Lecture 7

Cameras and Imaging Geometry

CS 3630

Topics

1. **Perspective Cameras**
2. **Pinhole Camera Model**
3. **Properties of projective Geometry**
4. **Stereo Vision**
5. **Stereo Geometry**
6. **Stereo Algorithms**

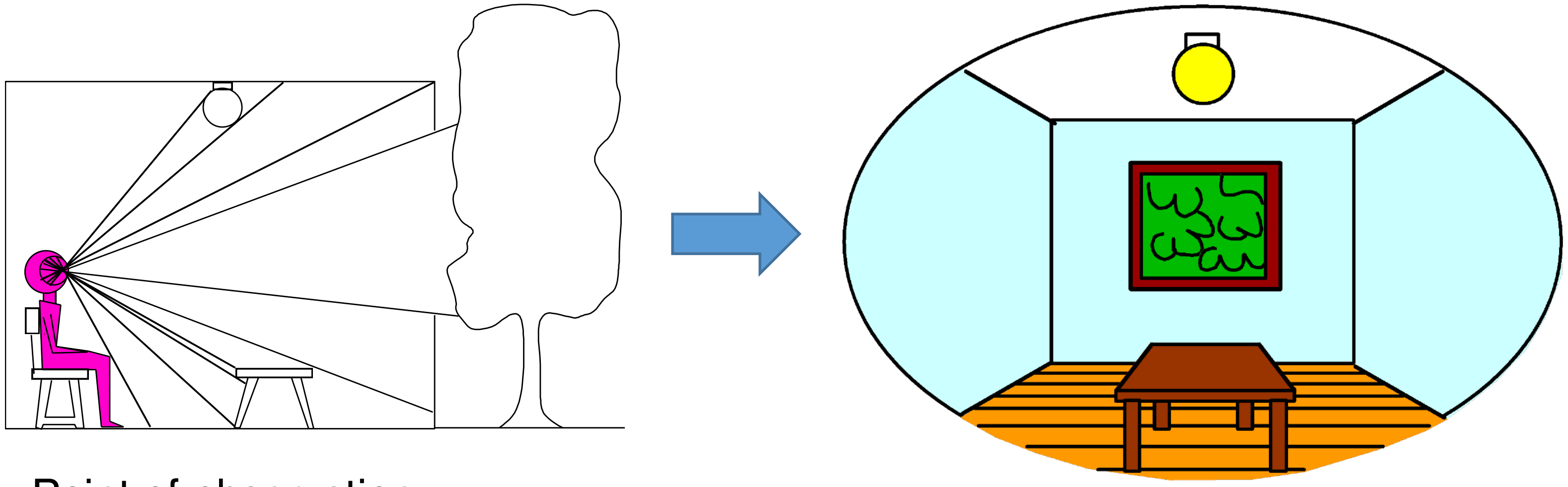
- Many slides borrowed from Frank Dellaert, James Hays, Irfan Essa, Sing Bing Kang and others.

Motivation

- We need to model the image formation process
- The camera can act as an (angular) measurement device
- Need a mathematical model for a simple camera
- Two cameras are better than one: metric measurements

1. Perspective Cameras

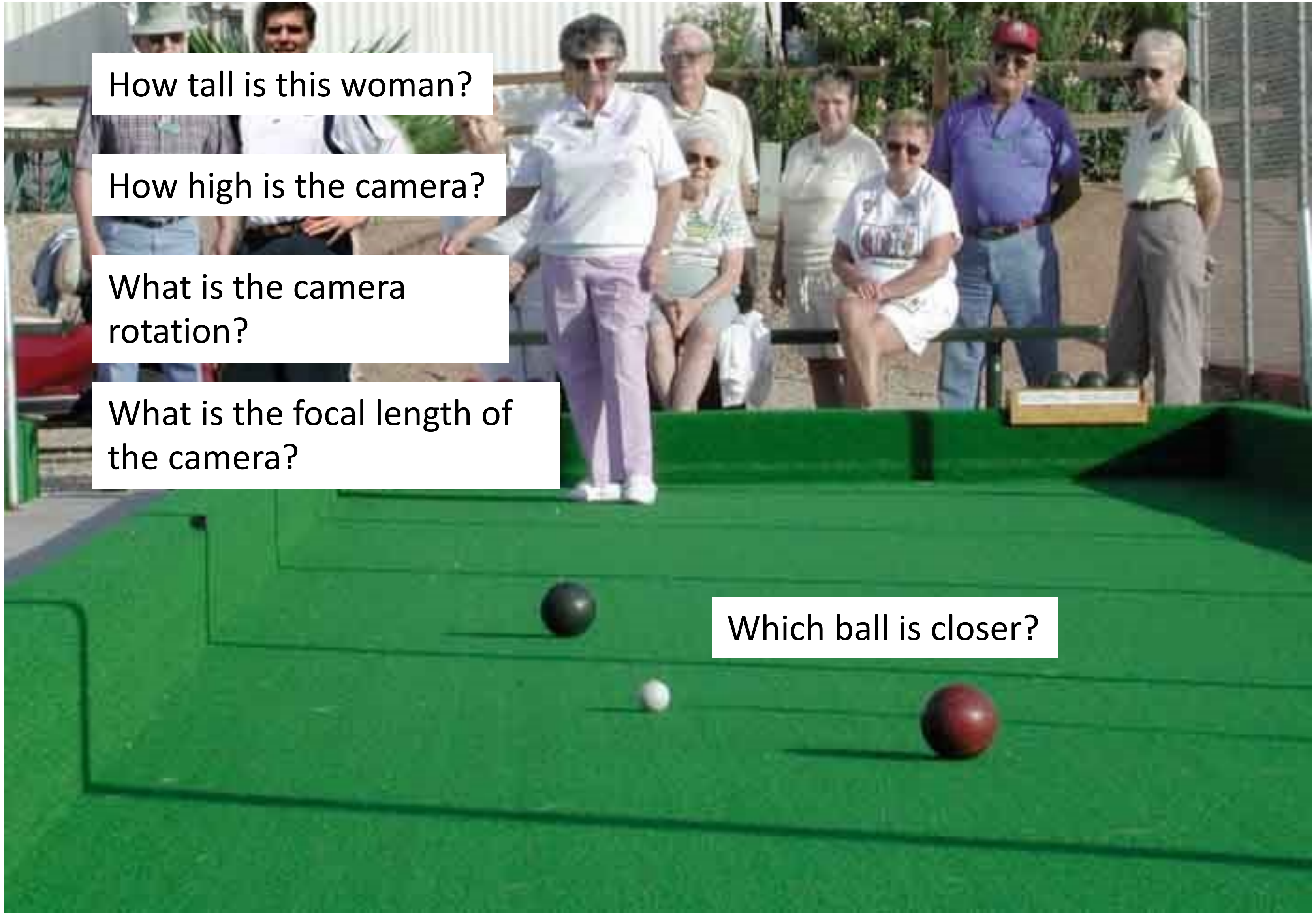
3D world *2D image*



Point of observation

- Recall: Computer Vision: Images to Models
- To do this, we first need to understand the image formation process.
- We concentrate here on *geometry* (not photometry)

Camera and World Geometry



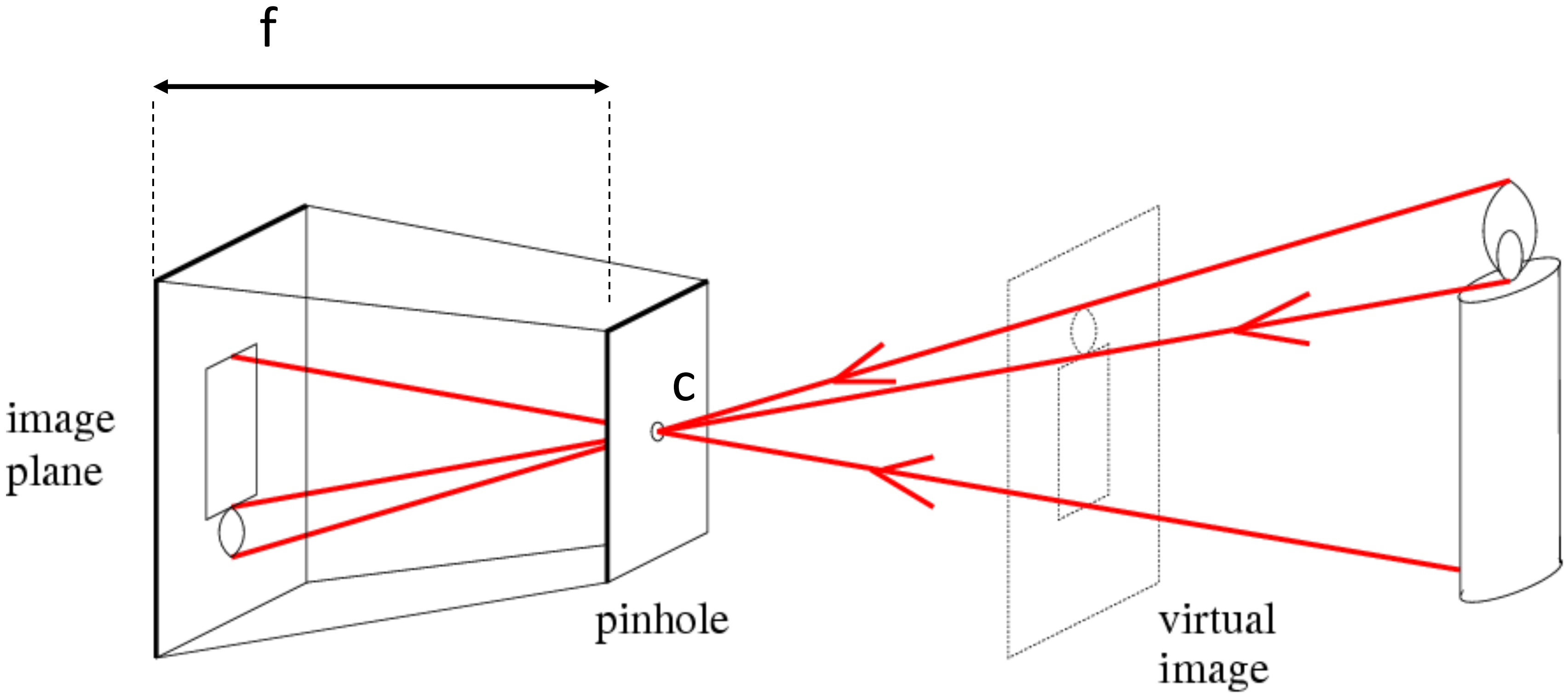
Slide source: Seitz

Projection can be tricky…

Slide source: Seitz

Projection can be tricky…

2. Pinhole camera model



f = focal length

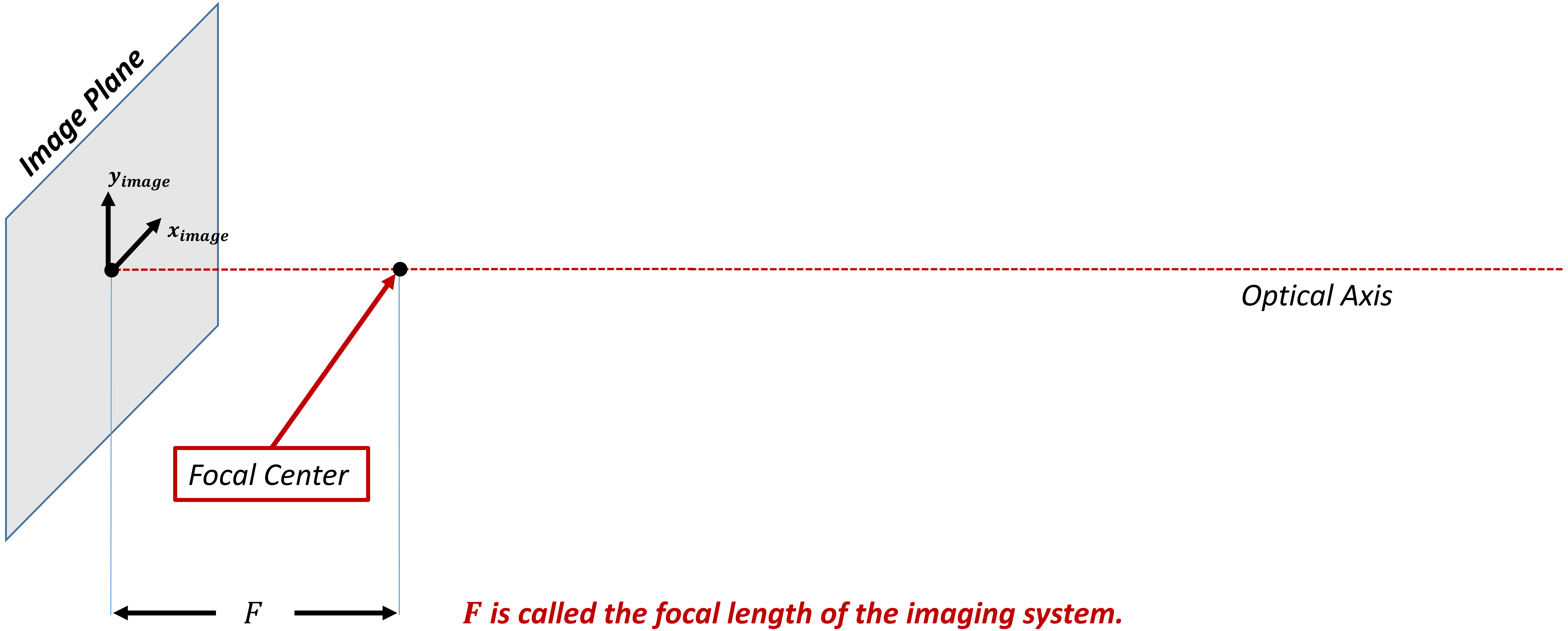
c = center of the camera

Figure from Forsyth

Pinhole Camera Geometry

The imaging geometry for the pinhole camera has several important properties:

