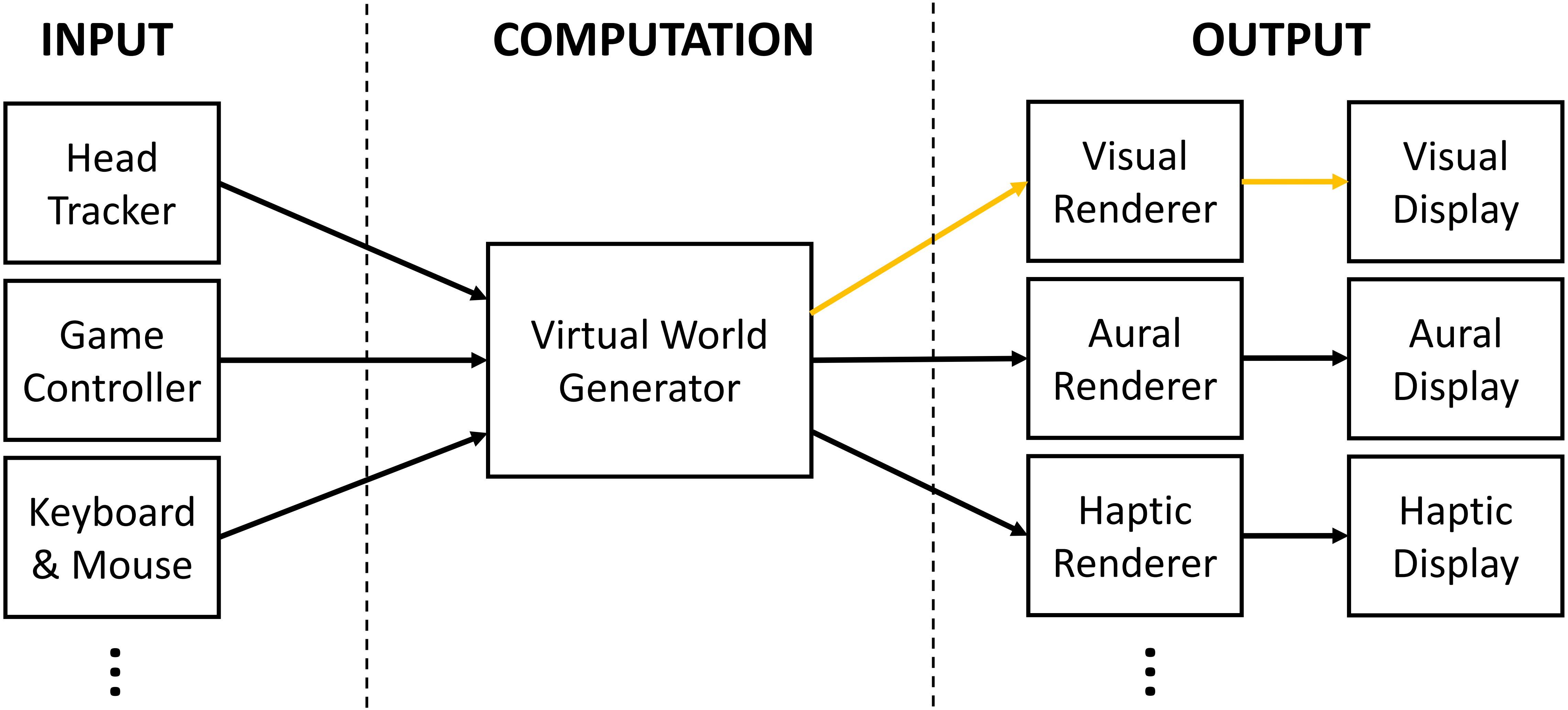
Camera Models

CS 6334 Virtual Reality

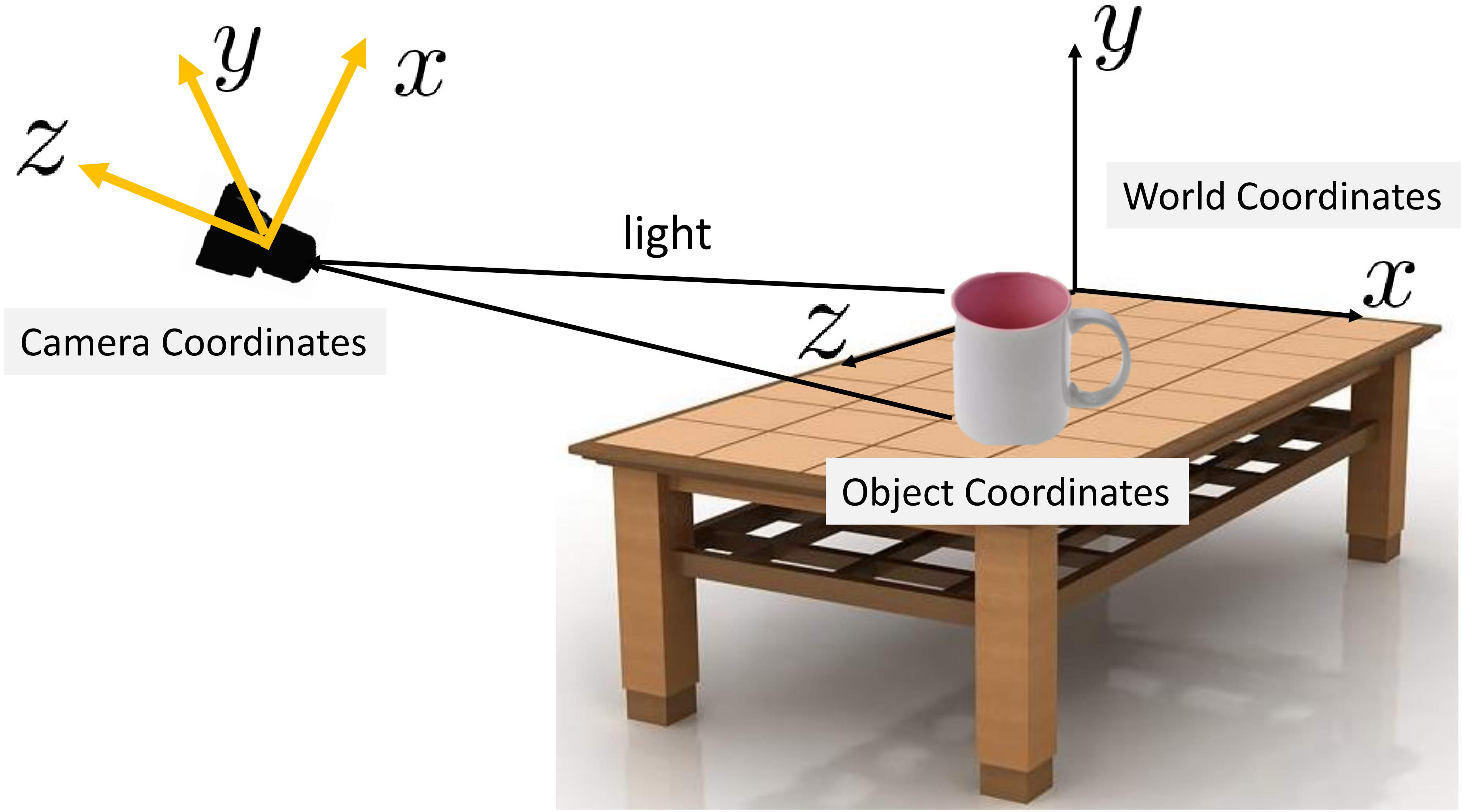
Professor Yu Xiang
The University of Texas at Dallas

