CS 3630!

Lecture 16:
Computer Vision:
Imaging Geometry

Topics

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

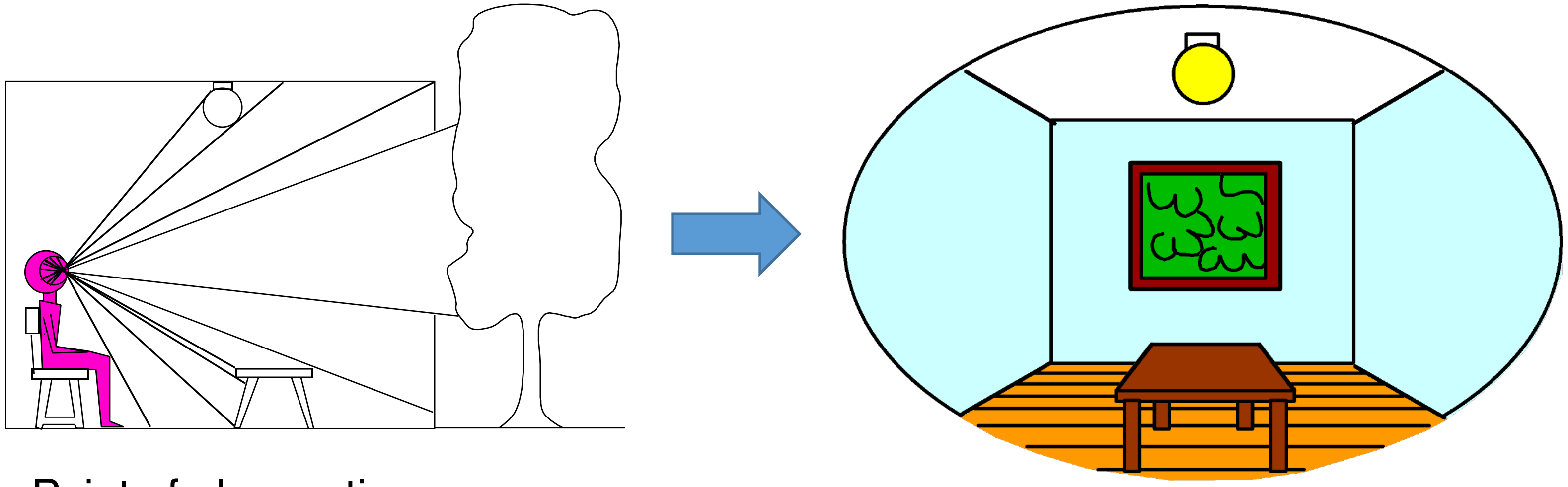
- Many slides borrowed from Frank Dellaert, James Hays, Irfan Essa, Sing Bing Kang and others.

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

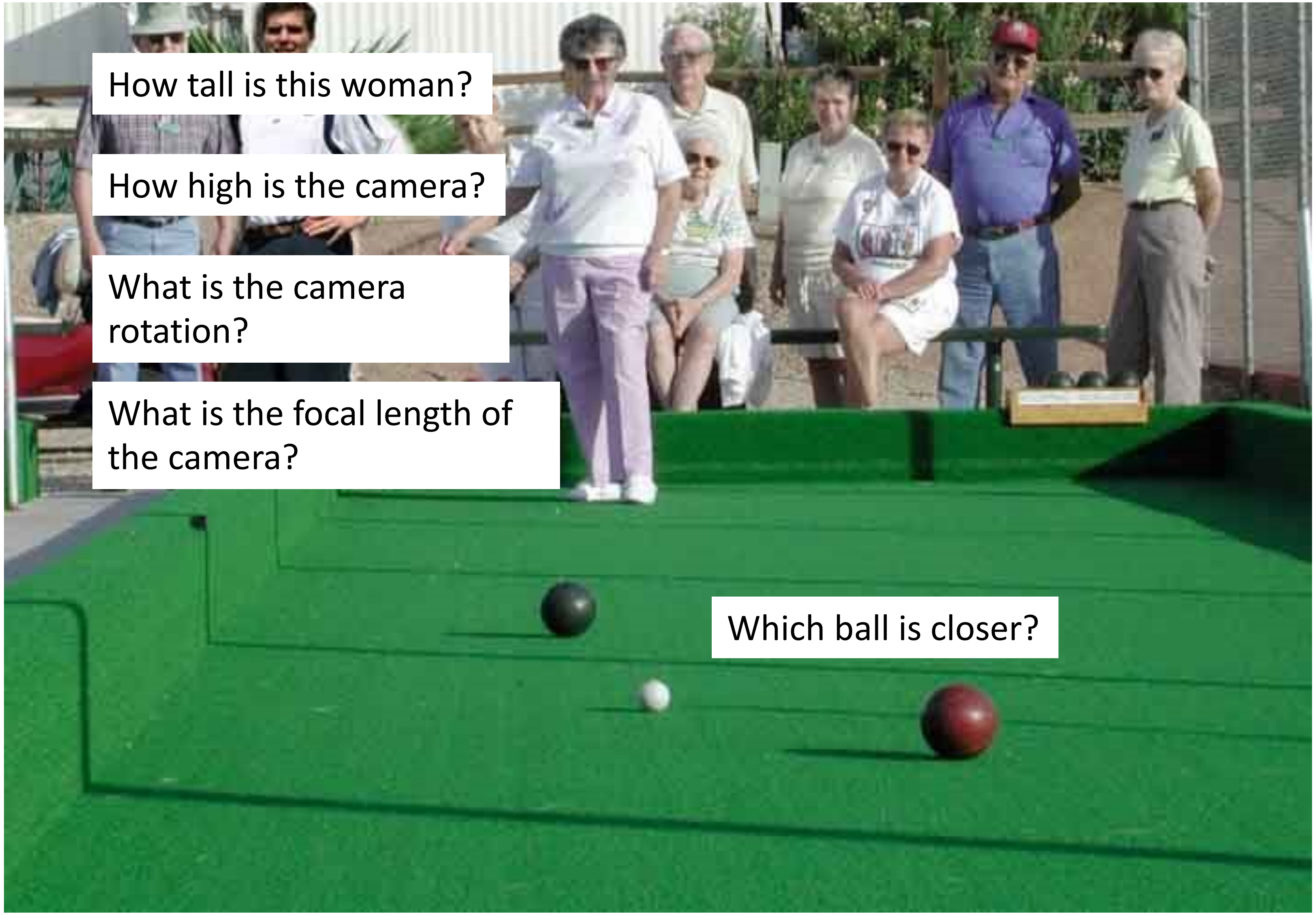
3D world *2D image*



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



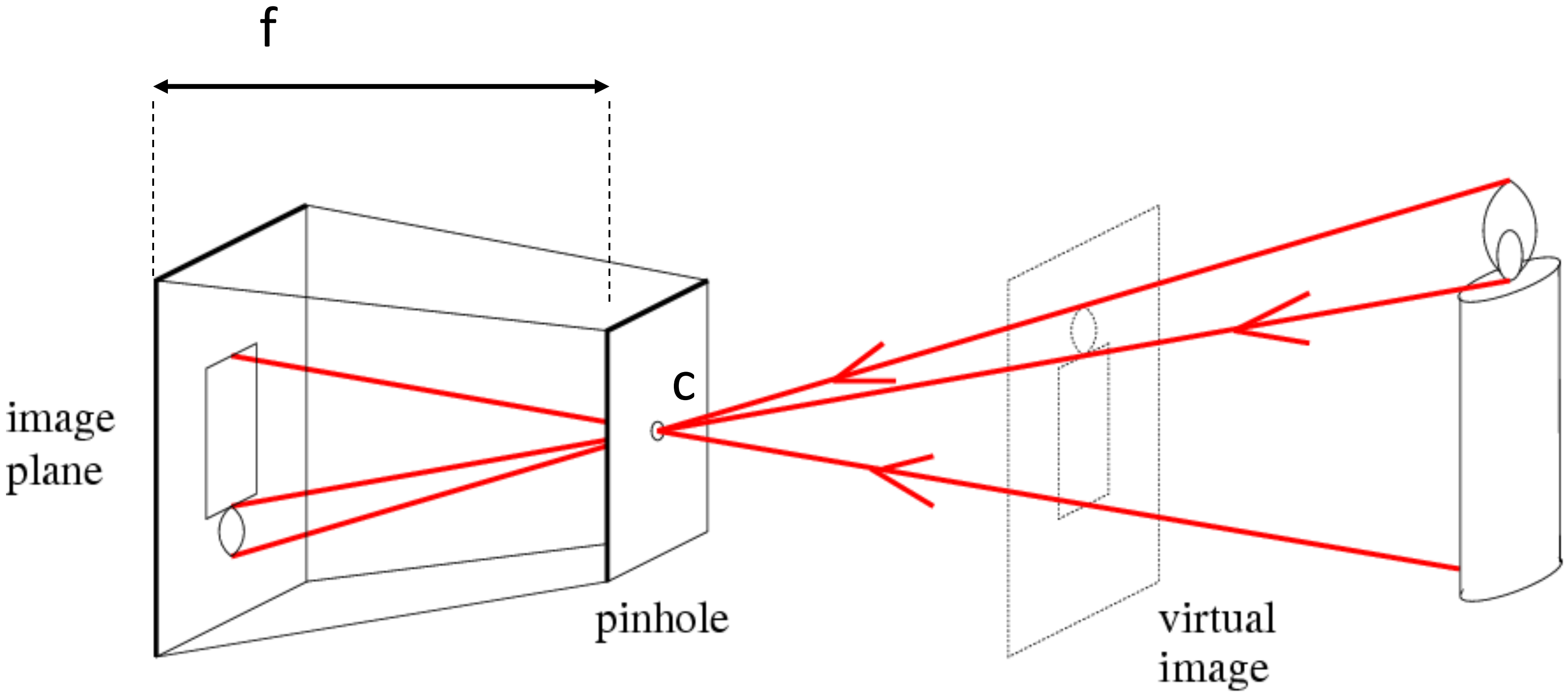
Slide source: Seitz

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

Camera obscura: the pre-camera

- Known during classical period in China and Greece (e.g. Mo-Ti, China, 470BC to 390BC)

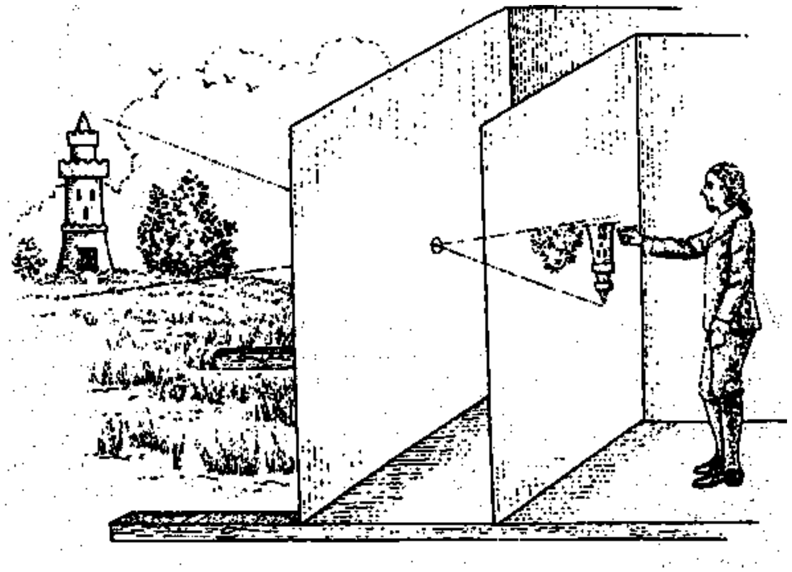
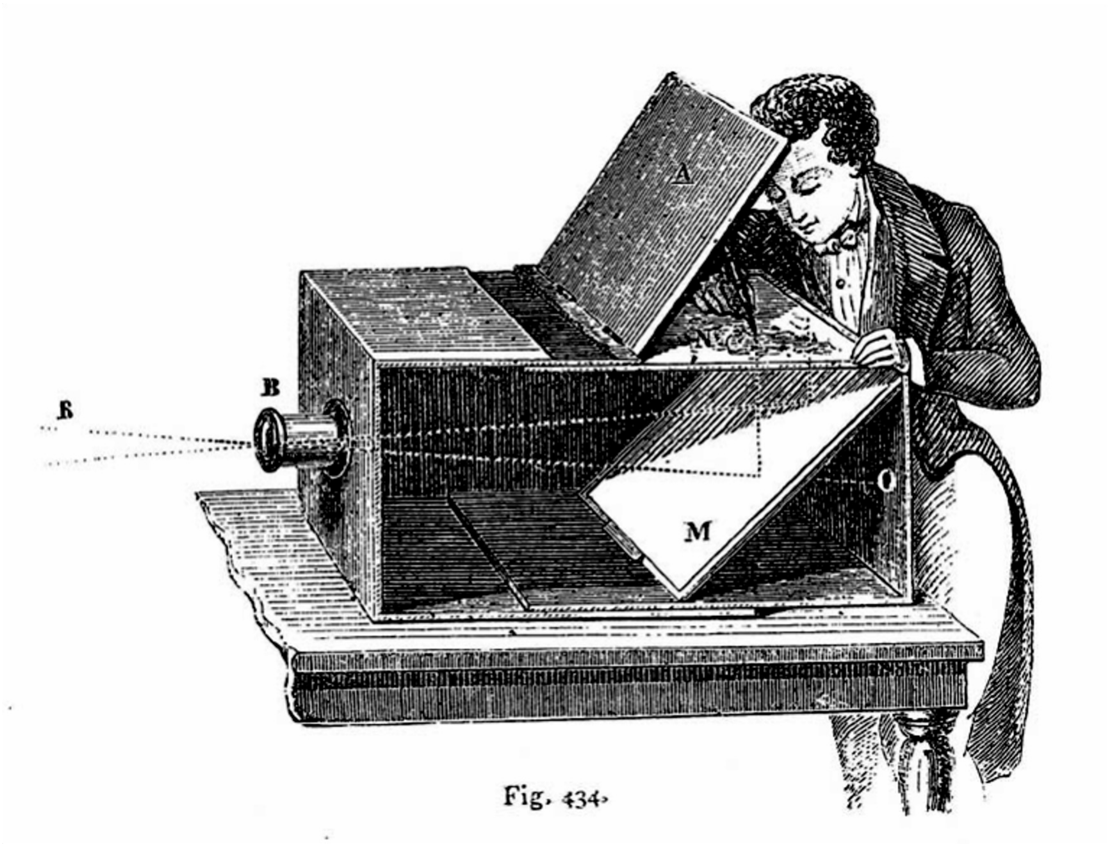


