ECE5554 – Computer Vision
Lecture 10b – Stereo Vision

Creed Jones, PhD

Today’s Objectives

Stereo Vision

- The development of stereo vision
- Binocular imaging
- Disparity
- The correspondence problem
- An example of stereo disparity calculation

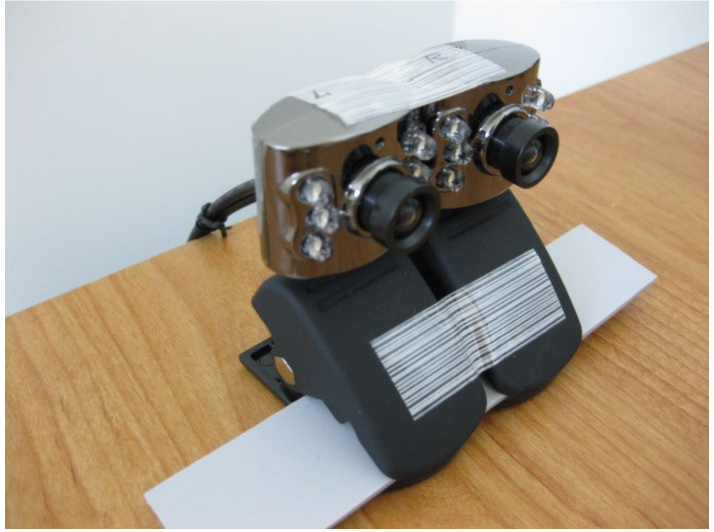
- Thanks to Dr. A. L. Abbott for many of the following slides

Stereo Vision consists of imaging a scene with two cameras, at a known spacing and orientation

- There are big advantages in having 2 eyes, rather than 1:
  - Redundancy
    (it's good to have a backup)
  - Stereopsis
    (assists in 3D perception)

- When 2 eyes (or cameras) are placed side by side, they receive slightly different views of a 3D scene

Source: http://www.activrobots.com



Source: http://www.starlino.com/opencv_qt_stereovision.html

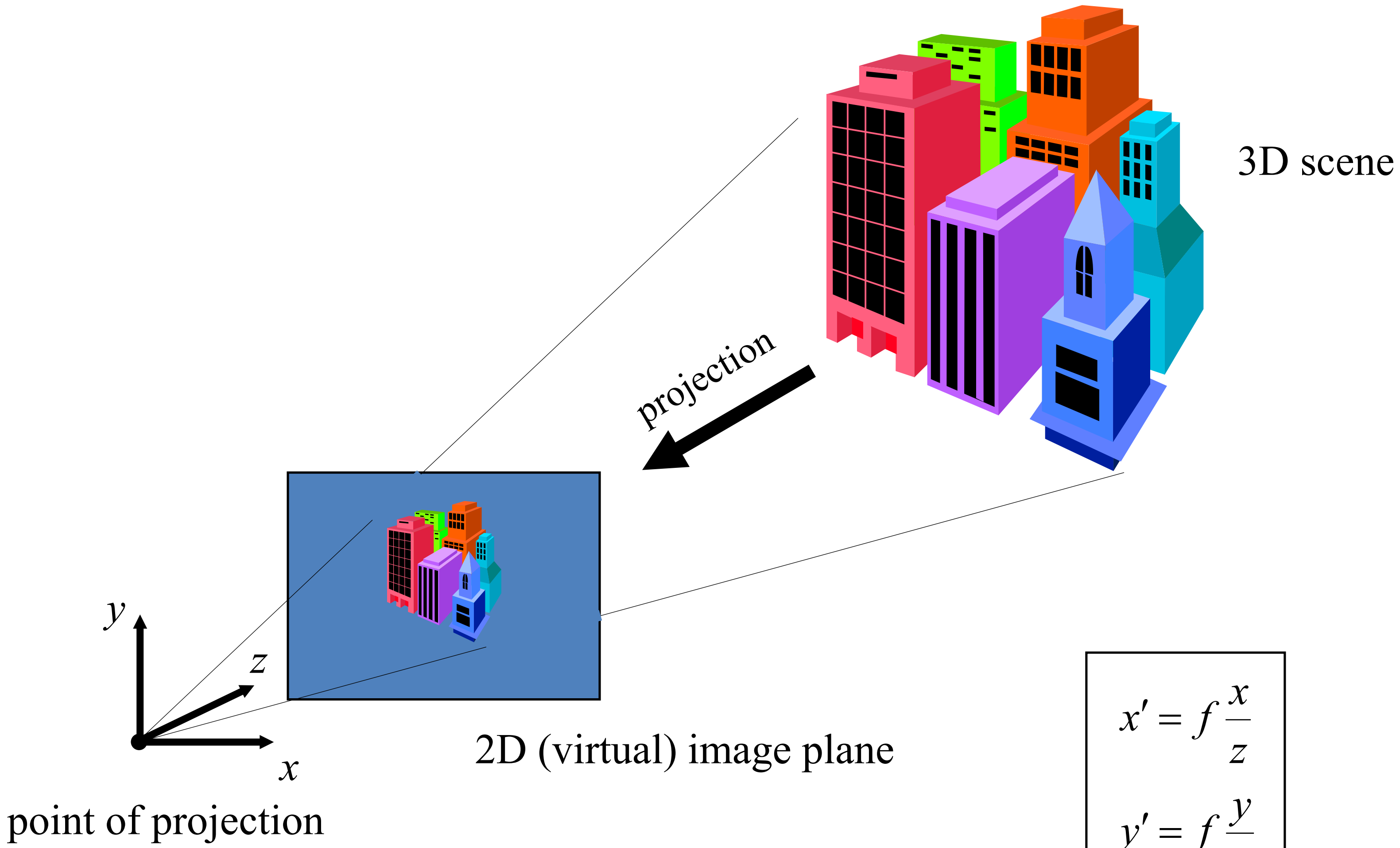
Source: Wikipedia

Why multiple views? 3D structure and depth are inherently ambiguous from single views

- We cannot determine an object’s location along the ray from the camera to the object…

Credit: Grauman, Labeznik

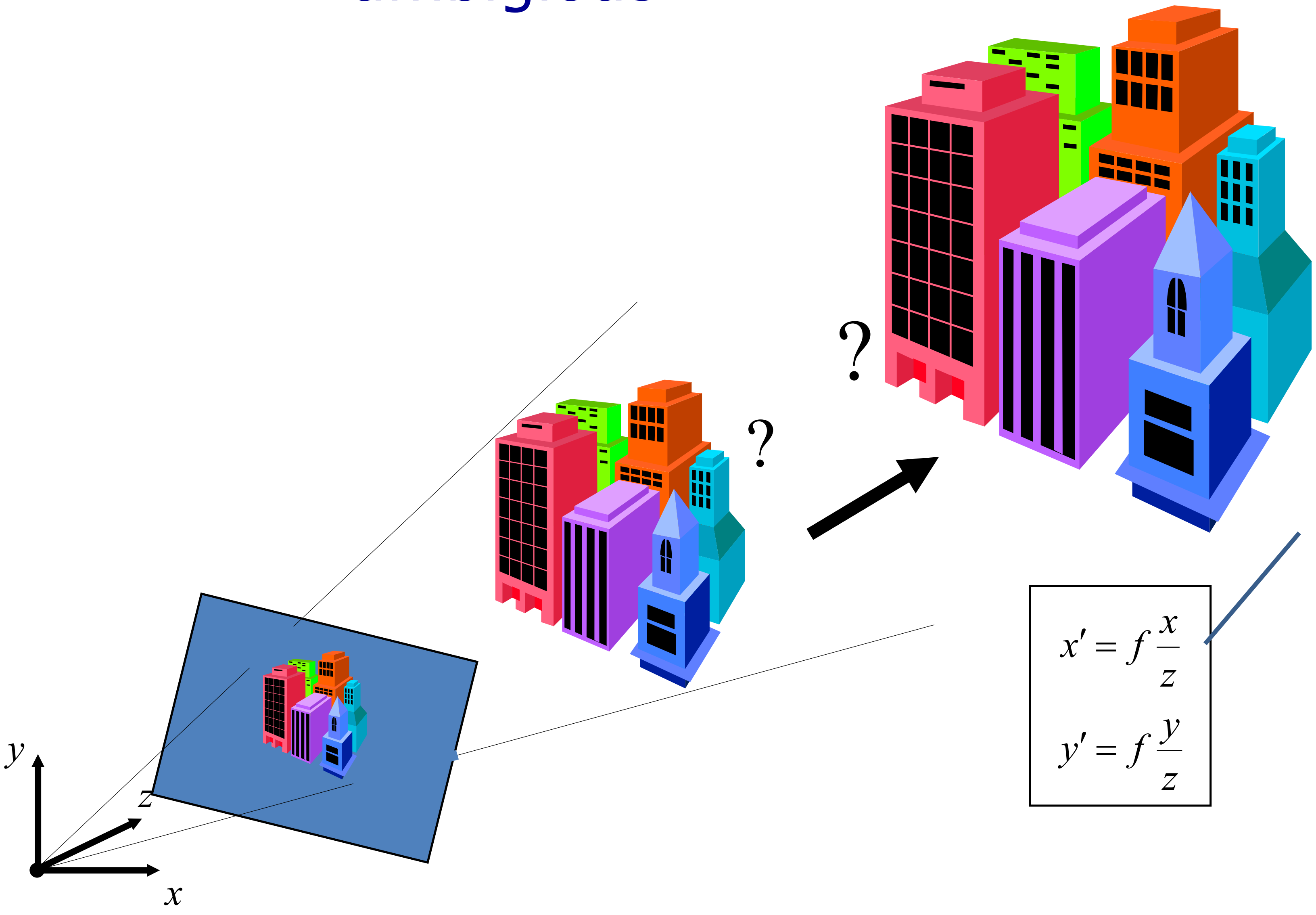
Due to perspective projection, we can’t determine an object’s position along the ray from the camera to the object…



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

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

The fundamental problem: backprojection is ambigious



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

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

There are <u>many</u> combinations of x and z that will produce the same image coordinate x’

How can we address this problem? What can we use to deduce distance to the object from the image?

- Humans use many different visual cues in order to perceive depth
  - shading
  - texture
  - focus
  - *binocular disparity*
  - etc.