Some slides of this lecture are courtesy Silvio Savarese

Review of VR Systems

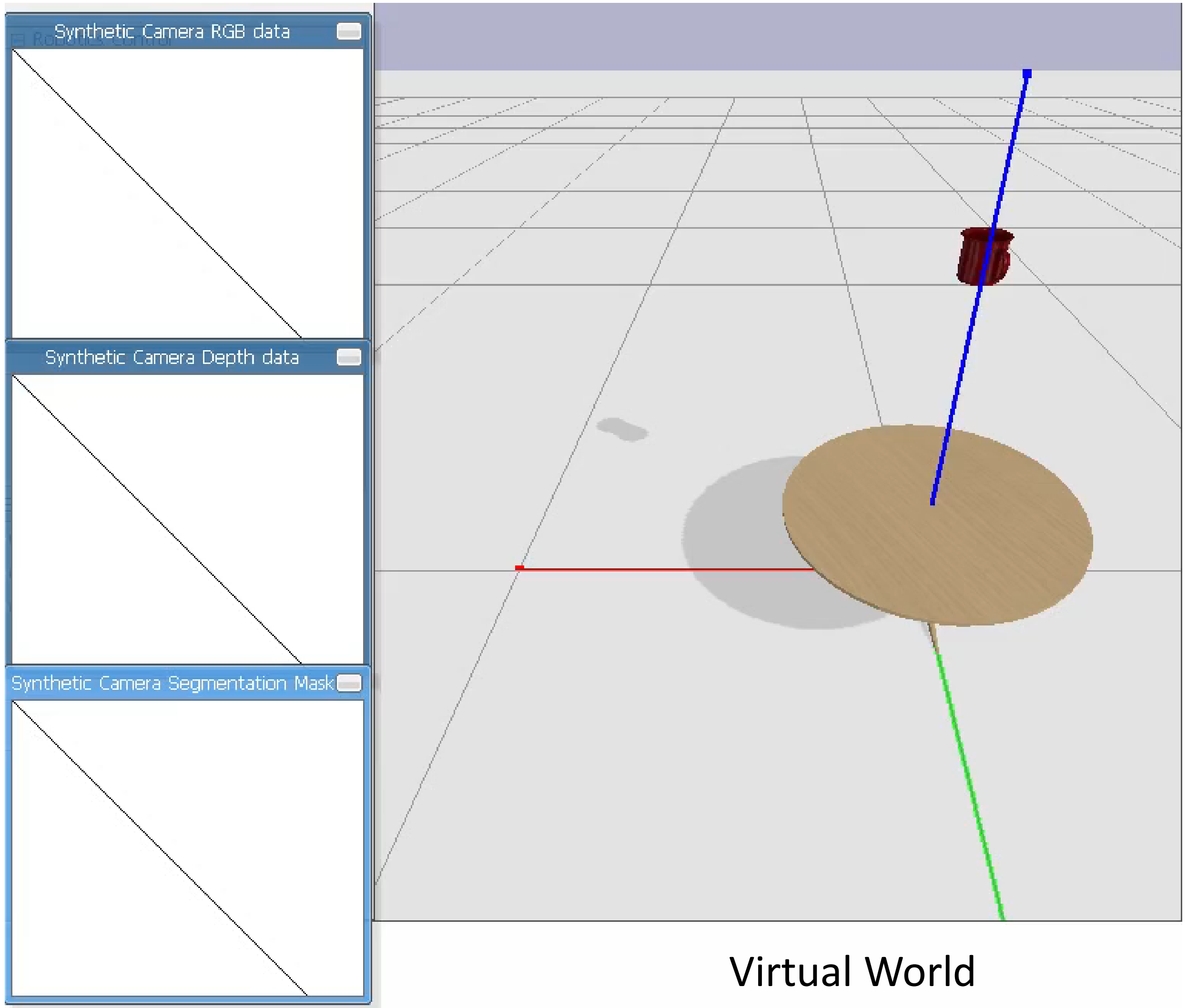


A Virtual World with a Camera



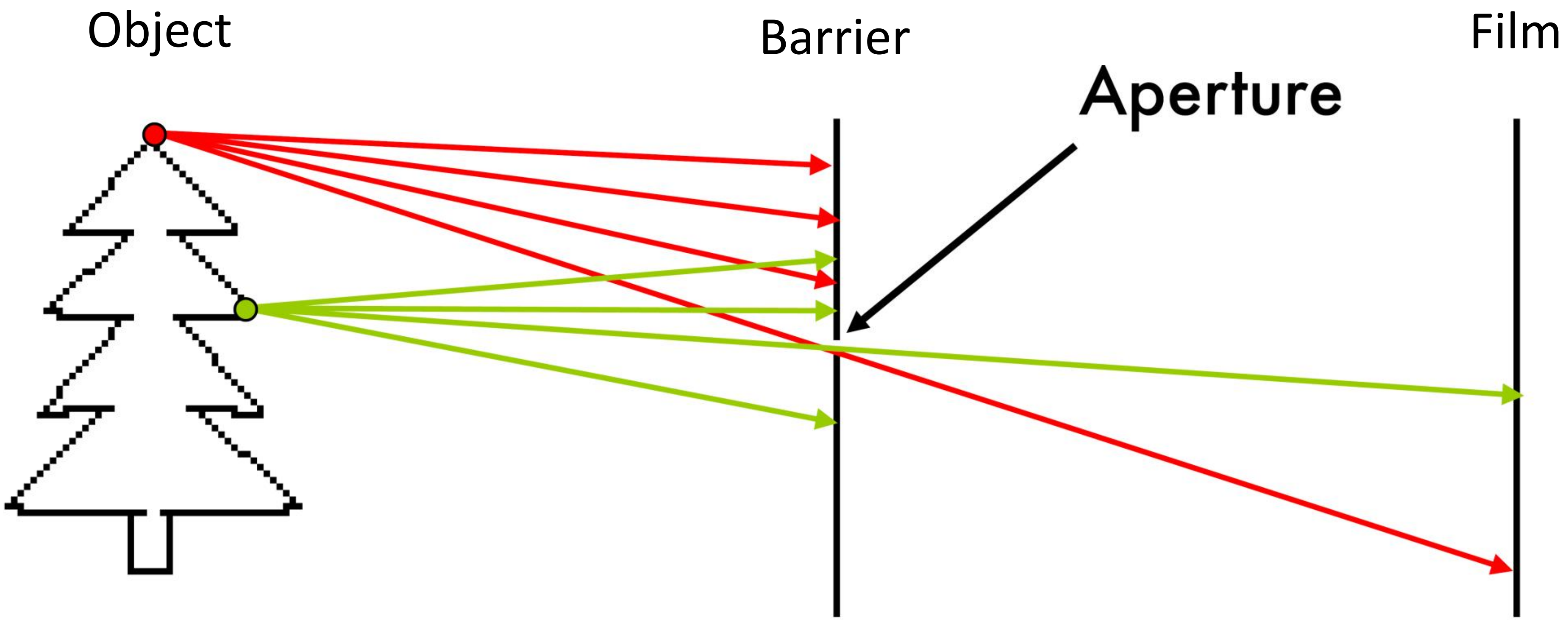
PyBullet with a Camera

Camera View

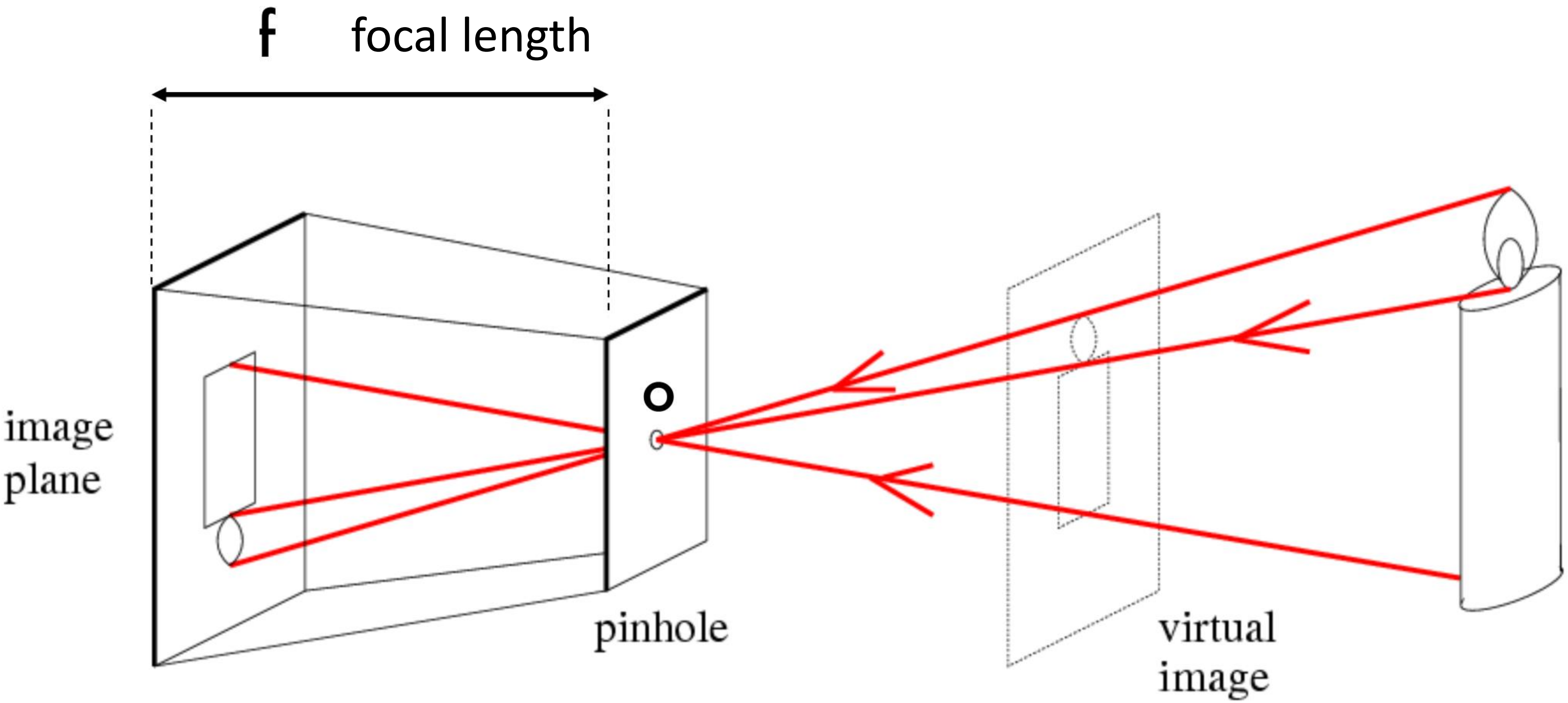


Virtual World

Pinhole Camera



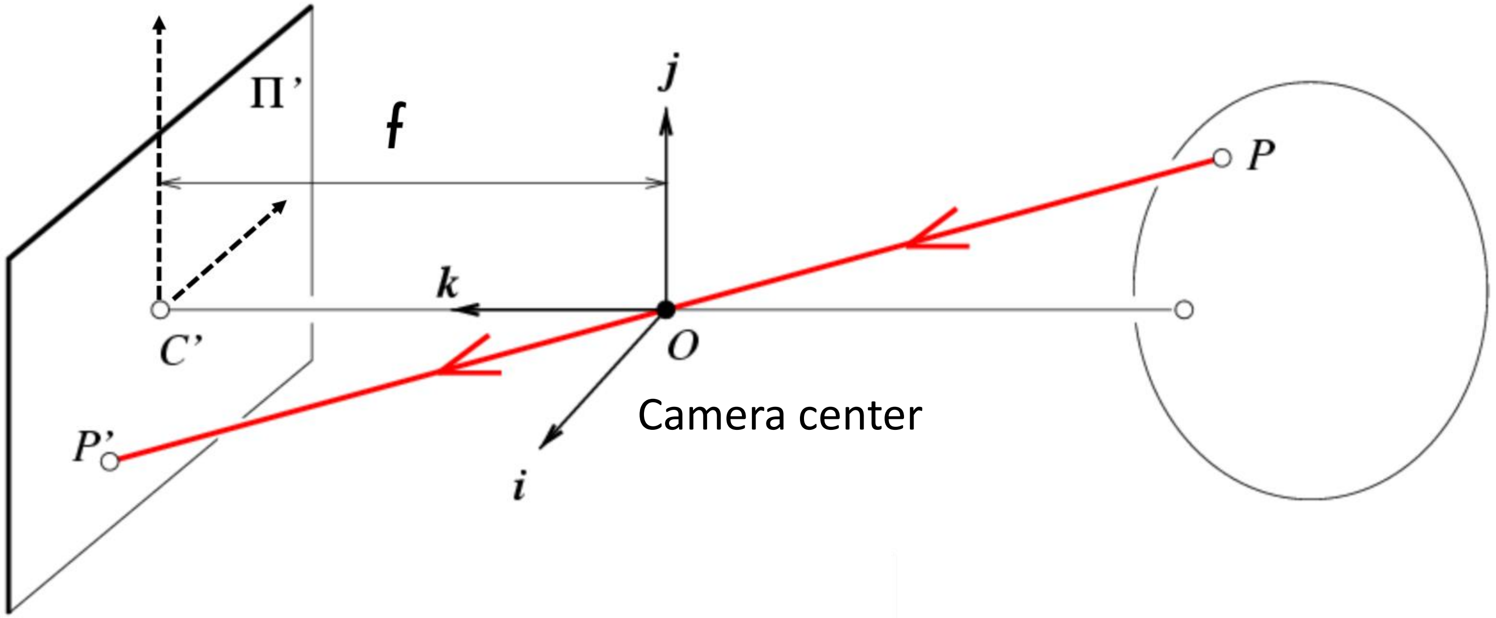
Pinhole Camera

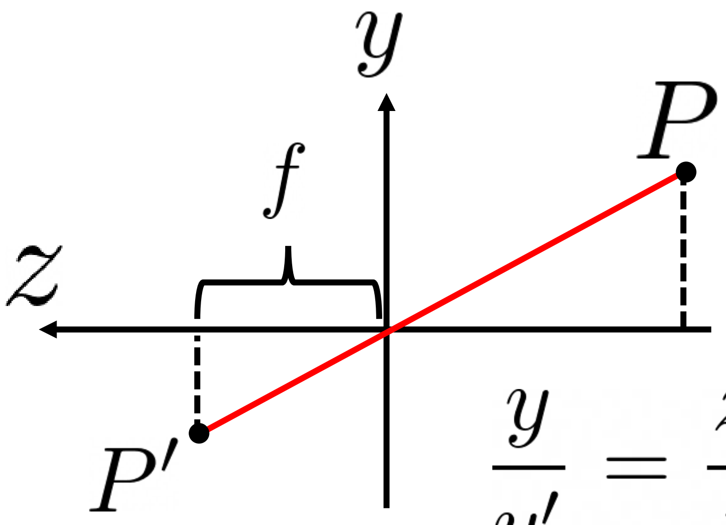