Illustration of Camera Obscura

Freestanding camera obscura at UNC Chapel Hill

Photo by Seth Ilys

Camera Obscura used for Tracing



Lens Based Camera Obscura, 1568

First Photograph

Oldest surviving photograph

- Took 8 hours on pewter plate

Joseph Niepce, 1826

Photograph of the first photograph

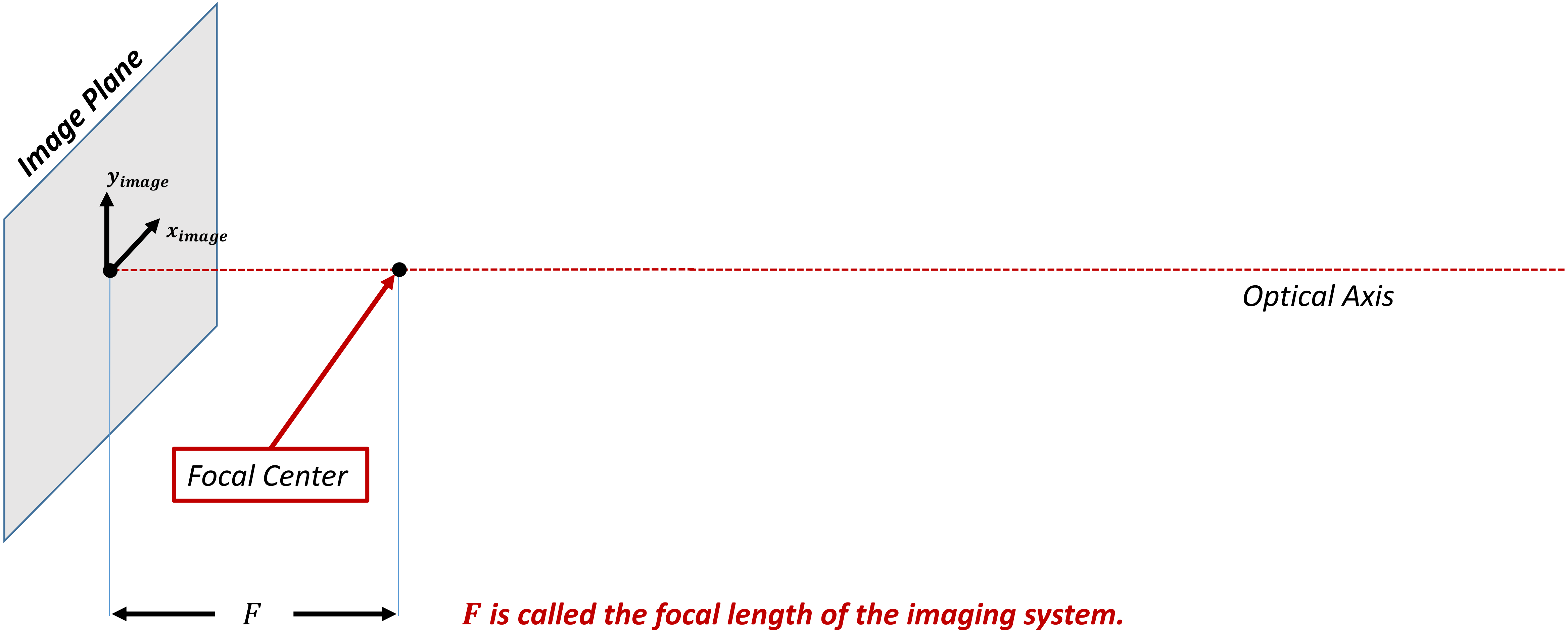
Stored at UT Austin

Niepce later teamed up with Daguerre, who eventually created Daguerrotypes

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.

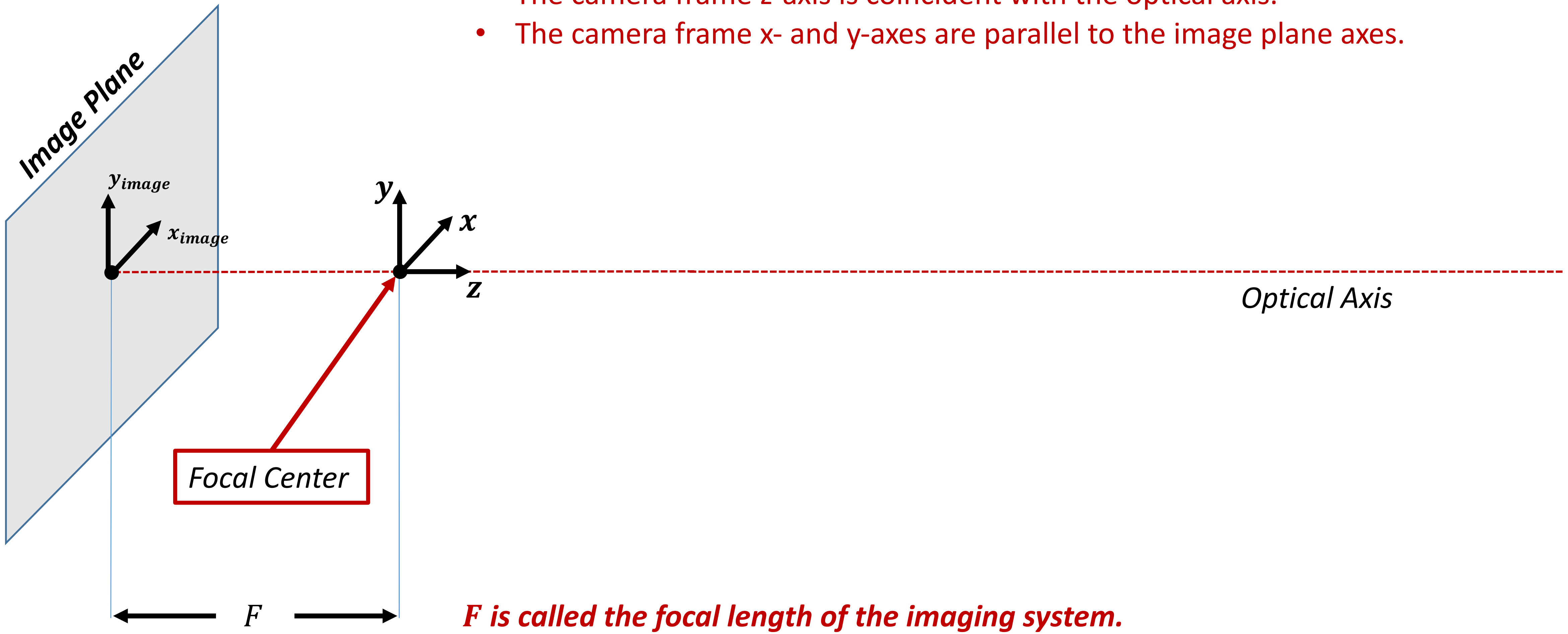


F is called the focal length of the imaging system.

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.

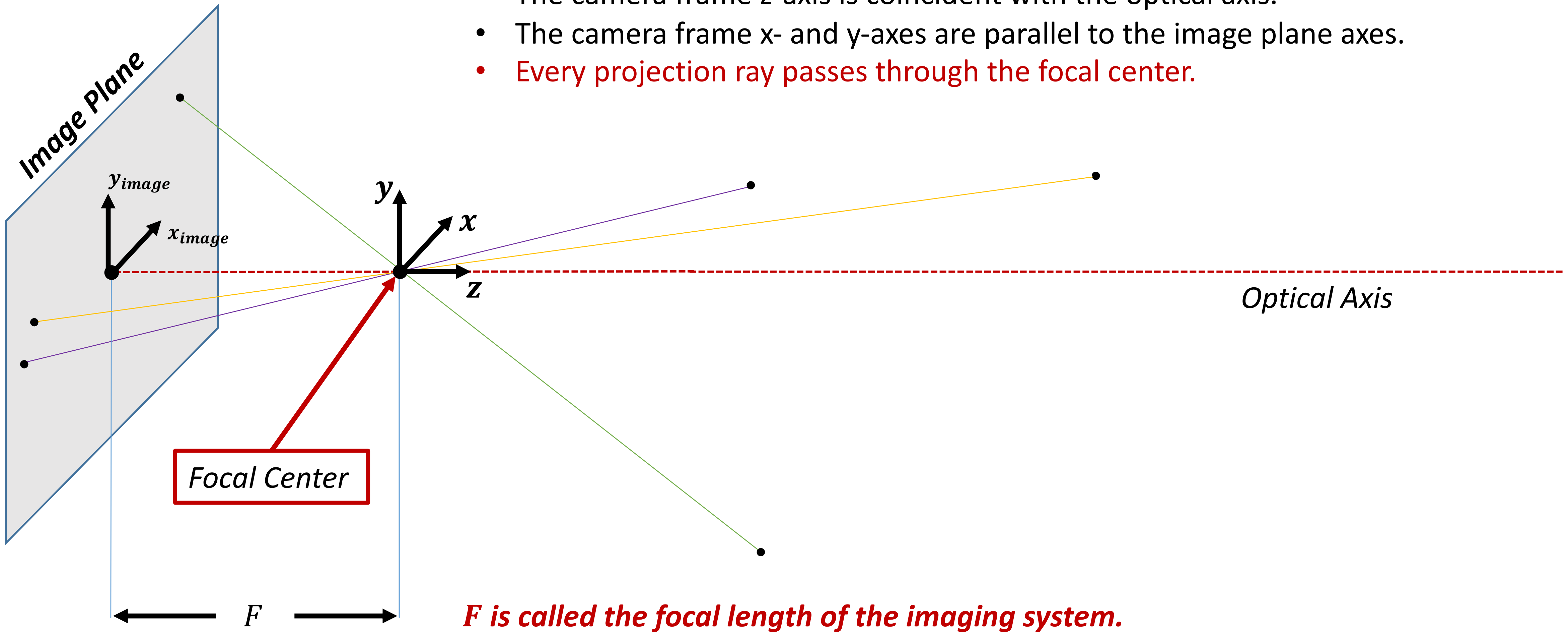


F is called the focal length of the imaging system.

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.

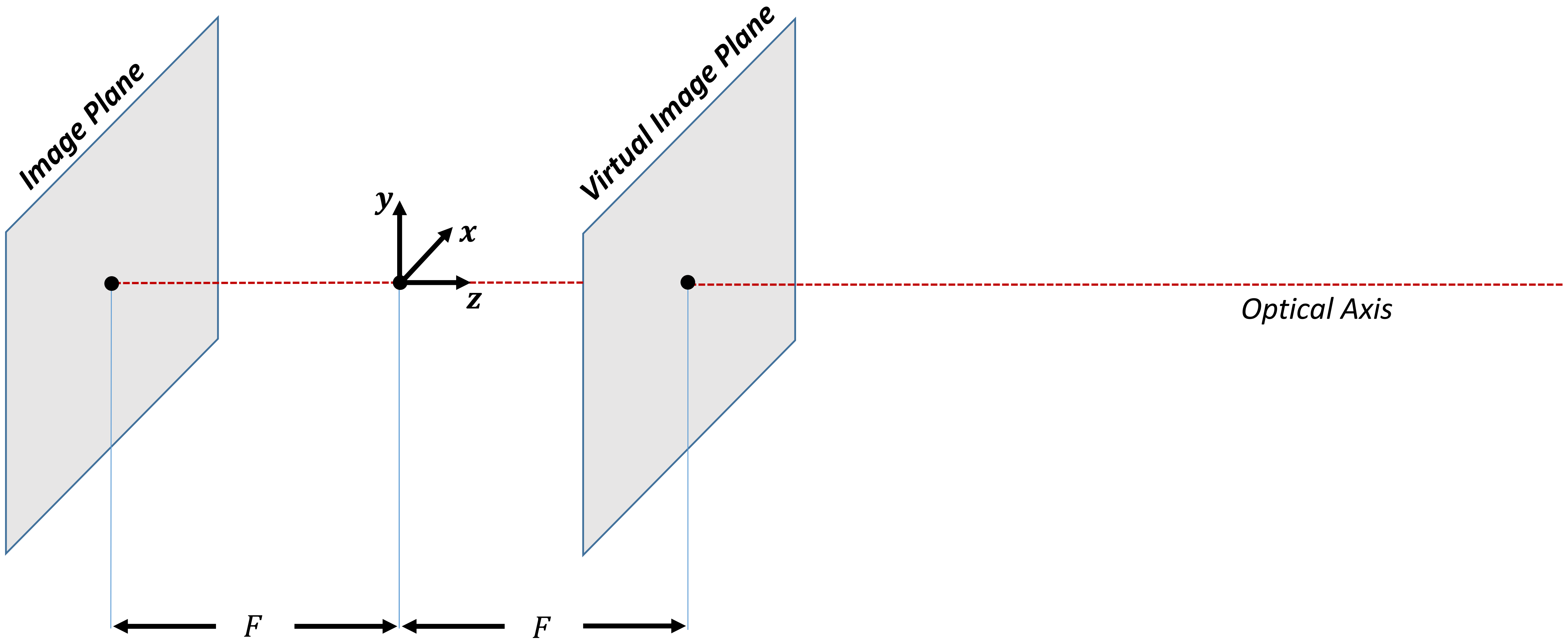


F is called the focal length of the imaging system.

Pinhole Camera

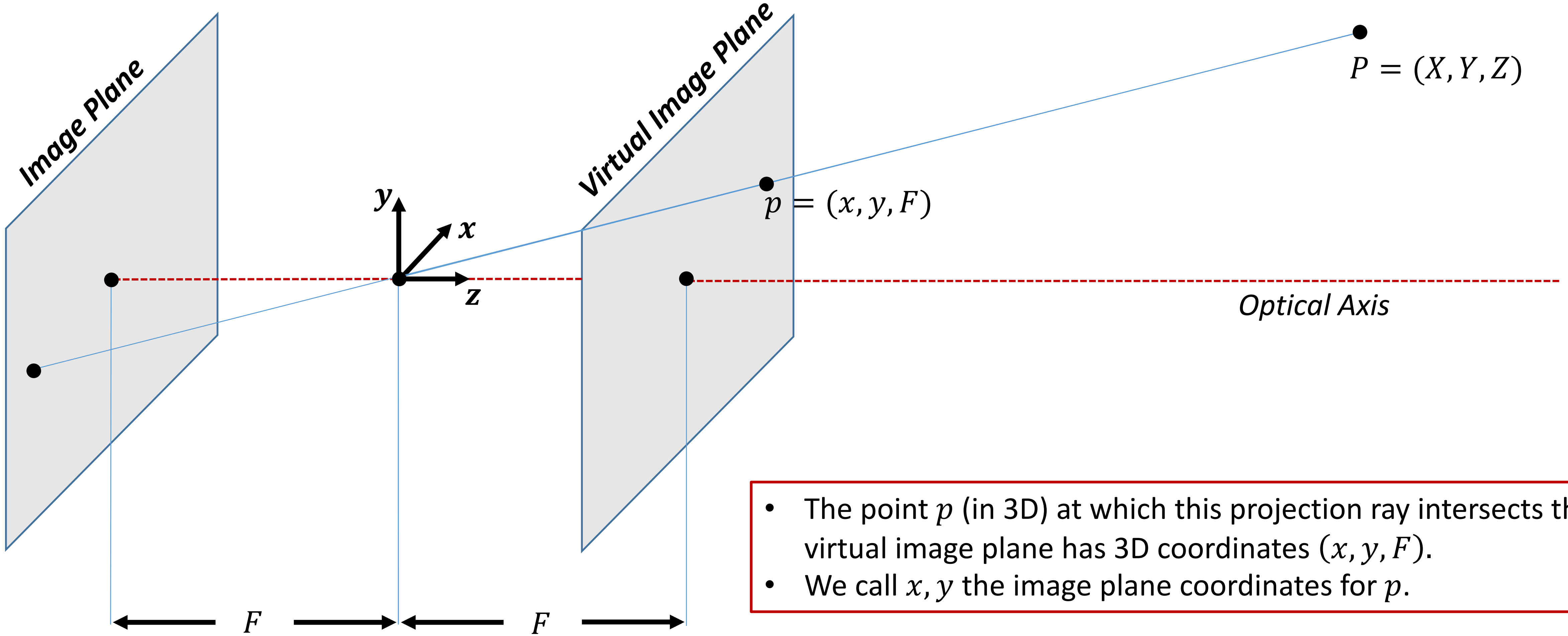
Life is so much easier if we insert a ***virtual image plane*** in front of the focal center.

