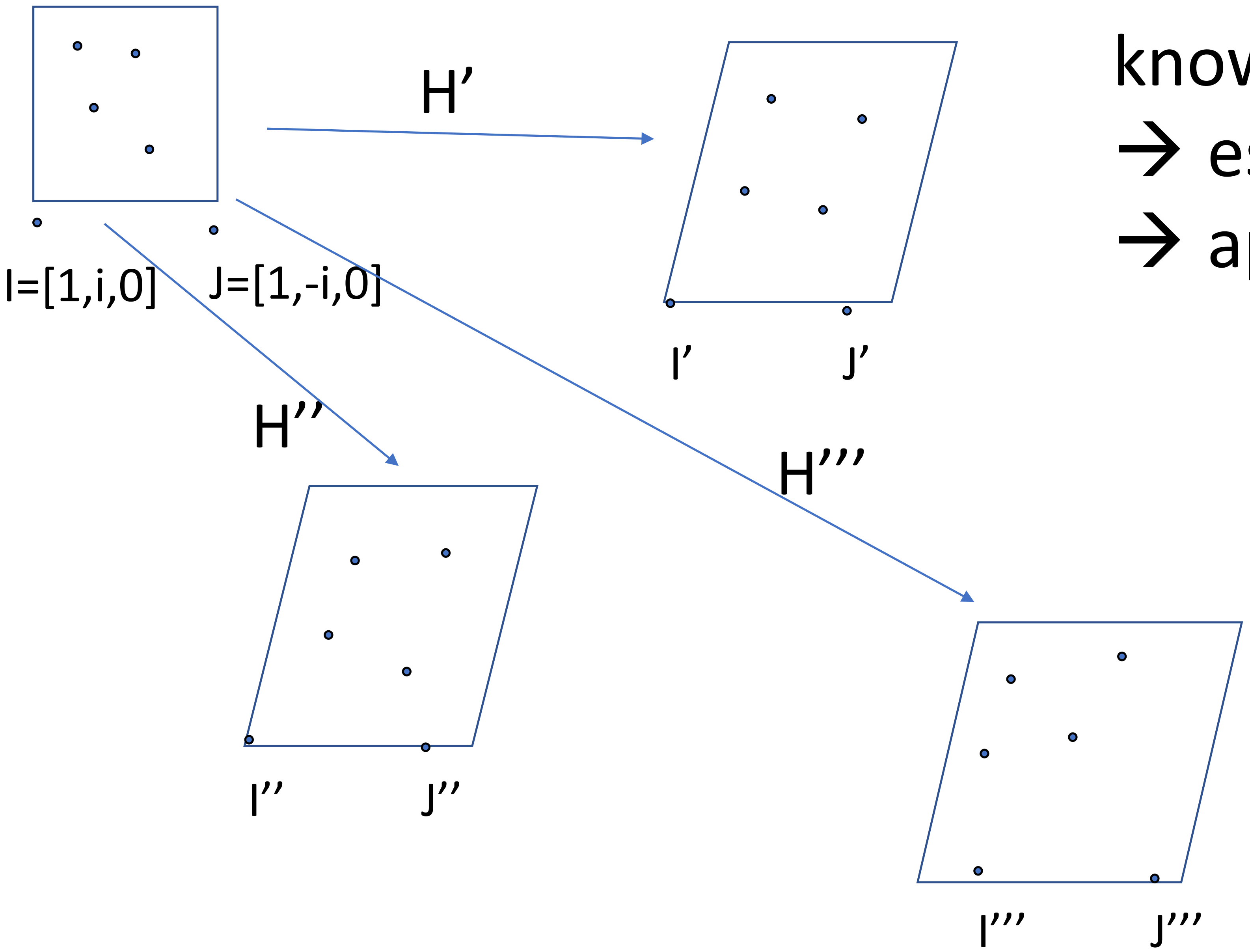
known planar scene



known planar scene:
take at least 3 images with
the same camera (constant $\mathbf{K}$)



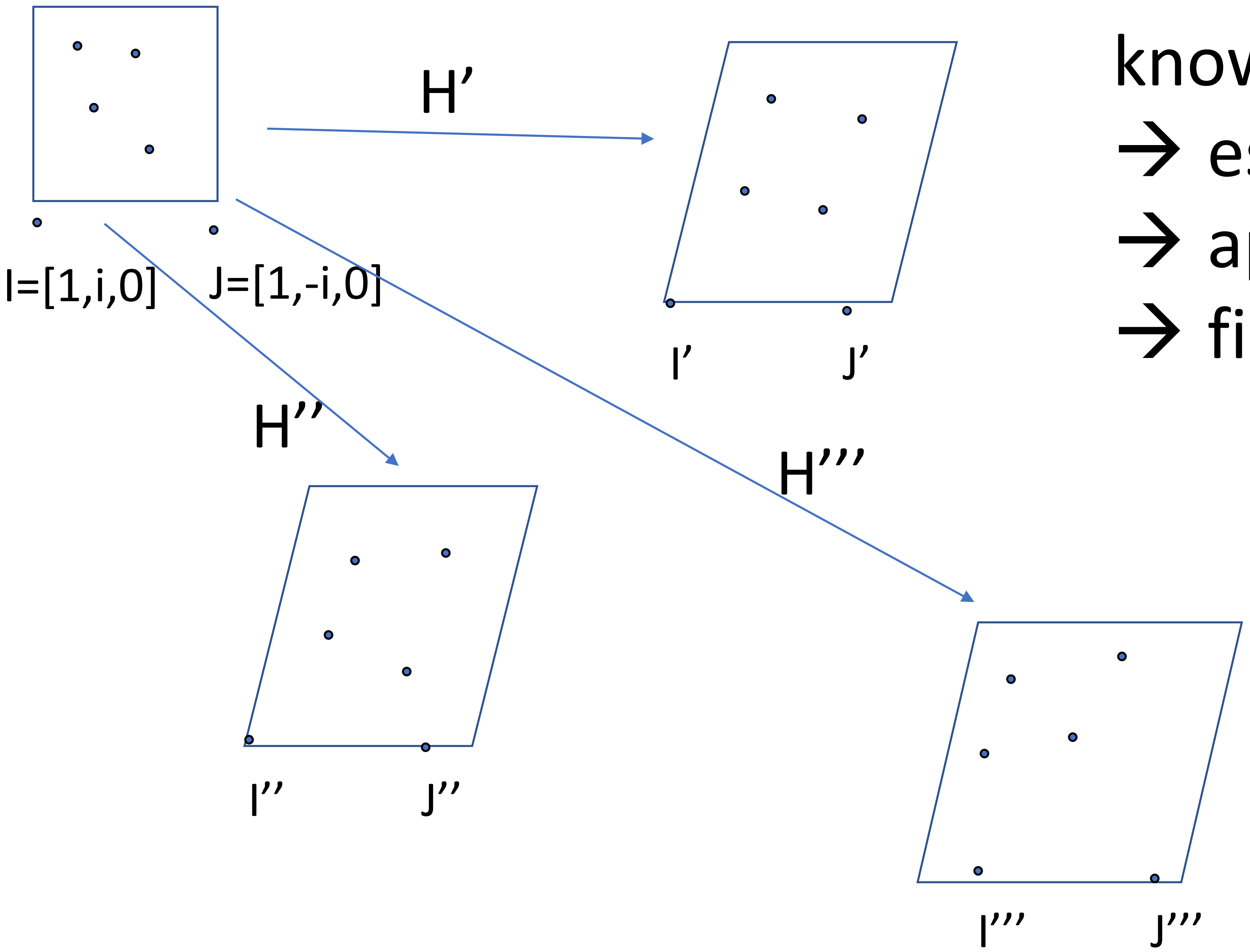
known planar scene
→ estimate homographies



known planar scene

→ estimate homographies

→ apply them to $\mathbf{I}$ and $\mathbf{J}$

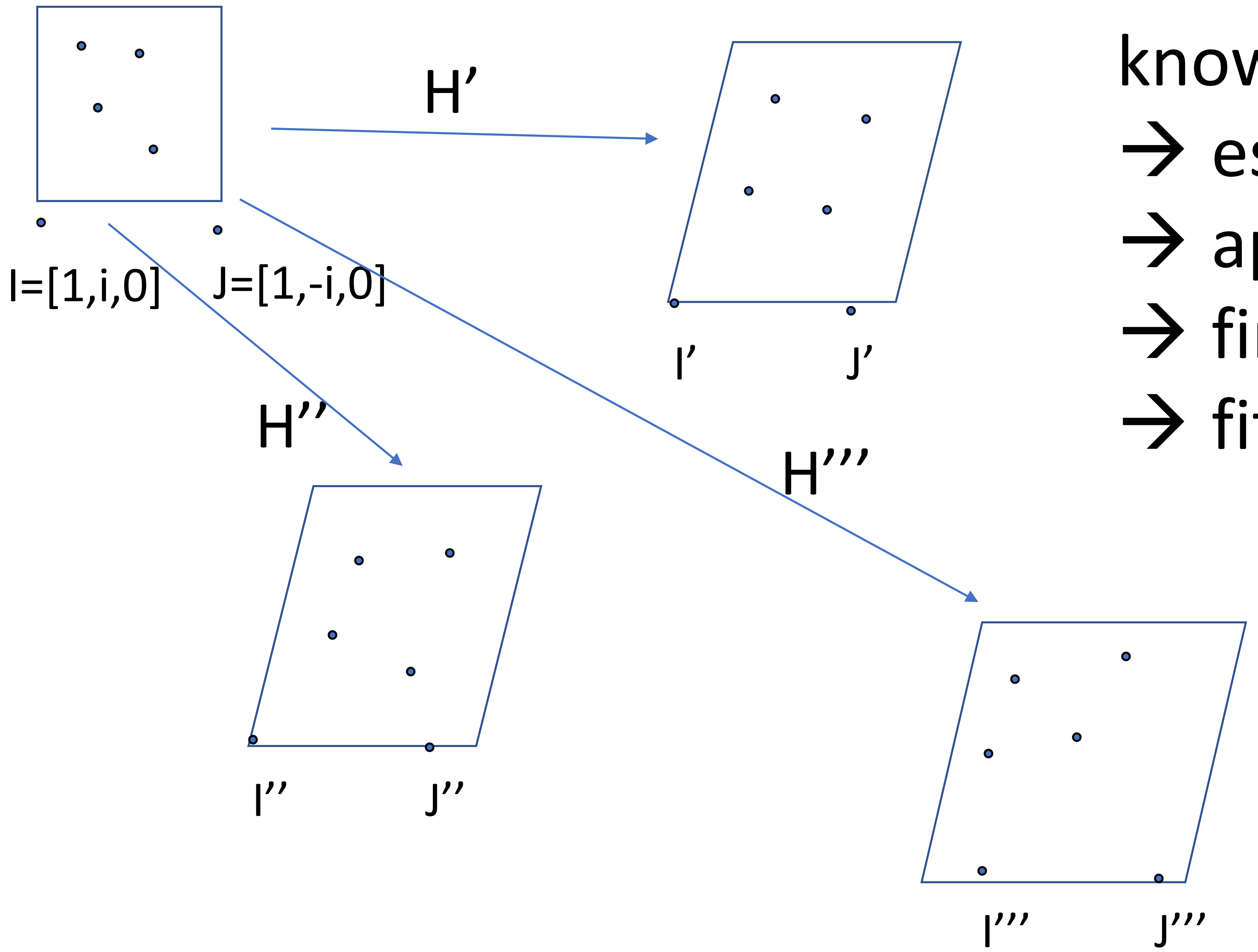


known planar scene

→ estimate homographies

→ apply them to **I** and **J**

→ find **I'**, **J'**, **I''**, **J''**, **I'''**, **J'''**



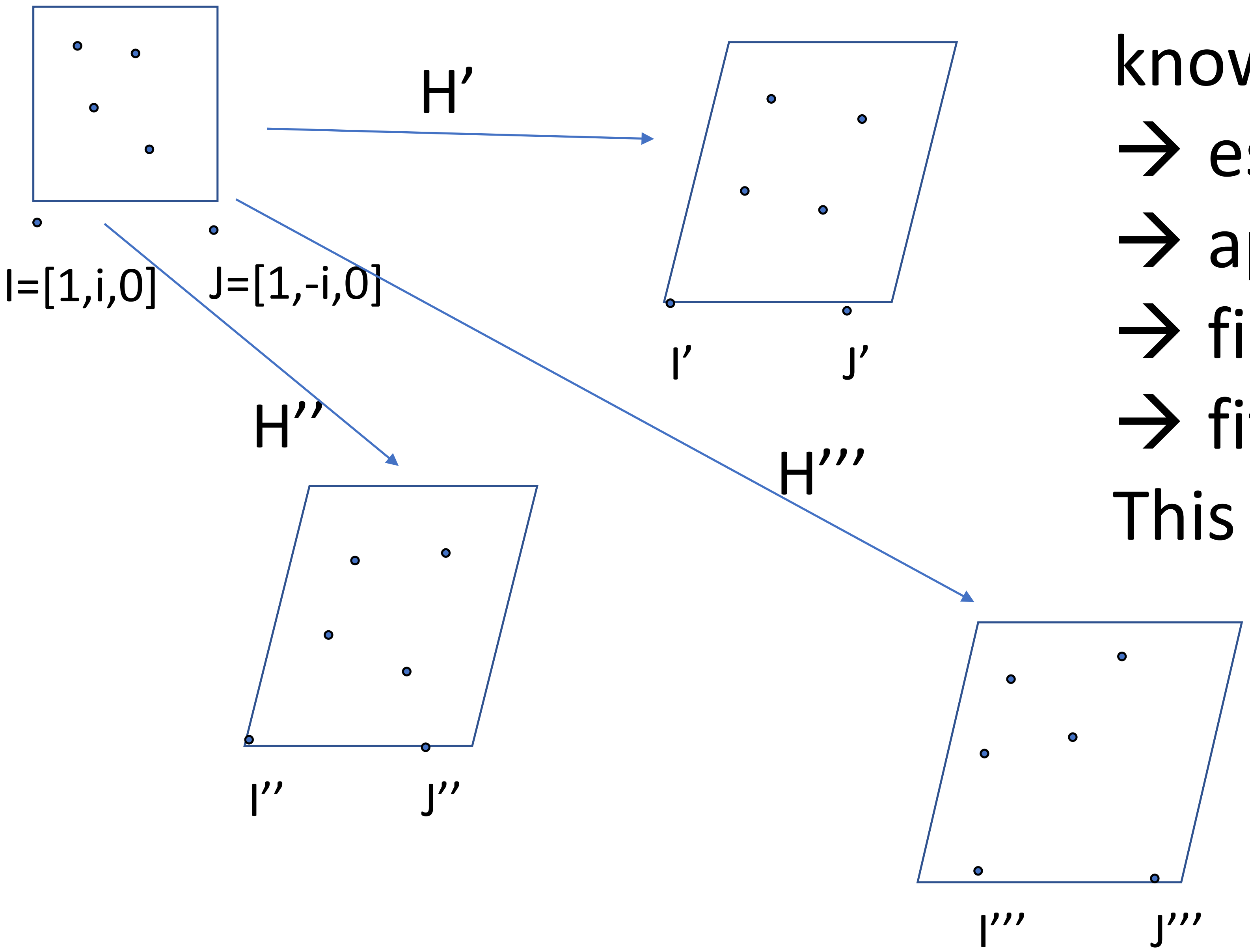
known planar scene

→ estimate homographies

→ apply them to $\mathbf{I}$ and $\mathbf{J}$

→ find $\mathbf{I}'$, $\mathbf{J}'$, $\mathbf{I}''$, $\mathbf{J}''$, $\mathbf{I}'''$, $\mathbf{J}'''$

→ fit a conic ω to them:



known planar scene

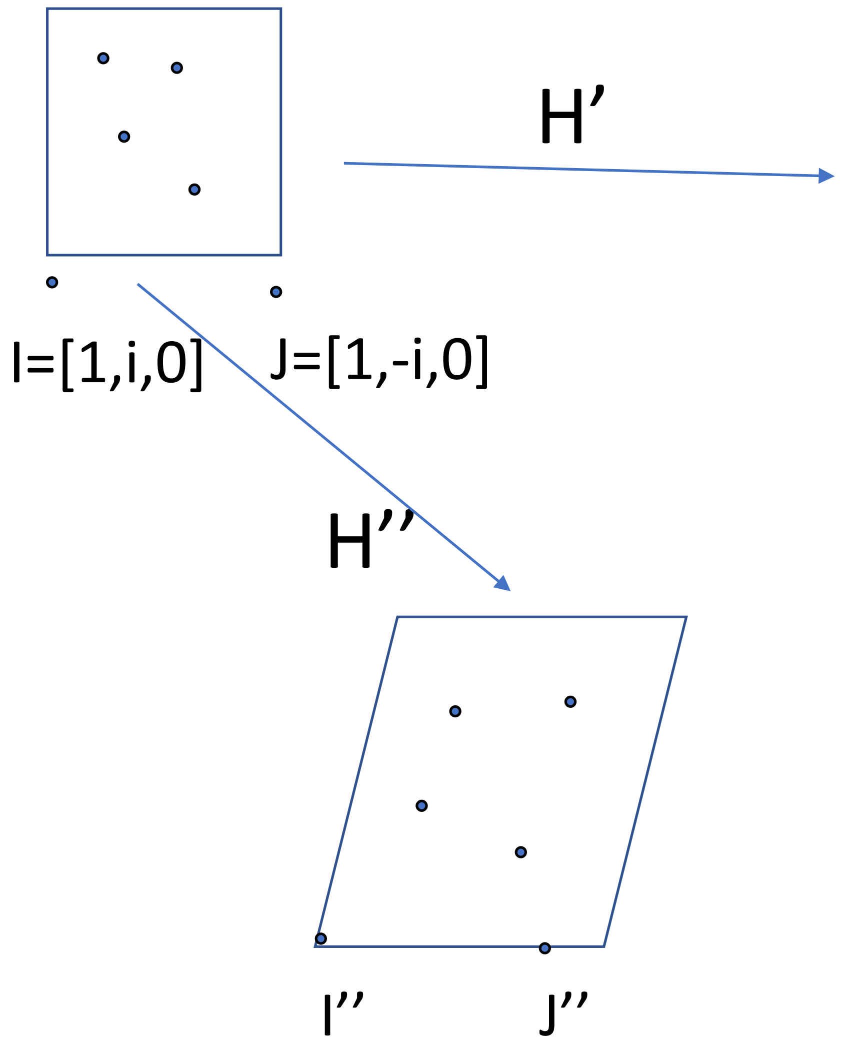
→ estimate homographies

→ apply them to $\mathbf{I}$ and $\mathbf{J}$

→ find $\mathbf{I}'$, $\mathbf{J}'$, $\mathbf{I}''$, $\mathbf{J}''$, $\mathbf{I}'''$, $\mathbf{J}'''$

→ fit a conic to them:

This conic is the IAC ω !!



known planar scene

→ estimate homographies

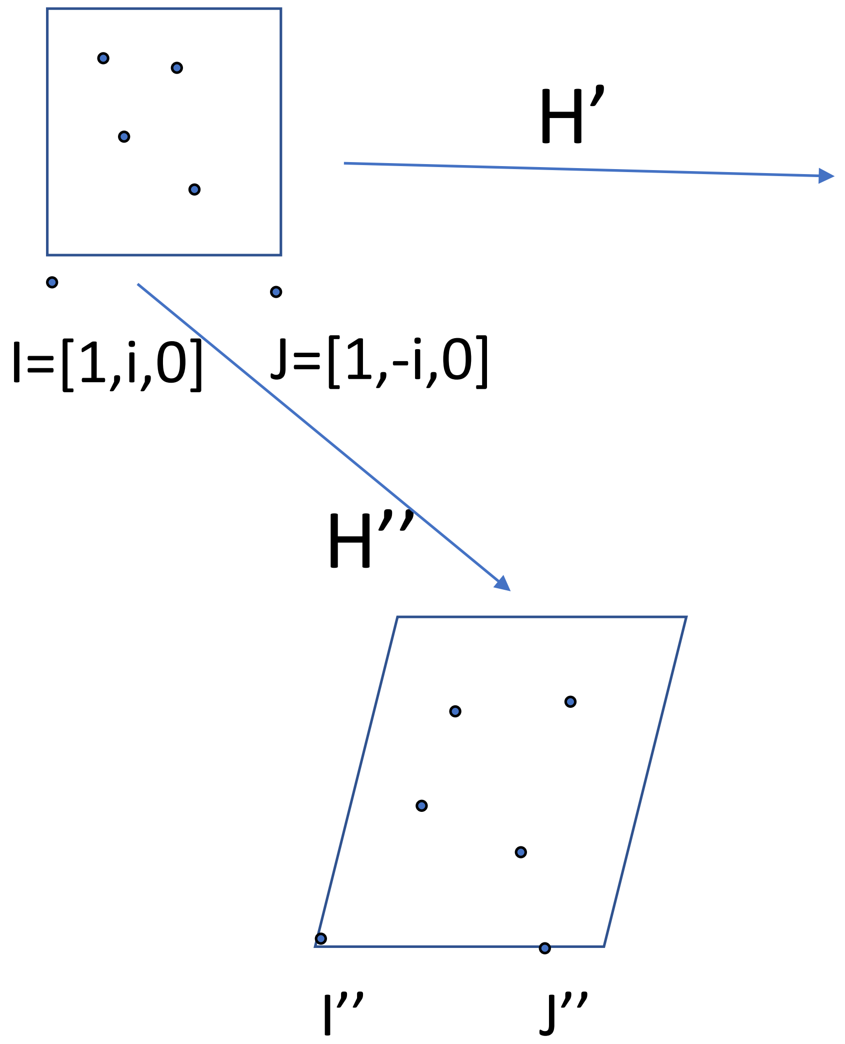
→ apply them to $\mathbf{I}$ and $\mathbf{J}$

→ find $\mathbf{I}'$, $\mathbf{J}'$, $\mathbf{I}''$, $\mathbf{J}''$, $\mathbf{I}'''$, $\mathbf{J}'''$

→ fit a conic to them:

→ This conic is the IAC ω !!

→ then take the DIAC ω^{-1}



known planar scene

→ estimate homographies

→ apply them to $\mathbf{I}$ and $\mathbf{J}$

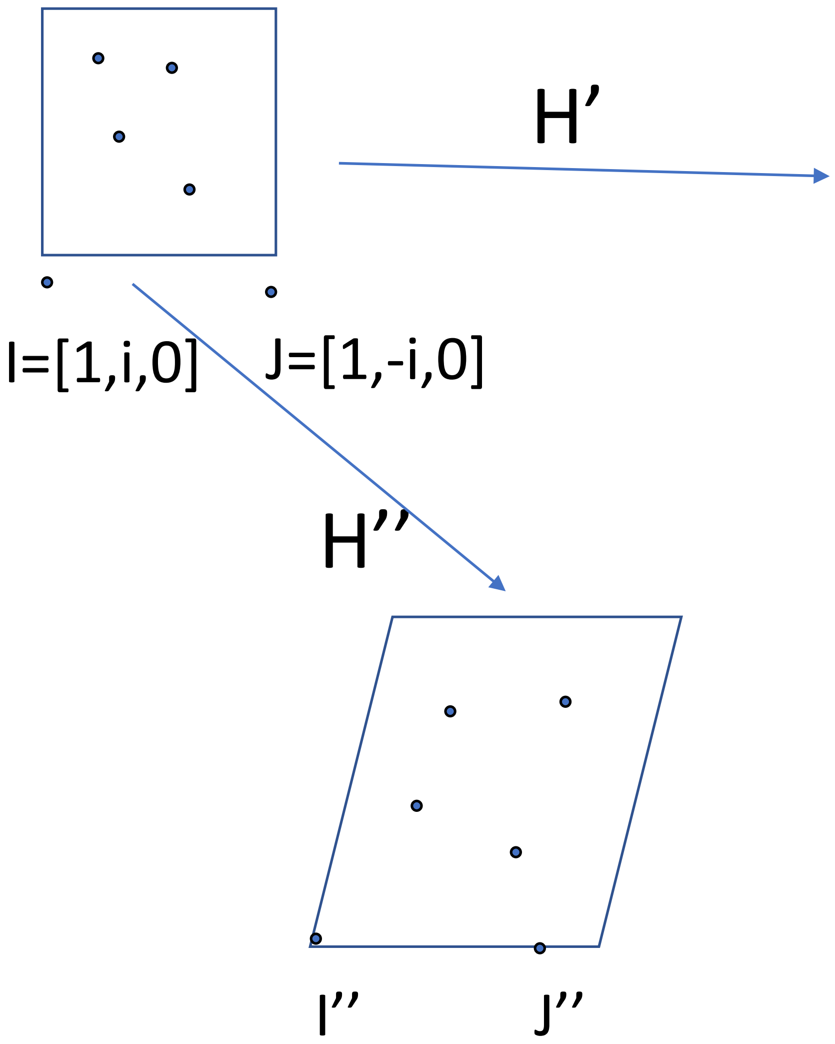
→ find $\mathbf{I}'$, $\mathbf{J}'$, $\mathbf{I}''$, $\mathbf{J}''$, $\mathbf{I}'''$, $\mathbf{J}'''$

→ fit a conic to them:

→ This conic is the IAC ω !!

