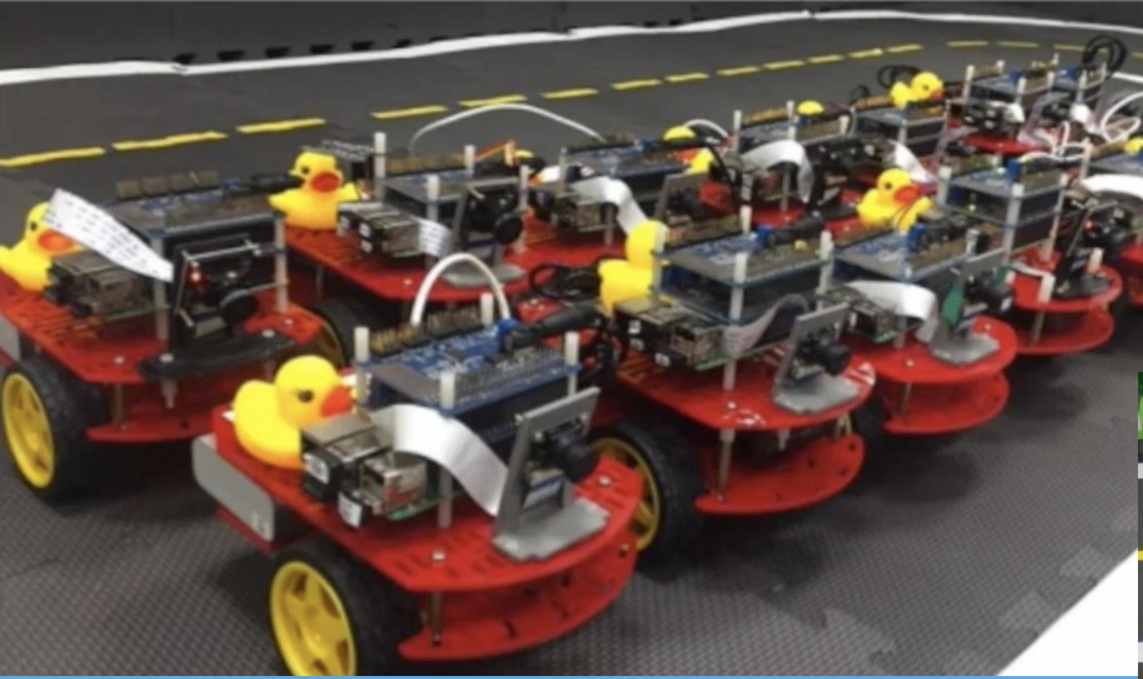
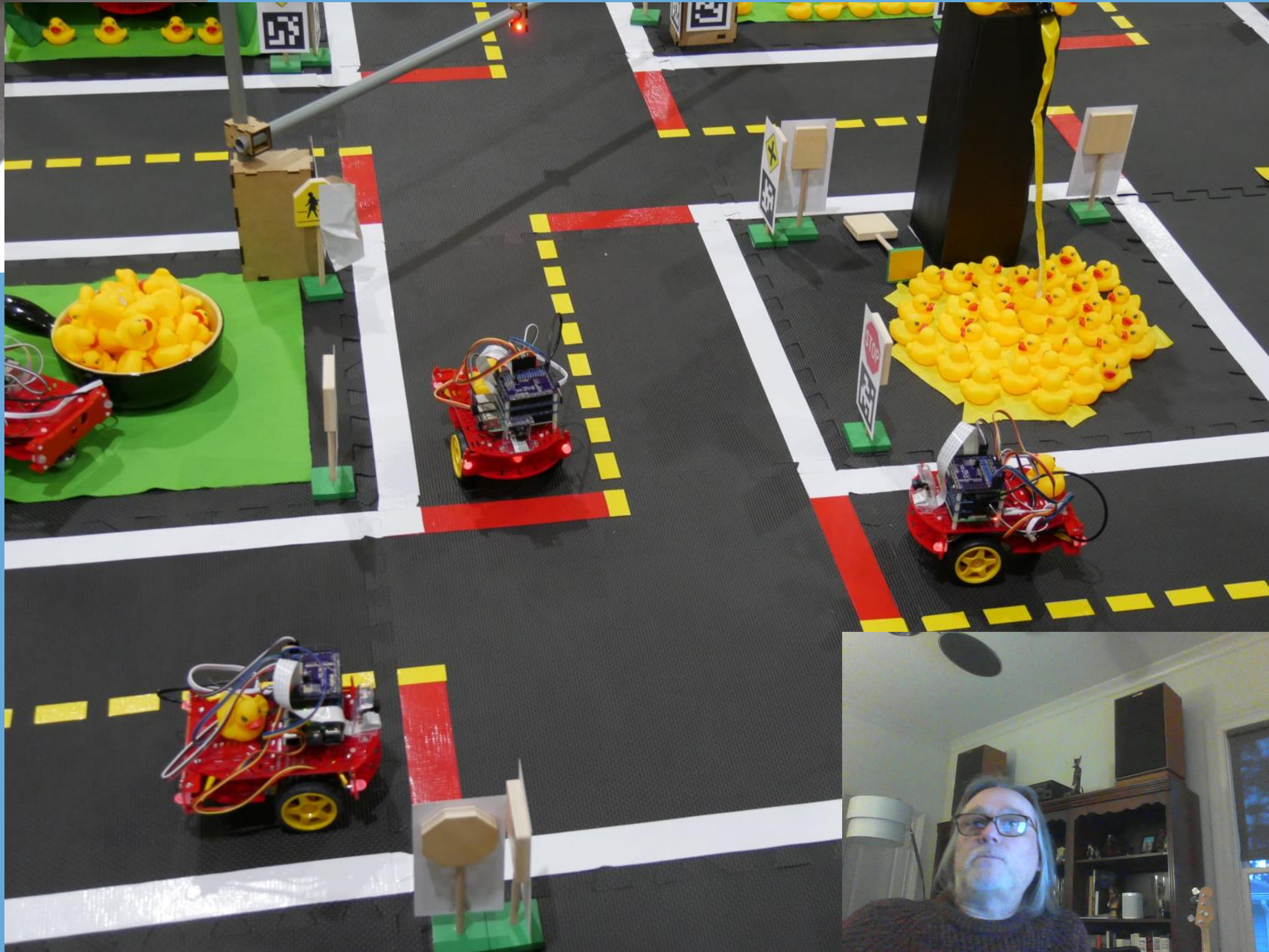


CS 3630!



Lecture 17: Computer Vision Fundamentals

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 James Hays, Irfan Essa, Sing Bing Kang and others.

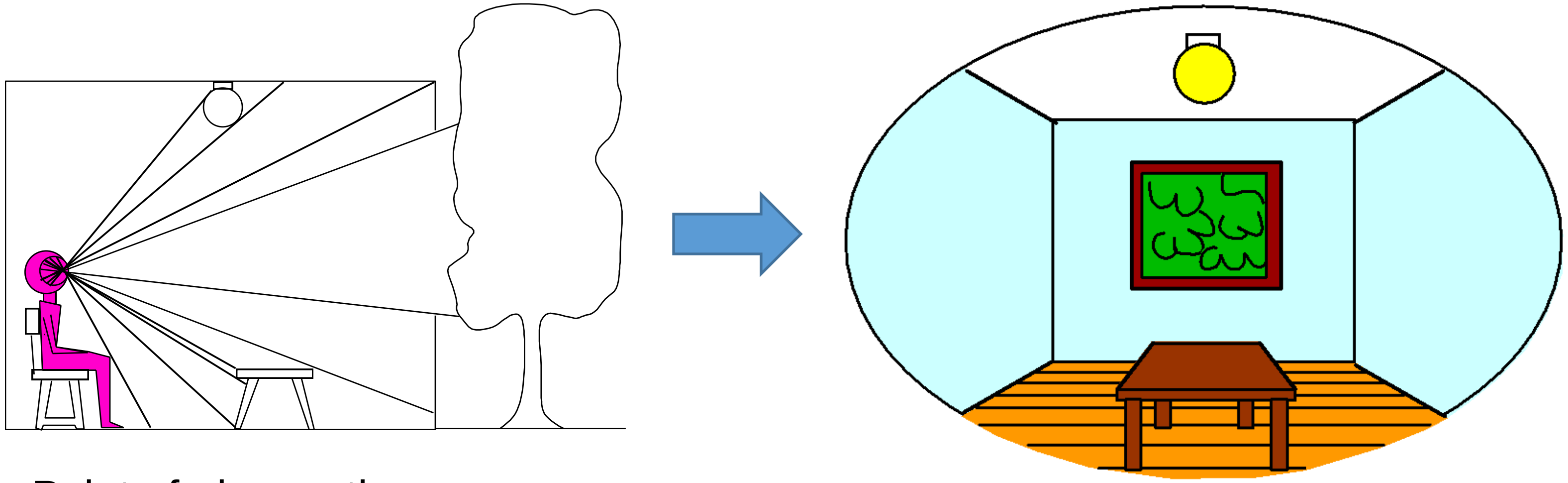
Motivation

- 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

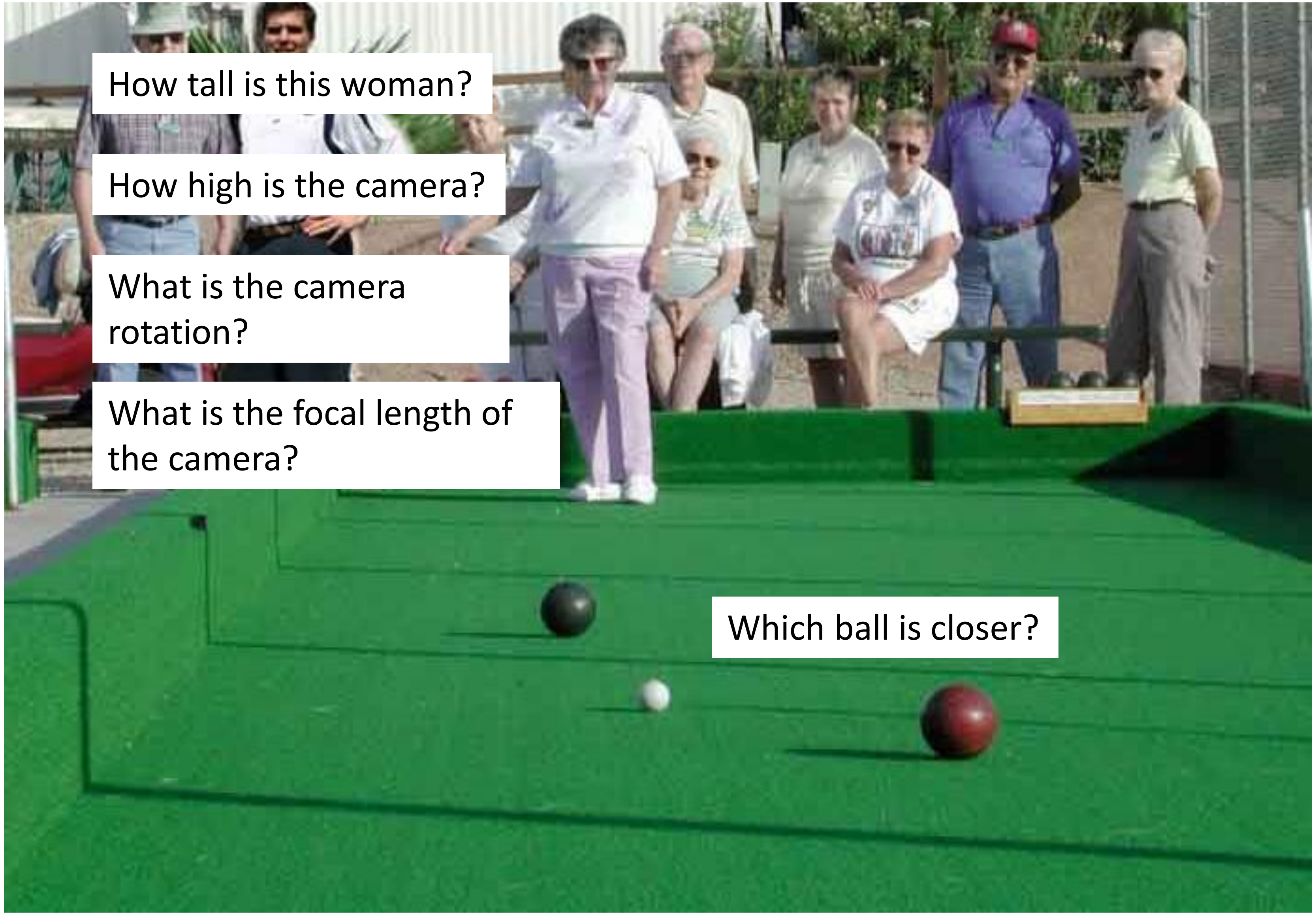
2D image



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



How tall is this woman?

How high is the camera?

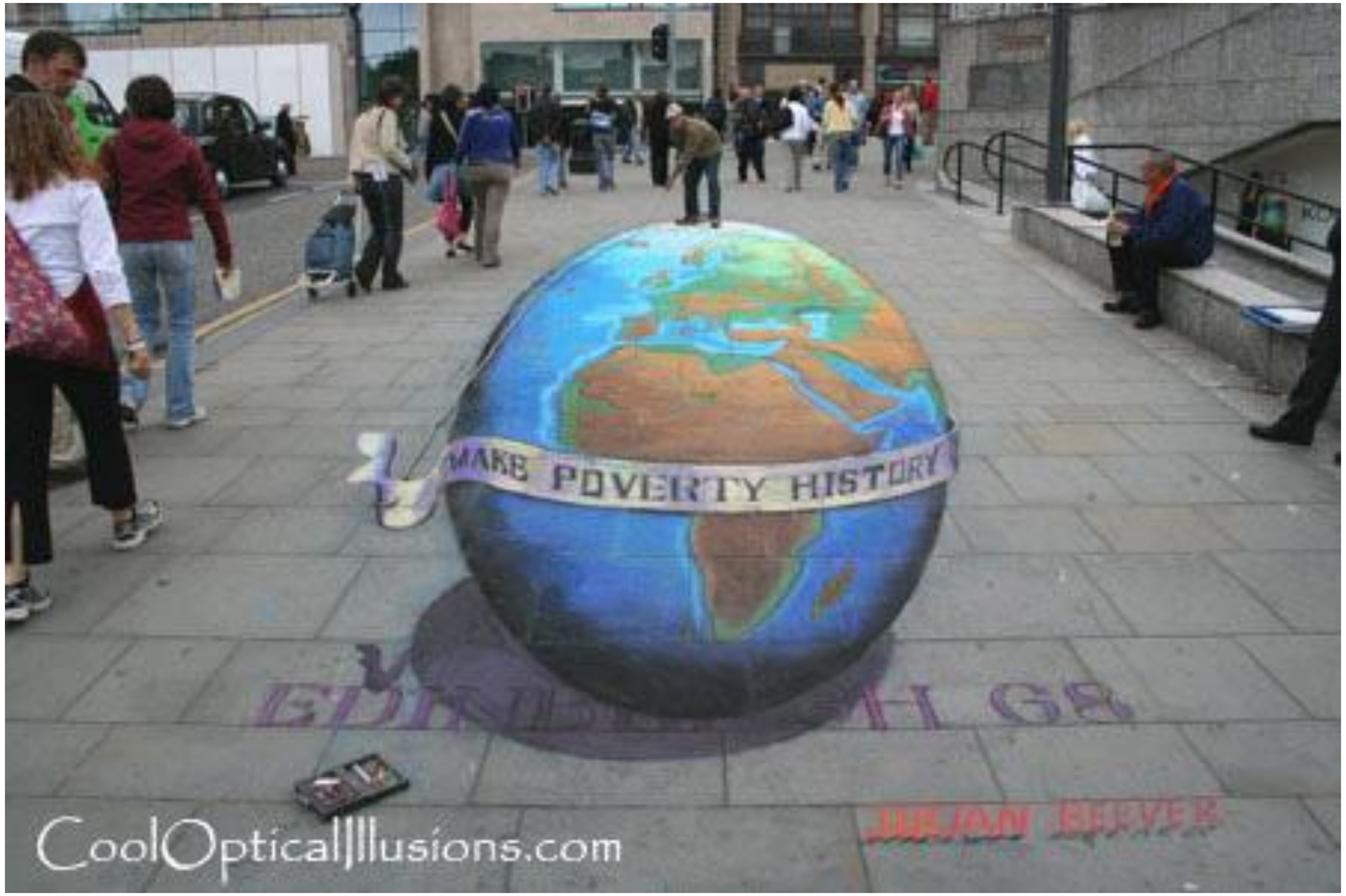
What is the camera rotation?

