THE UNIVERSITY OF BRITISH COLUMBIA

CPSC 425: Computer Vision

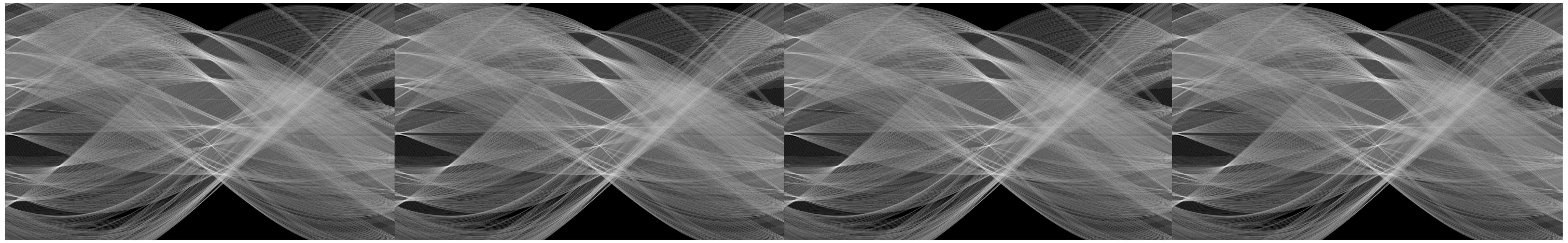


Image Credit: Ioannis (Yannis) Gkioulekas (CMU)

Lecture 23: Stereo Vision

Menu for Today (**October 31, 2018**)

Topics:

- Stereo Vision
- iClicker Quiz
- Epipolar Constraint
- Stereo Correspondences

Redings:

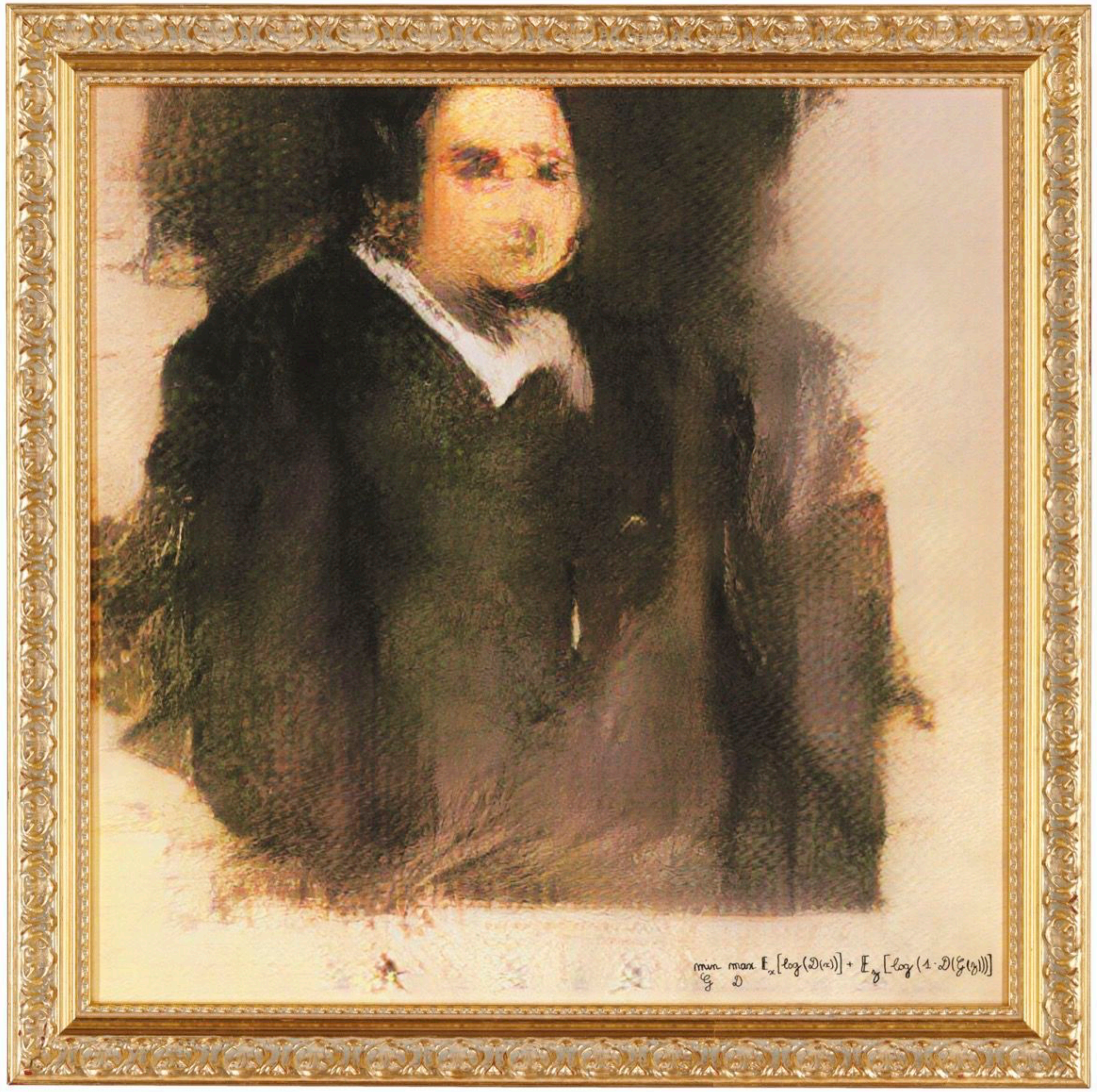
- **Today's** Lecture: Forsyth & Ponce (2nd ed.) 7.1.1, 7.2.1, 7.4, 7.6
- **Next** Lecture: None

Reminders:

- **Assignment 3**: Texture Syntheis is **due today**
- **Assignment 4**: Local Invariant Features and RANSAC