→ then take the DIAC ω^{-1}

→ find $\mathbf{K}$ by Cholesky factorisation of $\omega^{-1} = \mathbf{K}\mathbf{K}^{\mathrm{T}}$



known planar scene

→ estimate homographies

→ apply them to $\mathbf{I}$ and $\mathbf{J}$

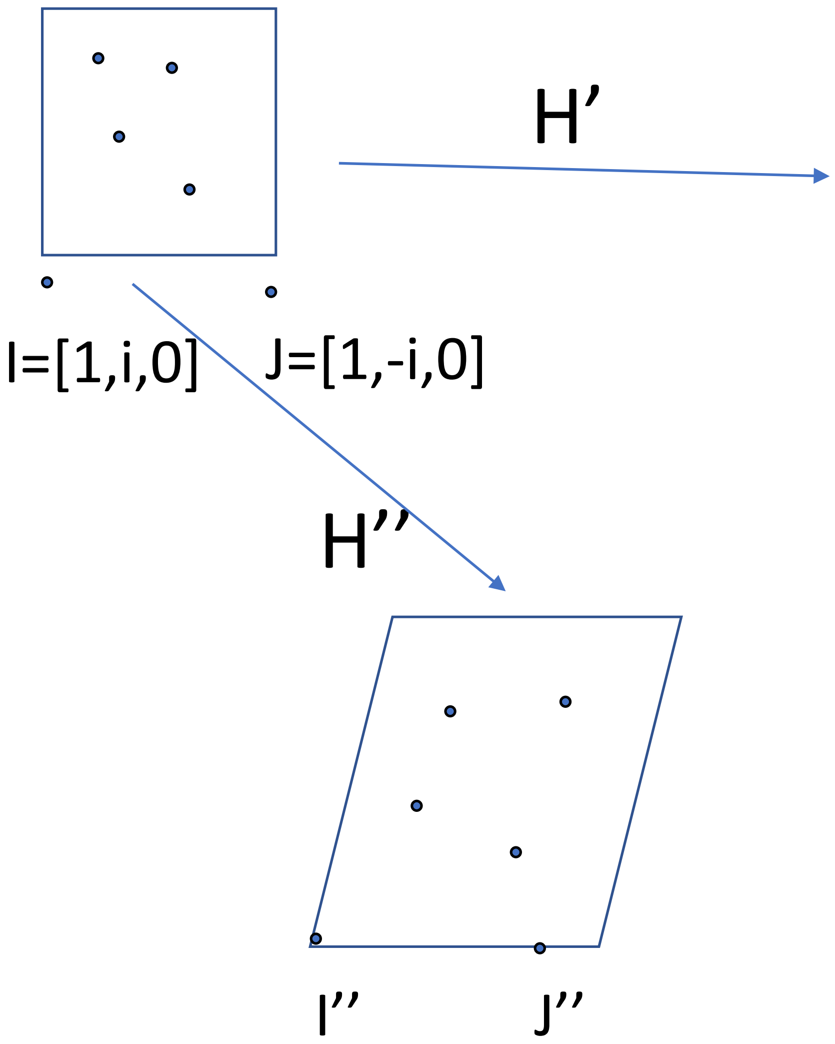
→ find $\mathbf{I}'$, $\mathbf{J}'$, $\mathbf{I}''$, $\mathbf{J}''$, $\mathbf{I}'''$, $\mathbf{J}'''$

→ fit a conic to them: (HOW?)

→ This conic is the IAC ω !!

→ then take DIAC ω^{-1}

→ find $\mathbf{K}$ by Cholesky factorisation of $\omega^{-1} = \mathbf{K}\mathbf{K}^{\mathrm{T}}$



known planar scene

→ estimate homographies

→ apply them to $\mathbf{I}$ and $\mathbf{J}$

→ find $\mathbf{I'}$, $\mathbf{J'}$, $\mathbf{I''}$, $\mathbf{J''}$, $\mathbf{I'''}$, $\mathbf{J'''}$

→ fit a conic to them: (HOW?)

→ This conic is the IAC ω !! (WHY?)

→ then take DIAC ω^{-1}

→ find $\mathbf{K}$ from Cholesky factorisation of $\omega^{-1} = \mathbf{K}\mathbf{K}^{\mathrm{T}}$

HOW? Fitting ω to images of circular points

$$I' = H'I = [h_1, h_2, h_3]I = [h_1, h_2, h_3]\begin{bmatrix}1\\i\\0\end{bmatrix} = h_1 + ih_2 \in \omega$$

$$(h_1 + ih_2)^T\omega(h_1 + ih_2) = 0$$

Real part = 0   and   imaginary part = 0

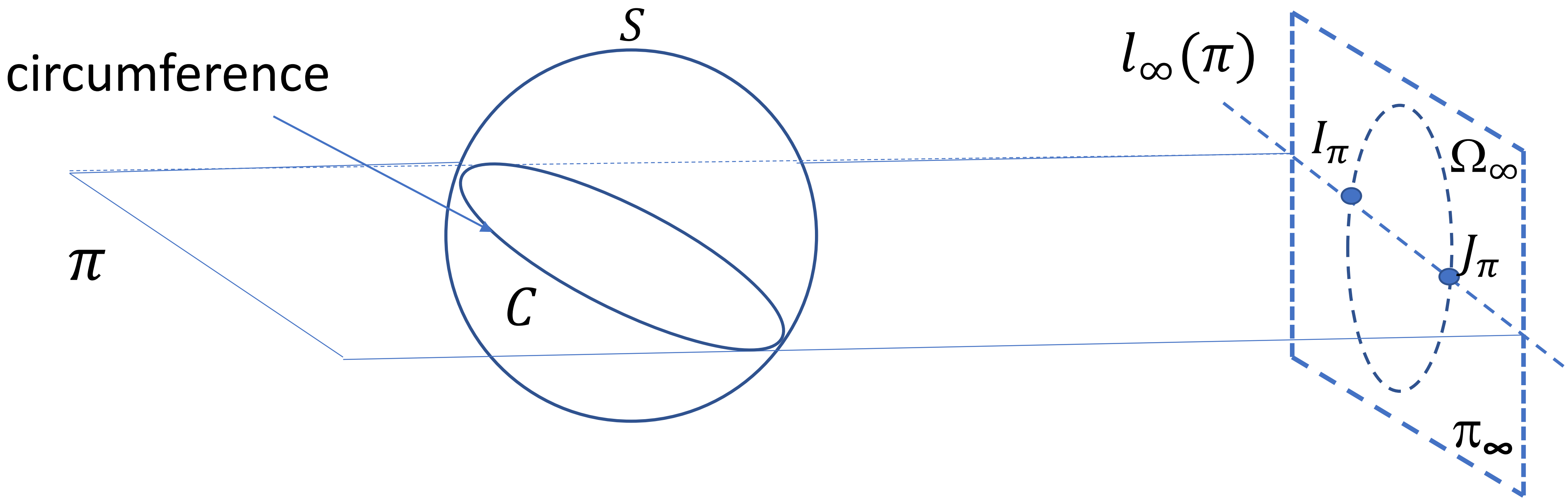
$$h_1{}^T\omega h_2 = 0$$
$$h_1{}^T\omega h_1 - h_2{}^T\omega h_2 = 0$$

2 **real** eqns for each image:

→ at least 3 images needed

WHY does IAC ω contain images of circular points?

1. Does the absolute conic Ω_∞ contain all the circular points (i.e. the circular points I_π, J_π of any plane belong to absolute conic Ω_∞) ?

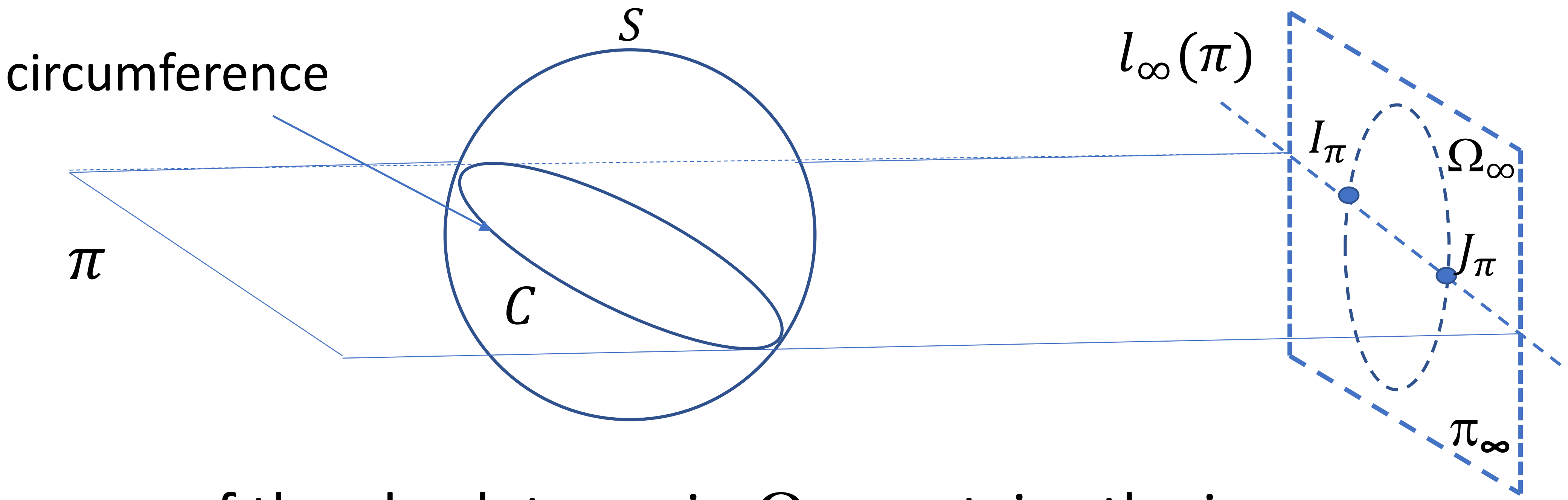


C crosses $l_\infty(\pi)$ at circular points I_π, J_π but $C \subset S$, thus $I_\pi, J_\pi \in S$

In addition, $I_\pi, J_\pi \in \pi_\infty \rightarrow I_\pi, J_\pi \in \pi_\infty \cap S = \Omega_\infty$

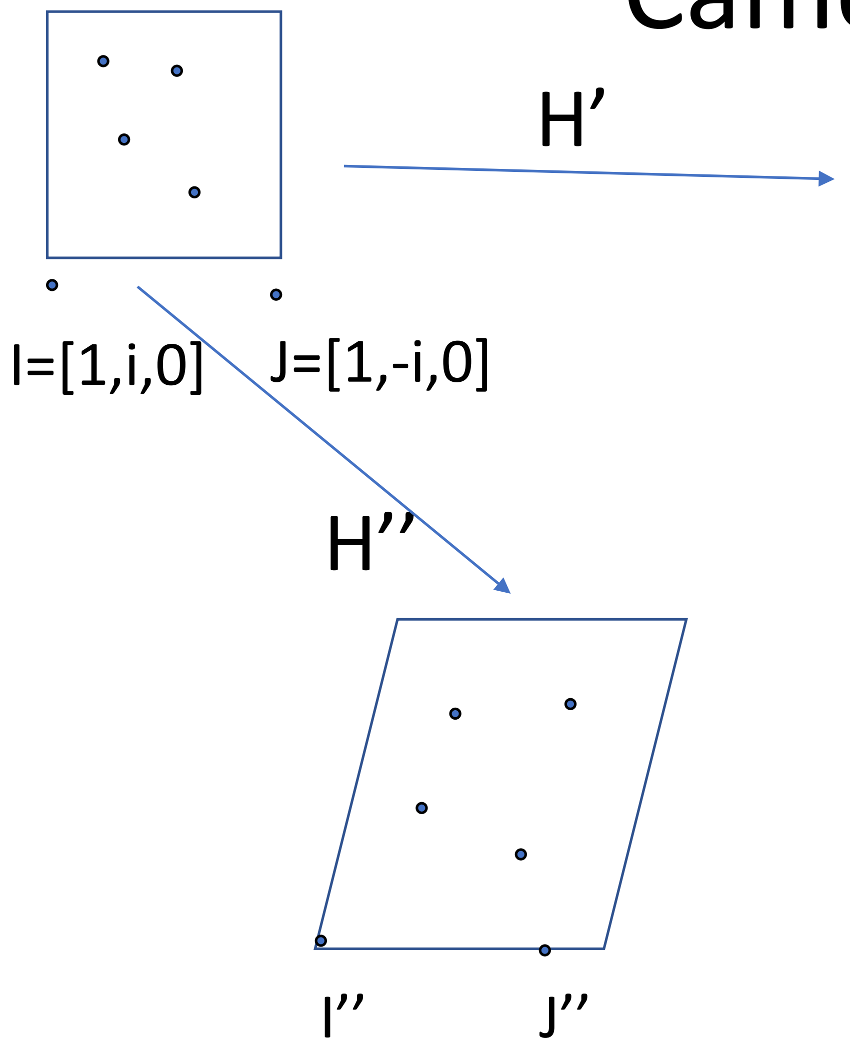
WHY does IAC ω contain images of circular points?

1. Yes: the absolute conic Ω_∞ contains all the circular points (i.e. the circular points I_π, J_π of any plane belong to absolute conic Ω_∞) !!



→ The image ω of the absolute conic Ω_∞ contains the image I'_π, J'_π of all the circular points I_π, J_π

Camera calibration from images of a known scene



Take >= 3 images

→ estimate homographies

→ apply them to $\mathbf{I}$ and $\mathbf{J}$

→ find $\mathbf{I}$’, $\mathbf{J}$’, $\mathbf{I}$’’, $\mathbf{J}$’’, $\mathbf{I}$’’’, $\mathbf{J}$’’’

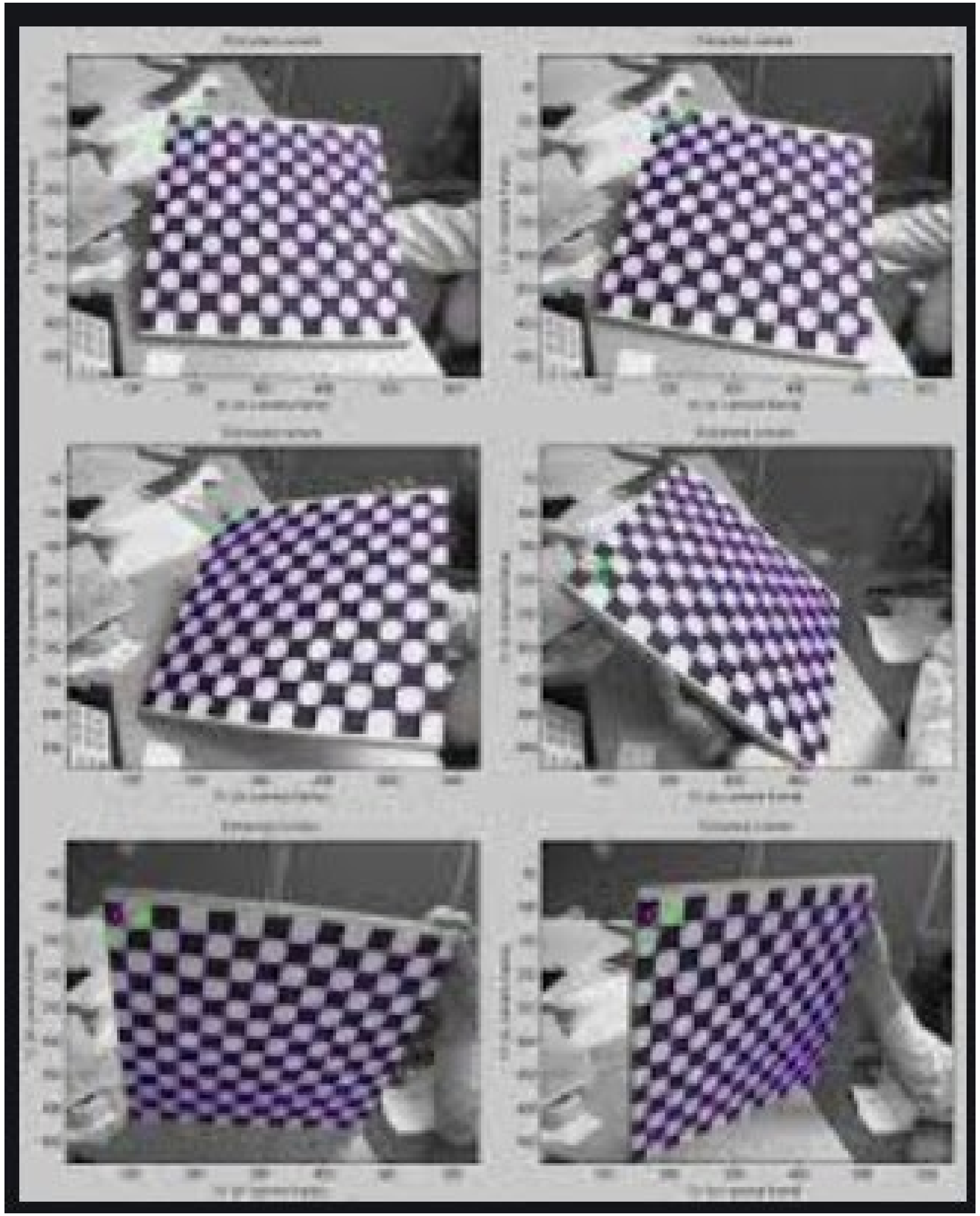
→ fit a conic to them:

→ This conic is the IAC ω !!

→ then take the DIAC ω^{-1}

→ find $\mathbf{K}$ by Cholesky factorisation of $\omega^{-1} = \mathbf{K}\mathbf{K}^{\mathrm{T}}$

Camera calibration toolbox
- implements Zhang method
- based on the IAC ω
- planar target (easily printable)
- several images ($\sim$ 20) to cope with noise
- also estimates distortion param.
- provides accurate calibration

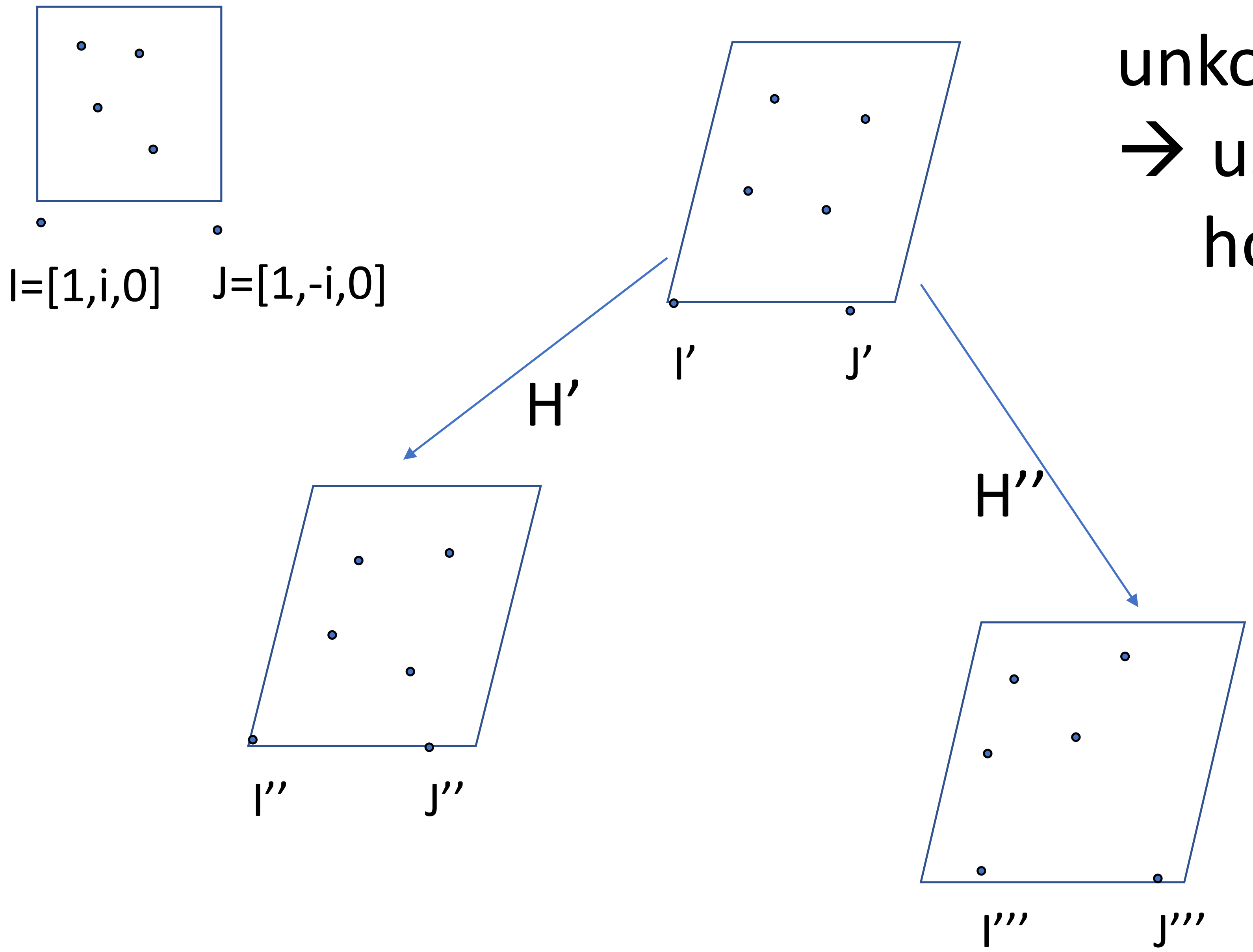


Matlab Calibration Toolbox also contains estimation of distortion parameters

$$x = x_o + (x_o - c_x)(K_1 r^2 + K_2 r^4 + \dots)$$
$$y = y_o + (y_o - c_y)(K_1 r^2 + K_2 r^4 + \dots)$$

$$r = (x_o - c_x)^2 + (y_o - c_y)^2 \quad .$$

Camera calibration from images of an **unknown** planar scene



unkown planar scene

→ use image-to-image homographies

$$I'' = H'I'$$
$$I'^T \omega I' = 0$$
$$I''^T \omega I'' = 0$$

$$I'^T H'^T \omega H' I' = 0$$

Unknowns: I' and J' and ω → at least 5 images
(each nonlinear eqn leads to 2 constraints: Re and Im part)

$$I'' = H'I'$$
$$I'^T \omega I' = 0$$
$$I''^T \omega I'' = 0$$

$$I'^T H'^T \omega H' I' = 0$$

Unknowns: I' and J' and ω → at least 5 images
(each nonlinear eqn leads to 2 constraints: Re and Im part)

Camera calibration from images of an **unknown** planar scene

- take images of a planar scene with camera with constant $\mathbf{K}$
- estimate image-image homographies
- formulate equations

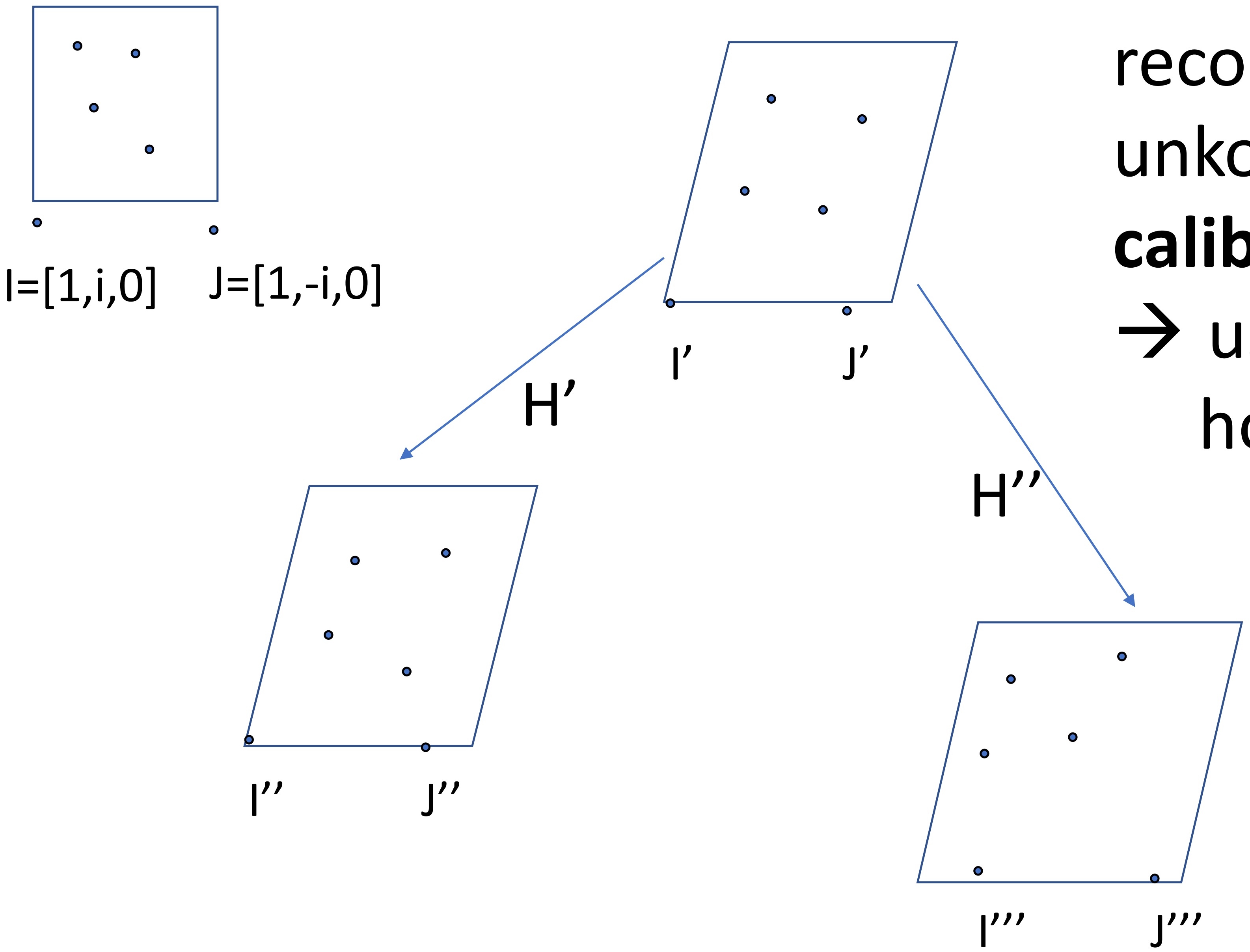
$$I'^T \omega I' = 0$$

$$I'^T H'^T \omega H' I' = 0$$

- solve them for ω and I'
- then take the DIAC ω^{-1}
- find $\mathbf{K}$ by Cholesky factorisation of $\omega^{-1} = \mathbf{K}\mathbf{K}^{\mathrm{T}}$

USUALLY LESS ACCURATE THAN ZHANG METHOD

Reconstruction of an unknown planar scene from calibrated images (i.e. images taken by calibrated cameras)



reconstruction of an unkown planar scene, but **calibrated camera**

→ use image-to-image homographies

same calibrated camera

$$I'' = H'I'$$

$$I'^T \omega I' = 0$$

$$I''^T \omega I'' = 0$$

$$I'^T H'^T \omega H' I' = 0$$

$\omega = \left(\mathrm{KK^T}\right)^{-1} = \mathrm{K^{-T}K^{-1}}$ is known after calibration

just 4 unknowns I' complex coordinates→ at least 2 images
(each eqn leads to 2 constraints: Re and Im part)

two different calibrated cameras

$$I'' = H'I'$$

$$I'^{T}\omega' I' = 0$$

$$I''^{T}\omega'' I'' = 0$$

$$I'^{T} H'^{T}\omega'' H' I' = 0$$

ω', ω'' known after calibration

just 4 unknowns I' complex coordinates → at least 2 images
(each eqn leads to 2 constraints: Re and Im part)

$$I'^T \omega' I' = 0$$
$$I'^T H'^T \omega'' H' I' = 0$$

reduces to intersection of two conics:
ω' and $H'^T \omega'' H'$
→
two resulting pairs of
imaged circular points
→
selection based on reprojection
or on an additional (third) image