Rotate the image plane by 180°

Cannot be implemented in practice
Useful for theoretic analysis

Central Projection in Camera Coordinates





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

Camera coordinates

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

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

Homogeneous Coordinates

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

homogeneous image coordinates

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

homogeneous scene coordinates

Conversion

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

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

Central Projection with Homogeneous Coordinates

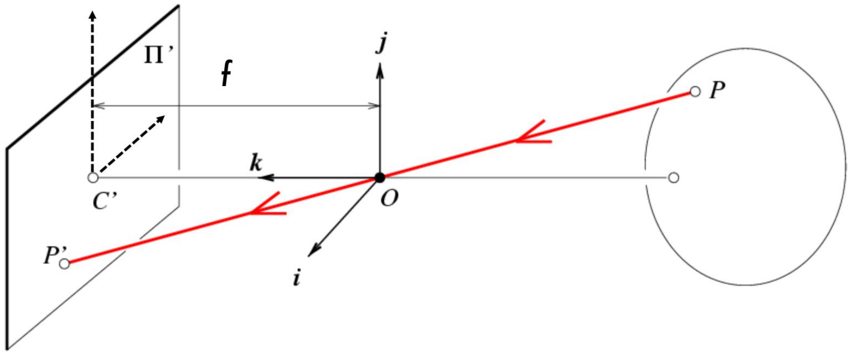
$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \longrightarrow \begin{bmatrix} f\,\frac{x}{z} \\ f\,\frac{y}{z} \end{bmatrix}$$

