Tutorial A2: Introduction to Visual Geometry

Peter Corke

Image formation: from 3D to 2D

Cave Paintings ~40,000 years ago

Ideal City (1470)

Piero della Francesca (1415–1492)

Figure 28 (Jan Vredeman de Vries, 1604).

Used with permission from Perspective, Dover Publications,1964.

trompe l'oeil **lˌtrômp ˈloil**

noun (pl. **trompe l'oeils** pronunc. **same**)

visual illusion in art, esp. as used to trick the eye into perceiving a painted detail as a three-dimensional object.

Trompe L'oeil Tuscan Window Mural 2009

Kristin Plansky l Used with permission.

New York City, Lower Manhattan, Front St.: **Richard Haas** *Trompe l'oeil* 1975

Vincent Desjardins, 2011 I CC A2.0

People are actually avoiding walking in the "hole" 2007

Joe Beever | CC A2.0

Stunning 3D chalk drawing from Zebit stops Liverpool shoppers in their tracks on Bold Street. *2012*

Bill Hunt Original art: **Zebit |** CC A2.0

Edgar Meuller http://www.metanamorph.com

Forced perspective *2011*

Seongbin Im | CC A2.0

On hands *2013*

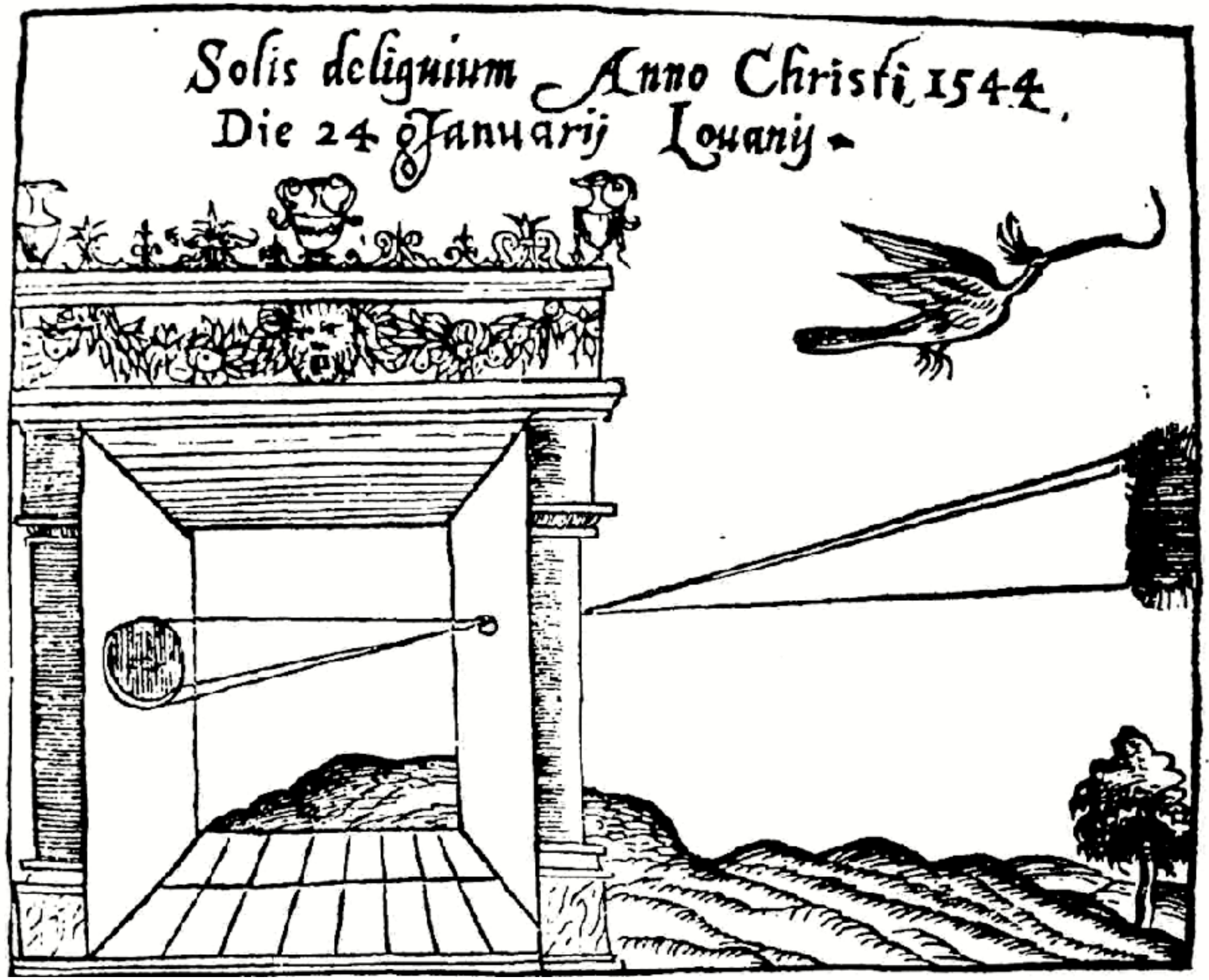
Kenzie Saunders | CC A2.0



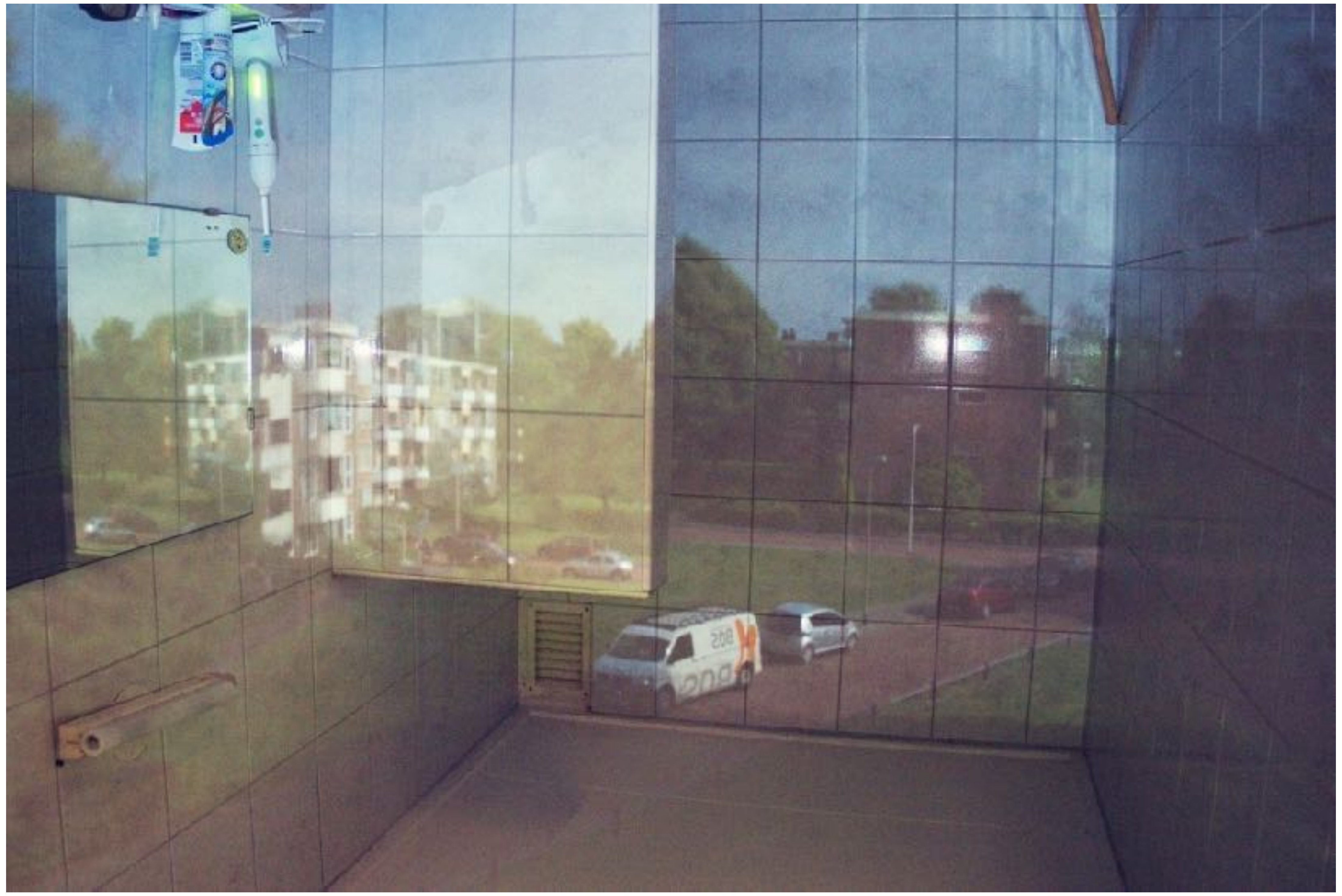