What is the focal length of the camera?

Which ball is closer?

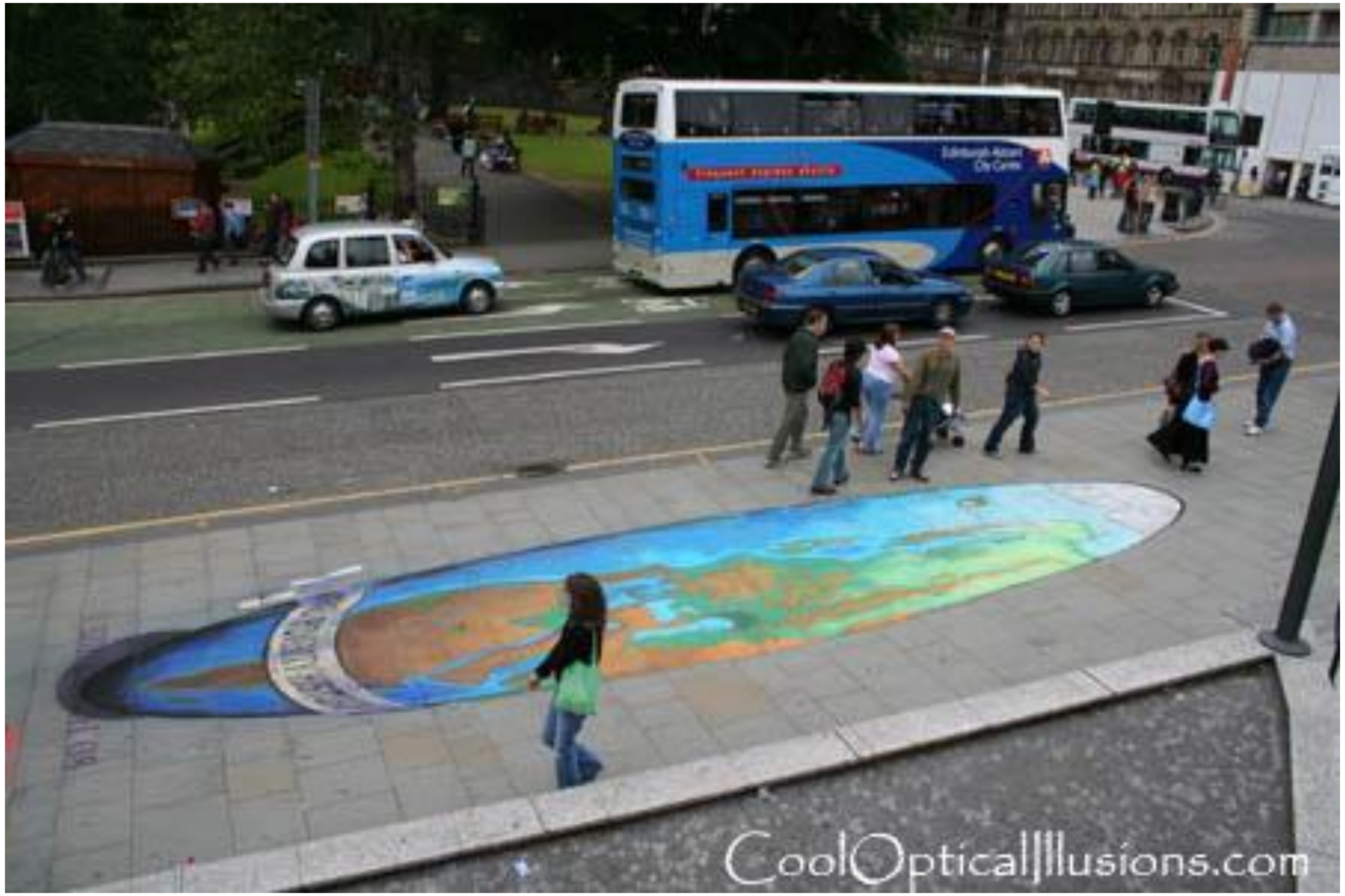
Slide source: Seitz

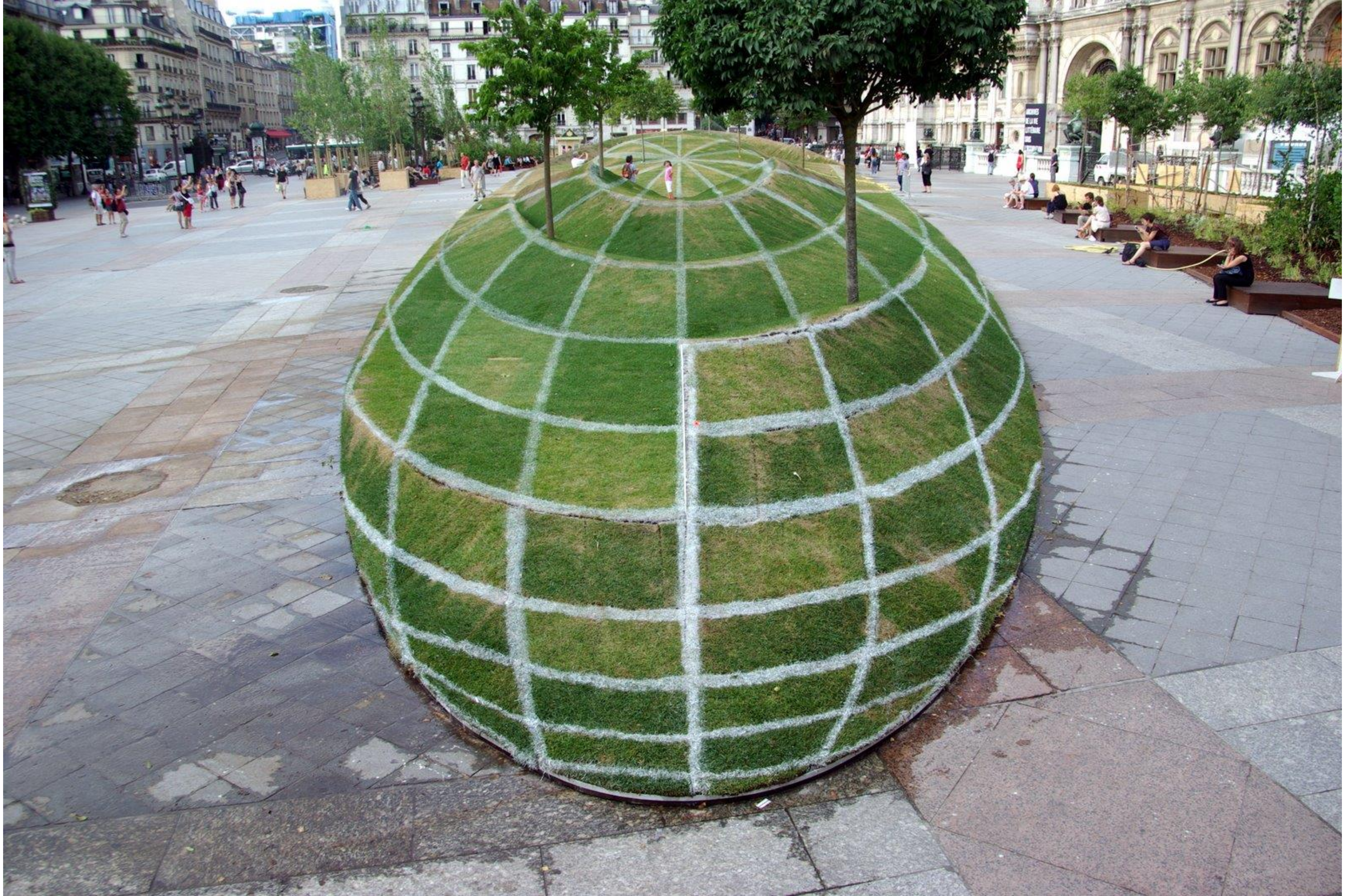
Projection can be tricky…

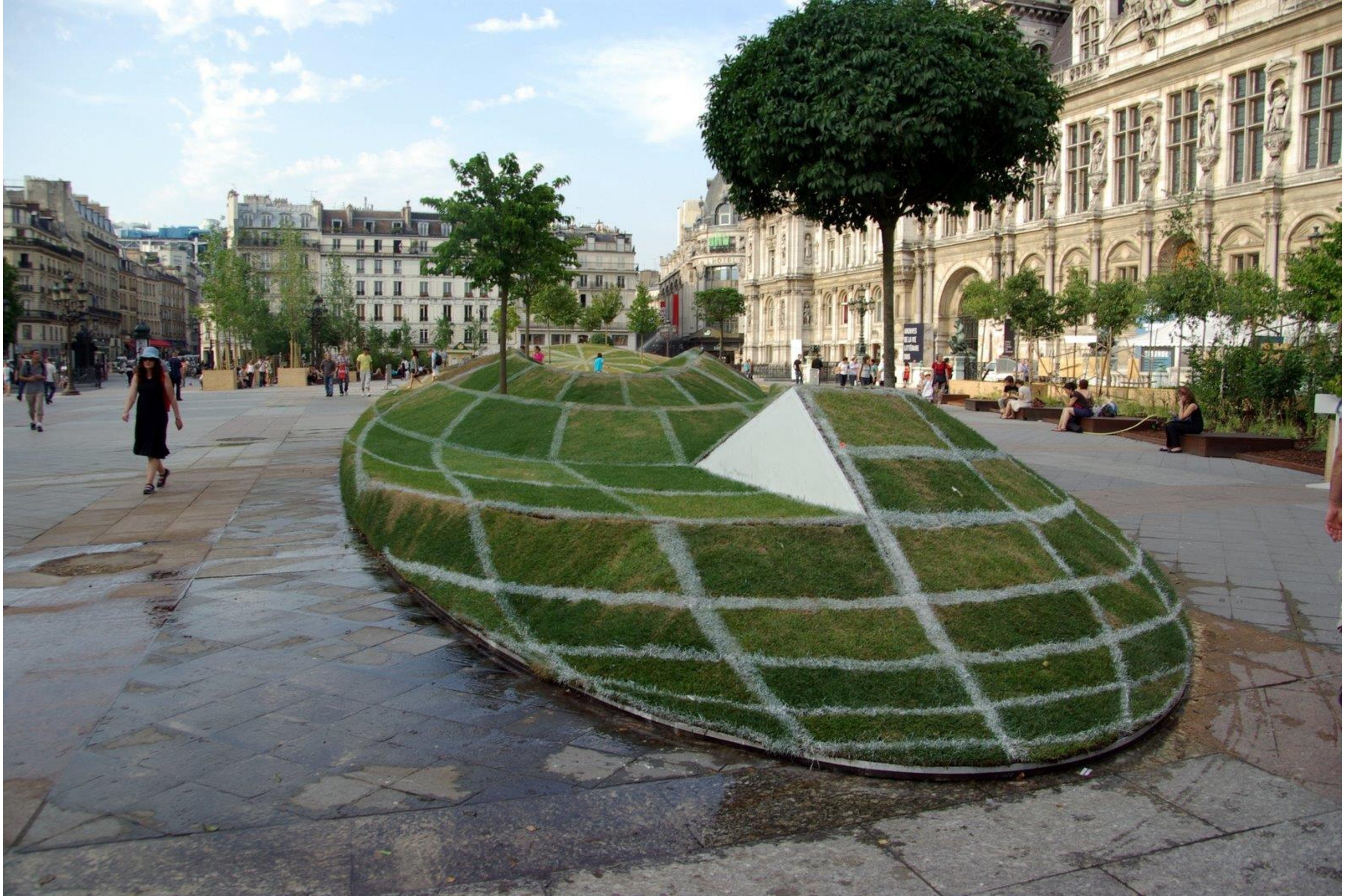


Slide source: Seitz

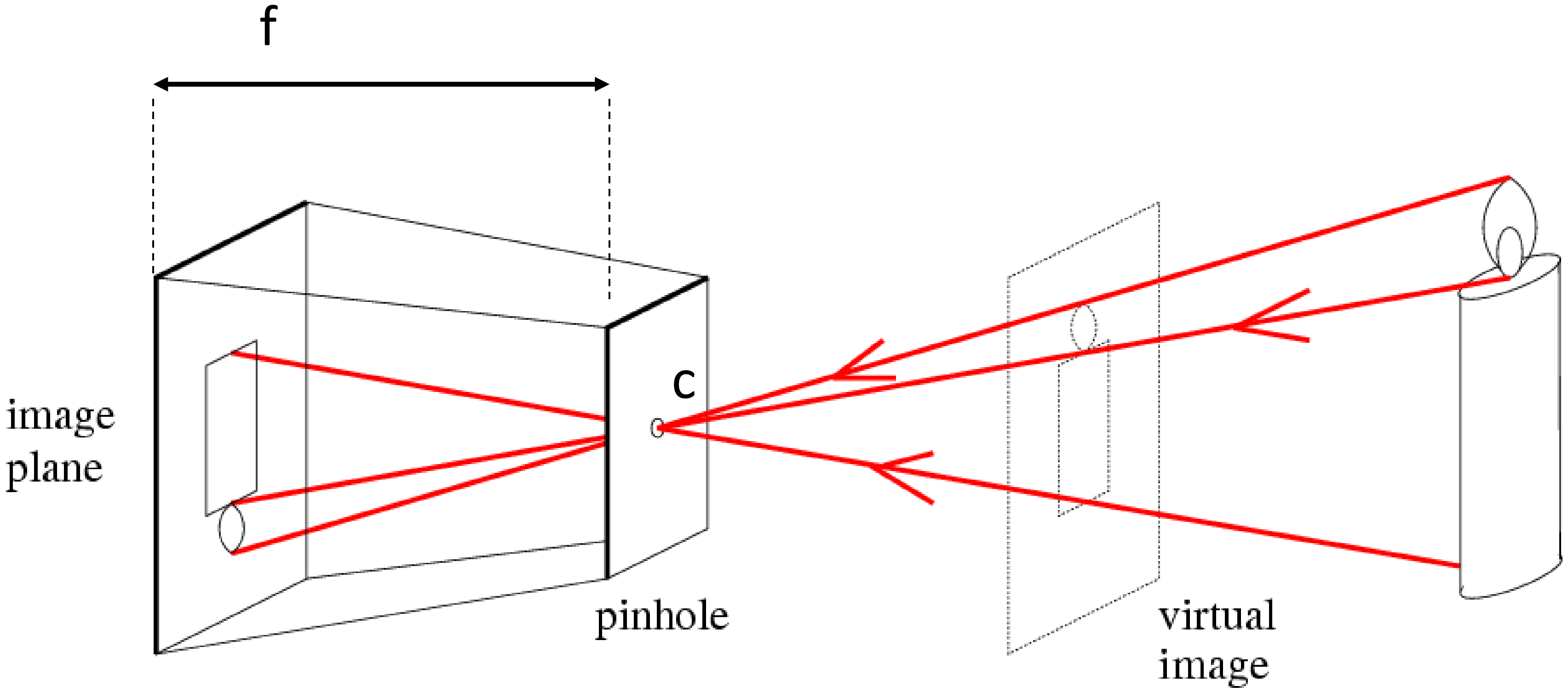
Projection can be tricky…







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)