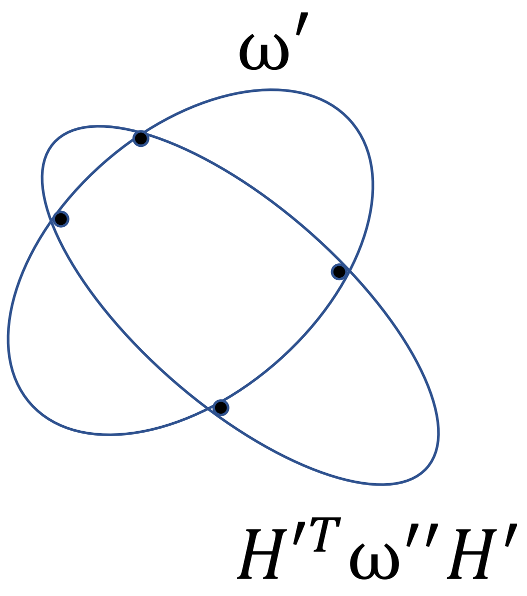


image of circular points: {I’, J’} = $l'_\infty \cap \omega$

- Image of the circular points →Image of the conic dual to the circular points

$$C_\infty^{*\prime} = I'J'^T + J'I'^T$$

- Singular value decomposition

$$\text{svd}(C_\infty^{*\prime}) = U\begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{vmatrix}U^T = H_{rect}^{-1} C_\infty^* H_{rect}^{-T}$$

- Rectifiying homography (from svd output U)

$$H_{rect} = U^T$$

In practice …

- Image of the circular points →Image of the conic dual to the circular points

$$C_\infty^{*\prime} = I'J'^T + J'I'^T$$

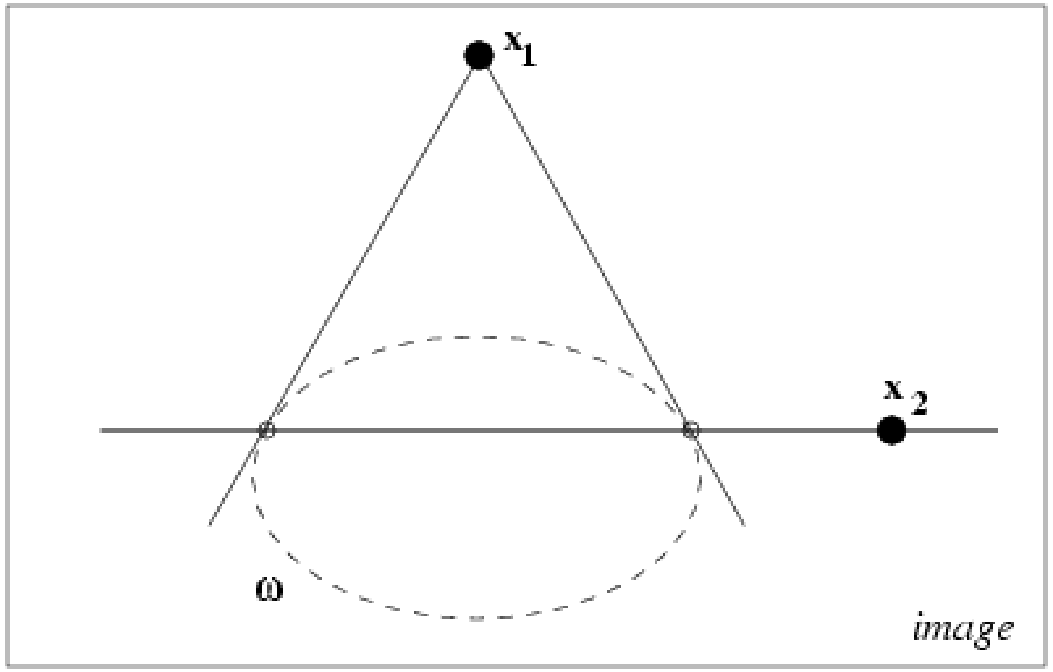
- Singular value decomposition

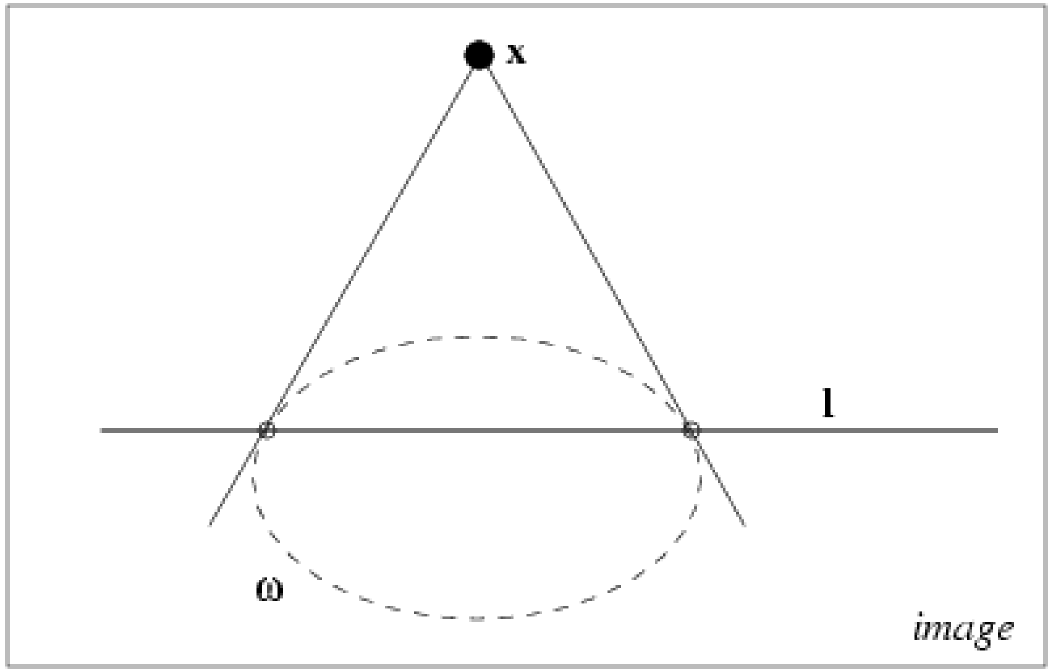
$$\text{svd}(C_\infty^{*\prime}) = U \begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & \varepsilon \end{bmatrix} U^T$$

- Rectifiying homography (from svd output U)

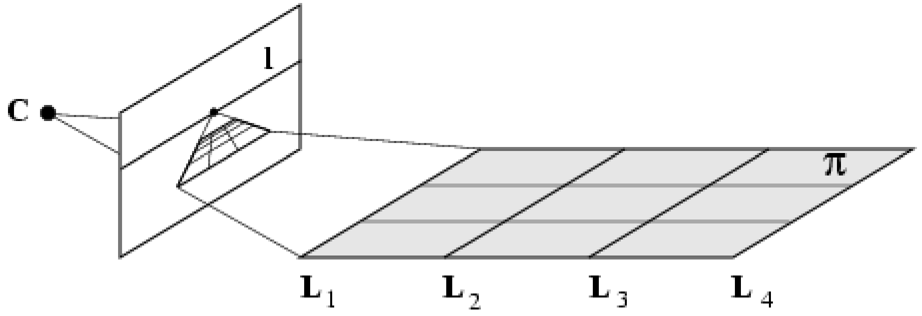
$$H_{rect} = (U \begin{bmatrix} \sqrt{s_1} & 0 & 0 \\ 0 & \sqrt{s_2} & 0 \\ 0 & 0 & 1 \end{bmatrix})^{-1} = \begin{bmatrix} \sqrt{{s_1}^{-1}} & 0 & 0 \\ 0 & \sqrt{{s_2}^{-1}} & 0 \\ 0 & 0 & 1 \end{bmatrix} U^T$$

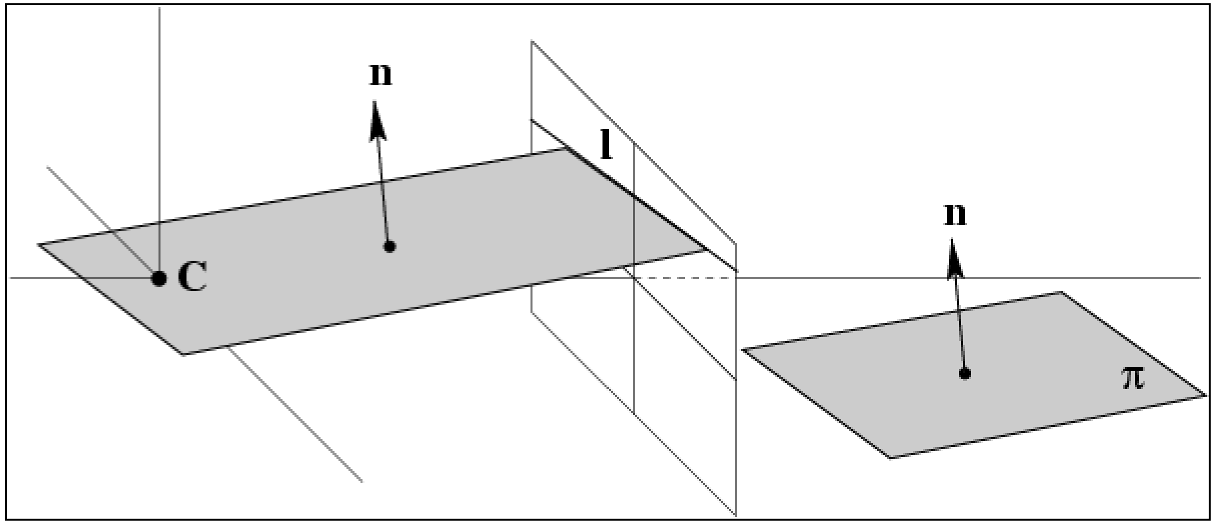
Orthogonality = pole-polar w.r.t. IAC



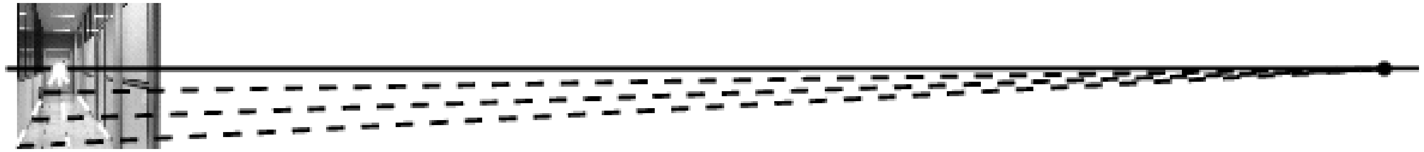


Vanishing lines







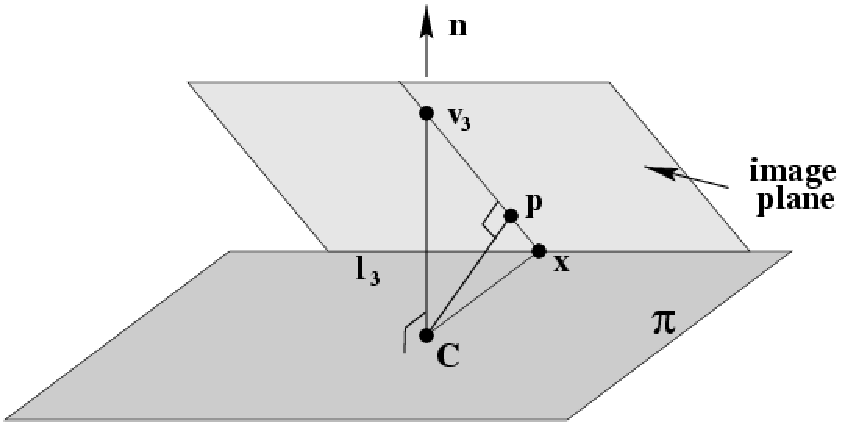
vanishing line: line through two vanishing points

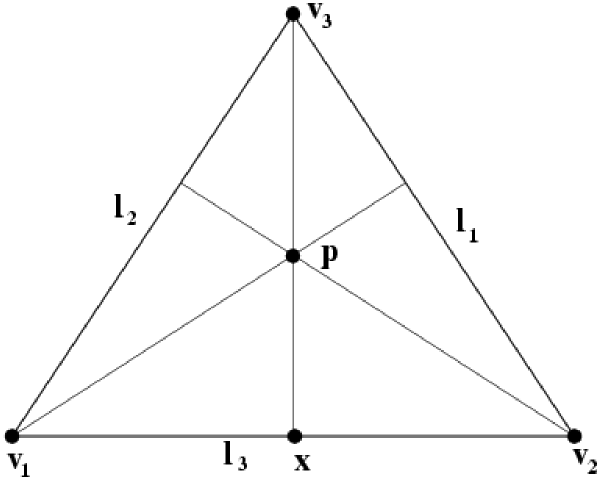
Orthogonality relation: nonlinear equation on ω reduces to linear for orthogonal directions

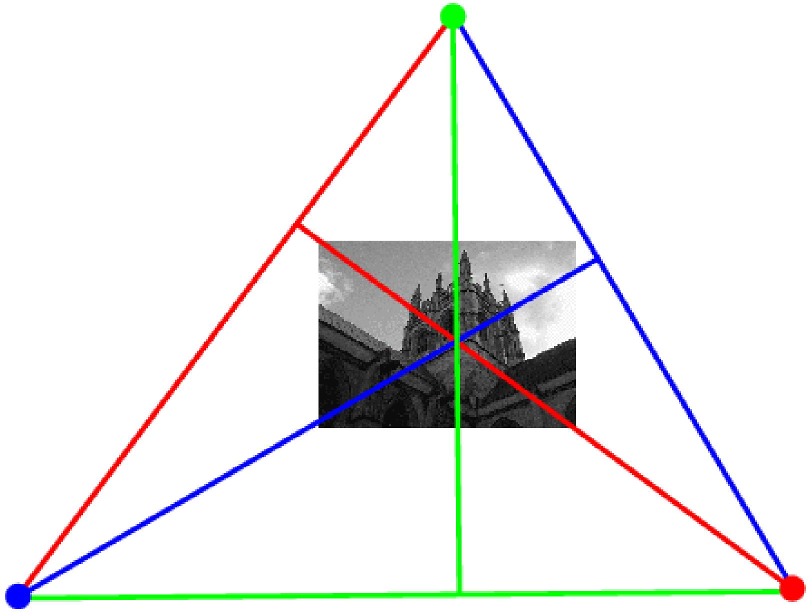
$$\cos\theta = \frac{\mathrm{v}_1^{\mathrm{T}}\omega\mathrm{v}_2}{\sqrt{\left(\mathrm{v}_1^{\mathrm{T}}\omega\mathrm{v}_1\right)\left(\mathrm{v}_2^{\mathrm{T}}\omega\mathrm{v}_2\right)}}$$

$$\mathrm{v}_1^{\mathrm{T}}\omega\mathrm{v}_2 = 0$$

Calibration of NATURAL CAMERAS from vanishing points of orthogonal directions







Orthogonality relation

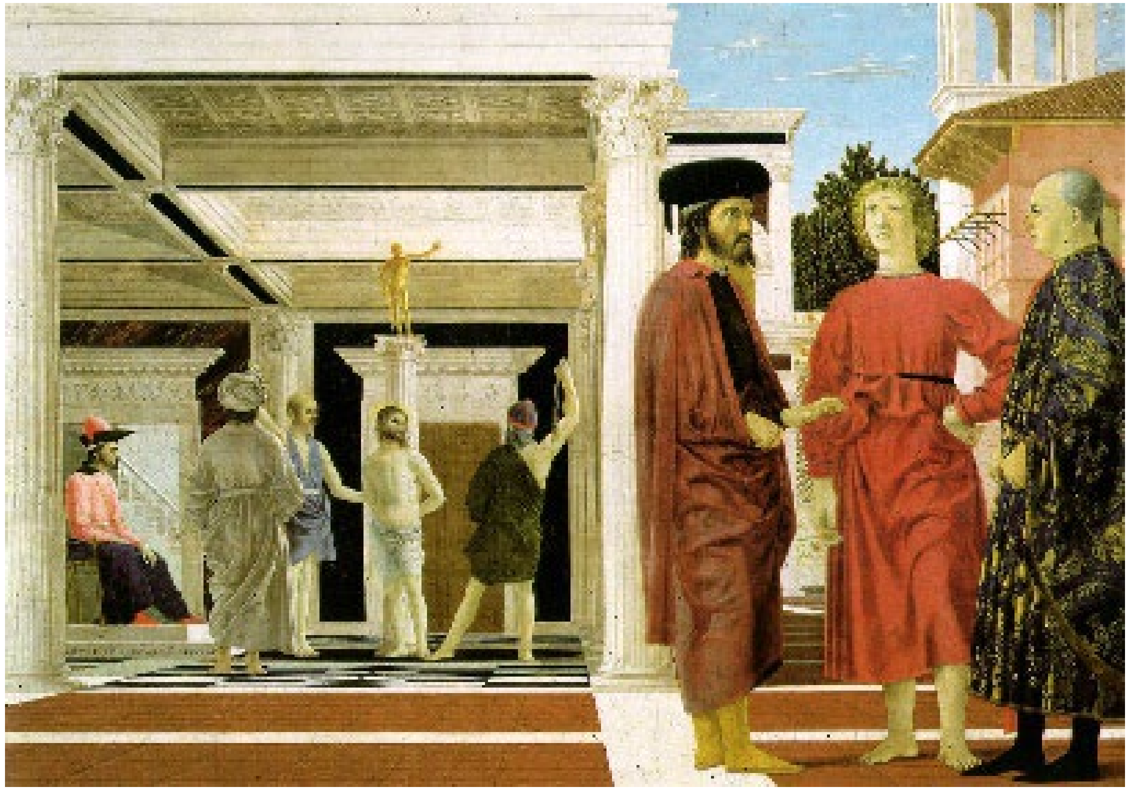
$$\cos\theta = \frac{\mathrm{v_1}^{\mathrm{T}}\omega\mathrm{v_2}}{\sqrt{(\mathrm{v_1}^{\mathrm{T}}\omega\mathrm{v_1})(\mathrm{v_2}^{\mathrm{T}}\omega\mathrm{v_2})}} = 0$$

$$\mathrm{v_1}^{\mathrm{T}}\omega\mathrm{v_2} = 0$$
$$\mathrm{v_2}^{\mathrm{T}}\omega\mathrm{v_3} = 0$$
$$\mathrm{v_3}^{\mathrm{T}}\omega\mathrm{v_1} = 0$$

The painter’s eye

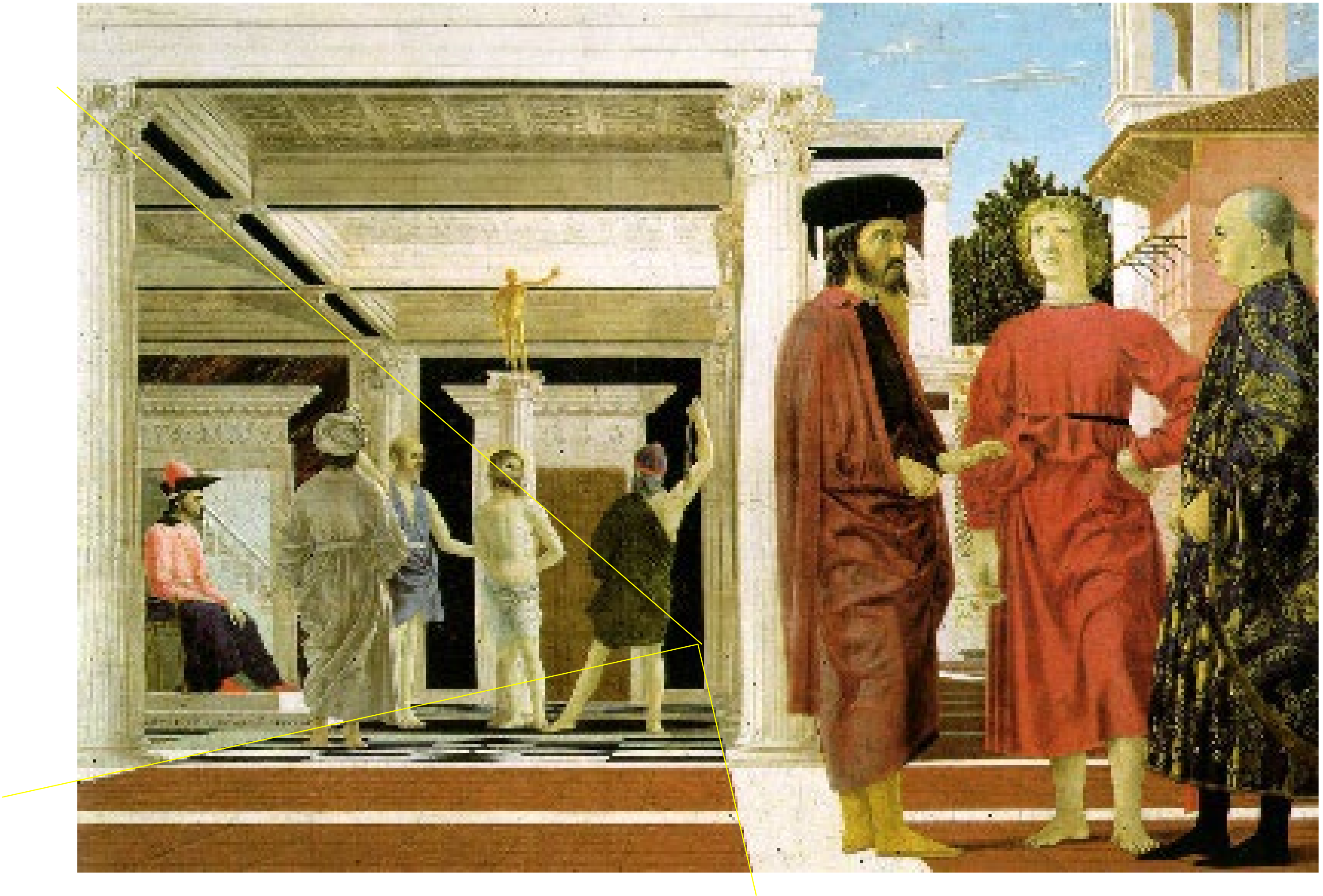
An Example of Camera calibration

where was the painter's eye (wrt to the painting)?

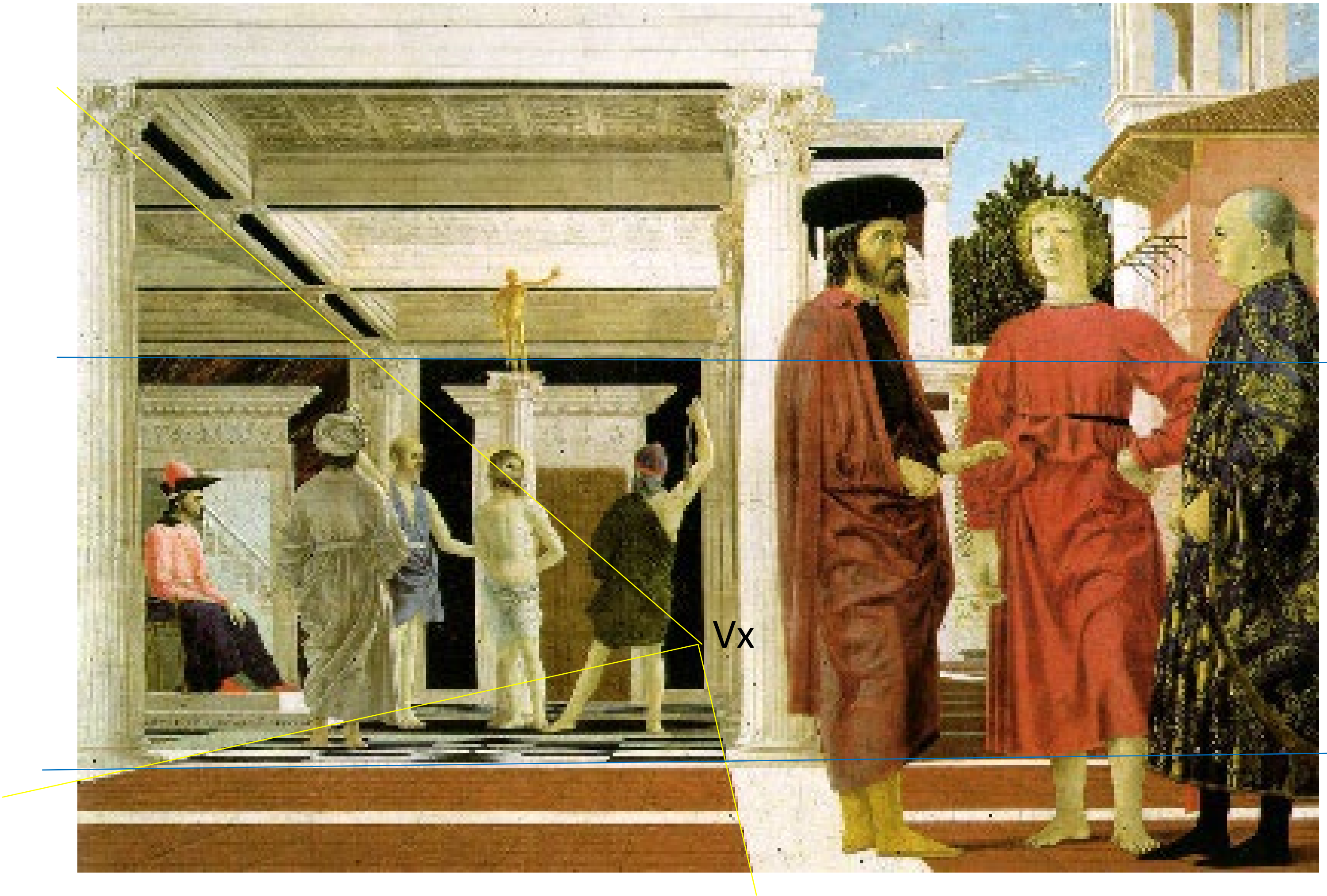


Assumptions: - the floor is horizontal
- floor tiles are square

Vanishing point of «x» direction: V_x



Vanishing point of «y» direction



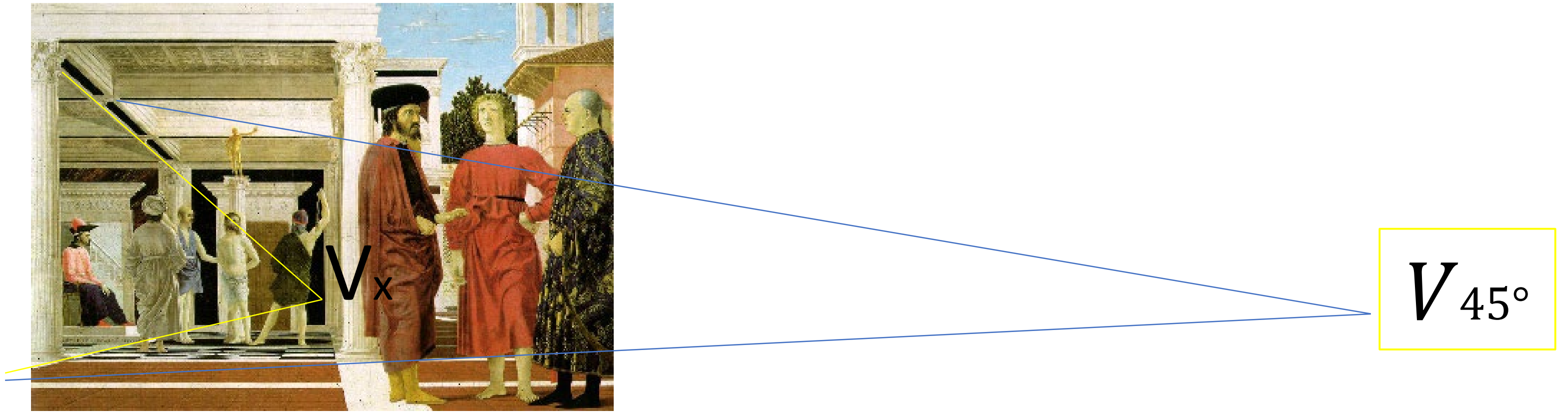
V_∞

Theorem: viewing ray associated to the vanishing point of a direction d is parallel to the direction **d**

