ENGN6528-Week09-3DVision

Slides are adapted from Steve Seize (UW), Noah Snavely (Cornell U)

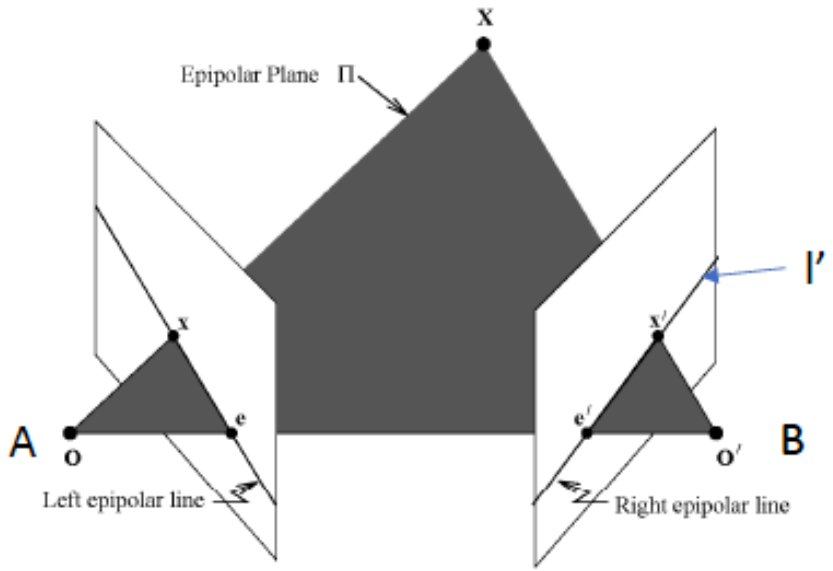
Announcements

- CLab-Assignment-3 has been released. Due date: 22nd May 2021.
- The tutorial this week is about 3D Vision.
- No Lab Session this week.

Outline

- Review
- Essential Matrix
- Triangulation
- Stereo

Review – Epipolar Geometry



- The epipolar line $\mathbf{l}'$ is the image of the ray through $\mathbf{x}$.
- The epipole $\mathbf{e}'$ is the point of intersection of the line joining the camera centres—the baseline—with the image plane.
- The epipole is also the image in one camera of the centre of the other camera.
- All epipolar lines intersect in the epipole.

Review - Fundamental Matrix

Properties of $\mathtt{F}$

- $\mathtt{F}$ is a rank 2 homogeneous matrix with 7 degrees of freedom.
- **Point correspondence**: If $\mathbf{x}$ and $\mathbf{x}'$ are corresponding image points, then $\mathbf{x}'^{\top}\mathtt{F}\mathbf{x} = 0$.
- **Epipolar lines**:
  - $\mathbf{l}' = \mathtt{F}\mathbf{x}$ is the epipolar line corresponding to $\mathbf{x}$.
  - $\mathbf{l} = \mathtt{F}^{\top}\mathbf{x}'$ is the epipolar line corresponding to $\mathbf{x}'$.
- **Epipoles**:
  - $\mathtt{F}\mathbf{e} = \mathbf{0} \qquad \mathtt{F}^{\top}\mathbf{e}' = \mathbf{0}$
- **Computation from camera matrices** $\mathtt{P}, \mathtt{P}'$:
  - $\mathtt{F} = [\mathtt{P}'\mathtt{C}]_{\times}\mathtt{P}'\mathtt{P}^{+}$, where $\mathtt{P}^{+}$ is the pseudo-inverse of $\mathtt{P}$, and $\mathtt{C}$ is the centre of the first camera. Note, $\mathbf{e}' = \mathtt{P}'\mathtt{C}$.
  - Canonical cameras, $\mathtt{P} = [\mathtt{I} \mid \mathbf{0}], \mathtt{P}' = [\mathtt{M} \mid \mathbf{m}]$, $\mathtt{F} = [\mathbf{e}']_{\times}\mathtt{M} = \mathtt{M}^{-\top}[\mathbf{e}]_{\times}$, where $\mathbf{e}' = \mathbf{m}$ and $\mathbf{e} = \mathtt{M}^{-1}\mathbf{m}$.

Computation of the Fundamental Matrix

Basic equations

Given a correspondence

$$\mathbf{x} \leftrightarrow \mathbf{x}'$$

The basic incidence relation is

$$\mathbf{x}'^{\top}\mathbf{F}\mathbf{x} = 0$$

May be written

$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0 \quad .$$

Single point equation - Fundamental matrix

Gives an equation :

$$(x'x, x'y, x', y'x, y'y, y', x, y, 1) \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{pmatrix} = 0$$

where

$$\mathbf{f} = (f_{11}, f_{12}, f_{13}, f_{21}, f_{22}, f_{23}, f_{31}, f_{32}, f_{33})^\top$$

holds the entries of the Fundamental matrix

Total set of equations

$$\mathtt{Af} = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{pmatrix} = \mathbf{0}$$

Solving the Equations

- Solution is determined up to scale only.
- Need 8 equations $\Rightarrow$ 8 points
- 8 points $\Rightarrow$ unique solution
- > 8 points $\Rightarrow$ least-squares solution.

for now F has 8 degrees of freedom

Least-squares solution

(i) Form equations $\mathtt{A}\mathbf{f} = \mathbf{0}$.