Points in the world
Image plane


Image plane

The pinhole camera



Pinhole images



Camera obscura 2011

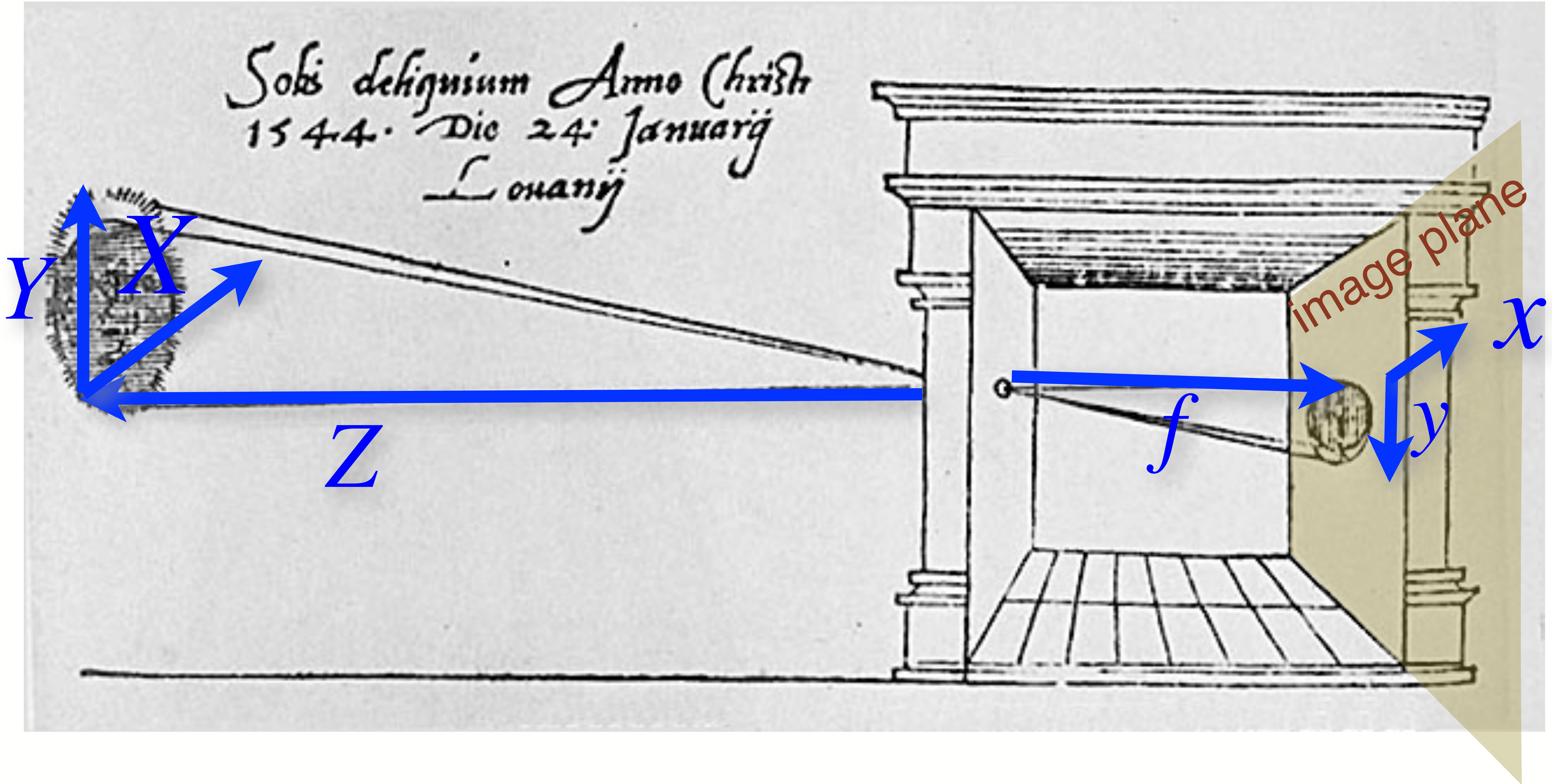
1banaan | CC A2.0

Camera obscura! 2011

half alive - soo zzzz | CC A2.0


Image plane

Simple imaging

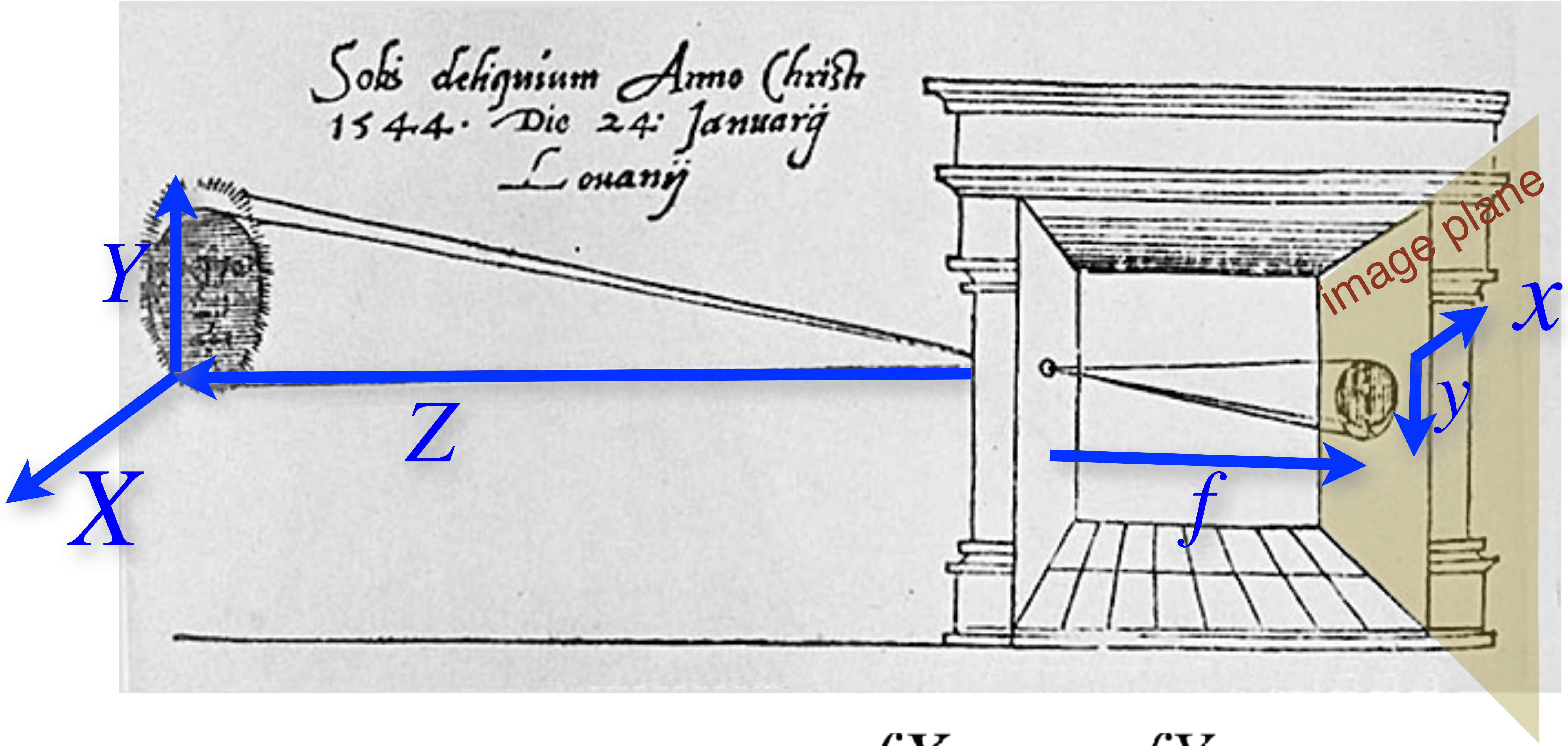


$$\frac{Y}{Z} = \frac{y}{f}$$

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

- Similar triangles
- Image formation is the mapping of scene points (X,Y,Z) to the image plane (x,y)
- Image is inverted

Simple imaging



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

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

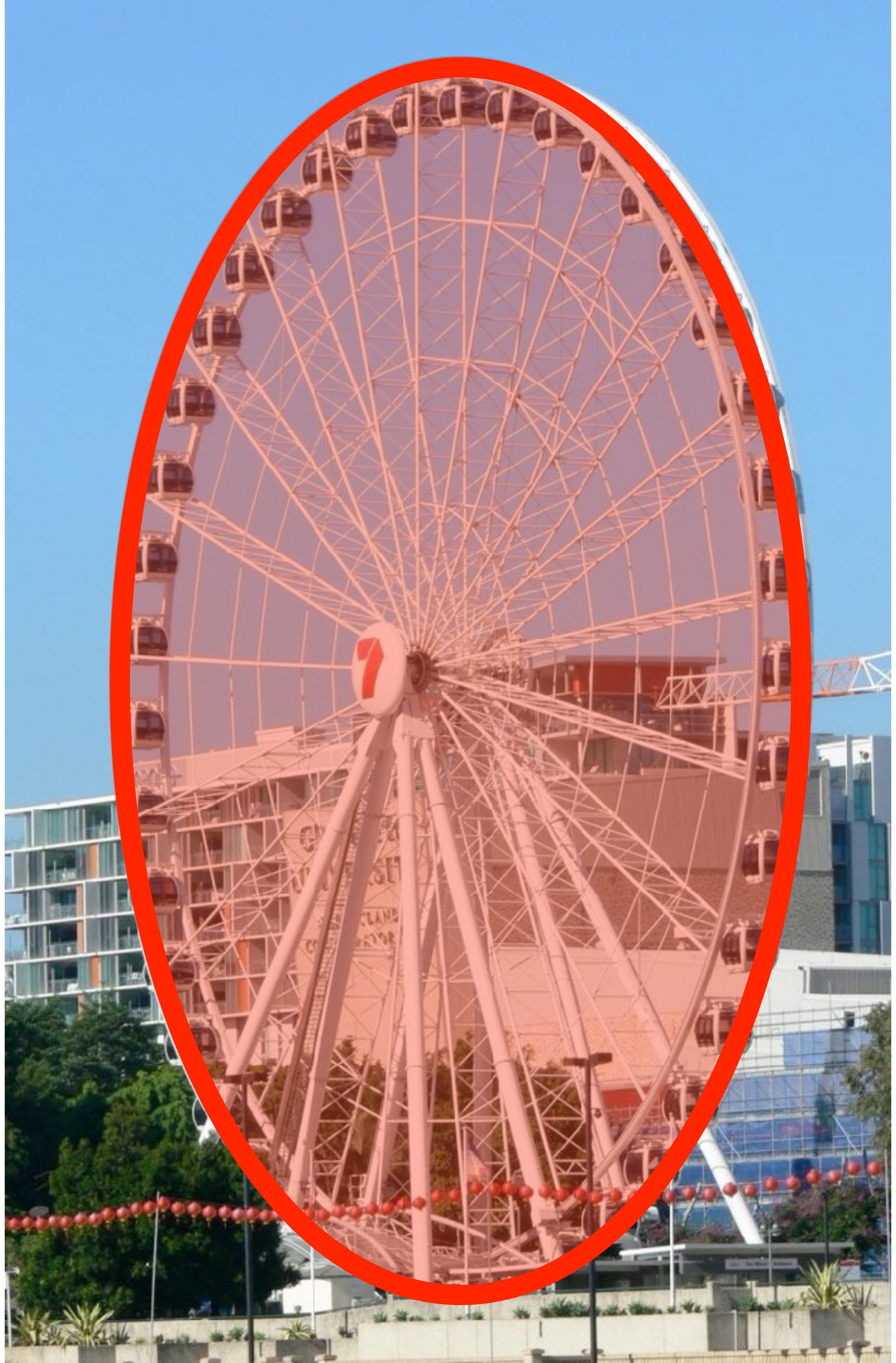
$$x = \frac{fX}{Z}, y = \frac{fY}{Z}$$

- 3D to 2D

$$(X, Y, Z) \mapsto (x, y)$$

- Perspective projection

$$\mathbb{R}^3 \mapsto \mathbb{R}^2$$



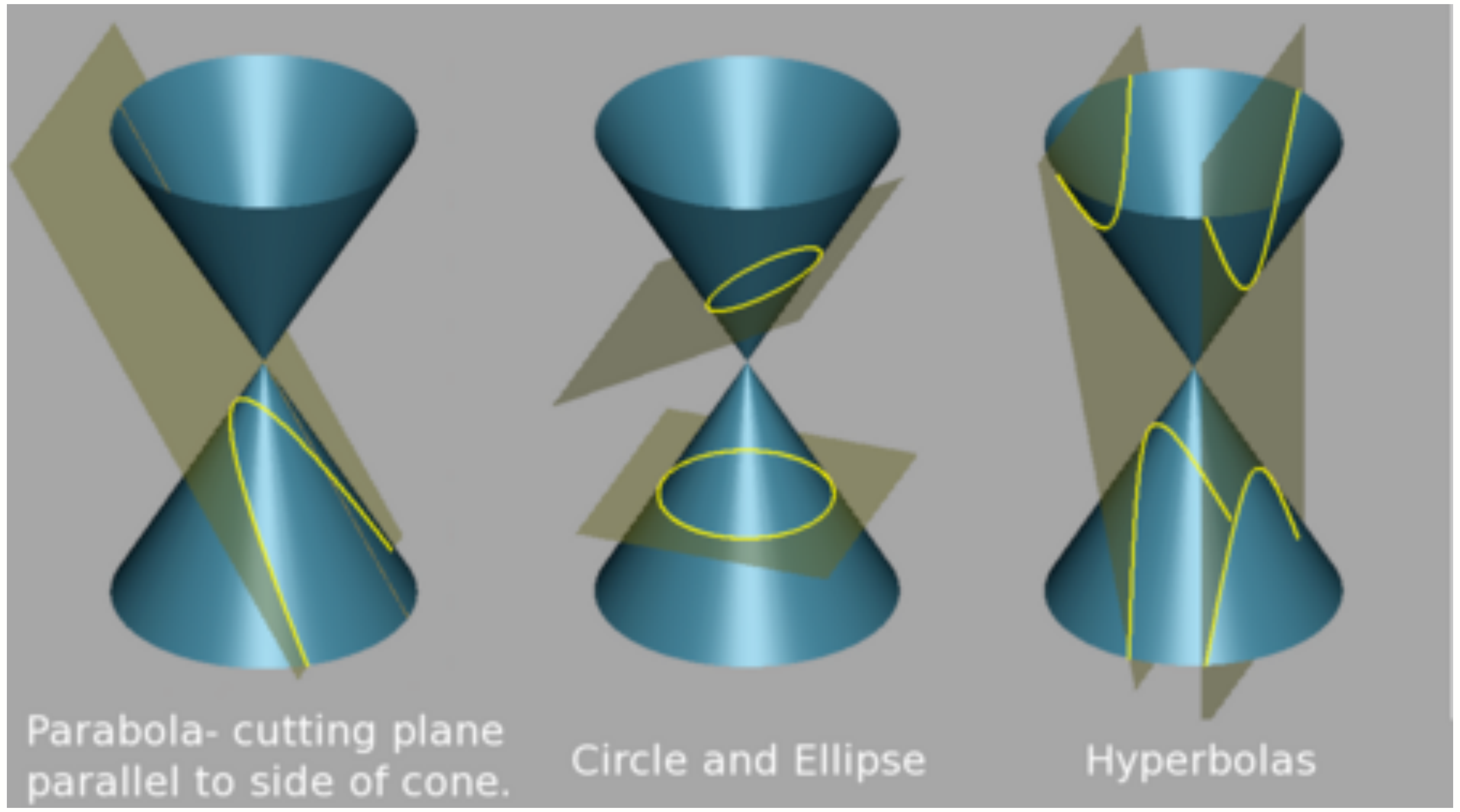
With kind permission of

Springer Science+Business Media

Perspective projection

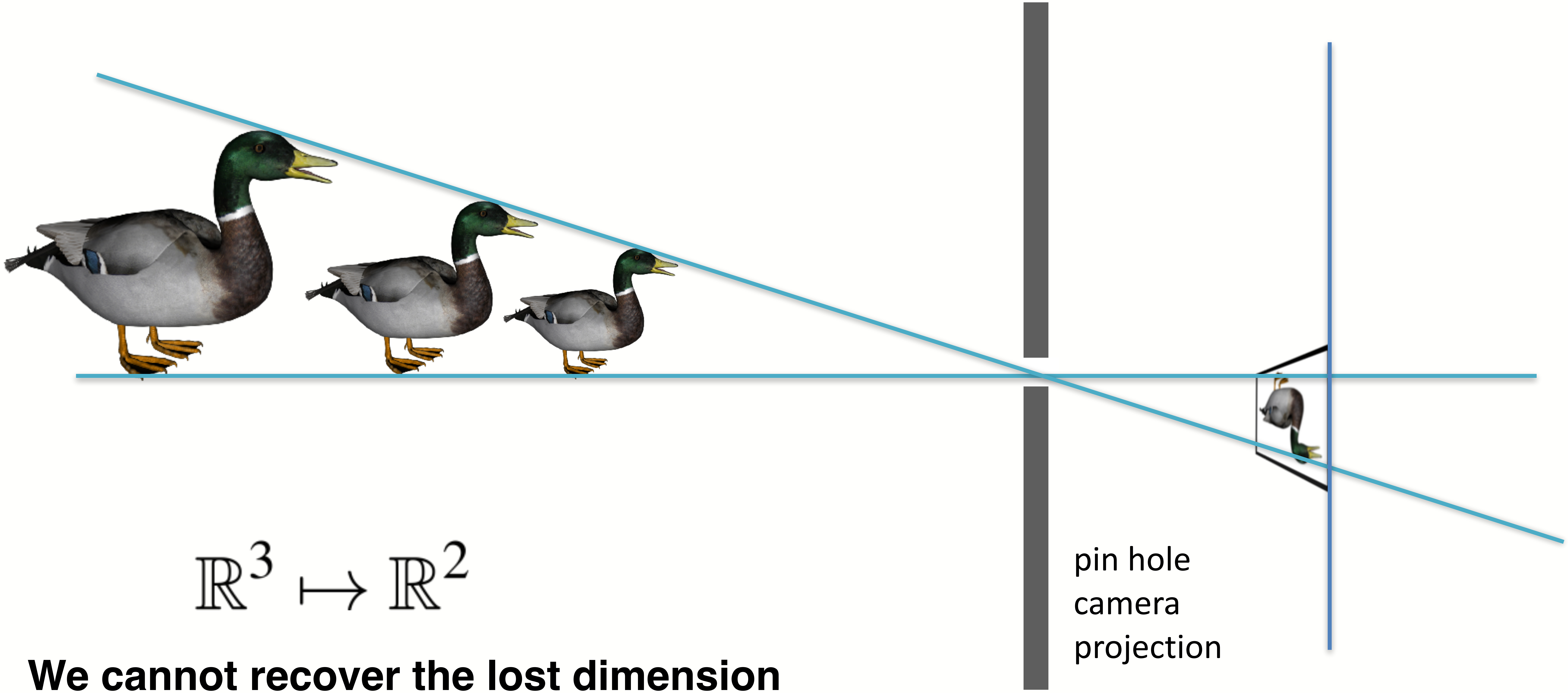
Maps

- Lines → lines
  - ↪ parallel lines not necessarily parallel
  - ↪ angles are not preserved
- Conics → conics