- Binocular disparity is the key to stereo vision
  - (More on those other topics later)

Stereo imaging:
2 or more views of a scene

Image from <u>left</u> camera

Image from <u>right</u> camera

Because of the different viewpoints,
small differences ("disparities") are present in the images

The importance of stereo disparity for determining depth was not always well understood

- Before 1838, everyone thought that these small differences were unimportant, or perhaps "noise" to be ignored

- In 1838, in the early years of photography, Wheatstone invented the stereoscope

An old stereopticon and a print used in it

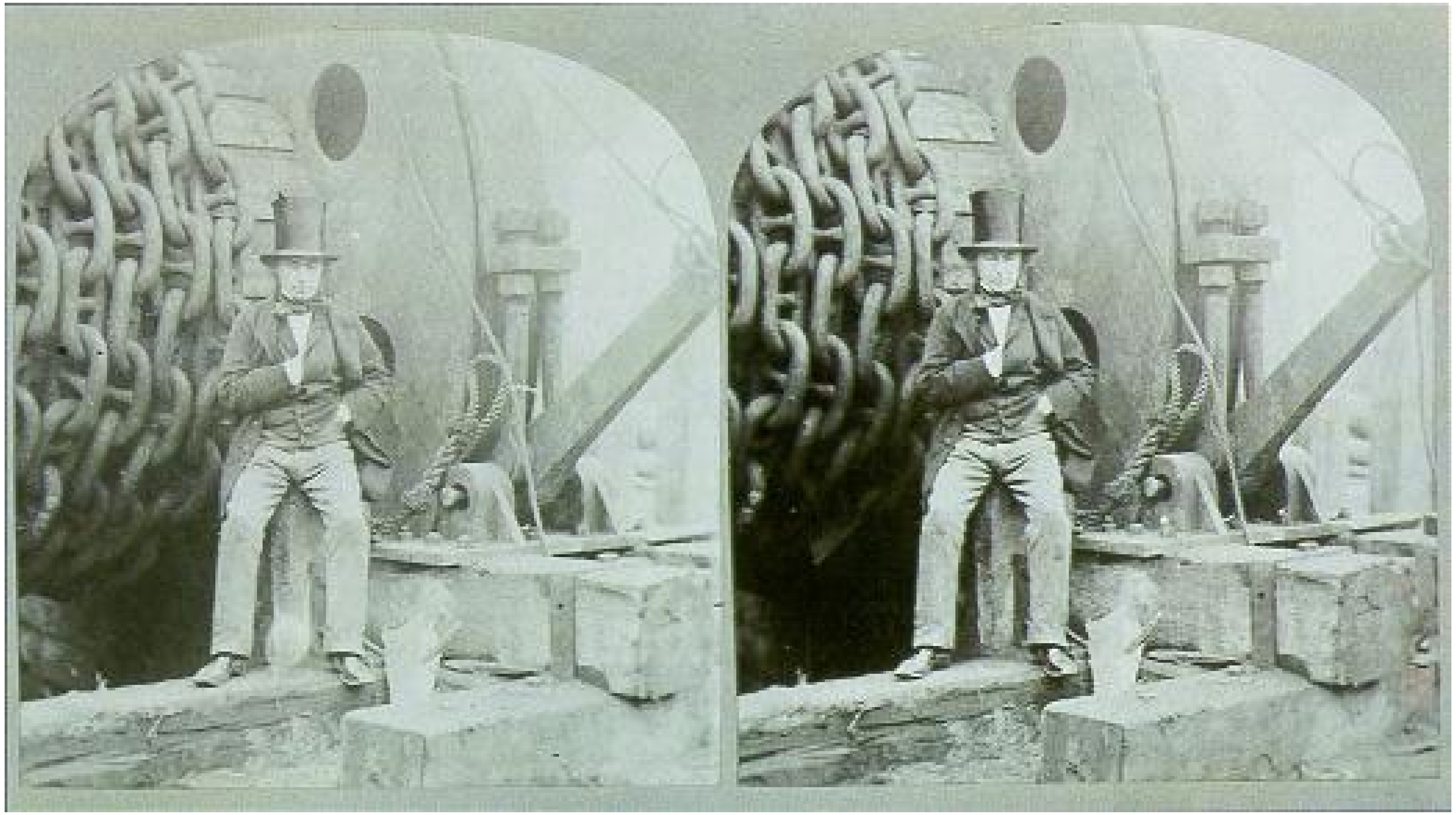


Image from <u>left</u> camera Image from <u>right</u> camera

Note the areas of difference...

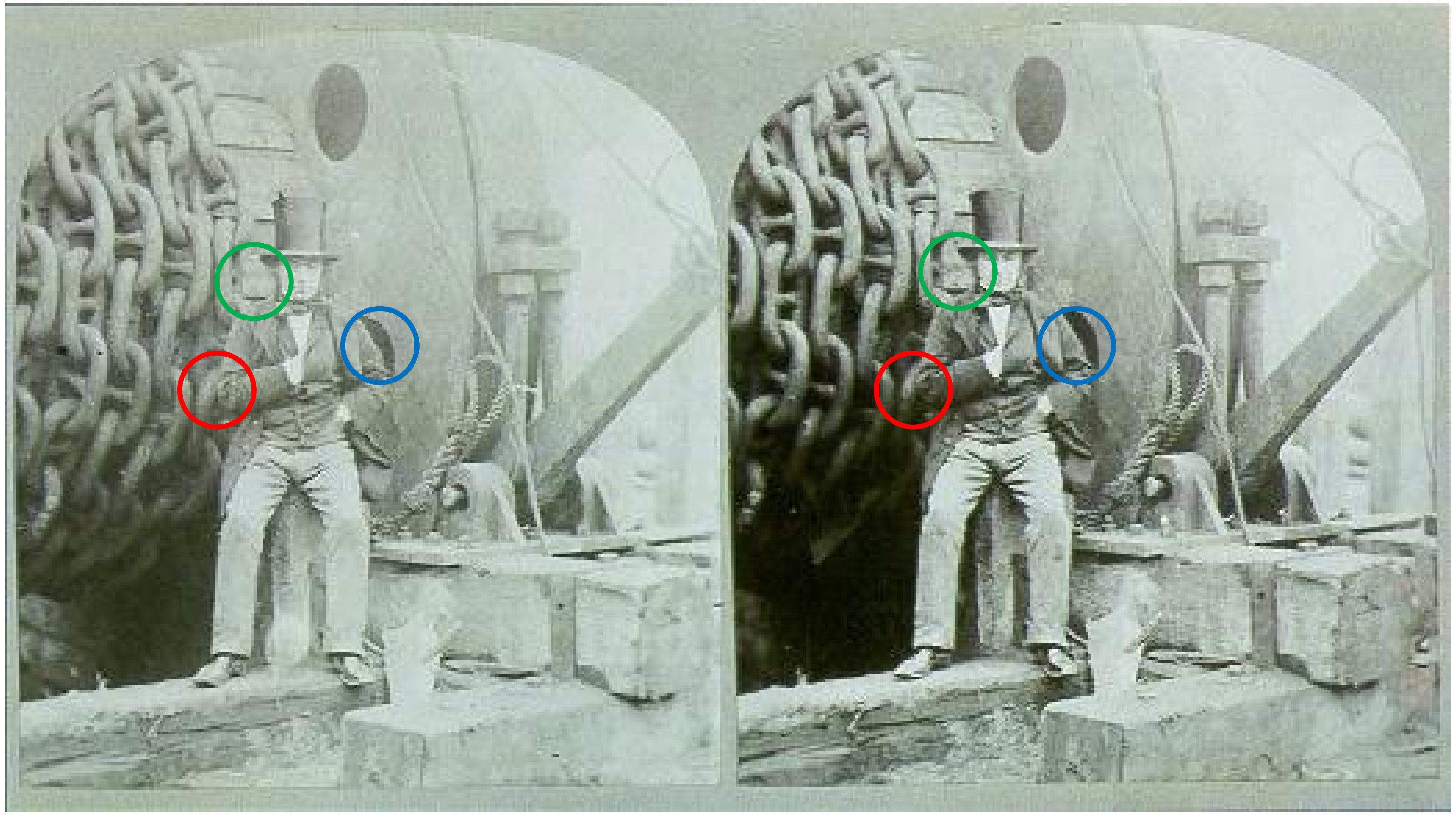


Image from <u>left</u> camera Image from <u>right</u> camera

Theodore Roosevelt at Yosemite, 1903

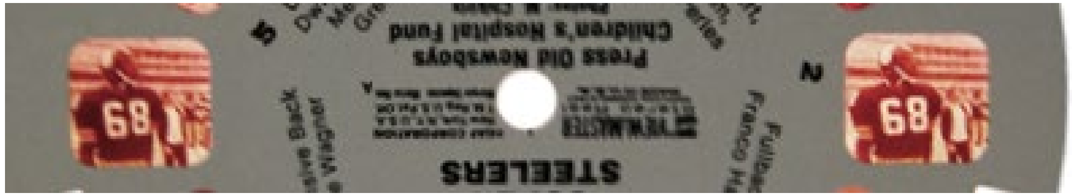
http://hdl.loc.gov/loc.pnp/stereo.1s02031

Theodore Roosevelt at Yosemite, 1903

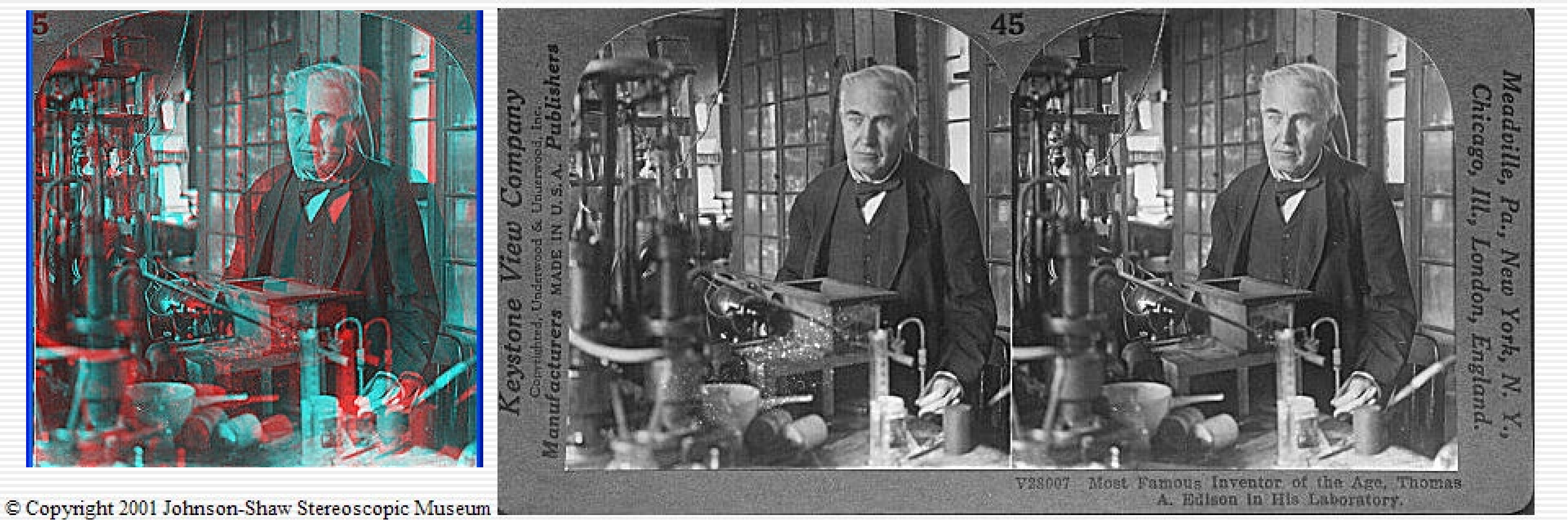
http://hdl.loc.gov/loc.pnp/stereo.1s02031

Stereo photography and stereo viewers use two pictures of the same scene taken from slightly different viewpoints and display them so that each eye sees only one of the images

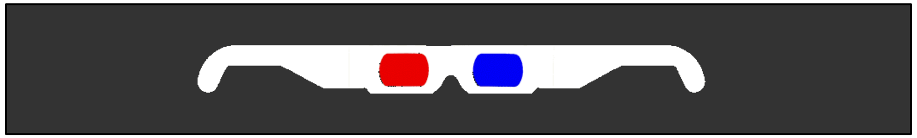
Image from fisher-price.com



Old-style 3D glasses split the image between the two eyes using color; the left eye only sees the red objects and the right eye only sees the blue