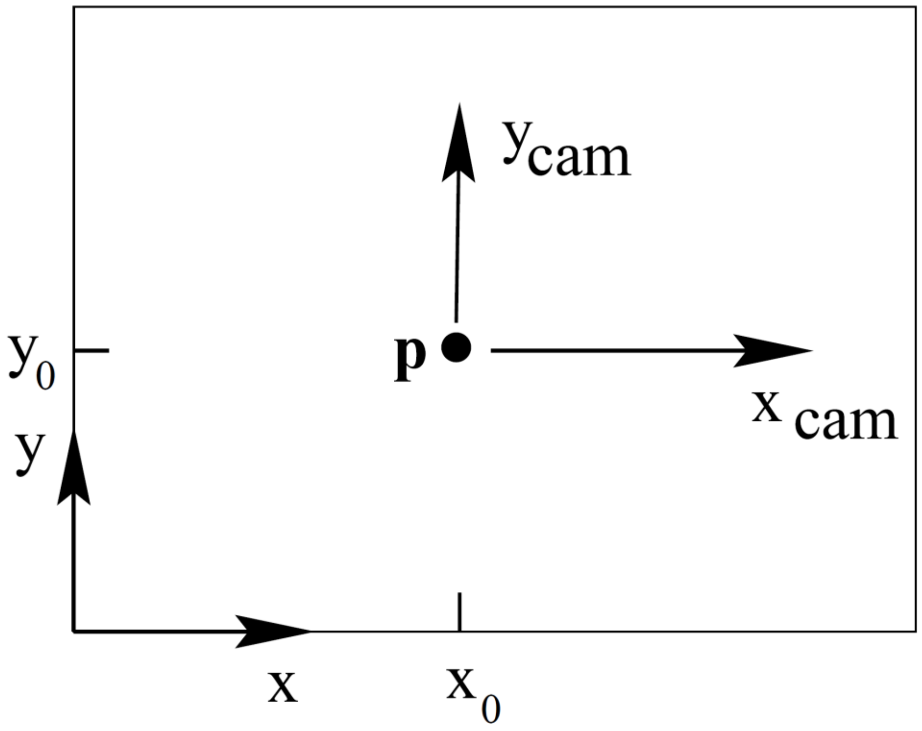
Central projection

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \longrightarrow \begin{bmatrix} fx \\ fy \\ z \end{bmatrix} = \begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

3x4 matrix

Principal Point Offset





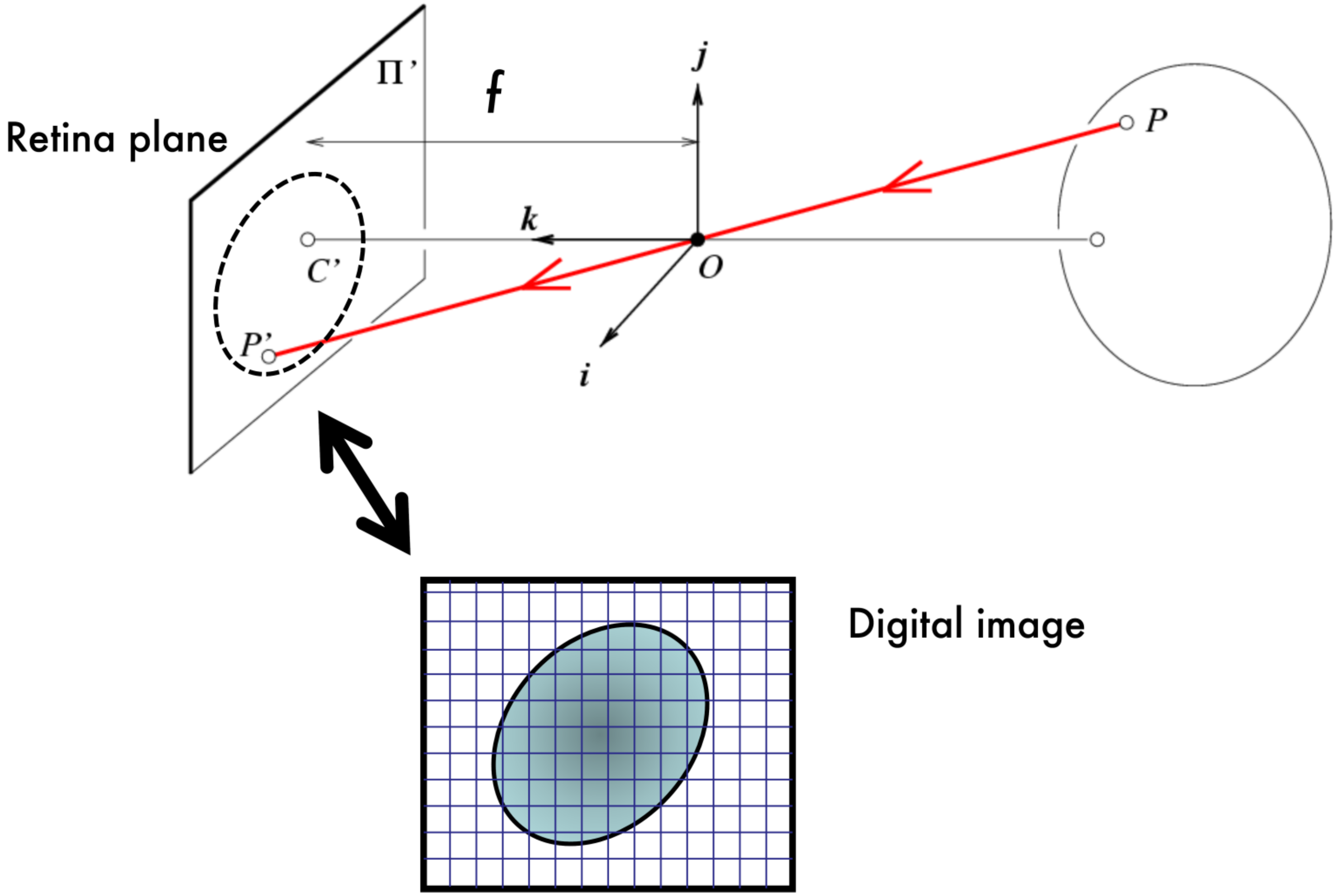
Principle point: projection of the camera center