- The image plane is located at distance F behind the focal center.
- The optical axis passes through the focal center, perpendicular to the image plane.

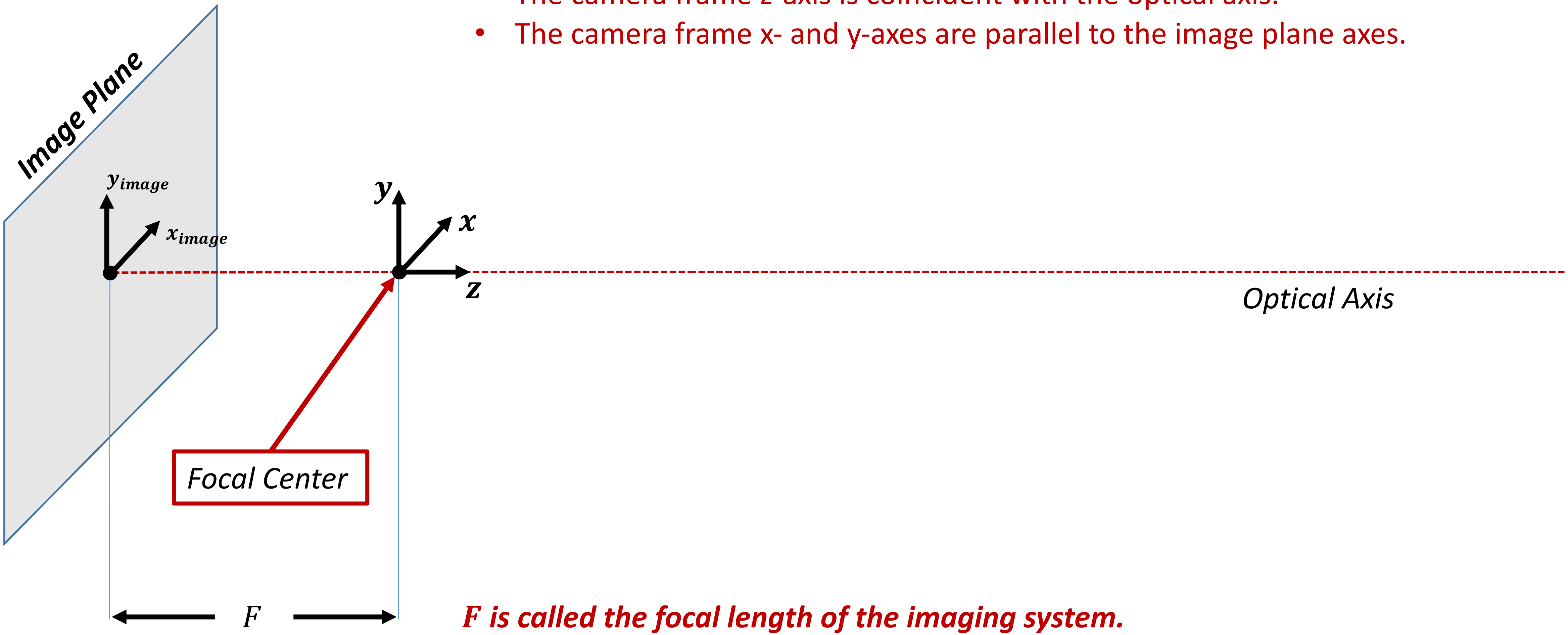


***F* is called the focal length of the imaging system.**

Pinhole Camera Geometry

The imaging geometry for the pinhole camera has several important properties:

- The image plane is located at distance F behind the focal center.
- The optical axis passes through the focal center, perpendicular to the image plane.
- The camera coordinate frame has its origin at the focal center.
- The camera frame z-axis is coincident with the optical axis.
- The camera frame x- and y-axes are parallel to the image plane axes.

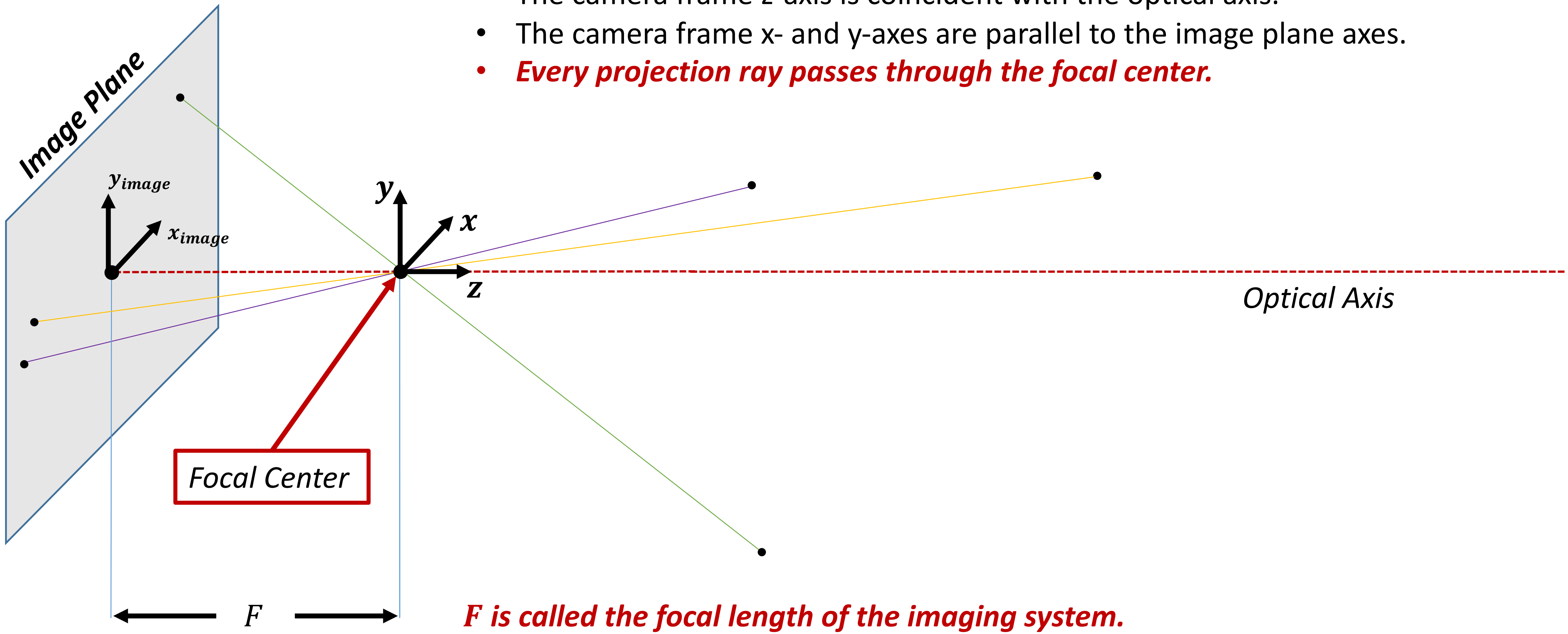


F is called the focal length of the imaging system.

Pinhole Camera Geometry

The imaging geometry for the pinhole camera has several important properties:

- The image plane is located at distance F behind the focal center.
- The optical axis passes through the focal center, perpendicular to the image plane.
- The camera coordinate frame has its origin at the focal center.
- The camera frame z-axis is coincident with the optical axis.
- The camera frame x- and y-axes are parallel to the image plane axes.
- ***Every projection ray passes through the focal center.***

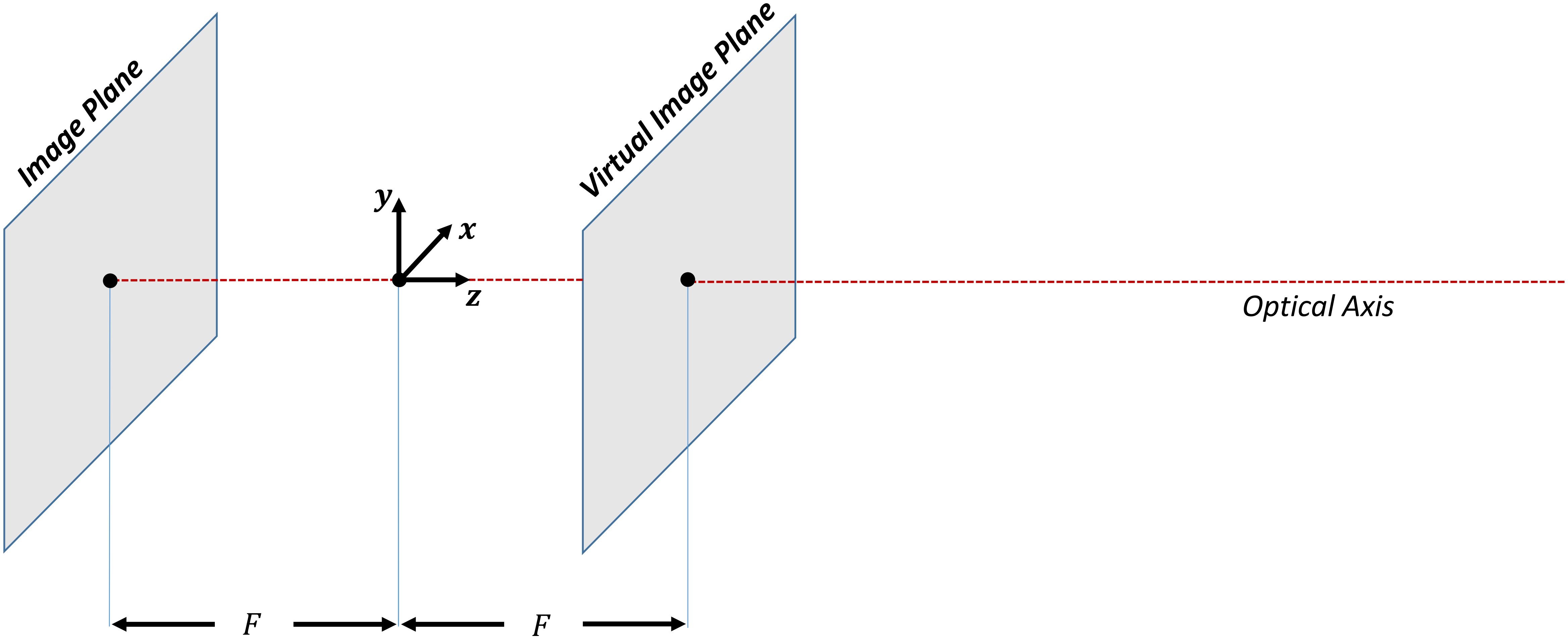


F is called the focal length of the imaging system.