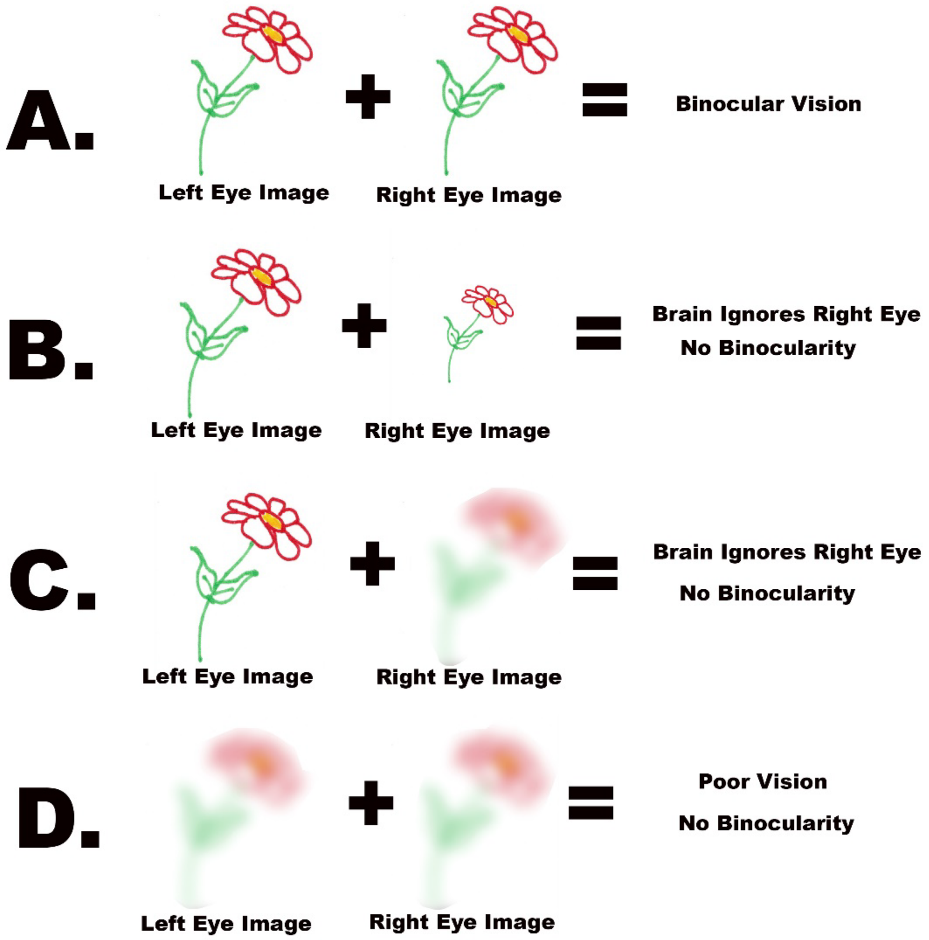
http://www.johnsonshawmuseum.org



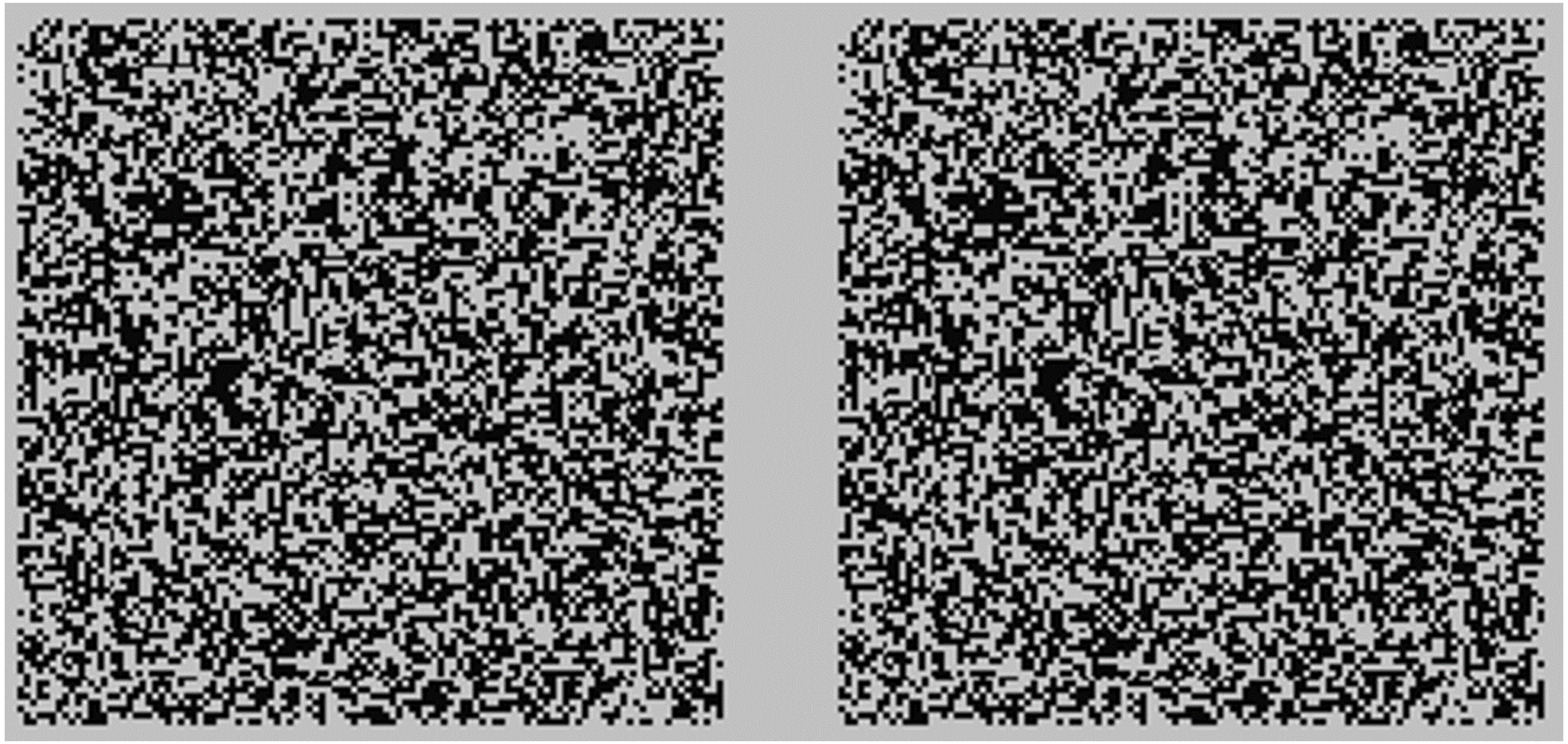
Humans do not normally notice the differences ("disparities") between the retinal images

- **Binocular fusion** takes place in the visual cortex

  (About 15% of the population do not experience binocular fusion)
- Psycholgists define something called a **cyclopean image**, which is a single "mental image" created the combination of left and right stimuli
- Before 1960, people wondered if stereopsis perhaps depended on other visual cues



In 1960, Julesz invented the *random-dot stereogram*



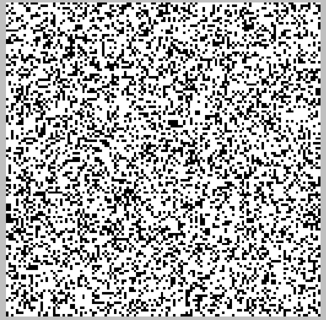
- *No monocular depth cues!*
- Create an image by placing dots at random; copy that image, and then adjust the dots slightly to introduce disparities
- When viewed stereoscopically, most people experience a vivid sensation of depth

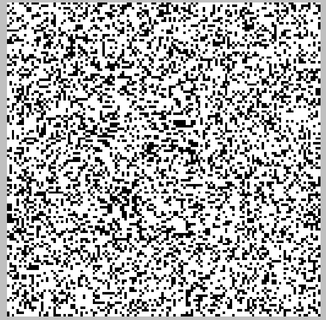
More recently:
single-image random-dot stereograms

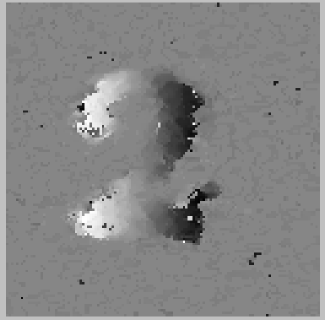
Credit: MagicEye.com

Random-Dot Stereograms

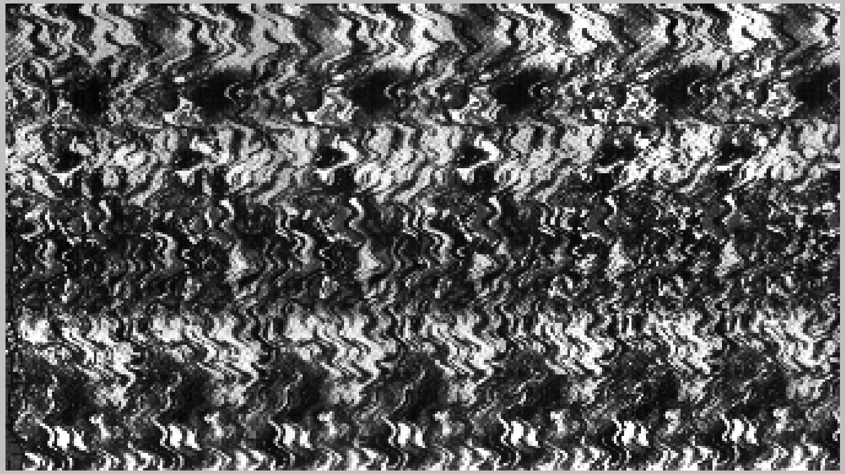
Coherence-based stereo utilizes simple disparity estimators which work directly with the image intensities. Obviously, an intensity-based algorithm might have difficulties with images composed only of black and white pixels, like classical random-dot stereograms. Here's the result of a calculation with such an image pair (see also here for a sparse RDS):

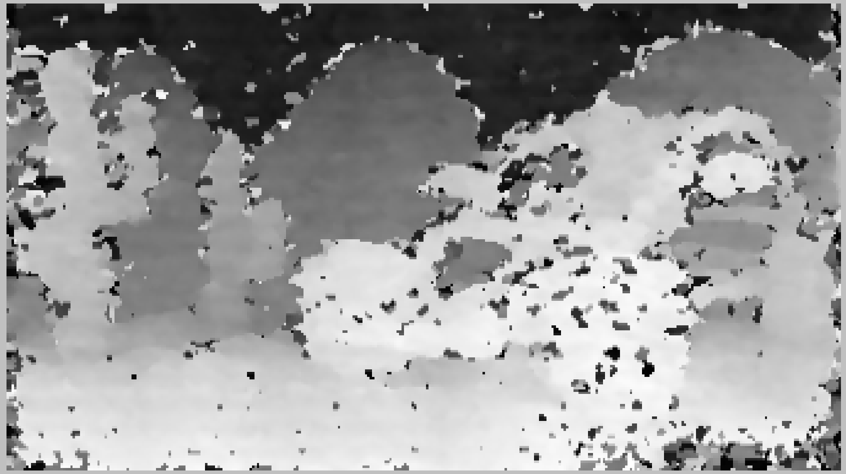

left picture


right picture


calculated disparity

Another difficult test for stereo algorithms is the repetitive structure found in many in SIRDS. Here's an example of a disparity map calculated from such an image:


