UNIVERSITÀ DI PARMA

# Camera Models

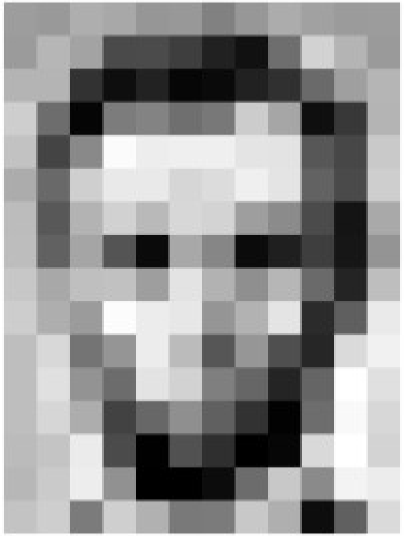
# Summary

- Pin Hole Camera
- Lenses
- Pin-Hole Camera Geometry

Courtesy of *CS231A · Computer Vision: from 3D reconstruction to recognition*, Prof. Silvio Savarese – Stanford University

# Camera Model

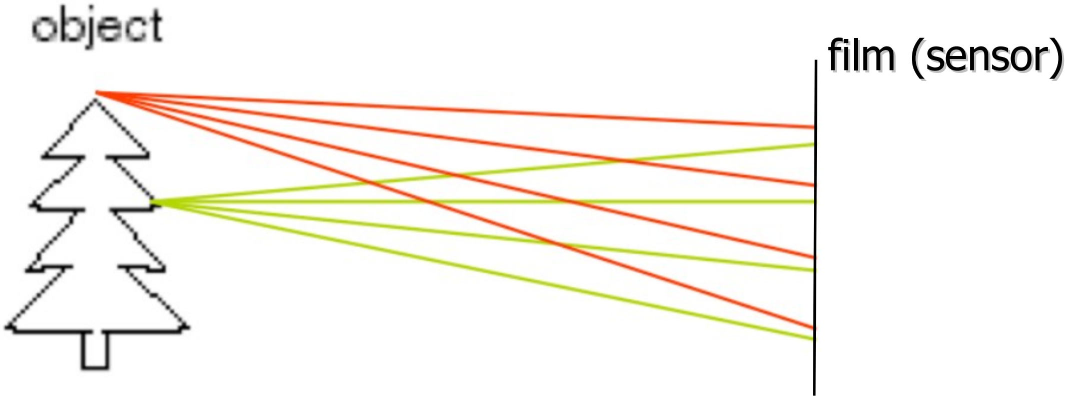
- Since now we discussed image processing
- Namely, we saw fundamental techniques to process a 2D matrix…
- How that image is created?
- What is the relation, if any, to the 3D world?
- During this lesson we will try to answer to those questions



# World and Camera



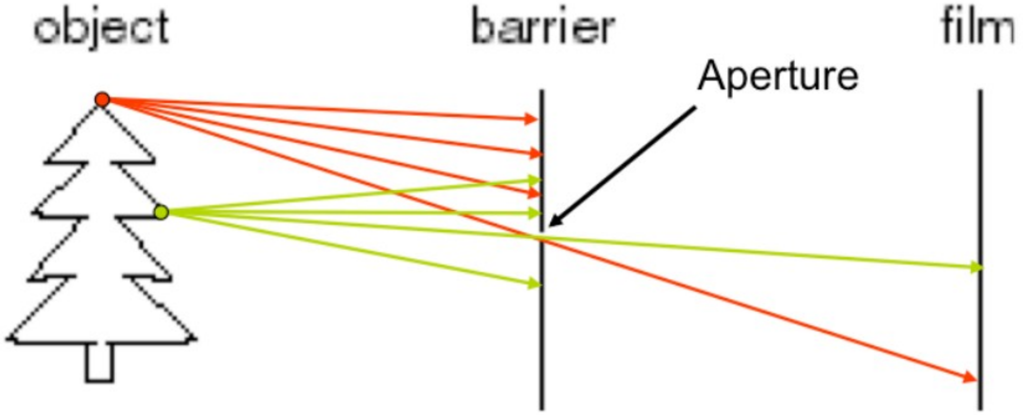
- Too much simple camera
  - A sensitive film in front of an object
  - What is the result?



# Pin-Hole Camera

UNIVERSITÀ DI PARMA

- We can add a barrier with a very small hole
  - So-called pin-hole or aperture
  - Now, only one ray hits the film in a given position

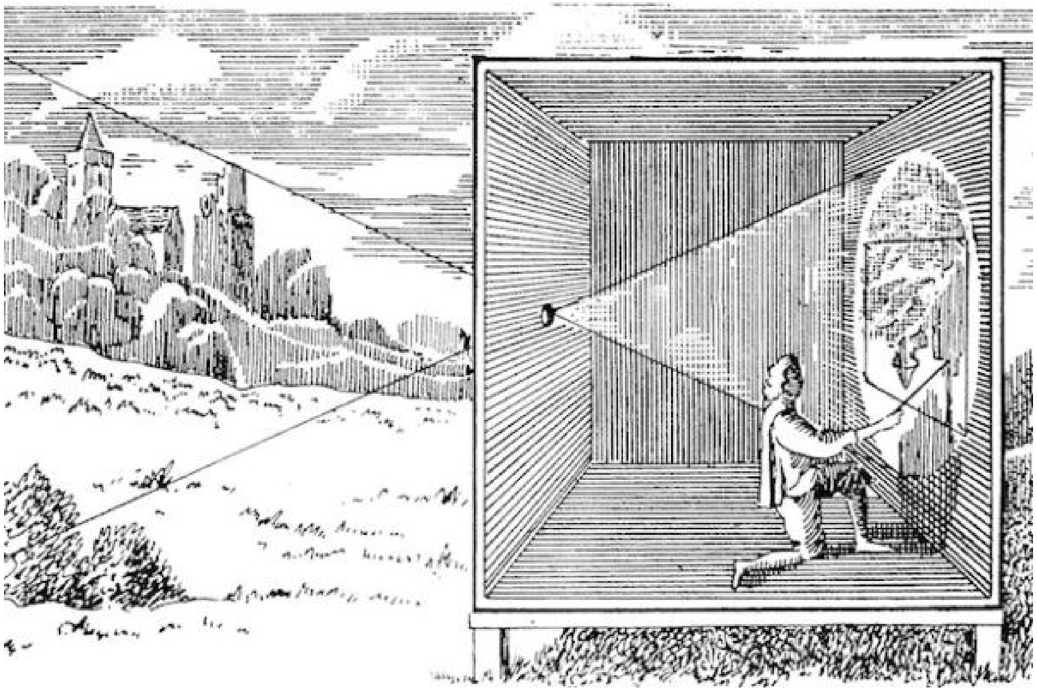


# Fontanellato

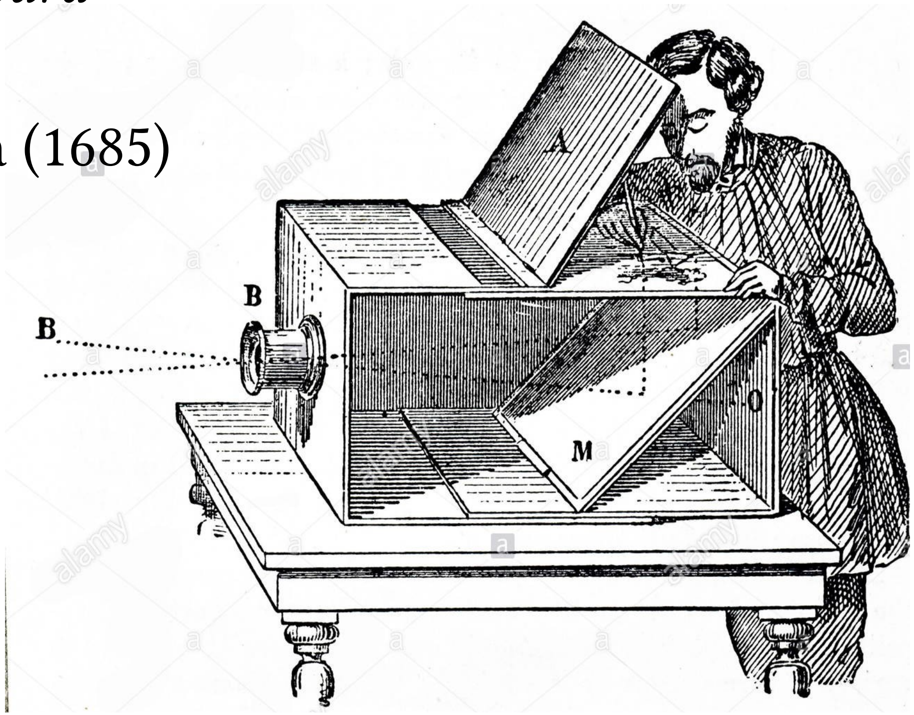


# History

- Milestones
  - Leonardo da Vinci's *Camera Obscura* (1502)

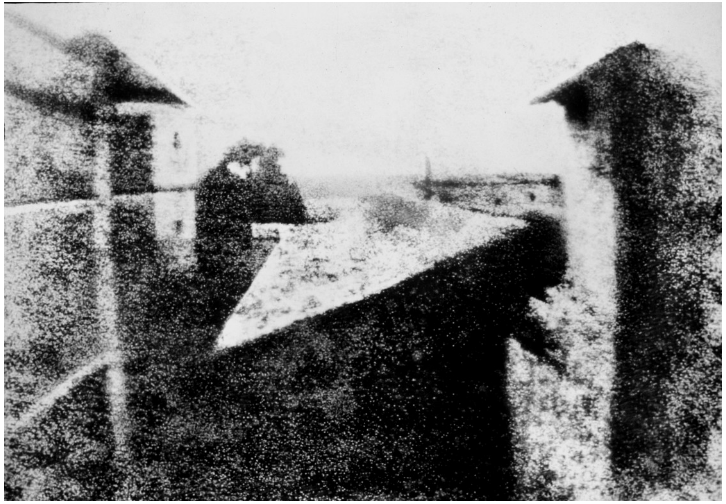


- Milestones
  - Leonardo da Vinci's *Camera Obscura* (1502)
  - Johan Zahn: first portable camera (1685)



# History

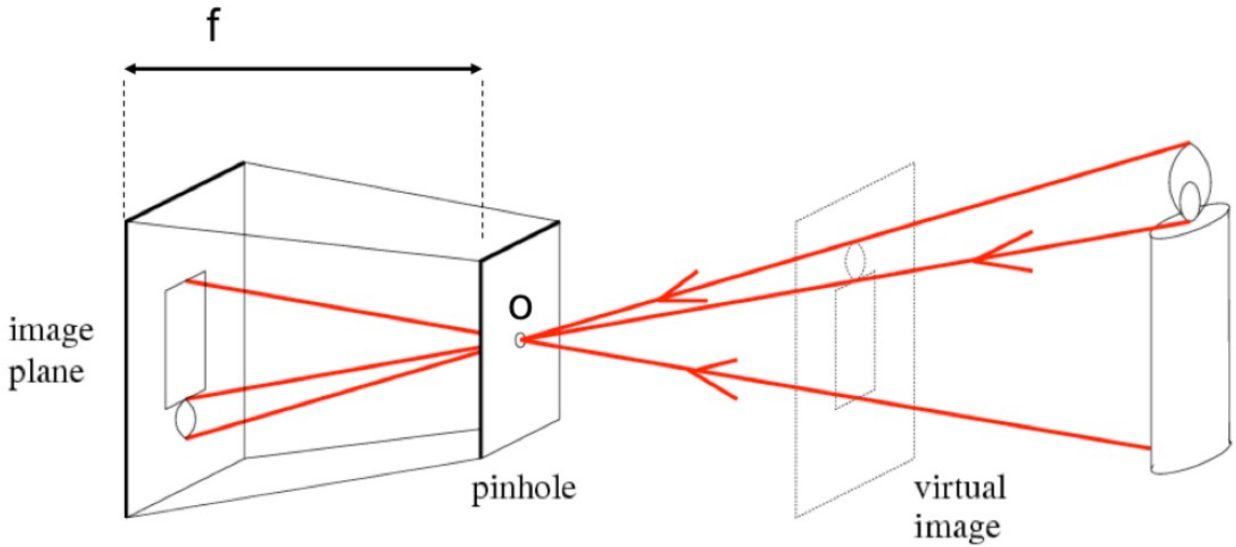
- Milestones
  - Leonardo da Vinci's *Camera Obscura* (1502)
  - Johan Zahn: first portable camera (1685)
  - Joseph Nicéphore Niépce: first photo (1822)



# Pin-Hole Camera

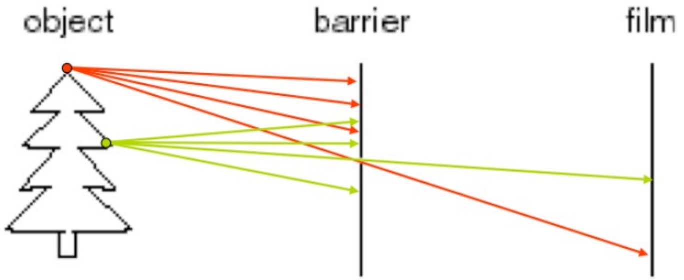
UNIVERSITÀ DI PARMA

- f ⟶ focal lenght
- o ⟶ pin-hole, aperture (center of the lens)