Pinhole Camera

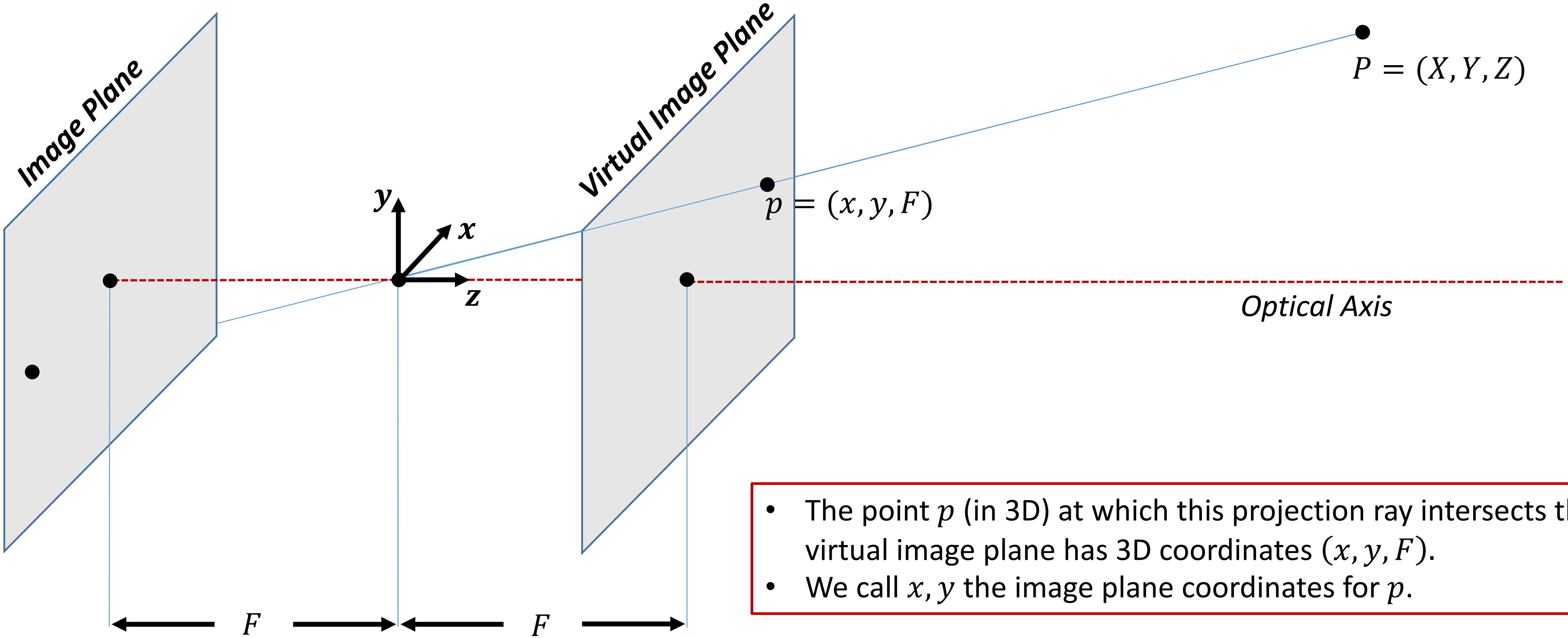
Life is so much easier if we insert a ***virtual image plane*** in front of the focal center.

No more need for upside-down image geometry!



Pinhole Camera

The point $P = (X, Y, Z)$ lies on a projection ray that passes through P and the focal center, and that intersects both the image plane and the virtual image plane.



- The point p (in 3D) at which this projection ray intersects the virtual image plane has 3D coordinates (x, y, F).
- We call x, y the image plane coordinates for p.