SIRD, coding a beach buggy

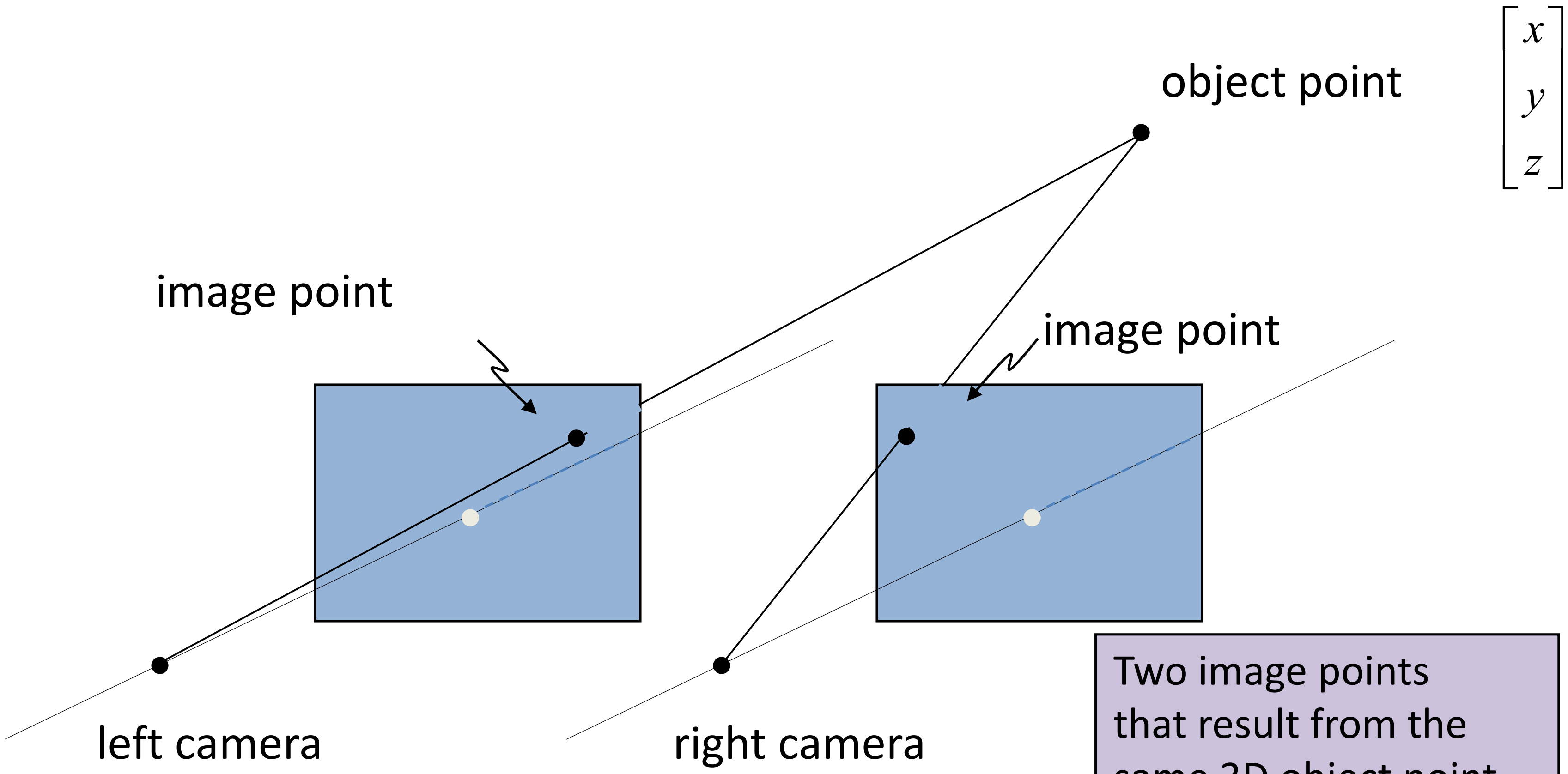

calculated disparity map

http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/HENKEL/research/stereo/rds/

Stereo chronology

- **1838**
  Wheatstone invents the stereoscope
- **1960**
  Julesz devises the first random-dot stereograms
- **1960s and 70s**
  Many attempts to develop stereo software
- **1979**
  Marr and Poggio propose a model of human stereo vision based on coarse-to-fine matching of edges
- **2000 and later**
  New feature detectors (e.g., SIFT and ORB) produce sparse sets of points that can be more reliably matched than edges alone; then use these results as “seed“ points for area-based matching

Binocular stereo



Two image points that result from the same 3D object point are called "corresponding points"

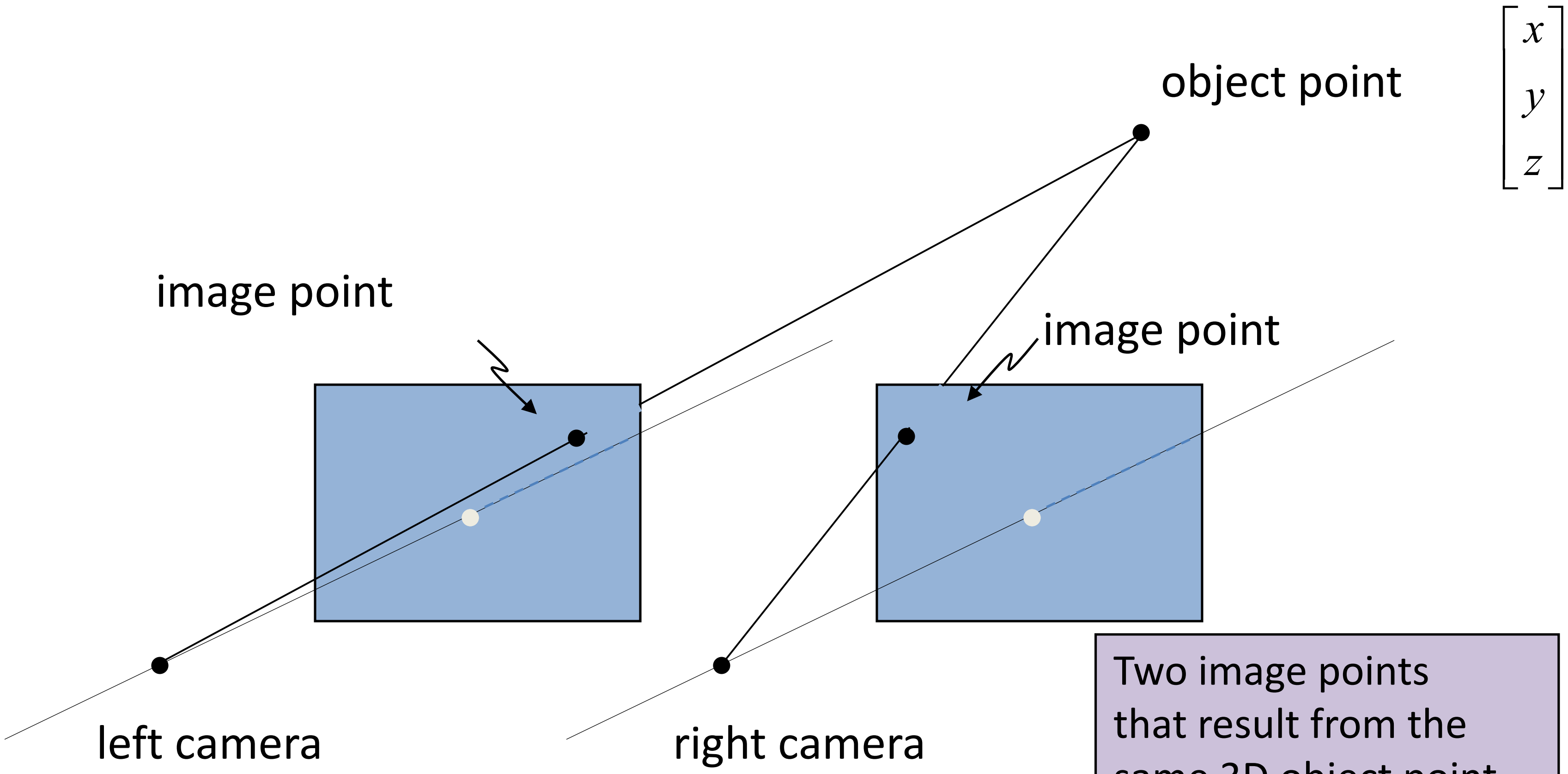
Given: Left and right images

Goal: Determine [*x, y, z*] wherever possible

Question – What if our eyes were above-and-below, instead of side-by-side?

- Would our stereo vision still work the same way?
- Would it work at all?
- Would we be able to sense depth?

Binocular stereo

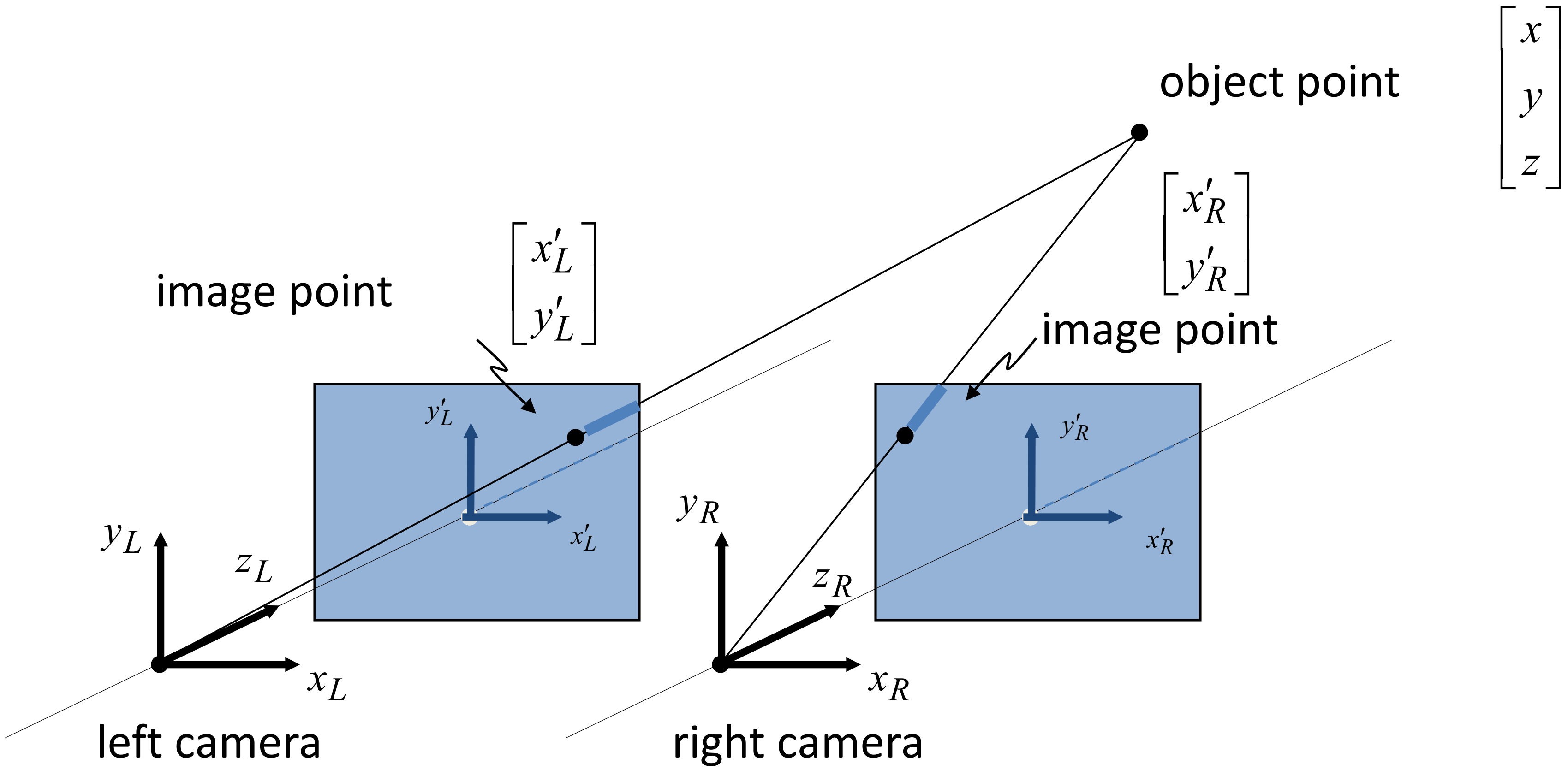


Two image points that result from the same 3D object point are called "corresponding points"