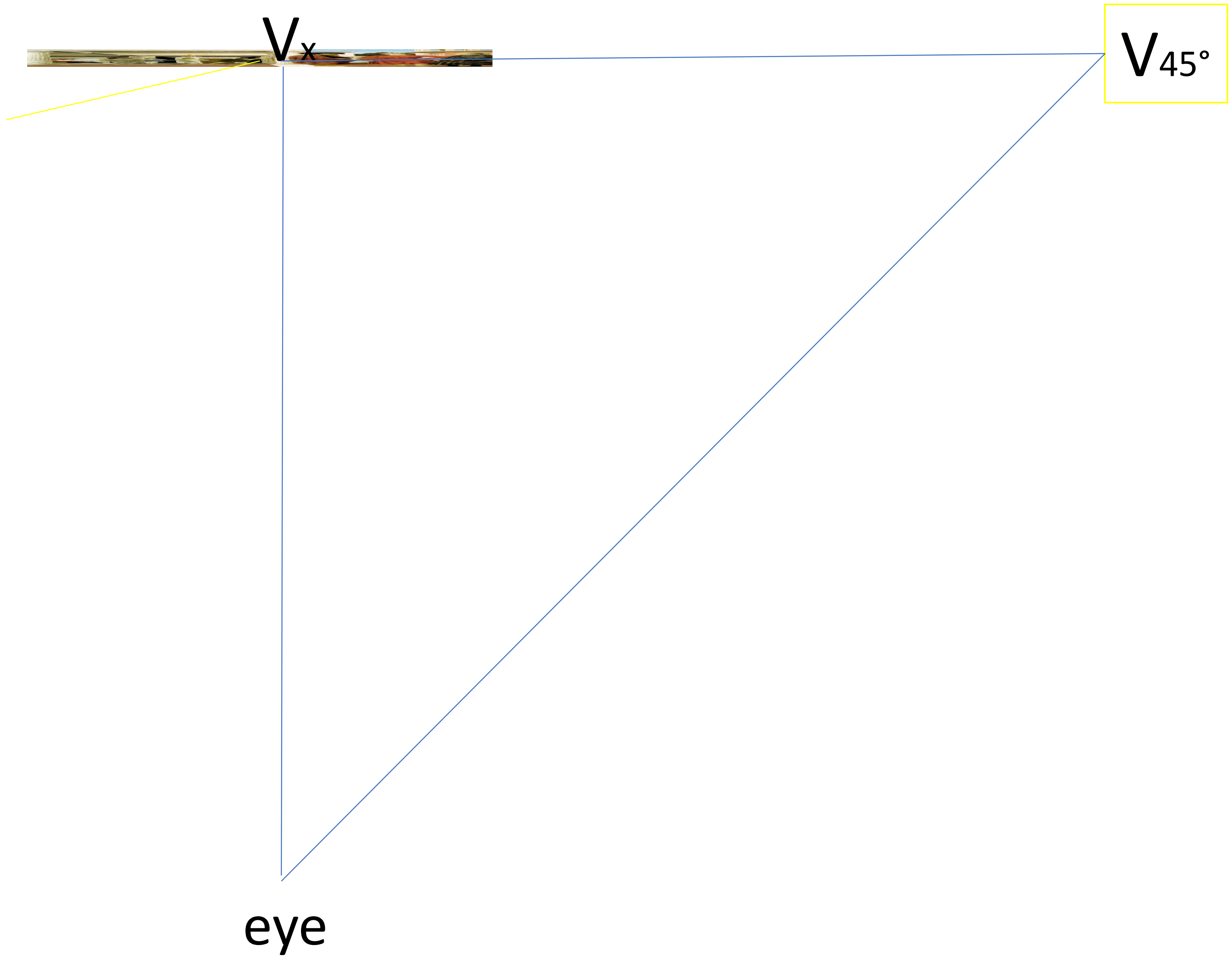
- Floor is horizontal → line (V_x, eye) is horizontal
- Vy is at the infinity → line (eye, V_∞) is parallel to the painting
- Theorem → line (V_∞, eye) is orthogonal to line (V_x, eye)
- → line (eye, V_x) is horizontal and perpendicular to y axis
- … eye is on the normal to the y-axis through the point V_x

- How far?

Tiles are square → diagonals form a 45° angle with x-direction



Theorem → line (eye, V_{45°) forms a 45° angle with line (eye, V_x)
floor horizontal → line (eye, V_{45°) is also horizontal



V_x
$V_{45°}$
eye

Camera calibration from

- a planar object of known (reconstructed) shape and
- a vanishing point of the normal to the plane

calibration after reconstruction of a planar face

Shape of the face: known after metric rectificaton

How much additional information is needed to calibrate the camera?

- Vanishing point of the direction normal to the reconstructed face

Vanishing point of vertical direction

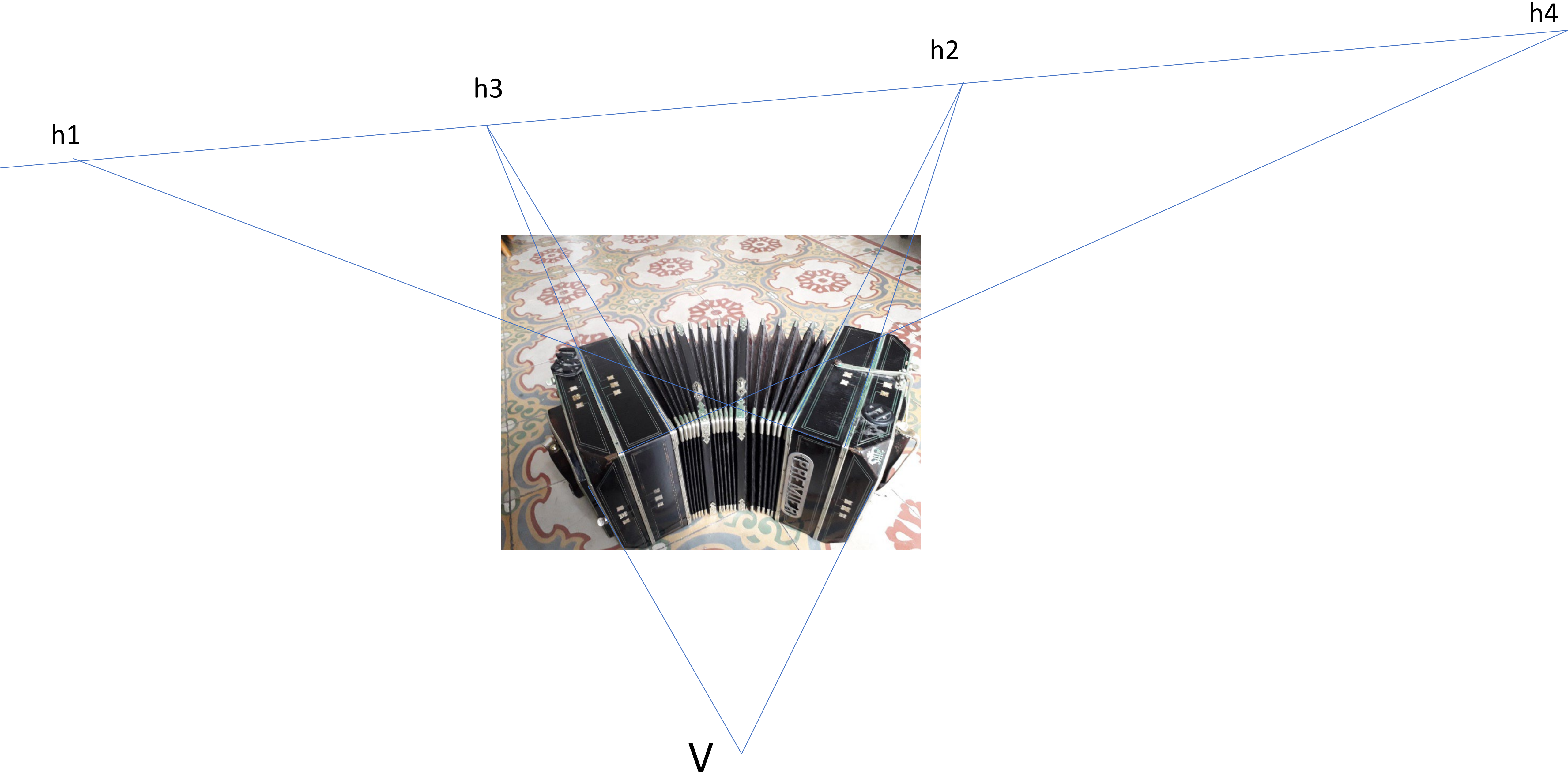
V

Camera calibration (assume skew = 0)

- Only four unknowns:

$$\omega = (KK^T)^{-1} = \begin{vmatrix} a^2 & 0 & -u_0 a^2 \\ * & 1 & -v_0 \\ * & * & f_Y^2 + a^2 u_0^2 + v_0^2 \end{vmatrix}$$

→ only four equations are needed


h1
h3
h2
h4
V

Calibration: rectified planar face plus vanishing point of the direction normal to the face

$$h_1^T \omega h_2 = 0$$
$$h_3^T \omega h_4 = 0$$
$$v^T \omega h_1 = 0$$
$$v^T \omega h_2 = 0$$

3° and 4° equations are linearly independent,

but there are no further ones (why?)

- → solve for ω
- → find $\mathbf{K}$ (by Cholesky factorisation of $\omega^{-1} = \mathbf{K}\mathbf{K}^{\mathrm{T}}$)

Calibration from rectified face plus orthogonal vanishing point

direct method:
independent of the chosen pairs of mutually orthogonal vanishing points

from

- the reconstructing homography H_R from given img to rectified img
- the image of the line at the infinity $\boldsymbol{l}'_{\infty}$
- the vanishing point $\mathbf{v}$ along the direction orthogonal to the face

Calibration from rectified face plus orthogonal vanishing points

- from $\mathbf{h}_1^T \omega \mathbf{v} = 0$ and $\mathbf{h}_2^T \omega \mathbf{v} = 0$, and $\boldsymbol{l}'_\infty = \mathbf{h}_1 \times \mathbf{h}_2$ →

$$\boxed{\boldsymbol{l}'_\infty = \omega \mathbf{v} \text{ (2 eqns)}}$$

- from $I' = H_R^{-1} I = H_R^{-1} \begin{bmatrix} 1 \\ i \\ 0 \end{bmatrix} = [h_1 \quad h_2 \quad h_3] \begin{bmatrix} 1 \\ i \\ 0 \end{bmatrix} = h_1 + i h_2,$

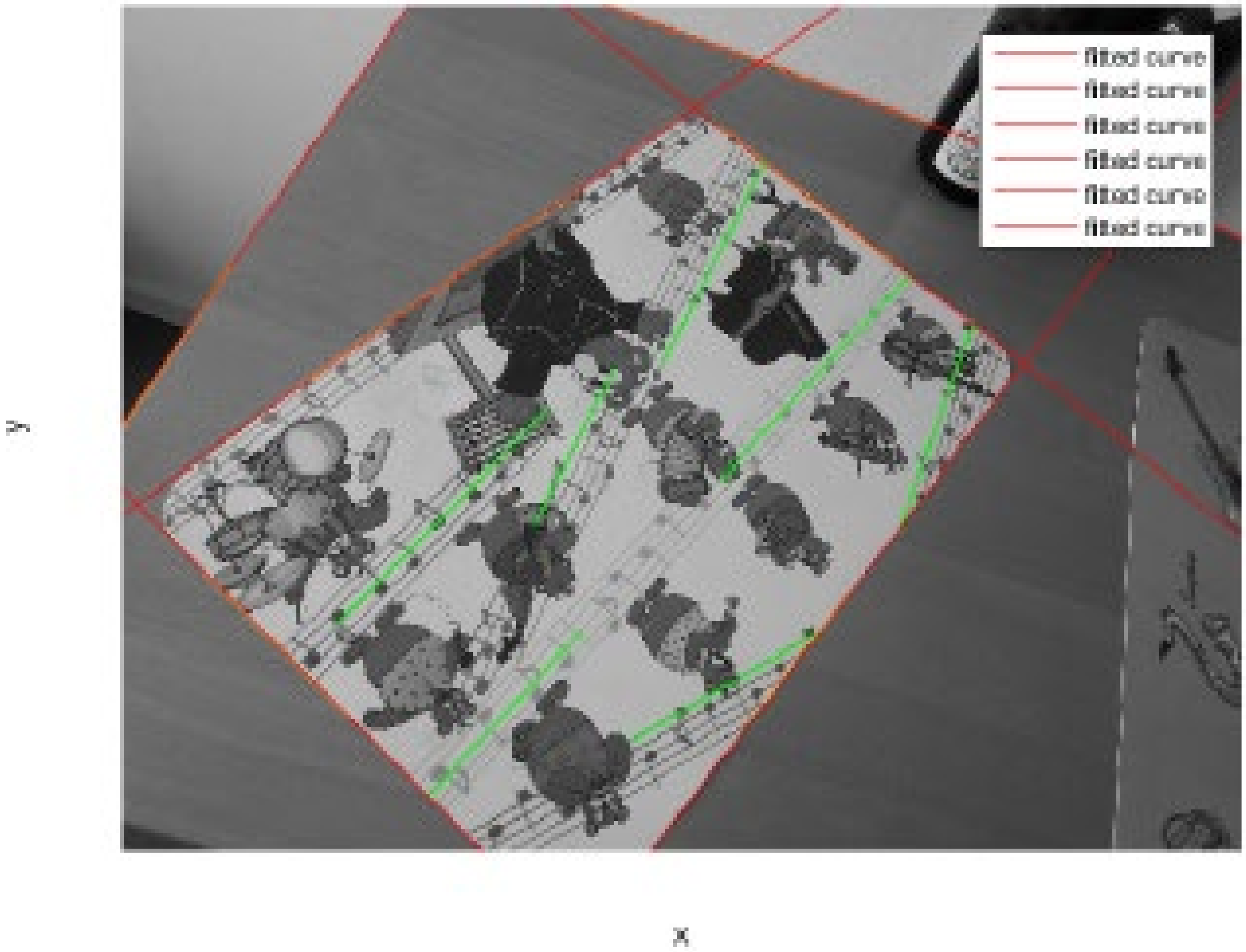
and $(h_1 + i h_2)^T \omega (h_1 + i h_2) = 0$ →

$$\boxed{\begin{gathered} {h_1}^T \omega h_2 = 0 \\ {h_1}^T \omega h_1 - {h_2}^T \omega h_2 = 0 \end{gathered}}$$

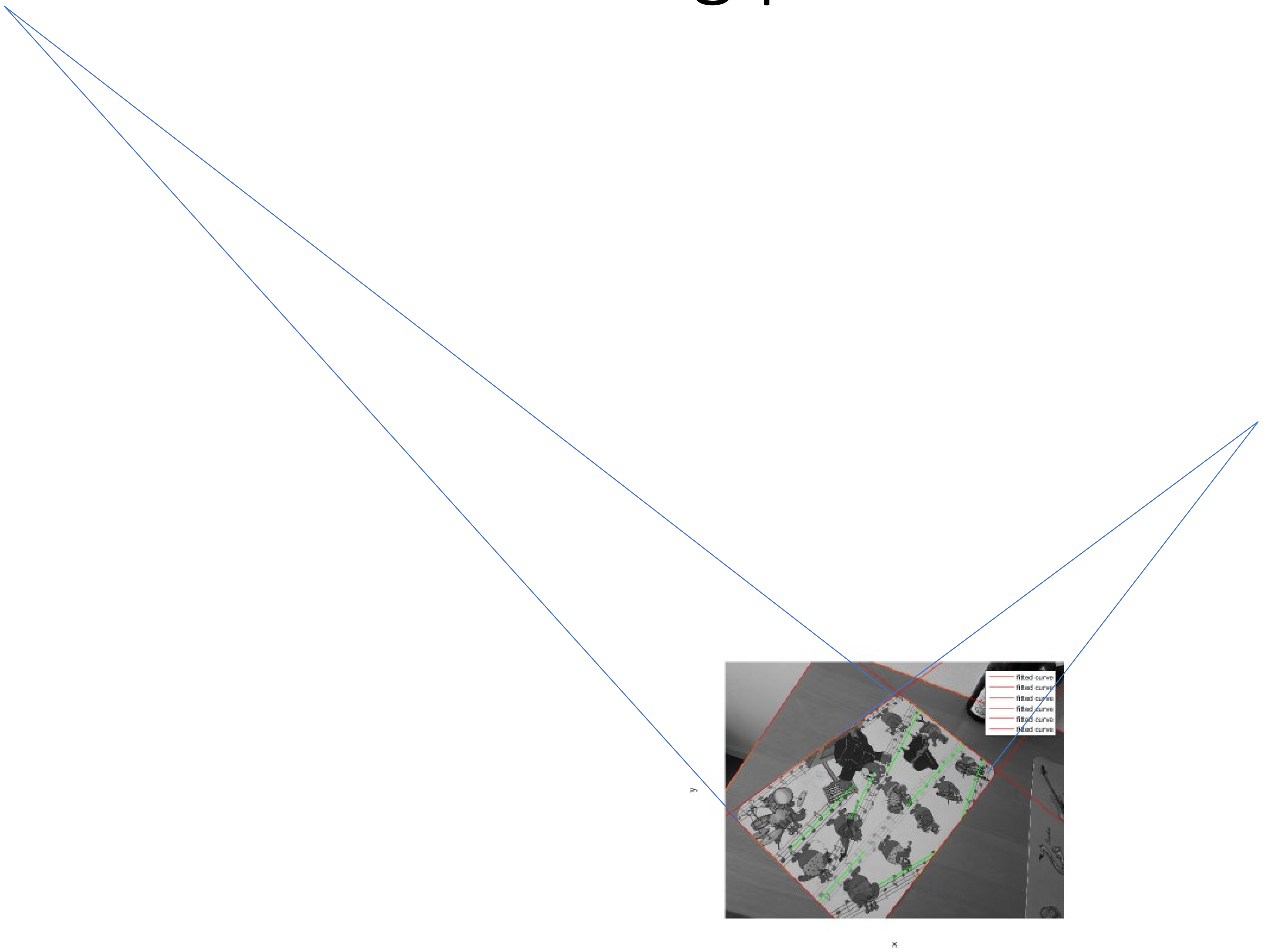
Rectification of a single calibrated image from vanishing points