Today's “**fun**” Example: AI Generated Portrait

Sold two days ago for $432,500 at British auction house



Lecture 22: Re-cap

The **Hough transform** is another technique for fitting data to a model

— a voting procedure

— possible model parameters define a quantized accumulator array

— data points “vote" for compatible entries in the accumulator array

A key is to have each data point (token) constrain model parameters as tightly as possible

Please get your **iClickers** — Quiz

Stereo Vision

Problem Formulation:

Determine depth using two images acquired from (slightly) different viewpoints

Key Idea(s):

The 3D coordinates of each point imaged are constrained to lie along a ray. This is true also for a second image obtained from a (slightly) different viewpoint. Rays for the same point in the world intersect at the actual 3D location of that point

Stereo Vision

With two eyes, we acquire images of the world from slightly different viewpoints

We perceive **depth** based on **differences in the relative position of points** in the left image and in the right image

Binoculars

Binoculars enhance binocular depth perception in two distinct ways:

1. magnification
2. longer baseline (i.e., distance between entering light paths) compared to the normal human inter-pupillary distance

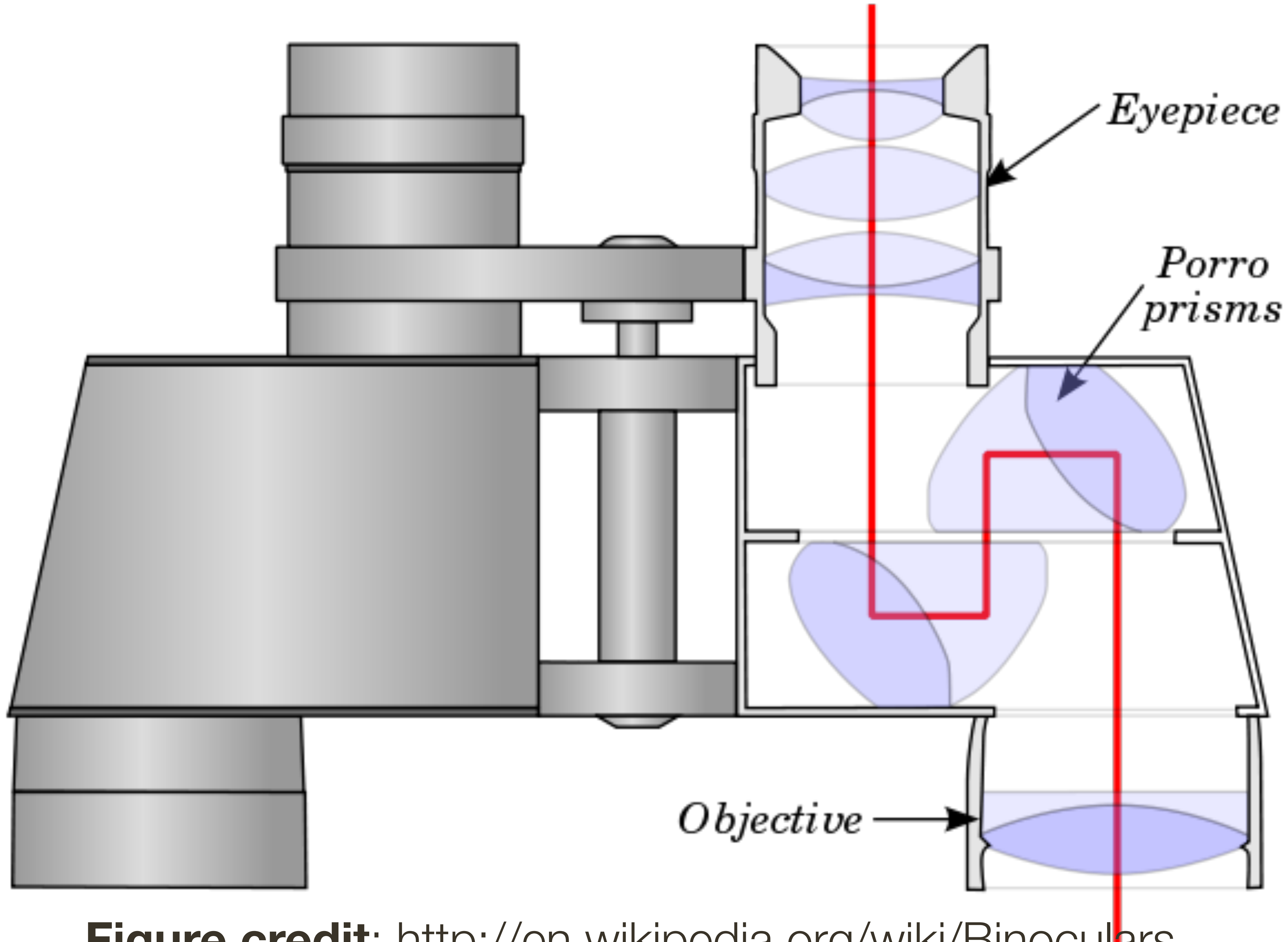


Figure credit: http://en.wikipedia.org/wiki/Binoculars

Stereo Vision

Task: Compute depth from two images acquired from (slightly) different viewpoints