Given: Left and right images

Goal: Determine [*x, y, z*] wherever possible

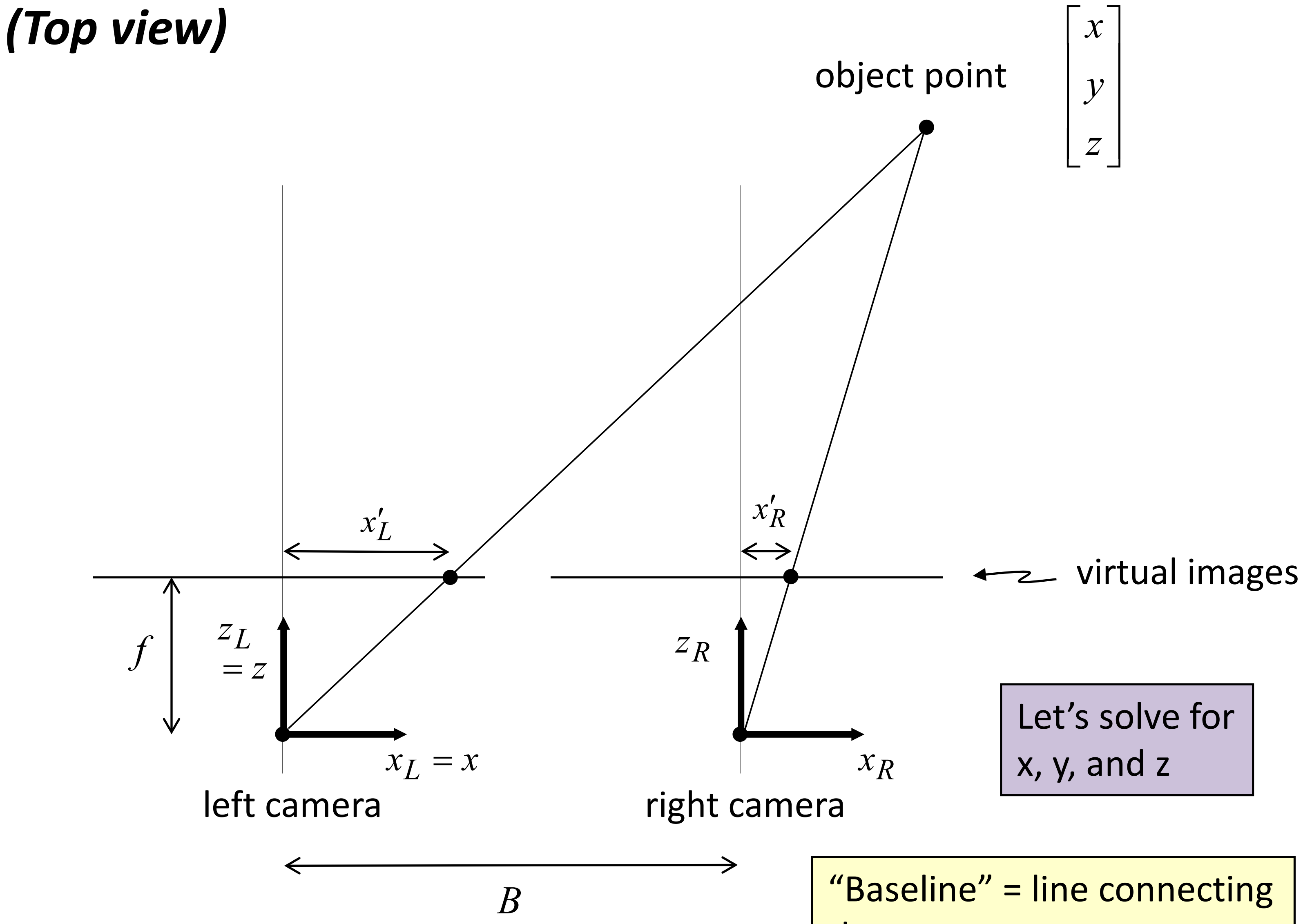
Binocular stereo



Given: Left and right images

Goal: Determine [*x*, *y*, *z*] wherever possible

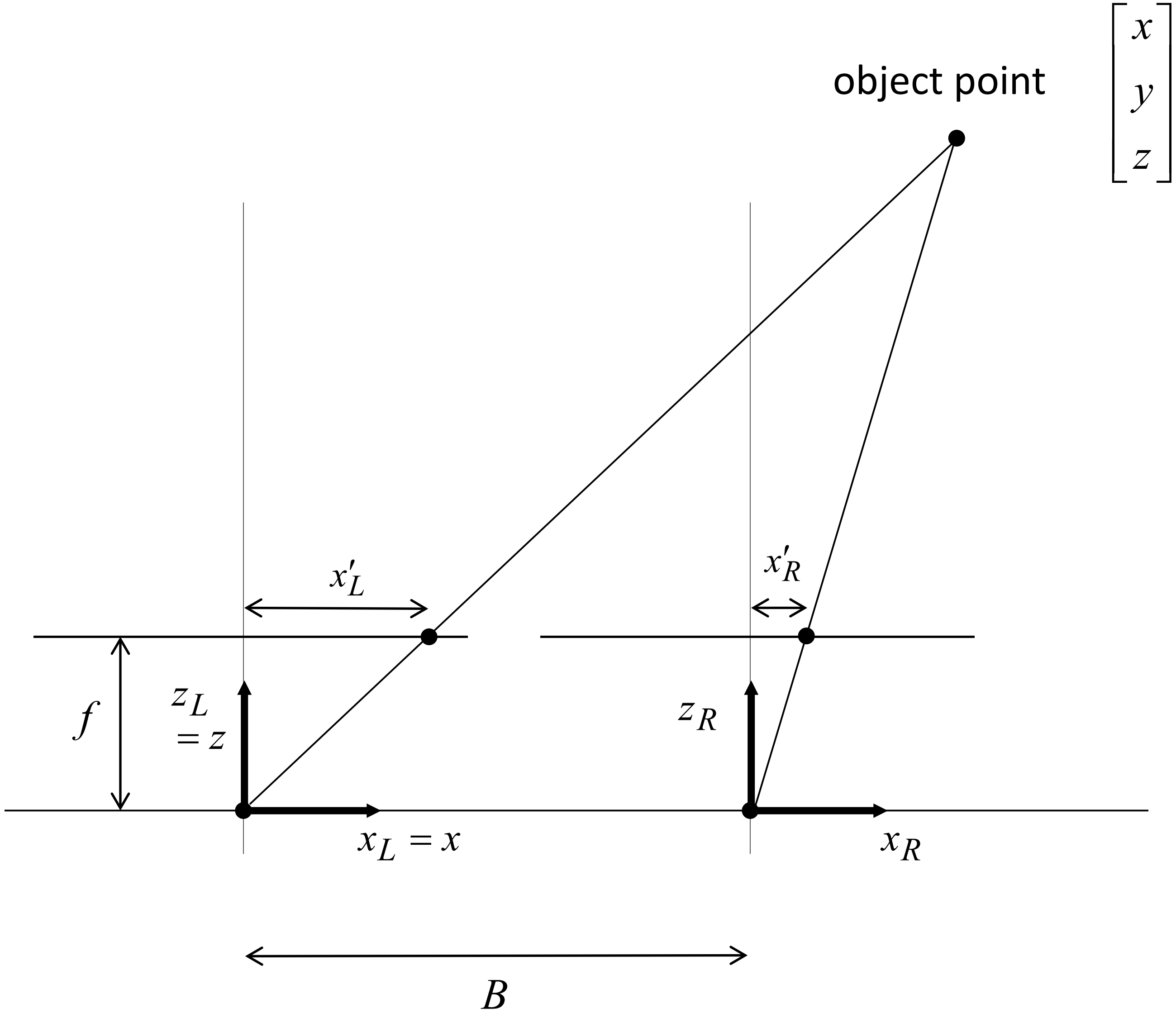
A simple stereo imaging system



Let's solve for x, y, and z

"Baseline" = line connecting the two camera centers

A simple stereo imaging system

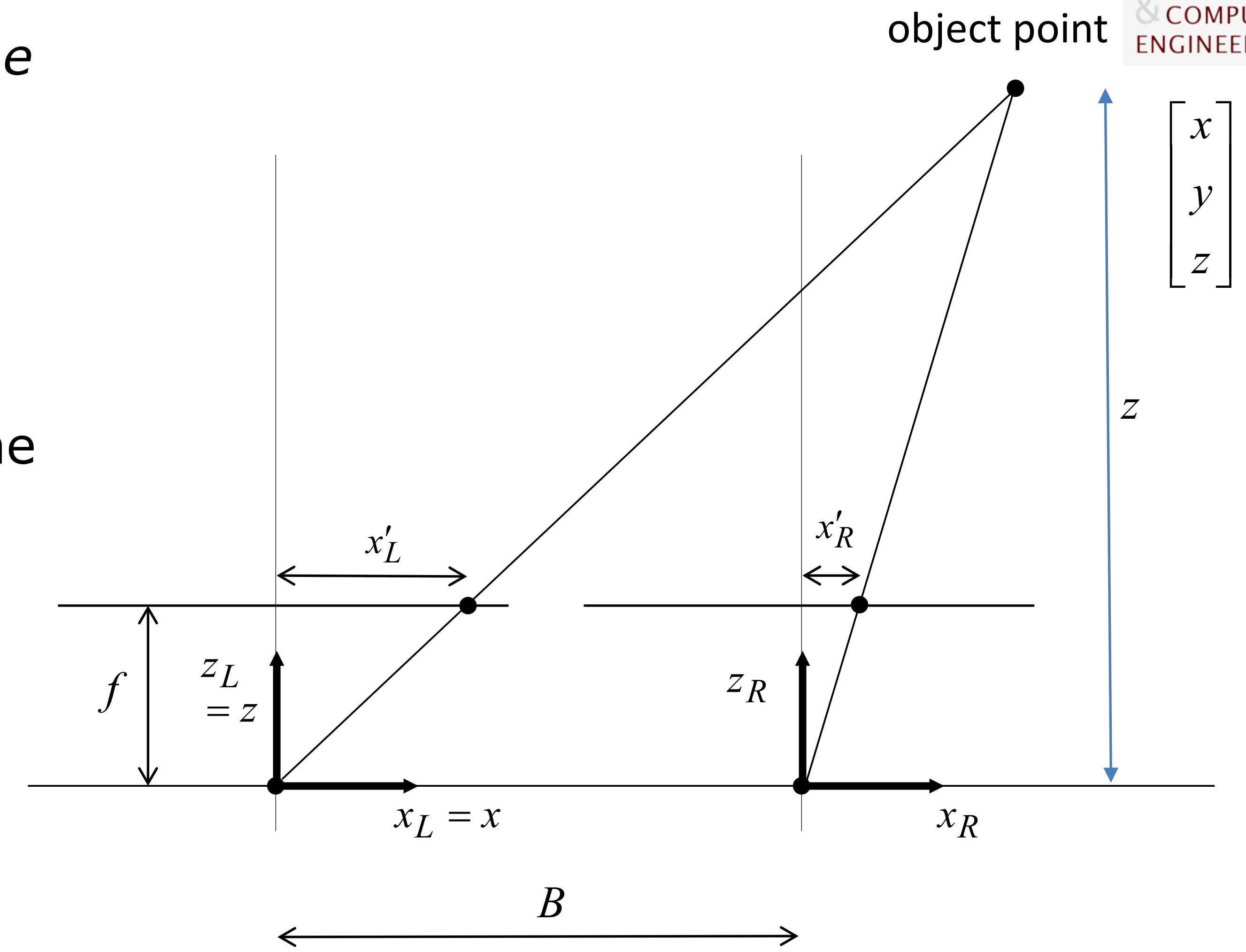


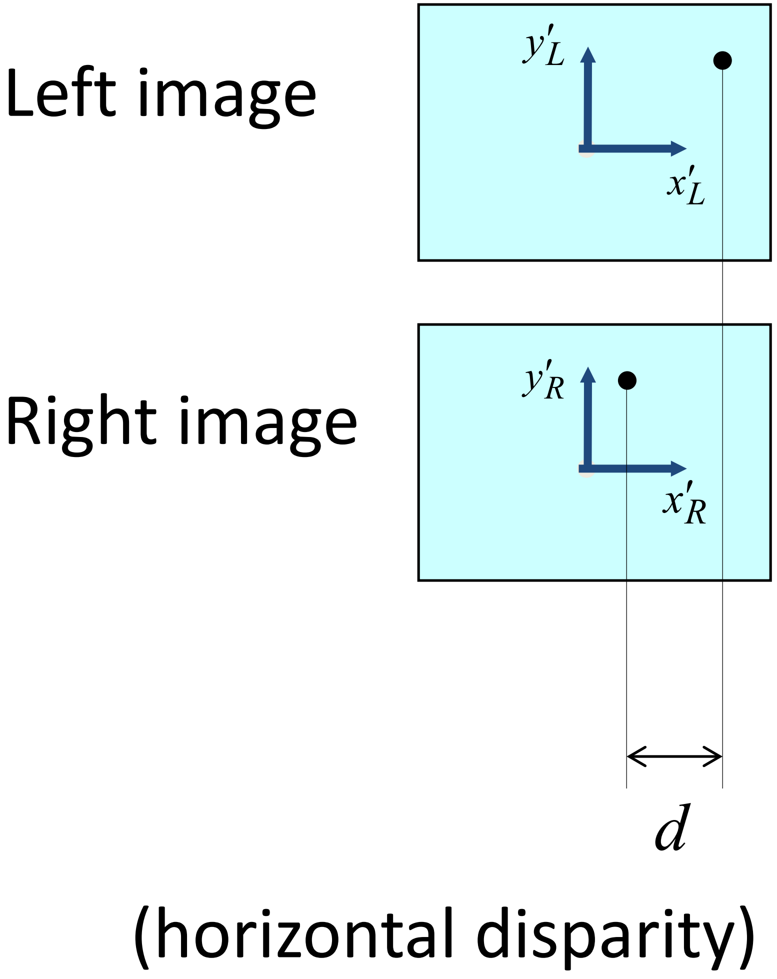
- With this simple geometry, we can use triangulation to solve for z *if the image locations are known:*

$$z = \frac{Bf}{x'_L - x'_R}$$

  - Assuming both cameras have the same optics; focal length f
- The quantity $d = x'_L - x'_R$ is called the horizontal **disparity:**

$$\boldsymbol{z = \frac{Bf}{d}}$$





- Disparity is the distance between corresponding points when the 2 images are superimposed
  - **Horizontal disparity**

$$x'_L - x'_R$$

  - **Vertical disparity**

$$y'_L - y'_R$$

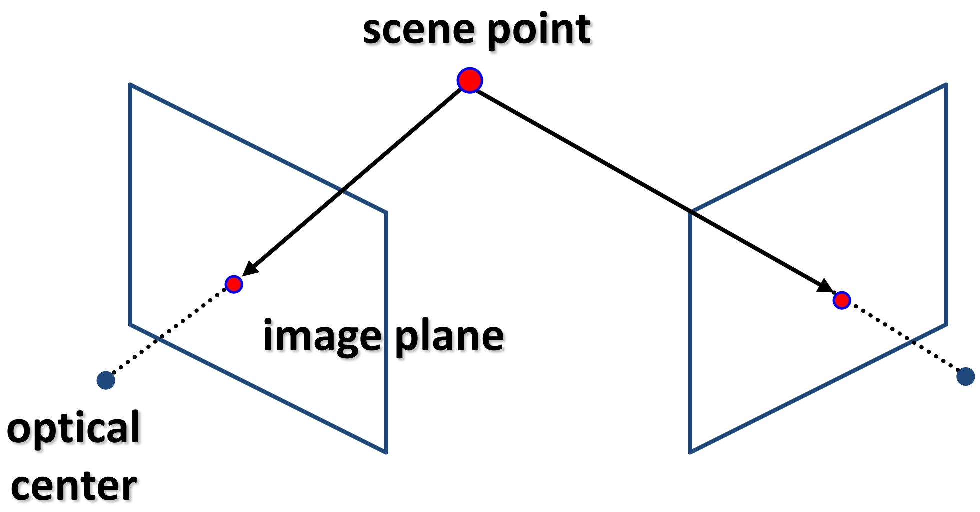
- With this parallel-axis imaging geometry, we can also show that

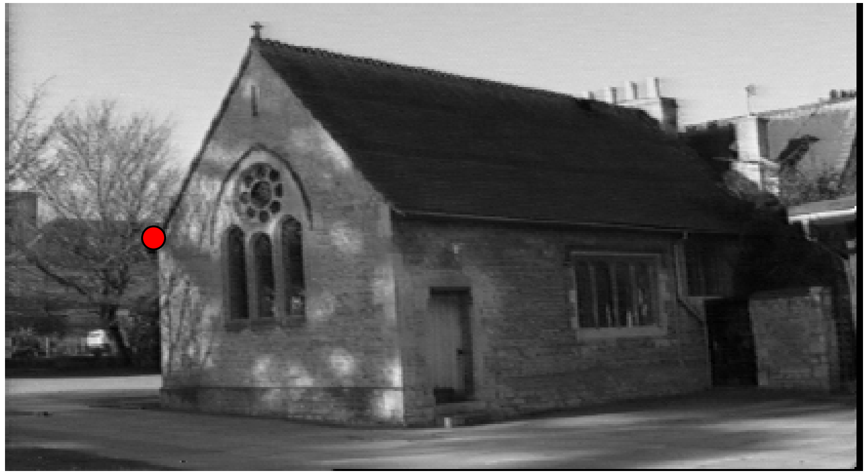
$$y'_L = y'_R$$

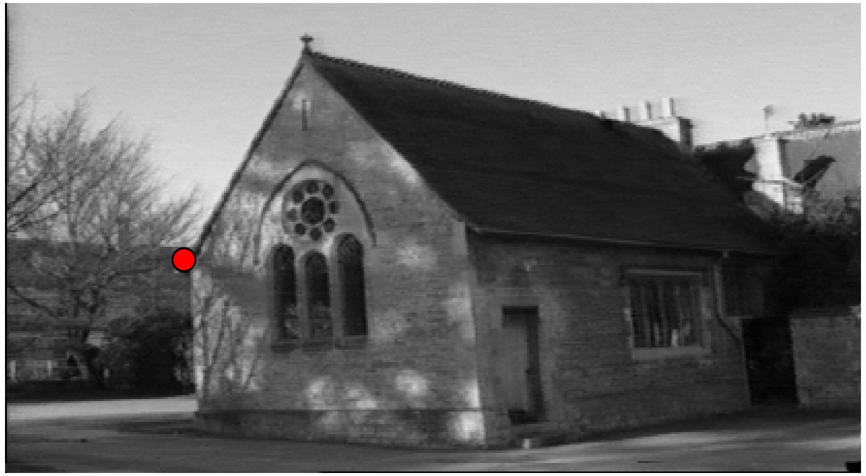
- Therefore, the complete equation for *stereo backprojection* in this case is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \frac{B}{d} \begin{bmatrix} x'_L \\ y'_L \\ f \end{bmatrix}$$