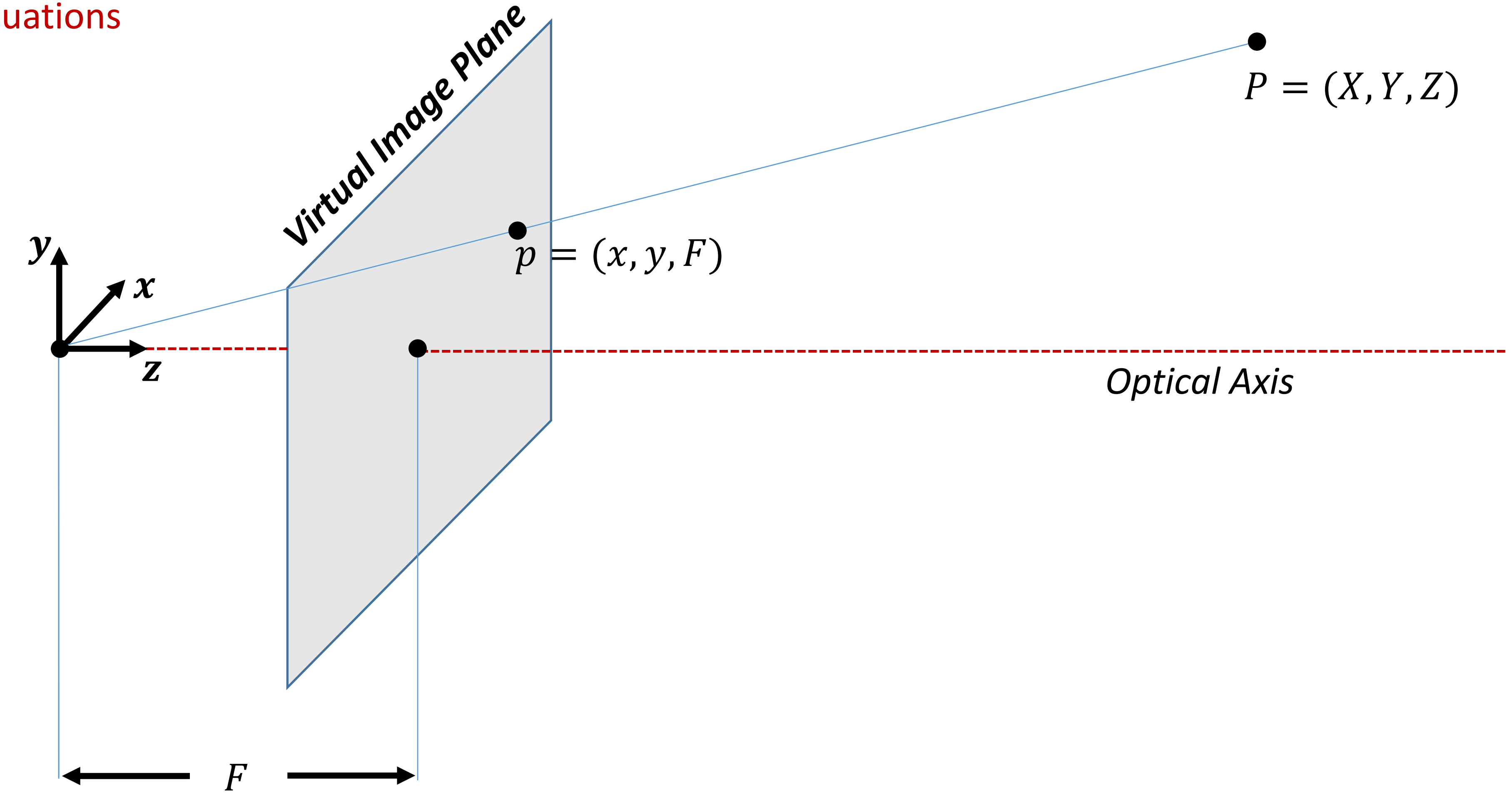
Pinhole Camera

Because p and P lie on the same projection ray through the origin, we have

$$\lambda p = P$$

We can write this as three equations

$$\lambda x = X$$
$$\lambda y = Y$$
$$\lambda F = Z$$



Pinhole Camera

Because p and P lie on the same projection ray through the origin, we have

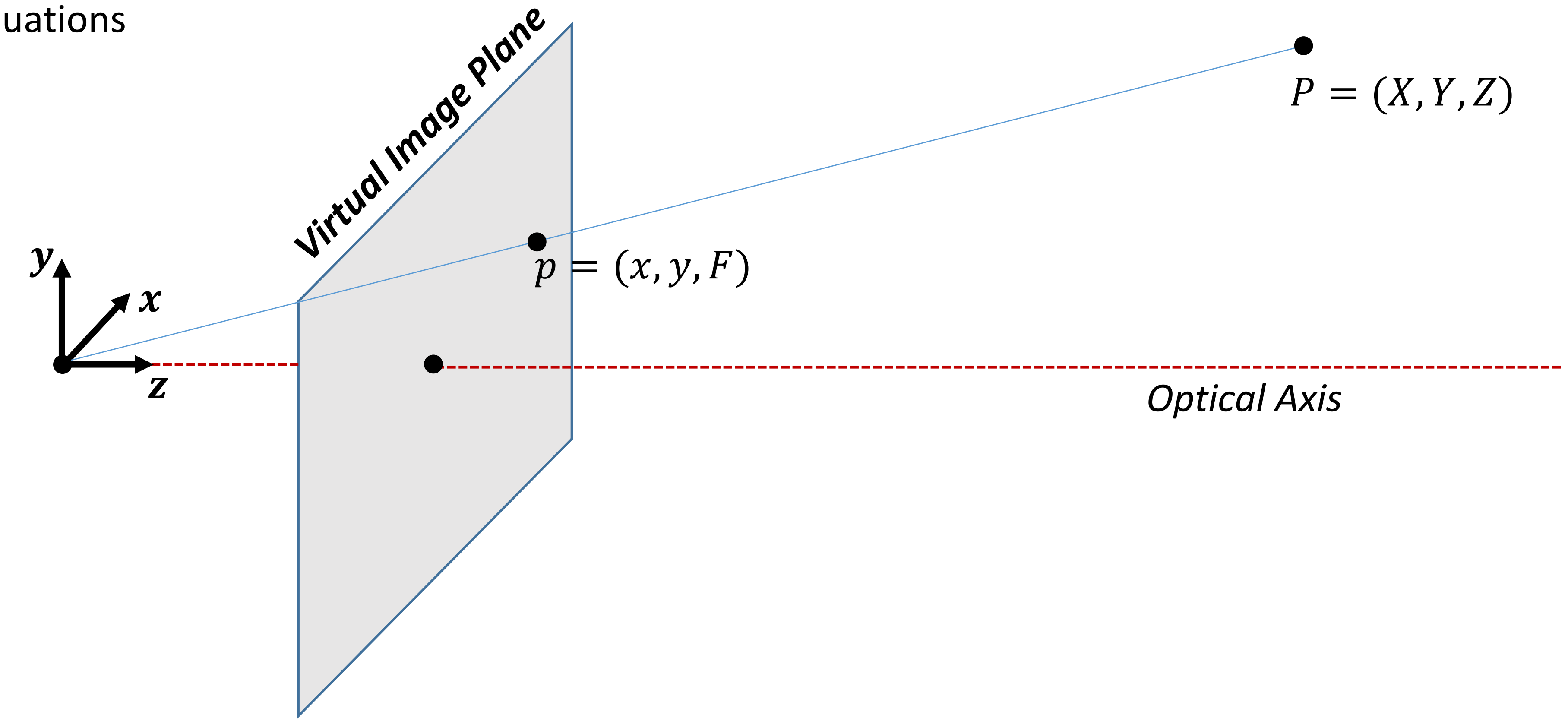
$$\lambda p = P$$

We can write this as three equations

$$\lambda x = X$$
$$\lambda y = Y$$
$$\lambda F = Z$$

Solving for λ yields

$$\lambda = \frac{Z}{F}$$



Pinhole Camera

Because p and P lie on the same projection ray through the origin, we have

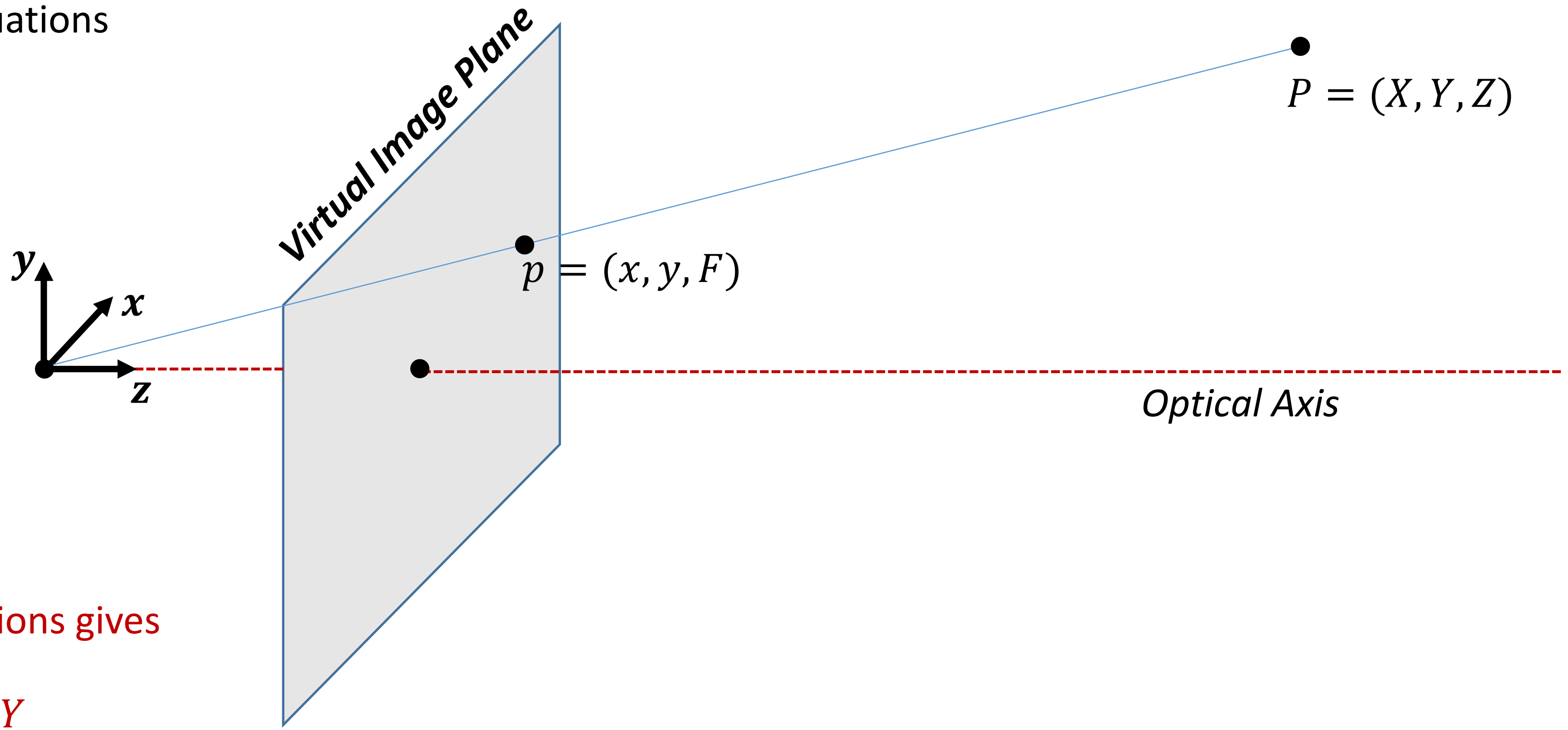
$$\lambda p = P$$

We can write this as three equations

$$\lambda x = X$$
$$\lambda y = Y$$
$$\lambda F = Z$$

Solving for λ yields

$$\lambda = \frac{Z}{F}$$



Substituting into the first equations gives

$$x = F\frac{X}{Z}, \qquad y = F\frac{Y}{Z}$$

Pinhole Camera

Because p and P lie on the same projection ray through the origin, we have

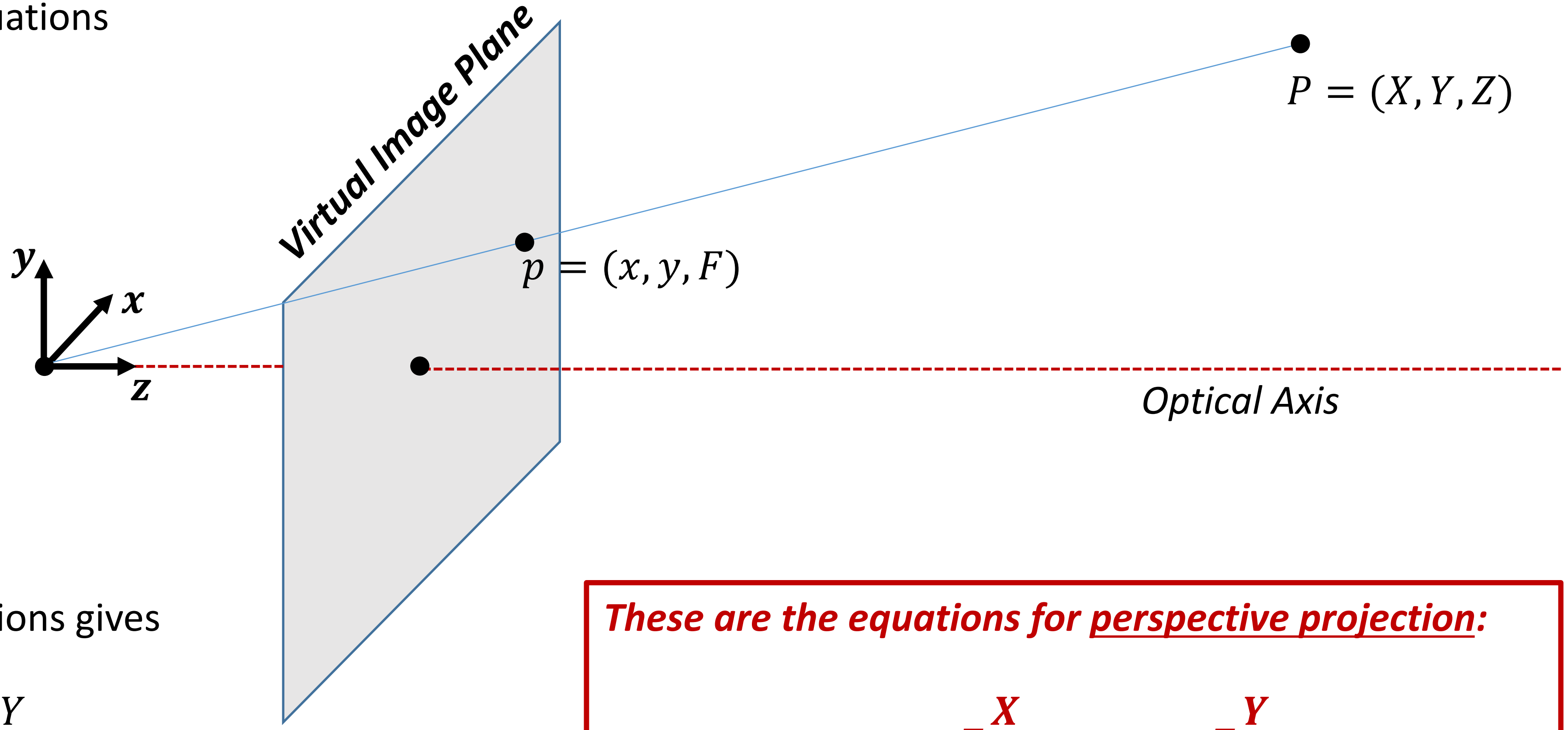
$$\lambda p = P$$

We can write this as three equations

$$\lambda x = X$$
$$\lambda y = Y$$
$$\lambda F = Z$$

Solving for λ yields

$$\lambda = \frac{Z}{F}$$



Substituting into the first equations gives

$$x = F\frac{X}{Z}, \qquad y = F\frac{Y}{Z}$$

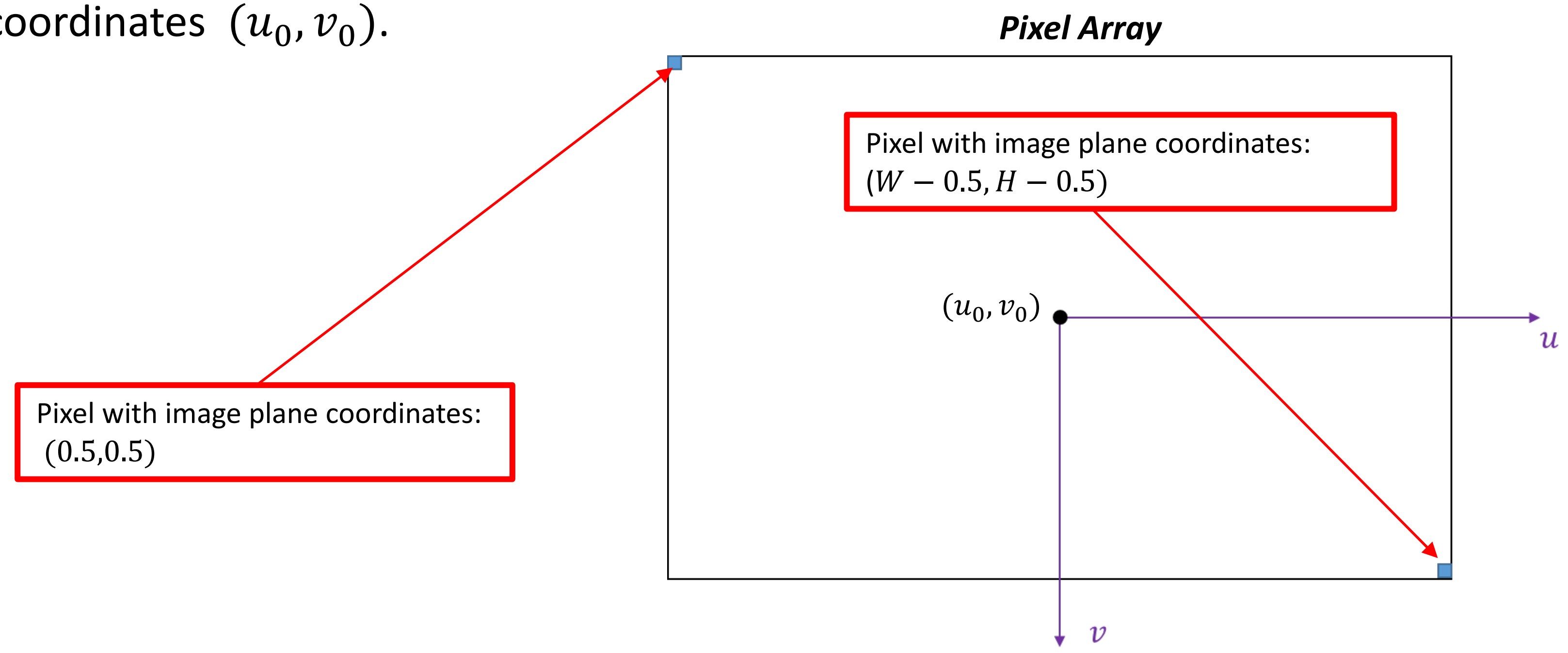
These are the equations for <u>perspective projection</u>:

$$\boldsymbol{x = F\frac{X}{Z}, \qquad y = F\frac{Y}{Z}}$$

Sensor Coordinates

- Instead of a continuous image plane, real cameras have a 2D array of sensors that correspond to pixels in the image.
- When we make measurements in an image, we measure ***sensor coordinates***, not image plane coordinates.

Sensor coordinate frame:
- The top, left pixel is location 0,0 in the sensor array.
- The bottom, right pixel has location $W-1, H-1$ in the sensor array.
- The sensor coordinates or a pixel, u, v correspond to the center of the corresponding pixel.
  - Top, left pixel is $(0.5, 0.5)$
  - Bottom right pixel is $(W-0.5, H-0.5)$
- Note that the v-axis points ***down***.
- The origin of the sensor frame has coordinates (u_0, v_0).