Approach: “Match” locations in one image to those in another

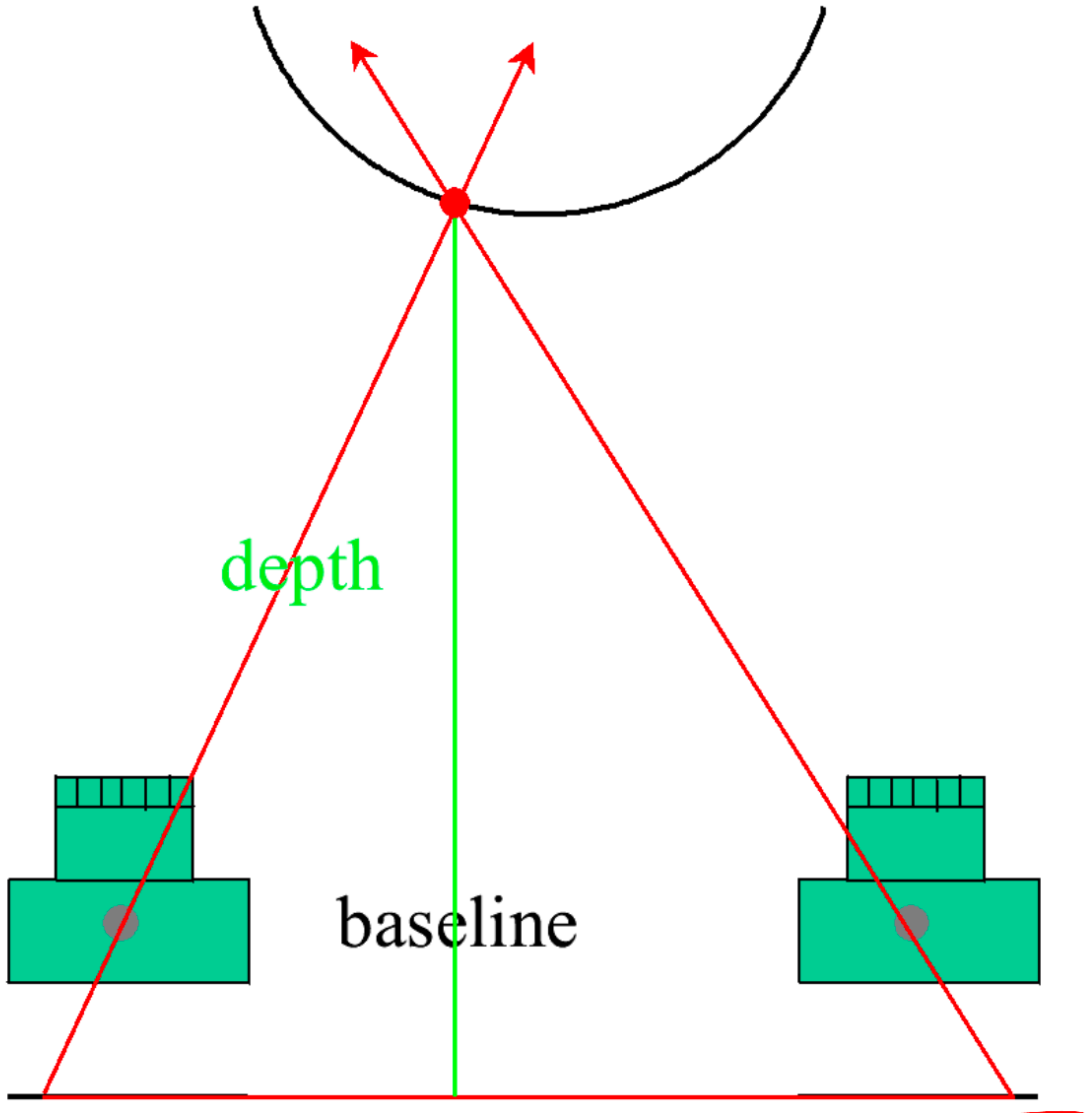
Sub-tasks:

— Calibrate cameras and camera positions

— Find all corresponding points (the hardest part)