Estimating depth with stereo; we can produce a *disparity map* for the entire scene







- Remember that distance to the object (or *range*) is inversely proportional to disparity

Credit: Grauman (adapted)

Example disparity map

Image $I_L(x,y)$

Disparity map $D(x,y)$

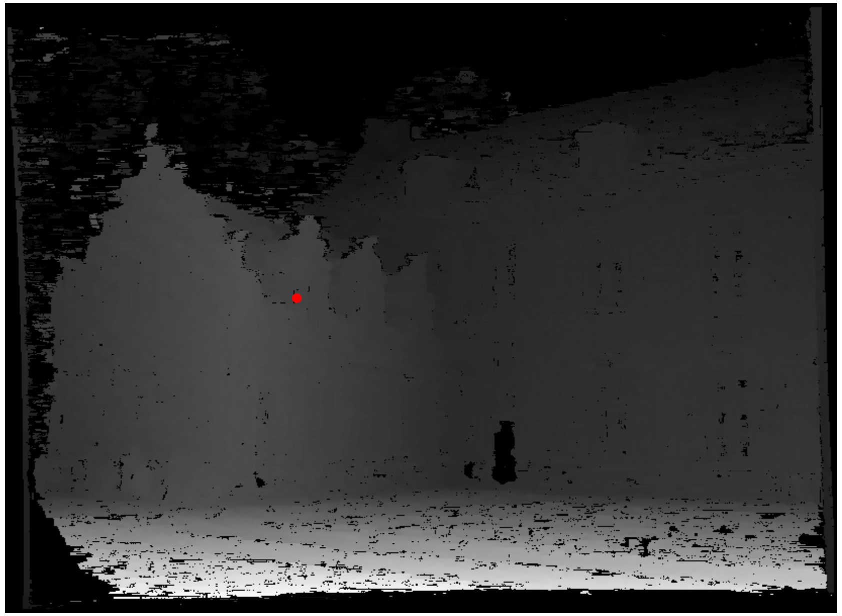


Image $I_R(x',y')$

- Remember that range is inversely proportional to disparity, so bright areas in the disparity image are closer to the cameras

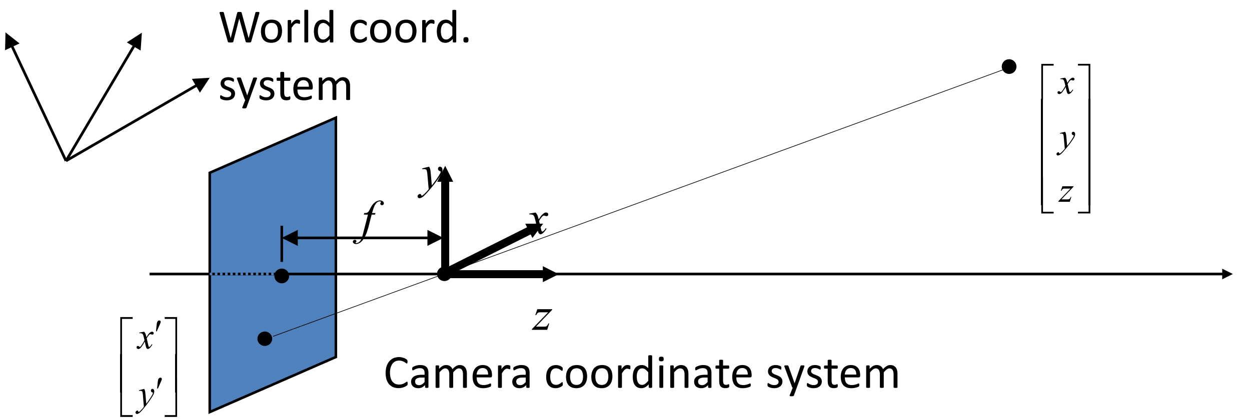
Credit: Grauman

Stereo range estimation is easy, in principle

- If the pose of each camera is known, and if 2D point correspondences are known, then the associated 3D point locations can be found using triangulation

- Two fundamental issues:
  - **Camera calibration**
  - **The "correspondence problem"**

Camera calibration (reminder)



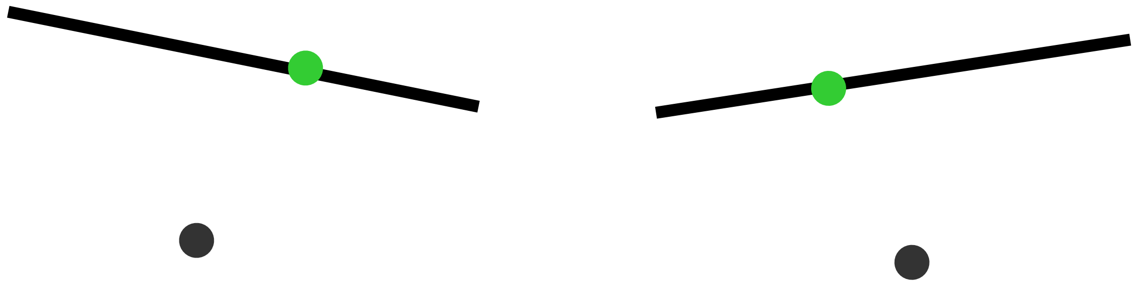
- Determine the following:
  - *Intrinsic* parameters: focal length,
    pixel sizes (mm), location of image center, radial lens distortion parameters
  - *Extrinsic* parameters: rotation matrix and translation vector, relative to a reference coordinate system
- For now, let's assume that these parameters are known