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.

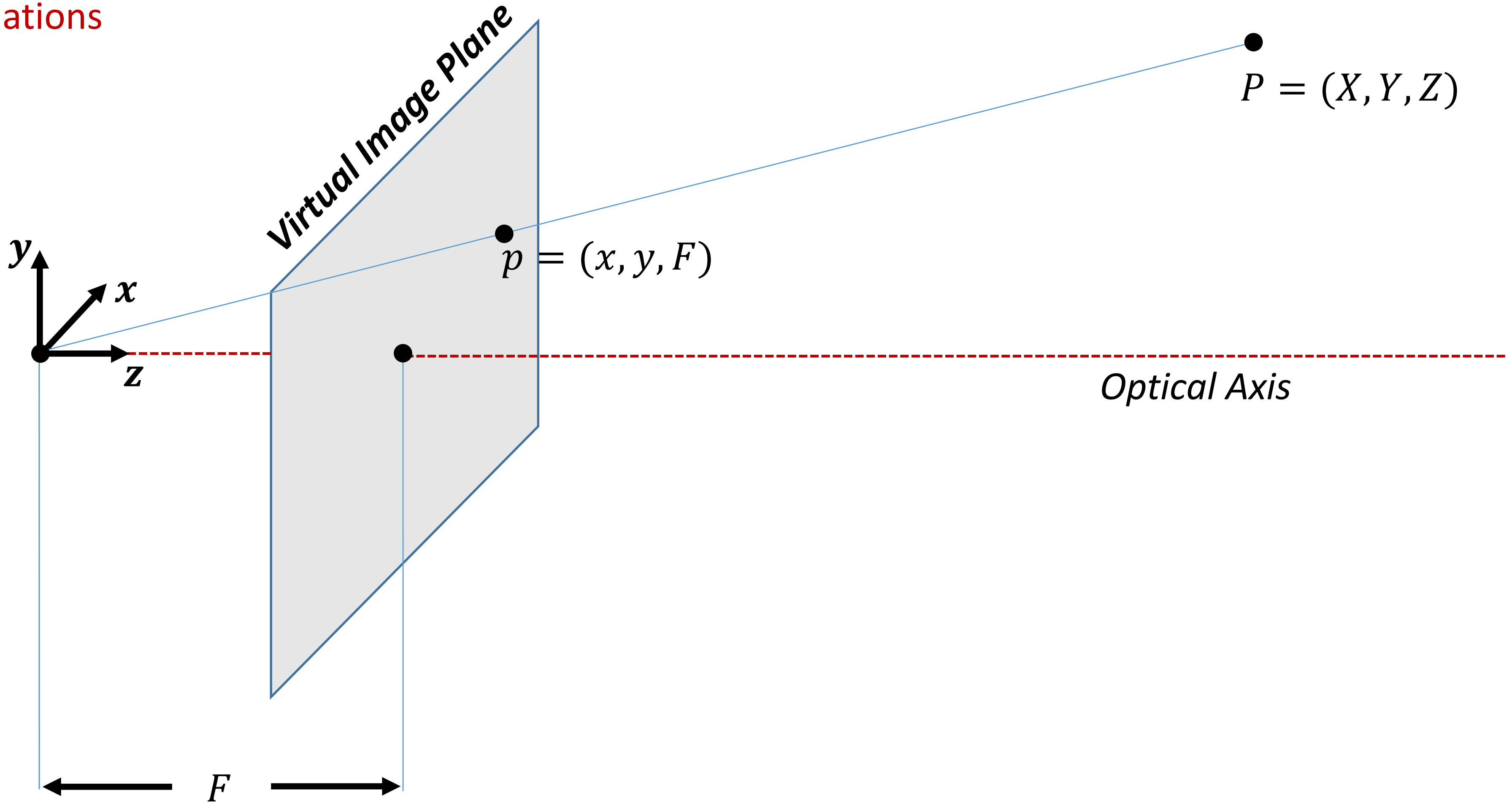
Pinhole Camera

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

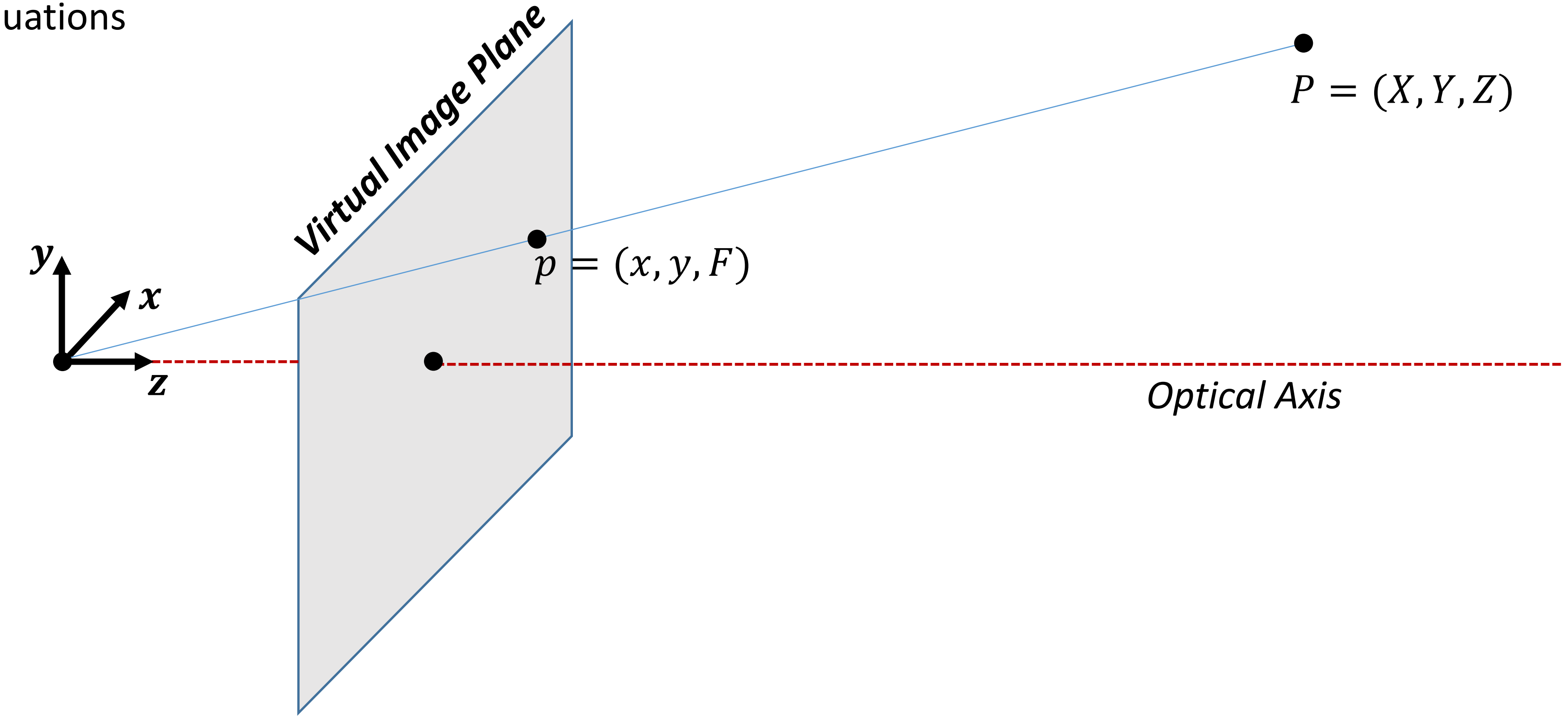
$$\lambda p = P$$

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

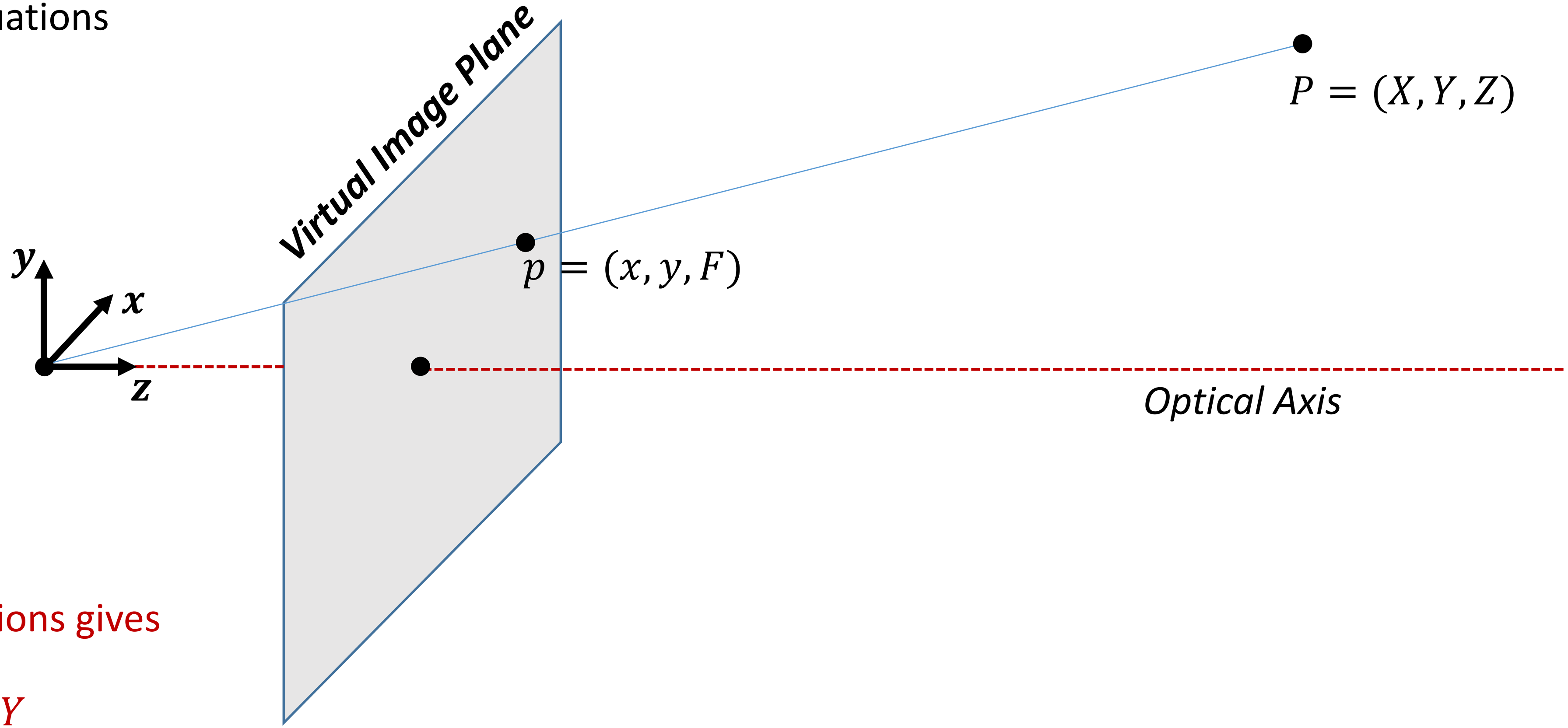
$$\lambda p = P$$

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

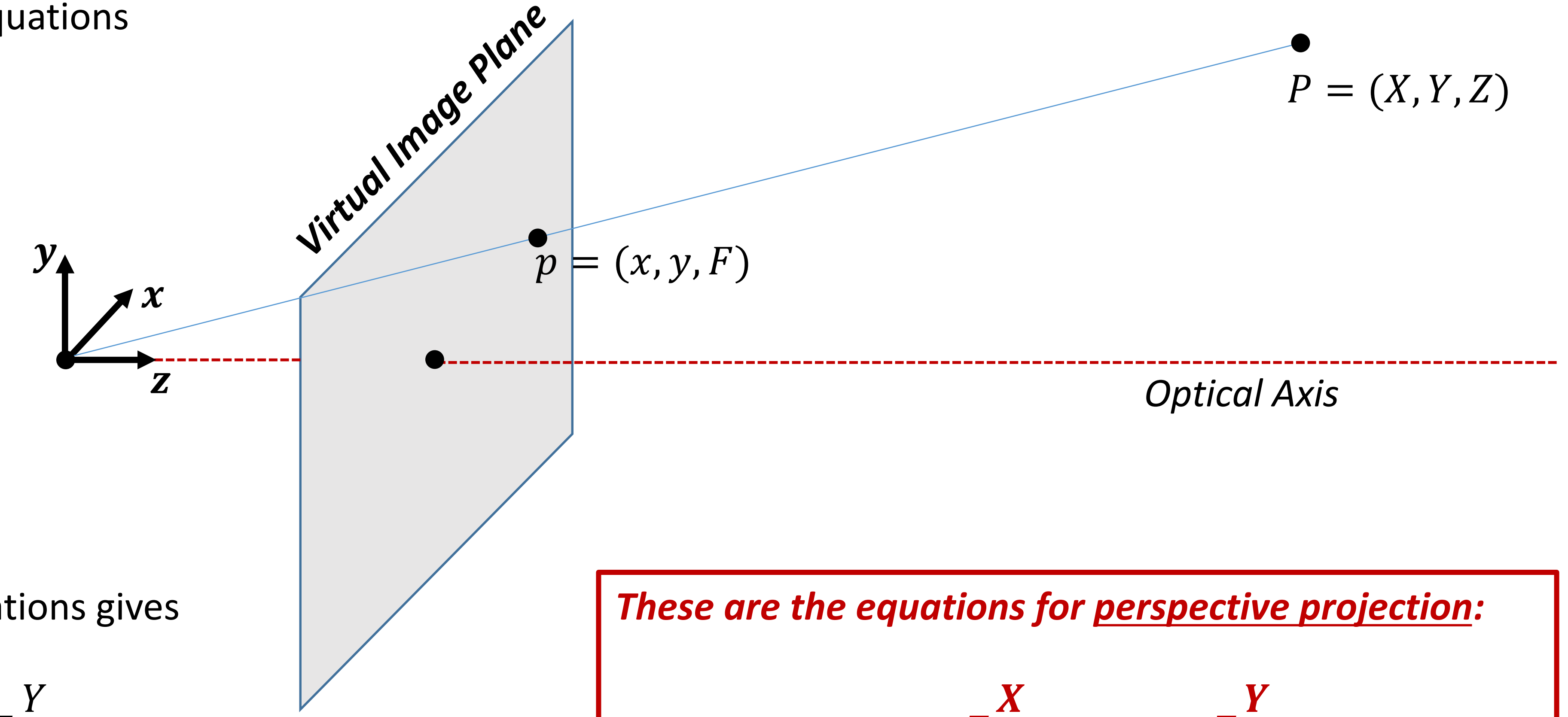
$$\lambda p = P$$

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

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

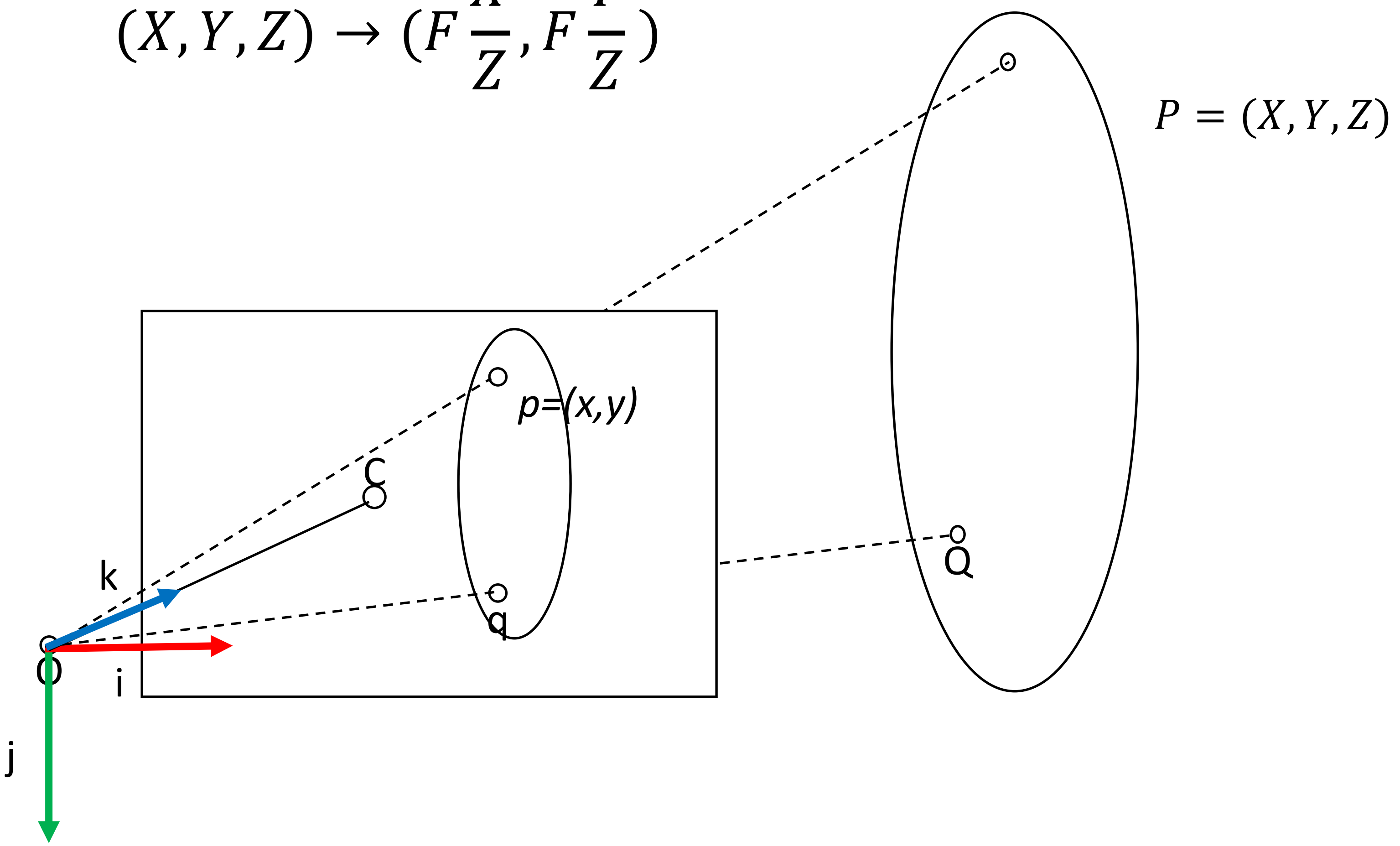
These are the equations for <u>perspective projection</u>:

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

Computer Vision Convention

- Fundamental equation:

$$(X, Y, Z) \rightarrow (F\frac{X}{Z}, F\frac{Y}{Z})$$



Homogeneous Coordinates