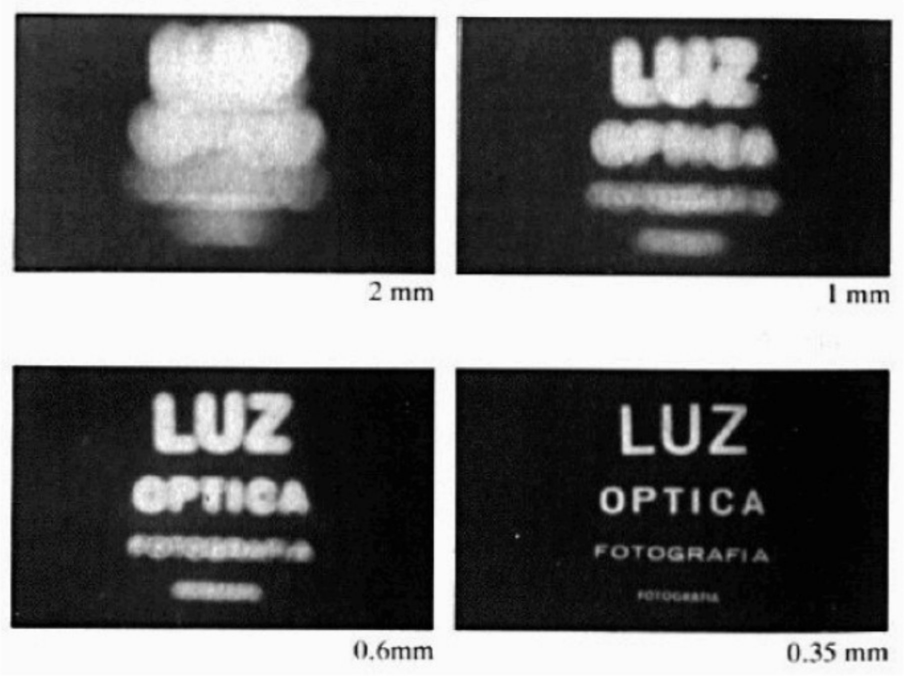
# Camera Aperture size

- The larger the pin-hole, the greater the number of rays
  - More rays ⟶ More light energy
  - More rays ⟶ Blurred image



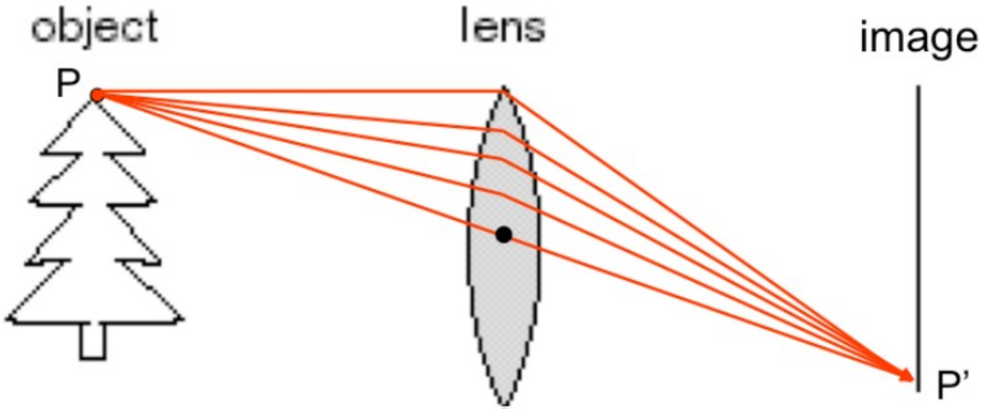

# Camera Aperture size

Aperture Size
decrease



# Lenses

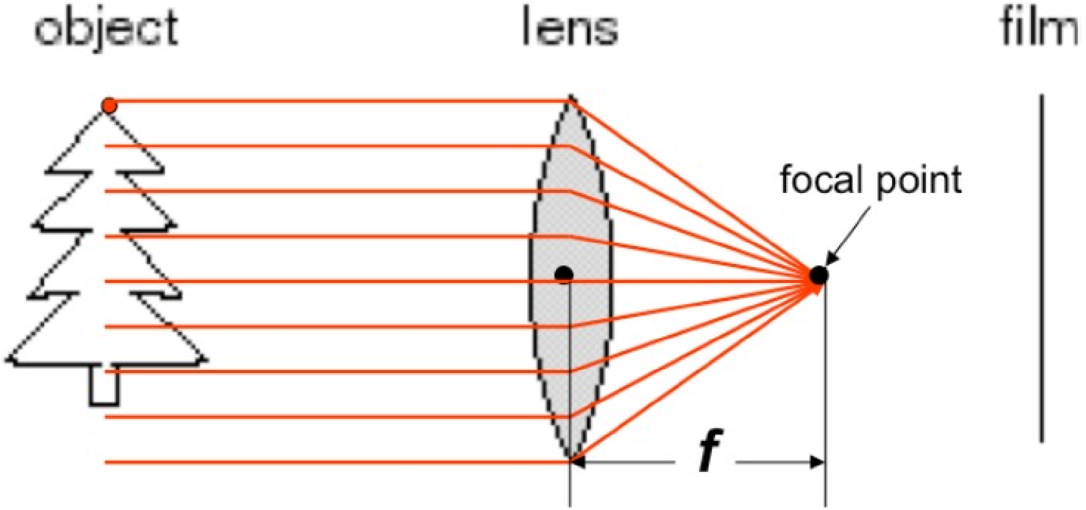
- Actual pin-holes are barely used
- Lenses are much more confortable (some problem anyway)
  - We can intercept more rays coming from same 3D point



# Model with Lenses



- Parallel light rays converge in a specific point
  - Focal point at distance $f$ from the center of the lens
- Only the ray passing through the center of the lens is not deviated

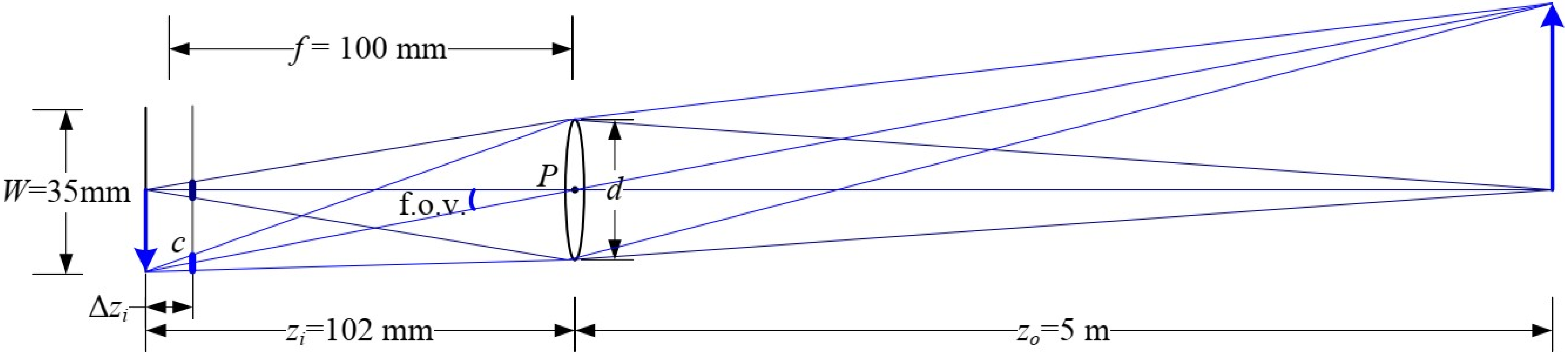


# Model with Lenses

UNIVERSITÀ DI PARMA

- Rays emitted from a given plane points in the world converge to points that lie on a specific plane
- The following formula can be used:
  - Varying $z_0$ means that also $z_i$ is modified

$$\frac{1}{f} = \frac{1}{z_i} + \frac{1}{z_0}$$

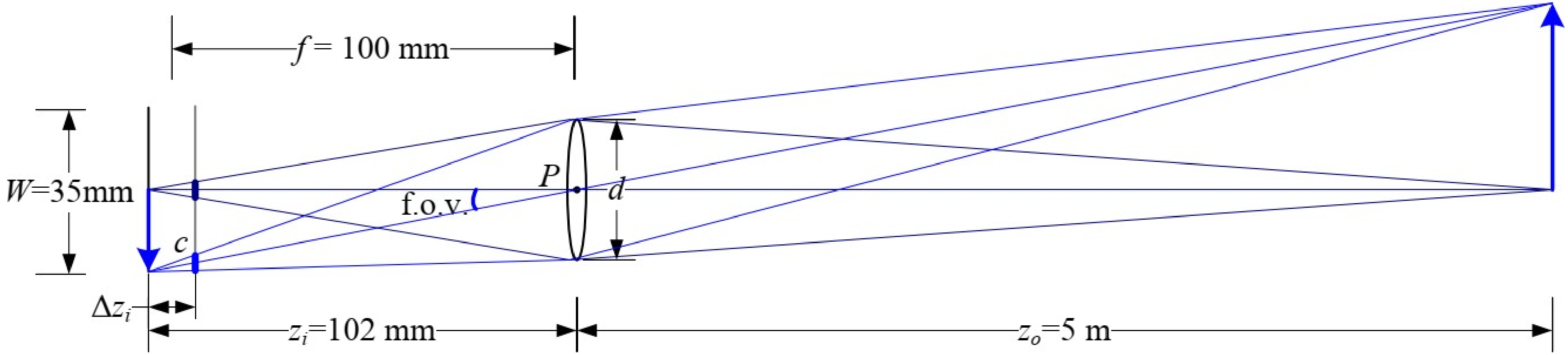


# Model with Lenses



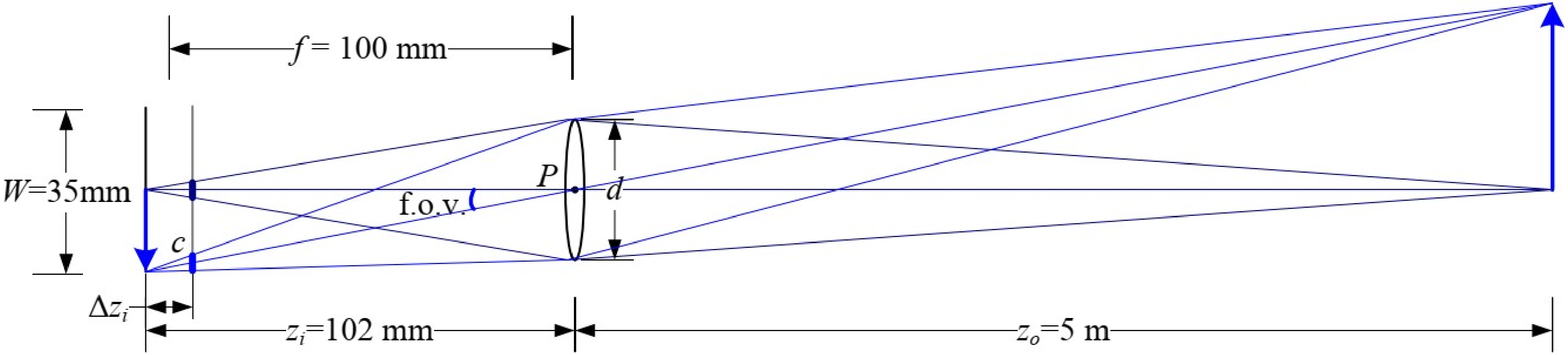
- Therefore the distance between lens and sensor give us the *perfect* focus distance
- For other points we have a **circle of confusion**

$$\frac{1}{f} = \frac{1}{z_i} + \frac{1}{z_0}$$



# Model with Lenses

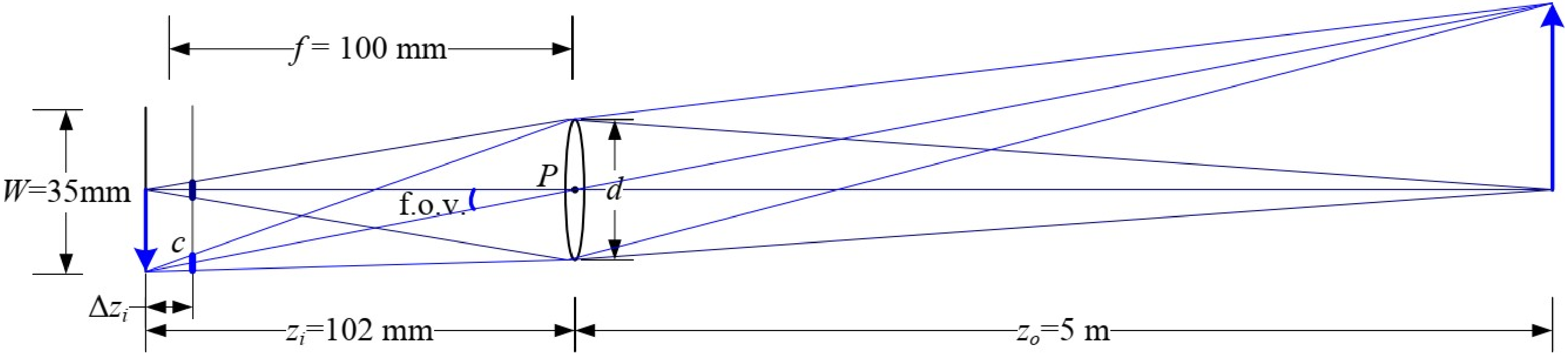
- Anyway in the real world sensor elements have a finite size
- Then we can consider a small portion of the world to be sufficiently in focus: **Shallow Depth of Field**