Duk at the English language Wikipedia

No unique inverse



$$\mathbb{R}^3 \mapsto \mathbb{R}^2$$

We cannot recover the lost dimension

- Any 2D image could be generated by one of an infinite number of possible 3D worlds


Image plane

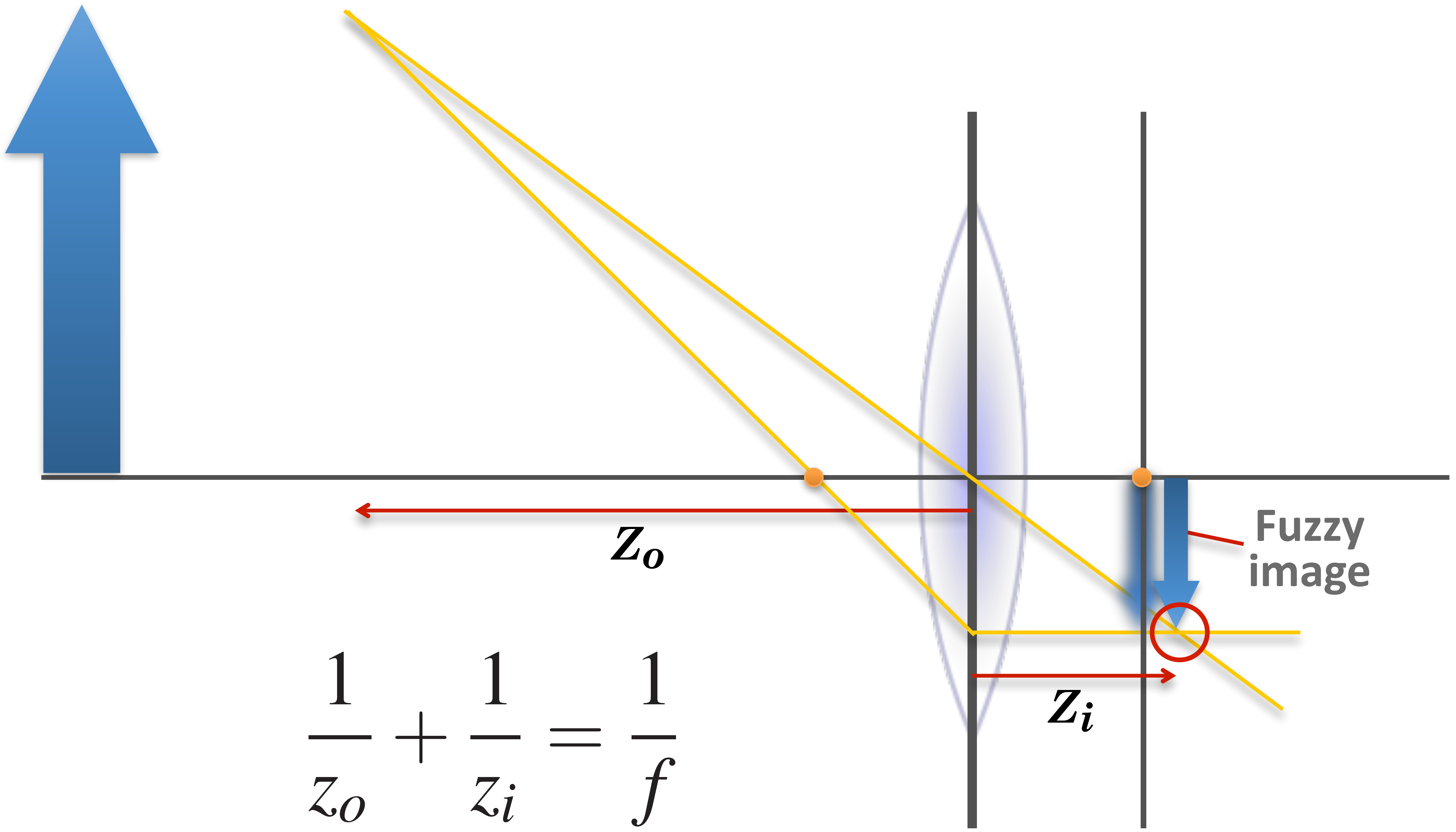
Use a lens to gather more light

George R. Lawrence 1900

f
ø

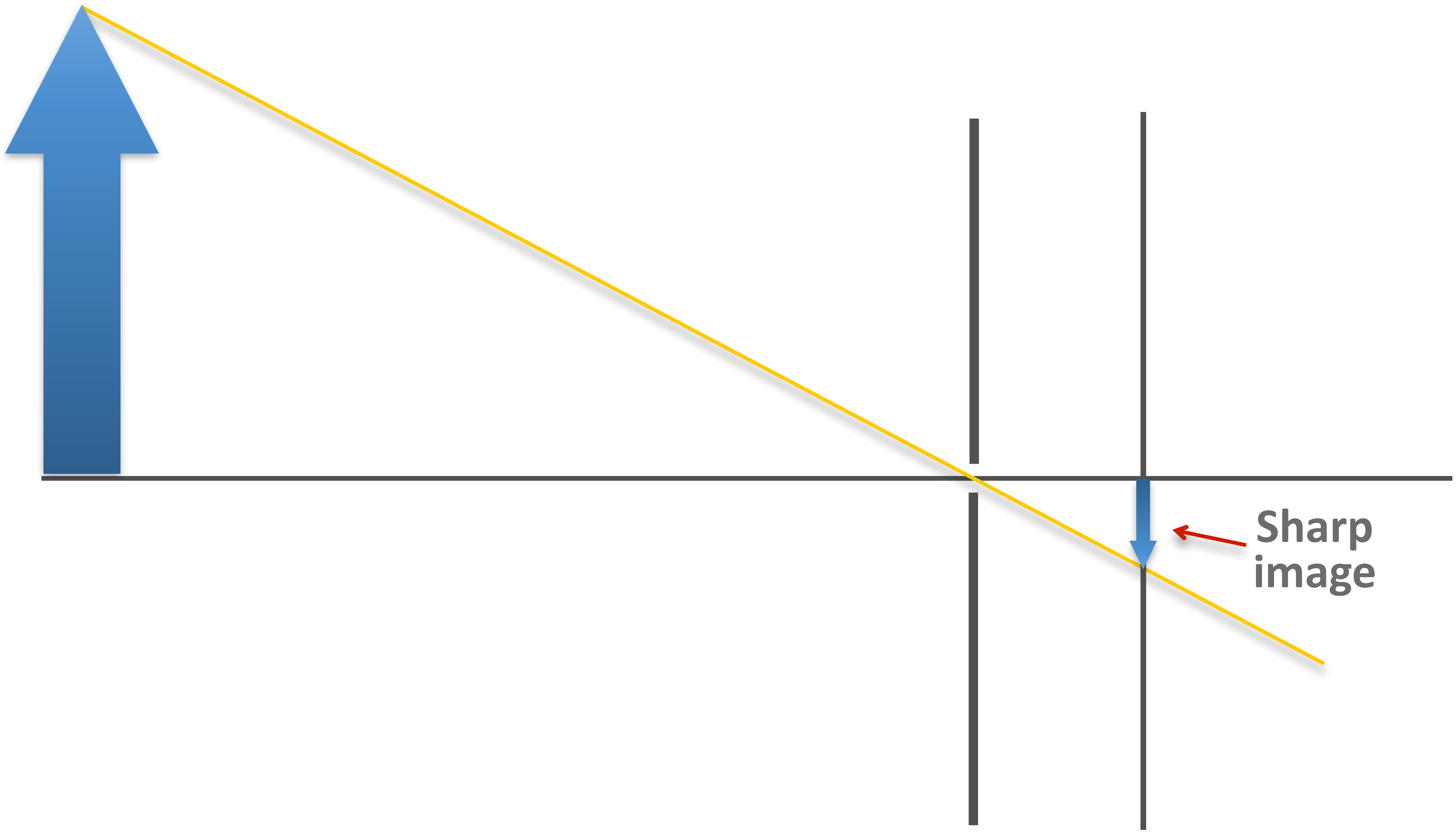
$F = \frac{f}{\phi}$
f
ø
F/1.4 lens
1 : 1.4
NIKKOR
50mm
5755899