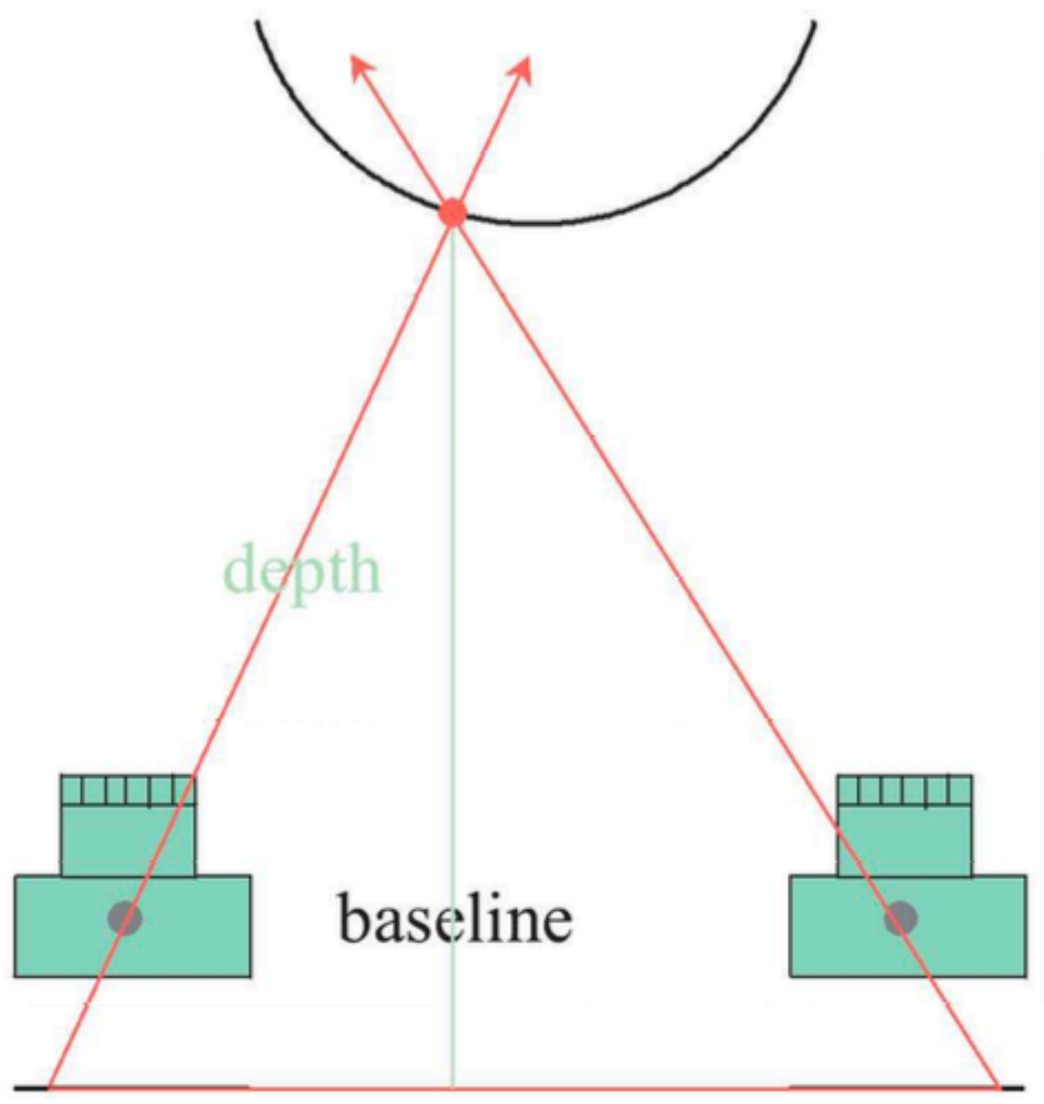
— Compute depth and surfaces

Stereo Vision

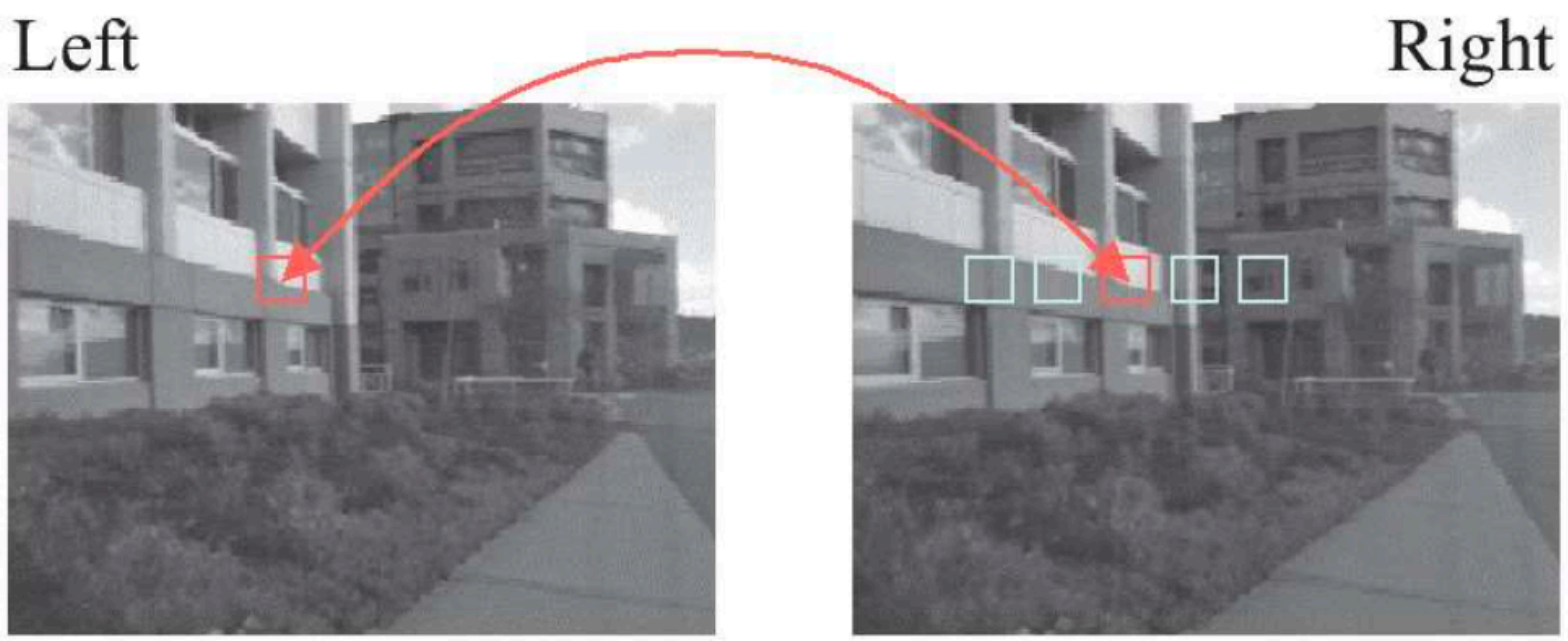


Slide credit: Trevor Darrell

Stereo Vision



Triangulate on two images of the same point