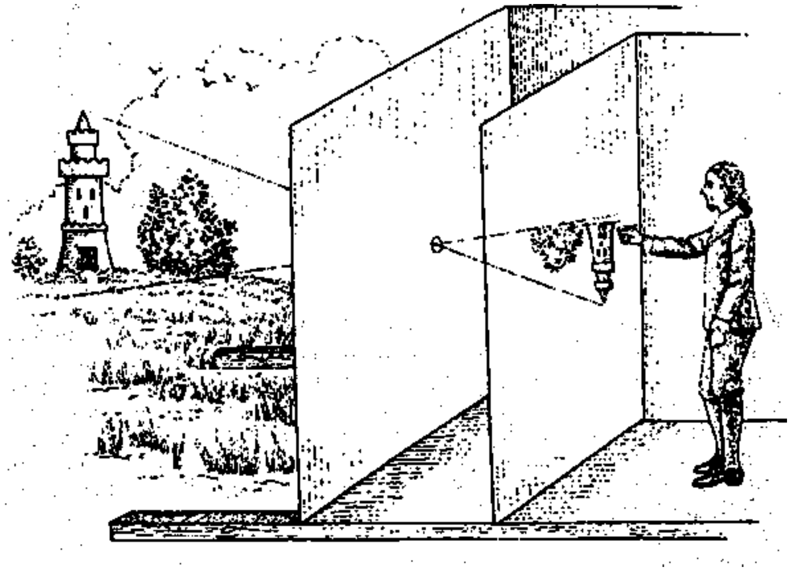
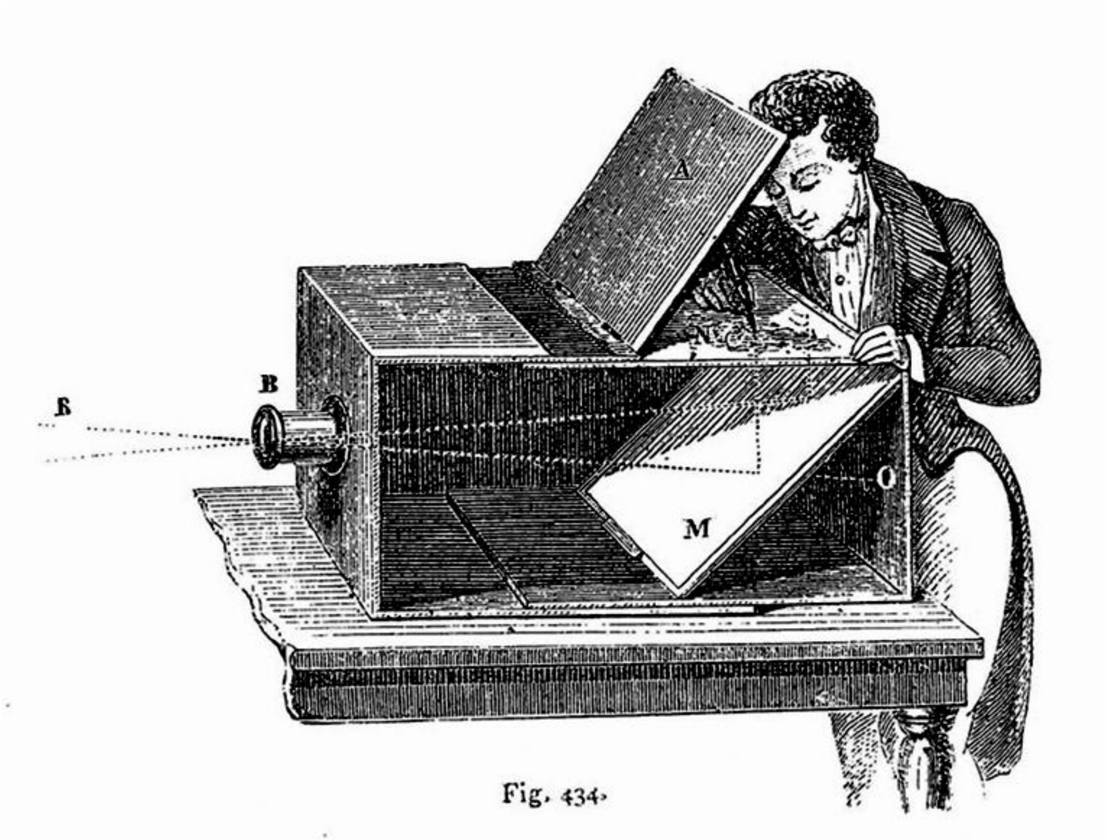


Illustration of Camera Obscura

Freestanding camera obscura at UNC Chapel Hill

Photo by Seth Ilys

Camera Obscura used for Tracing

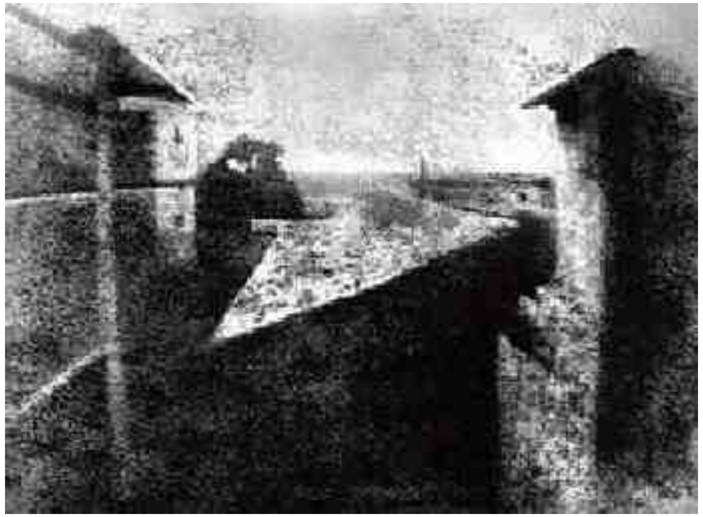


Lens Based Camera Obscura, 1568

First Photograph

Oldest surviving photograph

- Took 8 hours on pewter plate



Joseph Niepce, 1826

Photograph of the first photograph

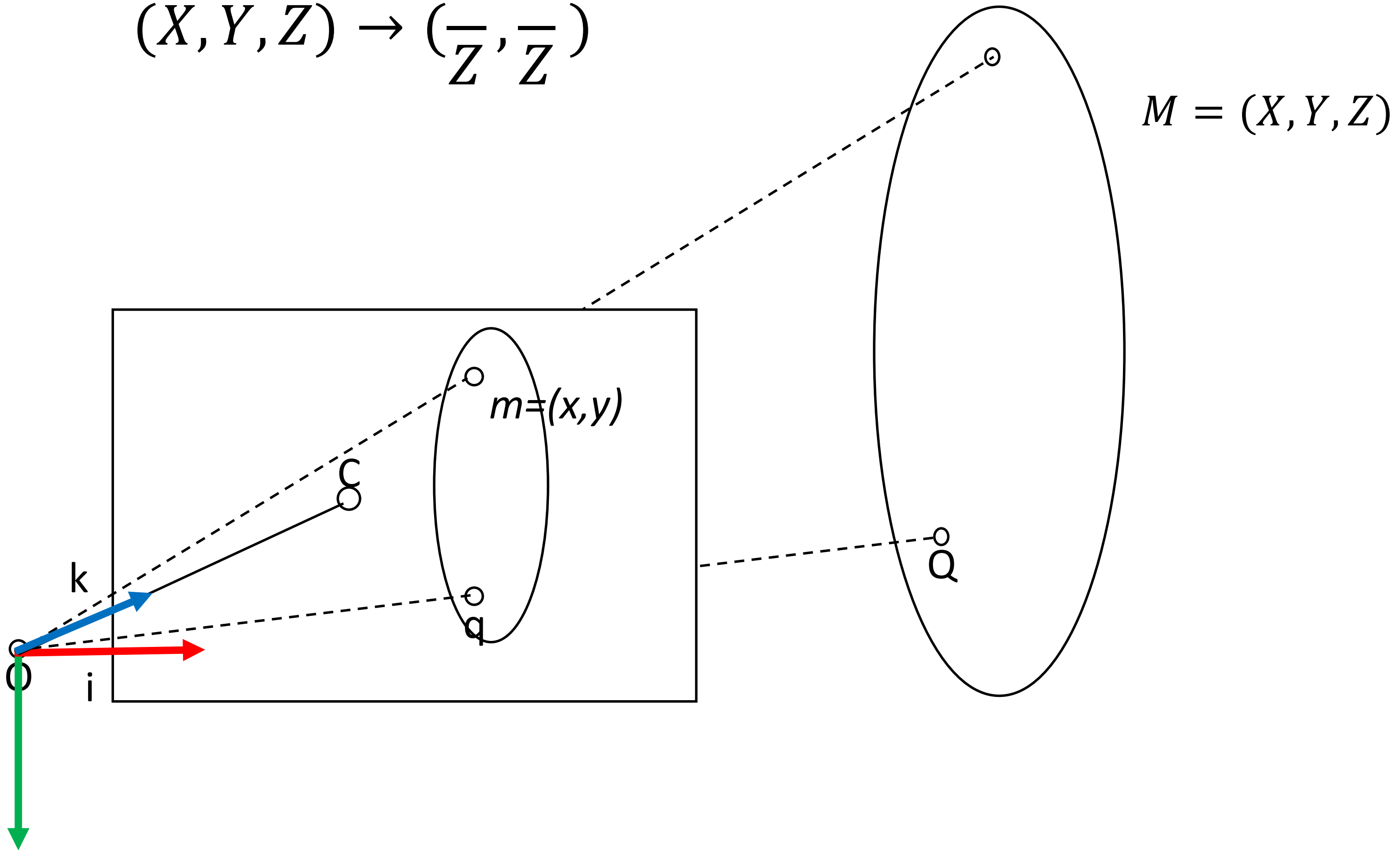
Stored at UT Austin

Niepce later teamed up with Daguerre, who eventually created Daguerrotypes

Pinhole Camera

- Fundamental equation:

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



Homogeneous Coordinates

Linear transformation of homogeneous (projective) coordinates

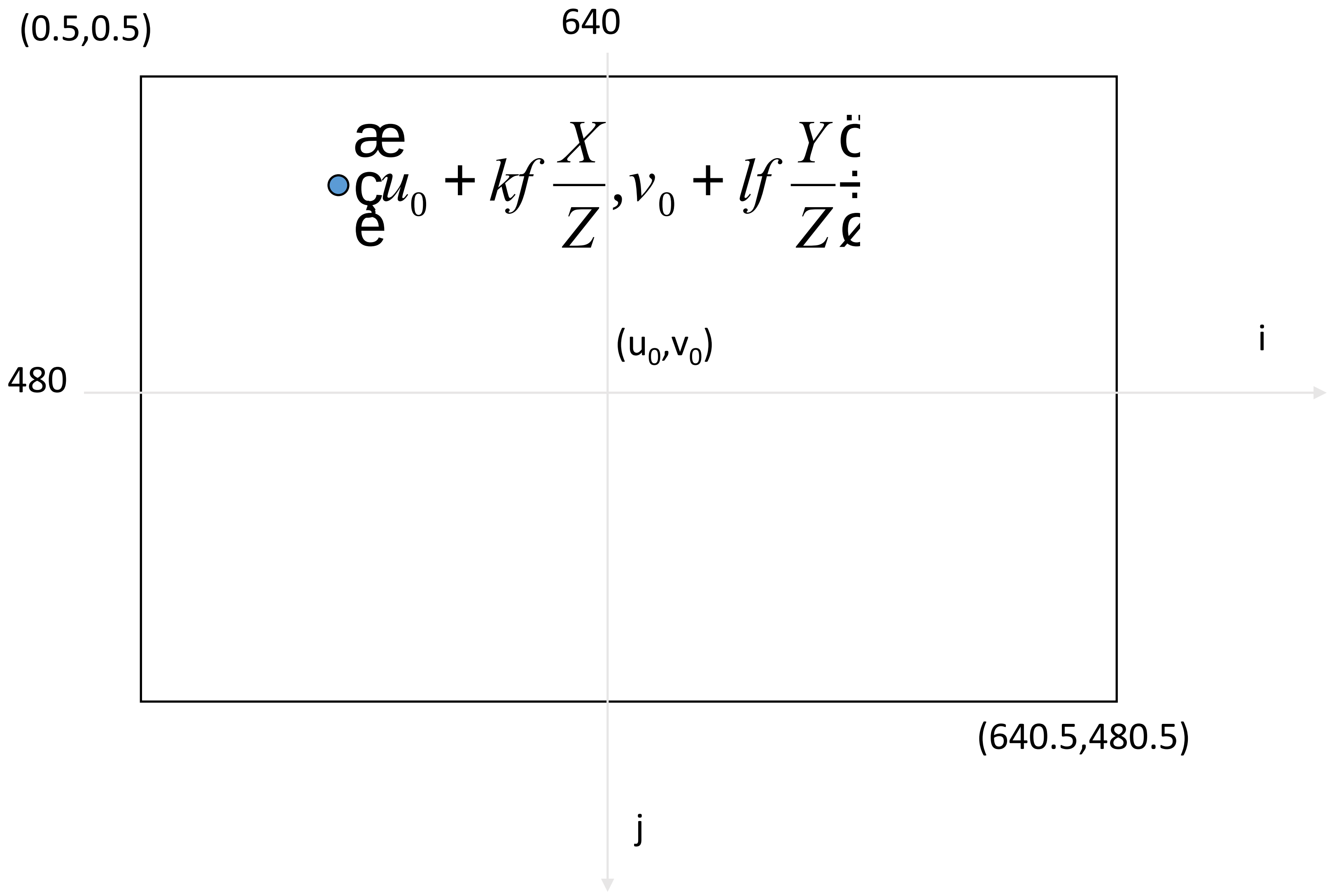
$$m = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} I & 0 \end{bmatrix} M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ T \end{bmatrix}$$

Recover image (Euclidean) coordinates by normalizing:

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

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

Pixel coordinates in 2D



$$\left(u_0 + kf\frac{X}{Z}, v_0 + lf\frac{Y}{Z}\right)$$

Intrinsic Calibration

3×3 Calibration Matrix K

$$m = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = K[I \quad 0]M = \begin{bmatrix} a & s & u_0 \\ & b & v_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ T \end{bmatrix}$$

skew

Recover image (Euclidean) coordinates by normalizing :

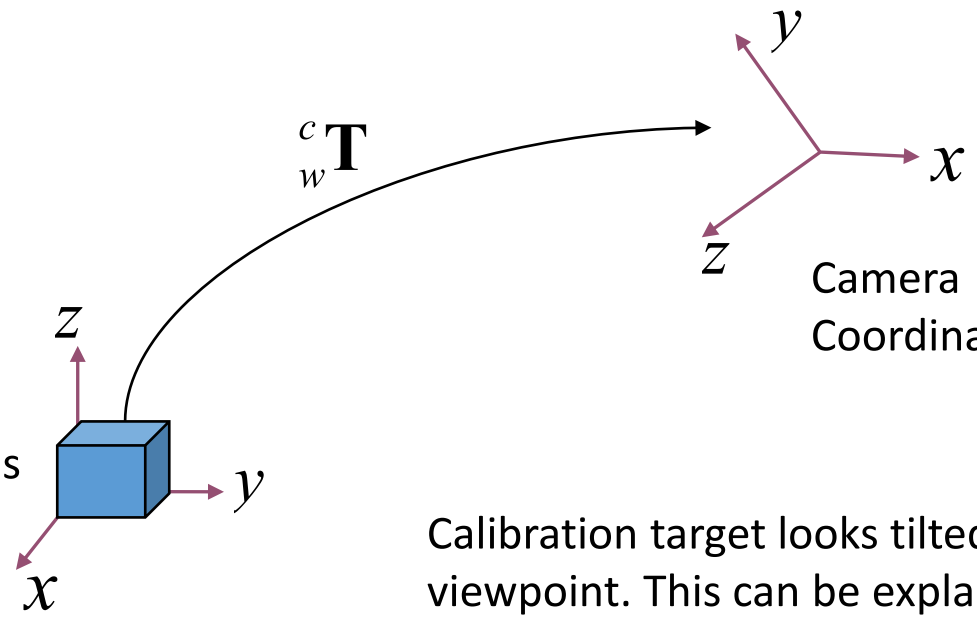
$$x = \frac{u}{w} = \frac{aX + sY + u_0}{Z}$$

$$y = \frac{v}{w} = \frac{bY + v_0}{Z}$$

5 Degrees of Freedom !

Camera Pose

In order to apply the camera model, objects in the scene must be expressed in *camera coordinates*.



Camera Coordinates

World Coordinates

Calibration target looks tilted from camera viewpoint. This can be explained as a difference in coordinate systems.