# Model with Lenses

- The depth of field depends on the f-number or focal ratio (f/#)
- The higher the f-number, the larger the depth of field

$$f/\# = \frac{f}{d}$$



# Model with Lenses

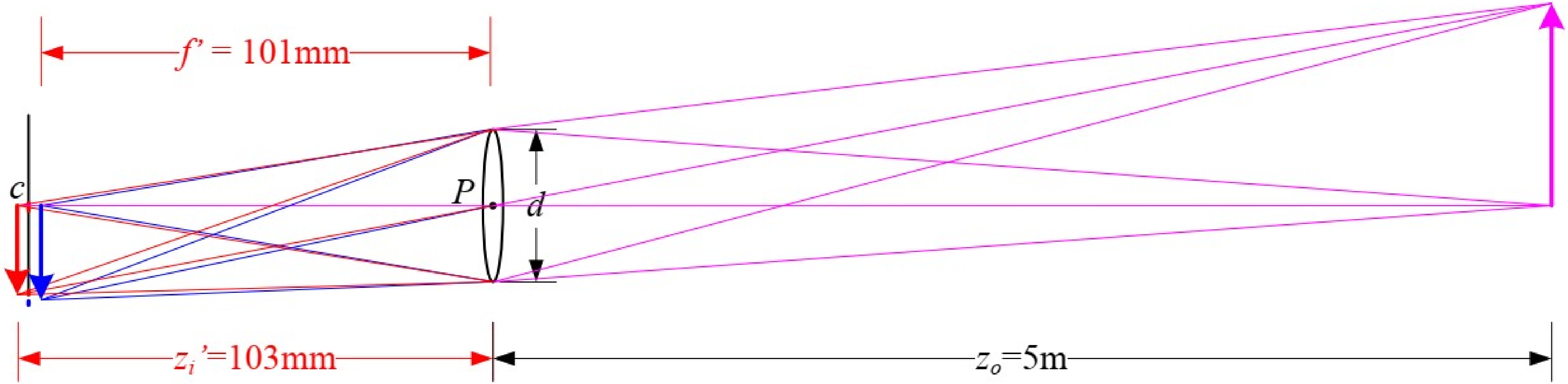
UNIVERSITÀ DI PARMA

- Example of shallow depth of field

# Model with Lenses

UNIVERSITÀ DI PARMA

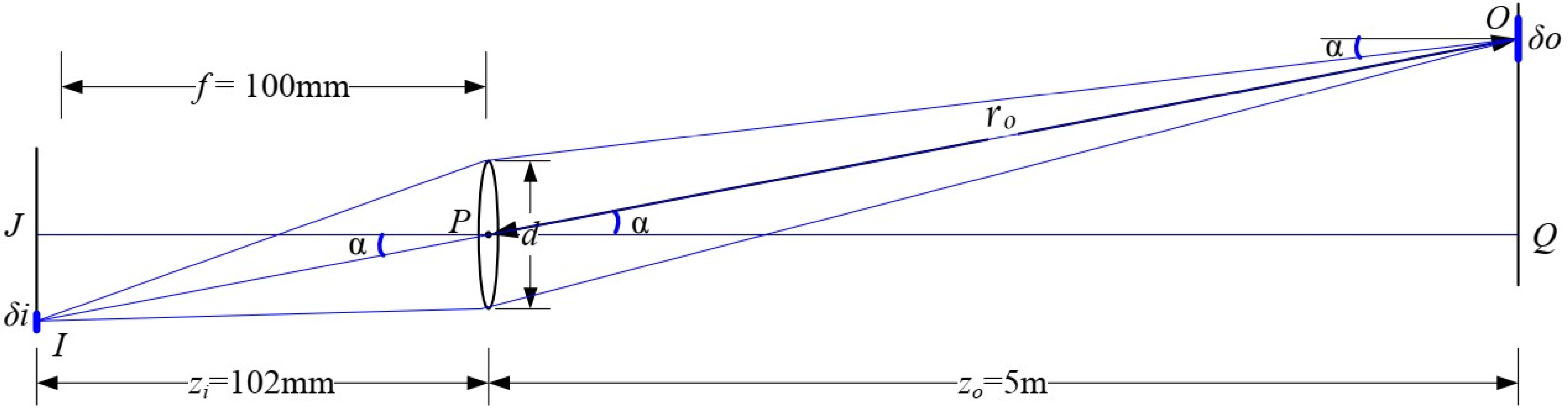
- Refractive index depends on wavelenght
- Different colors are then projected in different positions
- **Chromatic Aberration**



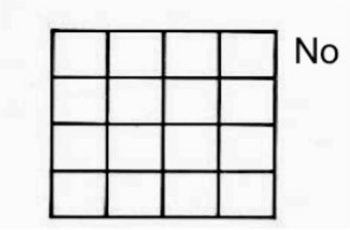
# Model with Lenses



- Areas in the borders/corners are typically darker...
- Distance from optical axis affects energy
  - Vignetting

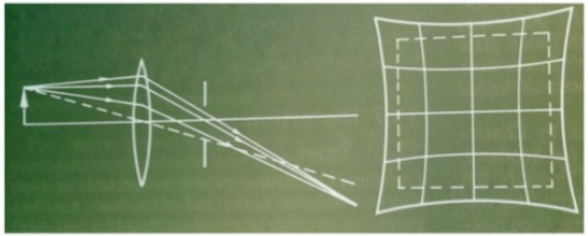


# Distortion



No distortion

Pin cushion



Barrel (fisheye lens)

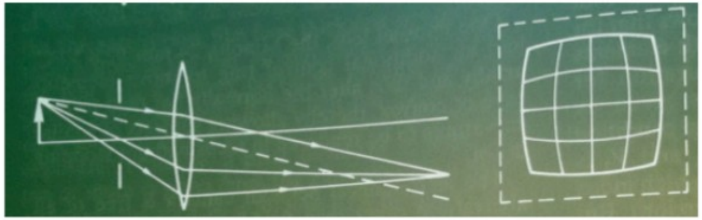
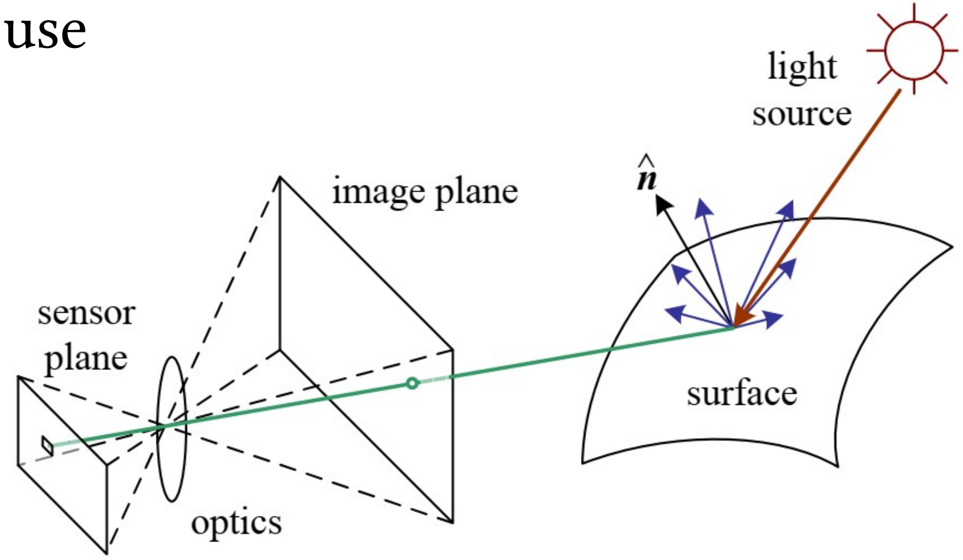



Image magnification decreases with distance from the optical axis

- Distortion effect is much more evident in lateral areas

# Pin-Hole Camera

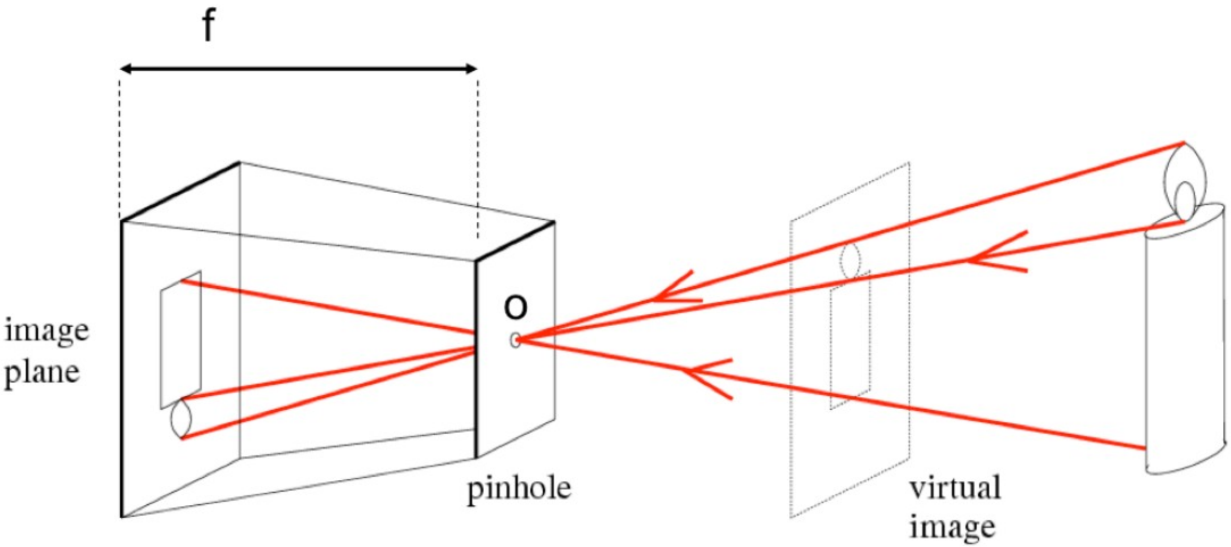
- We saw many issues related to the use of a lens
- This also affects the camera model
- Anyway in the following we will simply assume to have:
  - Thin lenses
  - Small angles of view
  - No or compensated chromatic aberration
  - No or rectified distortion effects



# Pin-Hole Camera

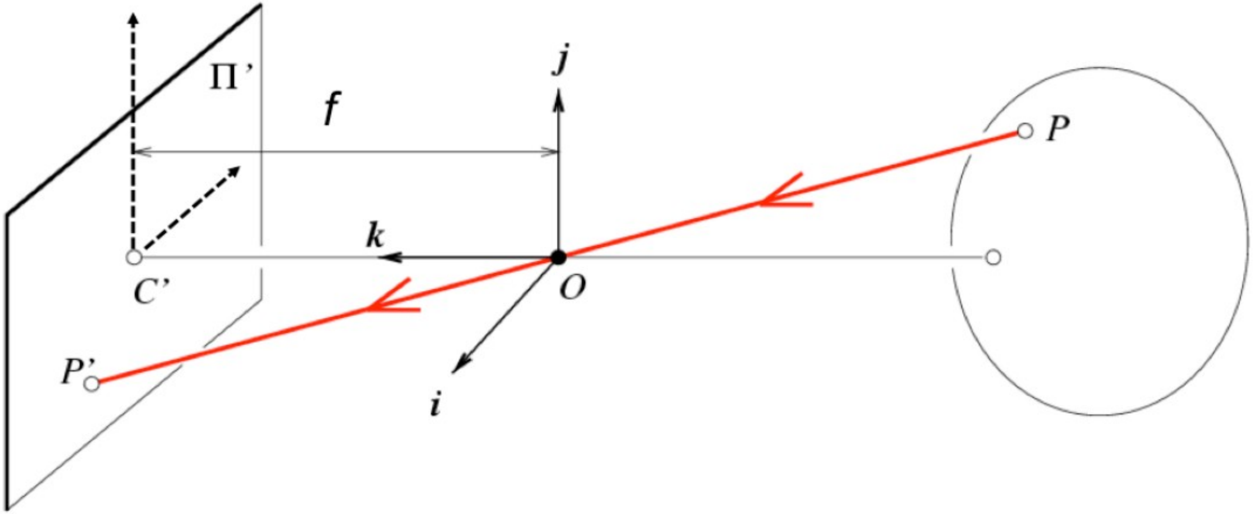
UNIVERSITÀ DI PARMA

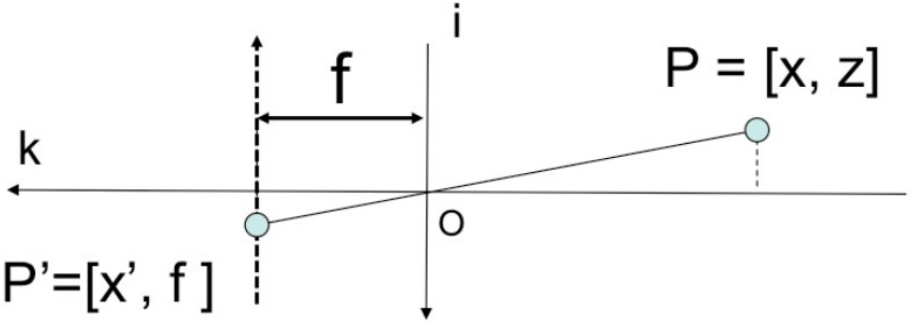
- f ⟶ focal lenght
- o ⟶ pin-hole, aperture (center of the lens)