Sensor Coordinates

<u>From image-plane coordinates to sensor coordinates</u>

To convert from image-plane coordinates to sensor coordinates u, v

- Scale x by pixel width
- Scale y by pixel height
- Shift coordinates by u_0, v_0:

$$u = u_0 + \alpha x, \qquad v = v_0 - \beta y$$

If we now substitute the perspective projection equations for x and y we obtain

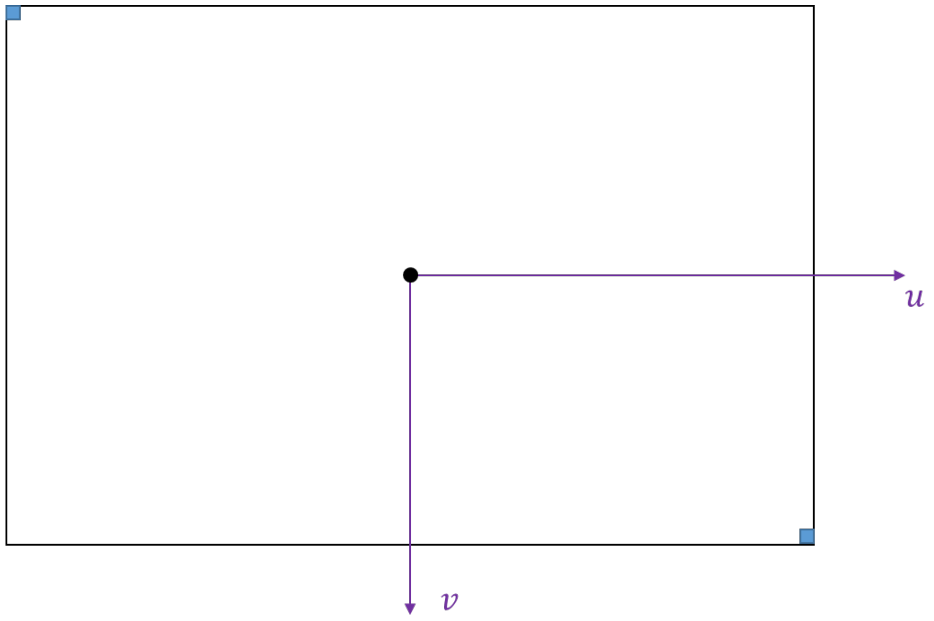
$$u = u_0 + \alpha F \frac{X}{Z}, \qquad v = v_0 - \beta F \frac{Y}{Z}$$

If the camera happens to have square pixels, then $\alpha = \beta$ and we can simplify this to

$$u = u_0 + f \frac{X}{Z}, \qquad v = v_0 - f \frac{Y}{Z}$$

Camera calibration is used to determine the values of u_0, v_0 and f.

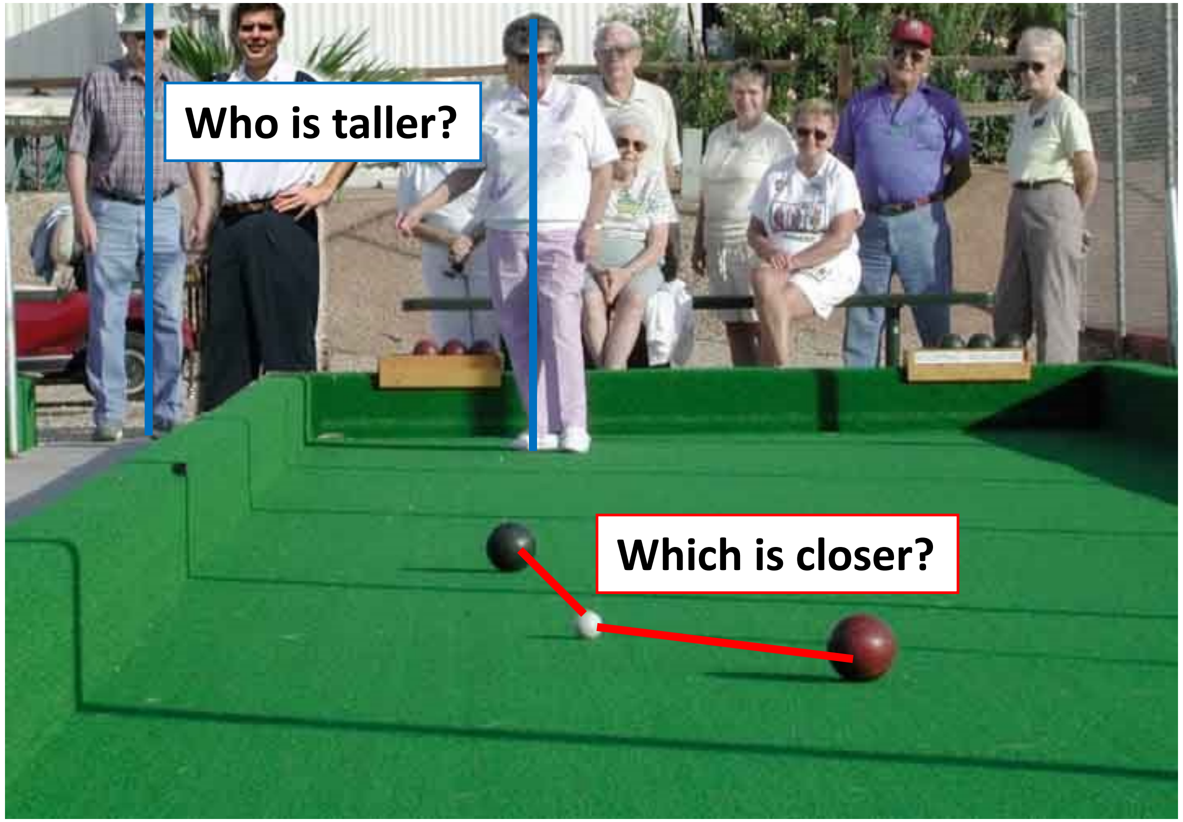
Pixel Array



3. Properties of projective Geometry

What is lost?

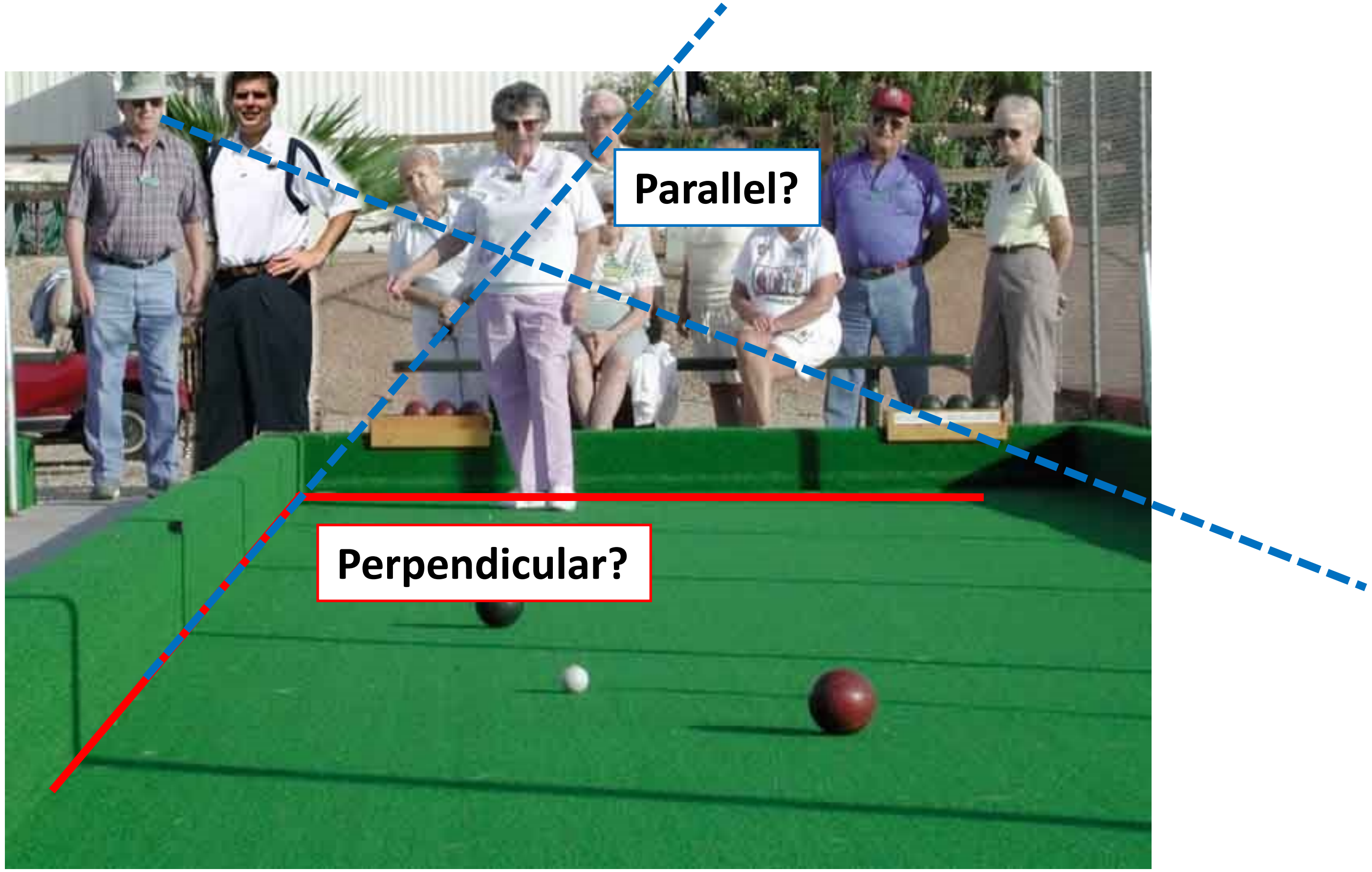
- Length



Properties of projective Geometry

What is lost?

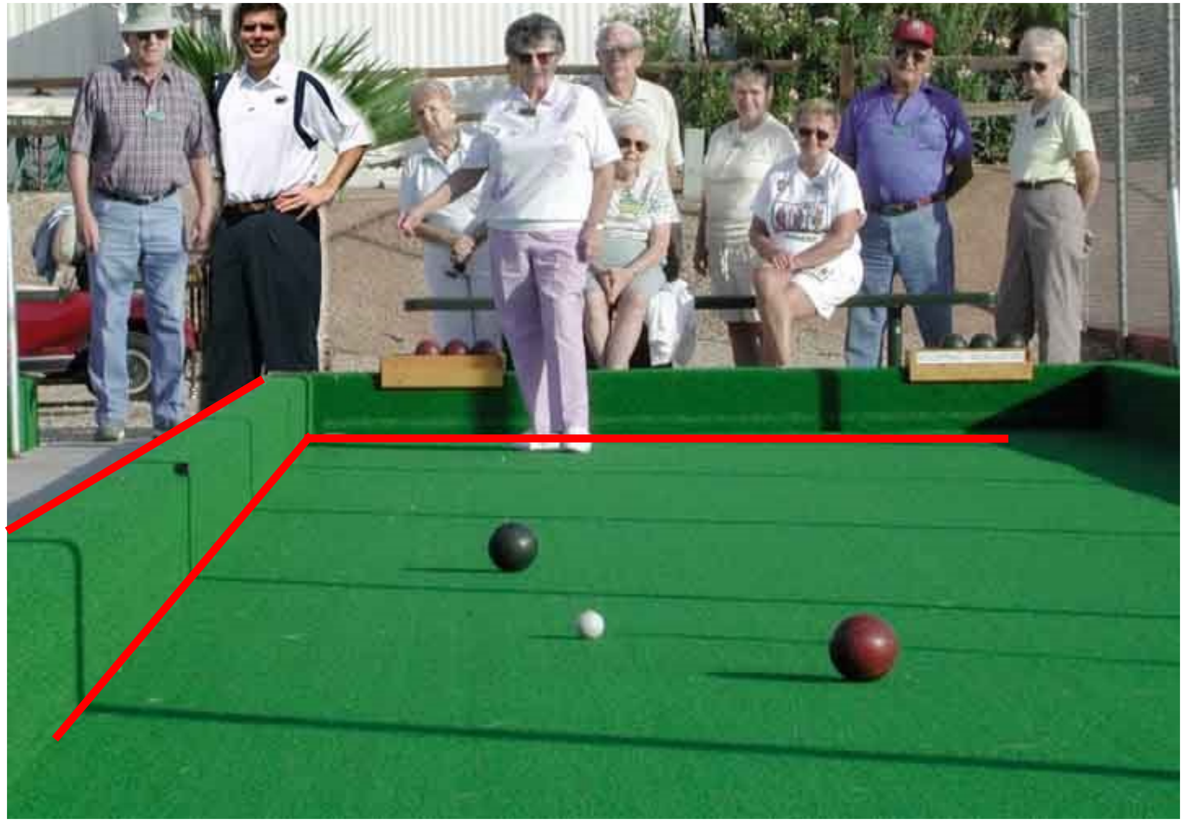
- Length
- Angles



Properties of projective Geometry

What is preserved?