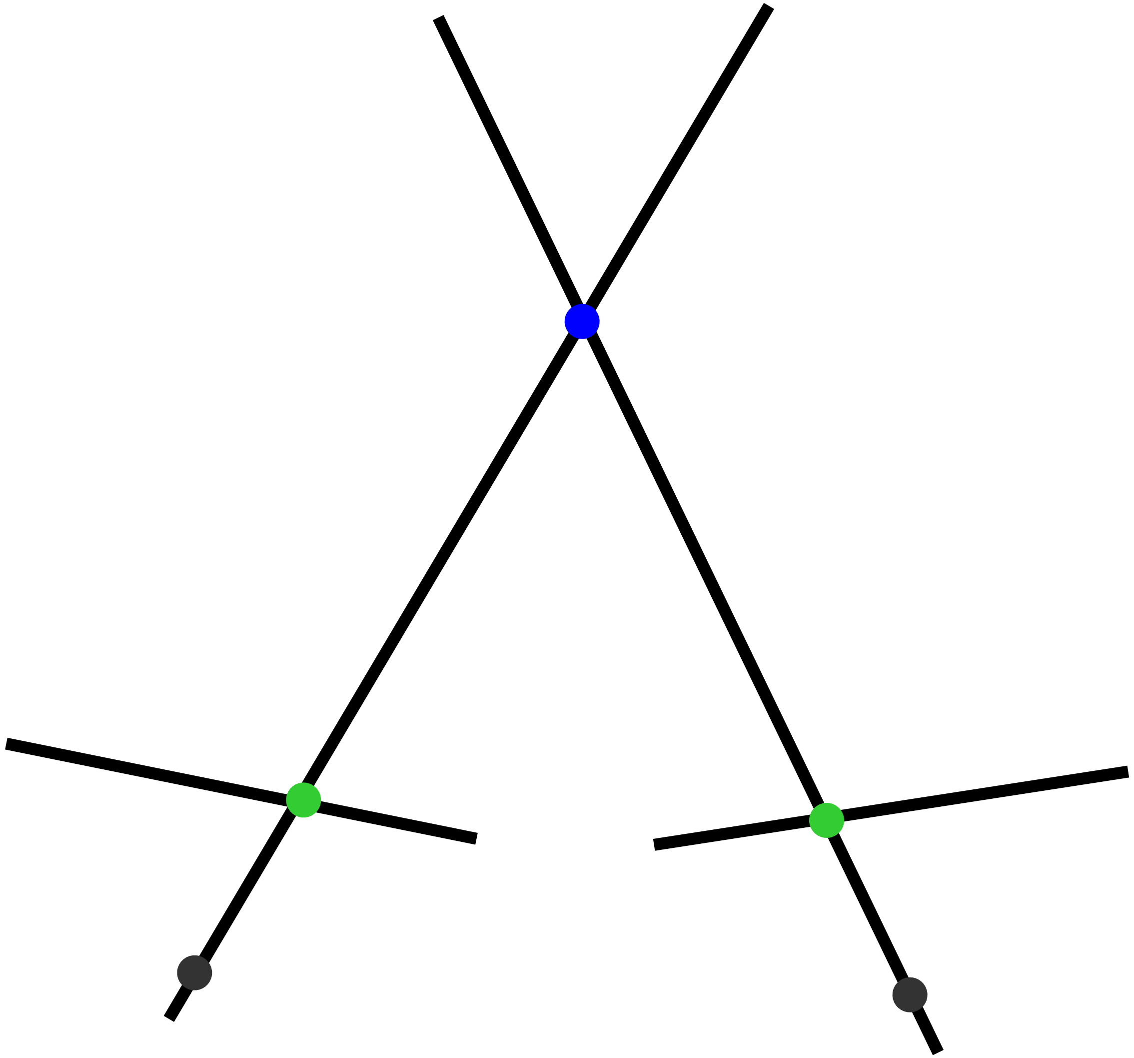
The correspondence problem

- Determine which points in one image correspond to points in the other image
- When a 3D point (x, y, z) projects onto 2 images, these image locations are called
  - **corresponding points**, or
  - **matching points**, or
  - a **stereo pair**, or
  - a **conjugate pair**

Why is it hard to identify correspondences?

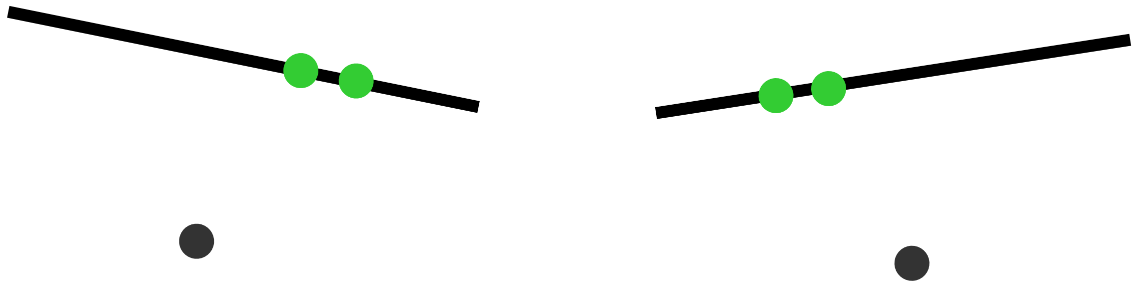
Consider a simple case:
only one feature point per image