Projective Camera Matrix

$$Camera = Calibration \times \Pr ojection \times Extrinsics$$

$$m = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \alpha & s & u_0 \\ & \beta & v_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ T \end{bmatrix}$$

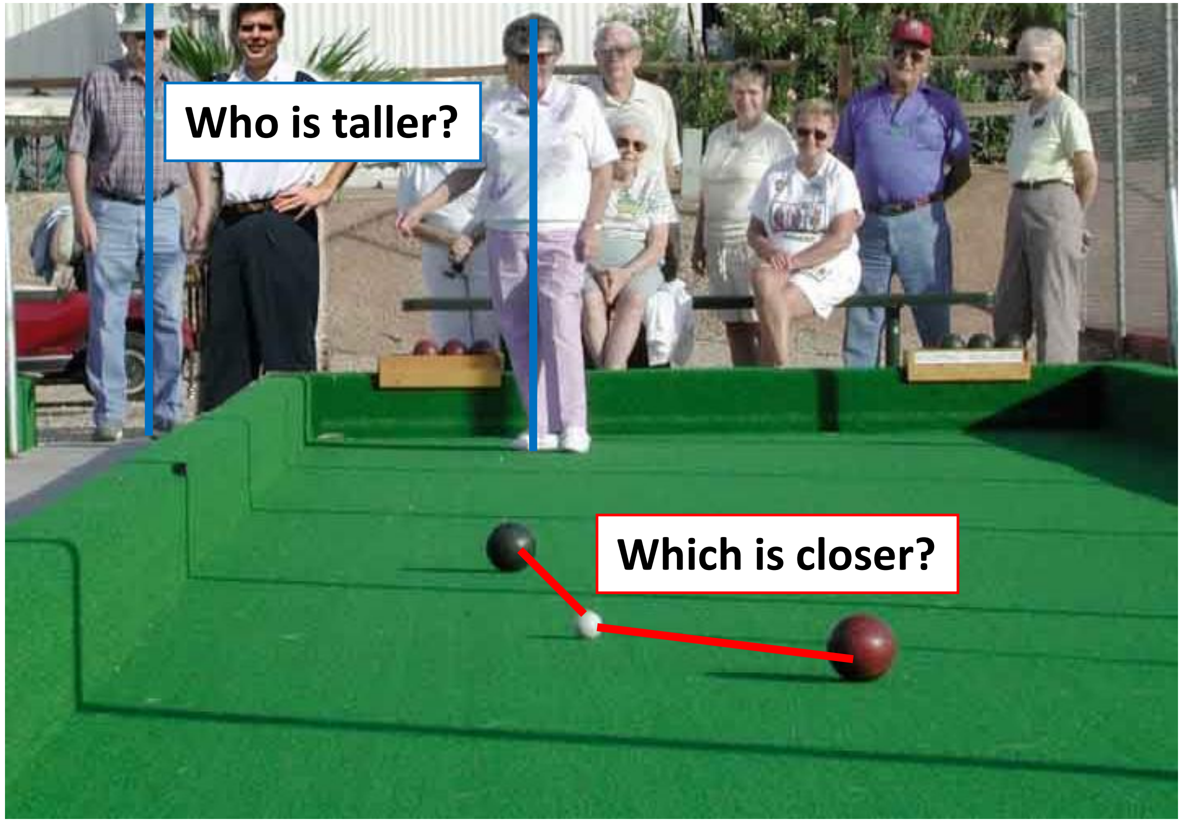
$$= K\begin{bmatrix} R & t \end{bmatrix} M = PM$$

5+6 Degrees of Freedom (DOF) = 11 !

3. Properties of projective Geometry

What is lost?

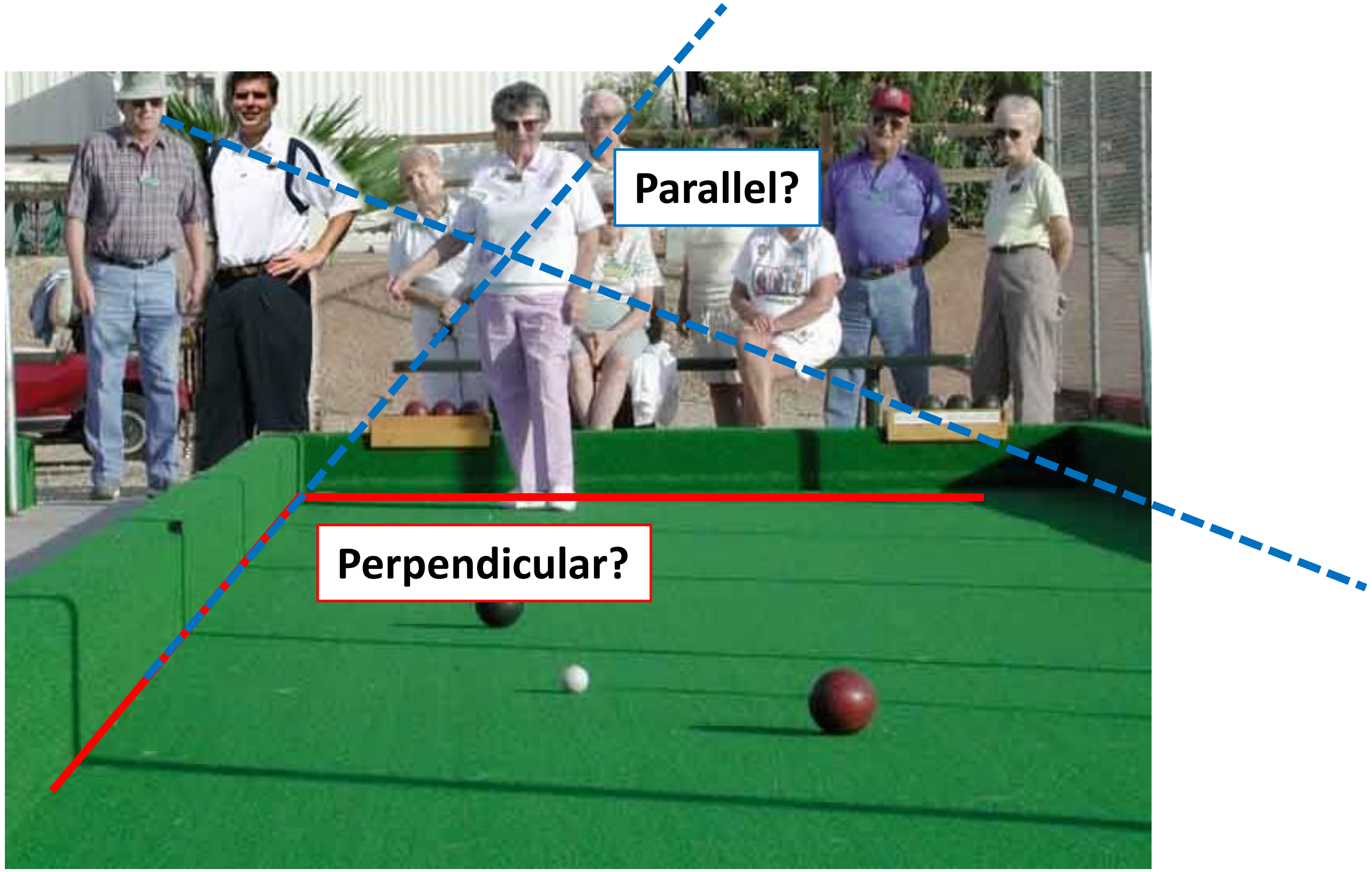
- Length



Properties of projective Geometry

What is lost?

- Length
- Angles



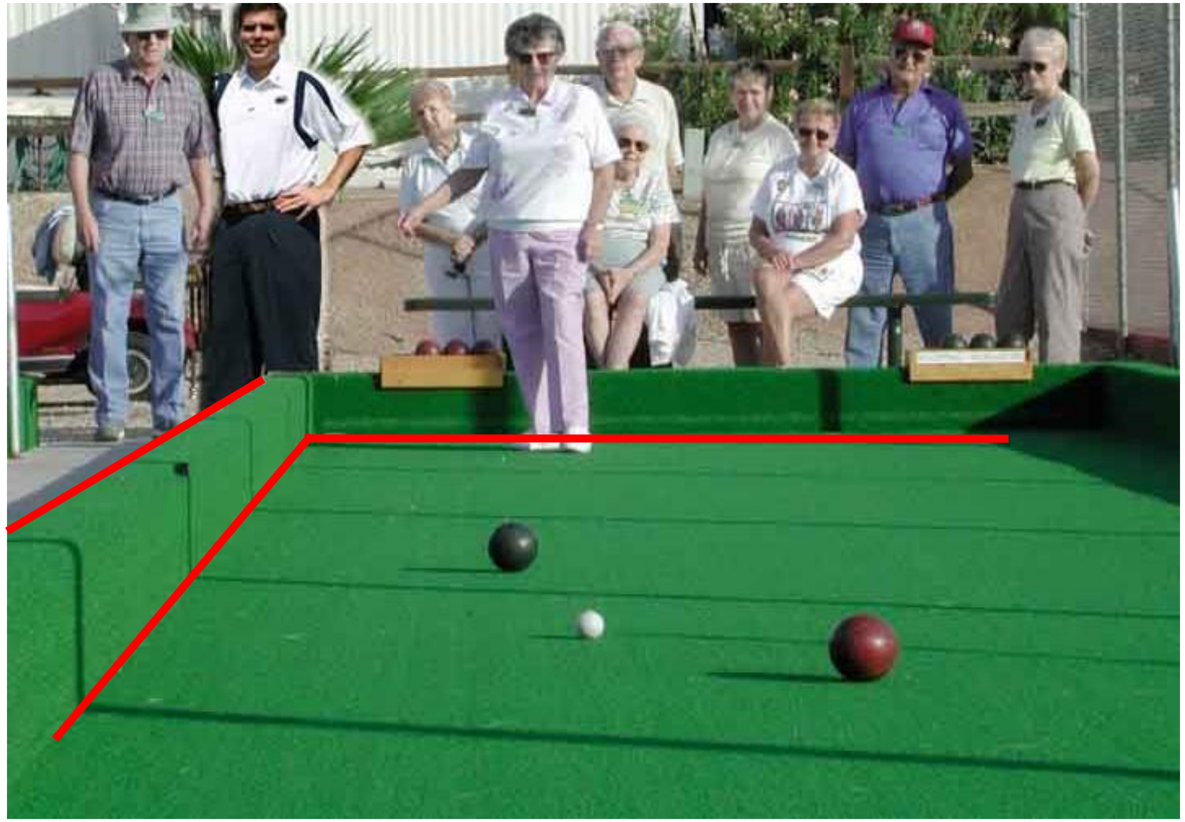
Parallel?

Perpendicular?

Properties of projective Geometry

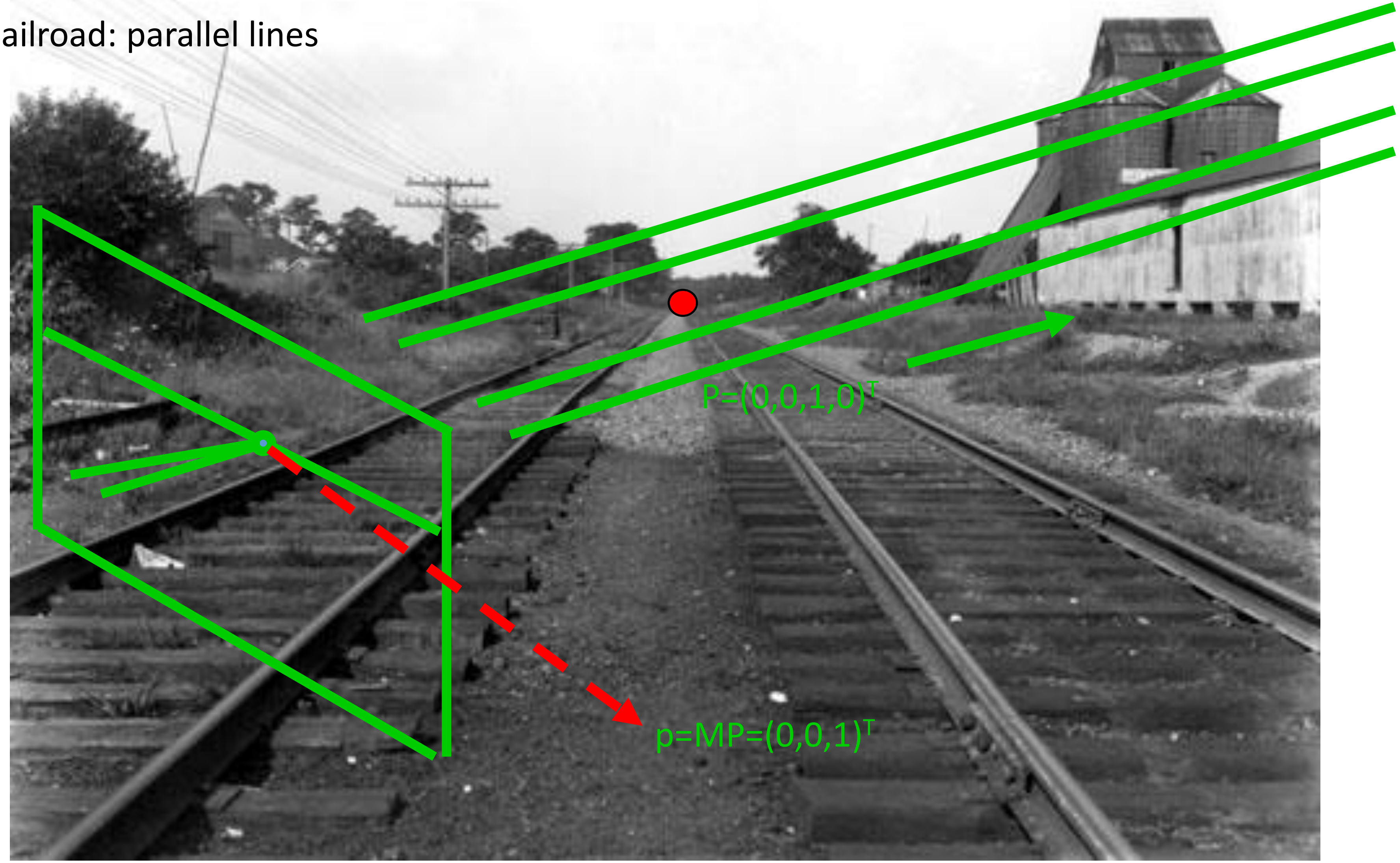
What is preserved?

- Straight lines are still straight

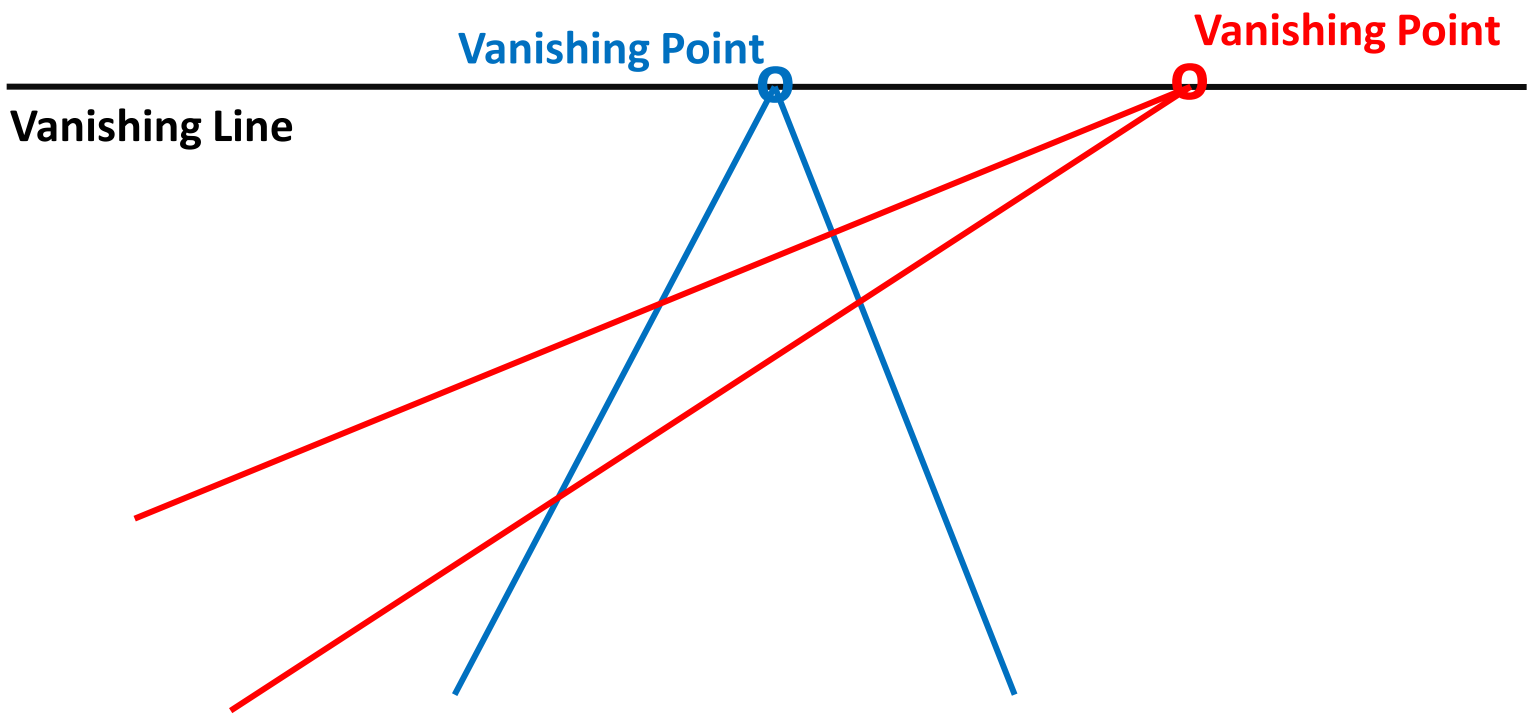


We can see infinity !