# Pin-Hole Camera





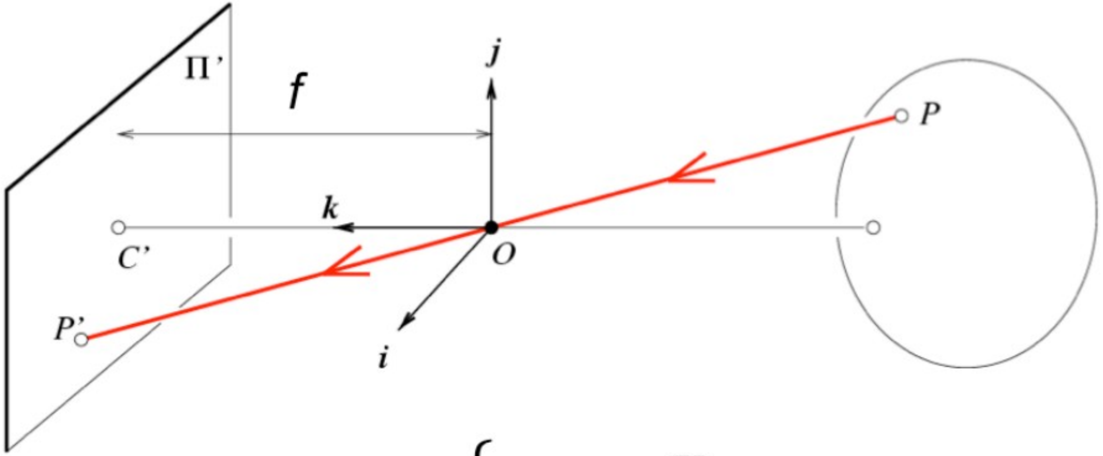

[Eq. 2]

$$\frac{x'}{f} = \frac{x}{z}$$

Courtesy: Silvio Savarese

# Pin-Hole Camera: perspective tranformation






$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow P' = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

[Eq. 1]

$$\begin{cases} x' = f \dfrac{x}{z} \\ y' = f \dfrac{y}{z} \end{cases}$$

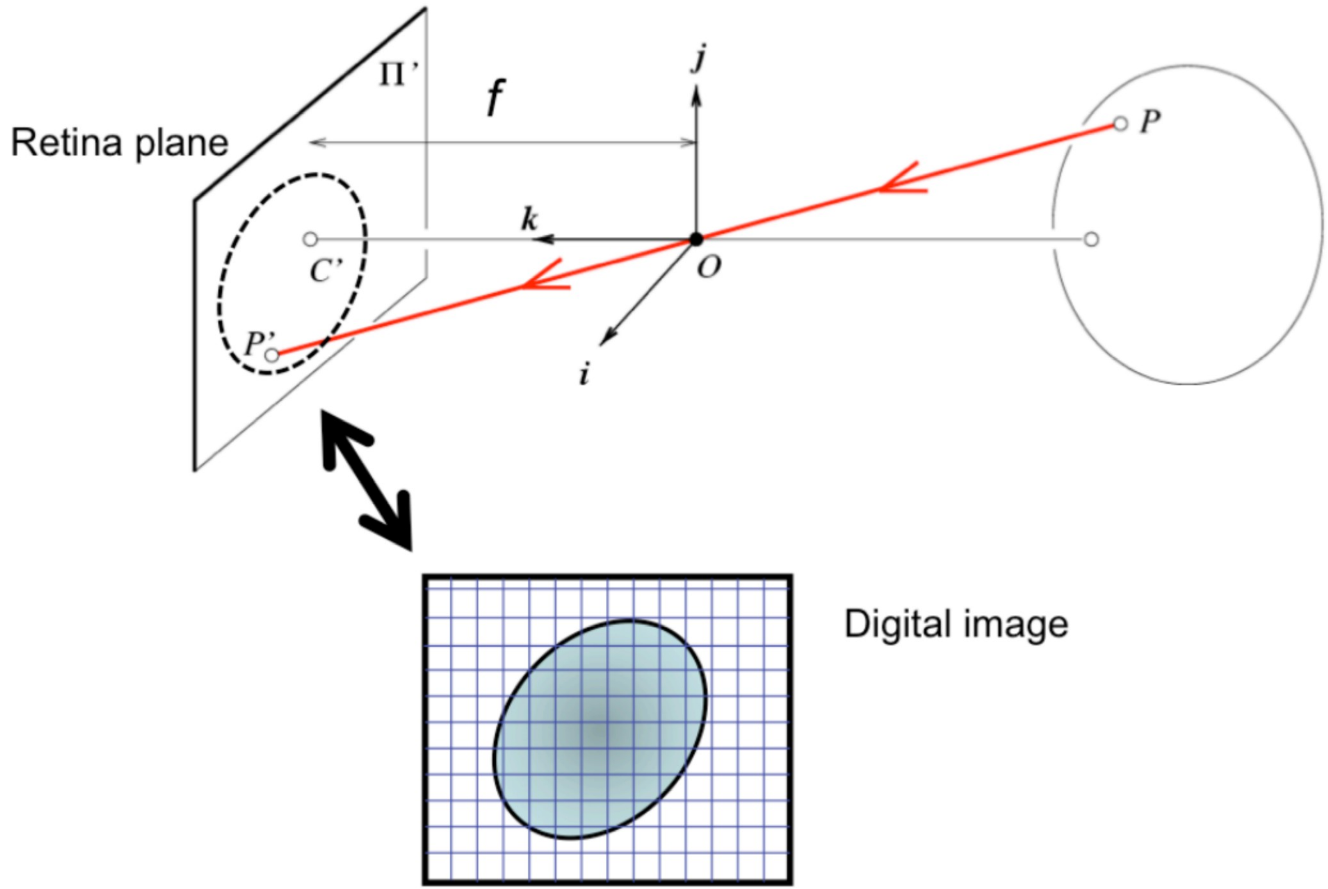
$$\Re^3 \xrightarrow{E} \Re^2$$

f = focal length
o = center of the camera

Courtesy: Silvio Savarese

# Image & Sensor Planes

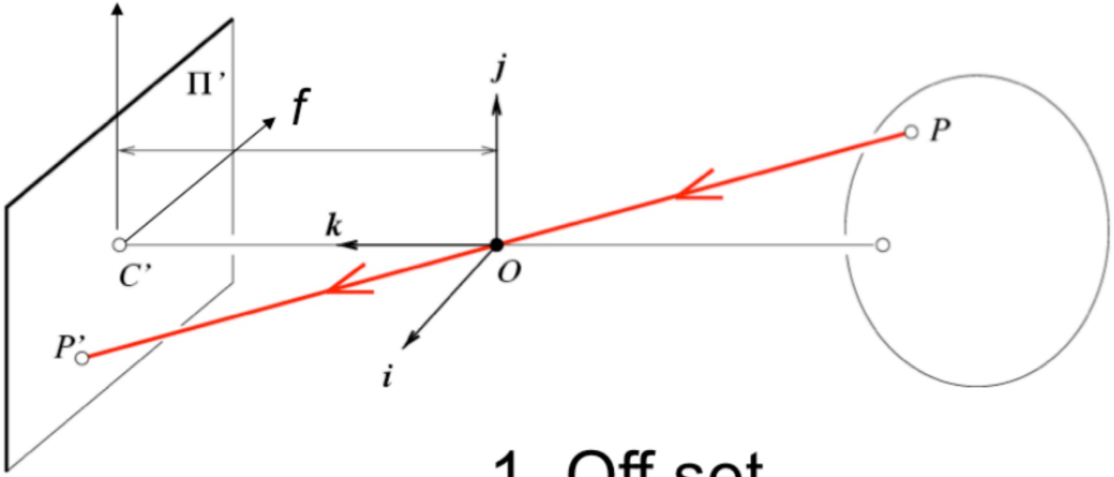




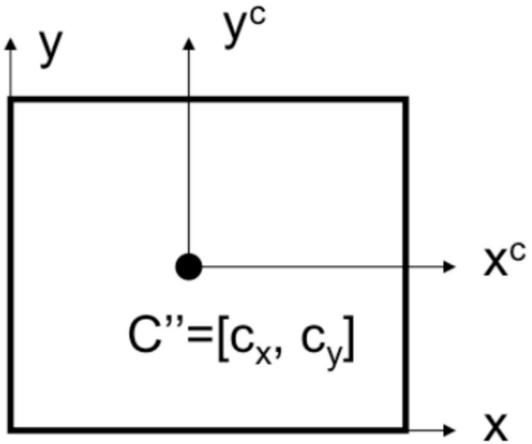

Pixels, bottom-left coordinate systems

Courtesy: Silvio Savarese

# Image & Sensor Planes → Origin




1. Off set

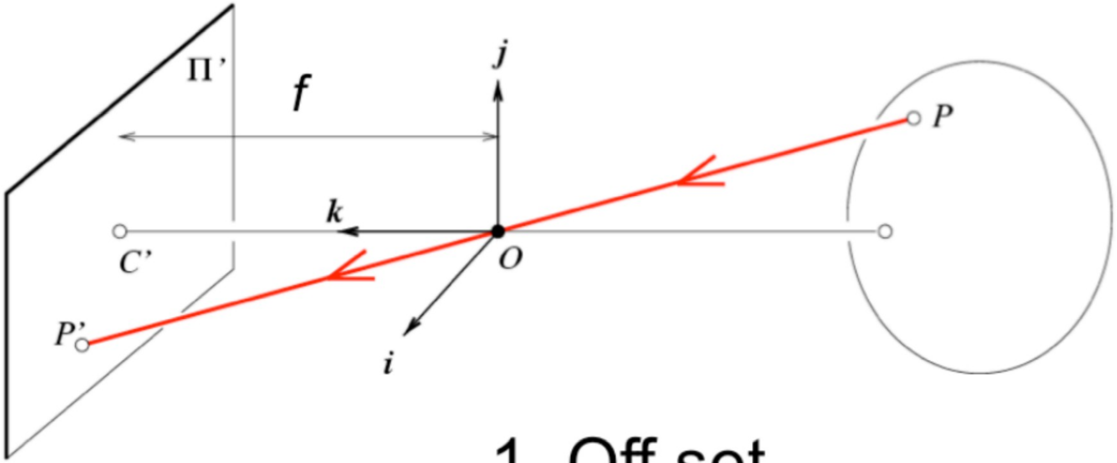


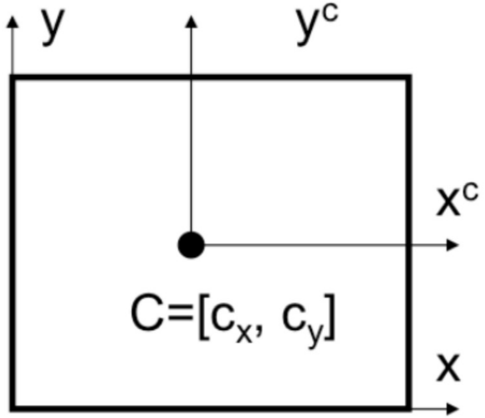

$$(x, y, z) \rightarrow (f\frac{x}{z} + c_x, f\frac{y}{z} + c_y)$$

[Eq. 5]

Courtesy: Silvio Savarese

# Image & Sensor Planes → Pixel size




1. Off set
2. From metric to pixels

$$(x, y, z) \rightarrow (\underbrace{\boxed{f\,k}}_{\alpha} \frac{x}{z} + c_x, \underbrace{\boxed{f\,l}}_{\beta} \frac{y}{z} + c_y)$$

[Eq. 6]