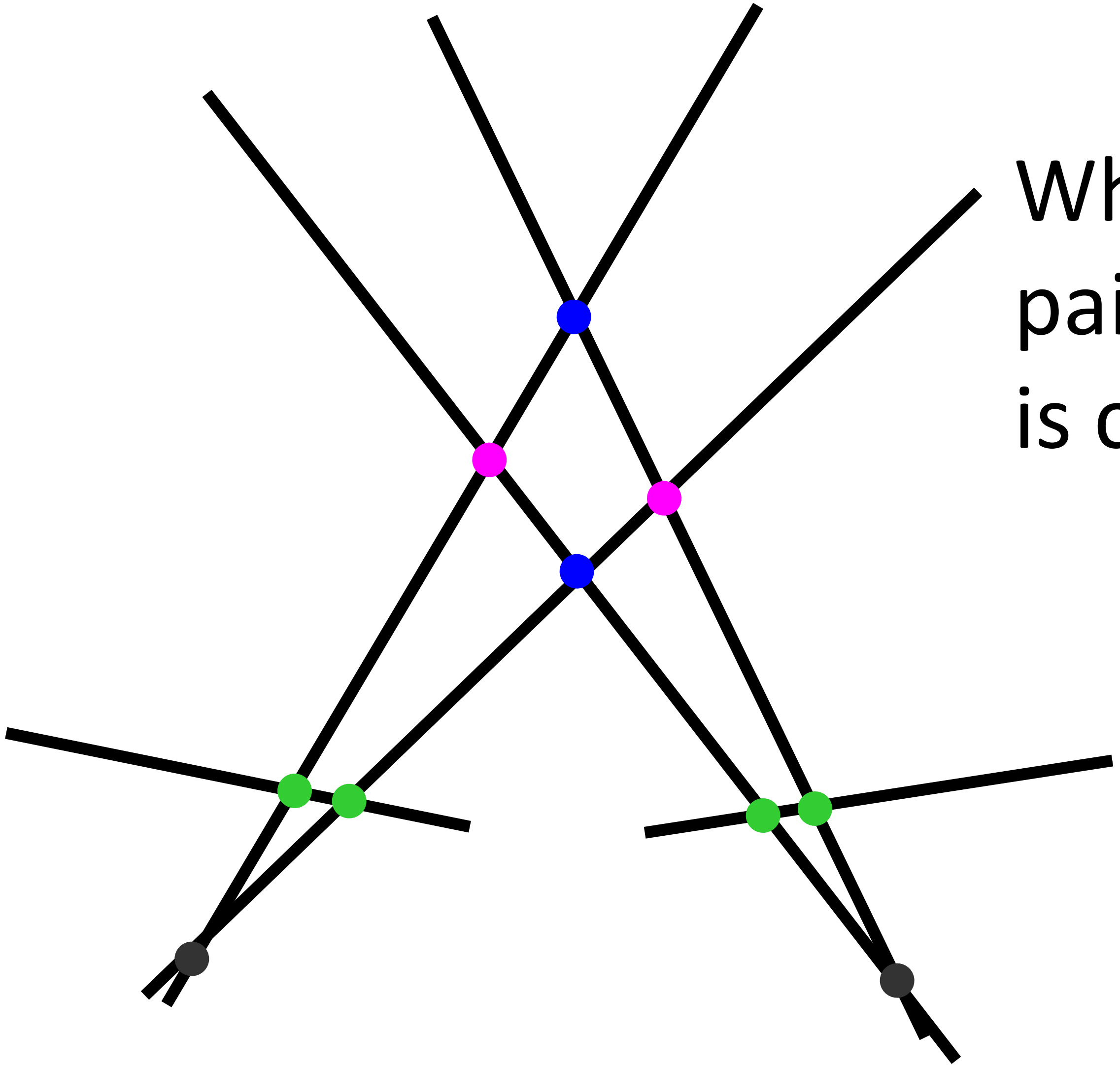


left camera right camera

Now consider 2 points per image



left camera right camera

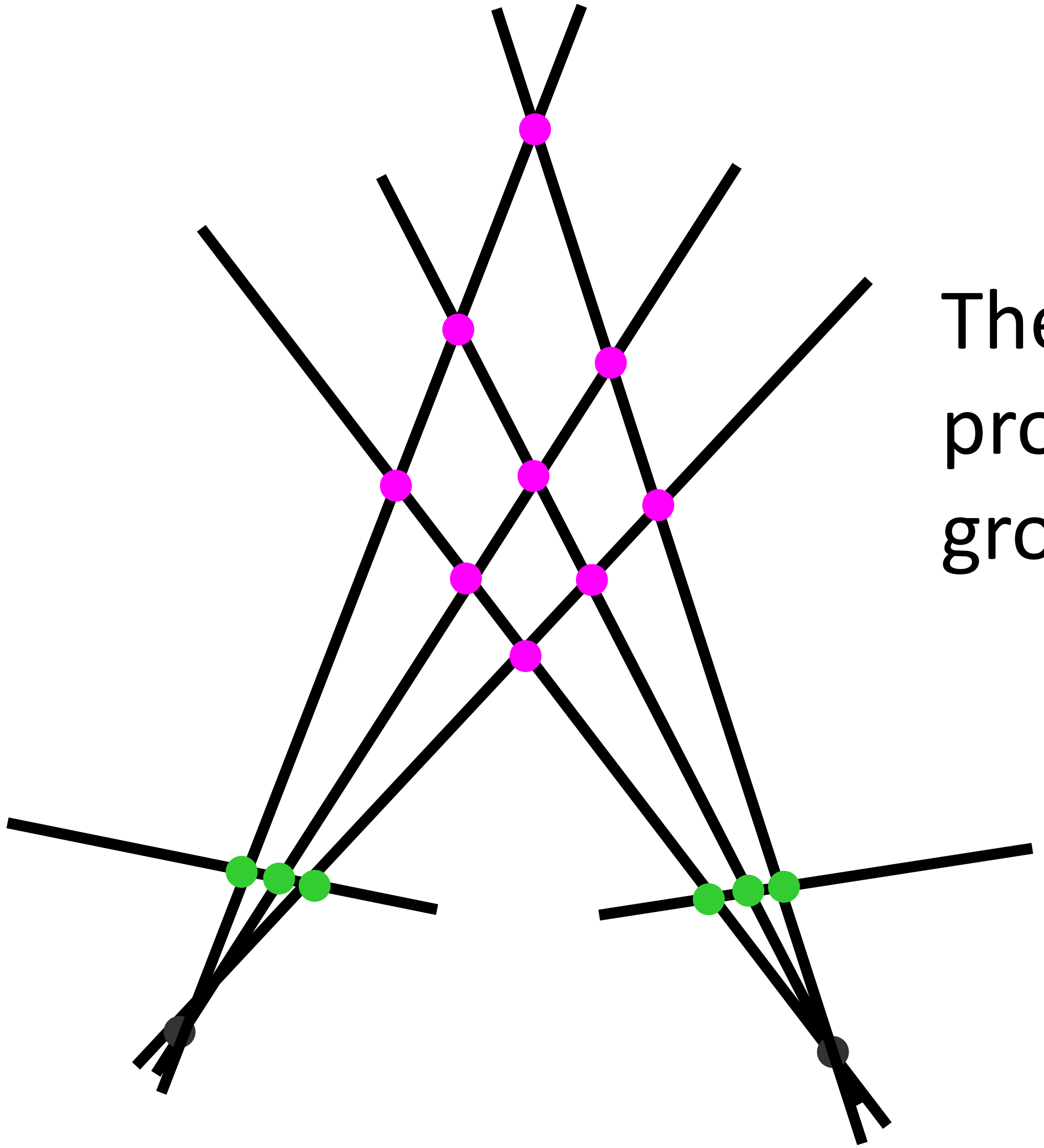


Which pairing is correct?

left camera

right camera

The problem grows …



left camera right camera

The correspondence problem highlights the need for constraints in stereo vision

- The correspondence problem is difficult (mathematically "ill-posed")
- *Yet biological vision performs very well!*
- Some common-sense constraints are possible:
  - Most surfaces of interest are opaque
  - Most surfaces are smooth, and discontinuities are relatively rare
  - Initial estimates are available

- An important geometric constraint is possible, too . . . .

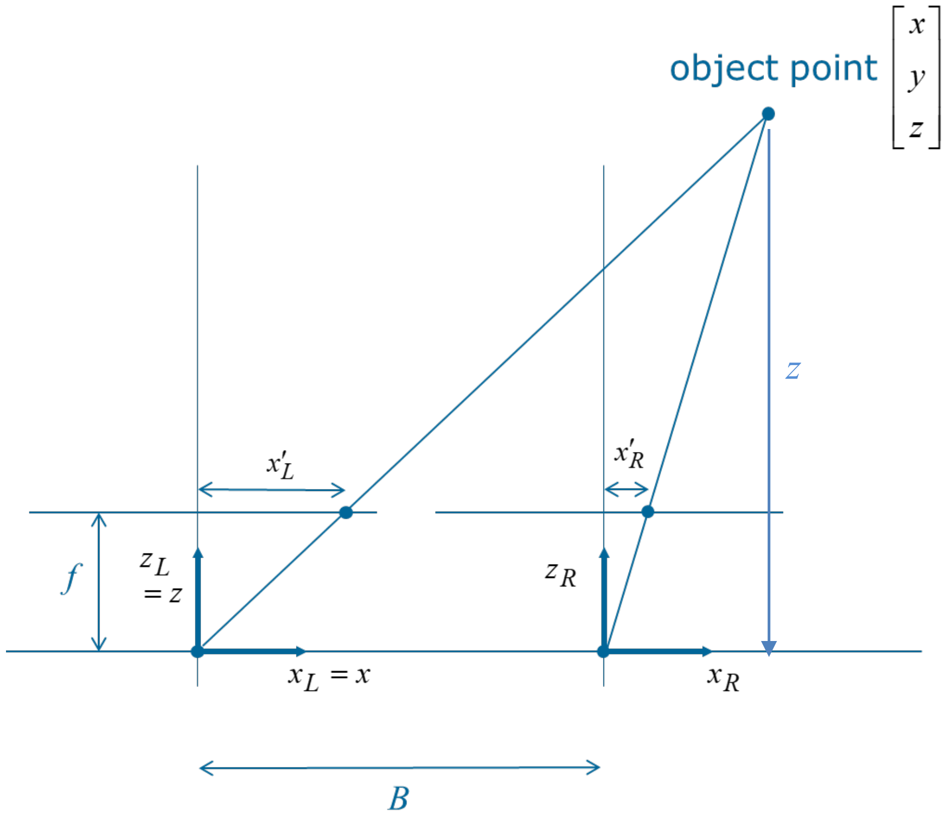
AN EXAMPLE OF STEREO DISPARITY CALCULATION

The simplest stereo geometry is the *standard rectified* geometry – obtained either by a careful optic setup or by rectifying the more general stereo images

- In either case, the images are parallel to the baseline, and all image edges are parallel to each other
  - Cameras are looking straight ahead, essentially
- This geometry allows the simplest relationship between distance and disparity

$$z = \frac{fB}{d}$$

- Consider units in this equation!



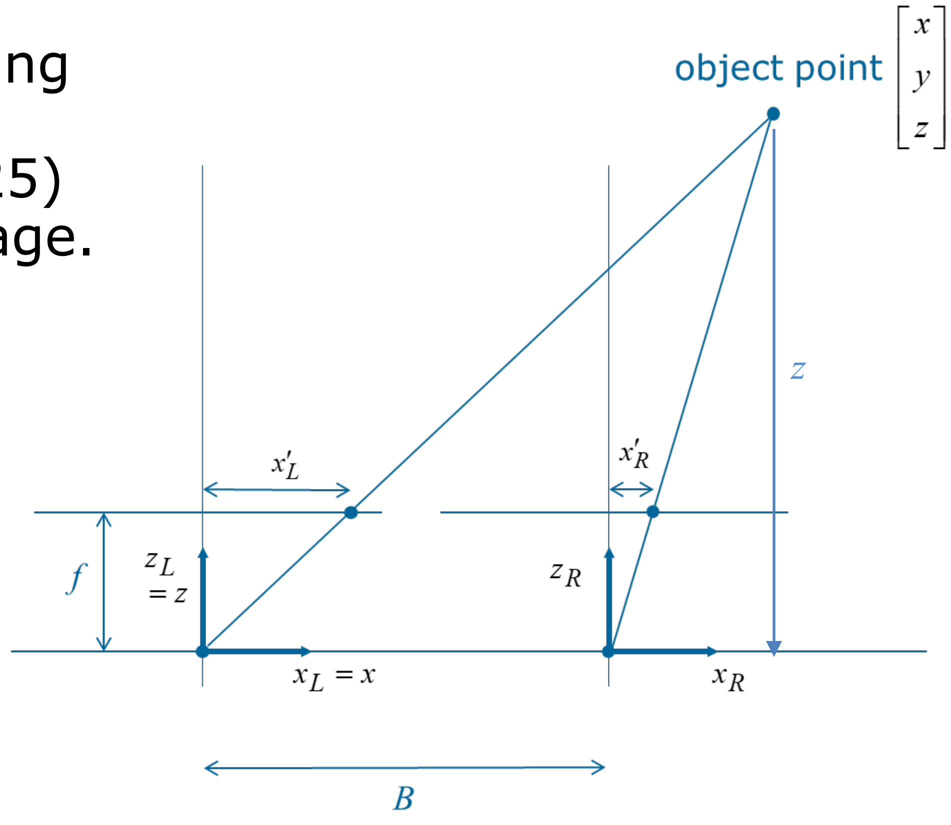
Since we want f expressed in pixels, we need an additional calculation involving the pixel spacing for the sensor

Consider the case of two cameras $10cm$ apart, imaging the same point in space: the lens having a $50mm$ focal length, and the pixel pitch of the sensor is $5\mu m$. The point is at location (100, 125) in the left image and (40, 125) in the right image.

$$d = 100 - 40 = 60\ pix$$

$$f = \frac{50\ mm}{5\ \mu m/_{pix}} = \frac{50 \times 10^{-3}\ m}{5 \times 10^{-6}\ m/_{pix}} = 10{,}000\ pix$$

$$z = \frac{fB}{d} = \frac{10{,}000\ pix\ (0.1\ m)}{60\ pix} = 16.67\ m$$



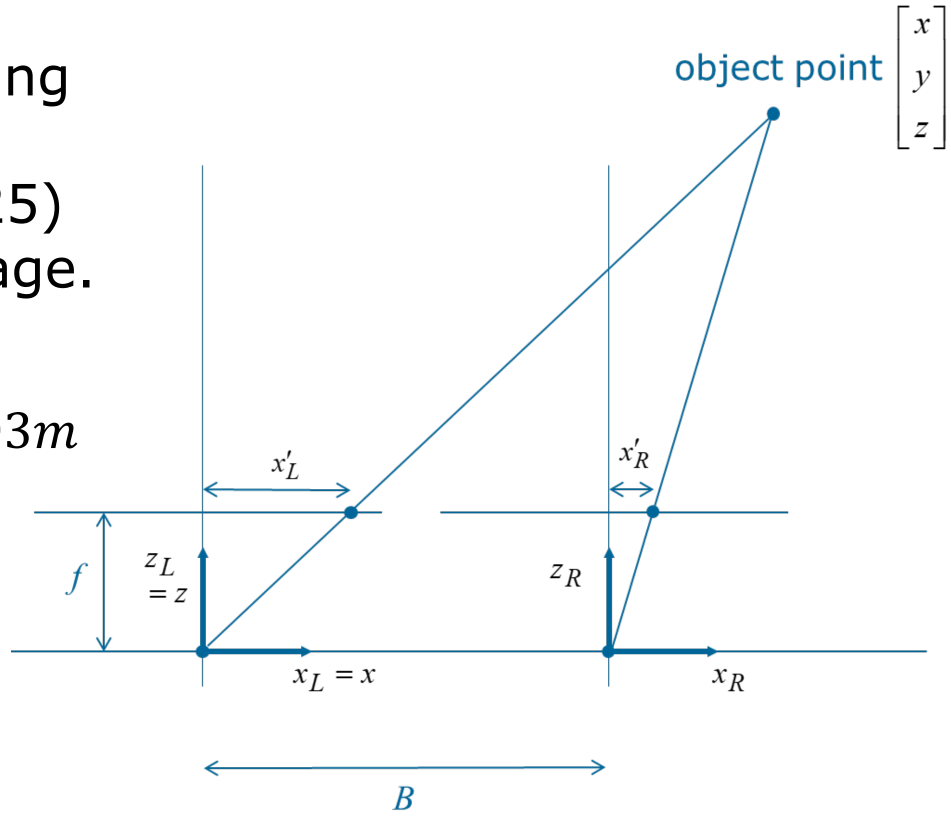
Alternatively, we can convert the disparity to real-world units at the imager and do the calculation in real-world units

Consider the case of two cameras $10cm$ apart, imaging the same point in space: the lens having a $50mm$ focal length, and the pixel pitch of the sensor is $5\mu m$. The point is at location (100, 125) in the left image and (40, 125) in the right image.

$$d = 100 - 40 = 60\, pix \left({}^{5\ \mu m}/_{pix} \right) = 300\mu m = 0.0003m$$

$$f = 50\, mm$$

$$z = \frac{fB}{d} = \frac{0.05m(0.1\ m)}{0.0003m} = 16.67\ m$$



Today’s Objectives

Stereo Vision

- The development of stereo vision
- Binocular imaging
- Disparity
- The correspondence problem
- An example of stereo disparity calculation

- Thanks to Dr. A. L. Abbott for many of the following slides