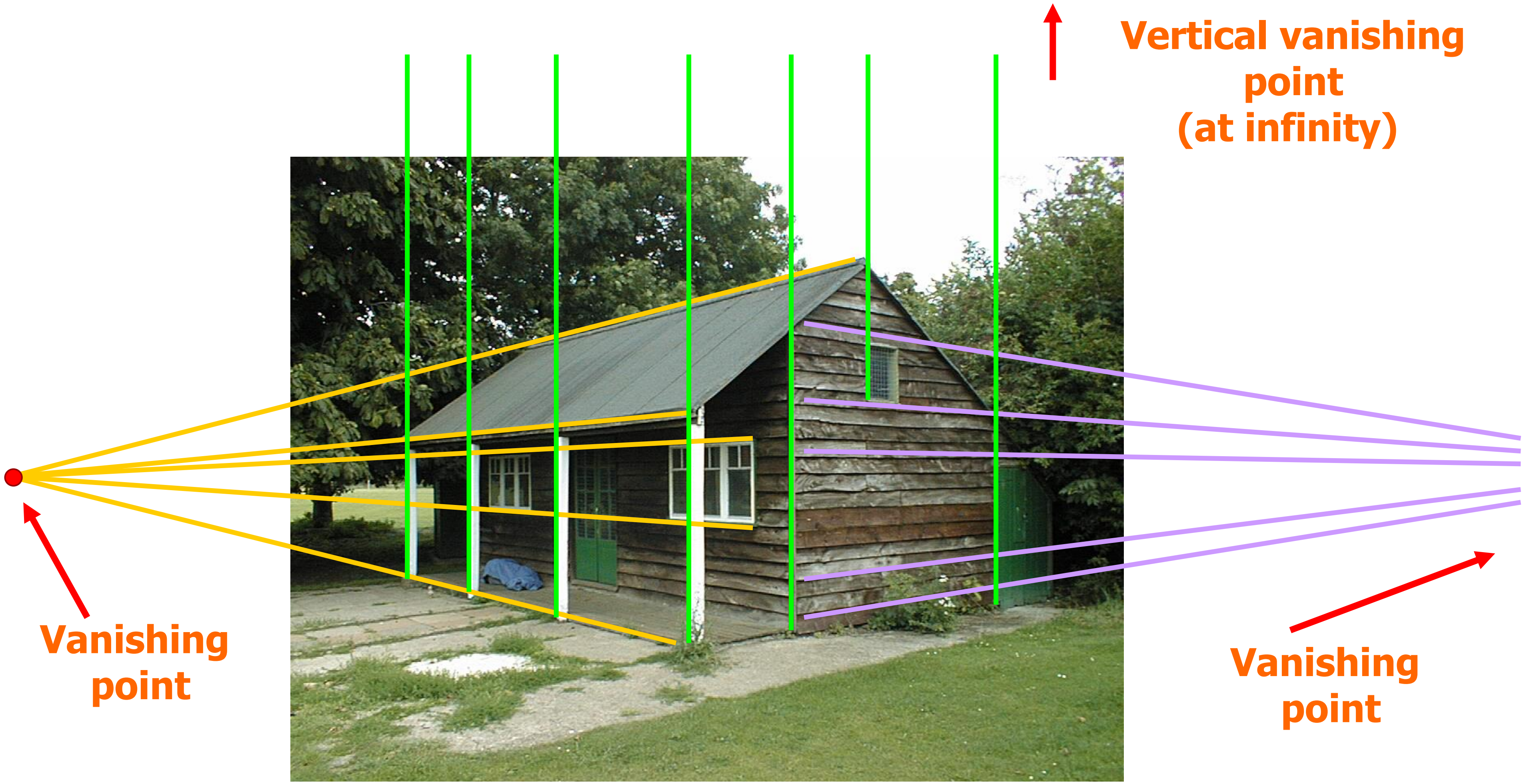
Railroad: parallel lines



Vanishing points and lines



Vanishing points and lines



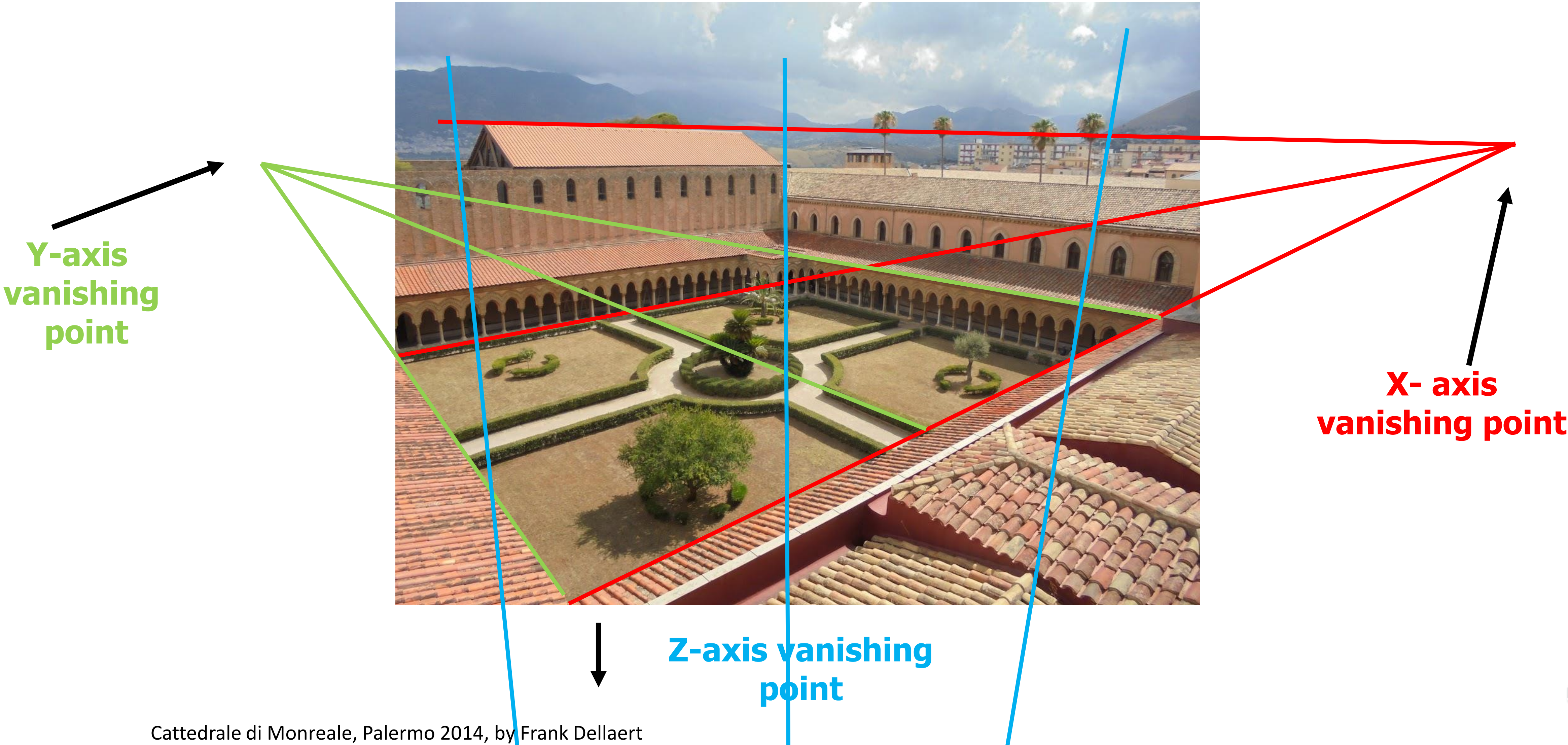
Vertical vanishing point (at infinity)

Vanishing point

Vanishing point

Slide from Efros, Photo from Criminisi

Vanishing points and lines

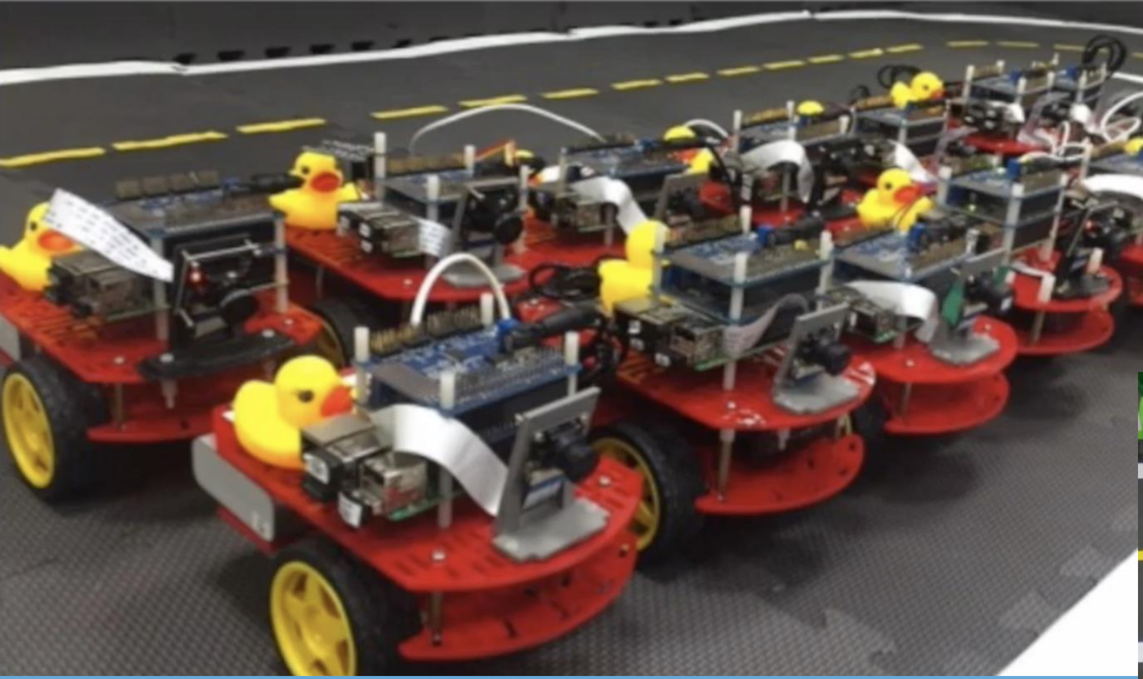


Y-axis vanishing point

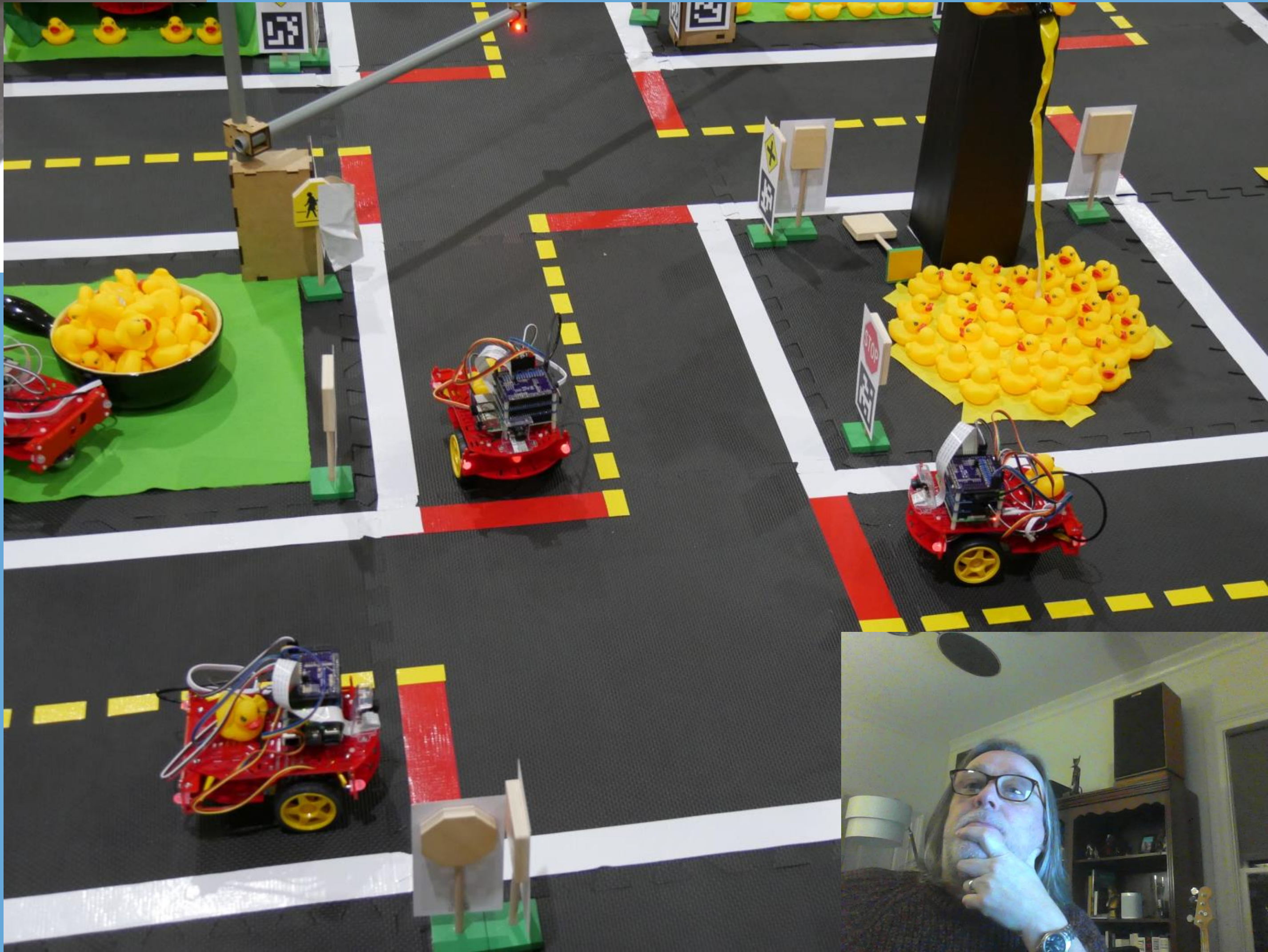
X- axis vanishing point

Z-axis vanishing point

Cattedrale di Monreale, Palermo 2014, by Frank Dellaert



CS 3630!