Linear transformation of homogeneous (projective) coordinates

$$m = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} I & 0 \end{bmatrix} M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ T \end{bmatrix}$$

Recover image (Euclidean) coordinates by normalizing:

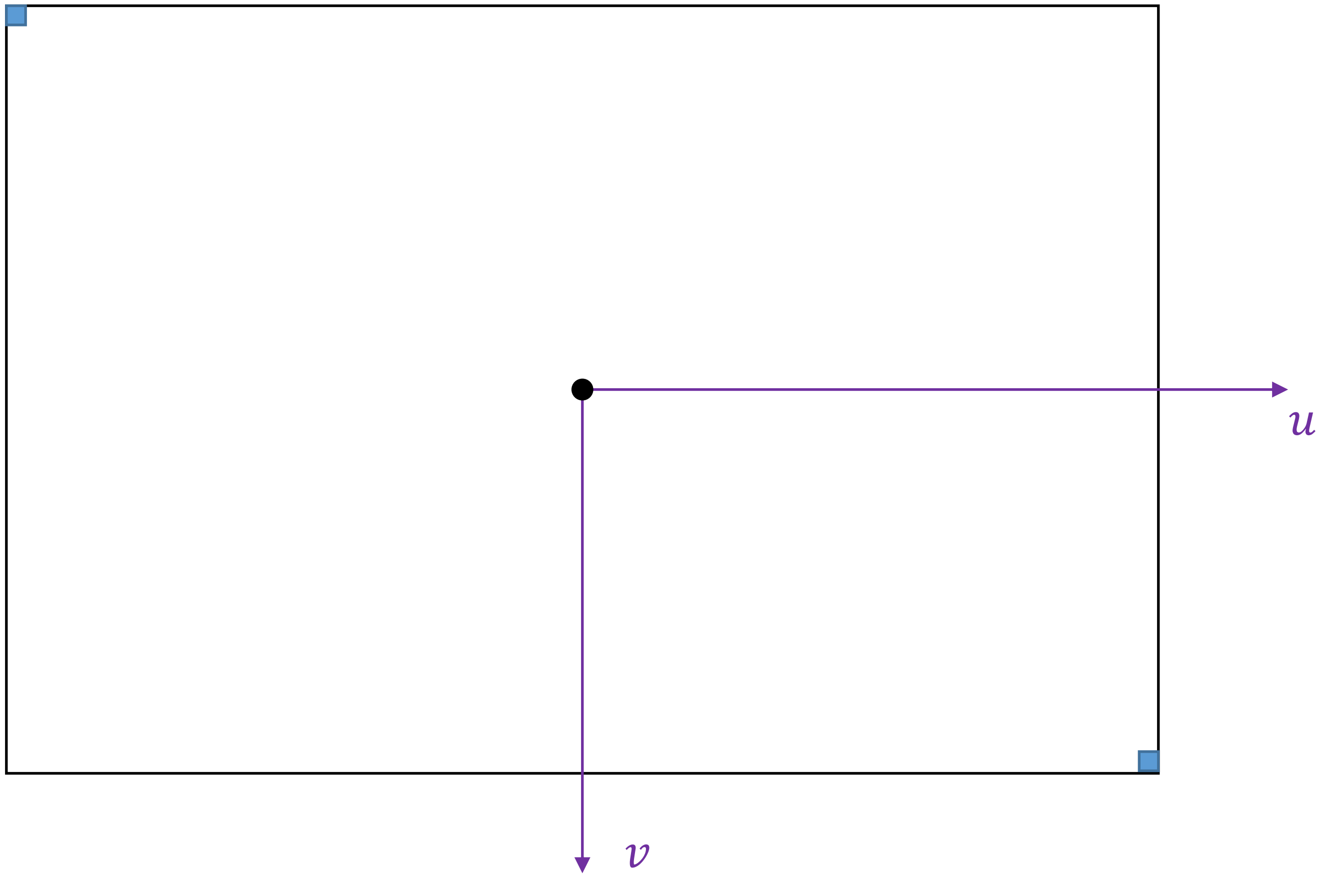
$$x = \frac{u}{w} = \frac{X}{Z}$$

$$y = \frac{v}{w} = \frac{Y}{Z}$$

Sensor coordinates (2D) convention

- 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:

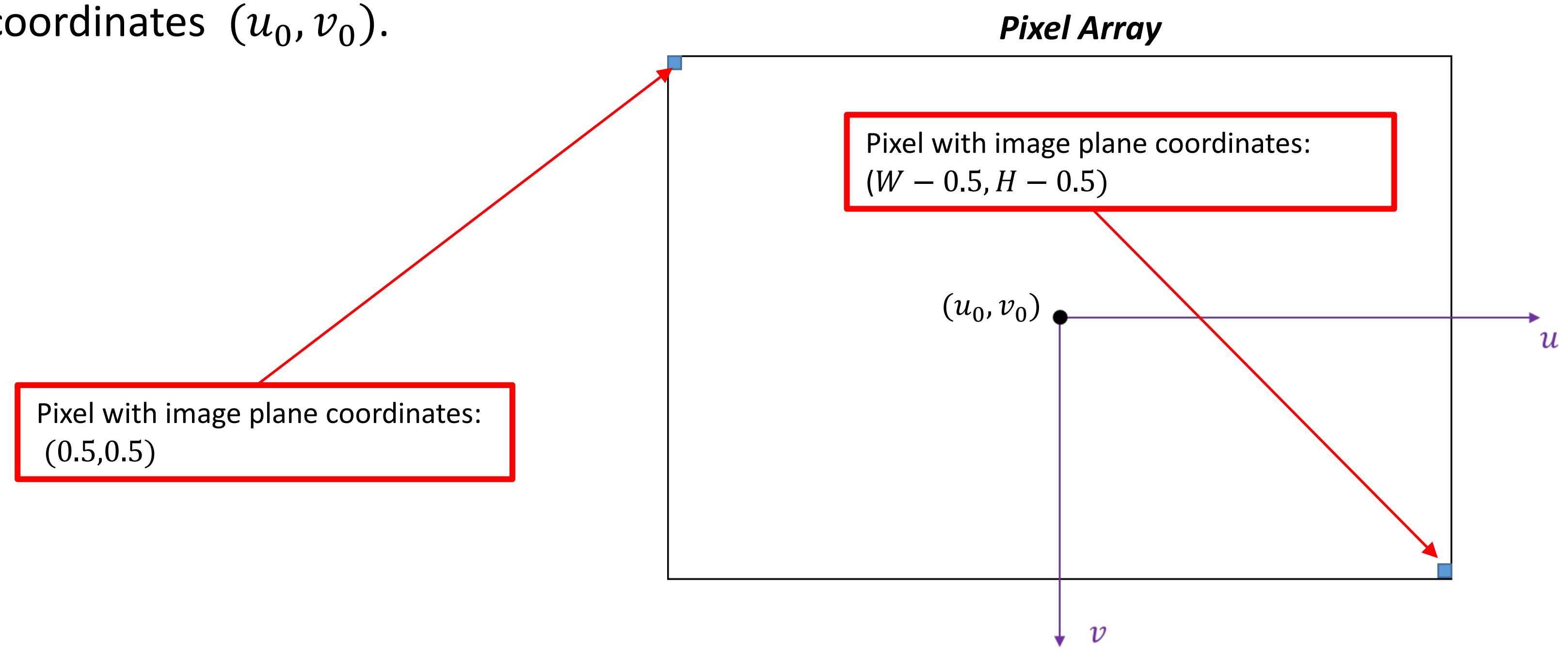


Sensor Coordinates

- 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

From image-plane coordinates to sensor coordinates

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

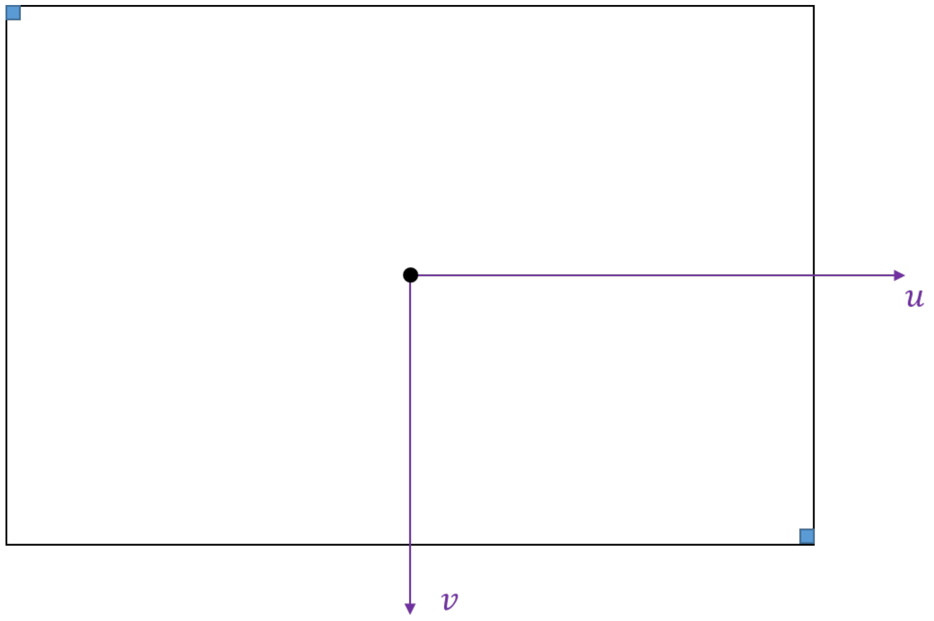
$$u = u_0 + \alpha F \frac{X}{Z}, \qquad v = v_0 + \beta F \frac{Y}{Z}$$

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.