DONNAS
2003
COLLIO
PODERE
DI PRADIS
Saxophone

Images of pairs of parallel lines



vanishing points



Vanishing line l'_∞

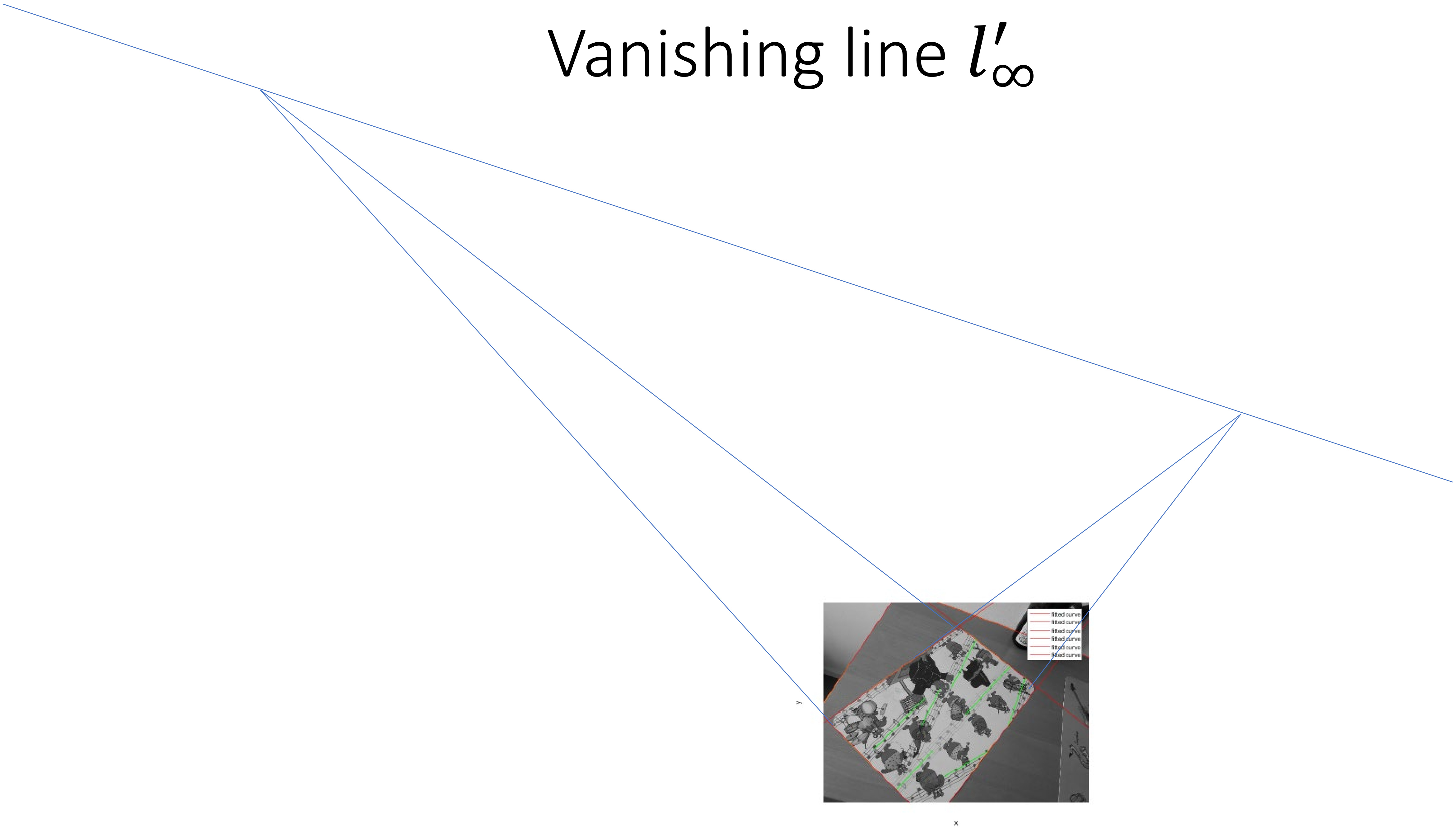


Image of the absolute conic $\omega = (\mathrm{KK^T})^{-1}$

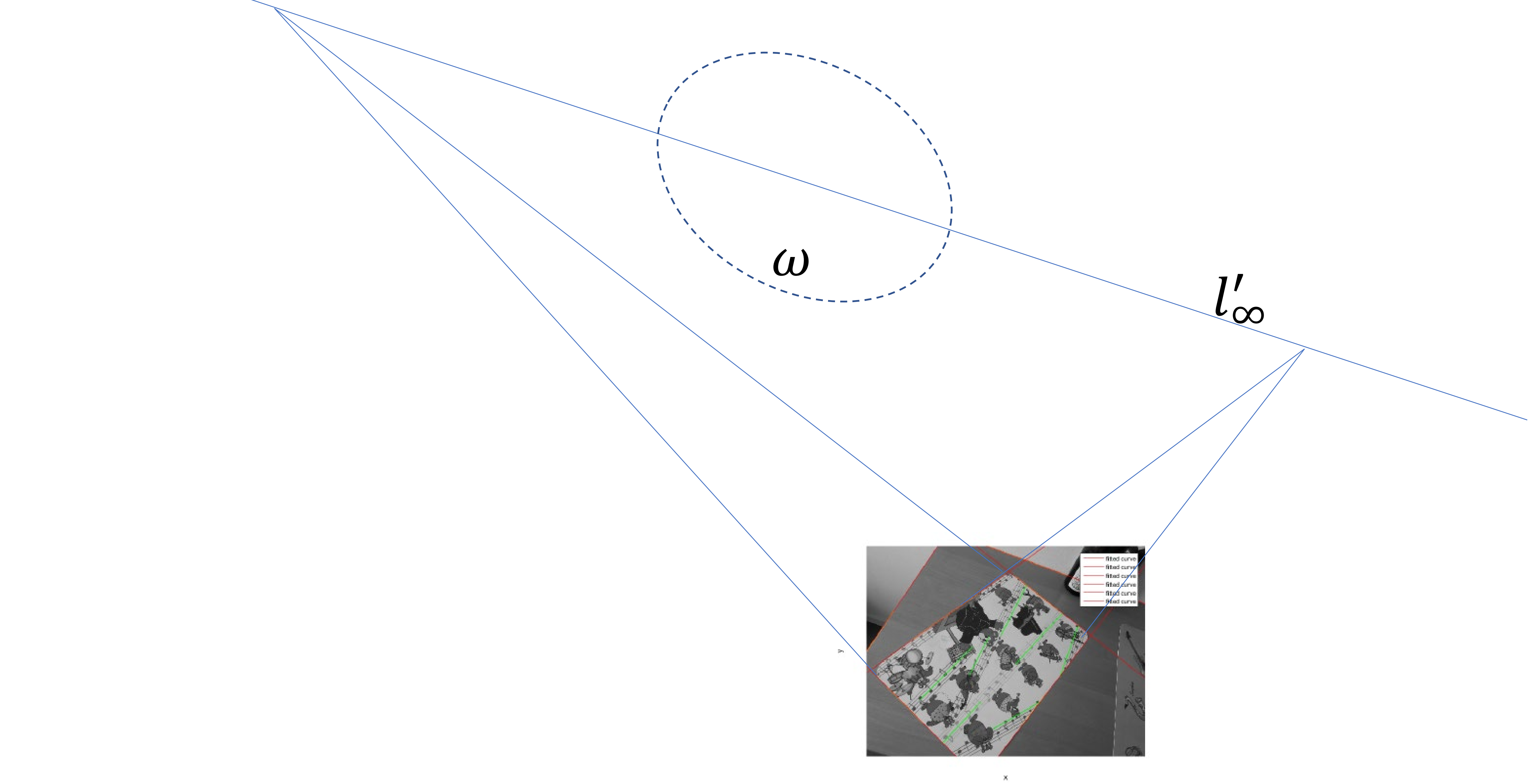


Image of the circular points $l'_\infty \cap \omega$

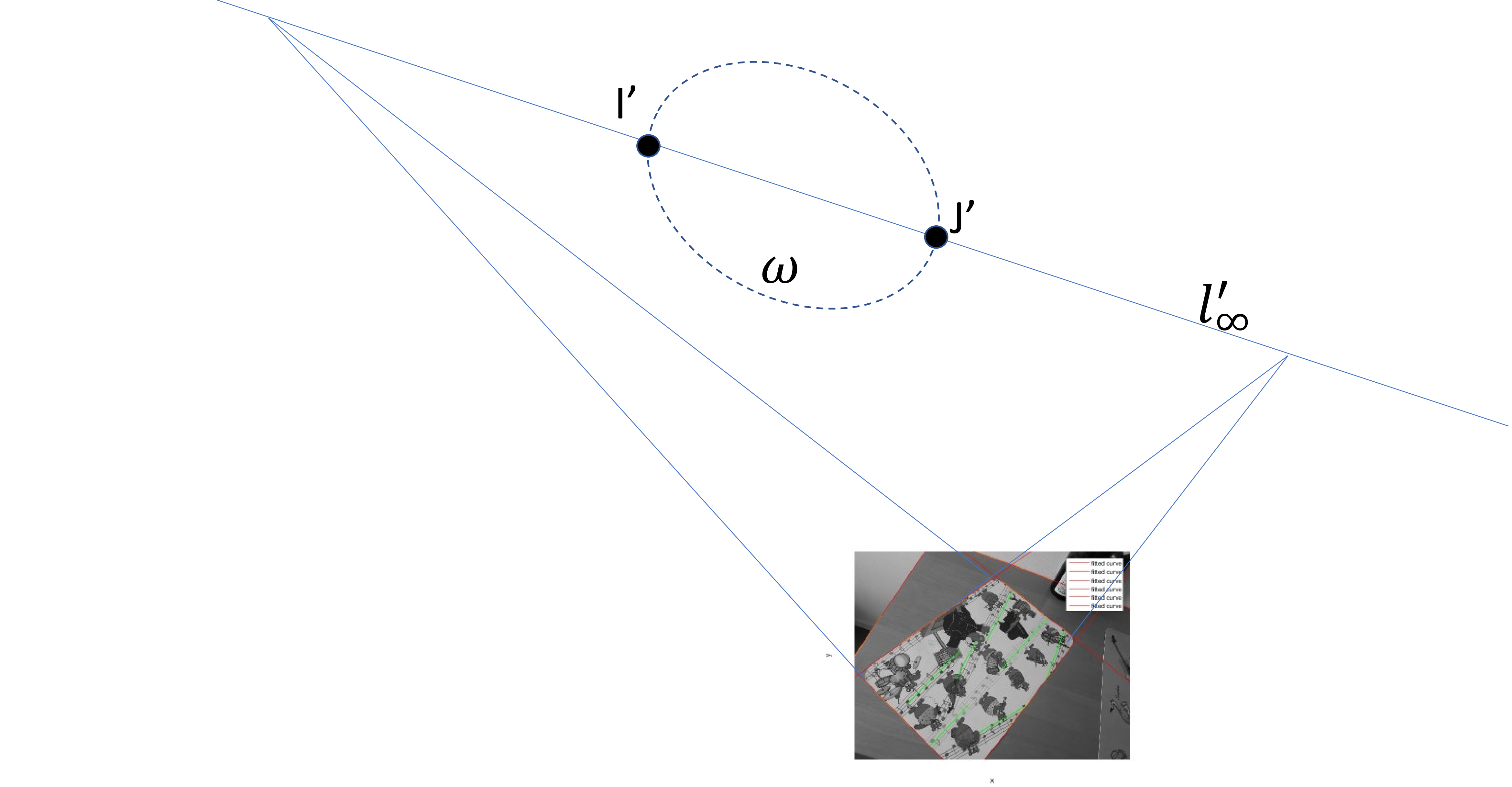
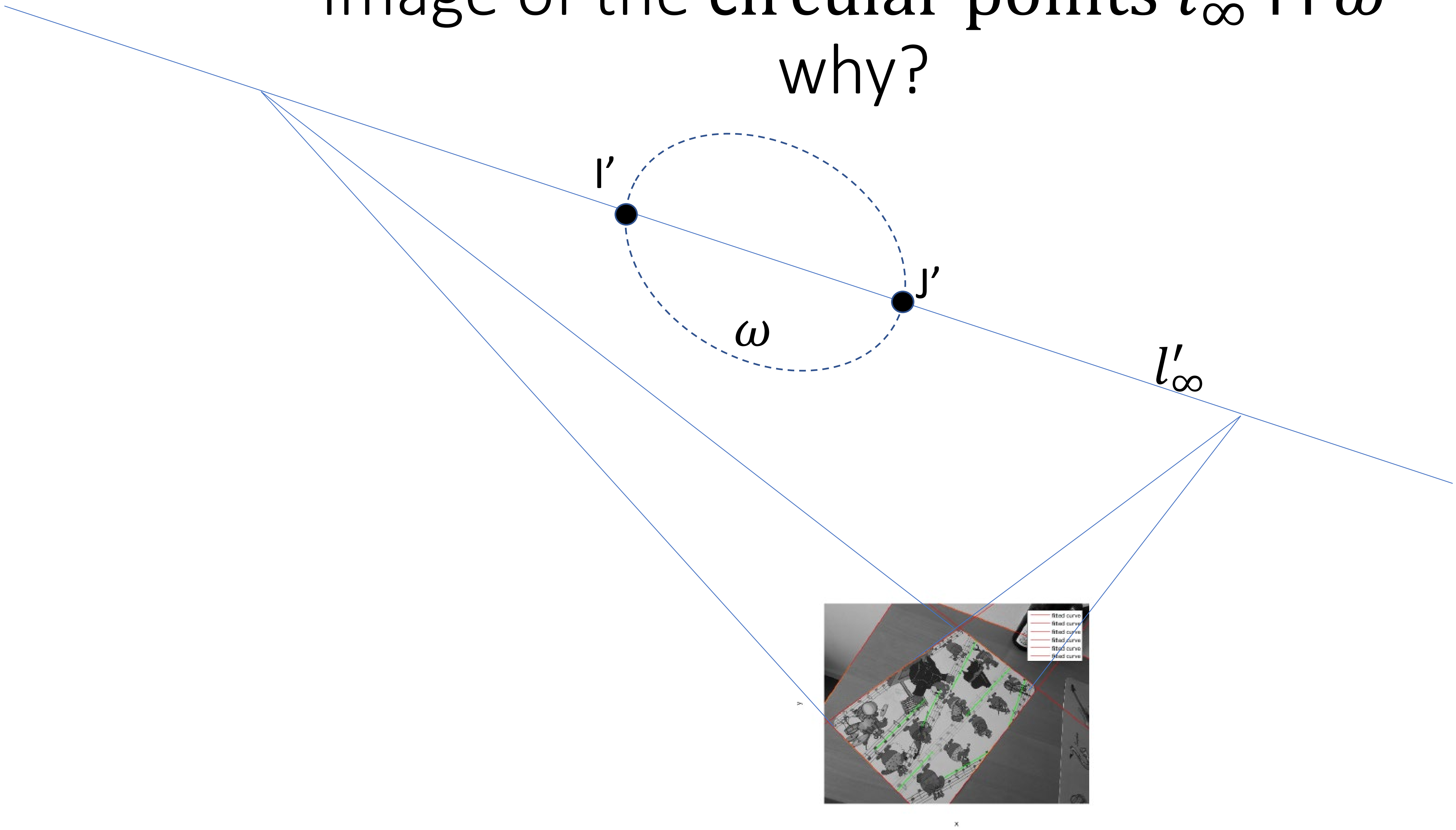
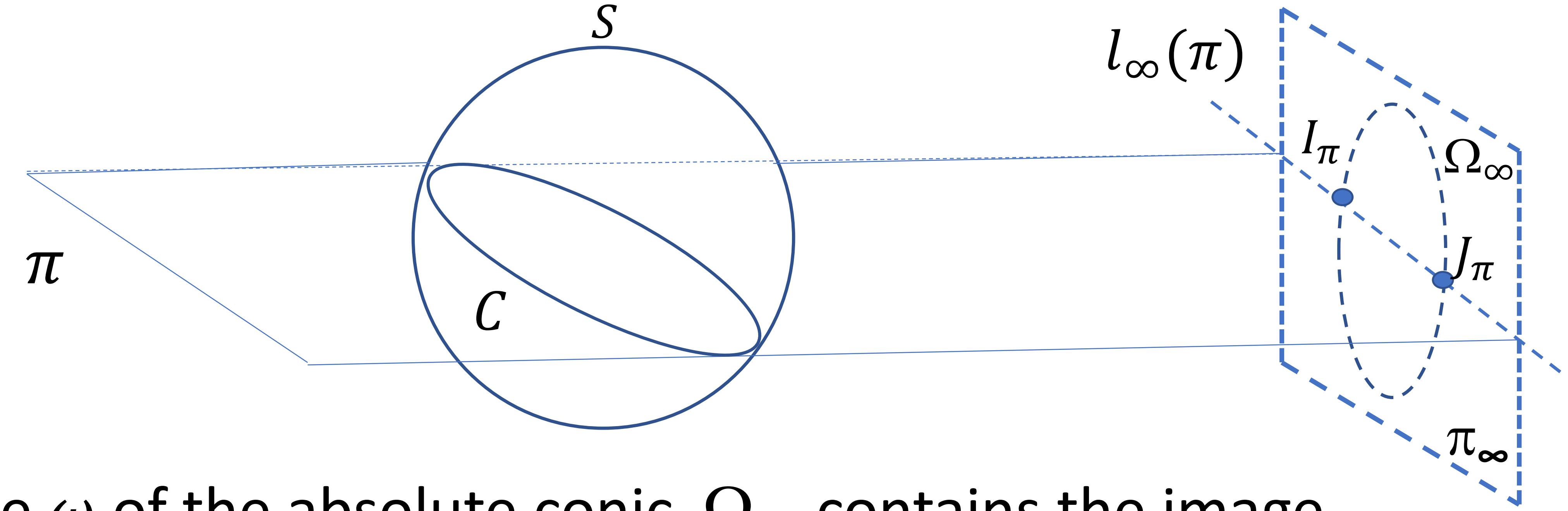


Image of the **circular points** $l'_\infty \cap \omega$ why?



remember: IAC ω contain images of circular points

1. The absolute conic Ω_∞ contains all the circular points (i.e. the circular points I_π, J_π of any plane belong to absolute conic Ω_∞) !!



2. The image ω of the absolute conic Ω_∞ contains the image I'_π, J'_π of all the circular points I_π, J_π

reconstruction of the planar scene

image of circular points: {I’, J’} = $l'_\infty \cap \omega$

- Image of the circular points →Image of the conic dual to the circular points

$$C_\infty^{*\prime} = I'J'^T + J'I'^T$$

- Singular value decomposition

$$\text{svd}(C_\infty^{*\prime}) = U\begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{vmatrix}U^T = H_{rect}^{-1} C_\infty^* H_{rect}^{-T}$$

- Rectifiying homography (from svd output U)

$$H_{rect} = U^T$$

In practice …

- Image of the circular points →Image of the conic dual to the circular points

$$C_\infty^{*\prime} = I' J'^T + J' I'^T$$

- Singular value decomposition

$$\mathrm{svd}(C_\infty^{*\prime}) = U \begin{bmatrix} s_1 & 0 & 0 \\ 0 & s_2 & 0 \\ 0 & 0 & \varepsilon \end{bmatrix} U^T$$

- Rectifiying homography (from svd output U)

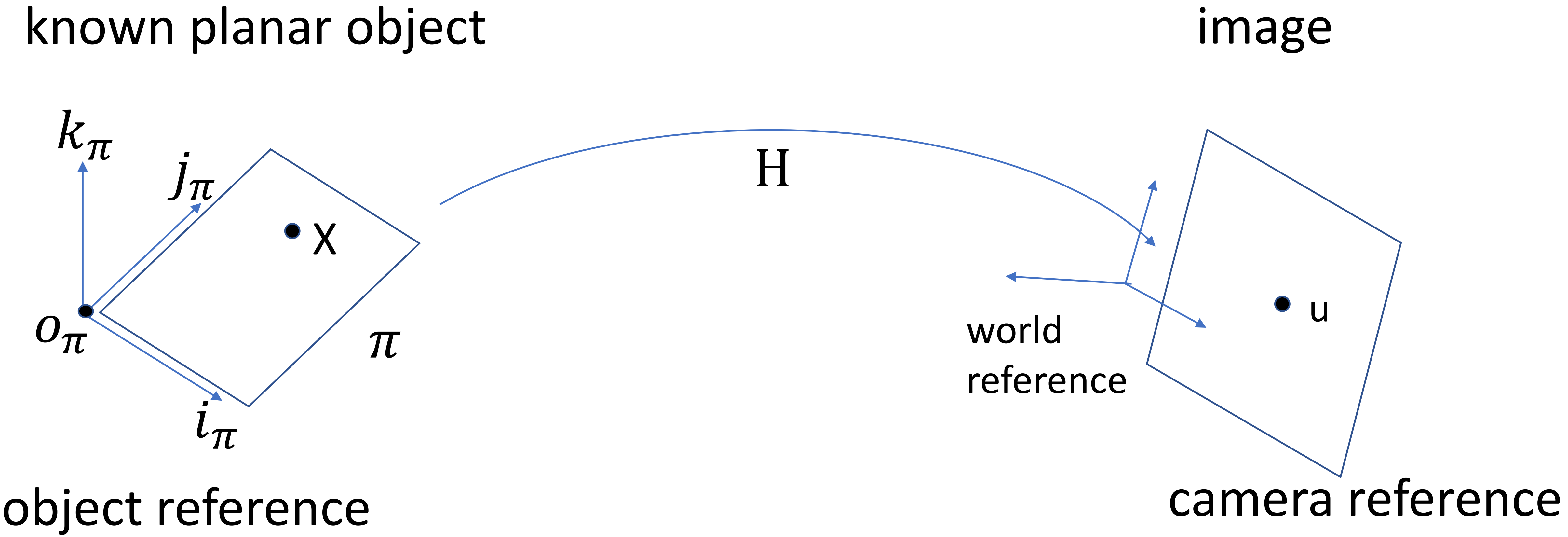
$$H_{rect} = (U \begin{bmatrix} \sqrt{s_1} & 0 & 0 \\ 0 & \sqrt{s_2} & 0 \\ 0 & 0 & 1 \end{bmatrix})^{-1} = \begin{bmatrix} \sqrt{{s_1}^{-1}} & 0 & 0 \\ 0 & \sqrt{{s_2}^{-1}} & 0 \\ 0 & 0 & 1 \end{bmatrix} U^T$$

Euclidean reconstruction = Image rectification

Localization of known (e.g. reconstructed) planar objects from a single calibrated image

Image of the upper face

known shape of the upper face
(e.g. after reconstruction)



world reference ≡ camera reference

$$\mathbf{X_\pi} = \begin{bmatrix} x \\ y \\ 0 \\ w \end{bmatrix} \qquad \mathbf{X_w} = \begin{bmatrix} \mathrm{i}_\pi & \mathrm{j}_\pi & \mathrm{k}_\pi & \mathrm{o}_\pi \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ w \end{bmatrix}$$

$$\mathbf{X_w} = \begin{bmatrix} \mathbf{i}_\pi & \mathbf{j}_\pi & \mathbf{k}_\pi & \mathbf{o}_\pi \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{i}_\pi & \mathbf{j}_\pi & \mathbf{o}_\pi \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{i}_\pi & \mathbf{j}_\pi & \mathbf{o}_\pi \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{\boldsymbol{\pi}}$$

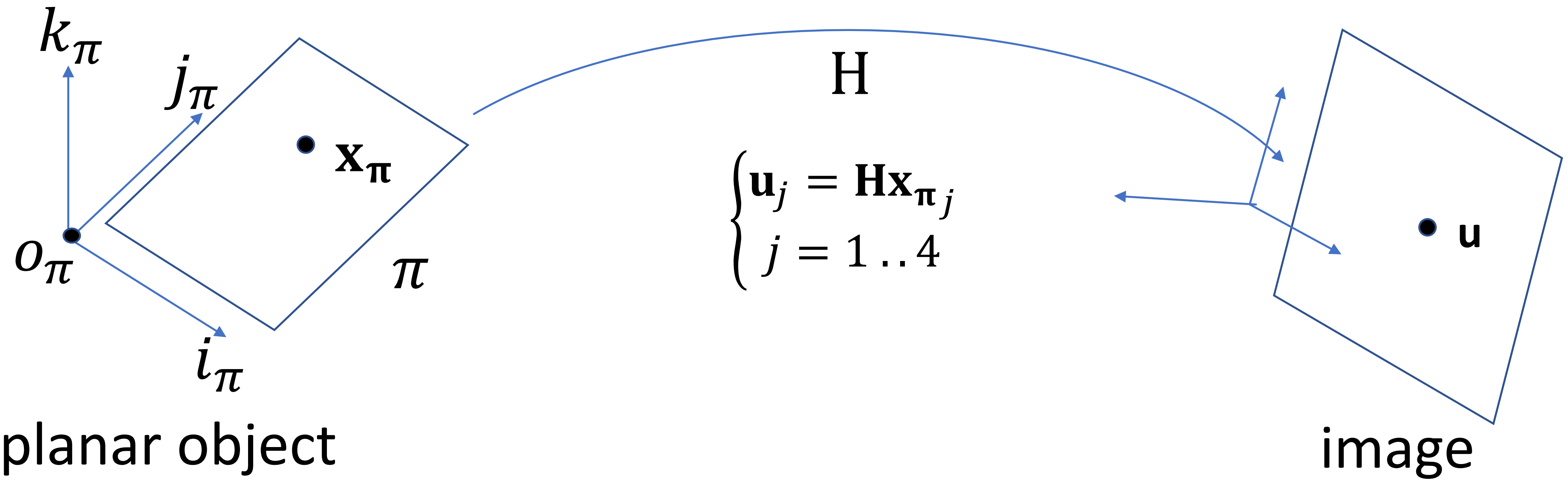
world reference on camera reference: $\mathbf{R} = \mathbf{I}_3, \mathbf{t} = \mathbf{0}$

$$\mathrm{P} = [\mathbf{KR} \quad \mathbf{KRt}] = [\mathbf{K} \quad \mathbf{0}]$$

$$\mathbf{u} = \mathrm{P}\mathbf{X_w} = [\mathbf{K} \quad \mathbf{0}] \begin{bmatrix} \mathbf{i}_\pi & \mathbf{j}_\pi & \mathbf{o}_\pi \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{\boldsymbol{\pi}} = \mathbf{K}[\mathbf{i}_\pi \quad \mathbf{j}_\pi \quad \mathbf{o}_\pi]\mathbf{x}_{\boldsymbol{\pi}}$$

plane π to image homography $\quad \mathbf{H} = \mathbf{K}[\mathbf{i}_\pi \quad \mathbf{j}_\pi \quad \mathbf{o}_\pi]$

H known from image and object model
(4 pairs of corresponding points)



$$\begin{cases} \mathbf{u}_j = \mathbf{H}\mathbf{x}_{\pi j} \\ j = 1 \,..\, 4 \end{cases}$$

pose of the planar object relative to the camera

$$\boxed{[\mathbf{i}_\pi \quad \mathbf{j}_\pi \quad \mathbf{o}_\pi] = \mathbf{K}^{-1}\mathrm{H}} \qquad (\mathbf{k}_\pi = \mathbf{i}_\pi \times \mathbf{j}_\pi)$$

Localization of a reconstructed face wrt camera

Localization from homography H
between the planar face and its image

Localization localization of the upper face

Planar object (upper planar face) wrt camera

$$[i_\pi \quad j_\pi \quad o_\pi] = \mathbf{K}^{-1}\mathbf{H}$$

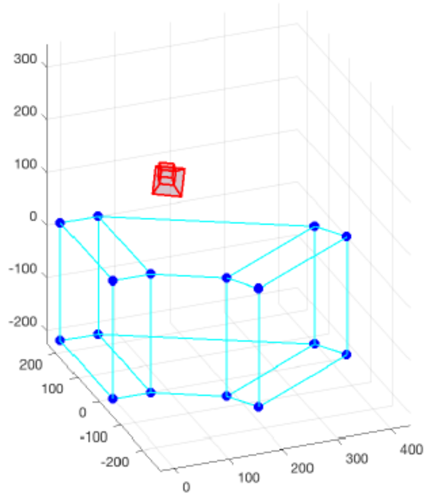
where $\mathbf{K}$ is the calibration matrix of the camera,

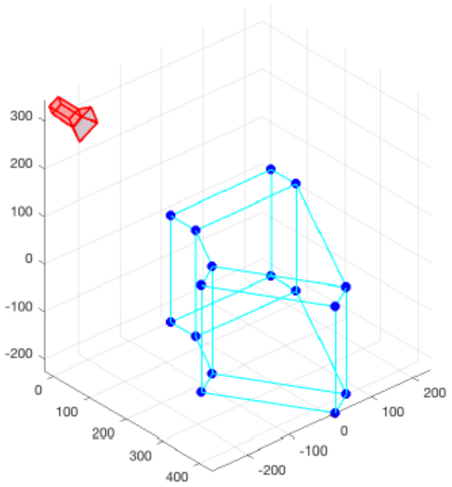
while

$\mathbf{H}$ is the homography from the planar face to its image

Localize camera wrt upper planar face:

$$\begin{bmatrix} i_\pi & j_\pi & i_\pi \times j_\pi & 0_\pi \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1}$$





remember

$$\mathbf{u} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\ \mathbf{X}_{world} = [\mathbf{KR} \quad \mathbf{Kt}]\ \mathbf{X}_{world}$$

$$\mathbf{u} = \mathbf{P}\ \mathbf{X}_{world} = [\mathbf{M} \quad \mathbf{m}]\ \mathbf{X}_{world}$$

$\rightarrow$ $\mathbf{M} = \mathbf{KR}$ and $\mathbf{m} = \mathbf{Kt}$

$$\mathbf{M} = \mathbf{KR}_{cam->world}$$

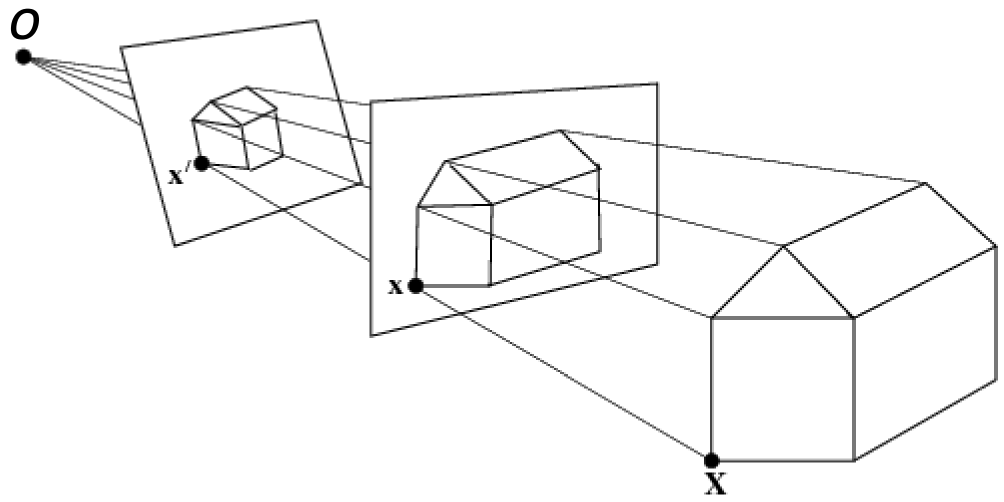
given **M**, find **K** and **R** by Q-R matrix decomposition of inv(**M**)

from $\mathbf{m} = -\mathbf{Mo}$ $\rightarrow$ $\mathbf{u} = [\mathbf{M} \quad \mathbf{m}]\ \mathbf{X}_{world} = \mathbf{M}[\mathbf{I} \quad -\boldsymbol{o}]\ \mathbf{X}_{world}$

$$\mathbf{u} = \mathbf{KR}[\mathbf{I} \quad -\boldsymbol{o}]\ \mathbf{X}_{world}$$

$$\rightarrow \boxed{\mathbf{P} = \mathbf{KR}[\mathbf{I} \quad -\boldsymbol{o}]}$$

Image transformations with fixed camera center



$$P = KR[I| - o] \ , P' = K'R'[I| - o]$$

$$P' = K'R'(KR)^{-1}P$$

$$x' = P'X = K'R'(KR)^{-1}PX = K'R'(KR)^{-1}x$$

$$x' = Hx \text{ with } H = K'R'(KR)^{-1} = K'R'R^{-1}K^{-1}$$

Translating the image plane: zooming (assuming fixed principal point)

$$\mathrm{x}' = \mathrm{Hx} = \mathrm{K}'\mathrm{R}'\mathrm{R}^{-1}\mathrm{K}^{-1}\mathrm{x} \text{ but, since } \mathrm{R}' = \mathrm{R}$$

$$\mathrm{x}' = \mathrm{K}'(\mathrm{K})^{-1}\mathrm{x}$$

$$\mathrm{H} = \mathrm{K}'(\mathrm{K})^{-1} = \begin{bmatrix} \mathrm{kI} & (1-\mathrm{k})\tilde{\mathrm{x}}_0 \\ 0^{\mathrm{T}} & 1 \end{bmatrix} \qquad k = f / f'$$

$$\mathrm{K}' = \begin{bmatrix} \mathrm{kI} & (1-\mathrm{k})\tilde{\mathrm{x}}_0 \\ 0^{\mathrm{T}} & 1 \end{bmatrix} \mathrm{K}$$