Lecture 17:
Computer Vision
Fundamentals

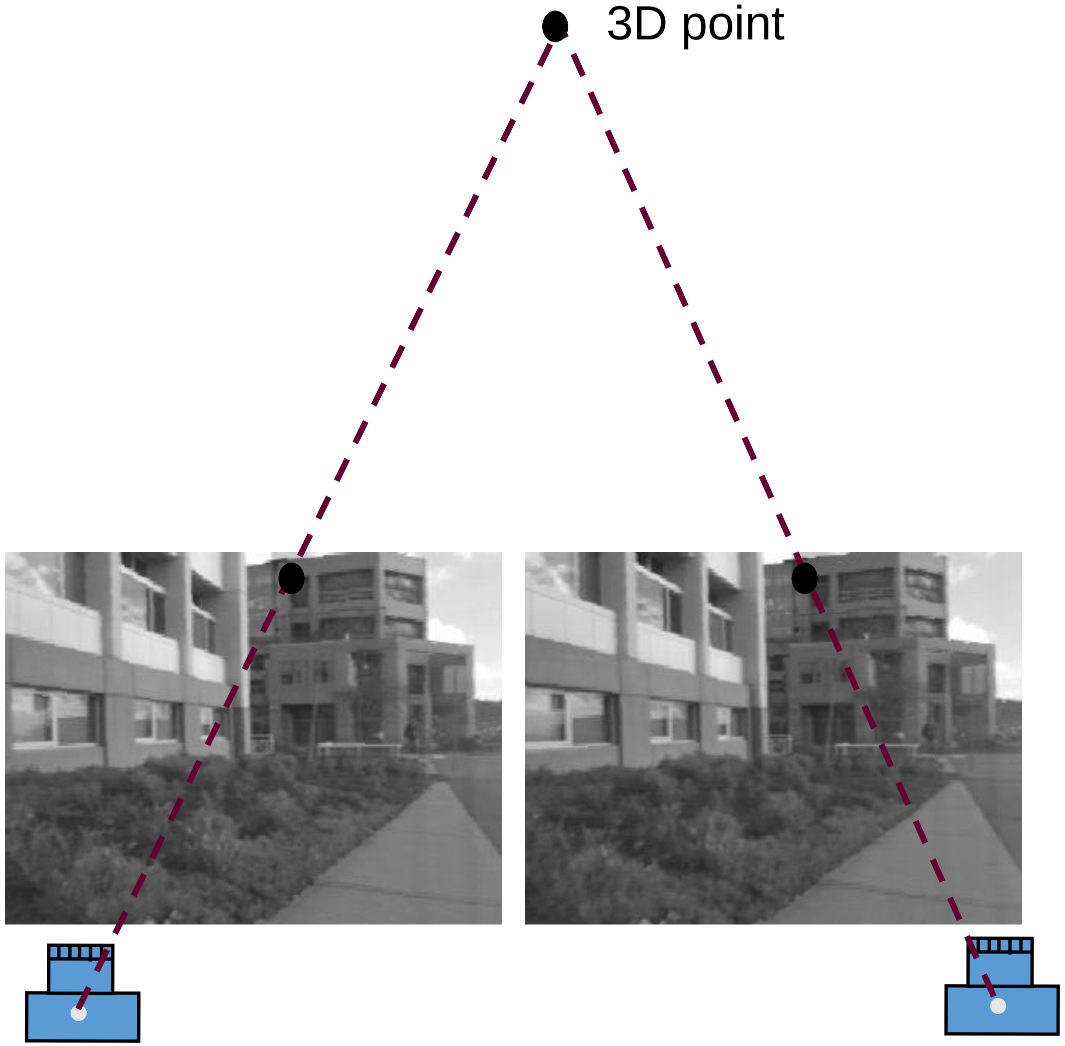
4. Stereo Vision

- 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

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

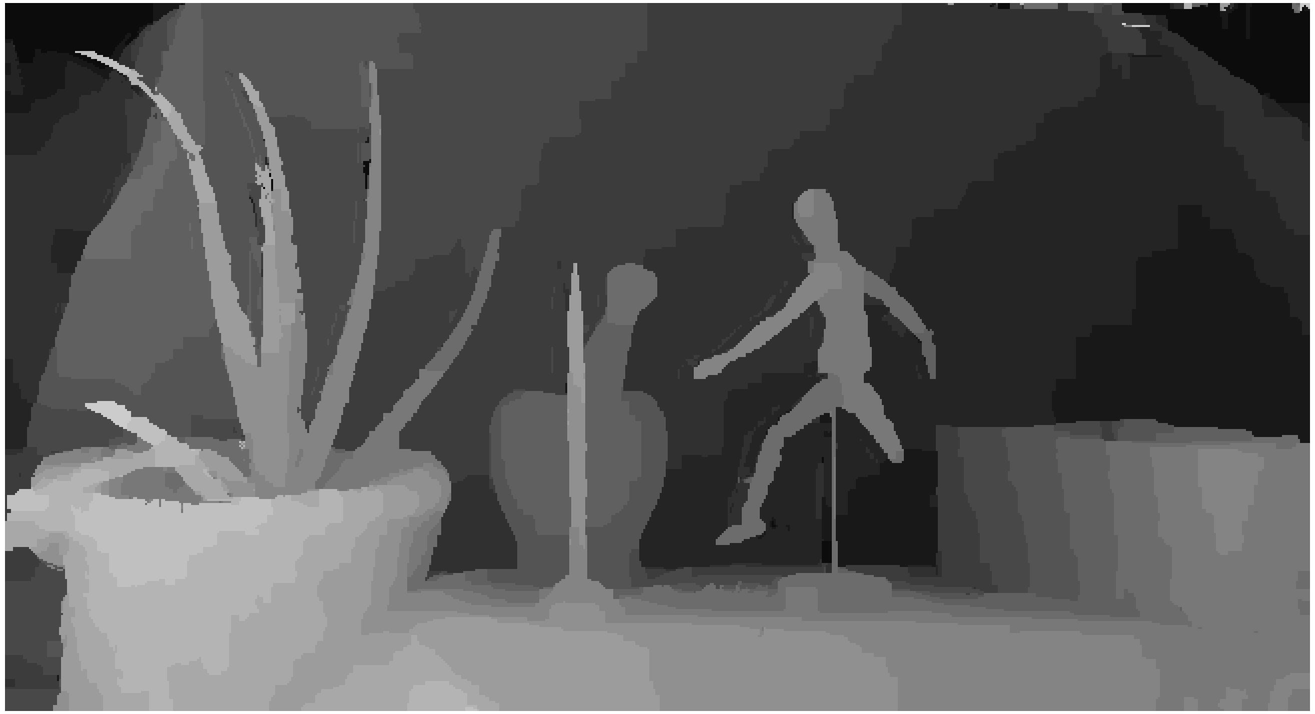


Example

Right Image

Left Image

Example



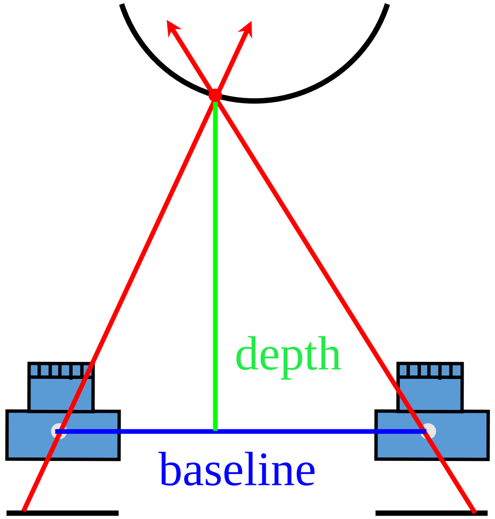
Right Image

Left Image

Disparity

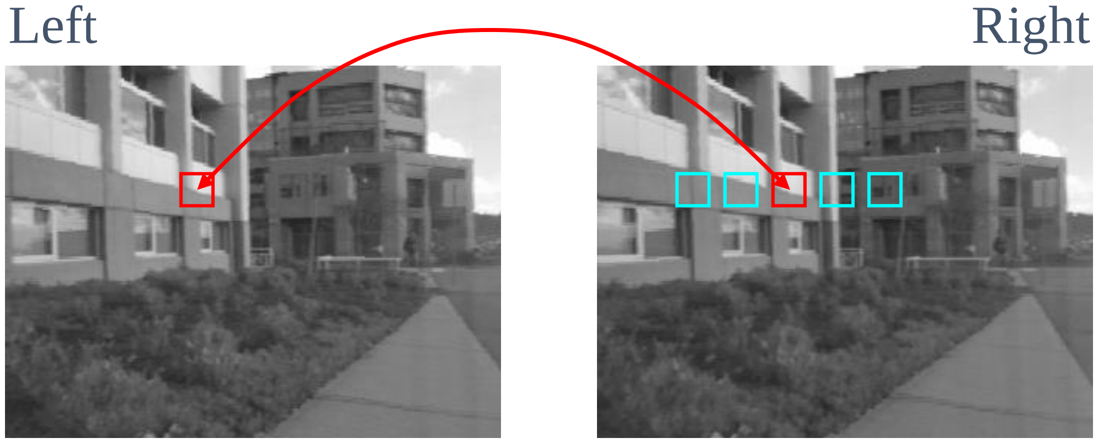
View Interpolation

Basic Idea of Stereo



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

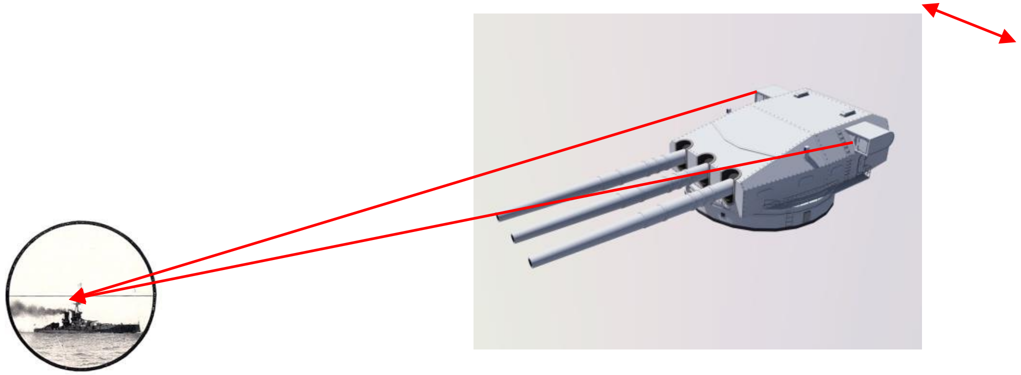
- Feature matching across views
- Calibrated cameras