Match correlation windows across scan lines

Image credit: Point Grey Research
Slide credit: Trevor Darrell

Point Grey Research **Digiclops**

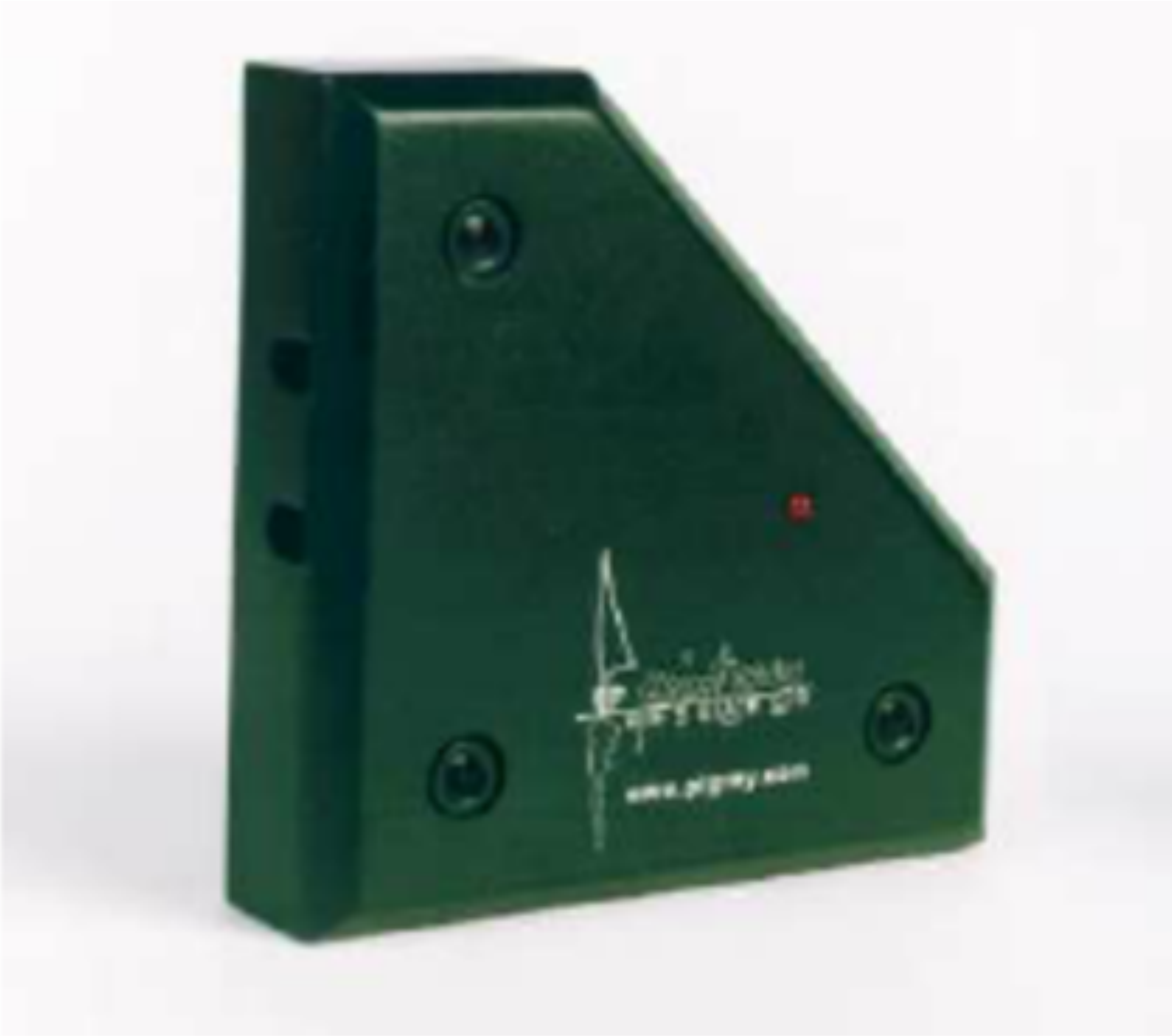
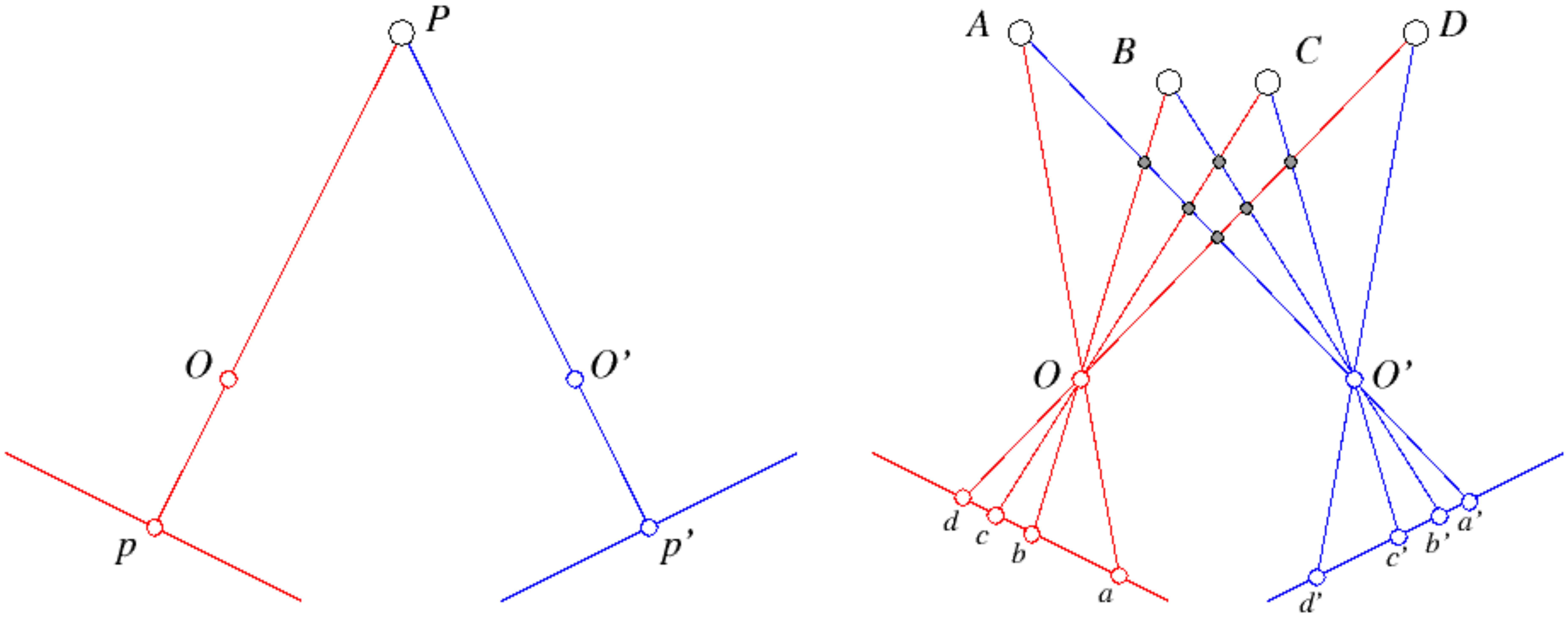