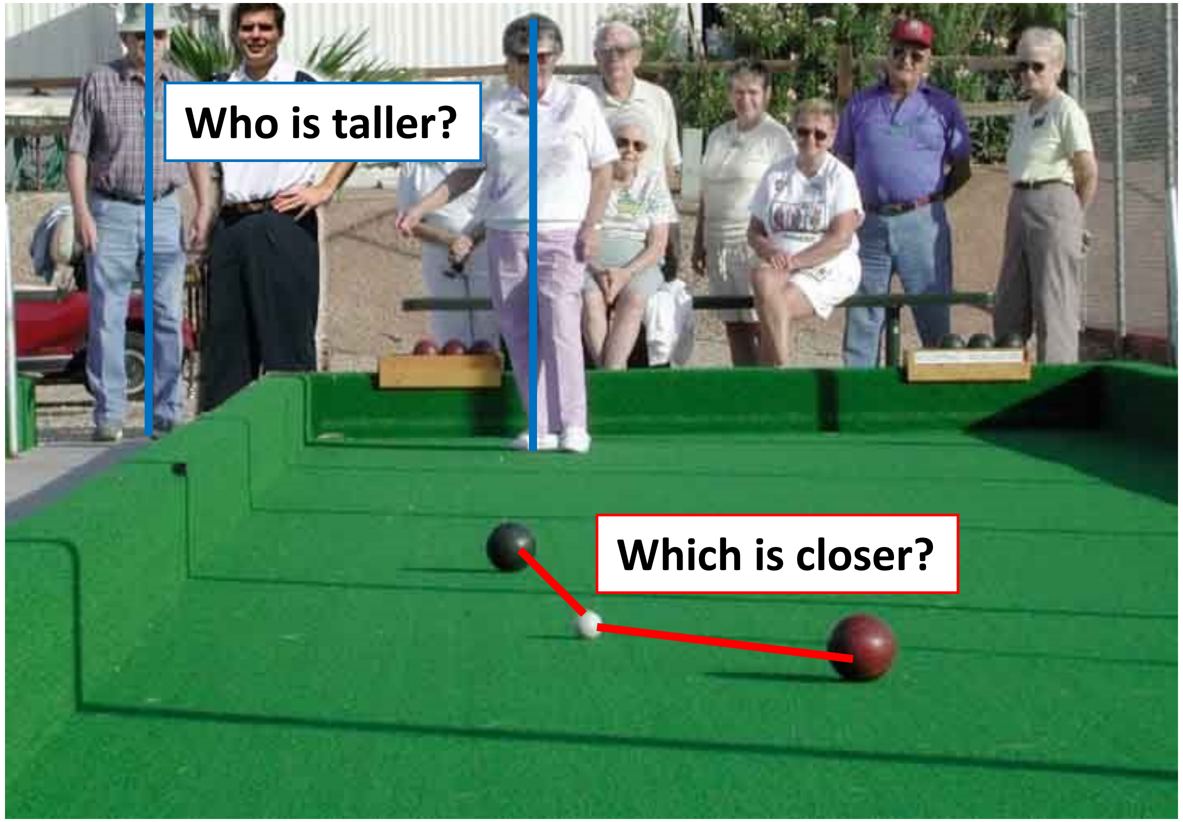
Pixel Array



3. Properties of projective Geometry

What is lost?

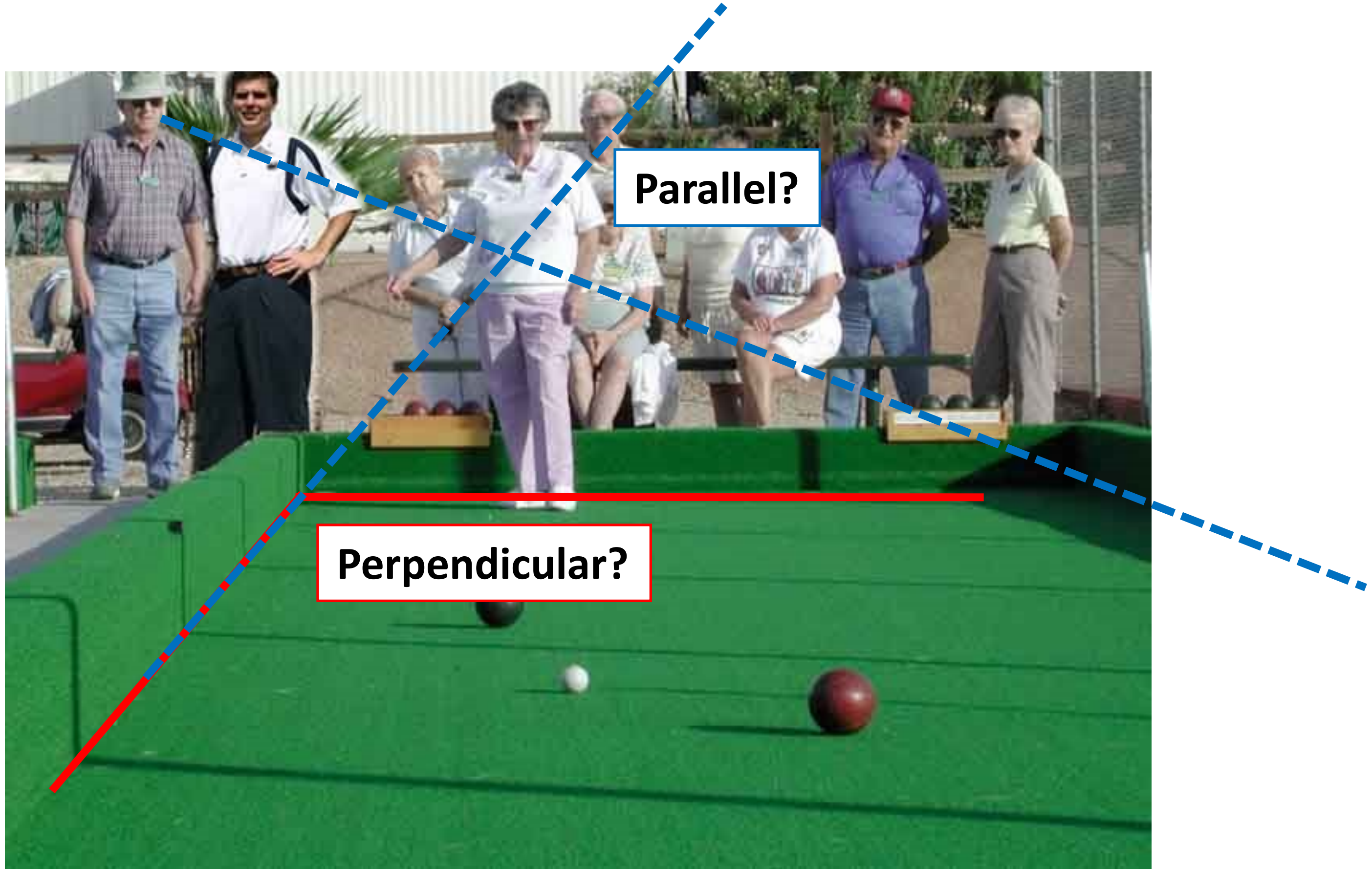
- Length



Properties of projective Geometry

What is lost?

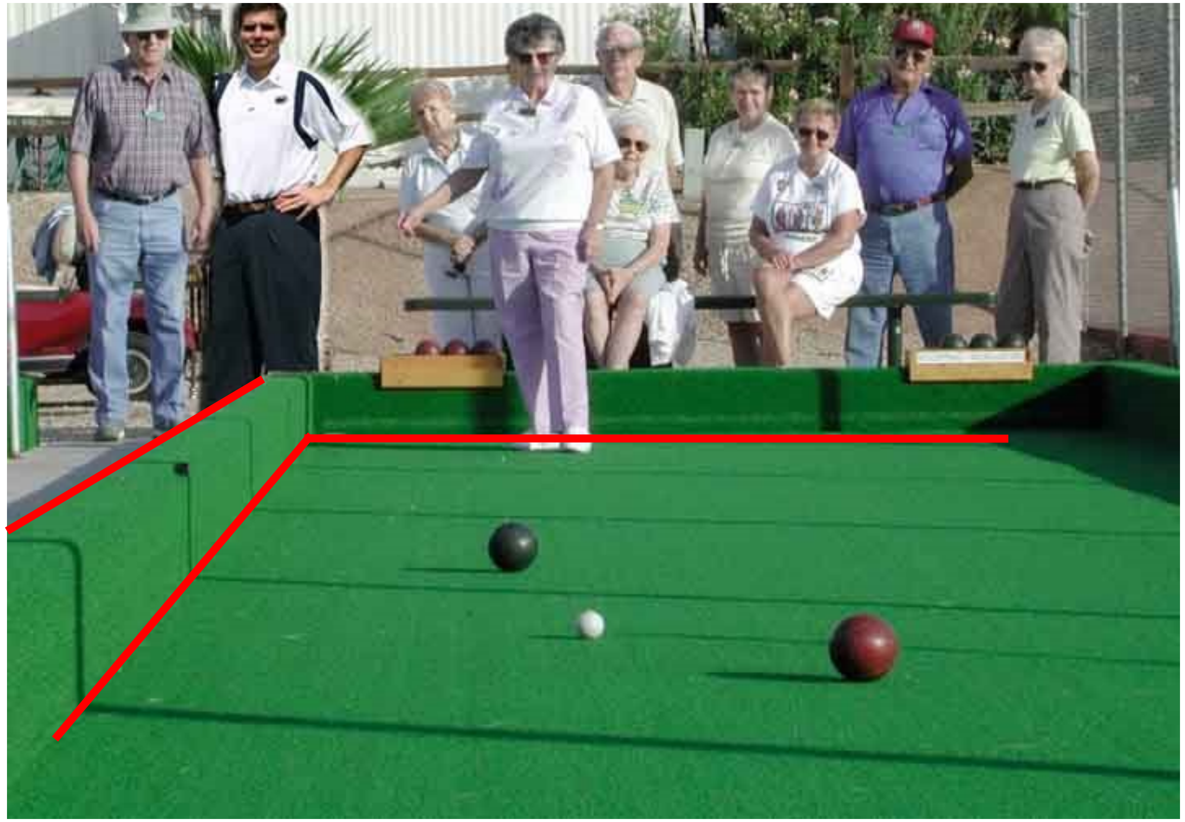
- Length
- Angles



Properties of projective Geometry

What is preserved?

- Straight lines are still straight



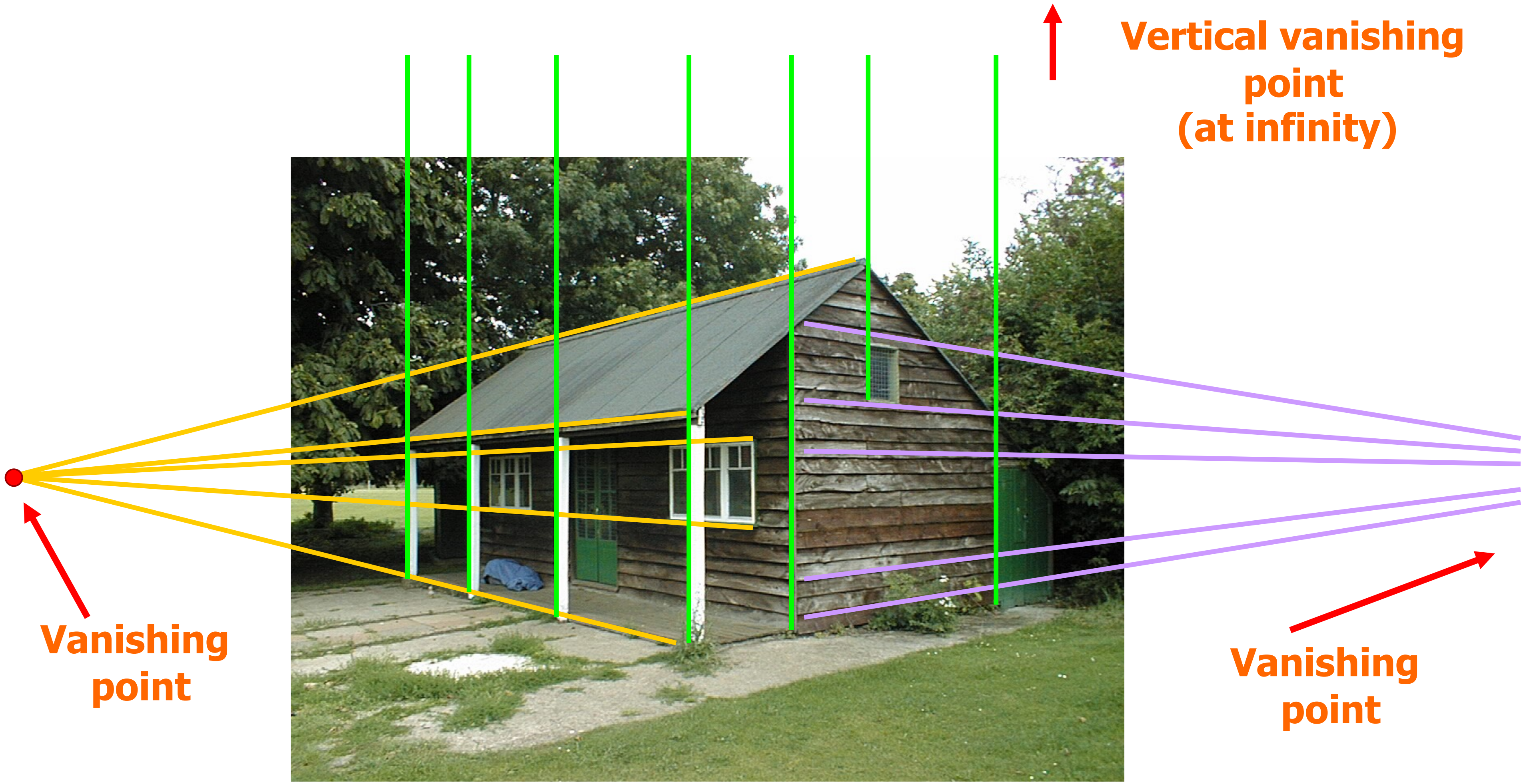
We can see infinity !

Where do parallel lines meet?

At infinity.