$$\frac{1}{z_o} + \frac{1}{z_i} = \frac{1}{f}$$

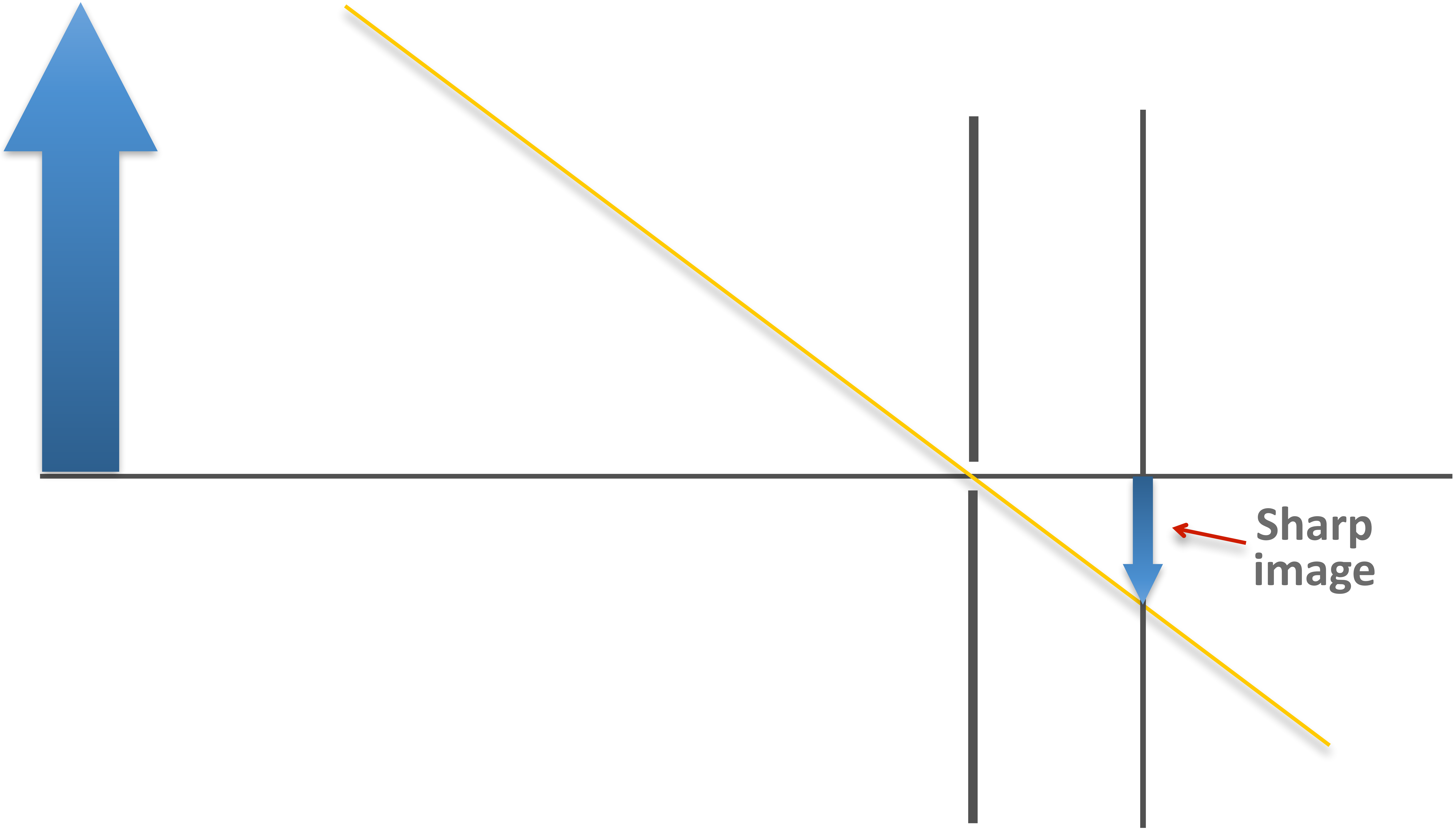


$$\frac{1}{z_o} + \frac{1}{z_i} = \frac{1}{f}$$

Pinhole camera doesn't need focus

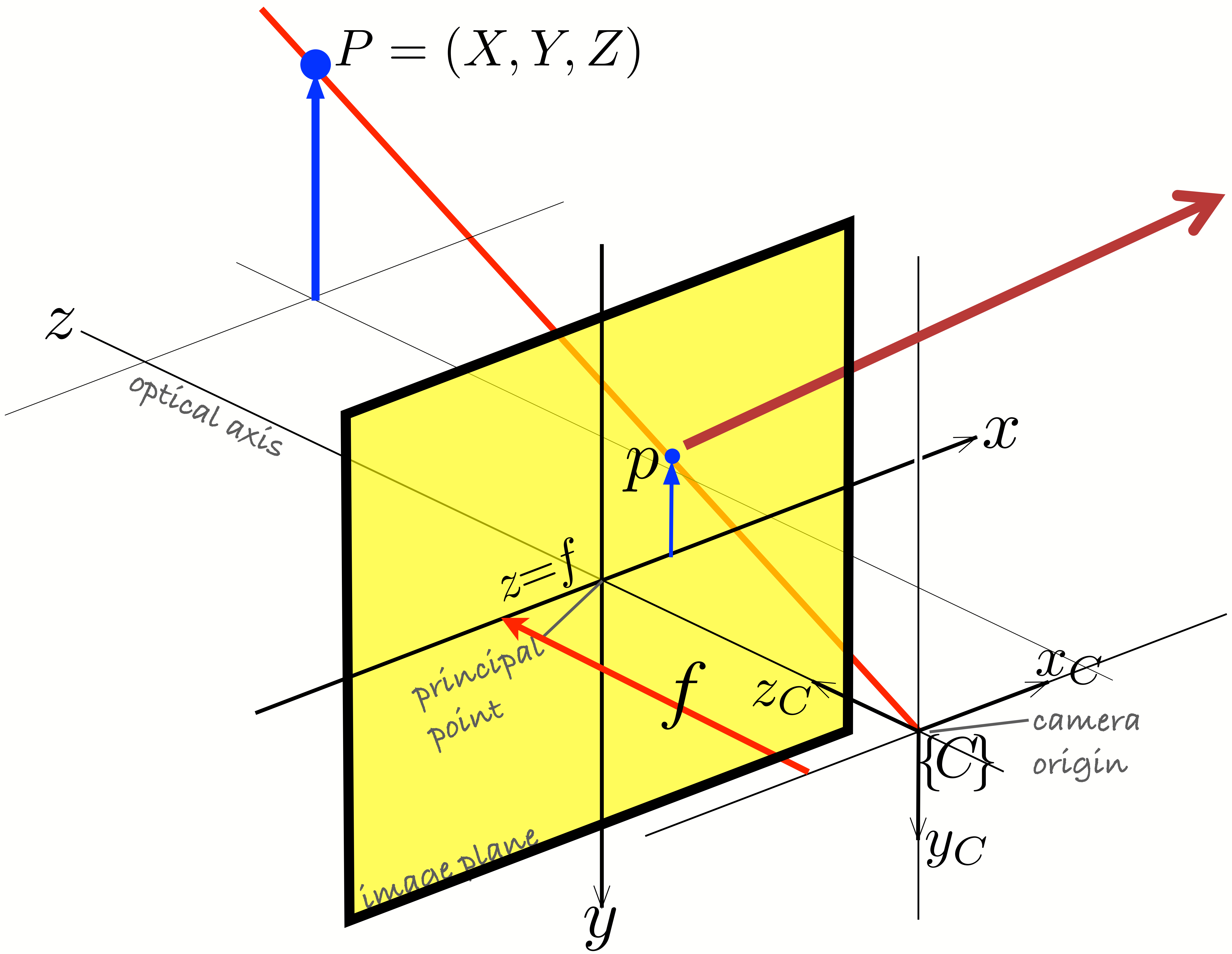


Pinhole camera doesn't need focus



Central projection camera model

Central projection model



$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Corke, P. I Reproduced with permission from Springer Science, & Business Media

Pin-hole model in homogeneous form

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$\tilde{x} = fX, \tilde{y} = fY, \tilde{z} = Z$$

$$x = \frac{\tilde{x}}{\tilde{z}}, \; y = \frac{\tilde{y}}{\tilde{z}}$$

$$\Rightarrow x = \frac{fX}{Z}, y = \frac{fY}{Z}$$

- Perspective transformation, with the pesky divide by Z, is **linear** in homogeneous coordinate form.

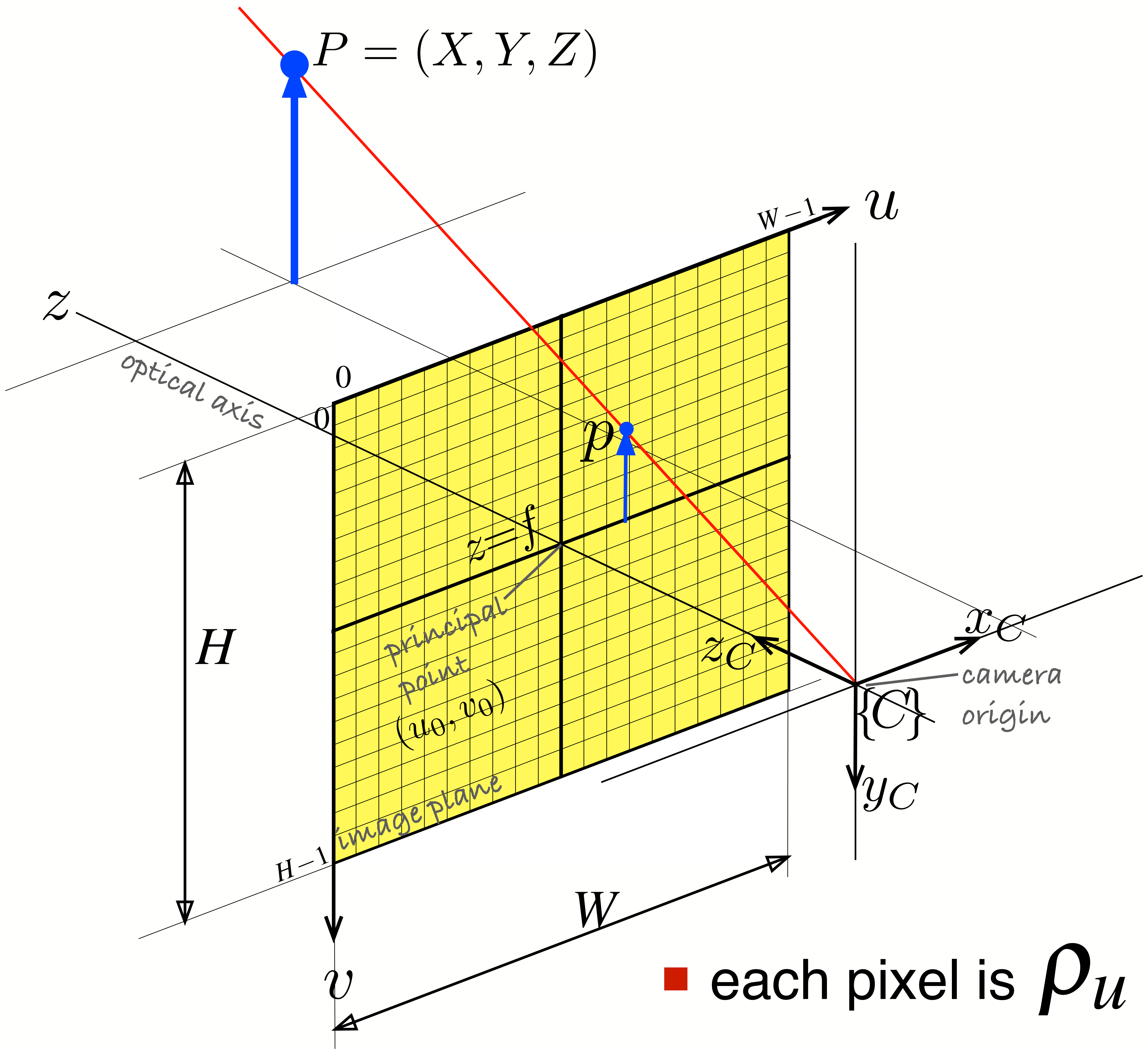
$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{pmatrix} = \boxed{\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