Units: $k,l$ : pixel/m

$f$ : m

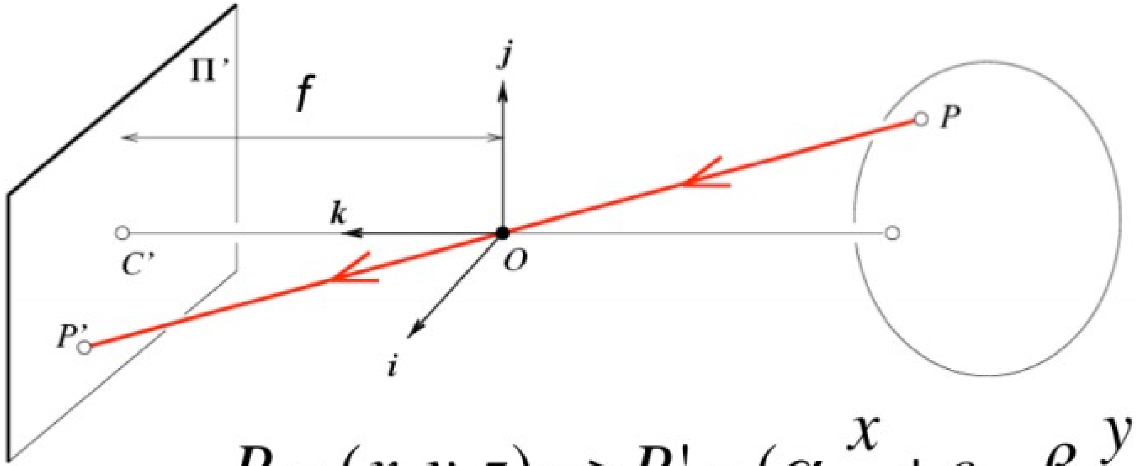
Non-square pixels

$\alpha, \beta$ : pixel

Courtesy: Silvio Savarese

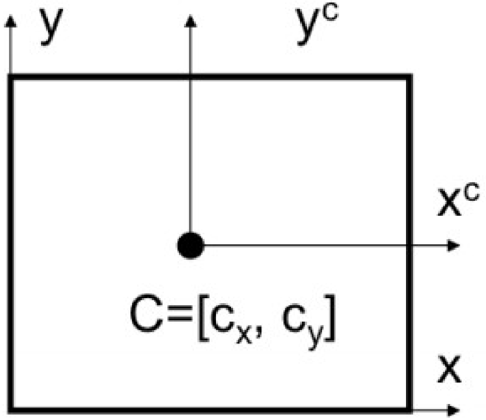
# Non Linear Transformation






$$P = (x, y, z) \rightarrow P' = \left(\alpha \frac{x}{z} + c_x, \beta \frac{y}{z} + c_y\right)$$

[Eq. 7]




- Is this a linear transformation?
  No — division by z is nonlinear
- Can we express it in a matrix form?

Courtesy: Silvio Savarese

# Homogeneous Coordinates



- The non linearity can be solved using Homogeneous Coordinates
- HC are an augmented representation of points
- We add another “coordinate”, i.e. $\mathbb{R}^n \longrightarrow \mathbb{R}^{n+1}$
- In 2D space P=(x,y) can be represented as P=(x,y,1)
  - Or more generally as (kx,ky,k)
  - The third value can be considered as a scale factor

# Homogeneous vs Euclidean

UNIVERSITÀ DI PARMA

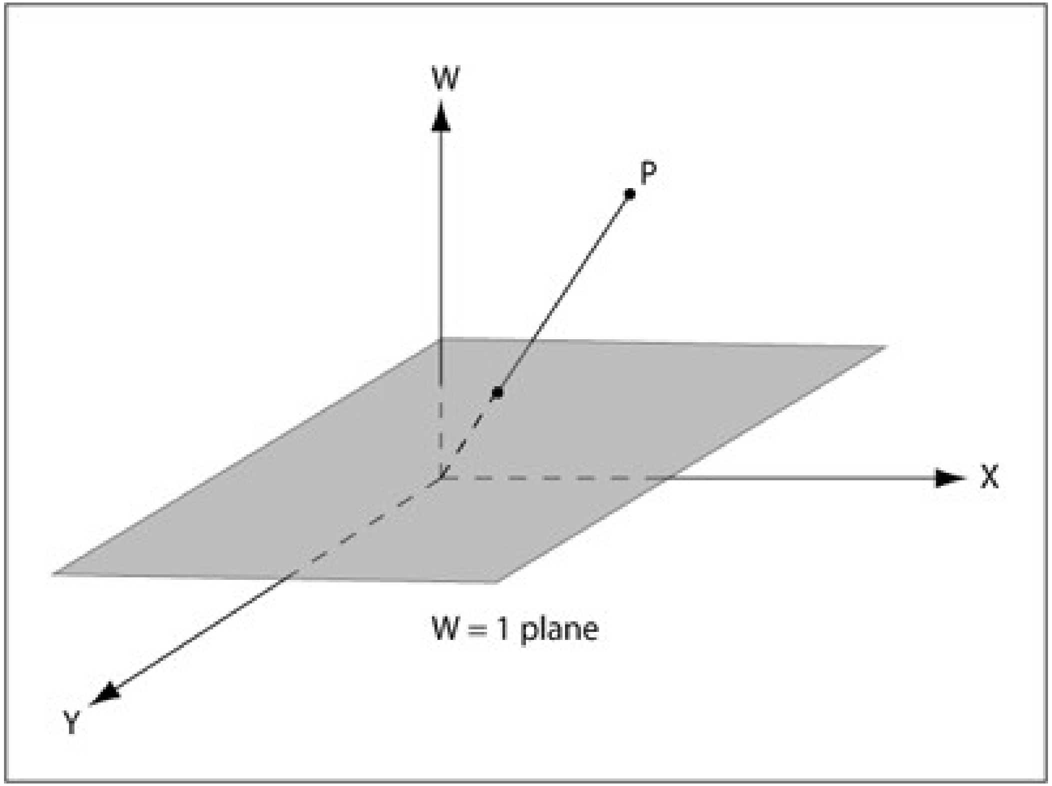
- Conversions are simple:
- Euclidean $\longrightarrow$ Homogeneous

$$(x,y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad (x,y,z) \Rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

- Homogeneous $\longrightarrow$ Euclidean

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w) \qquad \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \Rightarrow (x/w, y/w, z/w)$$

- A geometric interpretation for HC can be given as follows

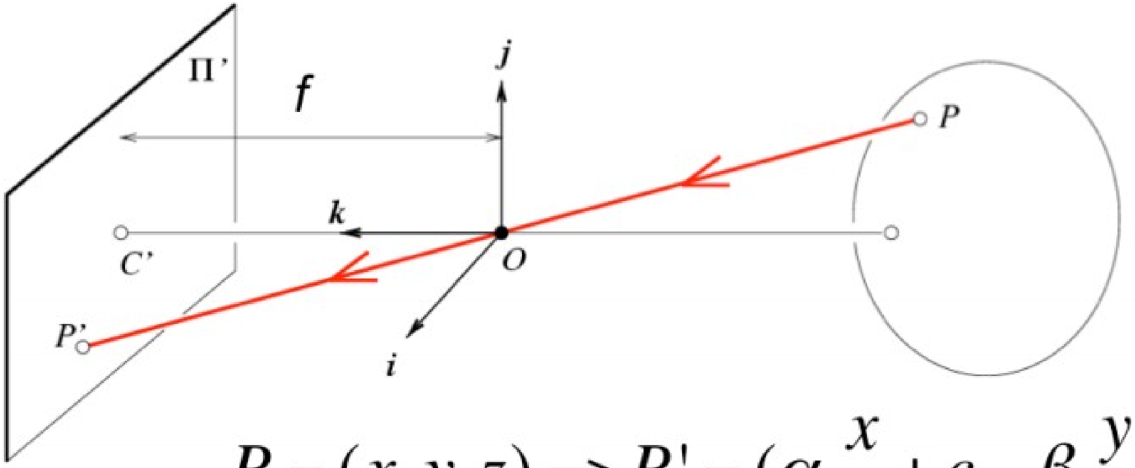


# Homogeneous Coordinates

- Three main reasons to use homogeneous coordinates
  - Simple representation of points and lines (no special cases)
    - Homogenous space contains more points than Euclidean one!
    - (x,y,0)
  - Simple representation of Euclidean Transformations
    - Traslation
    - Scale
    - Rotation
  - Simple representation of perspective projections

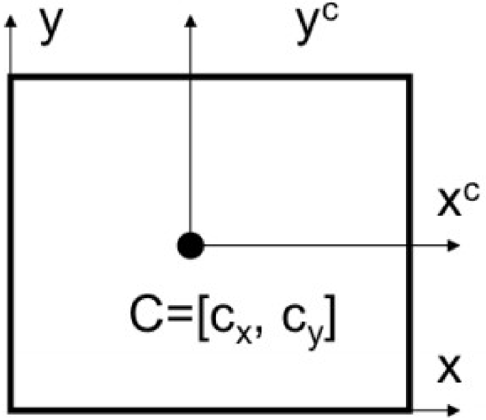
# Non Linear Transformation (again)






$$P = (x, y, z) \rightarrow P' = (\alpha \frac{x}{z} + c_x, \beta \frac{y}{z} + c_y)$$

[Eq. 7]




- Is this a linear transformation?

  No — division by z is nonlinear

- Can we express it in a matrix form?

Courtesy: Silvio Savarese

# Perspective Transformation

UNIVERSITÀ DI PARMA

- $P \longrightarrow P'$ projection becomes $P_h \longrightarrow P_h'$
- The $P=[x\ y\ z]$ in the 3D space is $P_h=[x\ y\ z\ 1]$ in Homogeneous reference system
- $P'$ was computed as $[\alpha(x/z)+c_x \quad \beta(y/z)+c_y]$
- $P_h'$ can be then $[\alpha x+c_x z \quad \beta y+c_y z \quad z]$
- Do you see how to express $P_h \longrightarrow P_h'$ as matrix product?

# Perspective Linear Transformation



$$P_h' = \begin{bmatrix} \alpha x + c_x z \\ \beta y + c_y z \\ z \end{bmatrix} = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}}_{P_h}$$

[Eq.8]

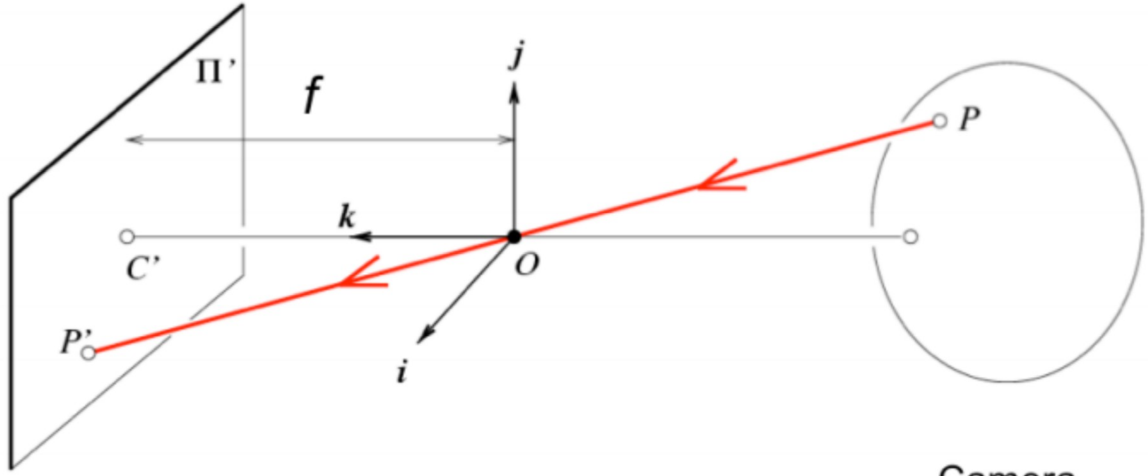
$$\overbrace{P_h'}^{\text{Homogenous}} \rightarrow \overbrace{P' = (\alpha \frac{x}{z} + c_x, \beta \frac{y}{z} + c_y)}^{\text{Euclidian}}$$

$$M = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Courtesy: Silvio Savarese

# The Intrinsic Matrix






Camera matrix K

[Eq.9]

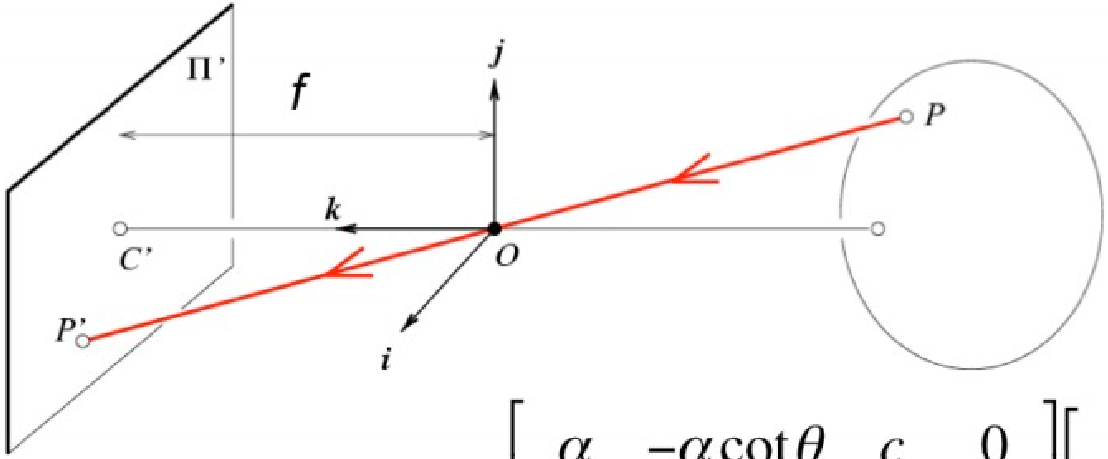
$$P' = M\,P$$

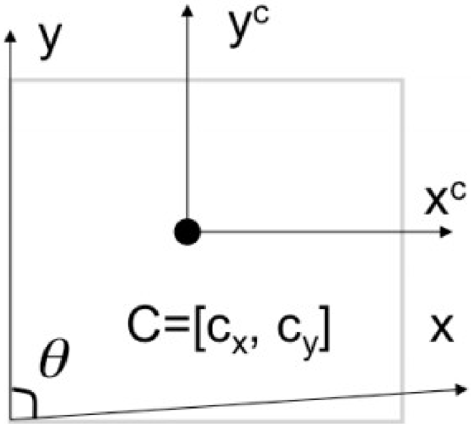
$$= K\begin{bmatrix} I & 0 \end{bmatrix} P$$

$$P' = \begin{bmatrix} \alpha & 0 & c_x & 0 \\ 0 & \beta & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Courtesy: Silvio Savarese