Principal point $\mathbf{p} = (p_x, p_y)$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \longrightarrow \begin{bmatrix} f\frac{x}{z} + p_x \\ f\frac{y}{z} + p_y \end{bmatrix}$$

$$\begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

From Metric to Pixels



Pixels, bottom-left coordinate systems

From Metric to Pixels

- Metric space, i.e., meters

$$\begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix}$$

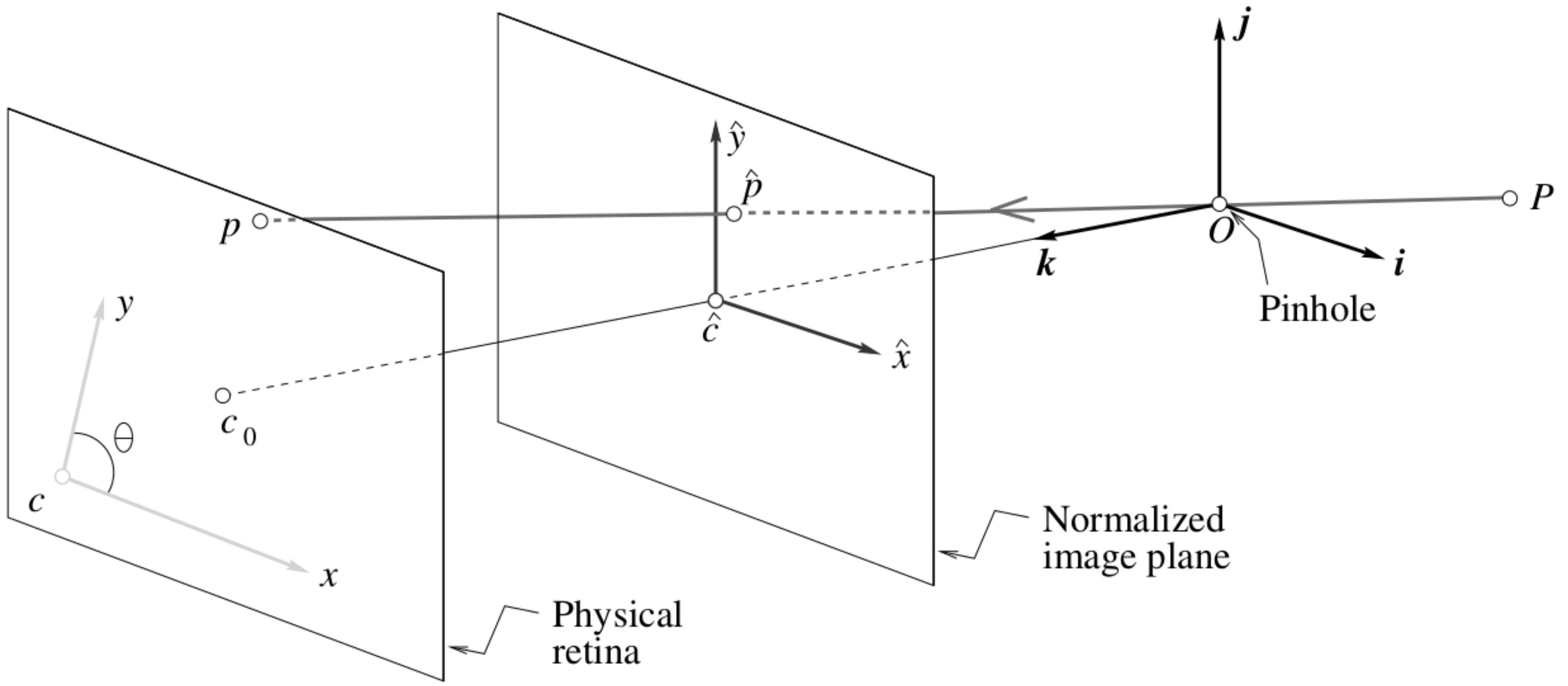
- Pixel space

$$\begin{bmatrix} \alpha_x & & x_0 & 0 \\ & \alpha_y & y_0 & 0 \\ & & 1 & 0 \end{bmatrix}$$

$$\alpha_x = f m_x$$
$$\alpha_y = f m_y$$
$$x_0 = p_x m_x$$
$$y_0 = p_y m_y$$

m_x, m_y Number of pixel per unit distance

Axis Skew



The skew parameter will be zero for most normal cameras.