3D to 2D

$$\boxed{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}$$

scaling/
zooming

Change of coordinates

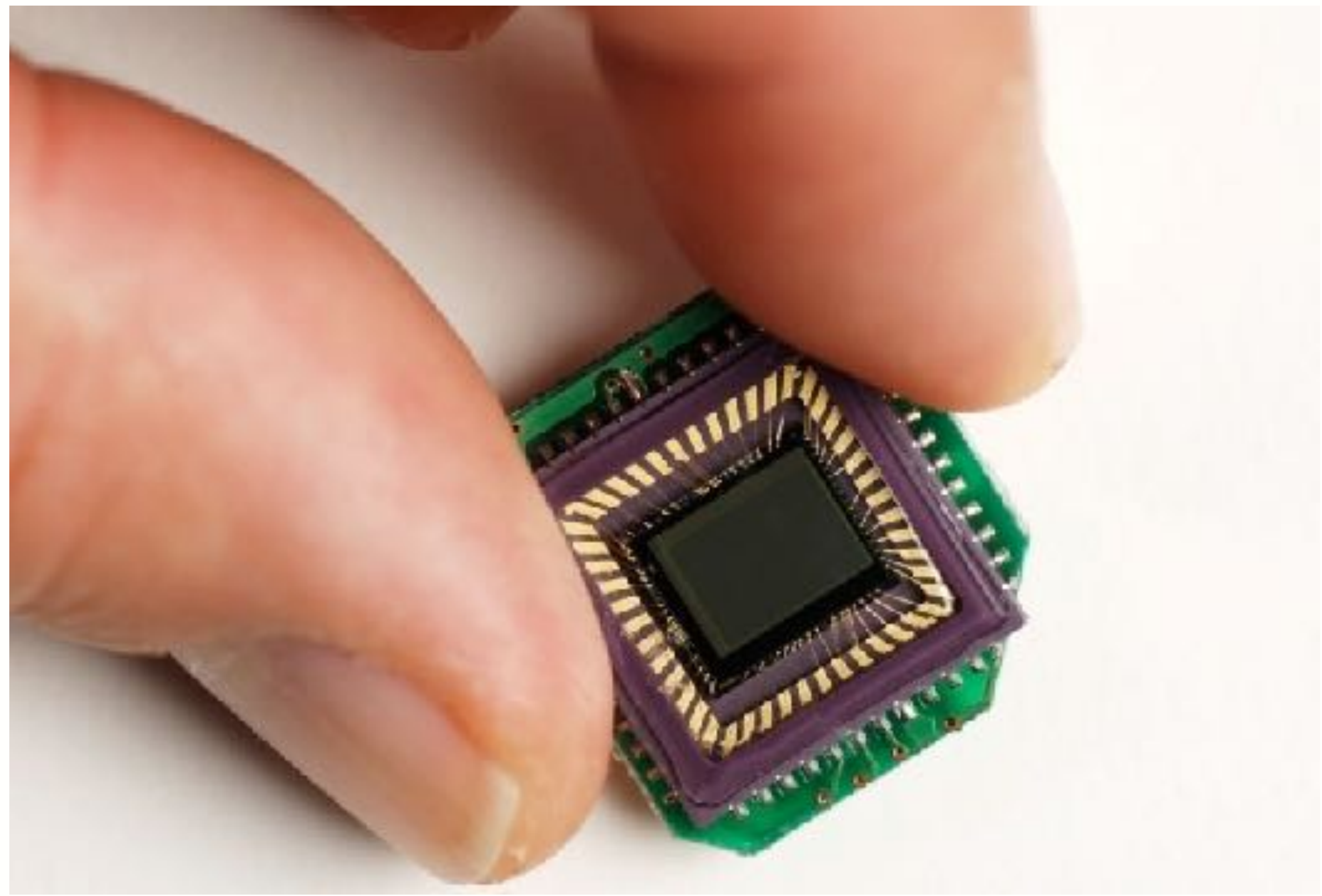


- scale point from metres to pixels
- shift the origin to top left corner

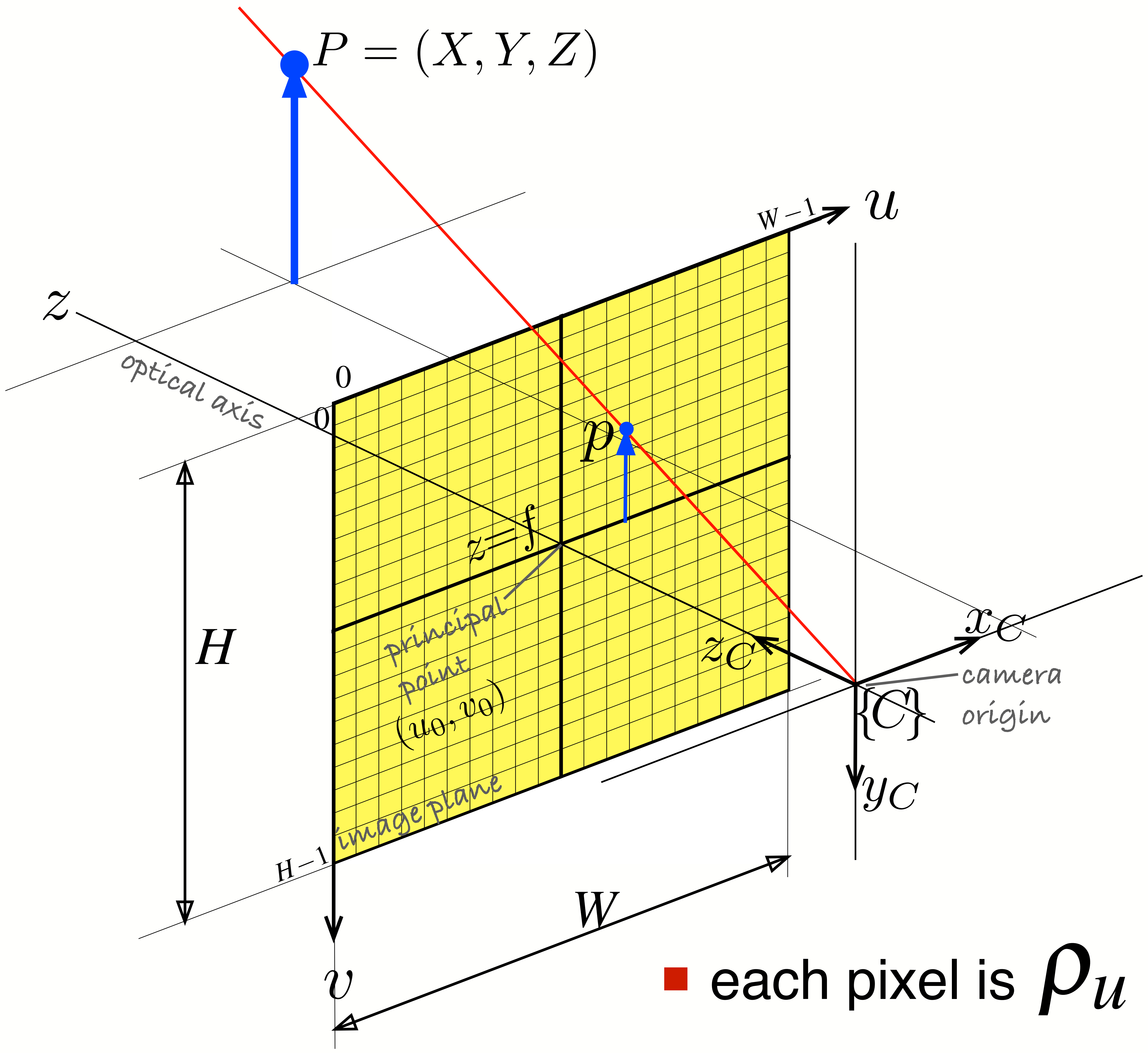
$$u = \frac{x}{\rho_u} + u_0$$

$$v = \frac{y}{\rho_v} + v_0$$

- each pixel is $\rho_u \times \rho_v$



Change of coordinates



- scale point from metres to pixels
- shift the origin to top left corner

$$u = \frac{x}{\rho_u} + u_0$$

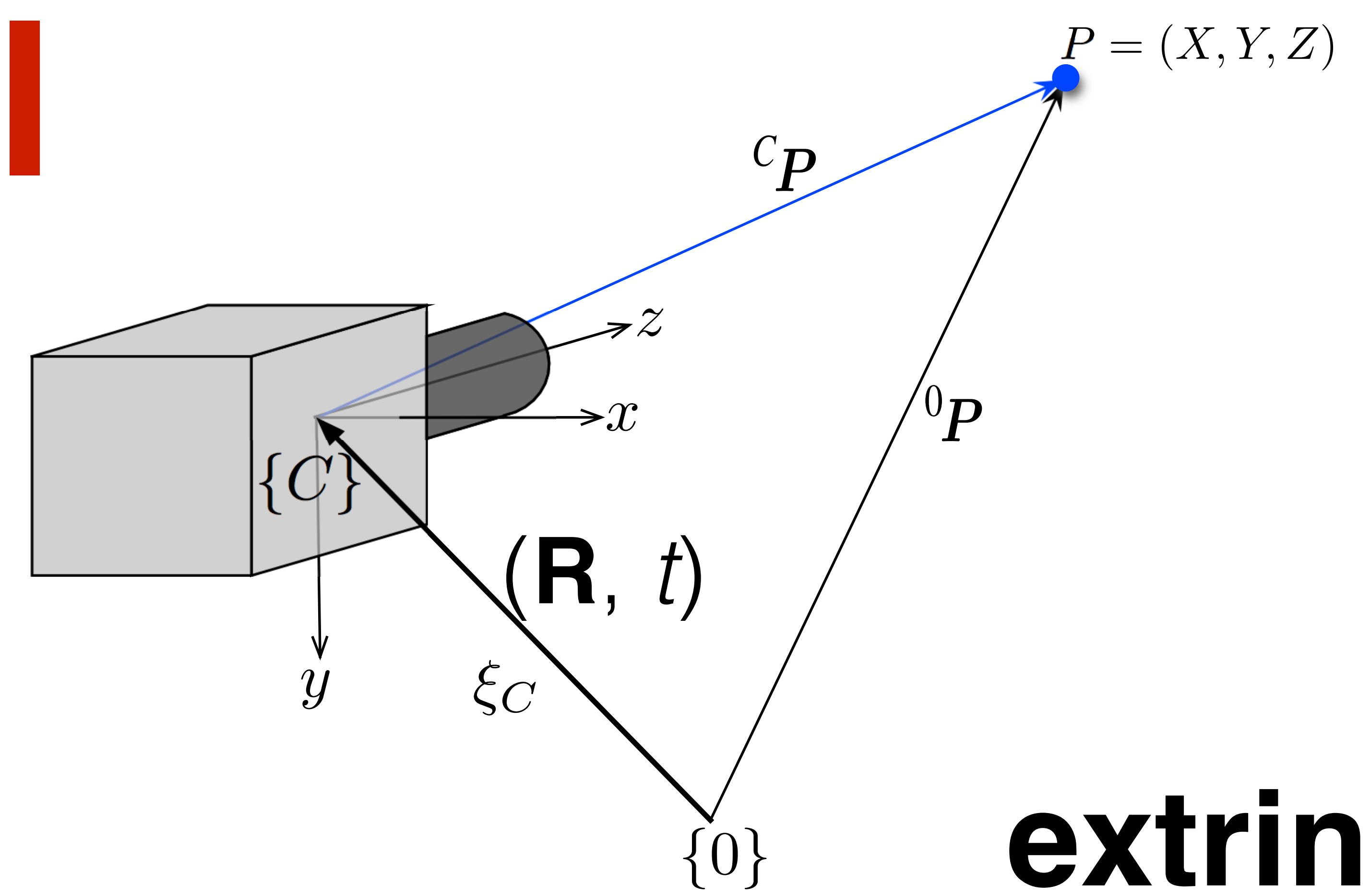
$$v = \frac{y}{\rho_v} + v_0$$

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} \frac{1}{\rho_u} & 0 & u_0 \\ 0 & \frac{1}{\rho_v} & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{pmatrix}$$

- each pixel is $\rho_u \times \rho_v$

$$p = \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \tilde{u}/\tilde{w} \\ \tilde{v}/\tilde{w} \end{pmatrix}$$

Complete camera model



$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

extrinsic parameters

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \underbrace{\underbrace{\begin{pmatrix} \frac{1}{\rho_u} & 0 & u_0 \\ 0 & \frac{1}{\rho_v} & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\mathbf{K}} \begin{pmatrix} \mathbf{R} & t \\ \mathbf{0}_{1\times 3} & 1 \end{pmatrix}^{-1}}_{\mathbf{C}} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \qquad \ominus \xi_C$$

intrinsic parameters

camera matrix

Camera **matrix**

- Mapping points from **the world** to an **image (pixel) coordinate** is simply a **matrix multiplication** using **homogeneous coordinates**

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$u = \frac{\tilde{u}}{\tilde{w}}, v = \frac{\tilde{v}}{\tilde{w}}$$

Scale **invariance**

- Consider an arbitrary scalar scale factor

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \lambda \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

- $\tilde{u}, \tilde{v}, \tilde{w}$ will all be scaled by λ

- but $u = \dfrac{\tilde{u}}{\tilde{w}}, v = \dfrac{\tilde{v}}{\tilde{w}}$

- so the result is unchanged

Normalized camera matrix

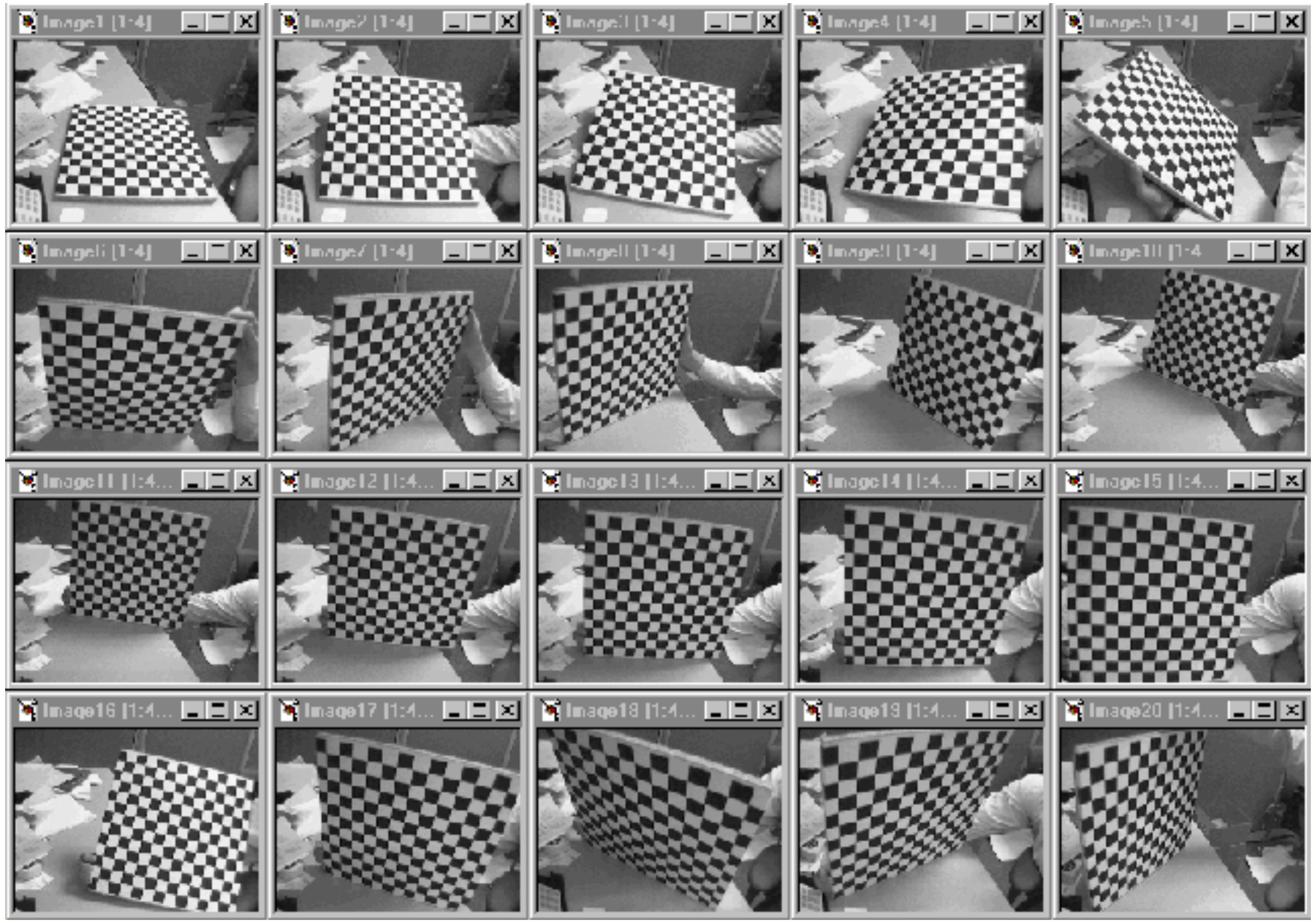
- Since scale factor is arbitrary we can fix the value of one element, typically C(3,4) to one.