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

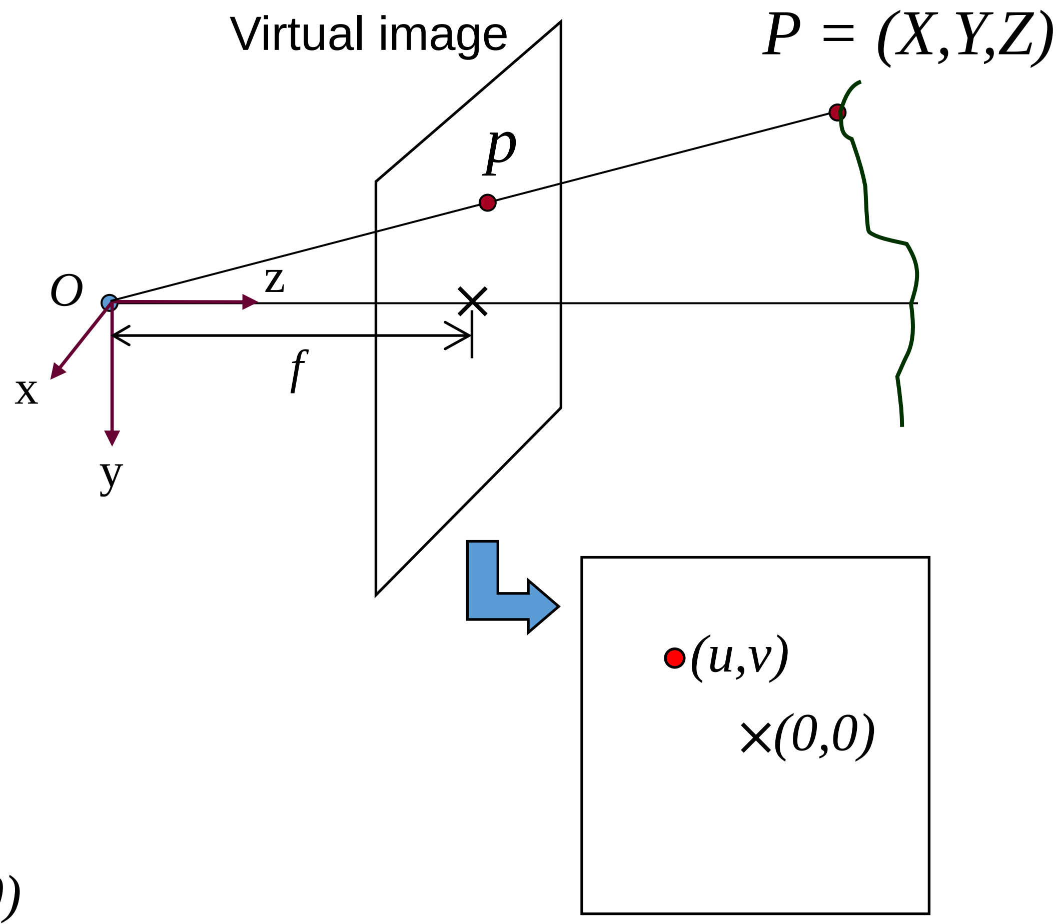


5. Stereo Geometry

- 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

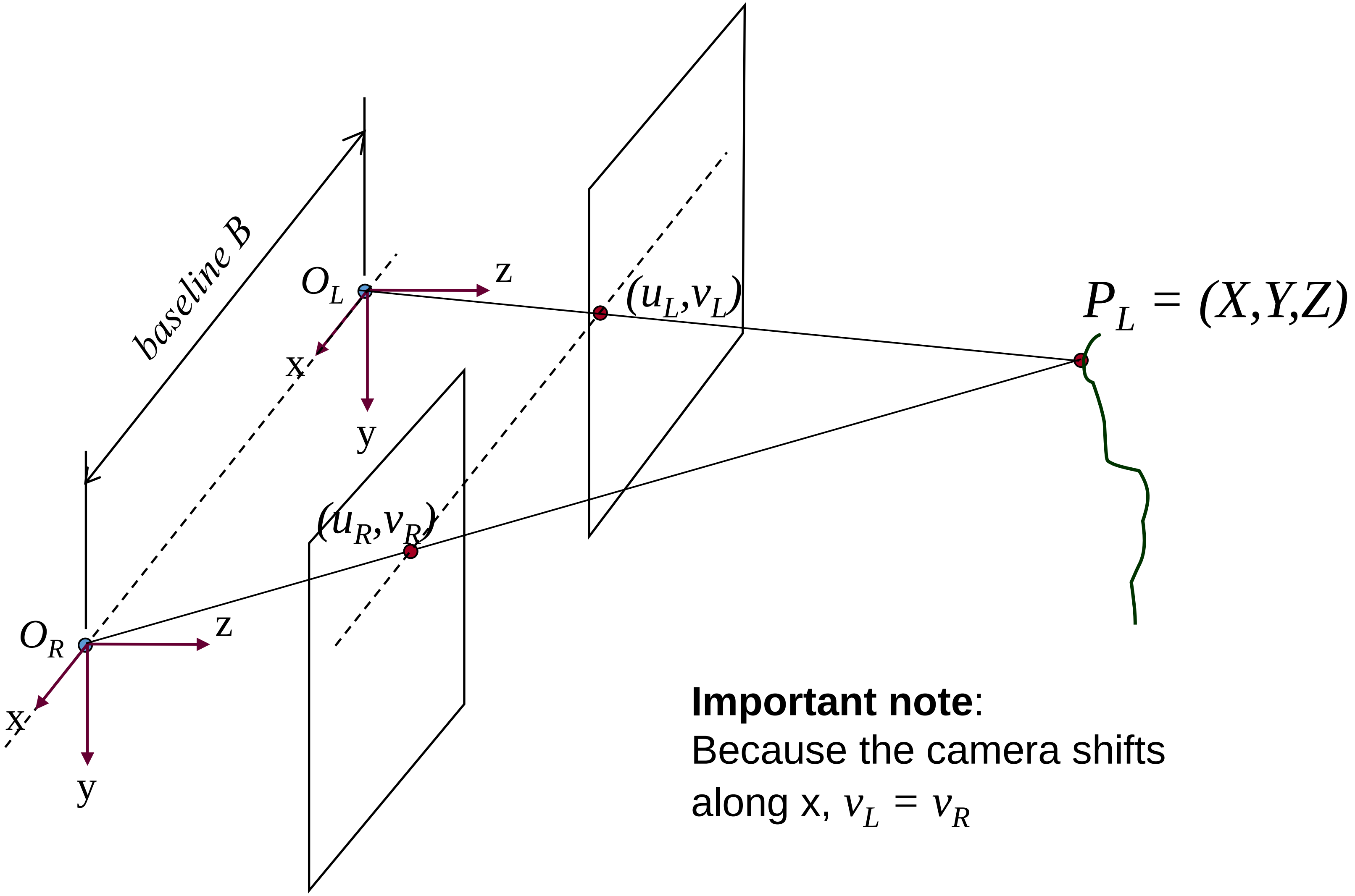


The 2D coordinates in the image are given by

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

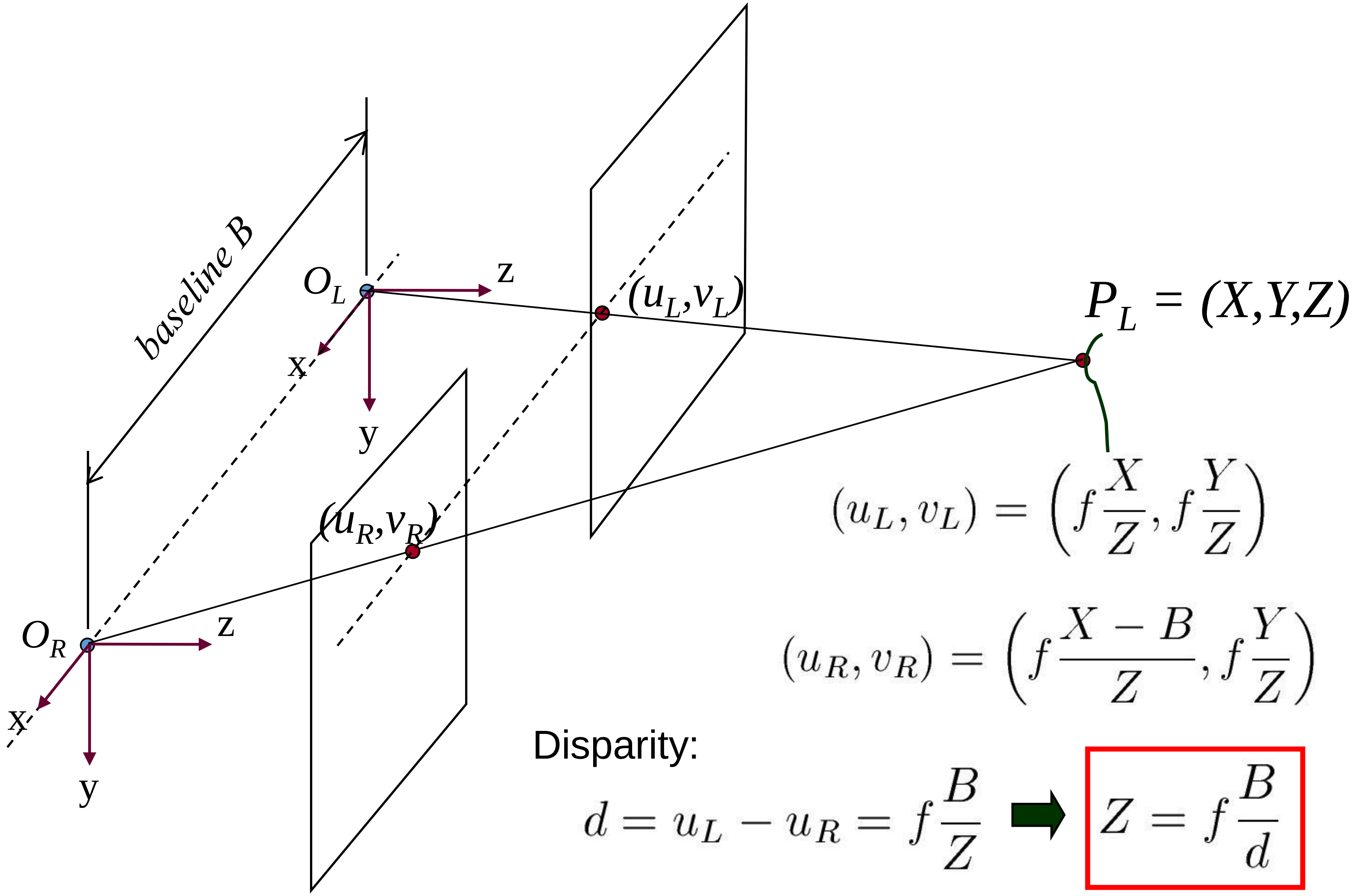
Note: image center is *(0,0)*

Basic Stereo Derivations



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

Basic Stereo Formula



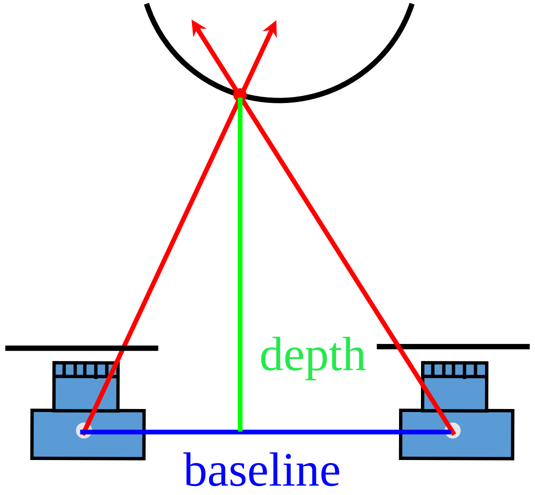
$$(u_L, v_L) = \left(f\frac{X}{Z}, f\frac{Y}{Z}\right)$$

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

Disparity:

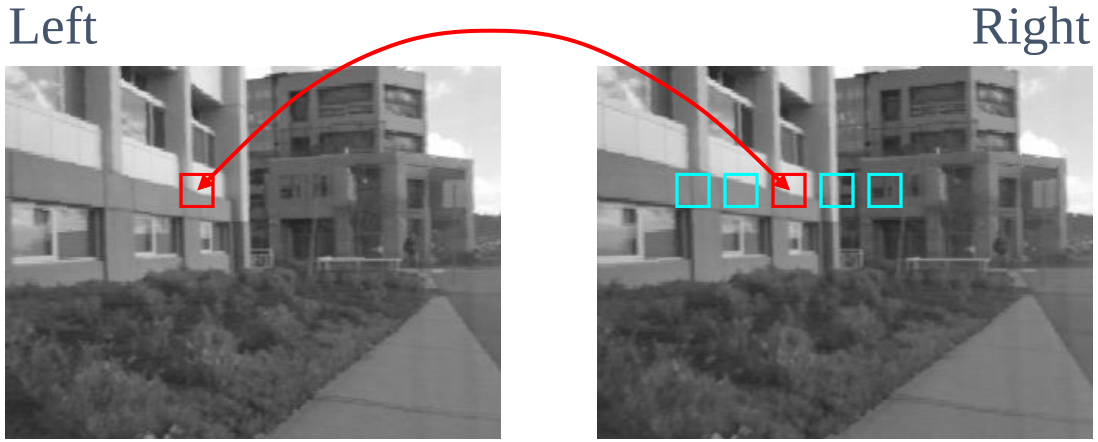
$$d = u_L - u_R = f\frac{B}{Z} \Rightarrow \boxed{Z = f\frac{B}{d}}$$

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



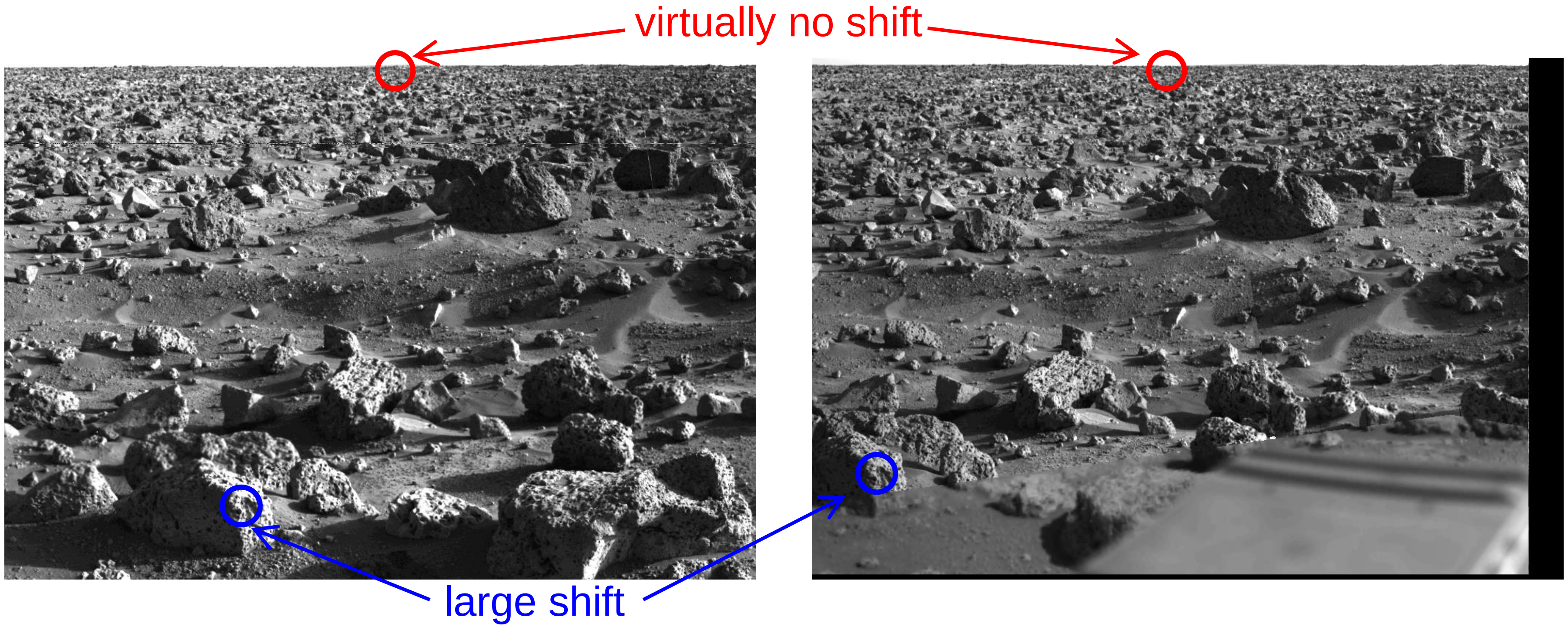
Matching correlation windows across scan lines

Components of Stereo Algorithms

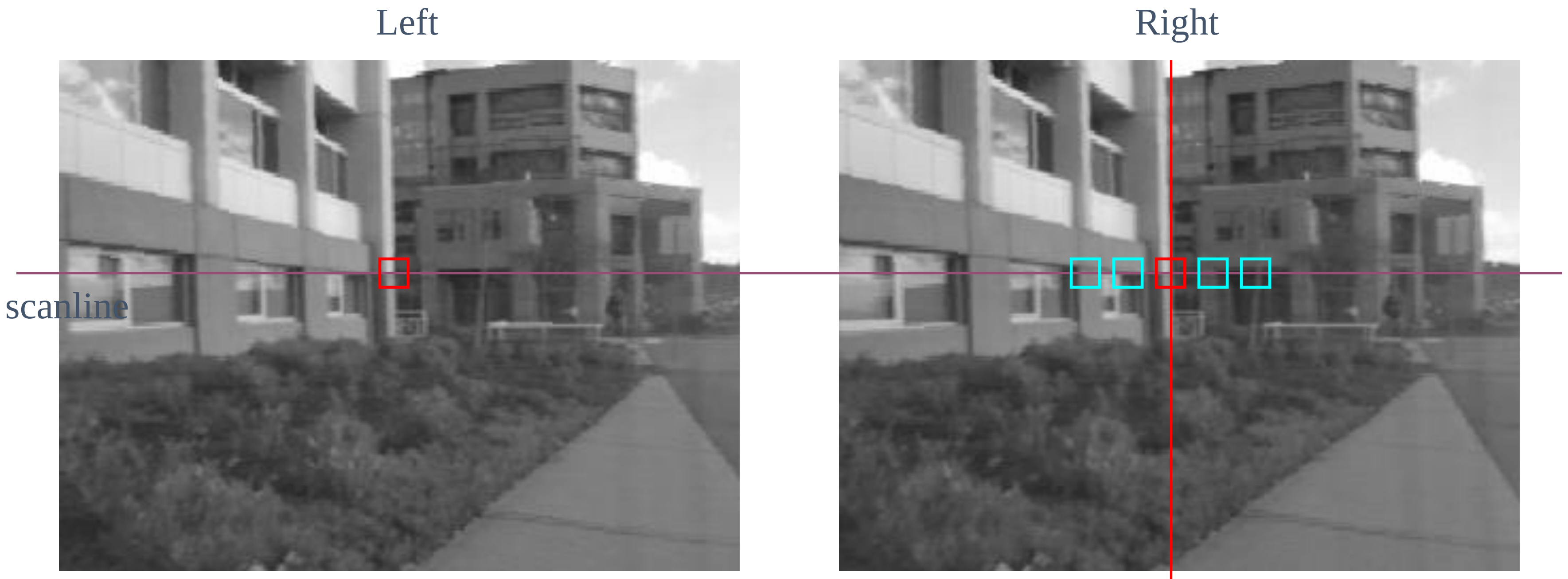
- Matching criterion (error function)
  - Quantify similarity of pixels
  - Most common: direct intensity difference
- Aggregation method
  - How error function is accumulated
  - Options: Pixel, edge, window, or segmented regions
- Optimization and winner selection
  - Examples: Winner-take-all, dynamic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

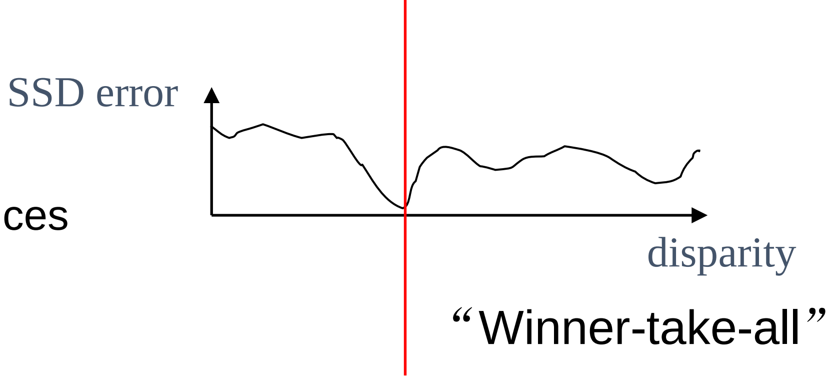
Stereo Correspondence

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



Correspondence Using Window-based Correlation



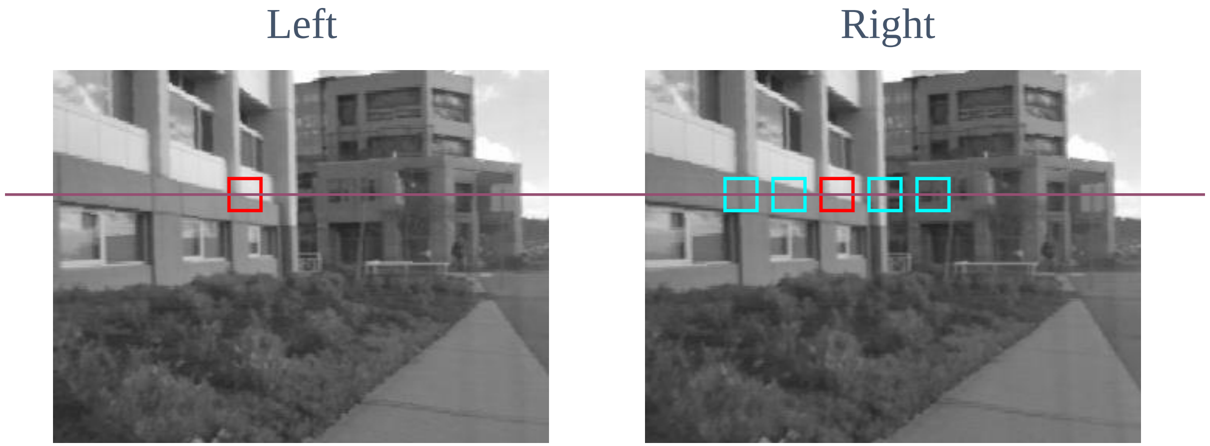


“Winner-take-all”

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:

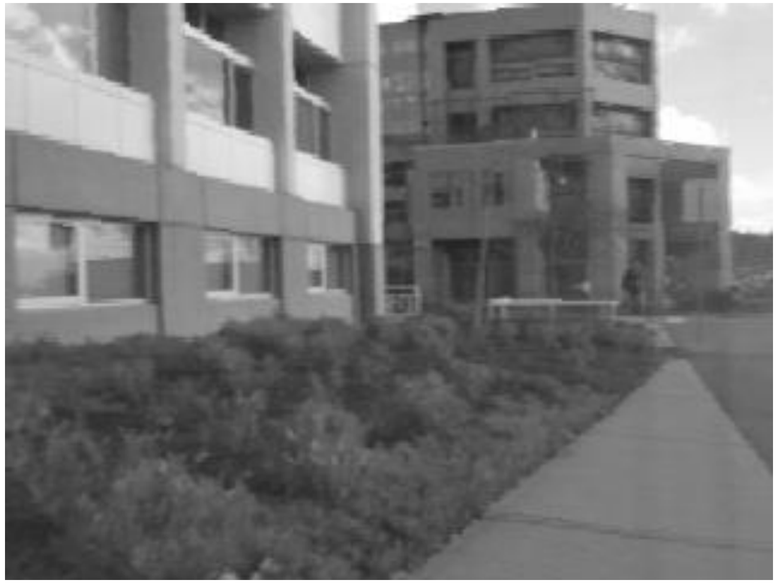
$$W_m(x,y) = \{u,v \mid x - \tfrac{m}{2} \le u \le x + \tfrac{m}{2}, y - \tfrac{m}{2} \le v \le y + \tfrac{m}{2}\}$$

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

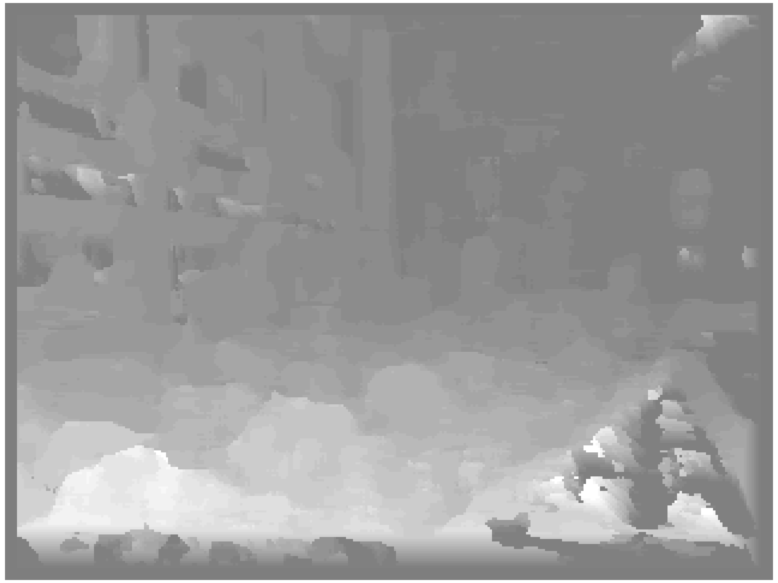
$$C_r(x,y,d) = \sum_{(u,v) \in W_m(x,y)} [I_L(u,v) - I_R(u-d,v)]^2$$