- Straight lines are still straight



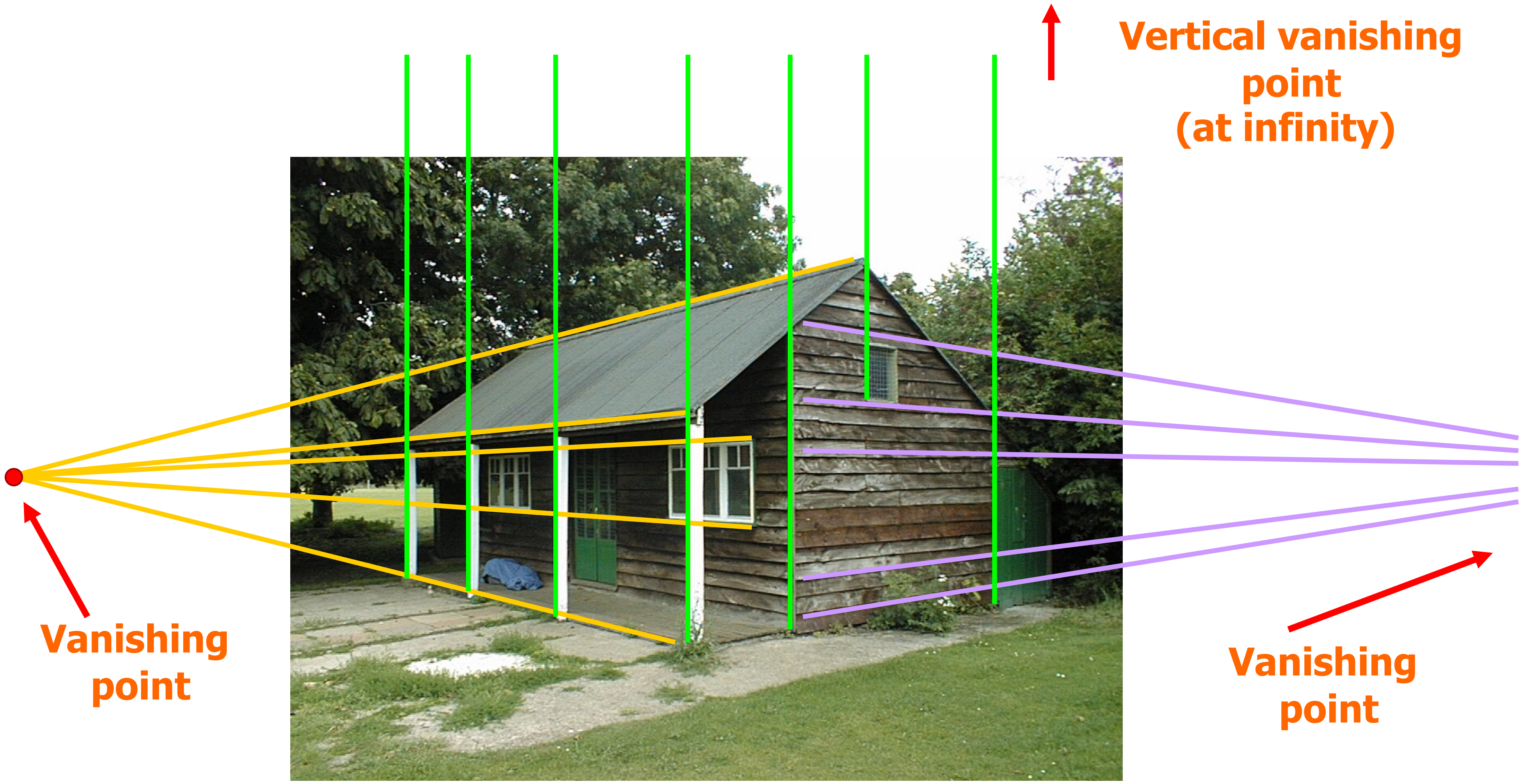
We can see infinity !

Where do parallel lines meet?

At infinity.

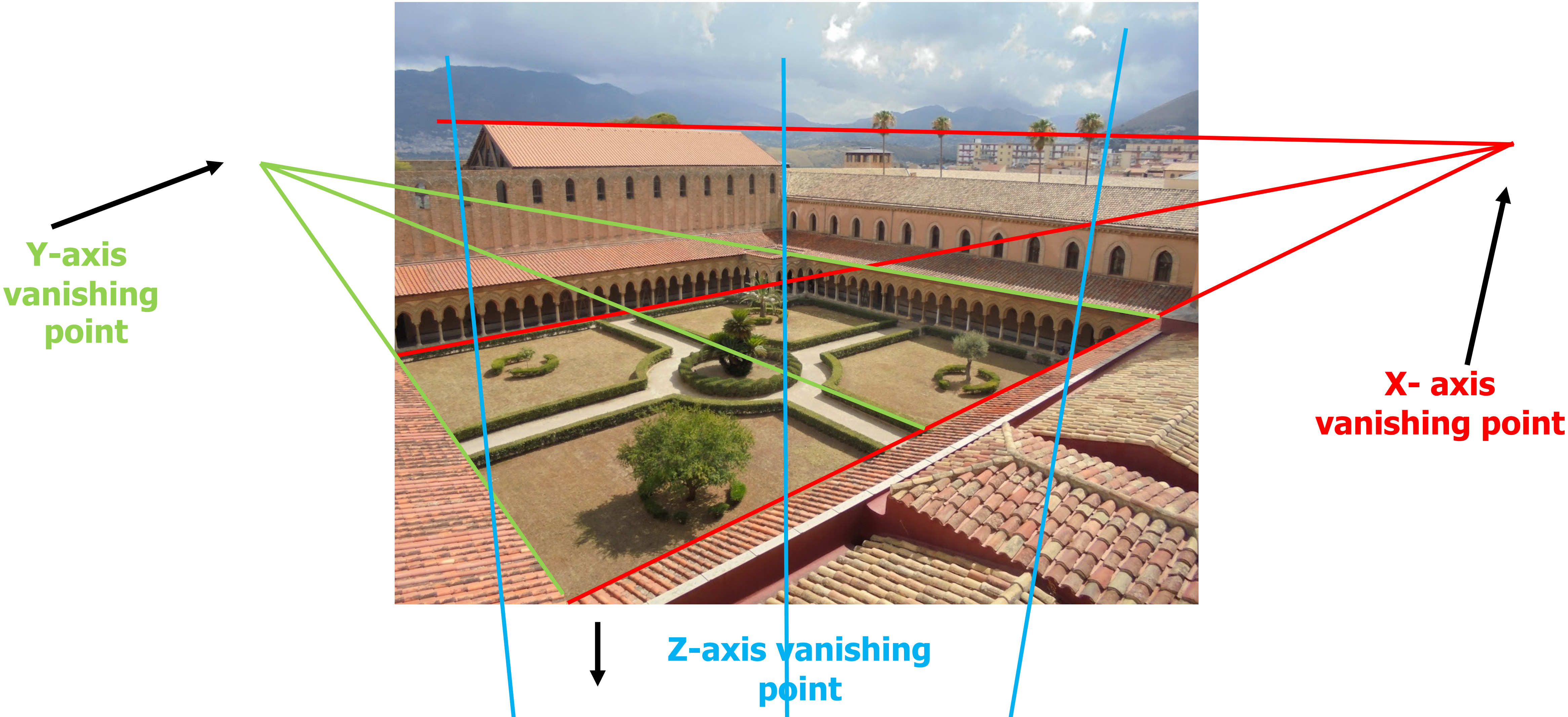
Railroad: parallel lines

Vanishing points and lines



Slide from Efros, Photo from Criminisi

Vanishing points and lines



Cattedrale di Monreale, Palermo 2014, by Frank Dellaert

Computing the Vanishing Point

$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}$ *is a point on the line*

$\begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}$ gives the direction of the line

η gives the distance along the line

- Suppose a line in 3D is specified by the equation

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + \eta \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}$$

- The image coordinates (using the perspective projection equations) of a point on the line are given by:

$$u = F\frac{x}{z} = F\frac{x_0 + \eta n_x}{z_0 + \eta n_z}, \qquad v = F\frac{y}{z} = F\frac{y_0 + \eta n_y}{z_0 + \eta n_z}$$

- Now, compute the limit as $\eta \rightarrow \infty$

$$u_\infty = \lim_{\eta\rightarrow\infty} F\frac{x_0 + \eta n_x}{z_0 + \eta n_z} = \lim_{\eta\rightarrow\infty} F\frac{\frac{x_0}{\eta} + n_x}{\frac{z_0}{\eta} + n_z} = F\frac{n_x}{n_z}$$

- Similarly, $v_\infty = F\frac{n_y}{n_z}$

4. Stereo Vision

- Humans use stereo vision
- Very useful in computer vision as well as it eliminates scale ambiguity

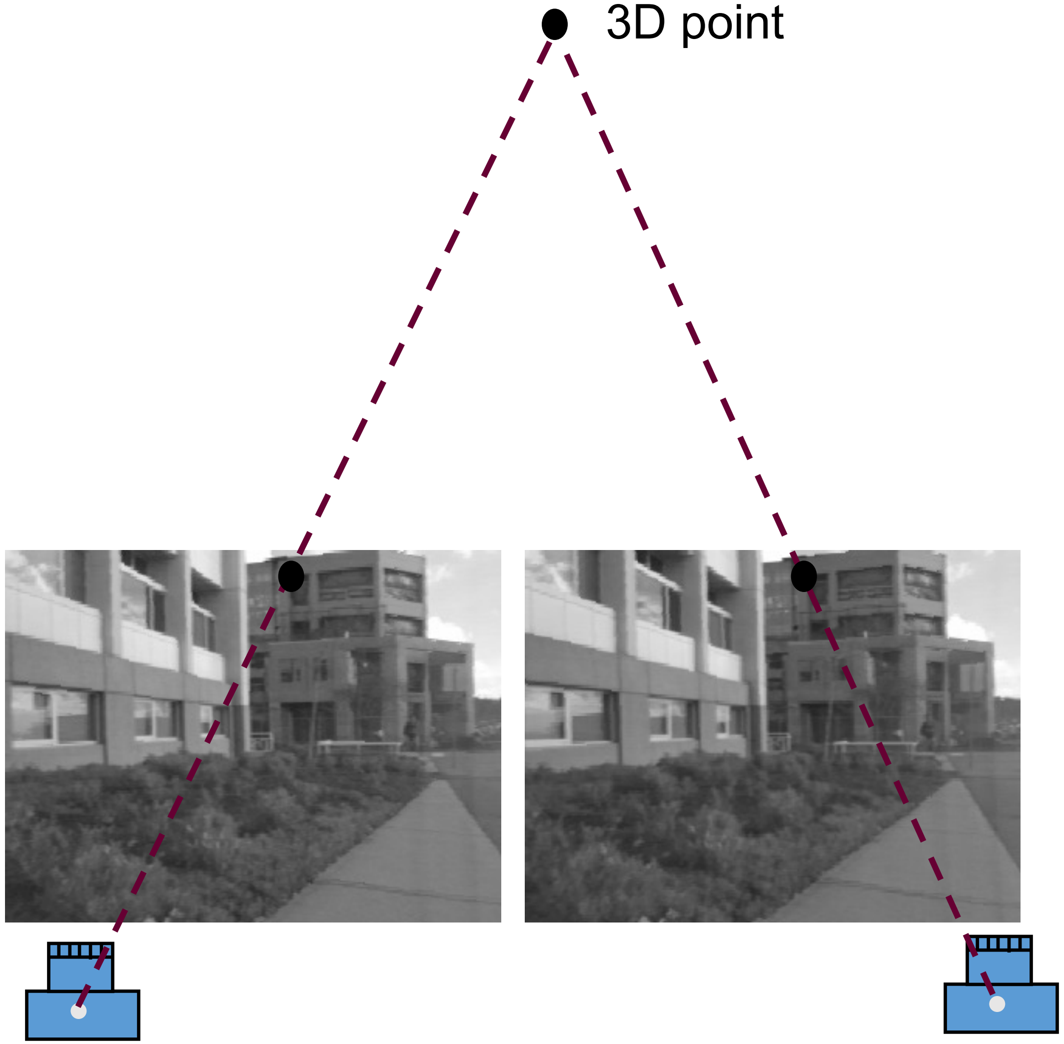
Many slides adapted from F&P and Sing Bing Kang guest lecture