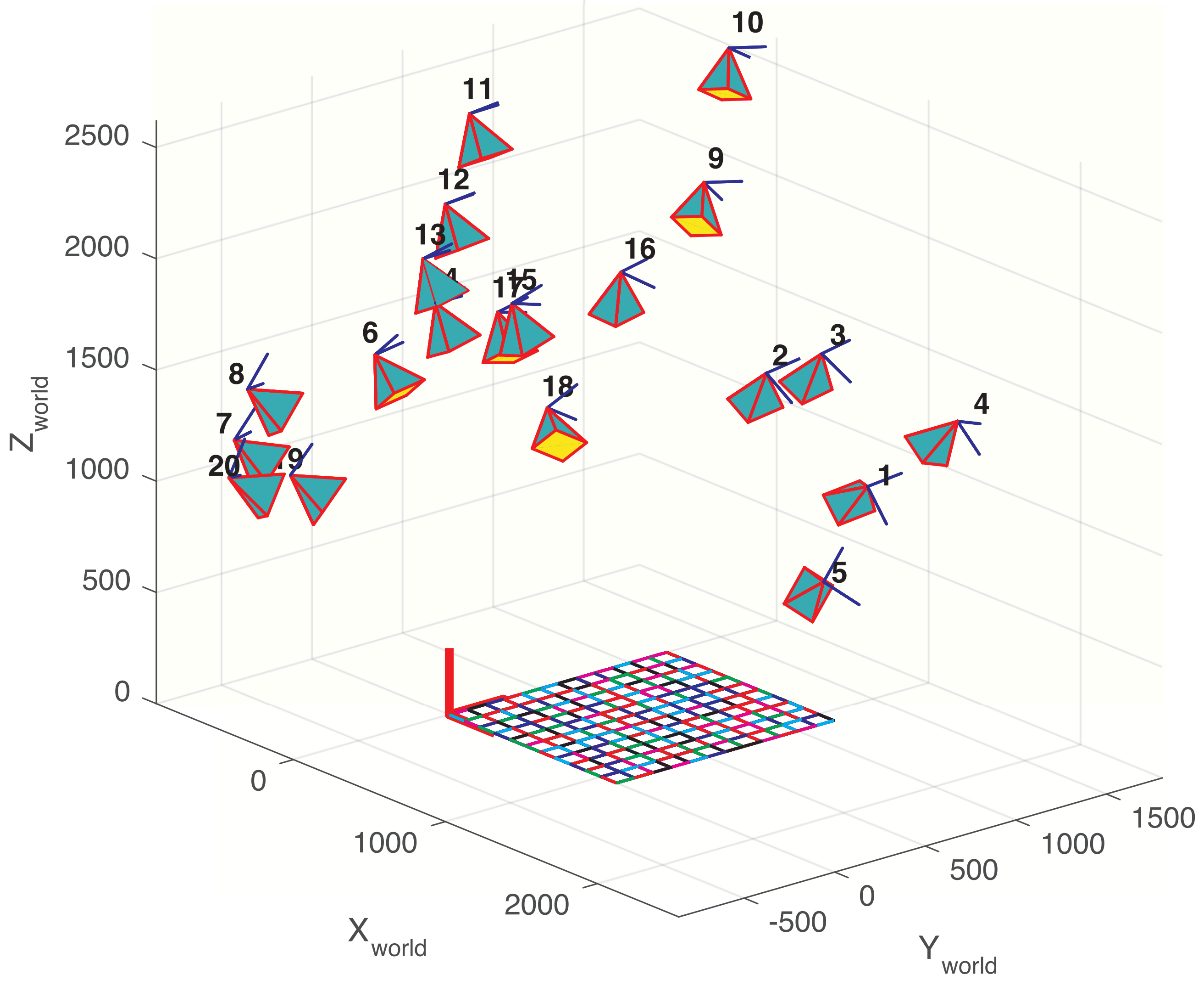
- focal length
- pixel size
- camera position
- & orientation

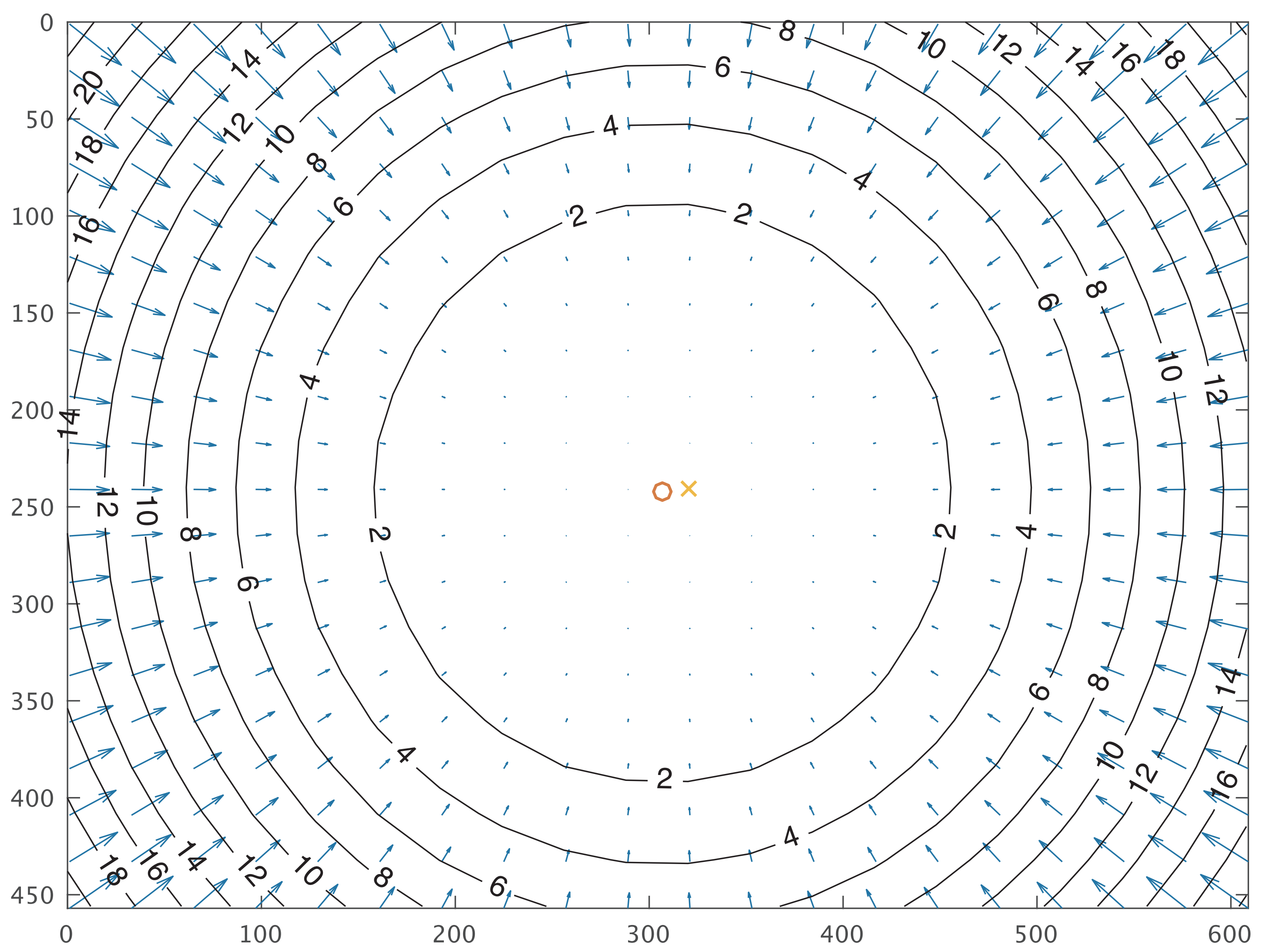
$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$u = \frac{\tilde{u}}{\tilde{w}}, v = \frac{\tilde{v}}{\tilde{w}}$$

Camera calibration







- Process to determine intrinsic and extrinsic camera parameters

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} \boxed{C_{11} \quad C_{12} \quad C_{13}} & \boxed{C_{14}} \\ \boxed{C_{21} \quad C_{22} \quad C_{23}} & \boxed{C_{24}} \\ \boxed{C_{31} \quad C_{32} \quad C_{33}} & 1 \end{pmatrix} \begin{pmatrix} \boxed{\begin{matrix} X \\ Y \\ Z \end{matrix}} \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} A_1 & b_1 \\ A_2 & b_2 \\ A_3 & 1 \end{pmatrix} \begin{pmatrix} P \\ 1 \end{pmatrix}$$

$$u = \frac{\tilde{u}}{\tilde{w}} = \frac{A_1 P + b_1}{A_3 P + 1} \qquad v = \frac{\tilde{v}}{\tilde{w}} = \frac{A_2 P + b_2}{A_3 P + 1}$$

$$\Rightarrow A_1 P + b_1 - uPA_3 = u \qquad \Rightarrow A_2 P + b_2 - vPA_3 = v$$

2x11 matrix

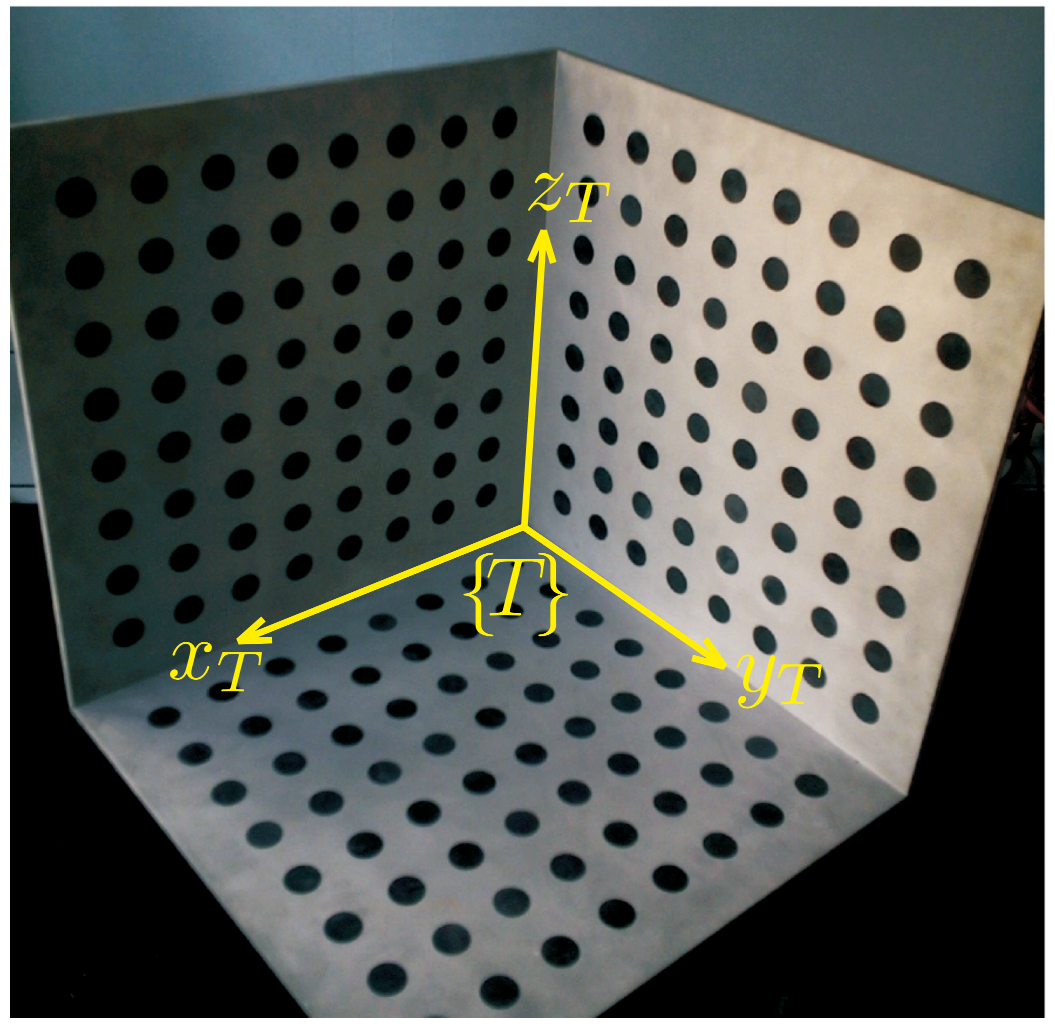
$$\begin{pmatrix} P & 1 & 0 & 0 & -uP \\ 0 & 0 & P & 1 & -vP \end{pmatrix} \begin{pmatrix} A_1 \\ b_1 \\ A_2 \\ b_2 \\ A_3 \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix}$$

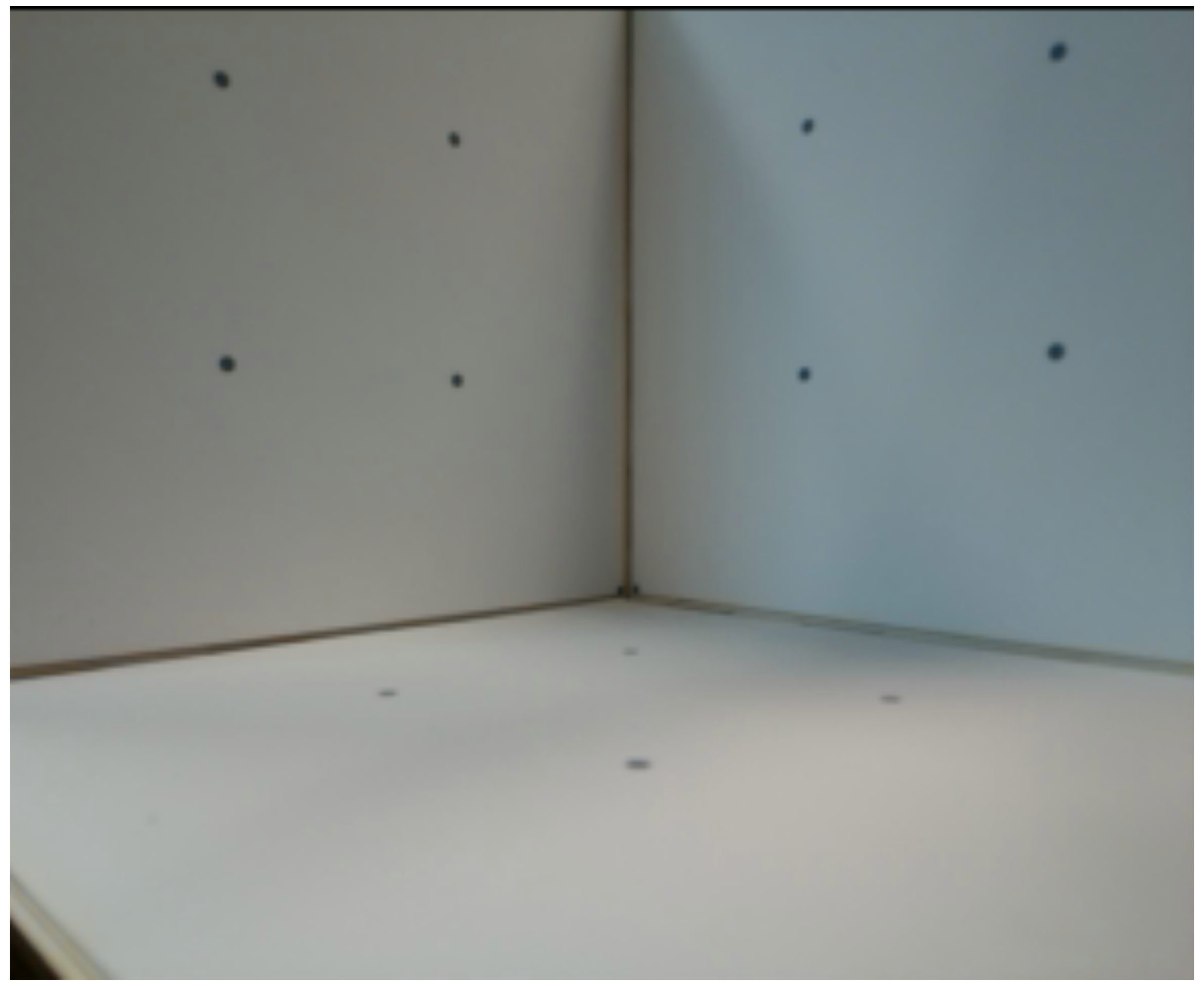
$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{21} & C_{22} & C_{23} & C_{24} & C_{31} & C_{32} & C_{33} \end{pmatrix}^T$$