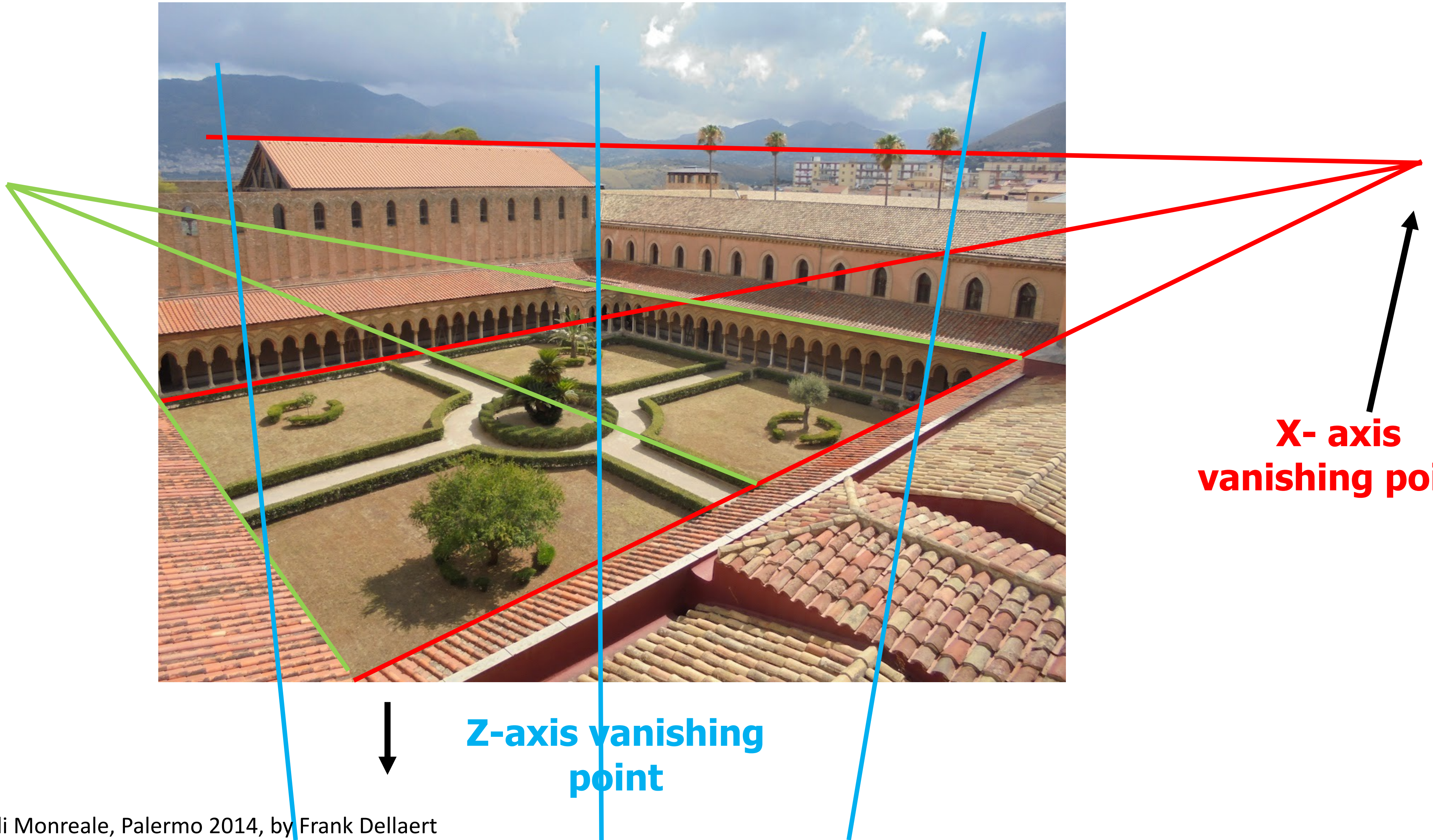
Railroad: parallel lines

Vanishing points and lines



Slide from Efros, Photo from Criminisi

Vanishing points and lines



Cattedrale di Monreale, Palermo 2014, by Frank Dellaert

4. Stereo Vision

- Stereo is used in the HVS
- Very useful in computer vision as well
- Eliminates scale ambiguity

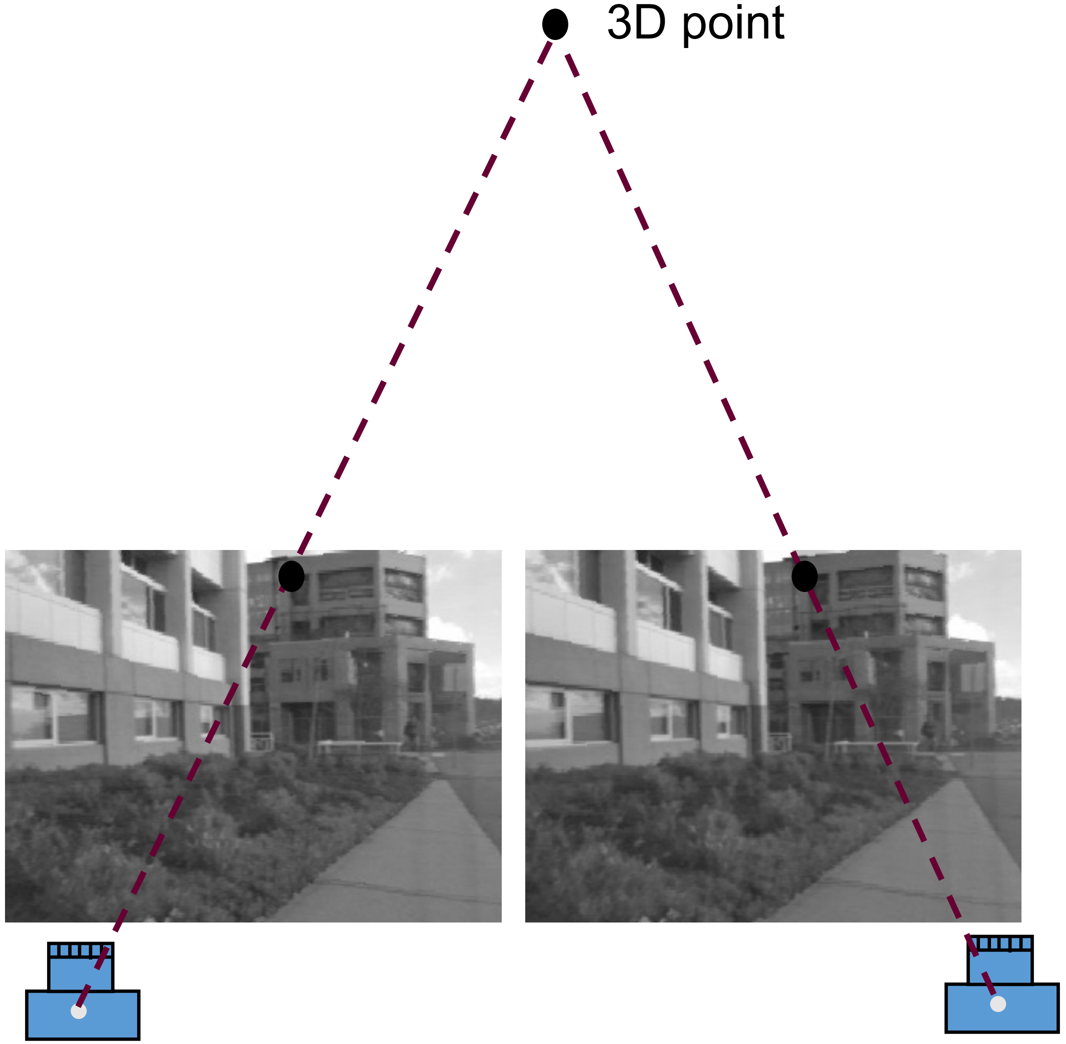
- Many slides adapted from F&P and Sing Bing Kang guest lecture

Etymology

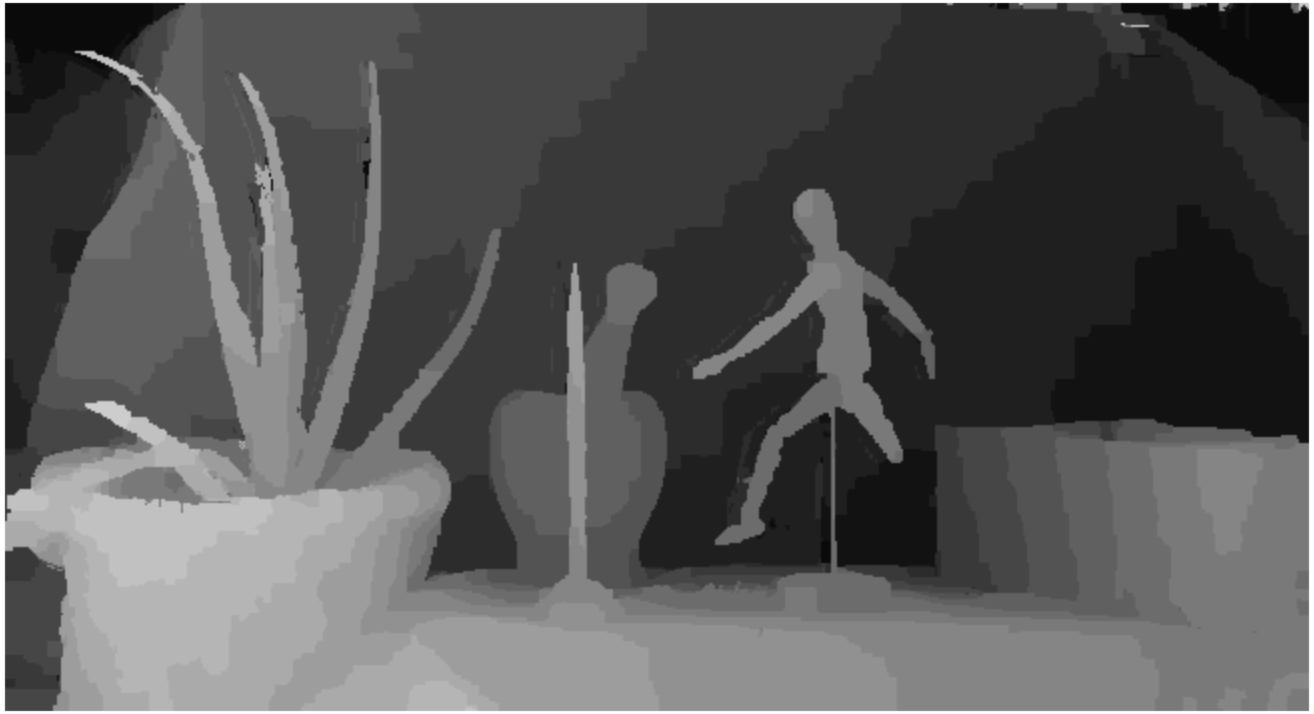
Stereo comes from the Greek word for *solid* (στερεο), and the term can be applied to any system using more than one channel

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



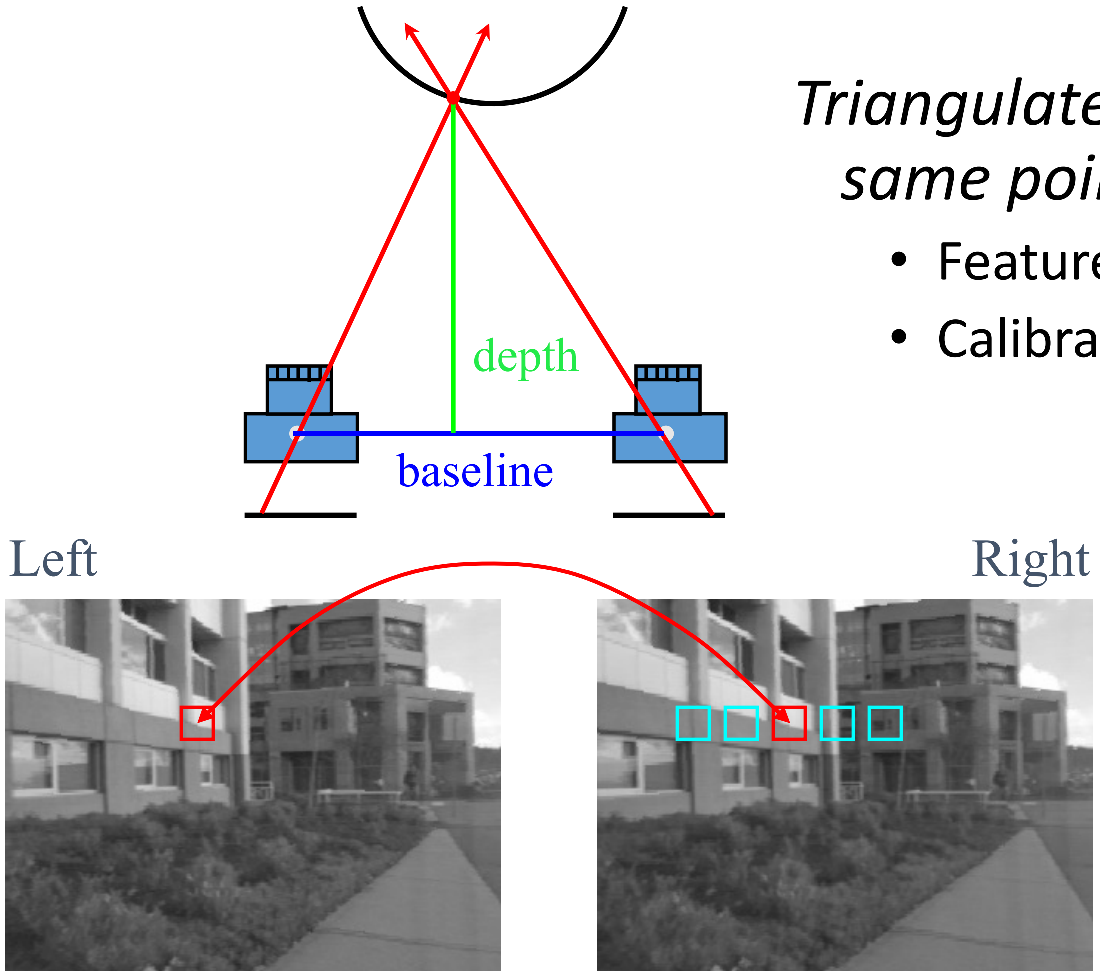
Example



Right Image
Left Image
Disparity

View Interpolation

Basic Idea of Stereo



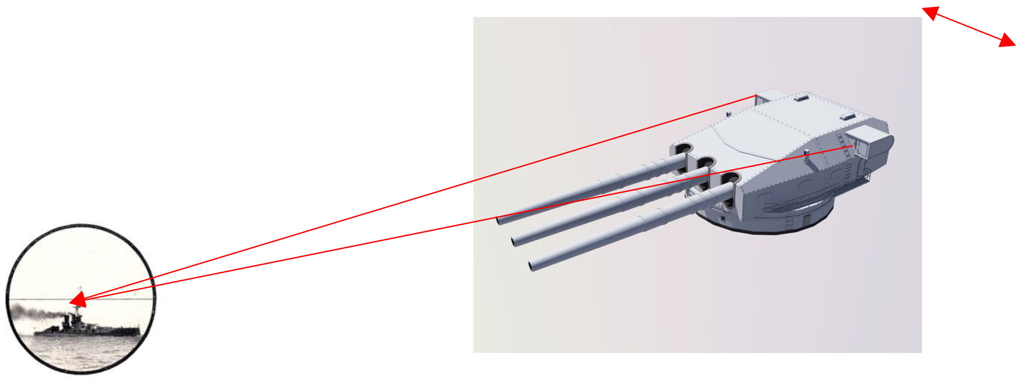
Triangulate on two images of the same point to recover depth.