$$\begin{bmatrix} \alpha_x & & x_0 & 0 \\ & \alpha_y & y_0 & 0 \\ & & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \longrightarrow \begin{bmatrix} \alpha_x \frac{x}{z} + x_0 \\ \alpha_y \frac{y}{z} + y_0 \end{bmatrix}$$

$$\begin{bmatrix} \alpha_x & -\alpha_x \cot(\theta) & x_0 & 0 \\ & \frac{\alpha_y}{\sin(\theta)} & y_0 & 0 \\ & & 1 & 0 \end{bmatrix}$$

https://blog.immenselyhappy.com/post/camera-axis-skew/

Camera Intrinsics

$$\begin{bmatrix} \alpha_x & -\alpha_x \cot(\theta) & x_0 & 0 \\ & \frac{\alpha_y}{\sin(\theta)} & y_0 & 0 \\ & & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Camera intrinsics

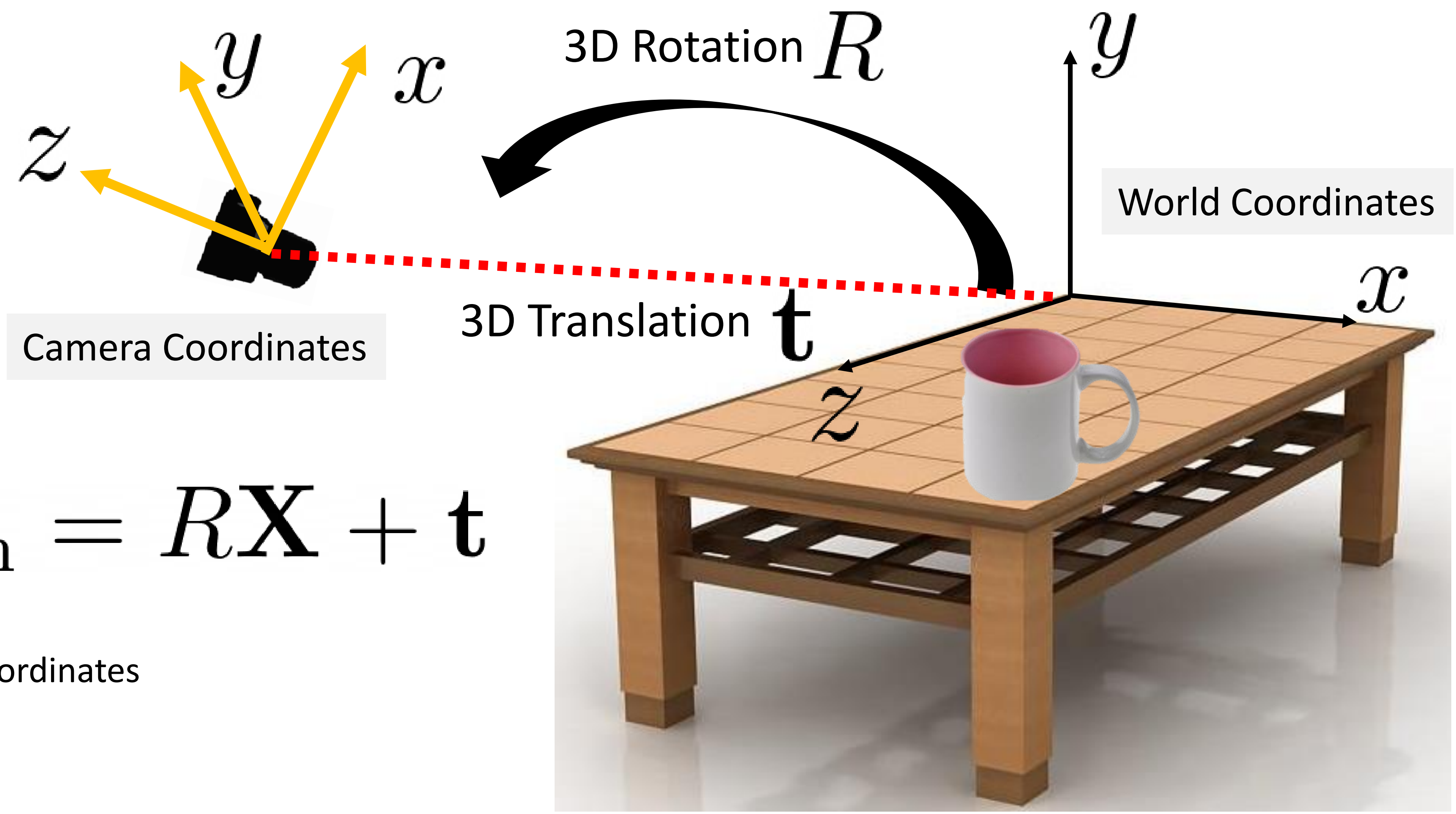
$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix}$$

$$\mathbf{x} = K[I|\mathbf{0}]\mathbf{X}_{\mathrm{cam}}$$

3x1 3x3 3x4 4x1

Homogeneous coordinates

Camera Extrinsics: Camera Rotation and Translation



$$\mathbf{X}_{\text{cam}} = R\mathbf{X} + \mathbf{t}$$

Euclidean coordinates

Camera Projection Matrix $P = K[R|\mathbf{t}]$

- Homogeneous coordinates

$$\mathbf{x} = K[I|\mathbf{0}]\mathbf{X}_{\text{cam}}$$
$$= K[R|\mathbf{t}]\mathbf{X}$$