For N points we stack these 2x11 matrices. To solve we need number of rows > number of columns, ie. N ≥ 6

$$\begin{pmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1Z_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 & -v_1Z_1 \\ & & & & & \vdots & & & & & \\ X_N & Y_N & Z_N & 1 & 0 & 0 & 0 & 0 & -u_NX_N & -u_NY_N & -u_NZ_N \\ 0 & 0 & 0 & 0 & X_N & Y_N & Z_N & 1 & -v_NX_N & -v_NY_N & -v_NZ_N \end{pmatrix} \begin{pmatrix} C_{11} \\ C_{12} \\ \vdots \\ C_{33} \end{pmatrix} = \begin{pmatrix} u_1 \\ v_1 \\ \vdots \\ u_N \\ v_N \end{pmatrix}$$

Will fail if all the points lie on a plane



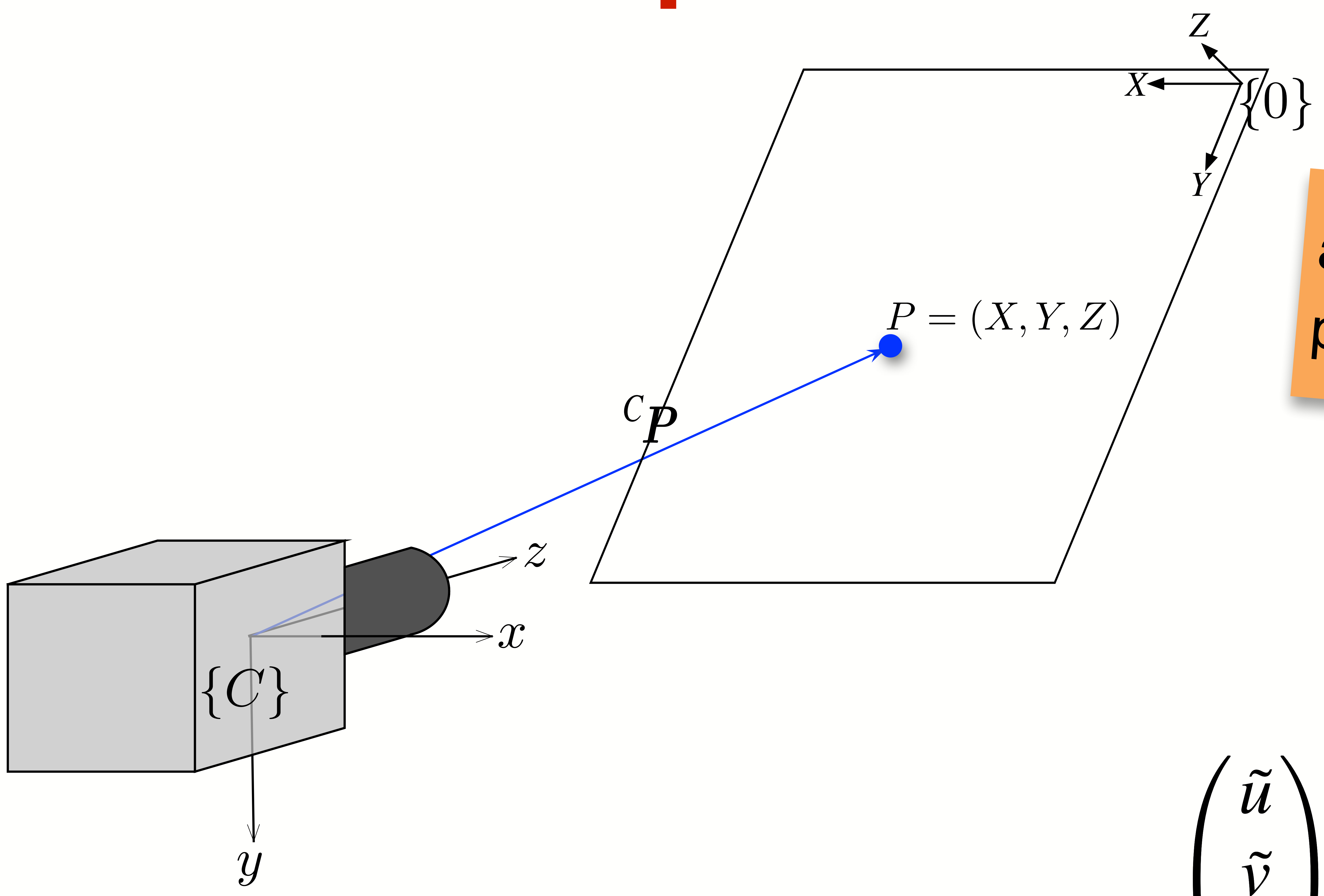


The camera matrix scrambles all the intrinsic and extrinsic parameters into 11 unique values. It is possible to extract K, R and t.

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \underbrace{\underbrace{\begin{pmatrix} \frac{1}{\rho_u} & 0 & u_0 \\ 0 & \frac{1}{\rho_v} & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\mathbf{K}} \begin{pmatrix} \mathbf{R} & t \\ \mathbf{0}_{1\times 3} & 1 \end{pmatrix}^{-1}}_{\mathbf{C}} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

The factors $f/\rho_u, f/\rho_v$ cannot be untangled - focal length in units of pixels

Points on a plane

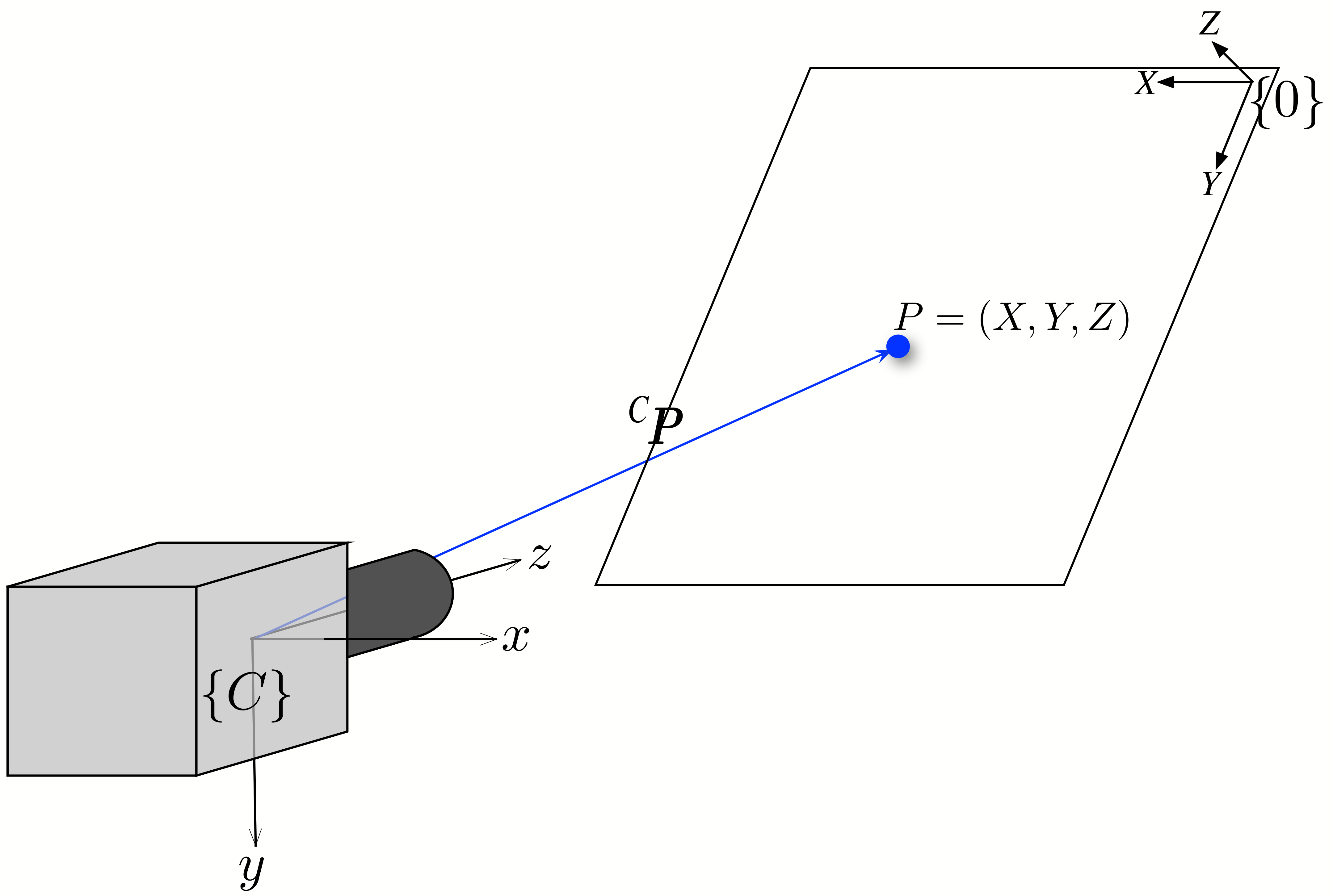


all points on the plane have z=0

3 x 3 matrix

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Planar homography



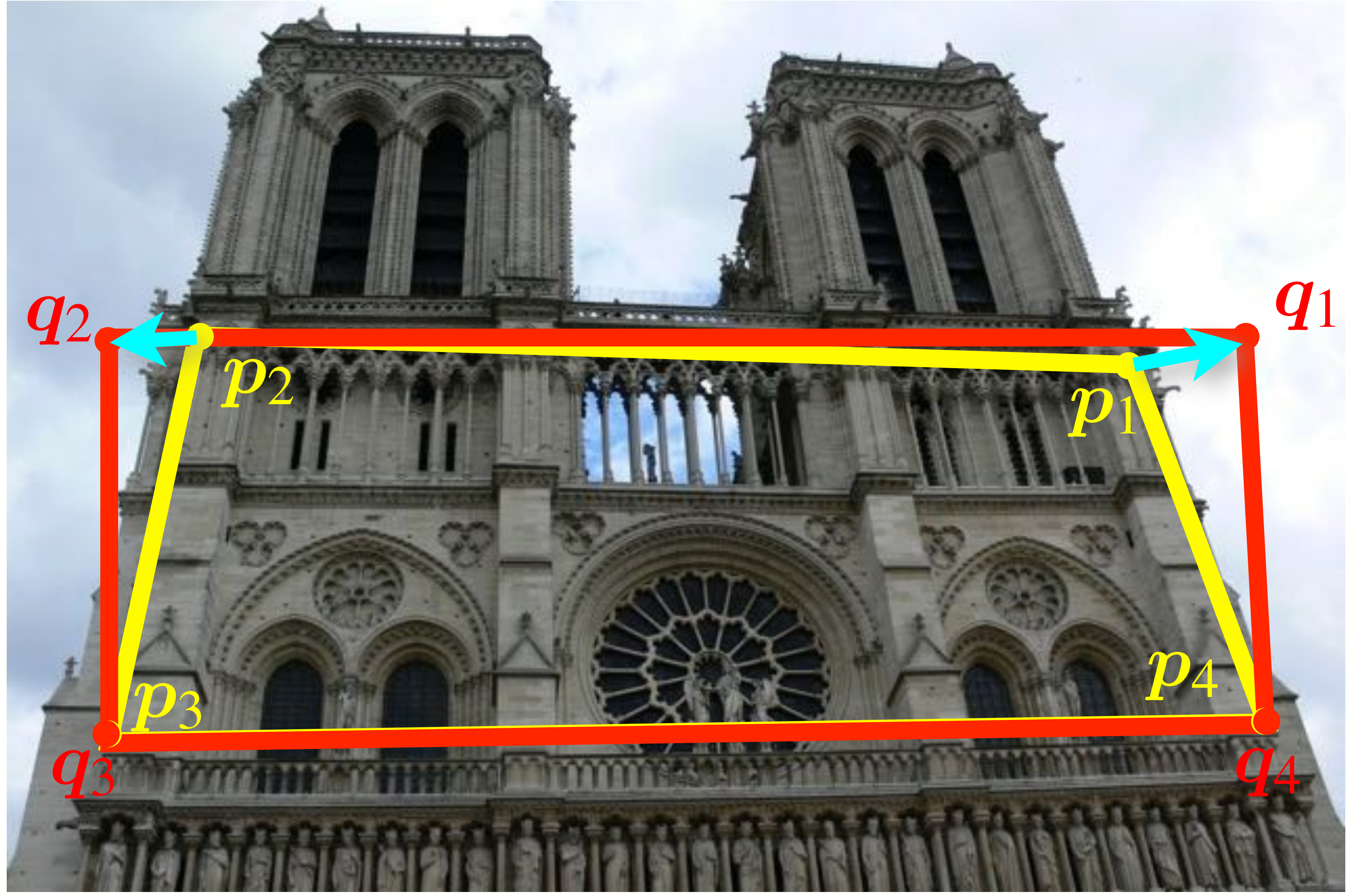
homography matrix

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

- Once again the scale factor is arbitrary
- 8 unique numbers in the homography matrix
- Can be estimated from 4 world points and their corresponding image points