Correspondence Using Correlation

Left



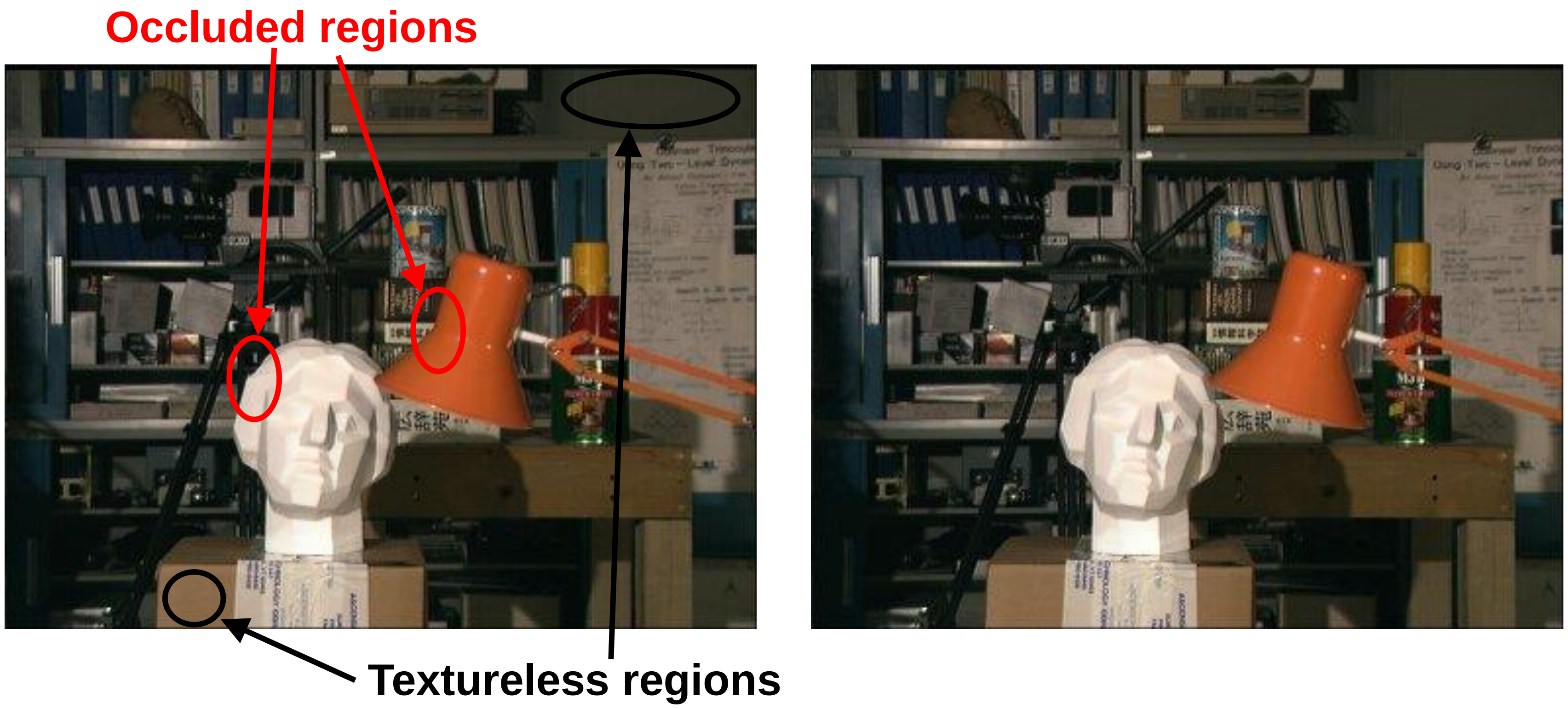
Disparity Map



Images courtesy of Point Grey Research

Two major roadblocks

- Textureless regions create ambiguities
- Occlusions result in missing data



Occluded regions

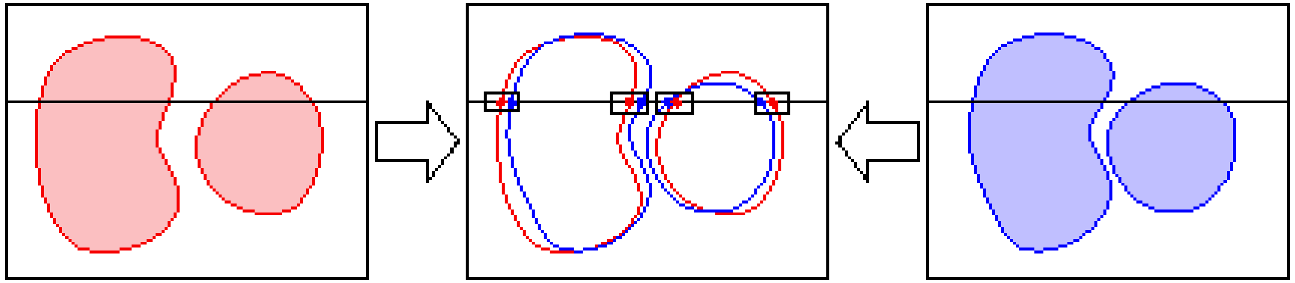
Textureless regions

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

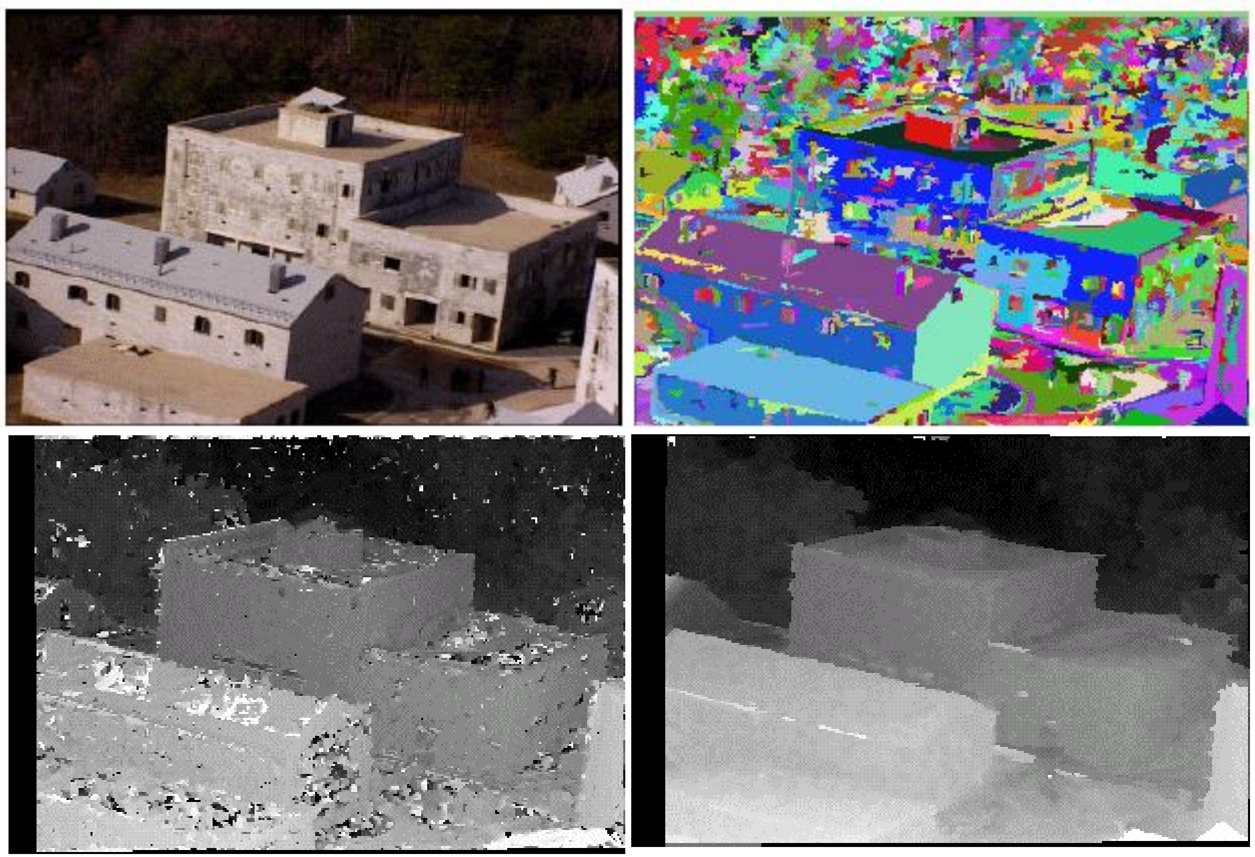
Edge-based Stereo

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



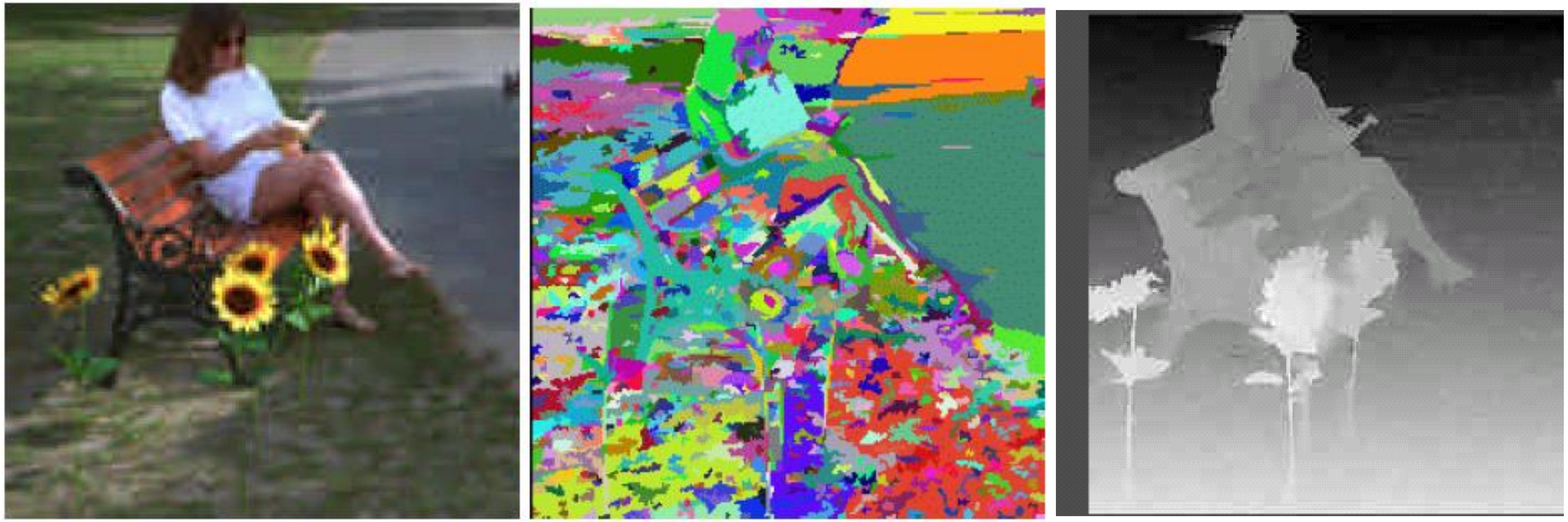
- Which method is better?
  - Edges tend to fail in dense texture (outdoors)
  - Correlation tends to fail in smooth featureless areas
  - Sparse correspondences

Segmentation-based Stereo



Hai Tao and Harpreet W. Sawhney

Another Example



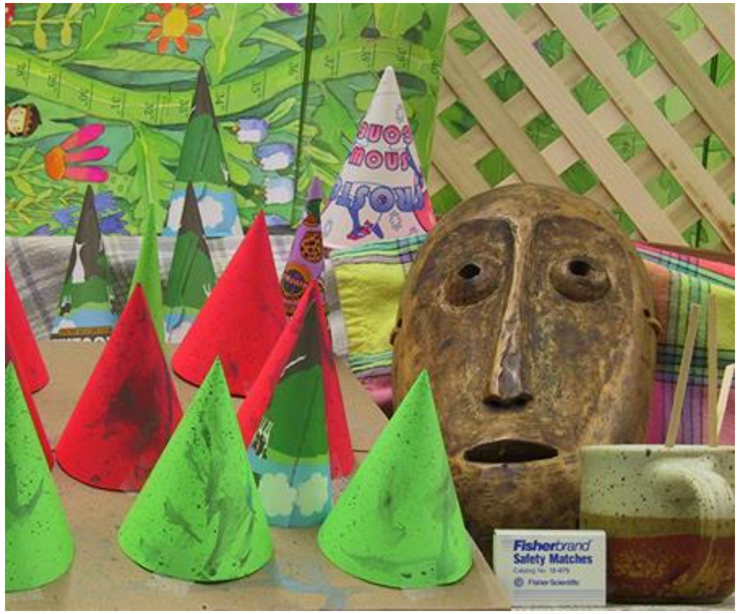
Stereo is Still Unresolved

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

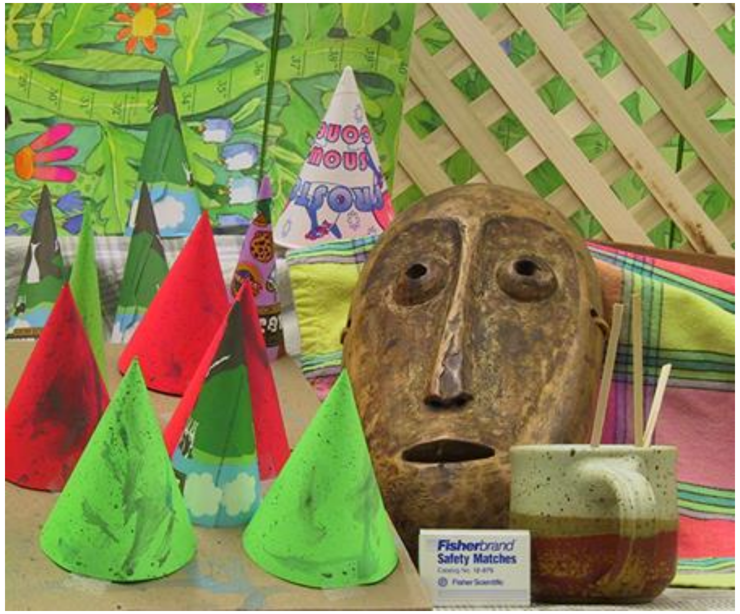
Hallmarks of
A Good Stereo Technique

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

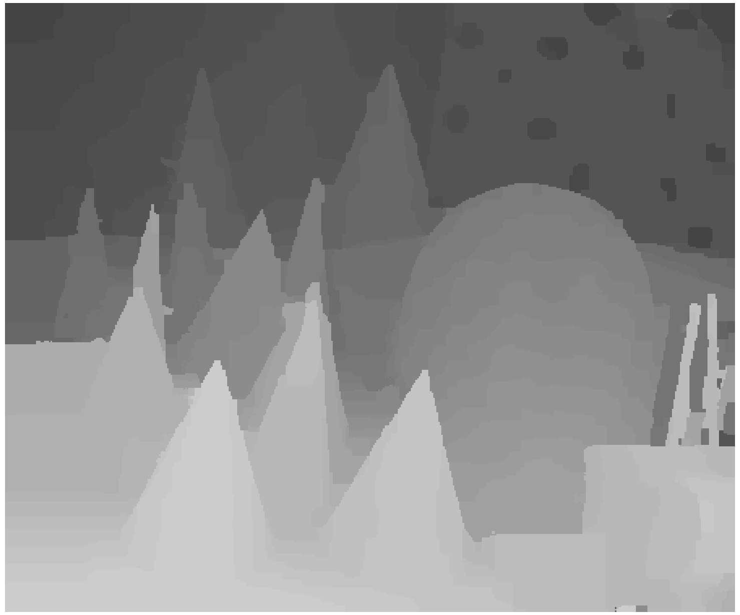
- Advanced: account for non-Lambertian surfaces



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 are still elusive