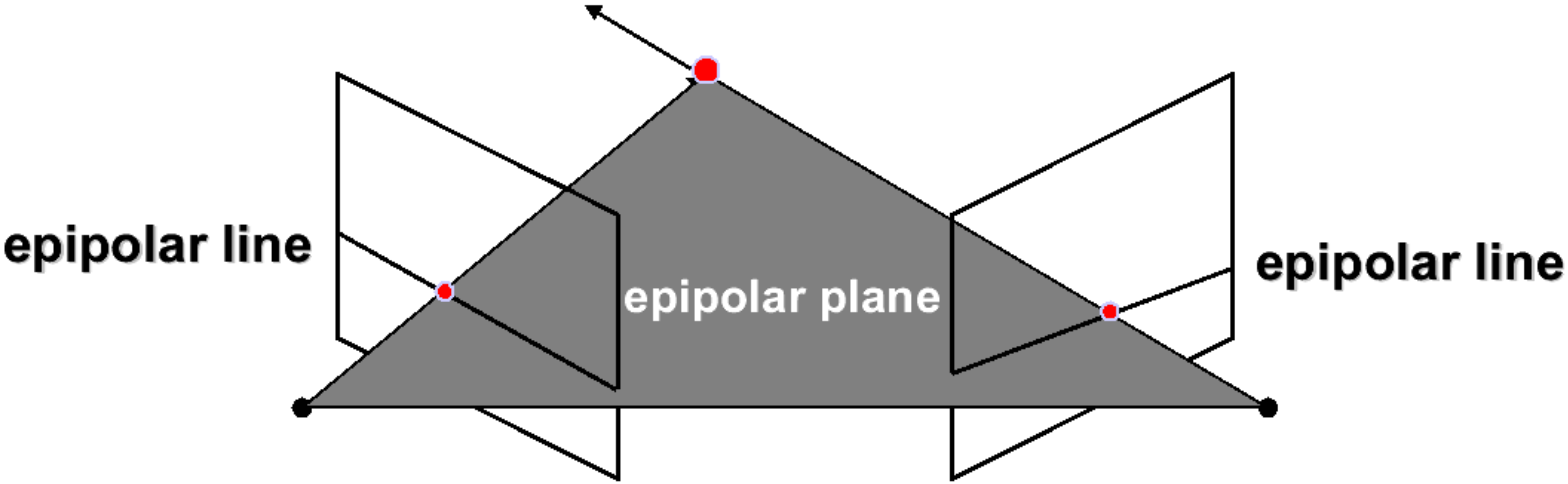
Image credit: Point Grey Research

Correspondence



Forsyth & Ponce (2nd ed.) Figure 7.2

The **Epipolar** Constraint



Matching points lie along corresponding epipolar lines

Reduces correspondence problem to 1D search along conjugate epipolar lines

Greatly reduces cost and ambiguity of matching

Slide credit: Steve Seitz

Simplest Case: **Rectified** Images

Image planes of cameras are **parallel**

Focal **points** are at same height

Focal **lengths** same

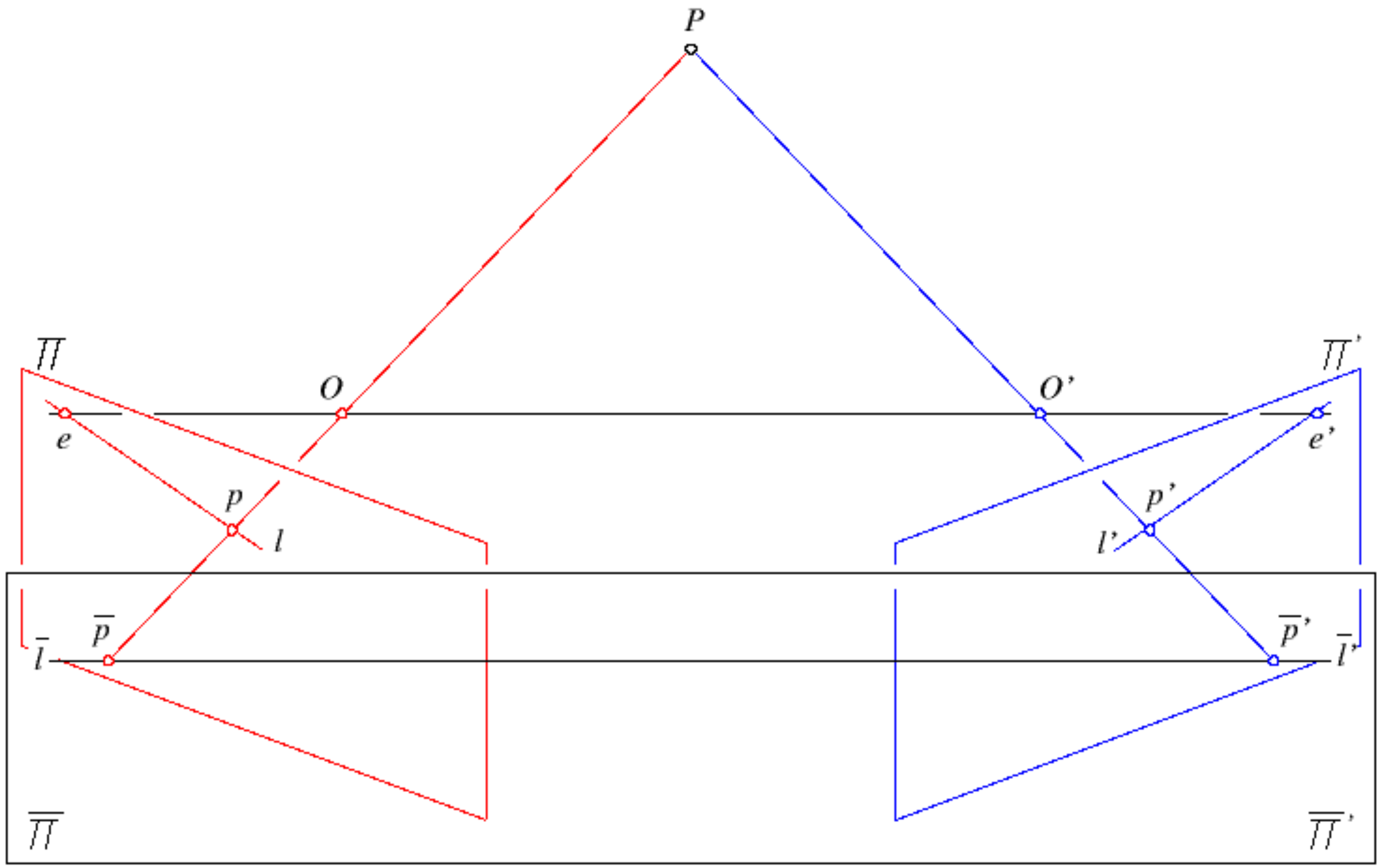
Then, **epipolar lines** fall along the **horizontal scan lines** of the images