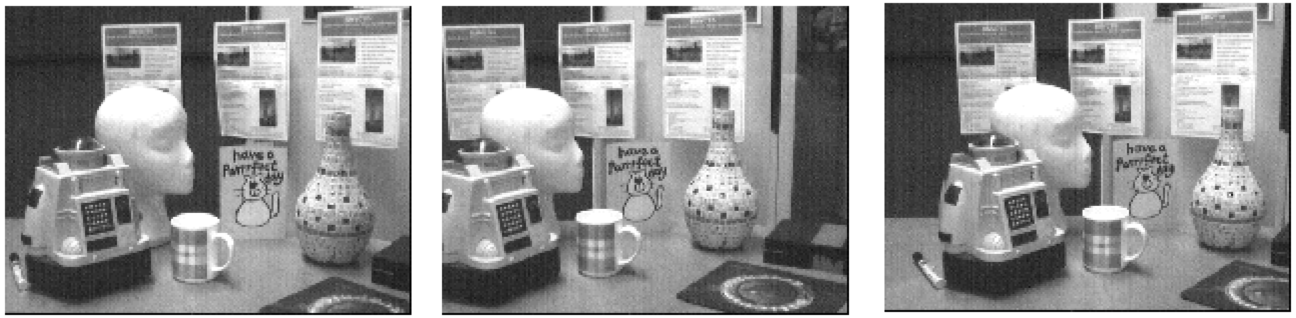
$$\rightarrow$$

$$\mathrm{K}' = \begin{bmatrix} \mathrm{kI} & (1-\mathrm{k})\tilde{\mathrm{x}}_0 \\ 0^{\mathrm{T}} & 1 \end{bmatrix} \begin{bmatrix} \mathrm{A} & \tilde{\mathrm{x}}_0 \\ 0^{\mathrm{T}} & 1 \end{bmatrix} = \begin{bmatrix} \mathrm{kA} & \tilde{\mathrm{x}}_0 \\ 0^{\mathrm{T}} & 1 \end{bmatrix} = \mathrm{K} \begin{bmatrix} \mathrm{kI} & 0 \\ 0^{\mathrm{T}} & 1 \end{bmatrix}$$

Camera rotation around camera center

$$\mathrm{x} = \mathrm{K}\,[\mathrm{I}|0]\,\mathrm{X}$$
$$\mathrm{x}' = \mathrm{K}'[\mathrm{R}|0]\,\mathrm{X} = \mathrm{K}'\mathrm{R}\,[\mathrm{I}|0]\,\mathrm{X} = \mathrm{K}'\,\mathrm{R}\mathrm{K}^{-1}\mathrm{K}[\mathrm{I}|0]\,\mathrm{X} = \mathrm{K}'\mathrm{R}\mathrm{K}^{-1}\mathrm{x}$$

$$\mathrm{H} = \mathrm{K}'\mathrm{R}\mathrm{K}^{-1}$$



Camera rotation around camera center with fixed intrinsic parameters

A camera is rotated about its centre with no change in the internal parameters..
Algebraically, if x, x' are the images of a point X before and after the pure rotation

$$\mathrm{x} = \mathrm{K}\,[\mathrm{I} \mid 0]\,\mathrm{X}$$
$$\mathrm{x}' = \mathrm{K}\,[\mathrm{R} \mid 0]\,\mathrm{X} = \mathrm{KR}\,[\mathrm{I} \mid 0\,]X = \mathrm{KRK}^{-1}\,\mathrm{K}[\mathrm{I} \mid 0]\mathrm{X} = \mathrm{K\,R\,K}^{-1}\mathrm{x}$$

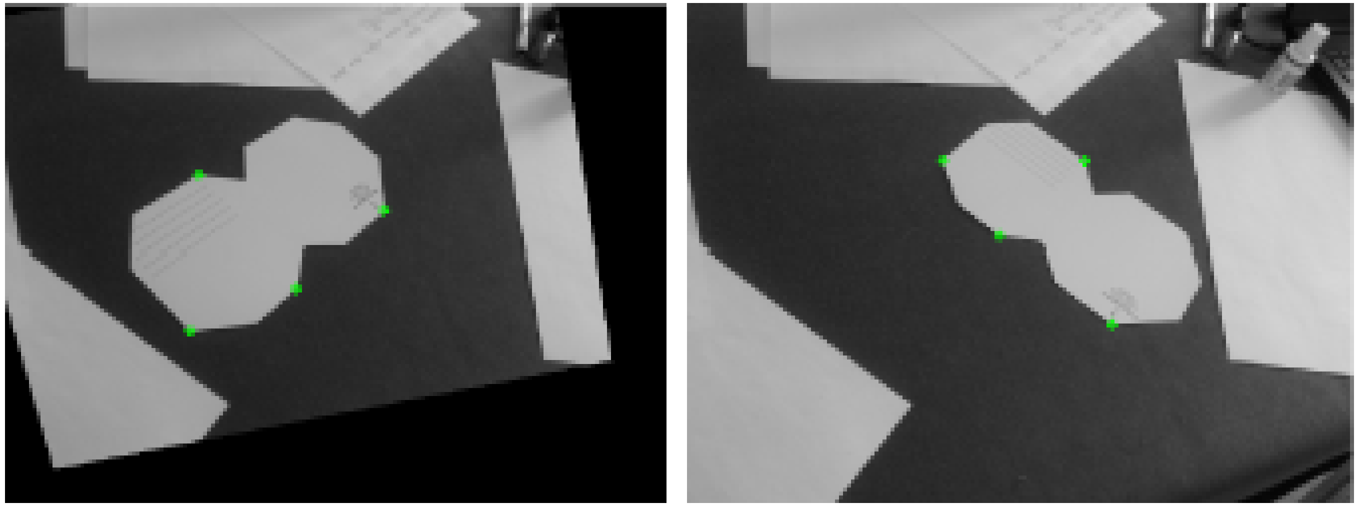
so that $\mathrm{x}' = H\,\mathrm{x}$ with $H = \mathrm{KRK}^{-1}$: conjugate rotation

eigenvalues of $H = \mathrm{KRK}^{-1}$: $\left\{\mu, \mu e^{i\theta}, \mu e^{-i\theta}\right\}$ = the same eigenvalues of R

where μ is an unknown scale factor (if H scaled such that $\det H = 1$, then μ = 1).
Hence, the rotation angle between views may be computed directly from the phase of the complex eigenvalues of H.
Similarly, it can be shown (exercise) that the eigenvector of H corresponding to the real eigenvalue is the vanishing point of the rotation axis.

Reminds to planar motion

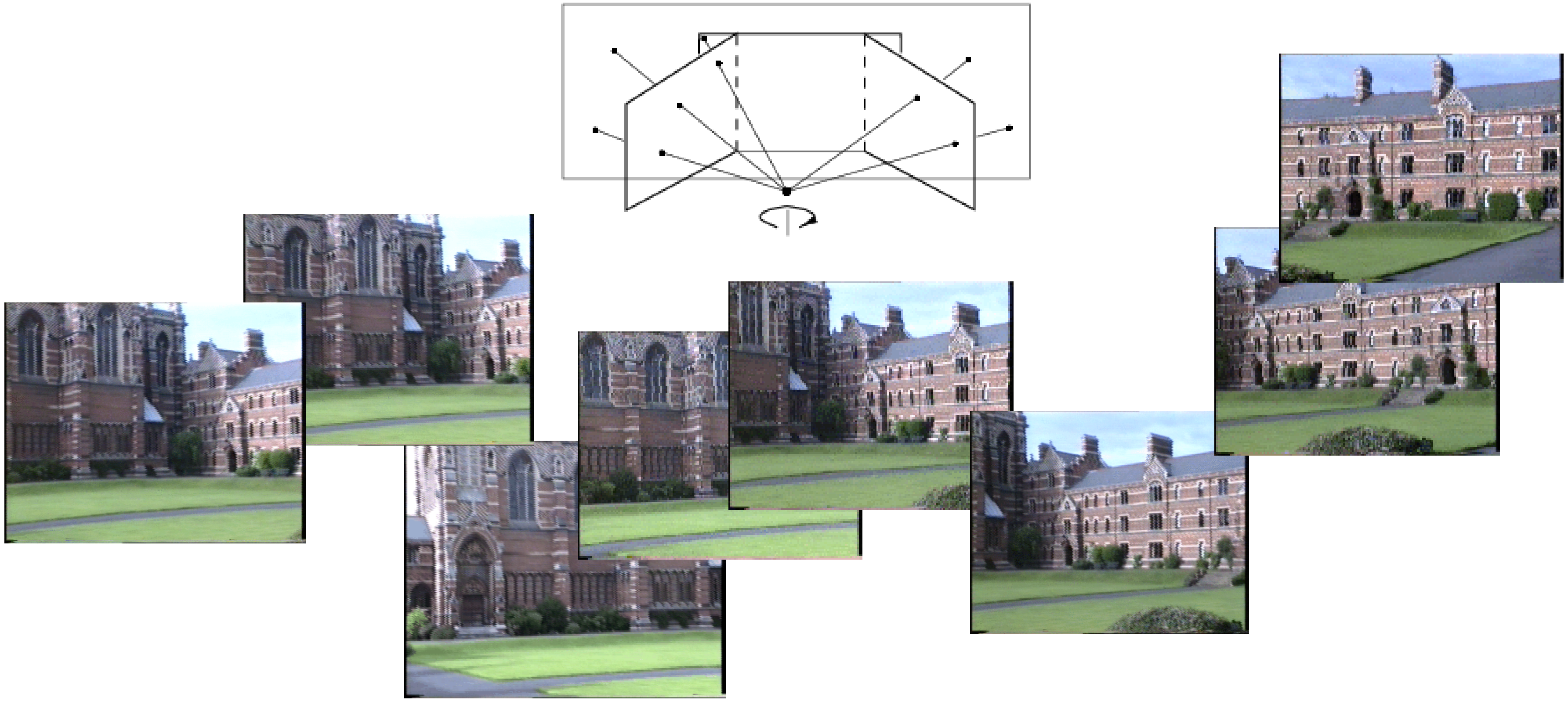
- find eigenvector-eigenvalues of H :
- eigenvalues are proportional to $\lambda' = 1, \lambda'' = e^{i\theta}, \lambda'' = e^{-i\theta}$
- eigenvector e' associated to $\lambda' = 1$ is the image of the C.O.R. $\boldsymbol{O}$
- angle θ is the rotation angle

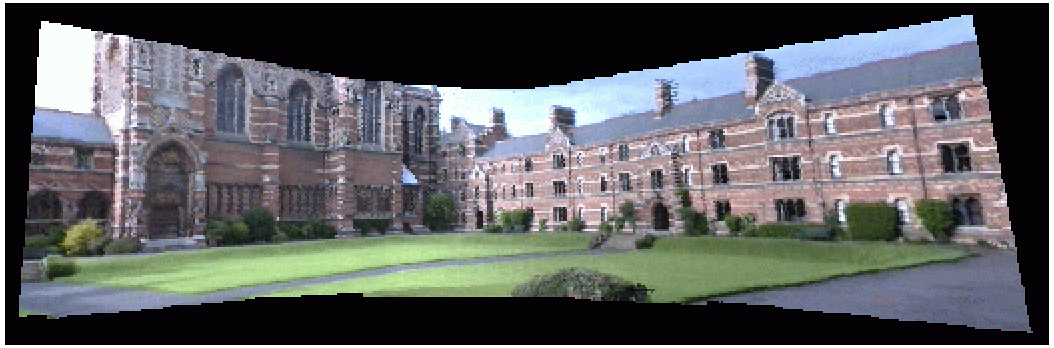


Common aspects and differences between them

- in planar motion, rotation does not need to be about the camera center
- in planar motion, rotation axis is perpendicular to the plane

Planar homography mosaicing





Planar homography mosaicing

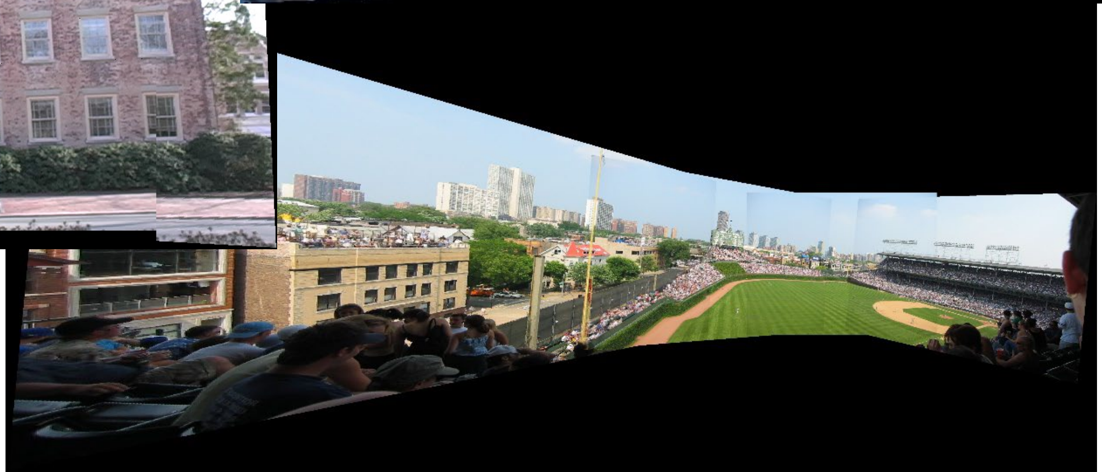
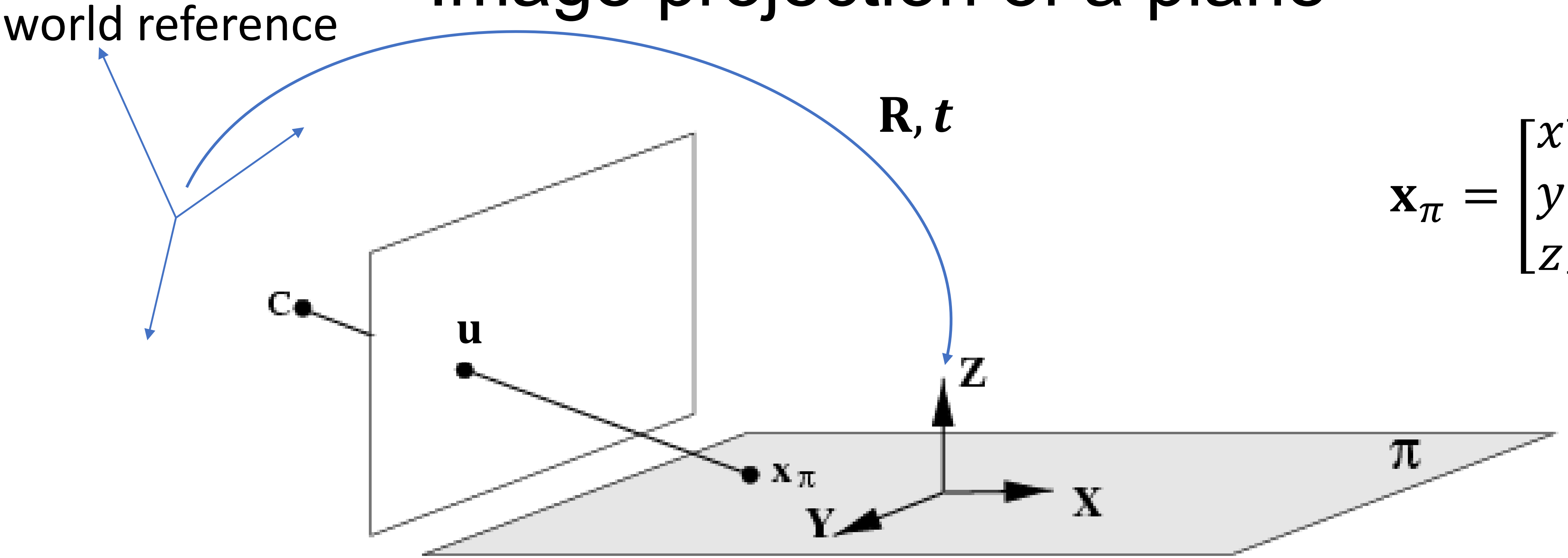


Image: projection and back-projection

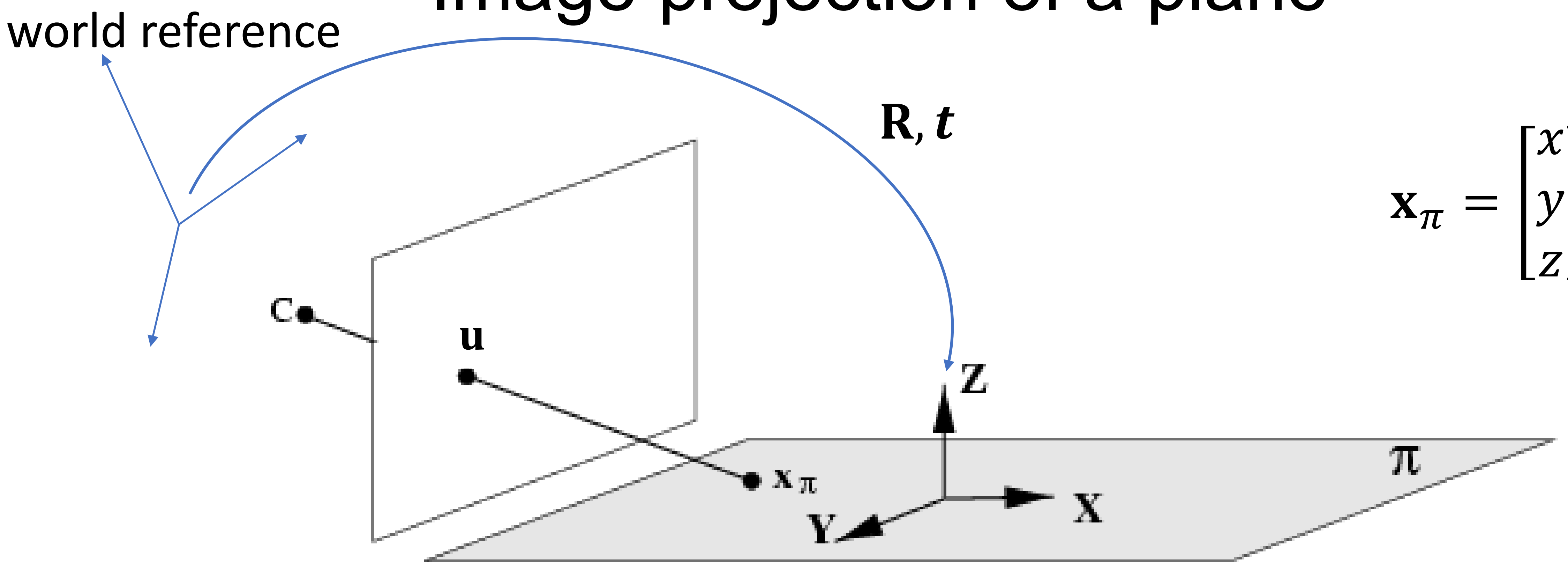
Image projection of a plane



$$\mathbf{x}_\pi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$X_w = \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ 0 & 1 \end{bmatrix} X_\pi$$

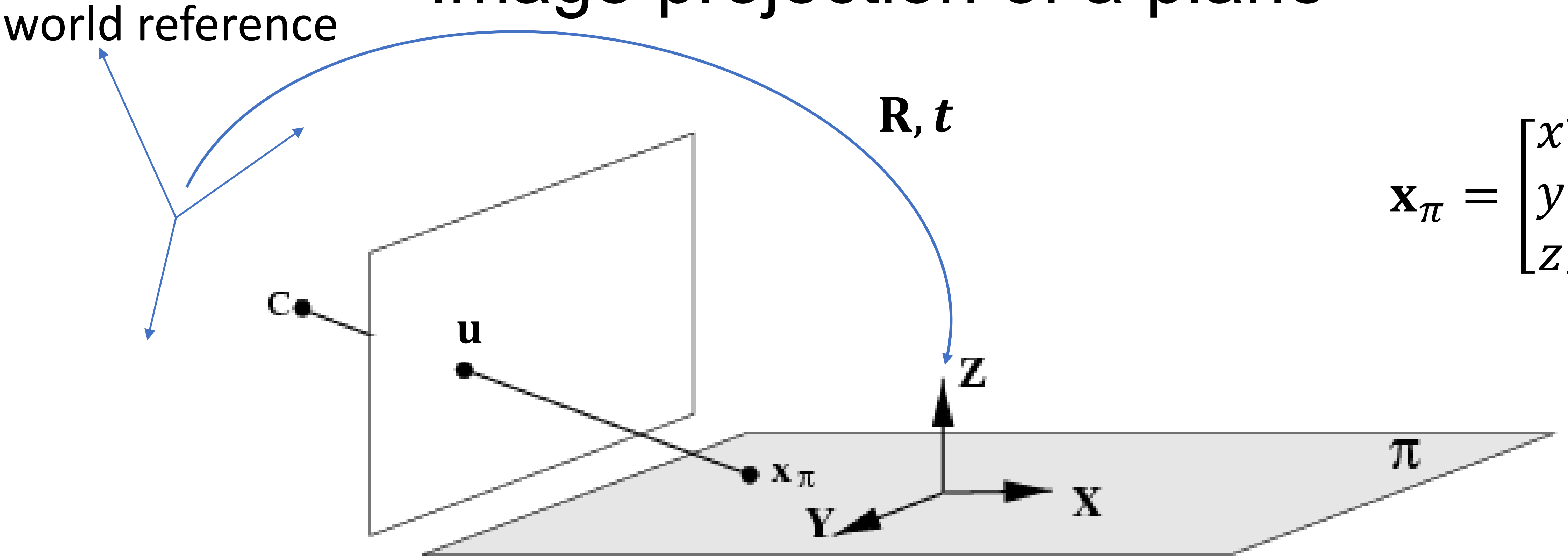
Image projection of a plane



$$\mathbf{x}_\pi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\mathbf{u} = \mathbf{P}X_w = [\mathbf{M} \quad \mathbf{m}]X_w = [\mathbf{M} \quad \mathbf{m}]\begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ 0 & 1 \end{bmatrix} X_\pi = [\mathbf{MR} \quad \mathbf{Mt} + \mathbf{m}]X_\pi$$

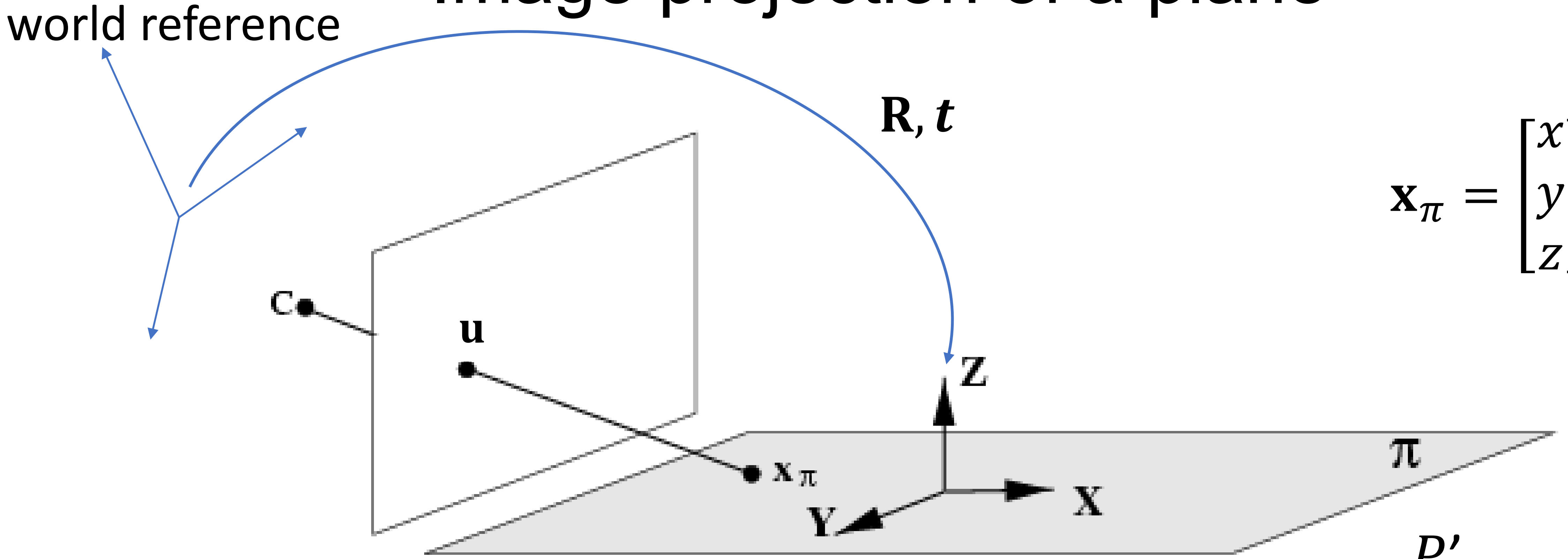
Image projection of a plane



$$\mathbf{x}_\pi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\mathbf{u} = \mathbf{P}X_w = [\mathbf{M} \quad \mathbf{m}]X_w = [\mathbf{M} \quad \mathbf{m}]\begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ 0 & 1 \end{bmatrix} X_\pi = \underbrace{[\mathbf{MR} \quad \mathbf{Mt} + \mathbf{m}]}_{\mathbf{P}'} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ \mathbf{0} \\ \boldsymbol{w} \end{bmatrix}$$