$$\mathbf{H} = \mathbf{R} + \frac{\boldsymbol{t}}{d}\boldsymbol{n}^T$$

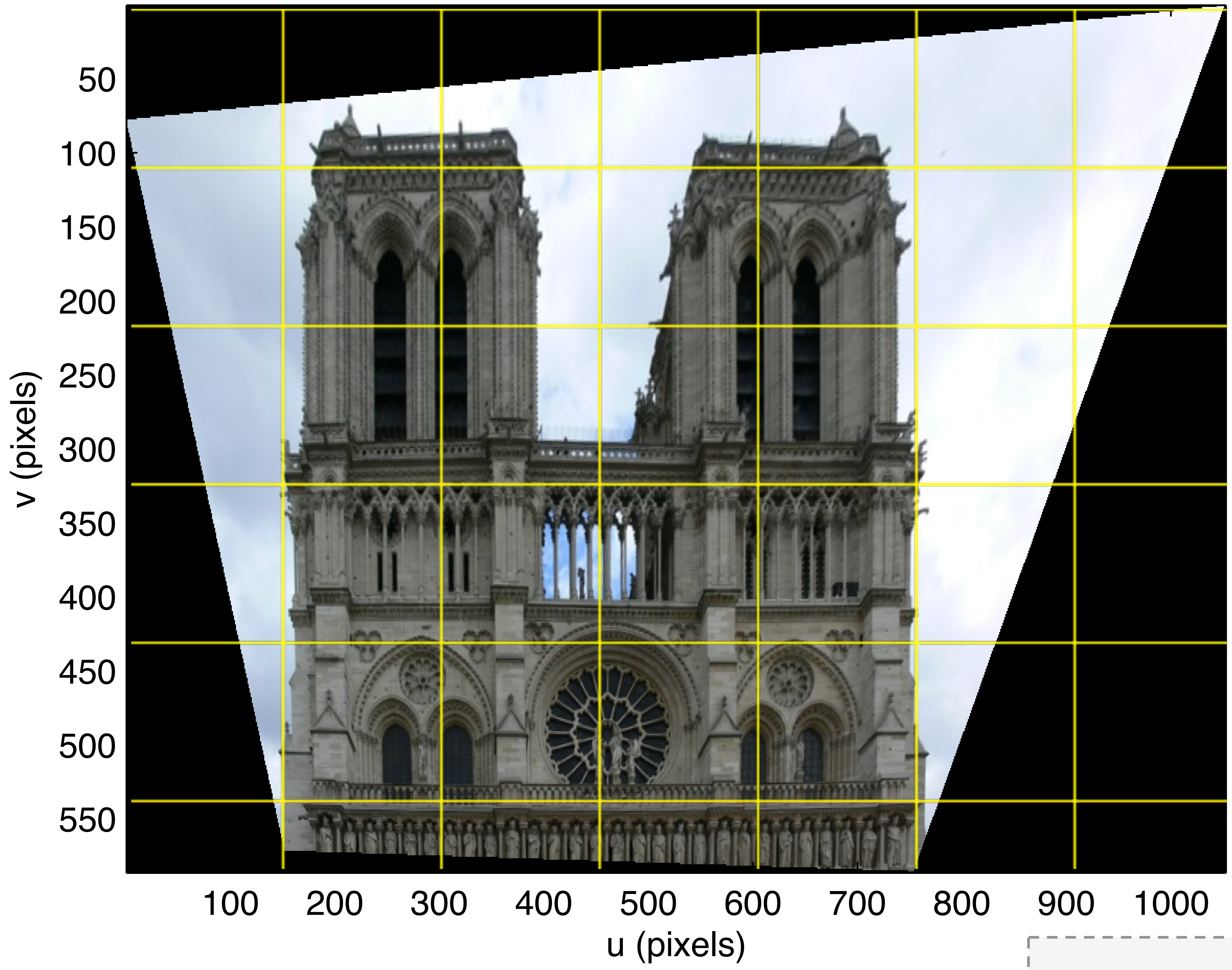
Perspective rectification



$$q = \mathbf{H}p$$

```
>>> H, _ = CentralCamera.points2H(P, Q)
H=
 1.4003  0.3827 -136.5900
-0.0785  1.8049 -83.1054
-0.0003  0.0016 1.0000
```

Perspective rectification



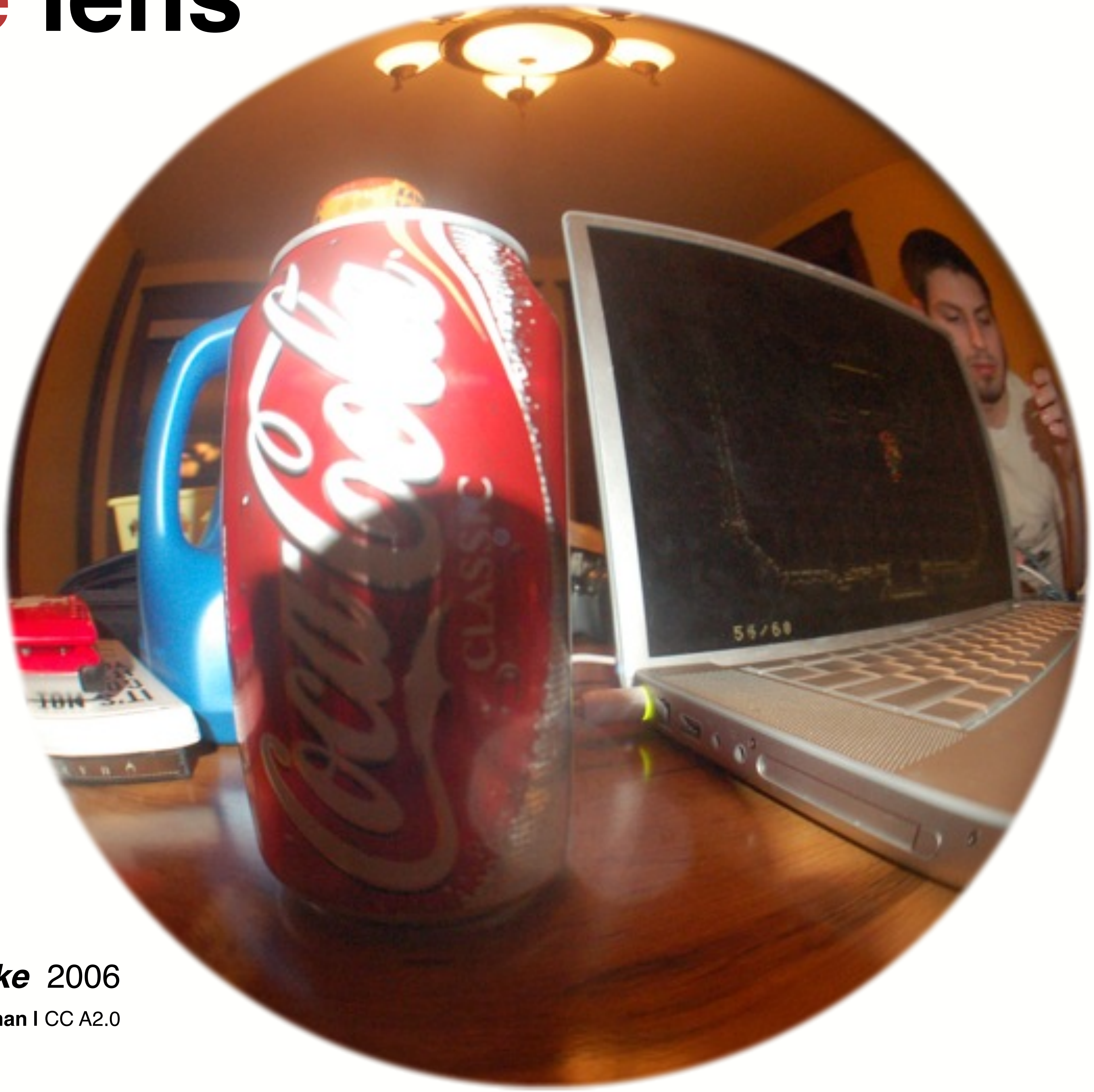
$$q = \mathbf{H}p$$

Corke, P. | Reproduced with permission from Springer Science, & Business Media

```
>>> im.warp_perspective(H).disp()
```


fina.org
arena
Ω OMEGA
NEVO 1
TANCOK 2
MANAUDOU 3
BOVELL 4
TO 5
MOROZOV 6
DERLYUGOV 7
CHEAH 8
DOHA QATAR

Fisheye lens



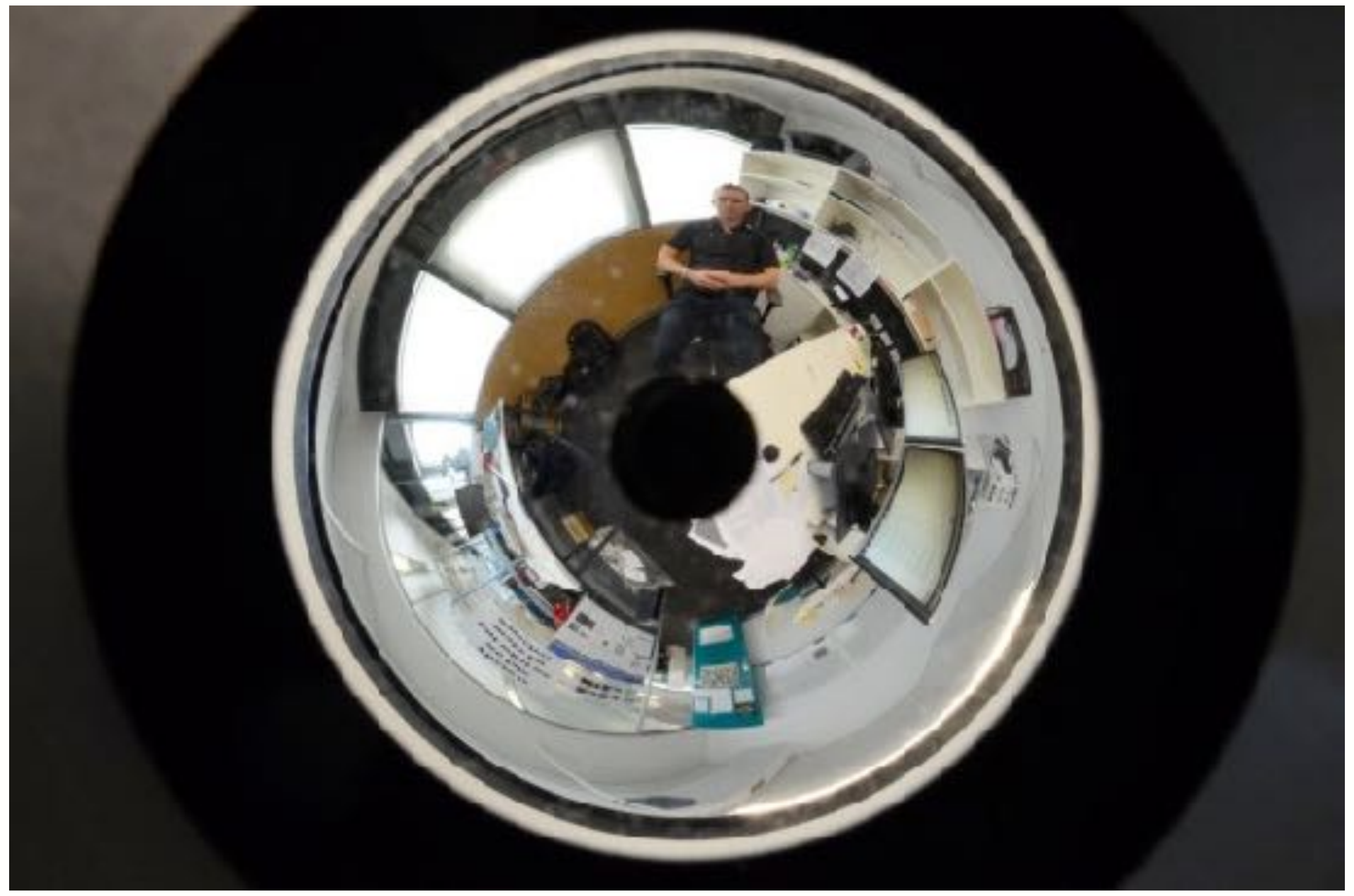
Fisheye coke 2006

Spiratone fisheye lens 2008

Panorama lens

Panorama long 2013

Michael Milford



Panorama round 2013

Michael Milford