- 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

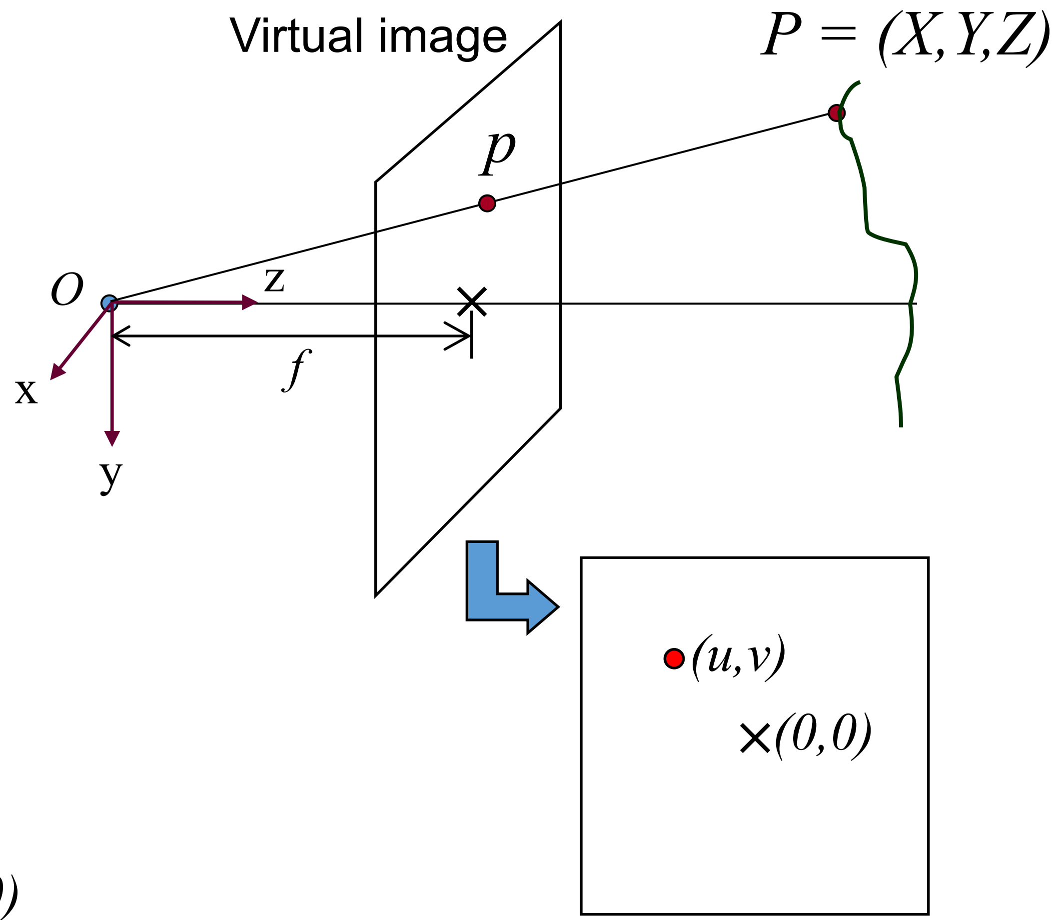


5. Stereo Geometry

- 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

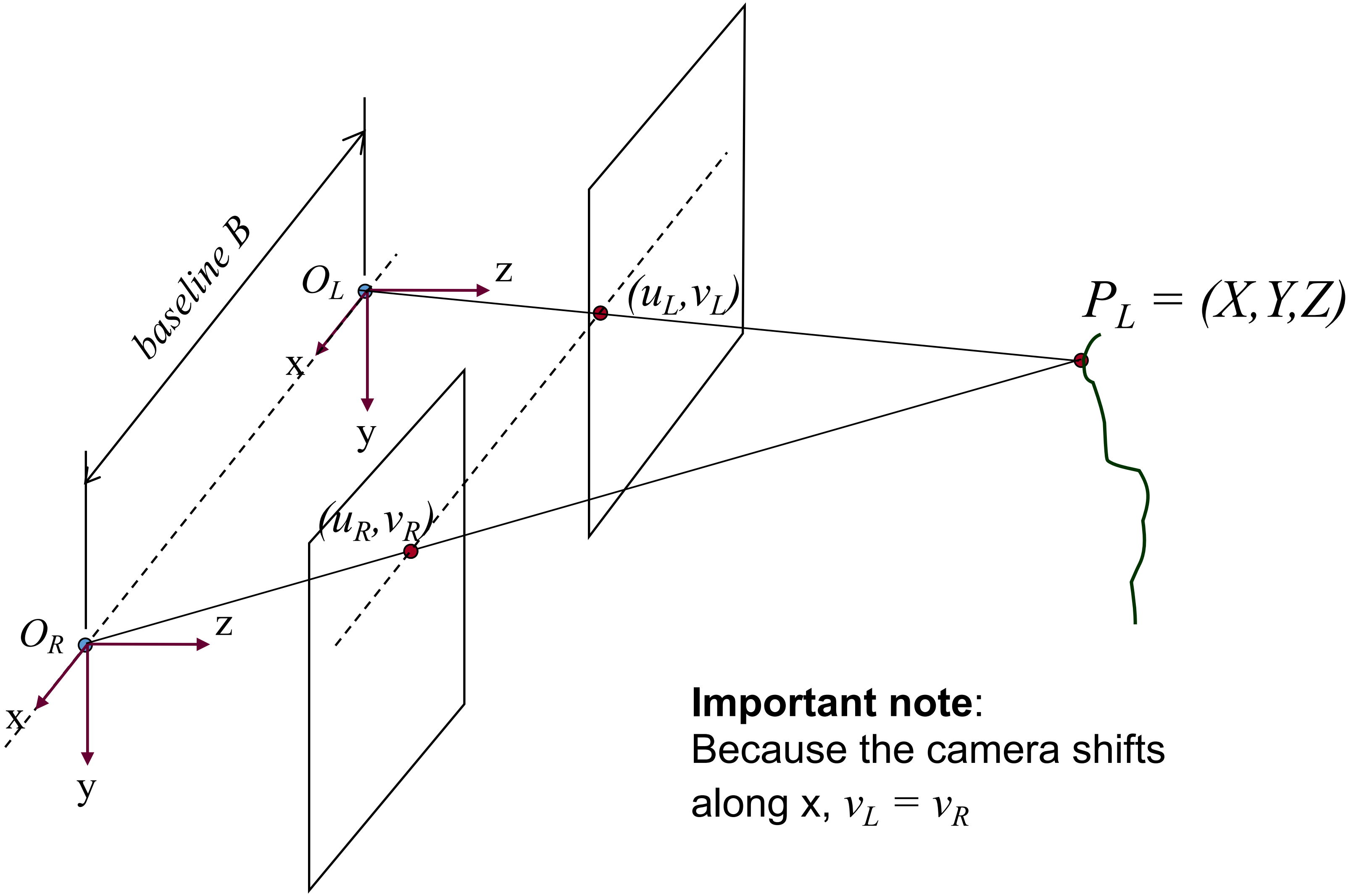


The 2D coordinates in the image are given by

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

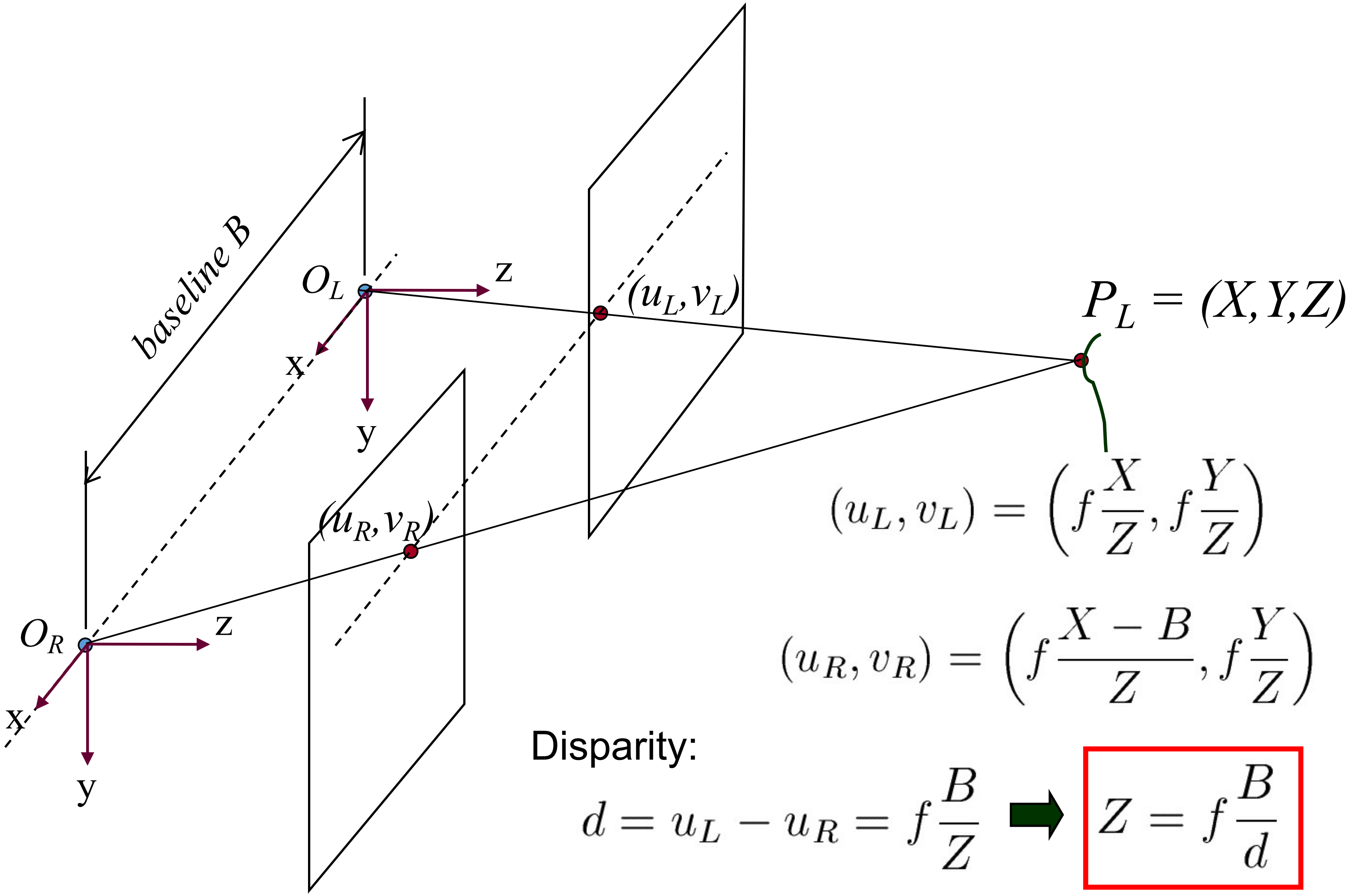
Note: image center is *(0,0)*

Basic Stereo Derivations



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

Basic Stereo Formula



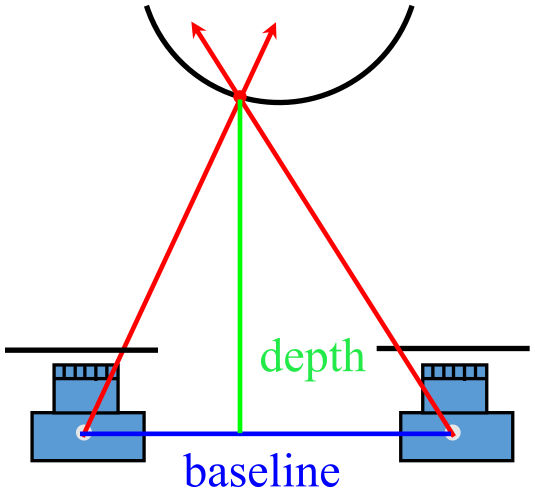
$$(u_L, v_L) = \left(f\frac{X}{Z}, f\frac{Y}{Z}\right)$$

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

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

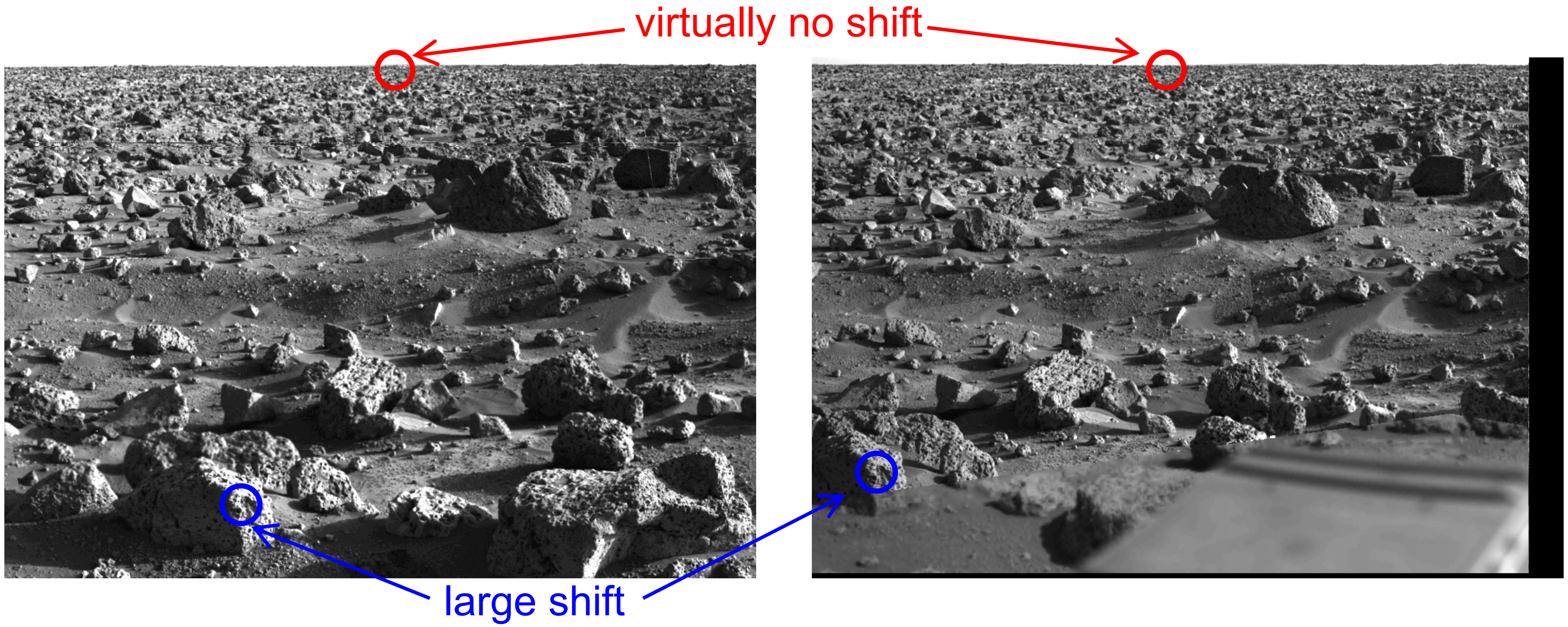
Matching correlation windows across scan lines

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

Stereo Correspondence

- Search over disparity to find correspondences
- Range of disparities can be large

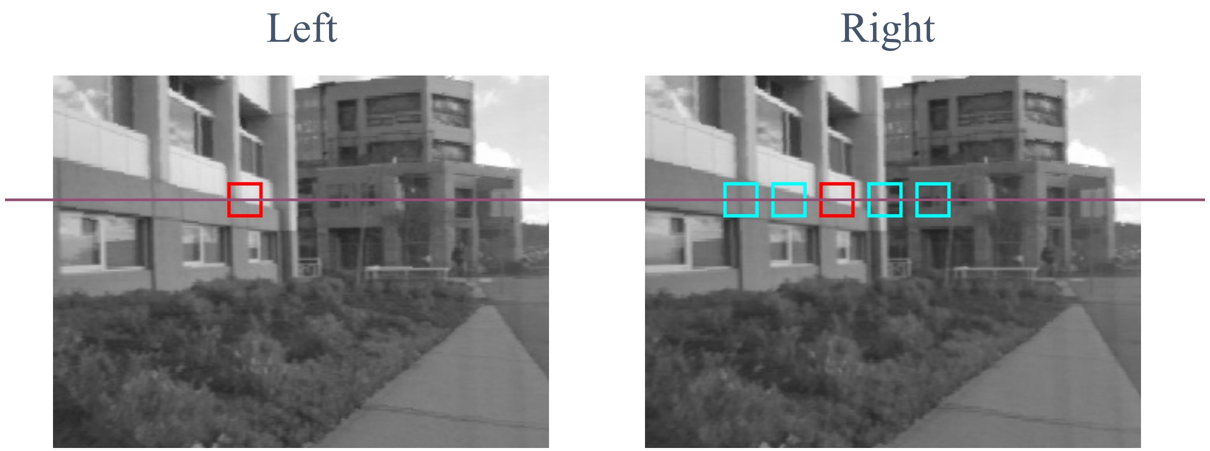


Correspondence Using Window-based Correlation

Matching criterion = Sum-of-squared differences

Aggregation method = Fixed window size

Sum of Squared (Intensity) Differences



w_L and w_R are corresponding m by m windows of pixels.