# Skewness



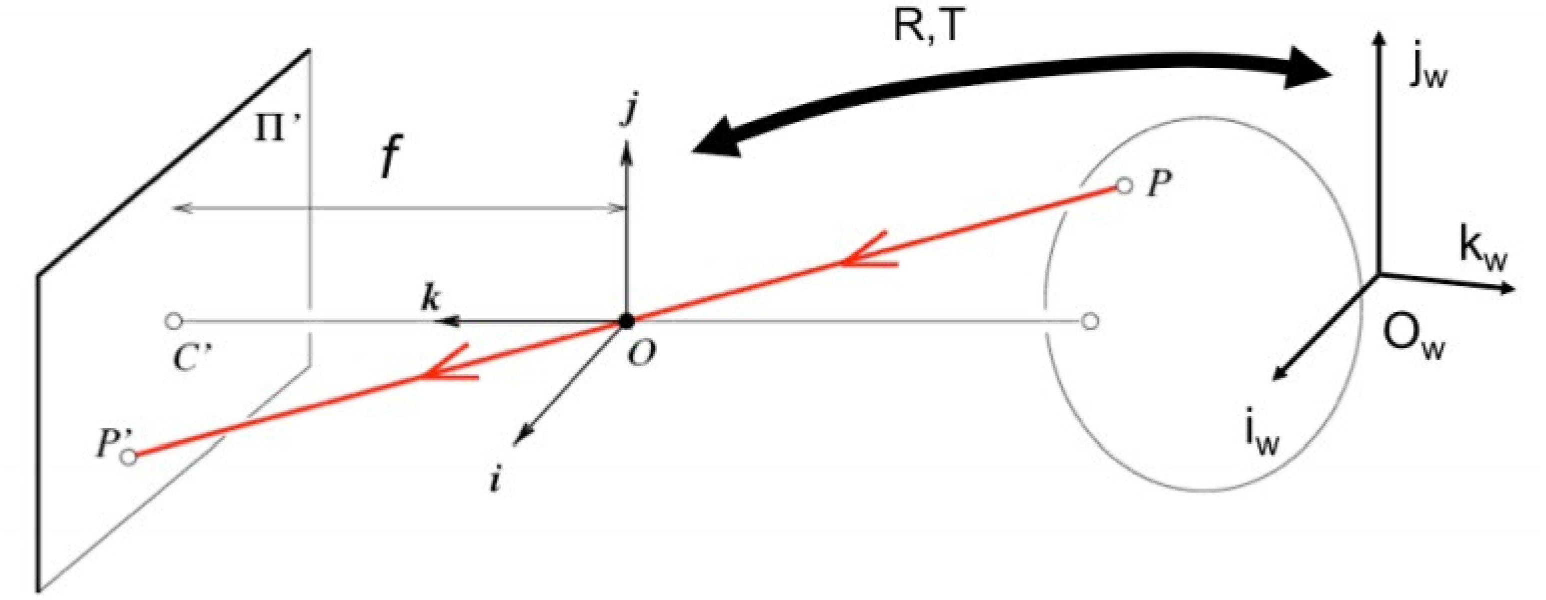

$$P' = \begin{bmatrix} \alpha & -\alpha\cot\theta & c_x & 0 \\ 0 & \dfrac{\beta}{\sin\theta} & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

How many degree does K have?
5 degrees of freedom!

Courtesy: Silvio Savarese

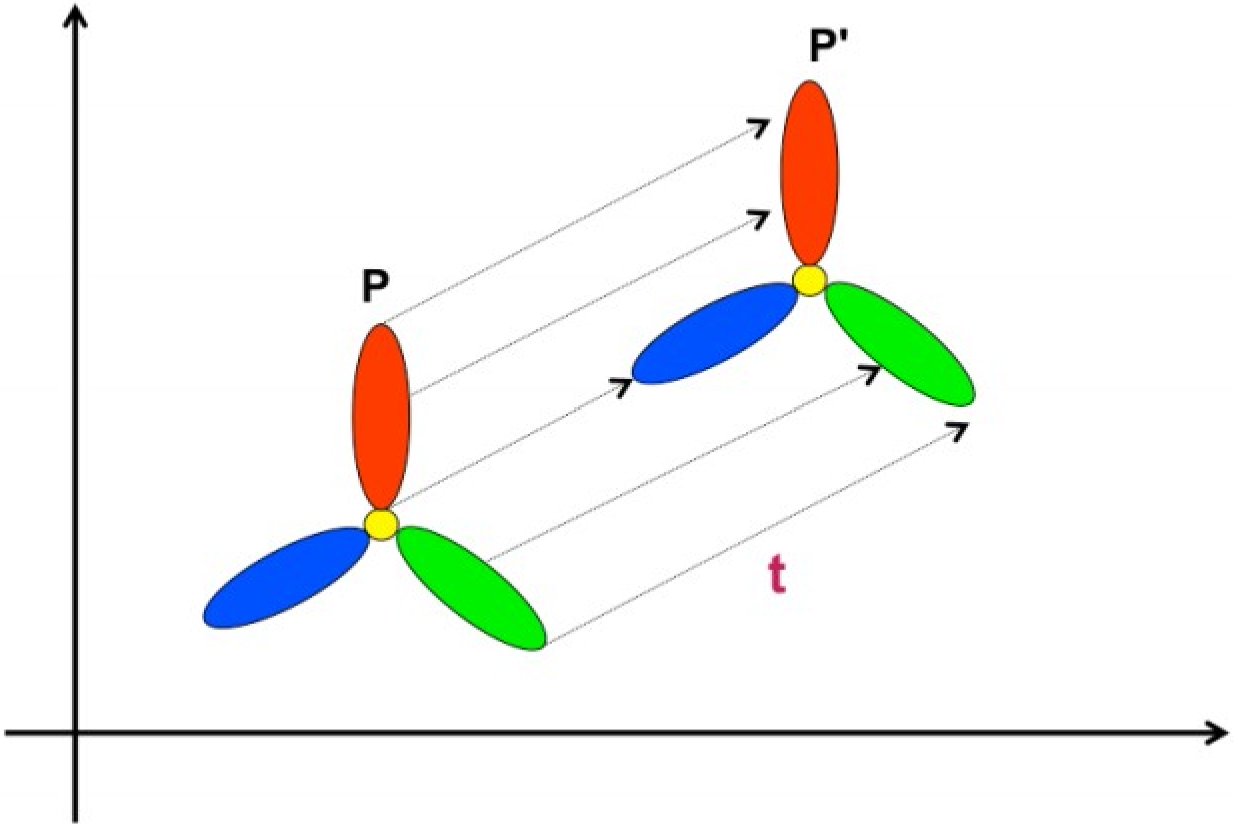
# Introducing the external world...

- Having a camera reference system is a bit limiting
- Usually a different reference system is used
- We need an additional transformation



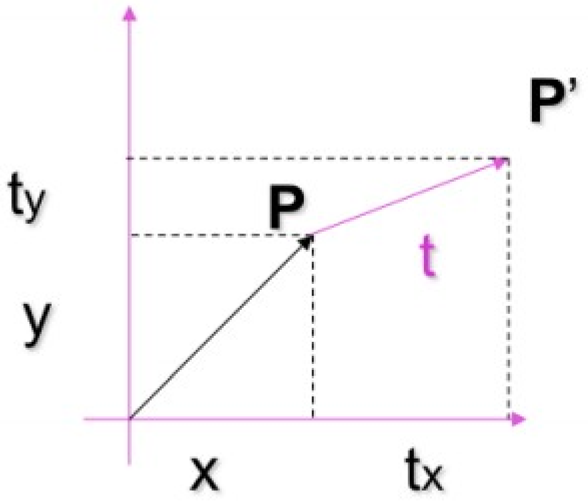

Courtesy: Silvio Savarese

# Review: 2D translation




Courtesy: Silvio Savarese

# Review: 2D translation



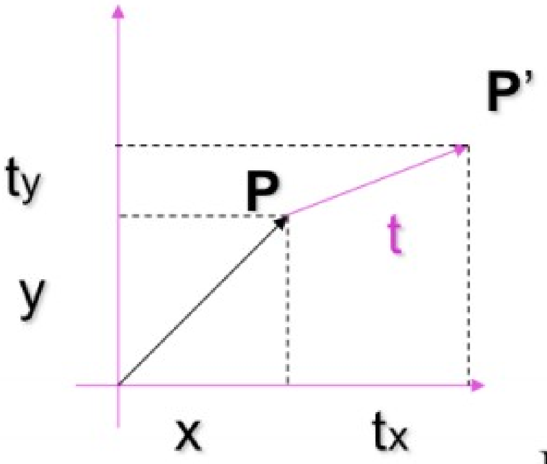

$$\mathbf{P} = (x, y)$$

$$\mathbf{t} = (t_x, t_y)$$

$$\mathbf{P}' = \mathbf{P} + \mathbf{t} = (x + t_x, y + t_y)$$

Courtesy: Silvio Savarese

# Review: homogeneous 2D translation



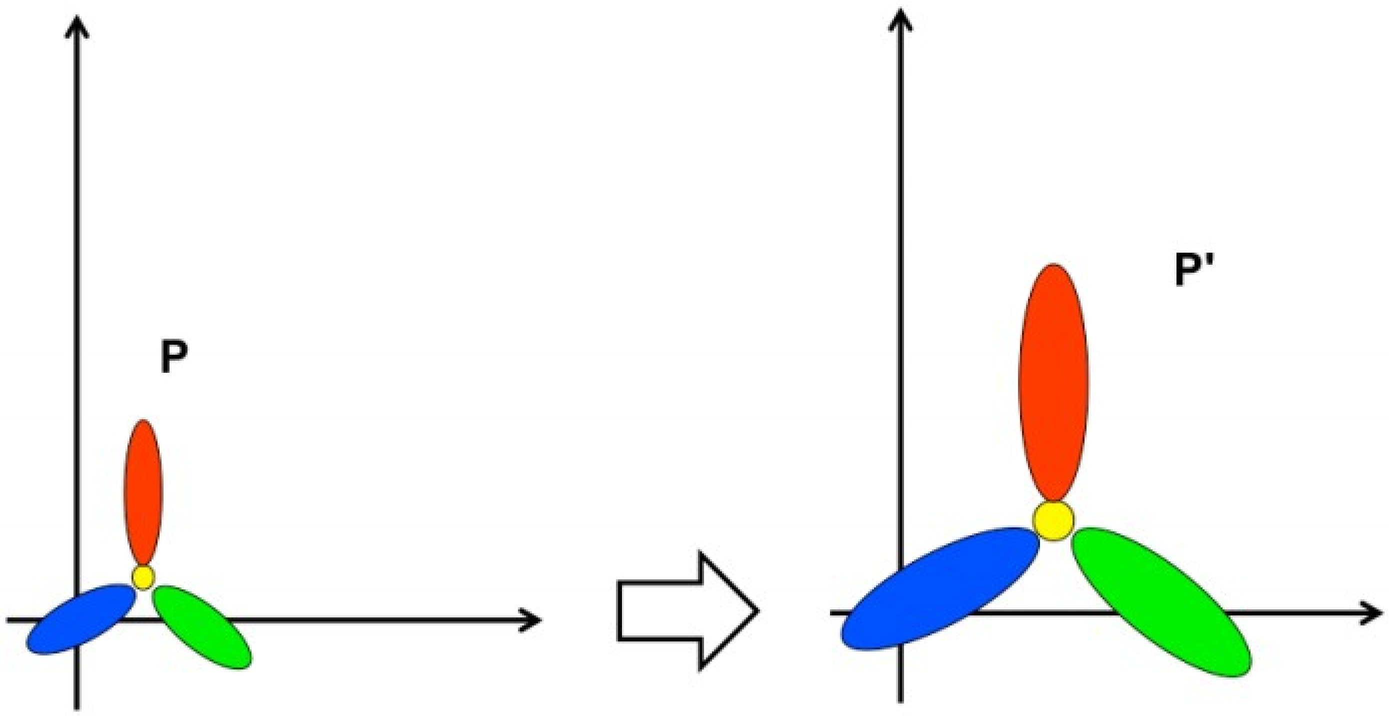

$$\mathbf{P} = (x, y) \rightarrow (x, y, 1)$$

$$\mathbf{P}' \rightarrow \begin{bmatrix} x + t_x \\ y + t_y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{I} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{T} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