We assume images have been **rectified** so that epipolar lines correspond to scan lines

— Simplifies algorithms

— Improves efficiency

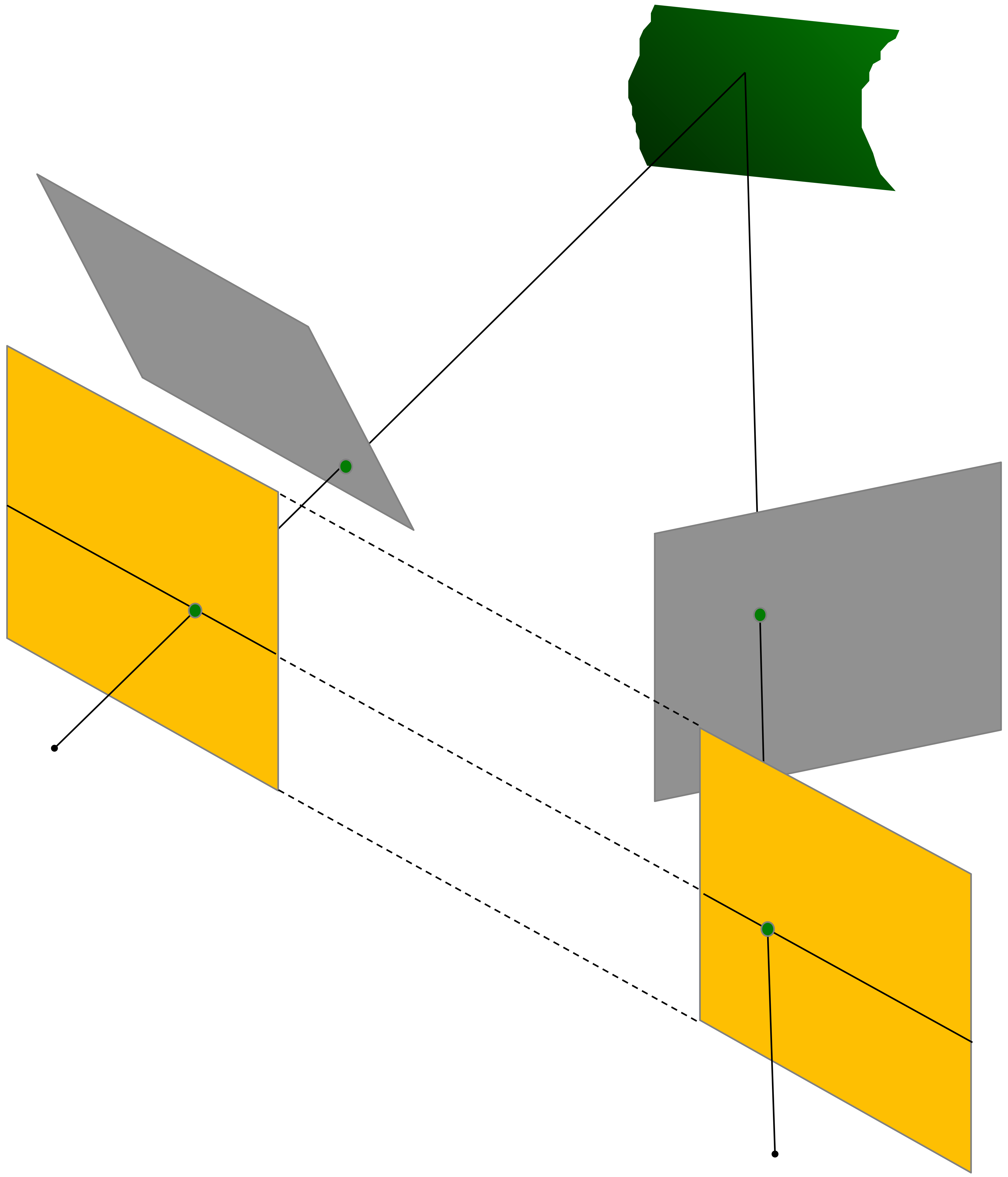
Rectified Stereo Pair



Rectified Stereo Pair

Reproject image planes onto a common plane parallel to the line between camera centers