Etymology

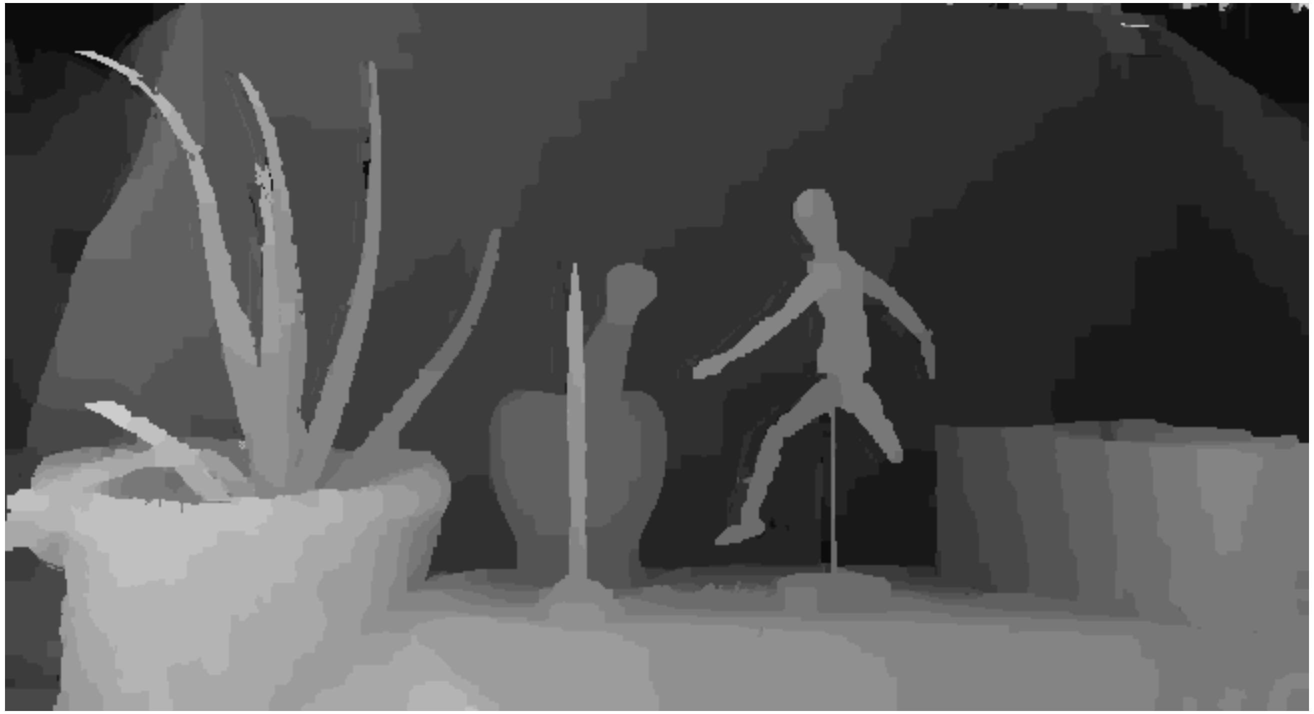
Stereo comes from the Greek word for *solid* (στερεο), and the term can be applied to any system using more than one channel

Effect of Moving Camera

- As camera is shifted (viewpoint changed):
  - 3D points are projected to different 2D locations
  - Amount of shift in projected 2D location depends on depth
- 2D shifts= stereo disparity



Example



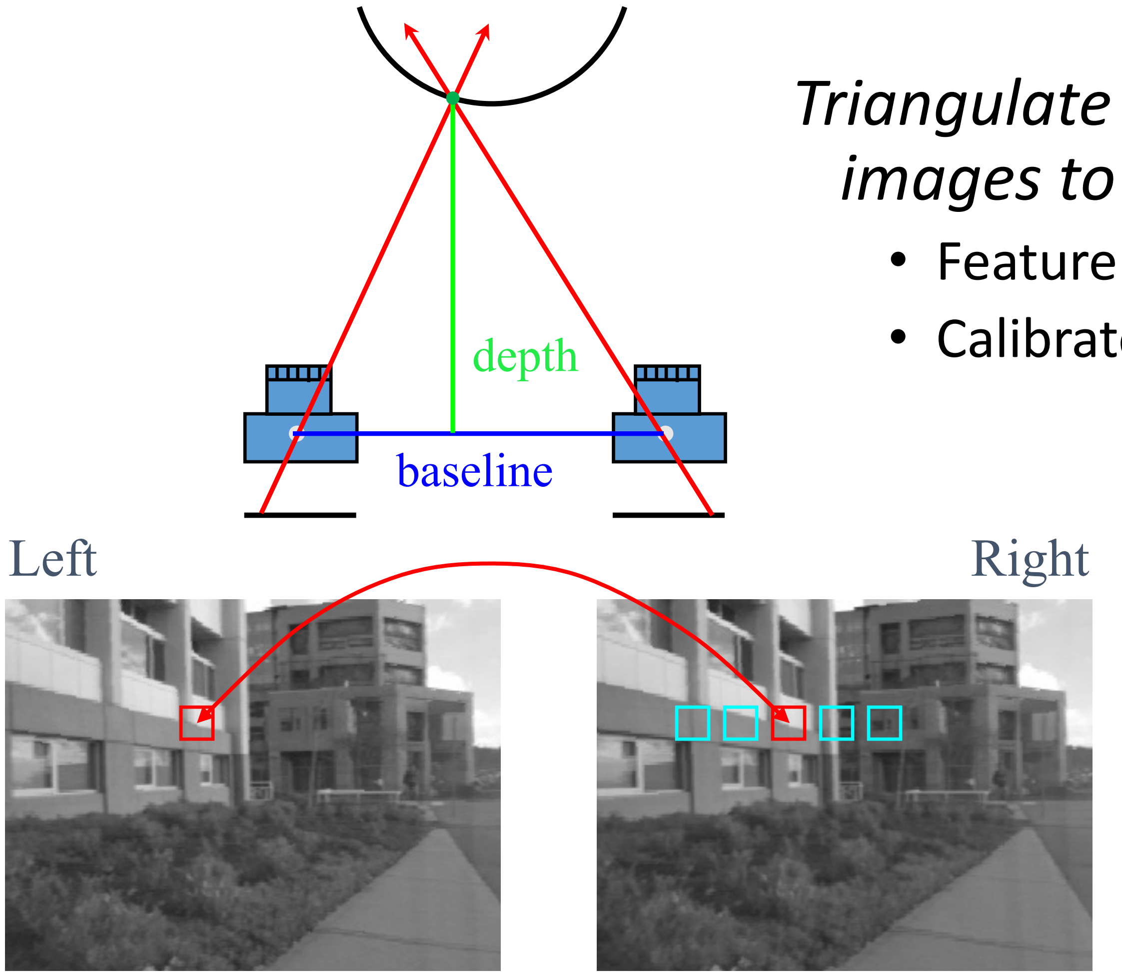
Right image

Left image

Disparity

View Interpolation

Basic Idea of Stereo



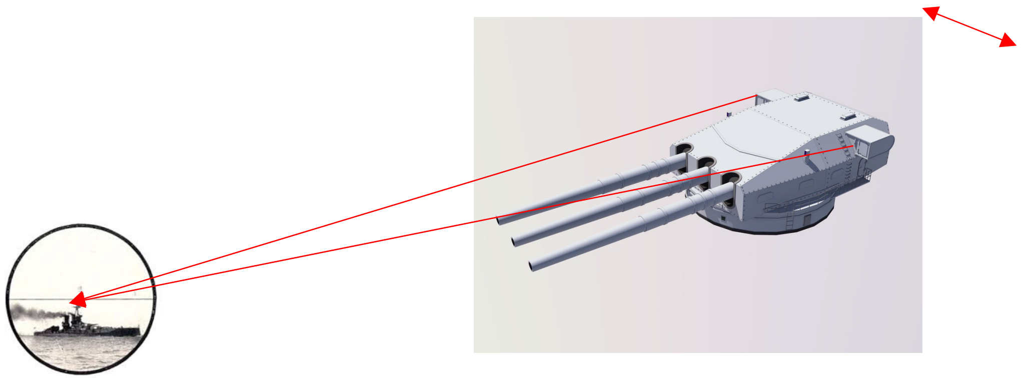
Triangulate the same point on two images to recover depth.

- Feature matching across views
- Calibrated cameras

Matching correlation windows across scan lines

Why is Stereo Useful?

- Passive and non-invasive
- Robot navigation (path planning, obstacle detection)
- 3D modeling (shape analysis, reverse engineering, visualization)
- Photorealistic rendering

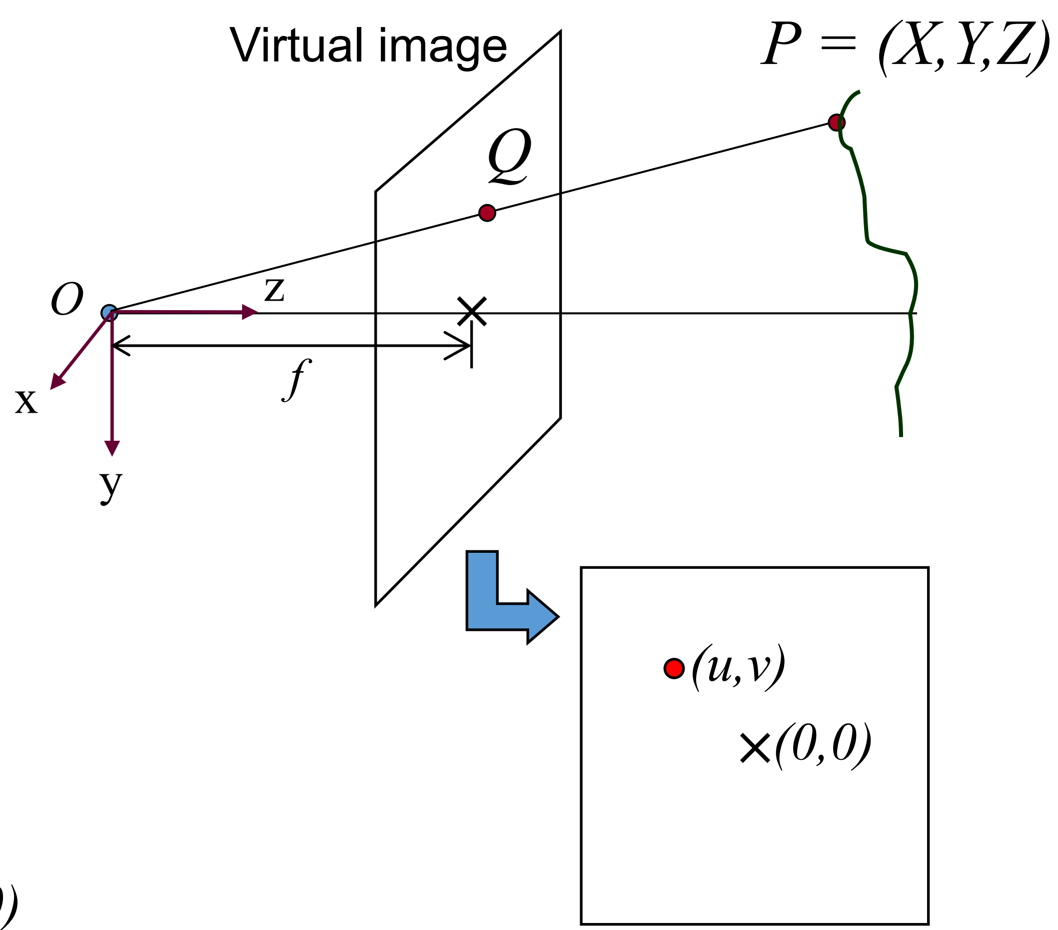


5. Stereo Geometry

- Recall: Pinhole model
- Now we have two !
- How to recover depth from two measurements?