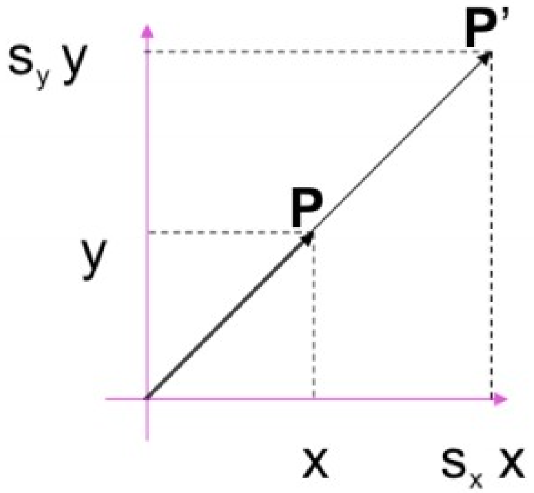
Courtesy: Silvio Savarese

# Review: 2D scaling




Courtesy: Silvio Savarese

# Review: homogeneous 2D scaling



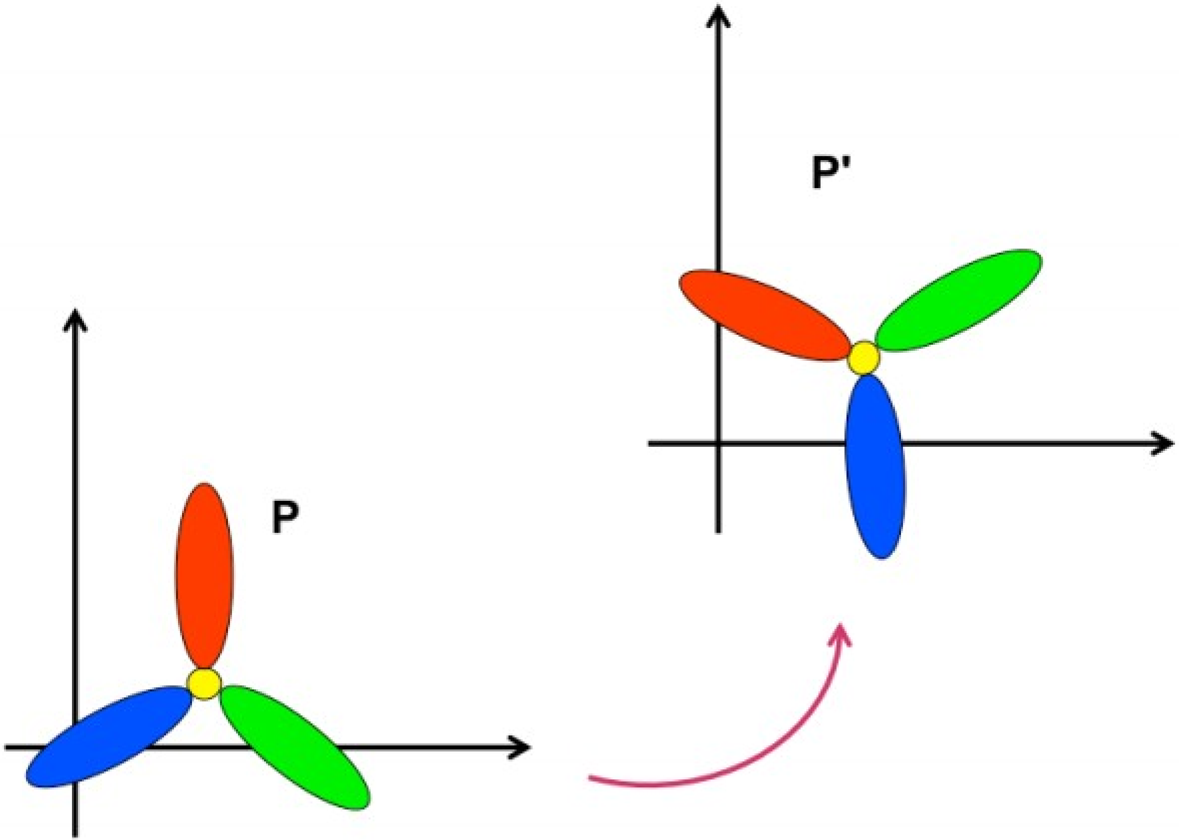

$$\mathbf{P} = (x, y) \rightarrow \mathbf{P}' = (s_x x, s_y y)$$

$$\mathbf{P} = (x, y) \rightarrow (x, y, 1)$$

$$\mathbf{P}' \rightarrow \begin{bmatrix} s_x x \\ s_y y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{S}} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{S}' & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{S} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Courtesy: Silvio Savarese

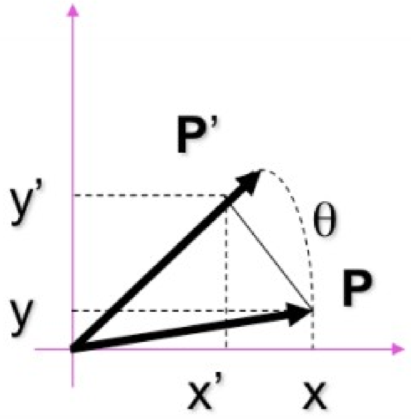
# Review: 2D rotation




Courtesy: Silvio Savarese

# Review: 2D rotation

- Rotate around the *z* axis by $\Theta$




$$x' = \cos\theta\, x - \sin\theta\, y$$

$$y' = \cos\theta\, y + \sin\theta\, x$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \qquad \mathbf{P' = R\ P}$$

How many degrees of freedom? 1

$$\mathbf{P'} \rightarrow \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Courtesy: Silvio Savarese

# Review: put everything togheter

$$\mathbf{P}' \rightarrow \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

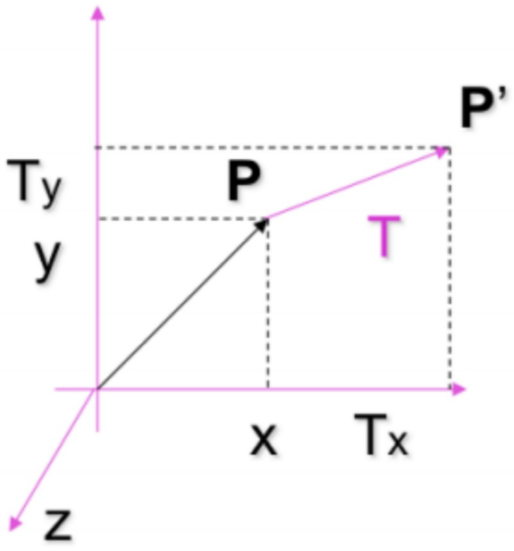
$$= \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \boxed{\begin{matrix} \mathbf{R}\,\mathbf{S} & \mathbf{t} \\ \mathbf{0} & 1 \end{matrix}} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

If $s_x = s_y$, this is a similarity transformation

Courtesy: Silvio Savarese

# Review: 3D translation




$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

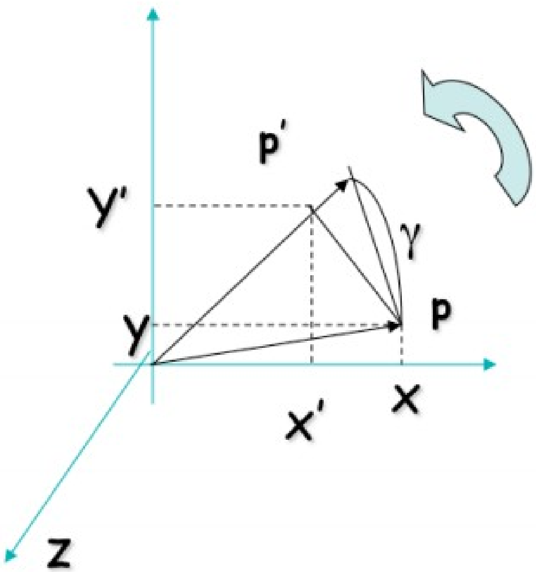
$$P' \rightarrow \begin{bmatrix} \mathbf{I} & T \\ 0 & 1 \end{bmatrix}_{4\times 4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Courtesy: Silvio Savarese

A translation vector in 3D has 3 degrees of freedom

# Review: 3D rotation (Euler)

Rotation around the coordinate axes, counter-clockwise:




$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$P' \rightarrow \begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix}_{4\times4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Courtesy: Silvio Savarese

A rotation matrix in 3D has 3 degree of freedom

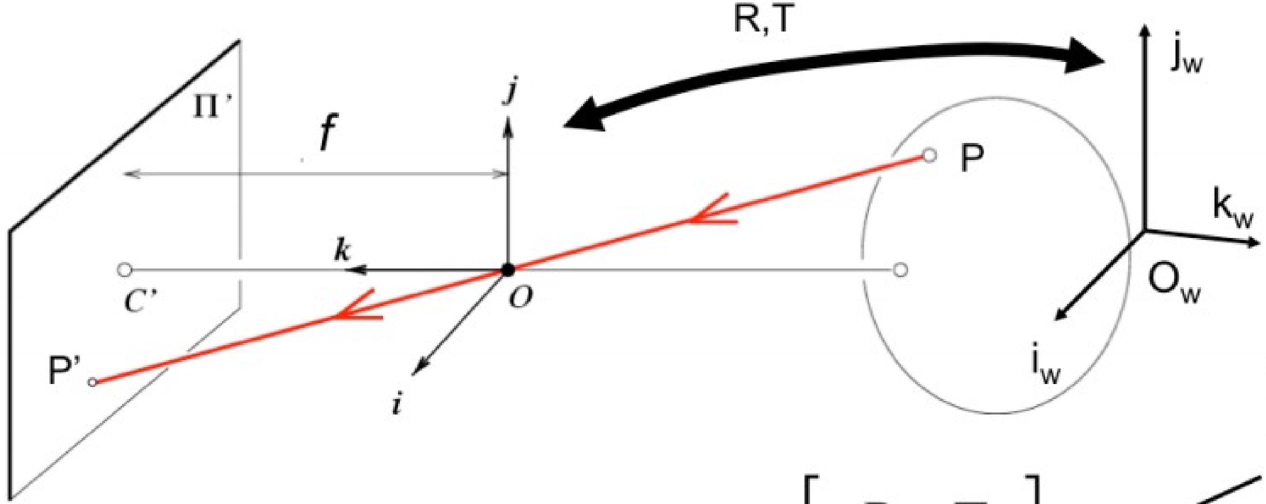
# Review: 3D rotation (Euler) & translation

$$R = R_x(\alpha)\ R_y(\beta)\ R_z(\gamma) \qquad T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

$$P' \rightarrow \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4\times4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Courtesy: Silvio Savarese

# World Reference System




In 4D homogeneous coordinates: $P = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4\times4} P_w \qquad \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$

Internal parameters

External parameters

$$P' = K\begin{bmatrix} I & 0 \end{bmatrix} P = K\begin{bmatrix} I & 0 \end{bmatrix}\begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4\times4} P_w \qquad P' = \underbrace{\boxed{K\begin{bmatrix} R & T \end{bmatrix}}}_{M} P_w$$

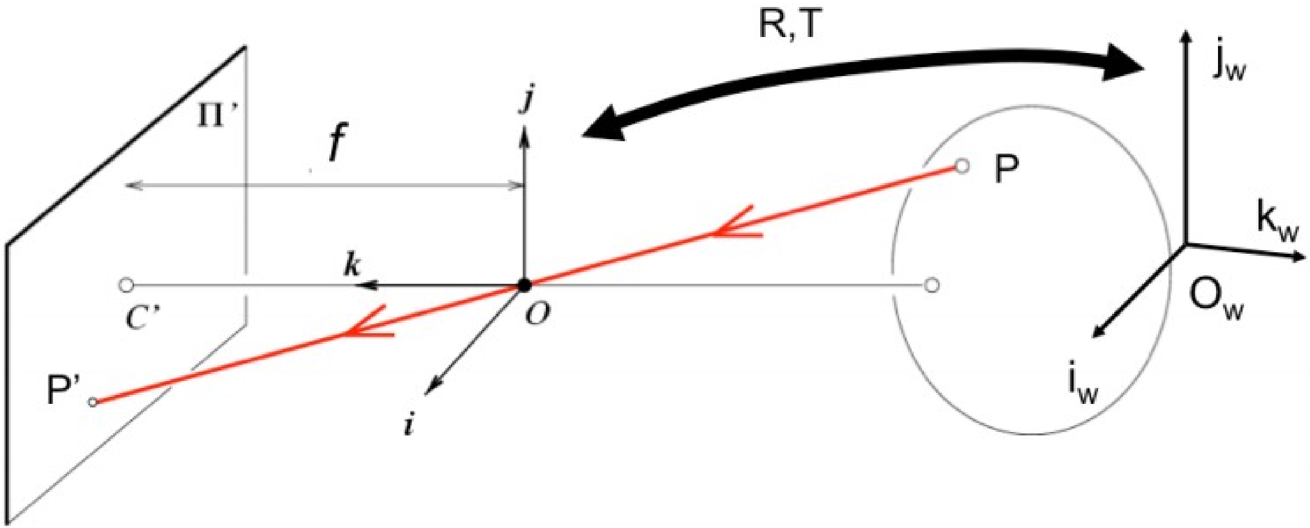
[Eq.11]