Image projection of a plane



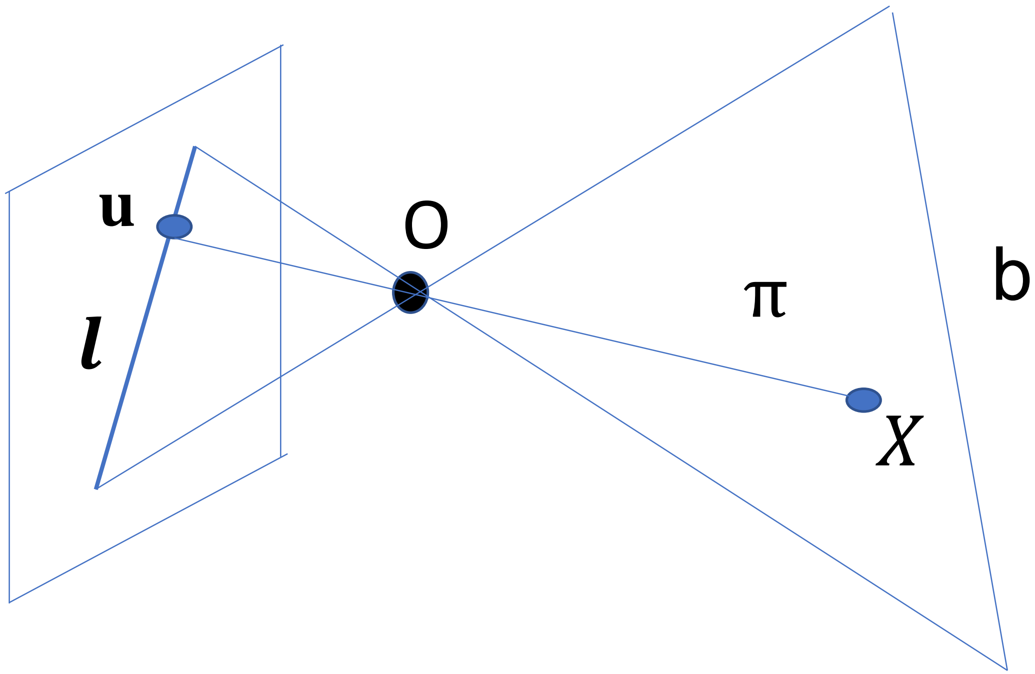
$$\mathbf{x}_\pi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\mathbf{u} = PX_w = [\mathbf{M} \quad \mathbf{m}]X_w = [\mathbf{M} \quad \mathbf{m}]\begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ 0 & 1 \end{bmatrix} X_\pi = \overbrace{[\mathbf{MR} \quad \mathbf{Mt} + \mathbf{m}]}^{P'} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ \mathbf{0} \\ \boldsymbol{w} \end{bmatrix}$$

$$= [\mathrm{p}'_{\mathbf{1}} \quad \mathrm{p}'_{\mathbf{2}} \quad \mathrm{p}'_{\mathbf{3}} \quad \mathrm{p}'_{\mathbf{4}}][\boldsymbol{x} \quad \boldsymbol{y} \quad \mathbf{0} \quad \boldsymbol{w}]^T = [\mathrm{p}'_{\mathbf{1}} \quad \mathrm{p}'_{\mathbf{2}} \quad \mathrm{p}'_{\mathbf{4}}]\mathbf{x}_{\boldsymbol{\pi}}$$

Back-projection of an image line

set of space points X, whose image projection $\mathbf{u} = \mathbf{P}X$ is on image line $\boldsymbol{l}$



back-projection

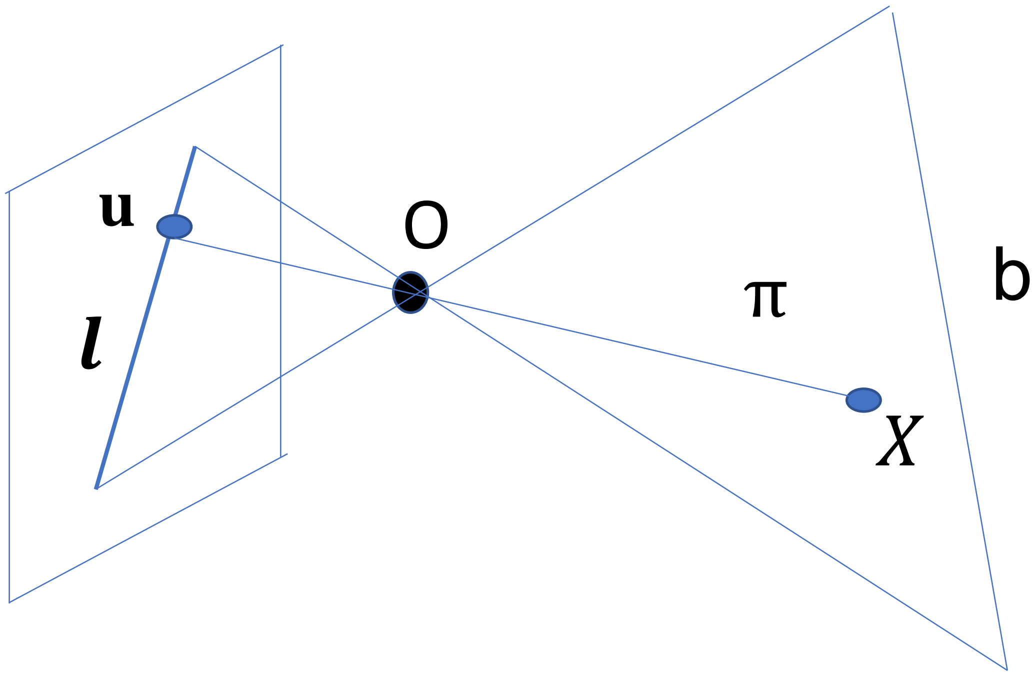
$$l^{\mathrm{T}}\mathbf{u} = \underline{l^{\mathrm{T}}\mathbf{P}X} = 0$$

$$\boldsymbol{\pi}^T X = 0$$

back-projection of $\boldsymbol{l}$ is plane $\boldsymbol{\pi} = \mathbf{P}^{\mathrm{T}}\boldsymbol{l}$

Back-projection of an image line

set of space points X, whose image projection $\mathbf{u} = \mathbf{P}X$ is on image line $\boldsymbol{l}$



back-projection

$$l^{\mathrm{T}}\mathbf{u} = l^{\mathrm{T}}\mathbf{P}X = 0$$

$$\boldsymbol{\pi}^T \mathrm{X} = 0$$

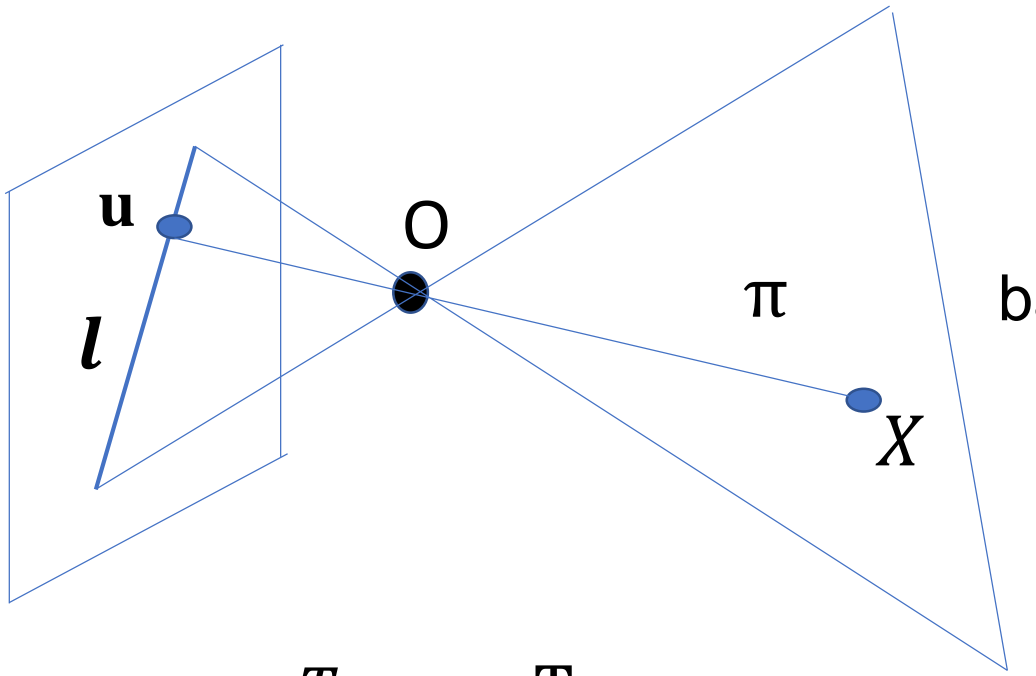
back-projection of $\boldsymbol{l}$ is plane $\boldsymbol{\pi} = \mathbf{P}^{\mathrm{T}}\boldsymbol{l}$

through O, in fact, since $\mathrm{O} = RNS(\mathbf{P})$

$$\boldsymbol{\pi}^T \mathrm{O} = \boldsymbol{l}^{\mathrm{T}}\mathbf{P}\,\mathrm{O} = \boldsymbol{l}^{\mathrm{T}}0 = 0$$

Back-projection of an image line

set of space points X, whose image projection is on image line $\boldsymbol{l}$



back-projection of $\boldsymbol{l}$ is $\boldsymbol{\pi} = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{l}$

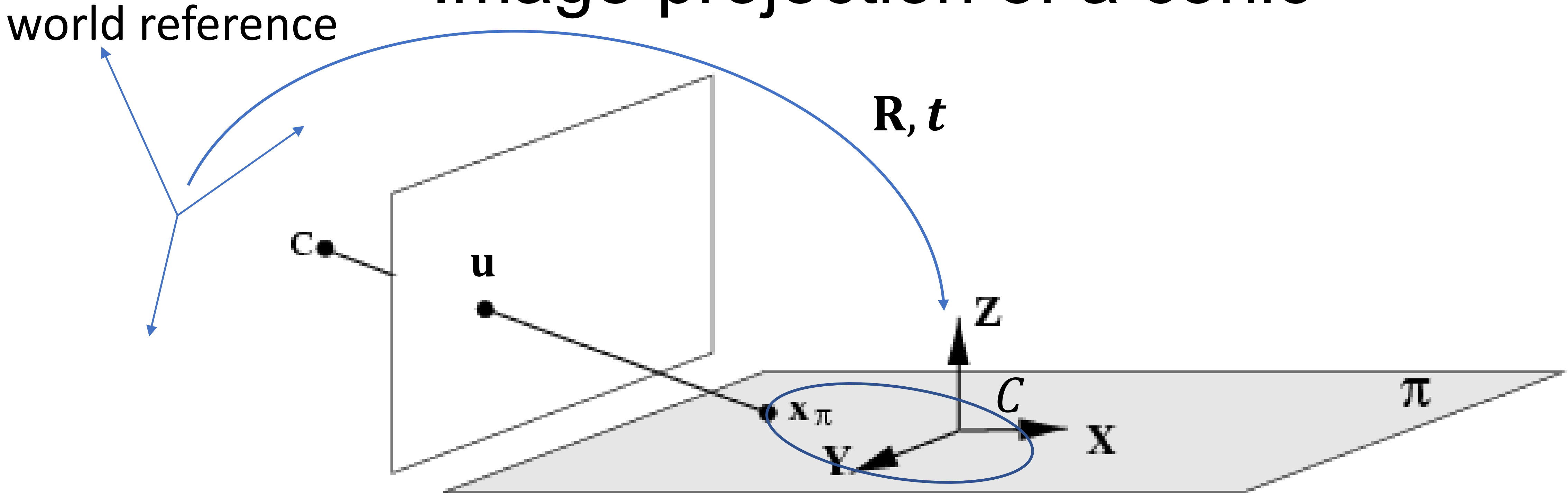
$$\boldsymbol{\pi}^{T}\mathrm{X} = l^{\mathrm{T}} PX = 0$$

try with $X = O = RNS(P)$:

$$\boldsymbol{\pi}^{T}\mathrm{O} = l^{\mathrm{T}} P\mathrm{O} = l^{\mathrm{T}}0 = 0$$

The backprojection of $\boldsymbol{l}$ is a plane $\boldsymbol{\pi} = \boldsymbol{P}^{\mathrm{T}}\boldsymbol{l}$ through O

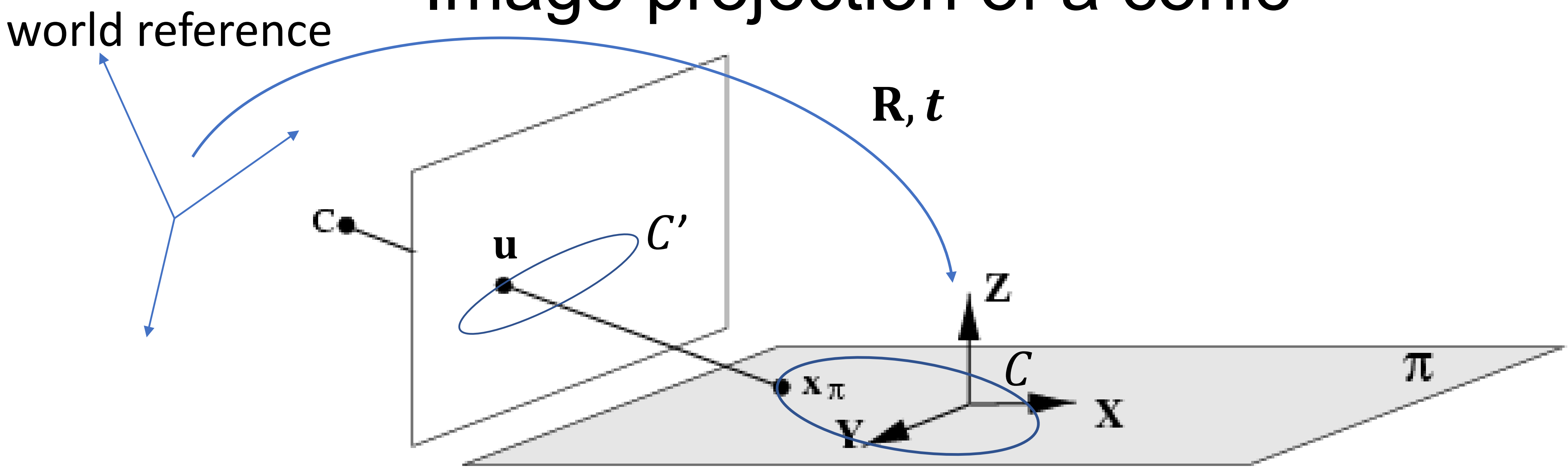
Image projection of a conic



$$\mathbf{u} = PX_w = [\mathbf{M} \quad \mathbf{m}]X_w = [\mathbf{M} \quad \mathbf{m}]\begin{bmatrix}\mathbf{R} & \boldsymbol{t} \\ 0 & 1\end{bmatrix}X_\pi = \underbrace{[\mathbf{MR} \quad \mathbf{Mt}+\mathbf{m}]}_{P'}\begin{bmatrix}\boldsymbol{x} \\ \boldsymbol{y} \\ \mathbf{0} \\ \boldsymbol{w}\end{bmatrix}$$

$$= [\mathrm{p'}_\mathbf{1} \quad \mathrm{p'}_\mathbf{2} \quad \mathrm{p'}_\mathbf{3} \quad \mathrm{p'}_\mathbf{4}][\boldsymbol{x} \quad \boldsymbol{y} \quad \mathbf{0} \quad \boldsymbol{w}]^{\boldsymbol{T}} = [\mathrm{p'}_\mathbf{1} \quad \mathrm{p'}_\mathbf{2} \quad \mathrm{p'}_\mathbf{4}]\mathbf{x}_{\boldsymbol{\pi}} = P'_{\boldsymbol{\pi}}\mathbf{x}_{\boldsymbol{\pi}}$$

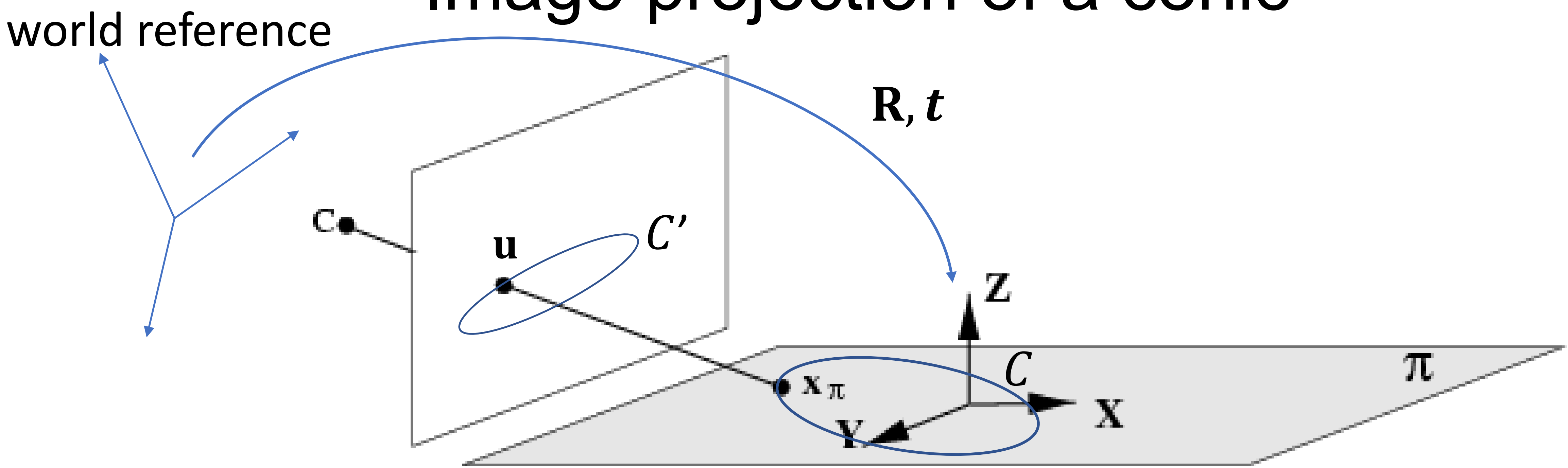
Image projection of a conic



$$\mathbf{x}_\pi{}^{T}\boldsymbol{C}\ \mathbf{x}_\pi = 0$$

$$\mathbf{u} = [\mathrm{p}'_{\mathbf{1}} \quad \mathrm{p}'_{\mathbf{2}} \quad \mathrm{p}'_{\mathbf{4}}]\ \mathbf{x}_\pi = P'_\pi\,\mathbf{x}_\pi \rightarrow\ \mathbf{x}_\pi = P'_\pi{}^{-\mathbf{1}}\mathbf{u}$$

Image projection of a conic



$$\mathbf{x}_\pi{}^T \boldsymbol{C}\ \mathbf{x}_\pi = 0$$

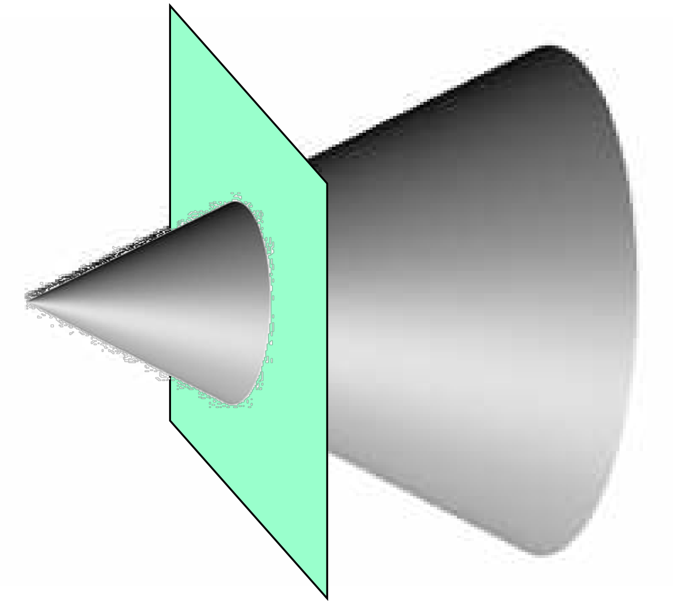
$$\mathbf{u} = [\mathrm{p'}_1 \quad \mathrm{p'}_2 \quad \mathrm{p'}_4]\ \mathbf{x}_\pi = P'_\pi\, \mathbf{x}_\pi \rightarrow\ \mathbf{x}_\pi = P'_\pi{}^{-1}\mathbf{u}$$

$$\rightarrow$$

$$\mathbf{u}^T P'_\pi{}^{-T} \boldsymbol{C}\, P'_\pi{}^{-1}\mathbf{u} = 0$$

$$\mathbf{u}^T \boldsymbol{C}'\, \mathbf{u} = 0 \text{ with } \boldsymbol{C}' = P'_\pi{}^{-T} \boldsymbol{C}\, P'_\pi{}^{-1}$$

Back-projection of an image conic



$$\mathrm{u} = PX$$

$$\mathrm{u}^{\mathrm{T}}\mathrm{C}\mathrm{u} = X^{\mathrm{T}}P^{\mathrm{T}}\mathrm{C}PX = X^{\mathrm{T}}\mathrm{Q}_{\mathrm{co}}X = 0$$

back-projection of an image conic is a rank-3 quadric,
→ a **cone**

$$\mathrm{Q}_{\mathrm{co}} = P^{\mathrm{T}}\mathrm{C}\,P$$

example:

$$\mathrm{Q}_{\mathrm{co}} = \begin{bmatrix} \mathrm{K}^{\mathrm{T}} \\ 0 \end{bmatrix} \mathrm{C}^{\mathrm{T}} [\mathrm{K} \mid 0] = \begin{bmatrix} \mathrm{K}^{\mathrm{T}}\mathrm{C}\mathrm{K} & 0 \\ 0 & 0 \end{bmatrix}$$

Apparent contour of a smooth surface

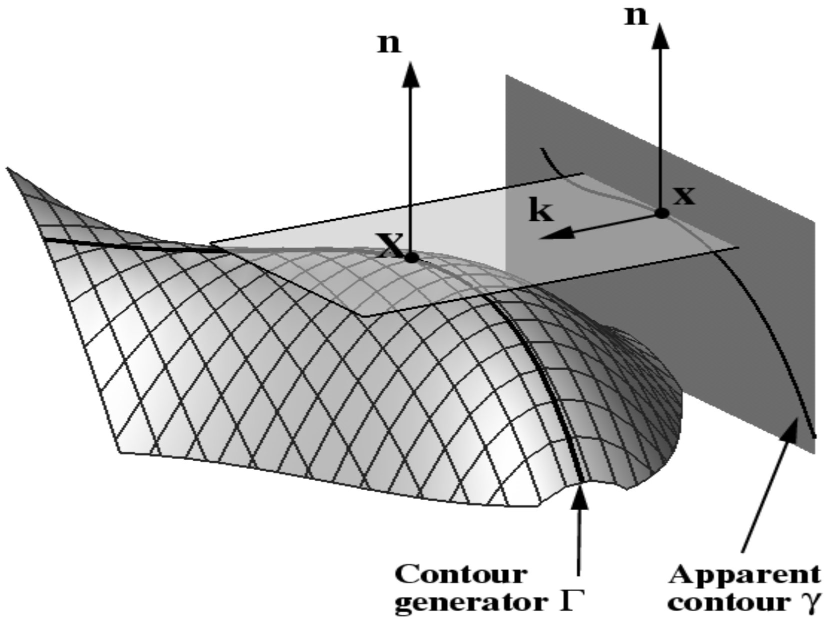
Image projection of smooth surfaces

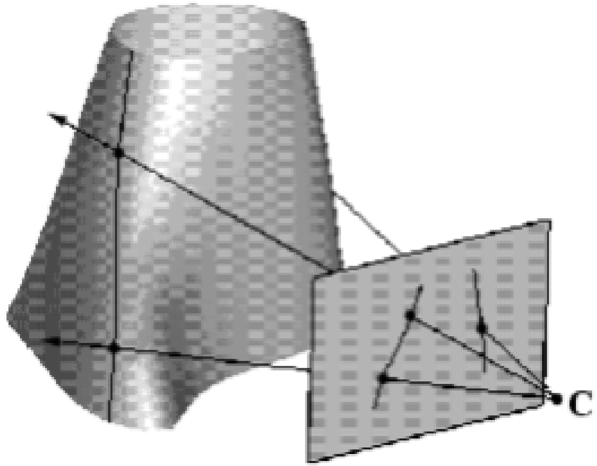
The **contour generator** Γ is the set of points X on a surface S, whose viewing rays are tangent to S.