Review: Pinhole Camera Model

3D scene point P is projected to a 2D point Q in the virtual image plane

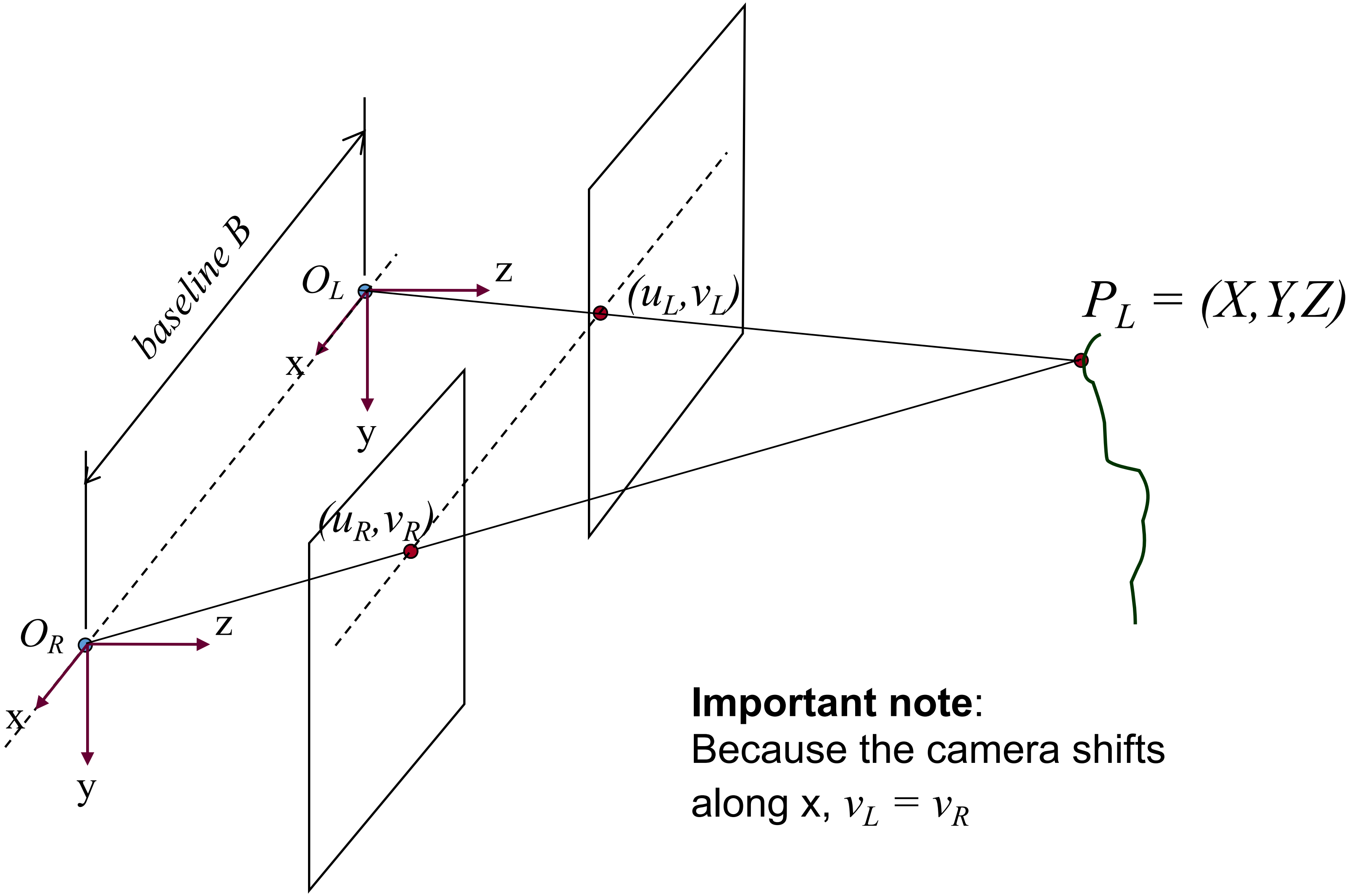


The 2D coordinates in the image are given by

$$(u, v) = \left(f\frac{X}{Z}, f\frac{Y}{Z}\right)$$

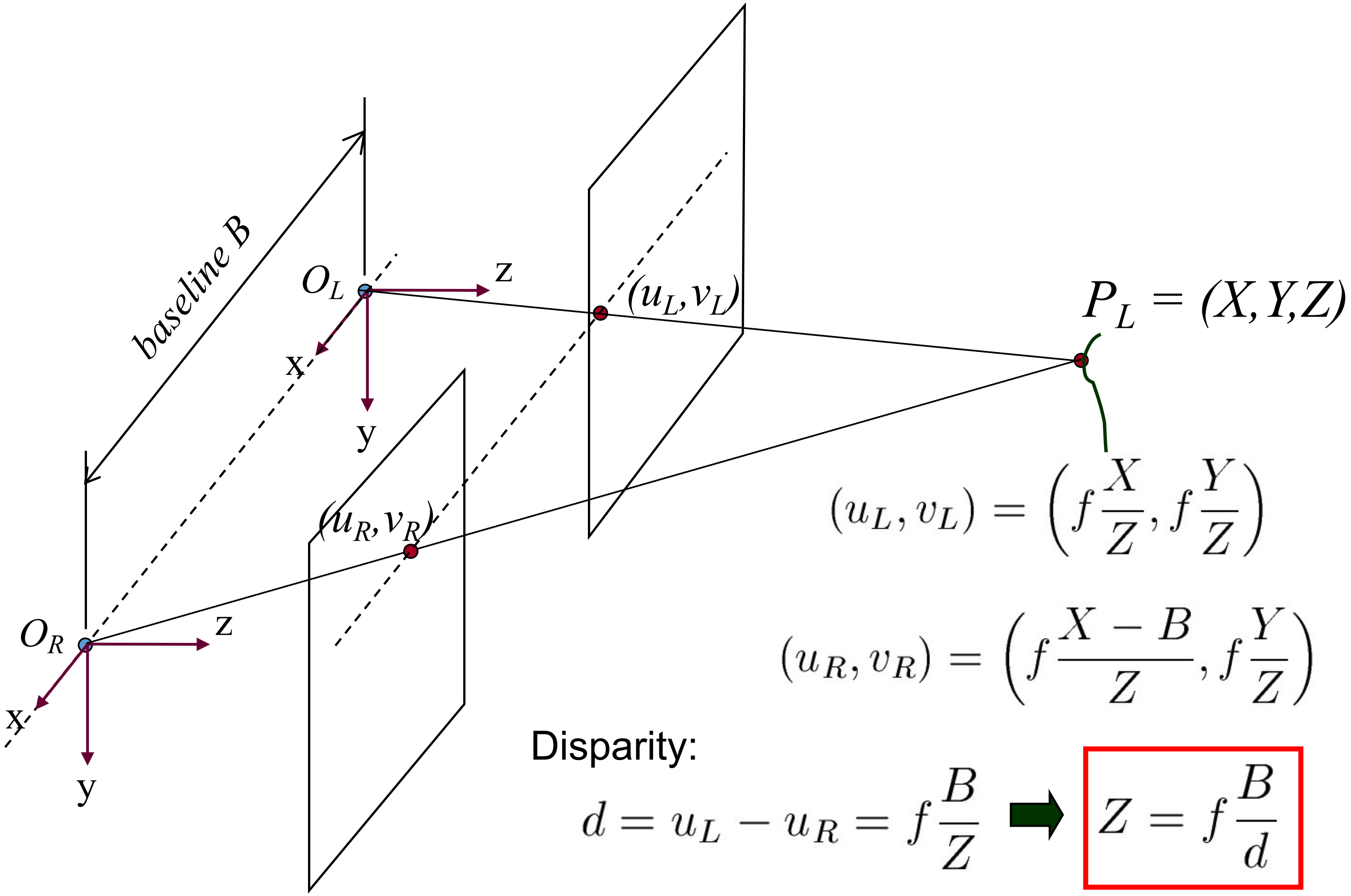
Note: image center is *(0,0)*

Basic Stereo Derivations



Important note:
Because the camera shifts along x, $v_L = v_R$

Basic Stereo Formula



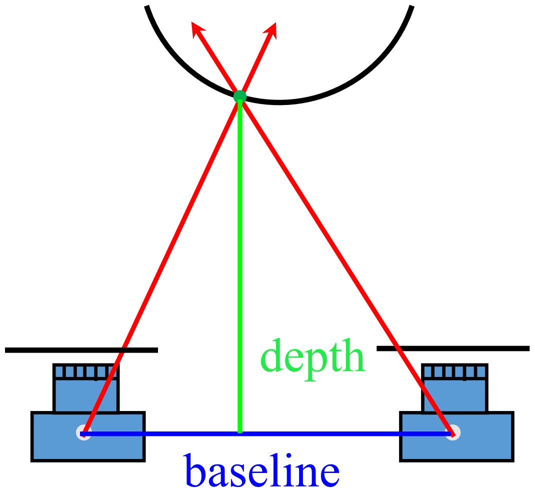
$$(u_L, v_L) = \left(f\frac{X}{Z}, f\frac{Y}{Z}\right)$$

$$(u_R, v_R) = \left(f\frac{X-B}{Z}, f\frac{Y}{Z}\right)$$

Disparity:

$$d = u_L - u_R = f\frac{B}{Z} \Rightarrow \boxed{Z = f\frac{B}{d}}$$

6. Stereo Algorithm



$$Z(x,y) = \frac{f\,B}{d(x,y)}$$

$Z(x, y)$ is depth at pixel (x, y)
$d(x, y)$ is disparity

Matching correlation windows across scan lines

Components of Stereo Algorithms

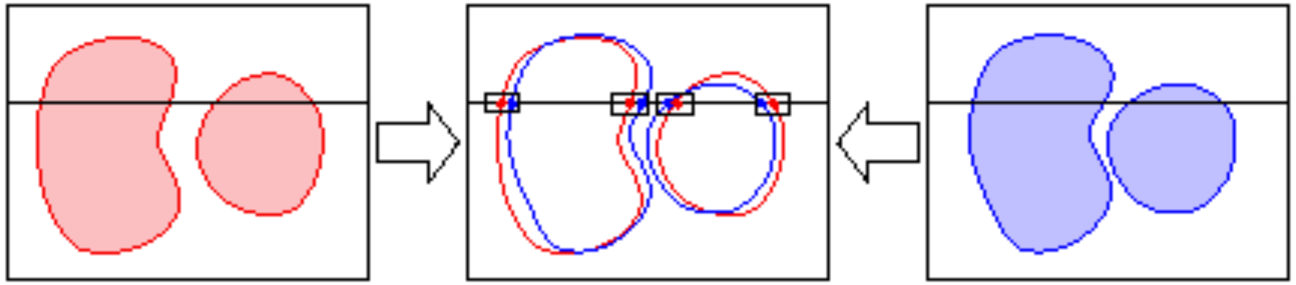
- Matching criterion (error function)
  - Quantify similarity of pixels
  - Most common: direct intensity difference
- Aggregation method
  - How error function is accumulated
  - Options: Pixel, edge, window, or segmented regions
- Optimization and winner selection
  - Examples: Winner-take-all, dynamic programming, graph cuts, belief propagation

Dealing with ambiguities and occlusion

- Ordering constraint:
  - Impose same matching order along scanlines
- Uniqueness constraint:
  - Each pixel in one image maps to unique pixel in other
- Can encode these constraints easily in dynamic programming

Edge-based Stereo

- Another approach is to match *edges* rather than windows of pixels:



- Which method is better?
  - Edges tend to fail in dense texture (outdoors)
  - Correlation tends to fail in smooth featureless areas
  - Sparse correspondences