Courtesy: Silvio Savarese

# World Reference System



- 11 degrees of freedom (5 + 3 + 3)

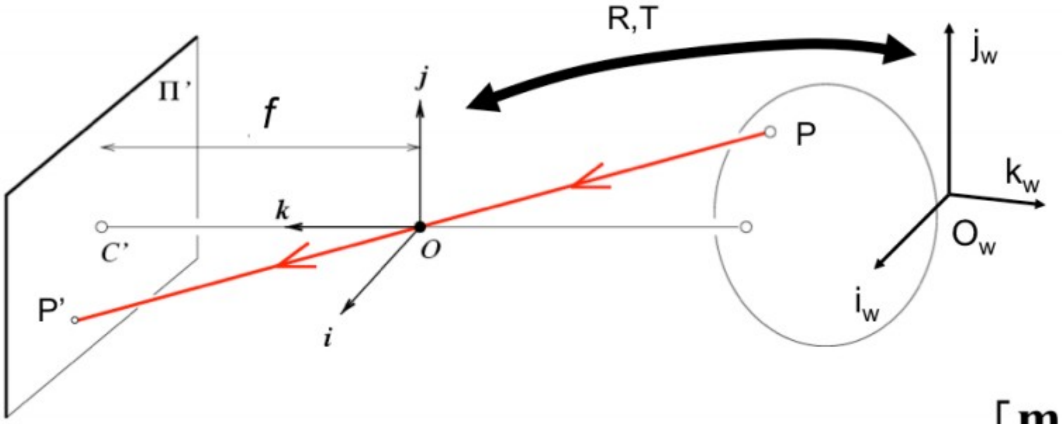



$$P'_{3\times1} = M_{3x4}\, P_w = K_{3\times3} \begin{bmatrix} R & T \end{bmatrix}_{3\times4} P_{w\,4\times1}$$

Courtesy: Silvio Savarese

# World Reference System

- Back to Euclidean coordinates




$$P'_{3\times1} = M\,P_w = K_{3\times3}\begin{bmatrix} R & T \end{bmatrix}_{3\times4} P_{w\,4\times1} \qquad M = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{bmatrix} P_W = \begin{bmatrix} \mathbf{m}_1 P_W \\ \mathbf{m}_2 P_W \\ \mathbf{m}_3 P_W \end{bmatrix} \xrightarrow{E} \left(\frac{\mathbf{m}_1 P_w}{\mathbf{m}_3 P_w}, \frac{\mathbf{m}_2 P_w}{\mathbf{m}_3 P_w}\right) \quad \text{[Eq.12]}$$

Courtesy: Silvio Savarese

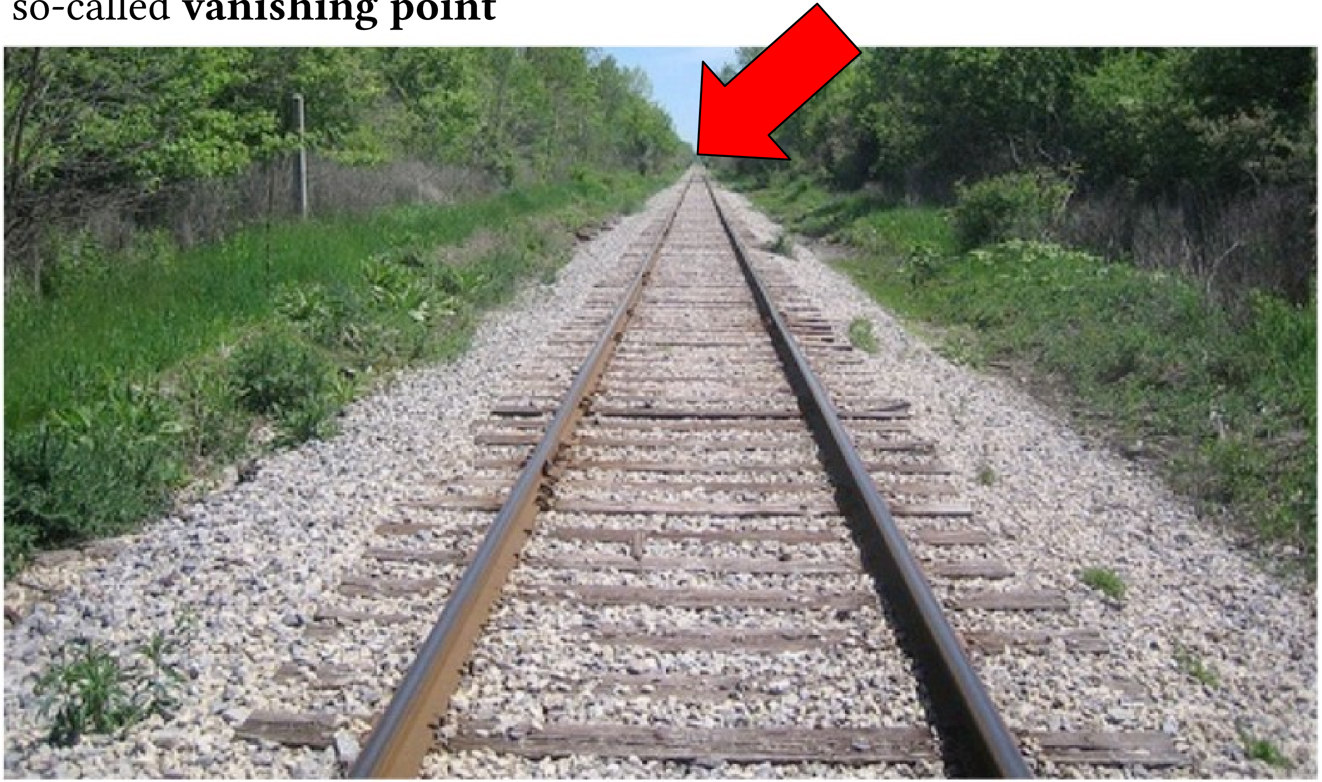
# Perspective transformation properties

- Points become…. Points!
- Lines become… Lines!
- Far away objects are smaller (divide by z)

# Perspective transformation properties

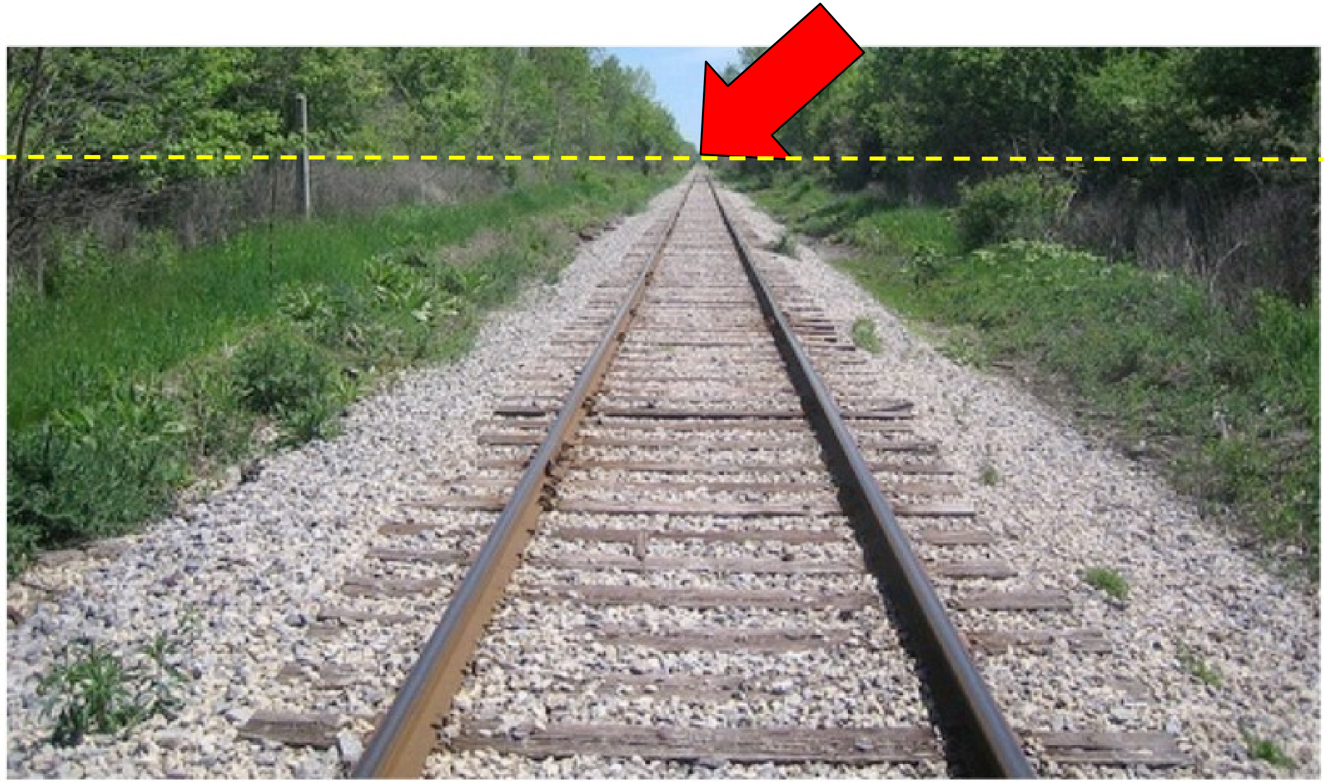


- Angles are not preserved
- Parallel lines intersect!
  - In the so-called **vanishing point**



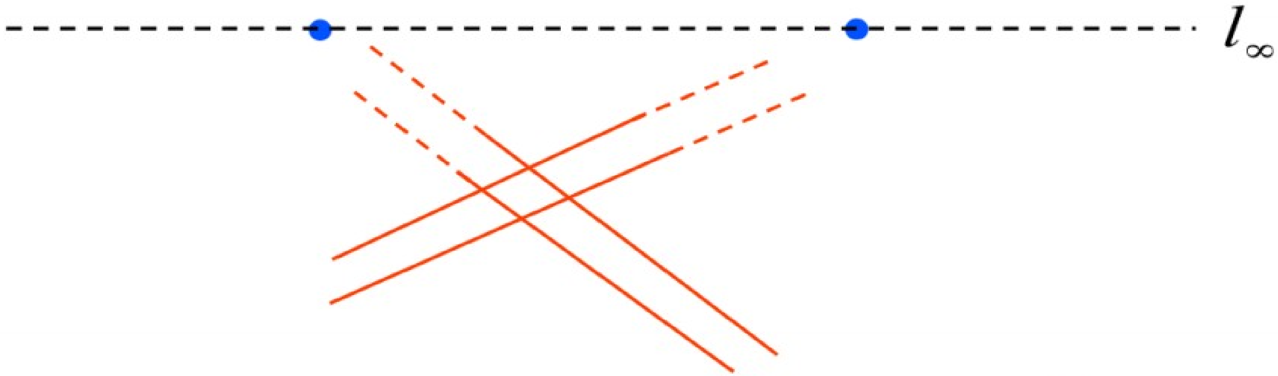
# Perspective transformation properties

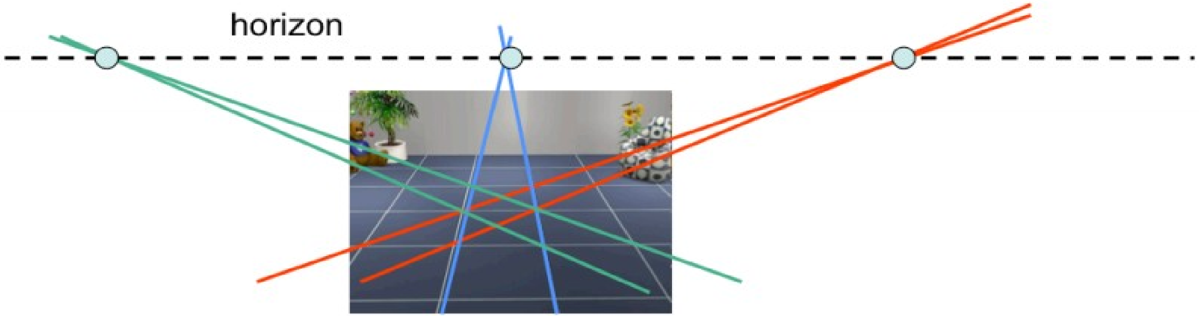
- Parallel lines that lies in the same plane have vanishing points on a line
- The Horizon (Vanishing Line)



# Perspective transformation properties

- Parallel lines that lies in the same plane have vanishing points on a line
- The Horizon (Vanishing Line)



# Pin Hole Geometry Recap

Intrinsic

$$M = K \cdot [I \quad 0] \cdot E = \boxed{K} \cdot [\boxed{R \quad T}] \in R^{3x4}$$

Extrinsic

$$P' = M \cdot P_w = K \cdot [R \quad T] \cdot P_w$$

2D Homogeneous

Model the perspective
transformation from 3D to 2D

3D Homogeneous