We define the window function:

$$W_m(x,y) = \{u,v \mid x - \tfrac{m}{2} \le u \le x + \tfrac{m}{2}, y - \tfrac{m}{2} \le v \le y + \tfrac{m}{2}\}$$

The SSD cost measures the intensity difference as a function of disparity:

$$C_r(x,y,d) = \sum_{(u,v)\in W_m(x,y)} [I_L(u,v) - I_R(u-d,v)]^2$$

Correspondence Using Correlation

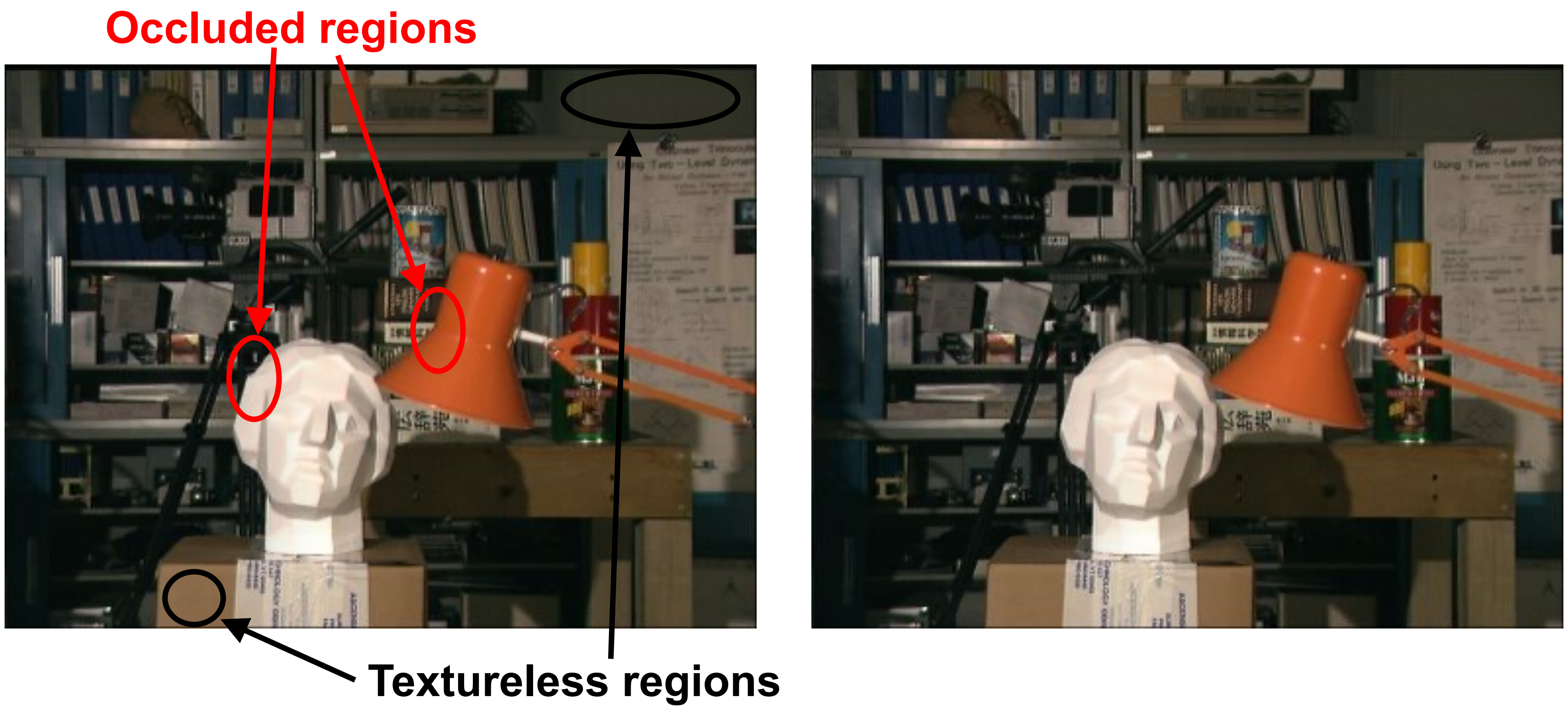
Left

Disparity Map

Images courtesy of Point Grey Research

Two major roadblocks

- Textureless regions create ambiguities
- Occlusions result in missing data

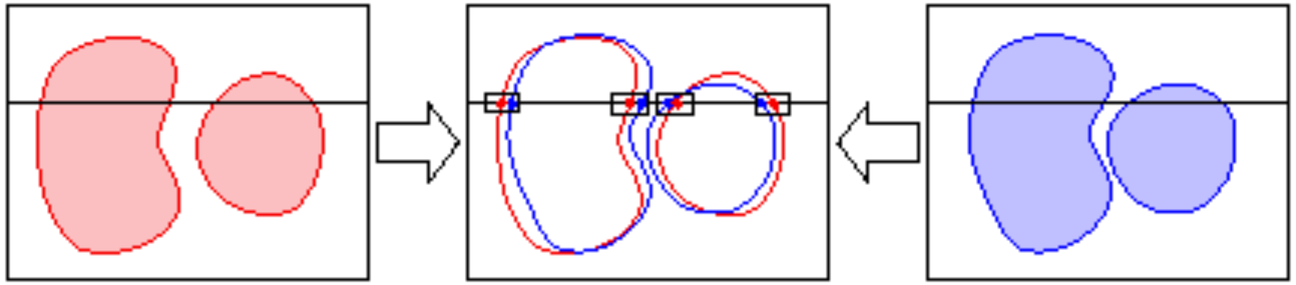


Dealing with ambiguities and occlusion

- 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

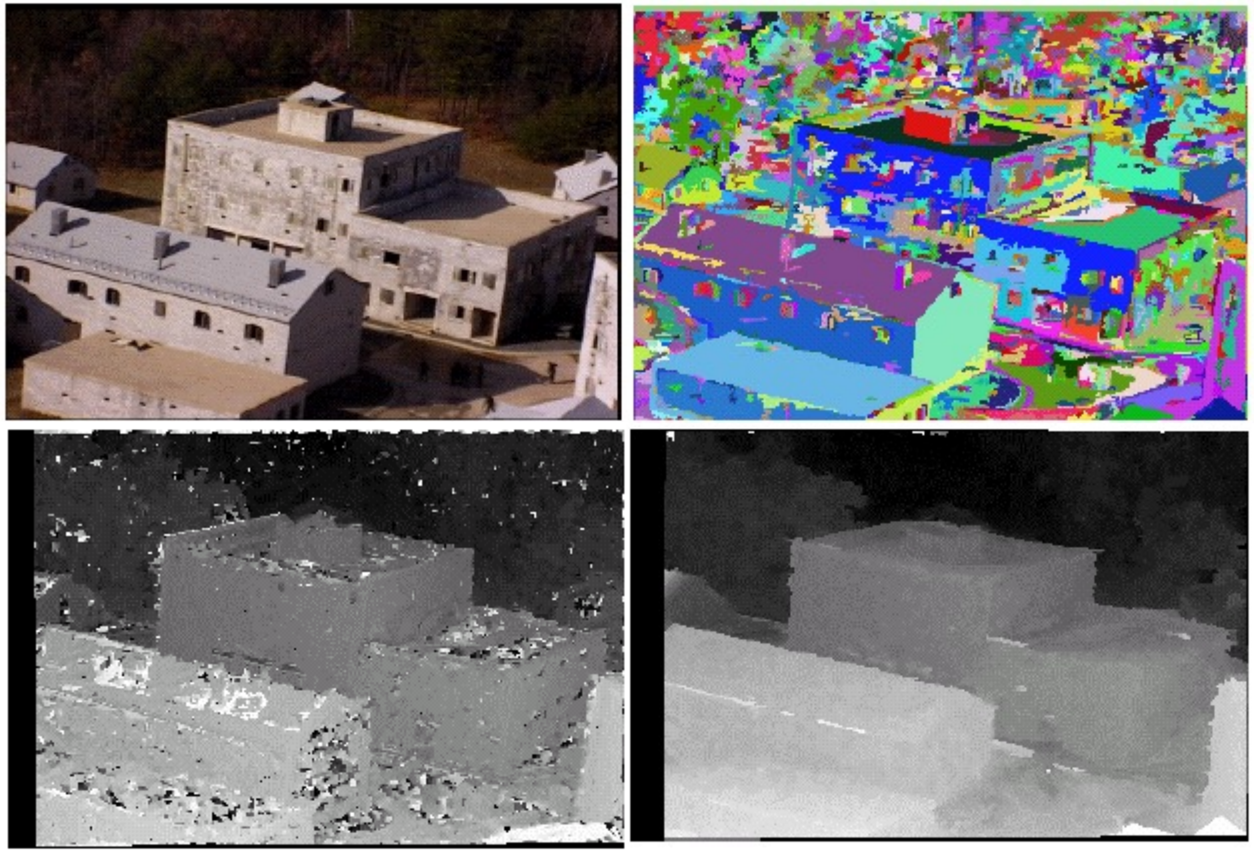
Edge-based Stereo

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



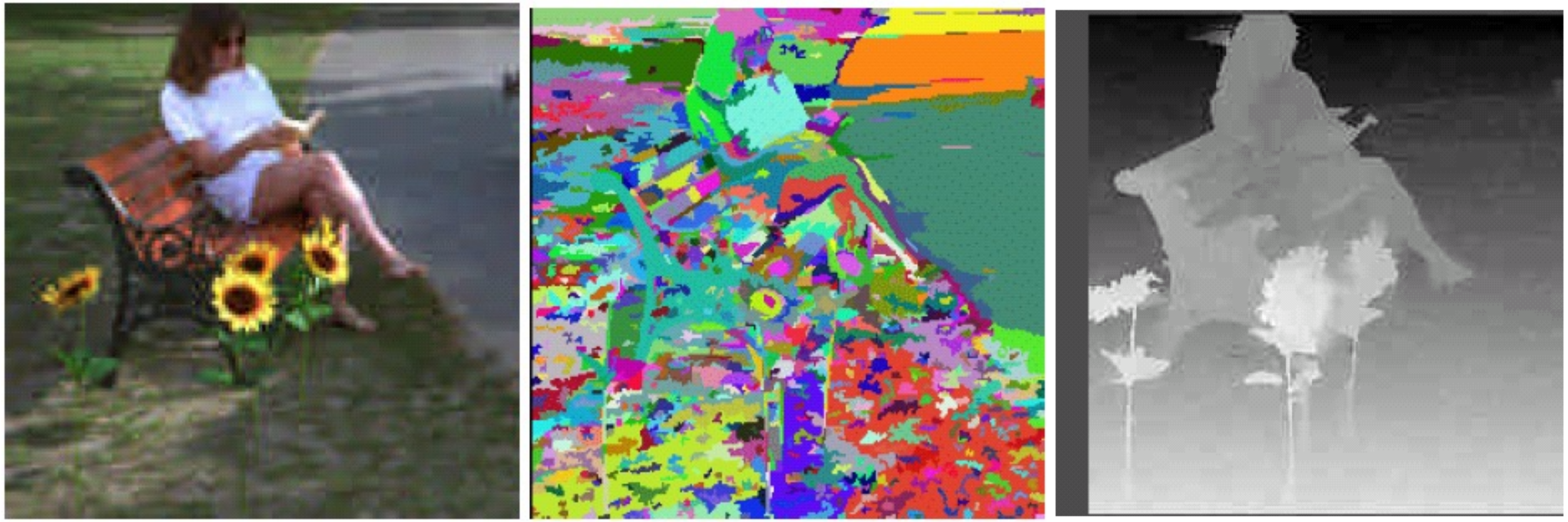
- Which method is better?
  - Edges tend to fail in dense texture (outdoors)
  - Correlation tends to fail in smooth featureless areas
  - Sparse correspondences

Segmentation-based Stereo



Hai Tao and Harpreet W. Sawhney

Another Example



Stereo is Still Unresolved

- Depth discontinuities
- Lack of texture (depth ambiguity)
- Non-rigid effects (highlights, reflection, translucency)

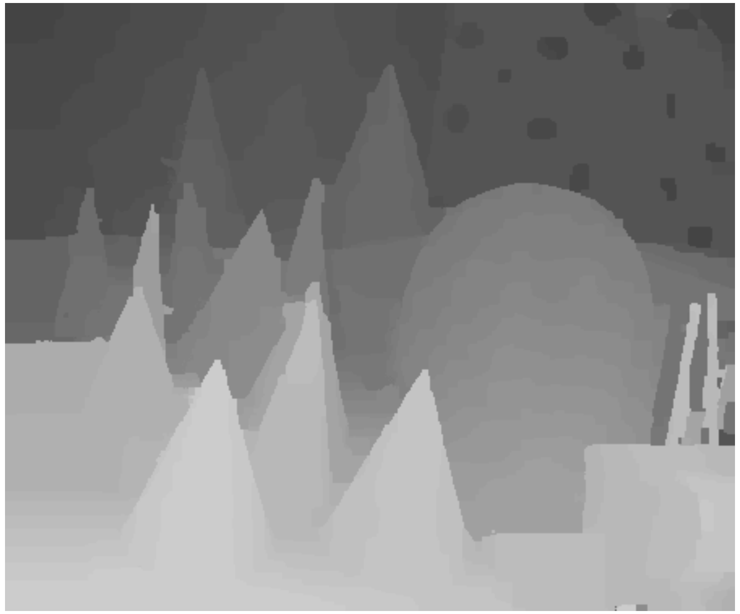
Hallmarks of A Good Stereo Technique

- Should account for occlusions
- Should account for depth discontinuity
- Should have reasonable shape priors to handle textureless regions (e.g., planar or smooth surfaces)

- Advanced: account for non-Lambertian surfaces

Left

Right



Disparity Map

Result of using a more sophisticated stereo algorithm

View Interpolation

Summary

1. Perspective Cameras Intro
2. Pinhole Camera Model defined
3. Properties of Projective Geometry
4. Stereo Vision can recover metric structure
5. Stereo Geometry is simply $Z = f\,B/d$
6. Amazing Stereo Algorithms were elusive, BUT, deep learning!!!!