The corresponding **apparent contour** is the image of Γ

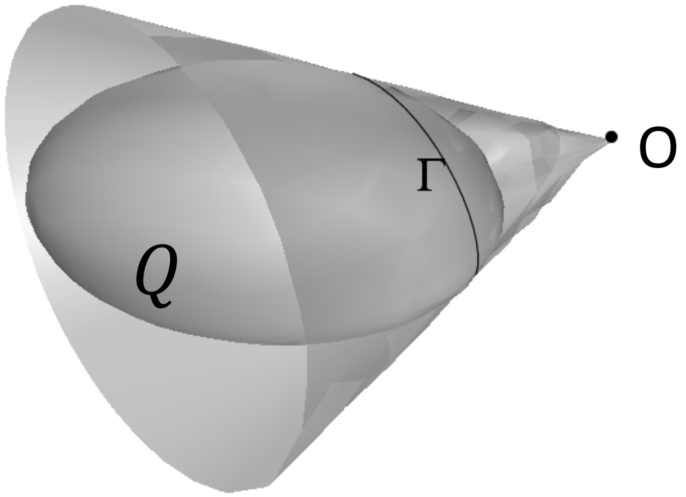
Γ depends only on position of projection center O,

Γ depends also on rest of the projection matrix P



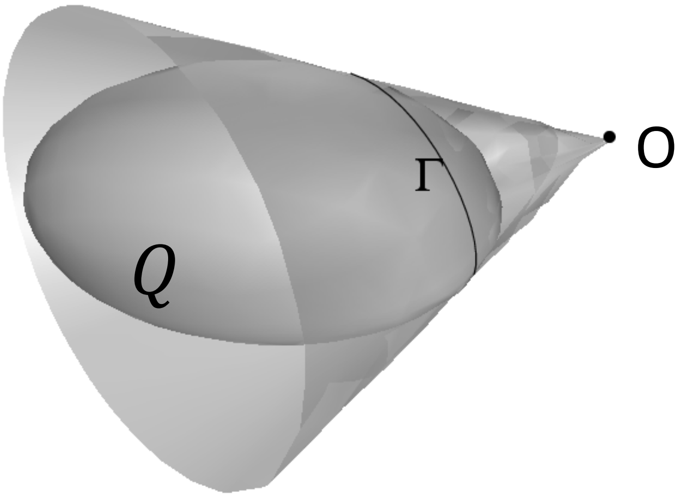


The apparent contour of a quadric



select those planes, tangent to the quadric Q, which go through camera center O

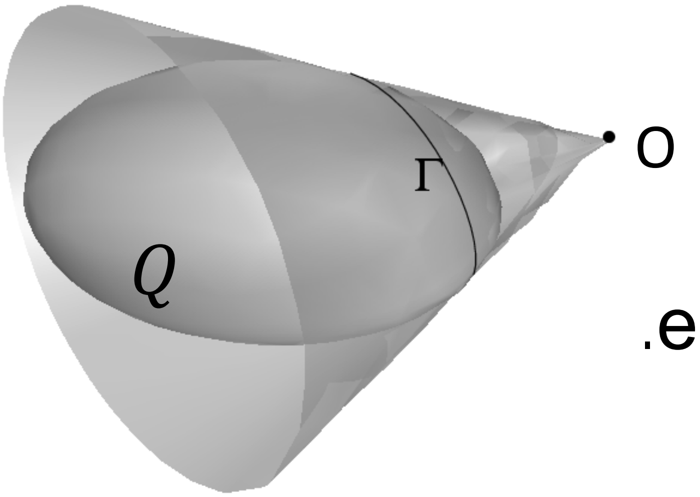
The apparent contour of a quadric



select those planes, tangent to the quadric Q,
i.e. belonging to dual quadric $Q^* = Q^{-1}$
which go through camera center O

$$\Pi^T Q^* \Pi = 0$$

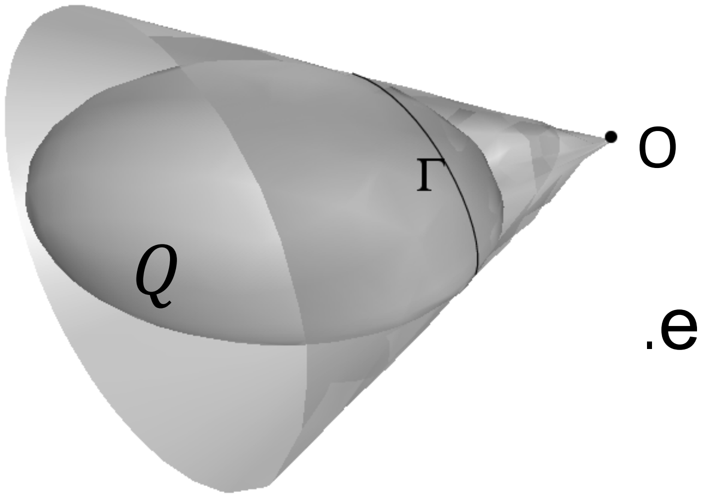
The apparent contour of a quadric



select those planes, tangent to the quadric Q,
i.e. belonging to dual quadric $Q^* = Q^{-1}$
which go through camera center O
.e., that are backprojections $\mathrm{P}^{\mathrm{T}}\boldsymbol{l}$ of some image lines $\boldsymbol{l}$

$$\Pi^{\mathrm{T}}\mathrm{Q}^*\Pi = l^{\mathrm{T}}\mathrm{P}\mathrm{Q}^*\mathrm{P}^{\mathrm{T}}l = 0$$

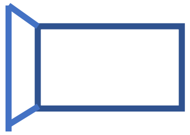
The apparent contour of a quadric



select those planes, tangent to the quadric Q,
i.e. belonging to dual quadric $Q^* = Q^{-1}$
which go through camera center O
.e., that are backprojections $\mathrm{P}^{\mathrm{T}}\boldsymbol{l}$ of some image lines $\boldsymbol{l}$

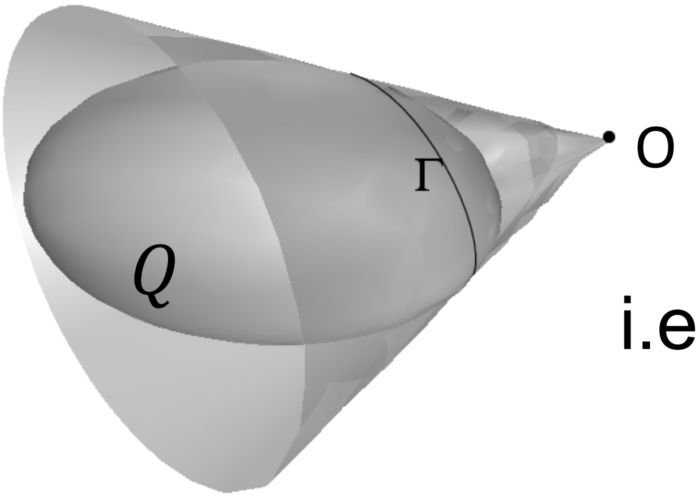
$$\Pi^{\mathrm{T}}\mathrm{Q}^*\Pi = l^{\mathrm{T}}\mathrm{P}\mathrm{Q}^*\mathrm{P}^{\mathrm{T}}l = 0$$

these planes Π are tangent
to the contour generator Γ



these lines $\boldsymbol{l}$ are image of planes Π
→ $\boldsymbol{l}$ are tangent to image of Γ
→ $\boldsymbol{l}$ are tangent to apparent contour γ

The apparent contour of a quadric

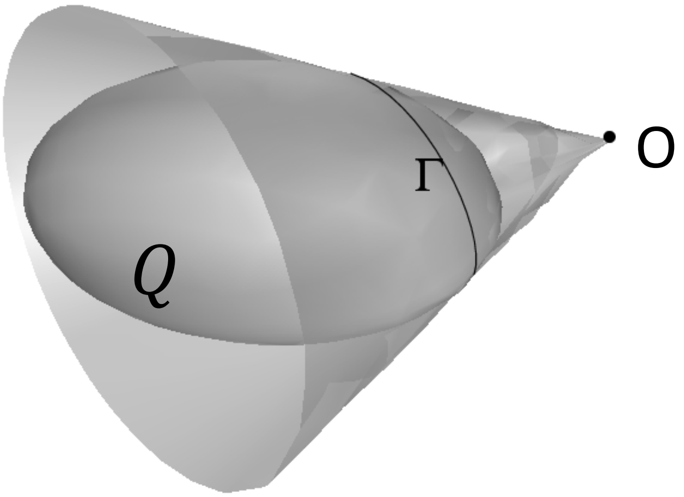


select those planes, tangent to the quadric Q,
i.e. belonging to dual quadric $Q^* = Q^{-1}$
which go through camera center O
i.e., that are backprojections $\mathrm{P}^{\mathrm{T}}\boldsymbol{l}$ of some image lines $\boldsymbol{l}$

$$\Pi^{\mathrm{T}}\mathrm{Q}^*\Pi = l^{\mathrm{T}}\mathrm{P}\mathrm{Q}^*\mathrm{P}^{\mathrm{T}}l = 0$$

these image lines $\boldsymbol{l}$ satisfy a quadratic equation
→ they belong to a dual conic $\mathrm{C}^* = \mathrm{P}\mathrm{Q}^*\mathrm{P}^T$

The apparent contour of a quadric

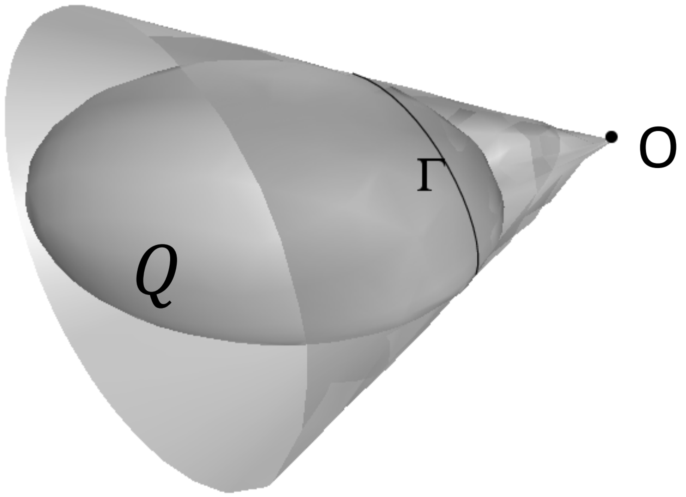


select those planes, tangent to the quadric Q,
i.e. belonging to dual quadric $Q^* = Q^{-1}$
which go through camera center O
i.e., that are backprojections of some image lines $\boldsymbol{l}$

$$\Pi^{\mathrm{T}}\mathrm{Q}^*\Pi = l^{\mathrm{T}}\mathrm{P}\mathrm{Q}^*\mathrm{P}^{\mathrm{T}}l = 0$$

these image lines $\boldsymbol{l}$ satisfy a quadratic equation
→ they belong to a dual conic $\mathrm{C}^* = \mathrm{P}\mathrm{Q}^*\mathrm{P}^T$
→ they are tangent to a conic $\mathrm{C} = \mathrm{C^*}^{-1} = (\mathrm{P}\mathrm{Q}^*\mathrm{P}^T)^{-1}$

The apparent contour of a quadric



select those planes, tangent to the quadric Q,
i.e. belonging to dual quadric $Q^* = Q^{-1}$
which go through camera center O
i.e., that are backprojections of some image lines $\boldsymbol{l}$

$$\Pi^{\mathrm{T}} \mathrm{Q}^* \Pi = l^{\mathrm{T}} \mathrm{P} \mathrm{Q}^* \mathrm{P}^{\mathrm{T}} l = 0$$

these image lines $\boldsymbol{l}$ satisfy a quadratic equation
→ they belong to a dual conic $\mathrm{C}^* = \mathrm{PQ}^*\mathrm{P}^T$
→ they are tangent to a conic $\mathrm{C} = \mathrm{C}^{*^{-1}} = (\mathrm{PQ}^*\mathrm{P}^T)^{-1}$

But $\boldsymbol{l}$ are tangent to conic $\mathrm{C} = \mathrm{C}^{*^{-1}} = (\mathrm{PQ}^*\mathrm{P}^T)^{-1}$ → $\boxed{\text{a.c } \gamma \text{ is the conic C}}$

Exercise: apparent contour of a cone?

Exercise: apparent contour of a cone?

- contour generator Γ: set of tangency points from O

→ two straight lines through the vertex V

- apparent contour γ : image of Γ

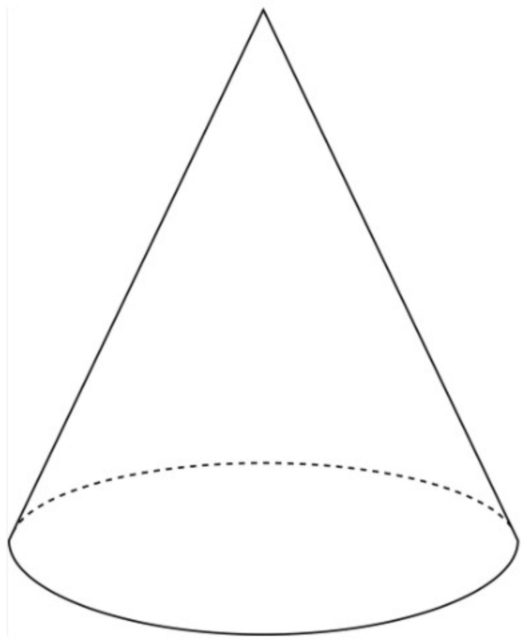
→ two image lines: i.e, a degenerate conic

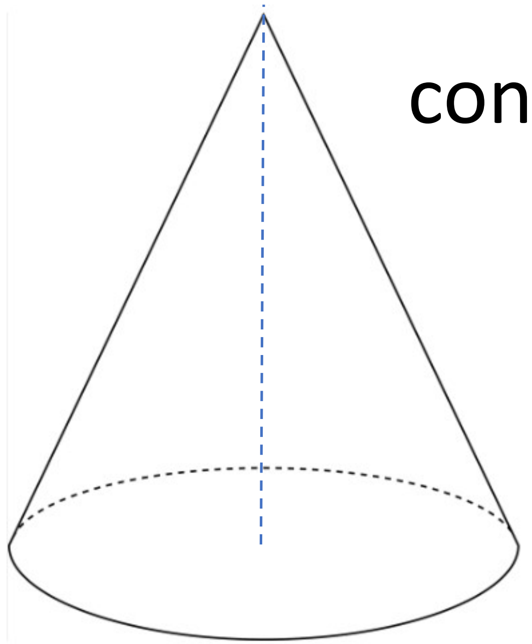
Example: contour generator of a **right** cone?

- contour generator Γ: set of tangency points from O
→ two straight lines through the vertex V

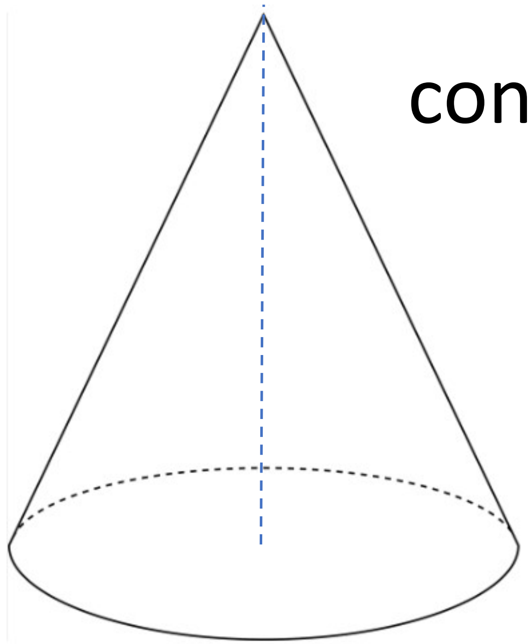
- **right** cone alone is **symmetric** wrt to its axis
→ right cone + viewpoint O : «less» symmetry,
which symmetry?

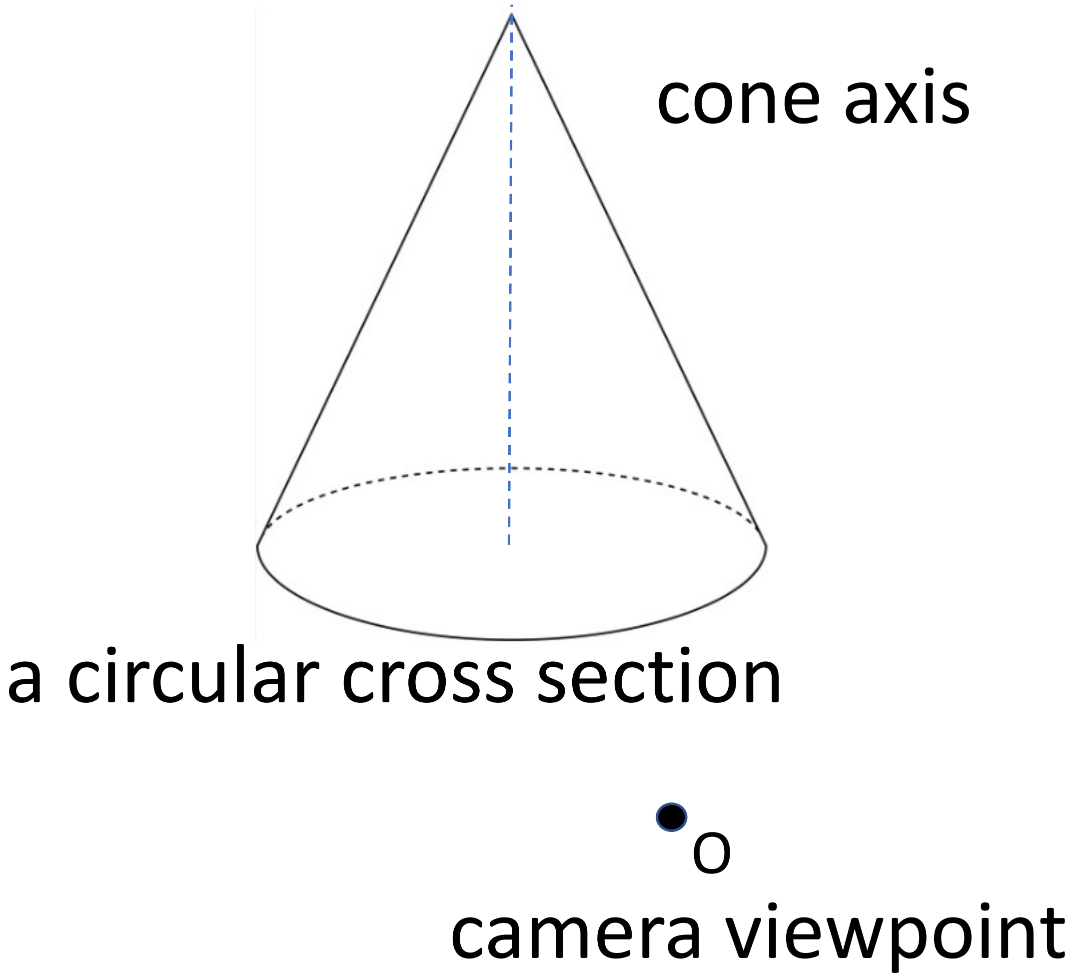
a right cone

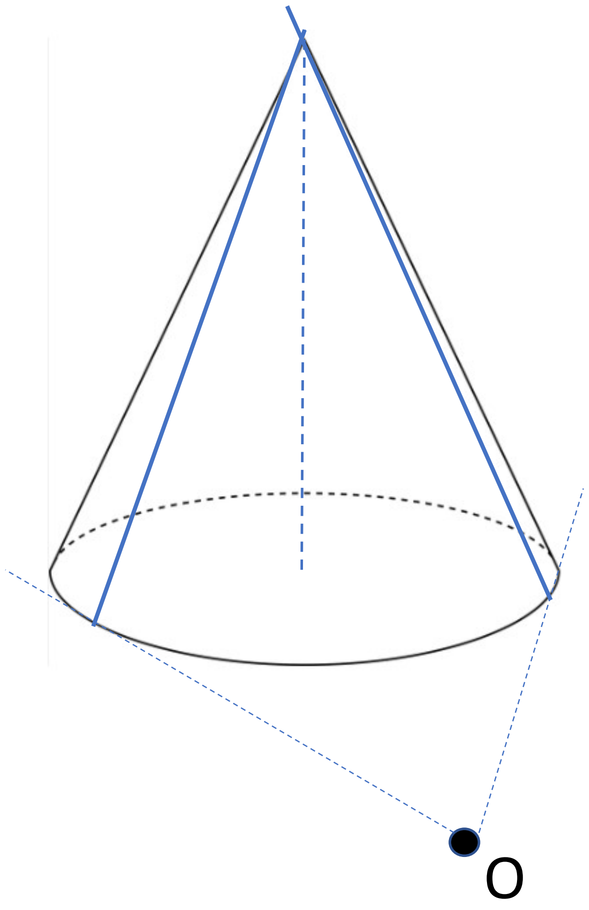




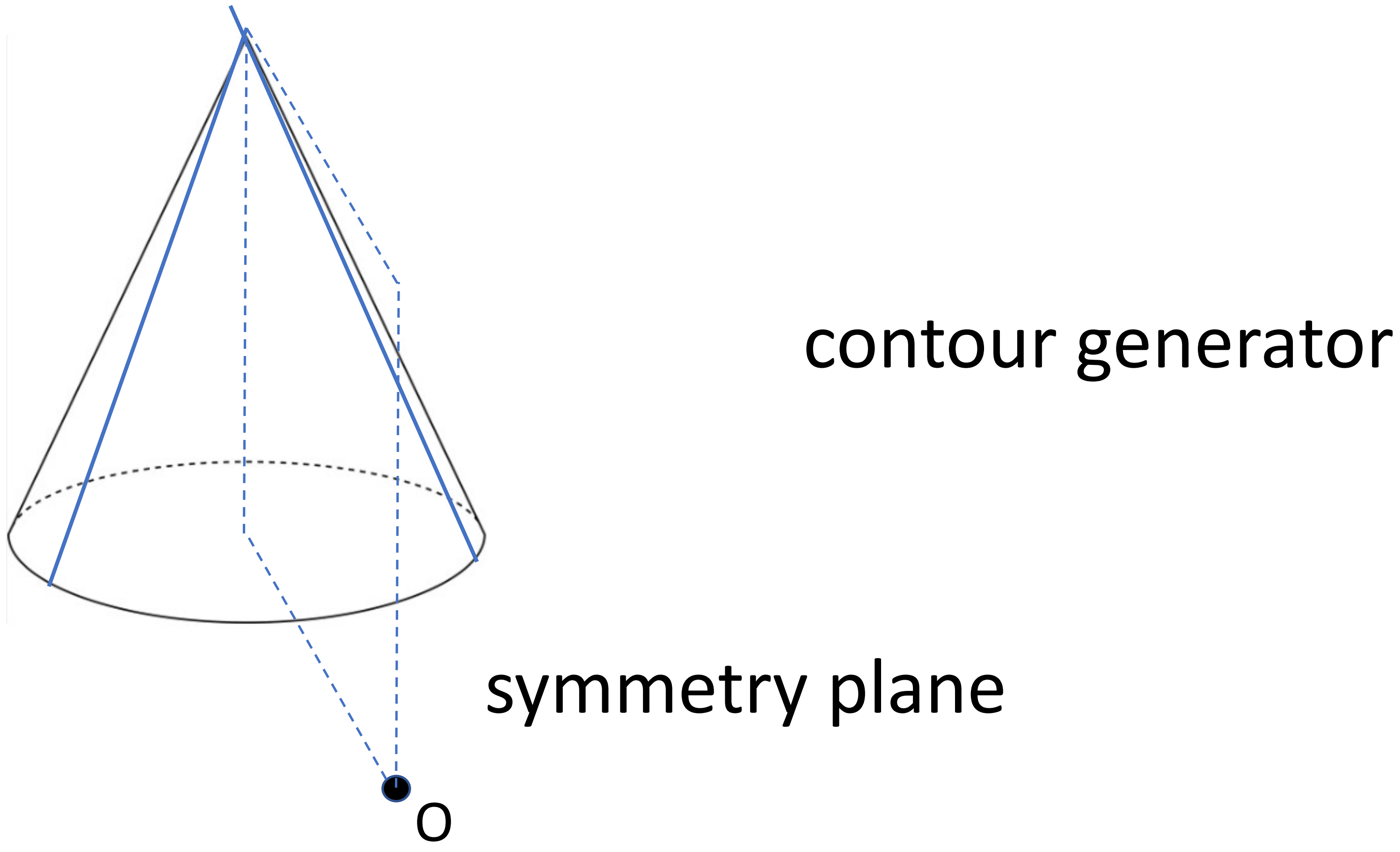
cone axis

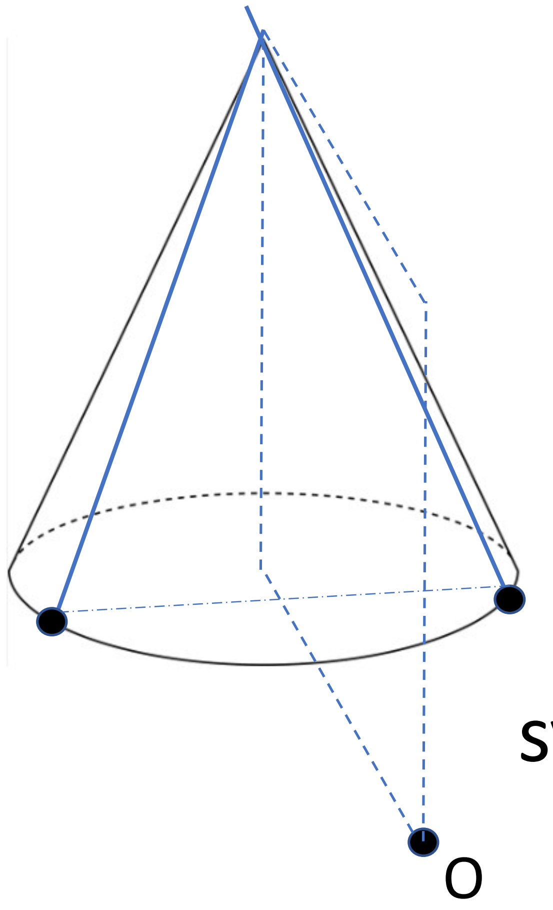

cone axis
a circular cross section


cone axis
a circular cross section
o
camera viewpoint



contour generator


contour generator
symmetry plane
O


contour generator
points are symmetric wrt symmetry plane
symmetry plane
O

Example: contour generator of a **right** cone?

- contour generator Γ: set of tangency points from O

→ two straight lines through the vertex V

- **right** cone e alone is **symmetric** wrt to its axis

→ right cone + viewpoint O : «less» symmetry,

which symmetry?

PLANAR SYMMETRY wrt plane through axis and O

Example: contour generator of a **right** cone?

- contour generator Γ: set of tangency points from O
→ two straight lines through the vertex V

- **right** cone alone is **symmetric** wrt to its axis
→ right cone + viewpoint O : «less» symmetry,
which symmetry?
PLANAR SYMMETRY wrt plane through axis and O
namely, wrt backprojection plane of the imaged axis

natural camera calibration
from 3 images of a rectangle

Classical approach

Three equations from vanishing points of three mutually orthogonal directions:

Drawbacks: estimation of vanishing points might be inaccurate

Alternative approach

- Mutuated from Zhang: images of a planar object of known shape

$\rightarrow$

- Here the object shape has a unique unknown parameter: the aspect ratio *a*, i.e., the ratio between «horizontal» and «vertical» lenght

Given a rectangle R of unknown aspect ratio a

- Take a first image of the rectangle: let r_1 be the extracted image
- Rectify this image by mapping it onto a rectangle R_1 by means of a homography H_1
- The aspect ratio of this reconstructed rectangle is $a_1 = sa$, where s is an unknown scale factor
- Therefore, the circular points of the plane containing the real rectangle R have been mapped onto two points (I_1, J_1) given by

$$(I_1, J_1) = \begin{bmatrix} s & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} (I, J)$$

- The first image of these circular points is therefore

$$(I_1', J_1') = H_1^{-1}(I_1, J_1) = H_1^{-1}\begin{bmatrix} s & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}(I, J)$$

- Consider now the second image R_2 of the rectangle: once it is partially rectified to a rectangle r_2, compute its aspect ratio $a_2 = s_2 a$. This time, the scale factor s_2 can be expressed in terms of s as

$$s_2 = \lambda s$$

with $\lambda = \frac{a_2}{a_1}$ known after the observation of the two aspect ratios

- If H_2 is the homography used to rectify the second image R_2 of the rectangle to the partially rectified rectangle r_2, then the second image of the circular points (I, J) is given by

$$(I_2', J_2') = H_2^{-1}(I_2, J_2) = H_2^{-1} \begin{bmatrix} \lambda s & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} (I, J)$$

and, similarly, the third image of the same circular points is

$$(I_3', J_3') = H_3^{-1}(I_3, J_3) = H_3^{-1} \begin{bmatrix} \mu s & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} (I, J)$$

Now we have three equations like in Zhang, but also an additional unknown scaling factor s

A further equation comes from the natural camera constraint:

An additional pair of imaged circular points is

$$(I, J) = \left(\begin{bmatrix} 1 \\ i \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ -i \\ 0 \end{bmatrix}\right)$$

in fact,

they are images of circular points of any plane parallel to the image plane

Therefore we have to solve

$$I_1'^T \omega I_1' = 0$$
$$I_2'^T \omega I_2' = 0$$
$$I_3'^T \omega I_3' = 0$$

and

$$I^T \omega I = 0$$

for both ω and s, with

$$\lambda = \frac{a_2}{a_1}$$

and

$$\mu = \frac{a_3}{a_1}$$