Need two homographies (3x3 transform), one for each input image reprojection

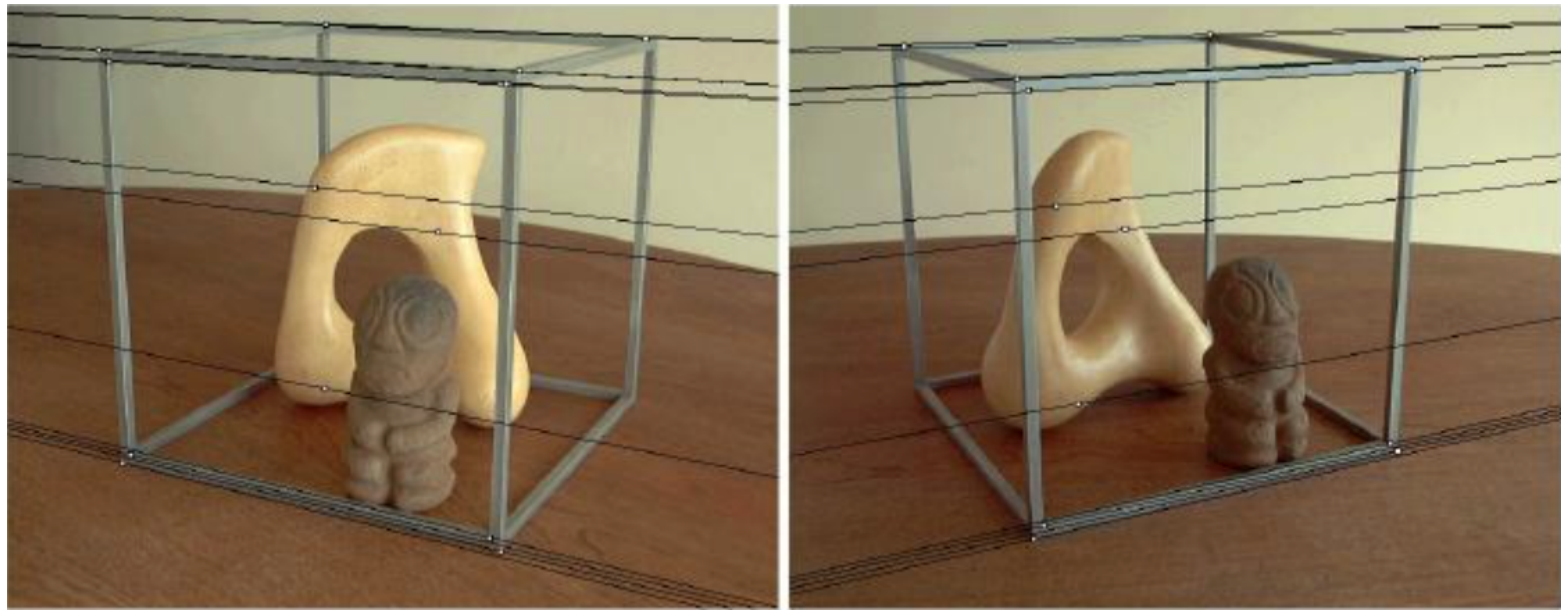


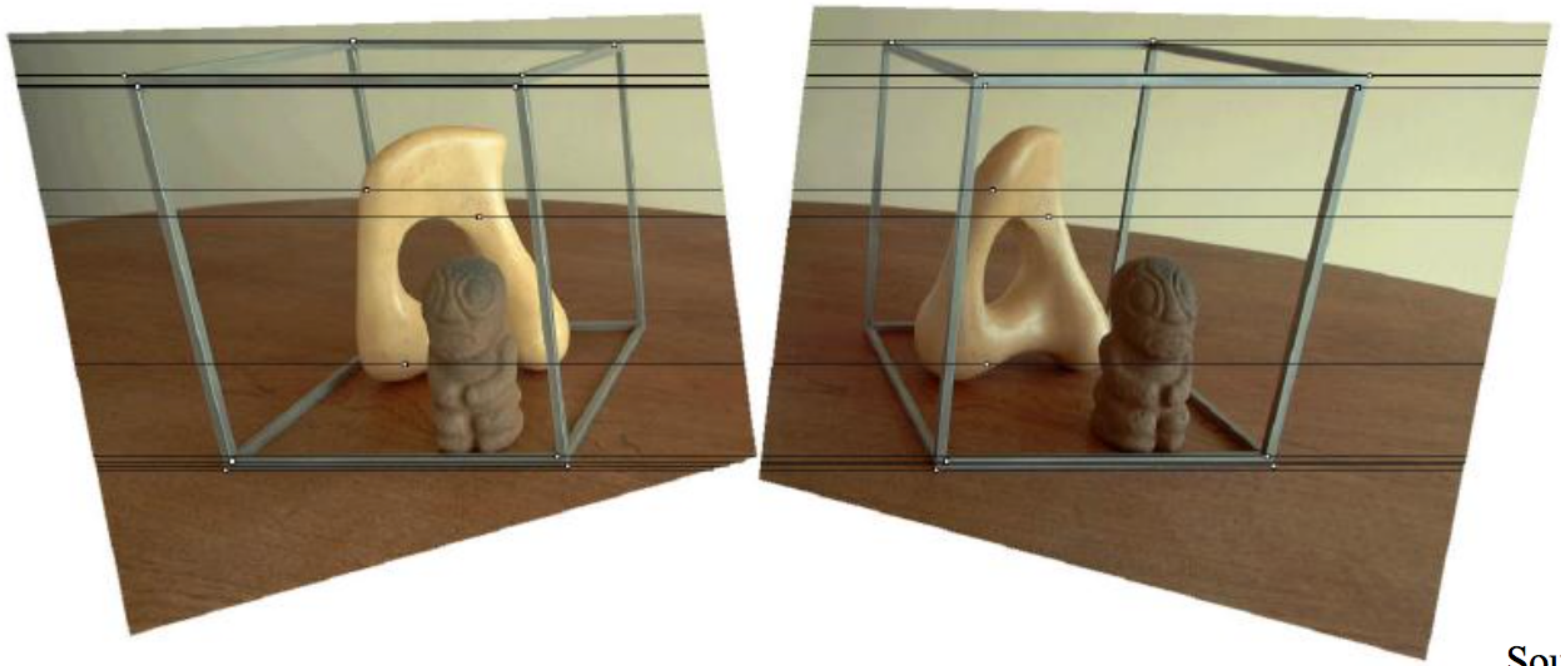
C. Loop and Z. Zhang. Computing Rectifying Homographies for Stereo Vision.Computer Vision and Pattern Recognition, 1999.

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)

Rectified Stereo Pair: Example

Before Rectification





After Rectification

Slide Credit: Ioannis (Yannis) Gkioulekas (CMU)