$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix}$$

3x1 3x3 3x4 4x1

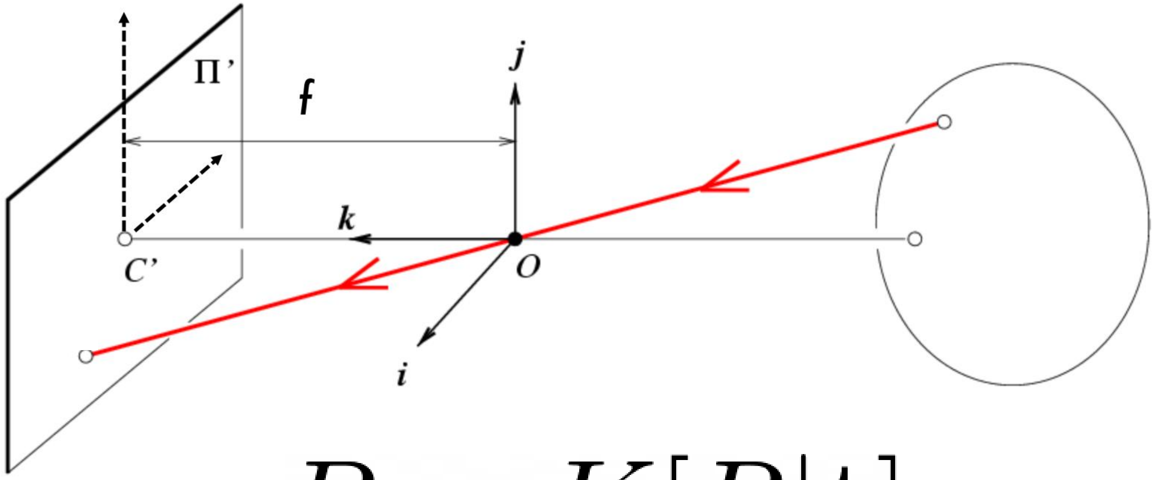
Image coordinates

World coordinates

Camera intrinsics

Camera extrinsics:
rotation and translation

Back-projection in World Coordinates



$$P = K[R|\mathbf{t}]$$

$$\mathbf{x} = P\mathbf{X}$$

- A pixel on the image backprojects to a ray in 3D

- The camera center O is on the ray
- $P^{+}\mathbf{x}$ is on the ray

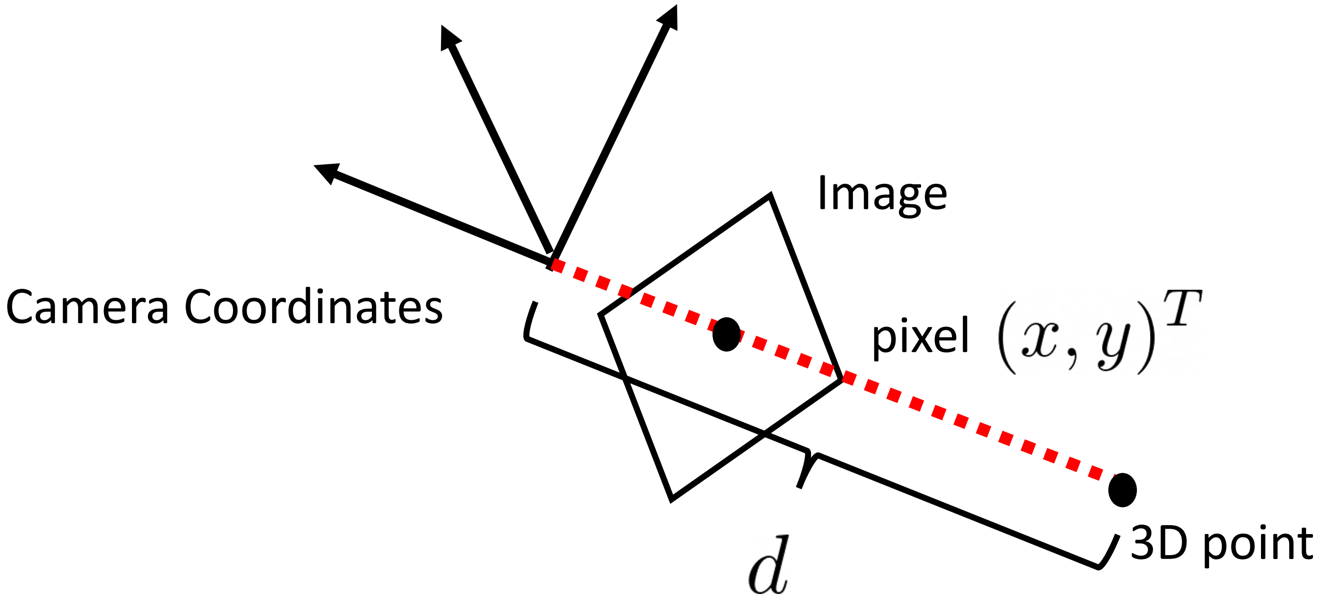
$$P^{+} = P^{T}(PP^{T})^{-1}$$

Pseudo-inverse

The ray can be written as

$$P^{+}\mathbf{x} + \lambda O$$

Back-projection in Camera Coordinates



$$P = K[I|\mathbf{0}]$$

$$\mathbf{x} = K[I|\mathbf{0}]\mathbf{X}_{\text{cam}}$$

$$K^{-1}\mathbf{x}$$

3D point with depth d : $dK^{-1}\mathbf{x}$

3D camera coordinates $\begin{bmatrix} d\frac{x-p_x}{f_x} \\ d\frac{y-p_y}{f_y} \\ d \end{bmatrix}$

Summary: Camera Models

- Camera projection matrix: intrinsics and extrinsics

$$P = K[R|\mathbf{t}]$$

3x3

3x4

Camera intrinsics

Camera extrinsics:
rotation and translation

Further Reading

- Section 6.1, Virtual Reality, Steven LaValle
- Multiview Geometry in Computer Vision, Richard Hartley and Andrew Zisserman, Chapter 9, Epipolar Geometry and the Fundamental Matrix
- Stanford CS231A: Computer Vision, From 3D Reconstruction to Recognition, Lecture 5 https://web.stanford.edu/class/cs231a/syllabus.html