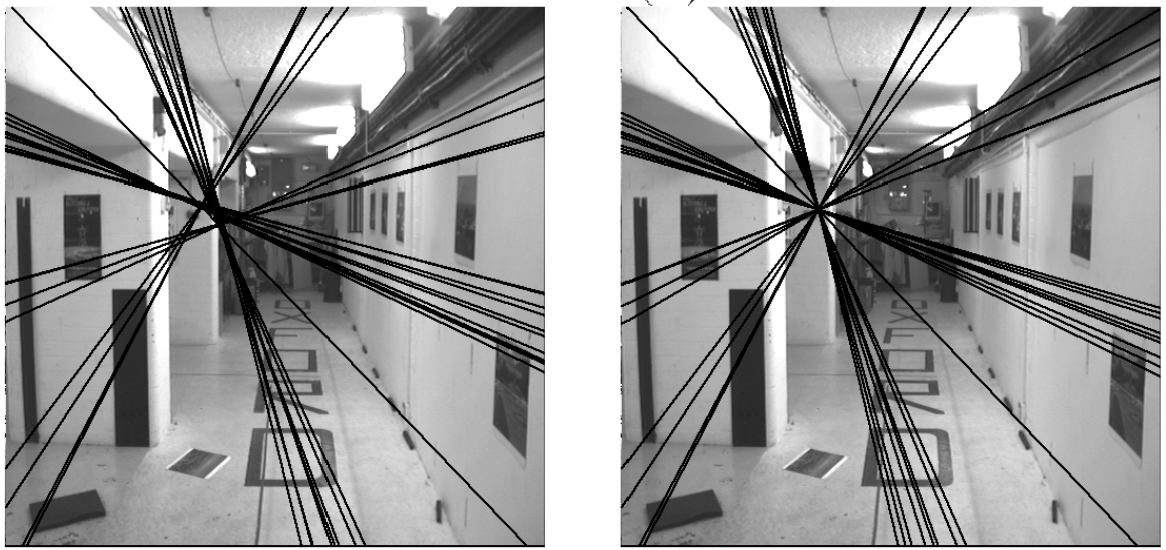
(ii) Take SVD : $\mathtt{A} = \mathtt{U}\mathtt{D}\mathtt{V}^\top$.

(iii) Solution is last column of $\mathtt{V}$ (corresp : smallest singular value)

(iv) Minimizes $||\mathtt{A}\mathbf{f}||$ subject to $||\mathbf{f}|| = 1$.

The singularity constraint

Fundamental matrix has rank 2 : $\det(\mathbf{F}) = 0$.



Left : Uncorrected $\mathbf{F}$ – epipolar lines are not coincident.

Right : Epipolar lines from corrected $\mathbf{F}$.

Correcting $\mathtt{F}$ using the Singular Value Decomposition

If $\mathtt{F}$ is computed linearly from 8 or more correspondences, singularity condition does not hold.

SVD Method

(i) SVD : $\mathtt{F} = \mathtt{U}\mathtt{D}\mathtt{V}^\top$

(ii) $\mathtt{U}$ and $\mathtt{V}$ are orthogonal, $\mathtt{D} = \mathrm{diag}(r, s, t)$.

(iii) $r \geq s \geq t$.

(iv) Set $\mathtt{F}' = \mathtt{U}\,\mathrm{diag}(r, s, 0)\,\mathtt{V}^\top$.

(v) Resulting $\mathtt{F}'$ is singular.

(vi) Minimizes the Frobenius norm of $\mathtt{F} - \mathtt{F}'$

(vii) $\mathtt{F}'$ is the "closest" singular matrix to $\mathtt{F}$.

Complete 8-point algorithm

8 point algorithm has two steps :

(i) **Linear solution.** Solve $\mathtt{A}\mathtt{f} = 0$ to find $\mathtt{F}$.

(ii) **Constraint enforcement.** Replace $\mathtt{F}$ by $\mathtt{F}'$.

Warning This algorithm is unstable and should never be used with unnormalized data (see next slide).

The normalized 8-point algorithm

Raw 8-point algorithm performs badly in presence of noise.

Normalization of data

- 8-point algorithm is sensitive to origin of coordinates and scale.
- Data must be translated and scaled to "canonical" coordinate frame.
- Normalizing transformation is applied to both images.
- Translate so centroid is at origin
- Scale so that RMS distance of points from origin is $\sqrt{2}$.
- "Average point" is $(1, 1, 1)^\top$.

Normalized 8-point algorithm

(i) **Normalization:** Transform the image coordinates :

$$\hat{\mathbf{x}}_i = \mathtt{T}\mathbf{x}_i$$
$$\hat{\mathbf{x}}'_i = \mathtt{T}'\mathbf{x}'_i$$

(ii) **Solution:** Compute $\mathtt{F}$ from the matches $\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i$

$$\hat{\mathbf{x}}_i'^{\top}\widehat{\mathtt{F}}\hat{\mathbf{x}}_i = 0$$

(iii) **Singularity constraint :** Find closest singular $\widehat{\mathtt{F}}'$ to $\widehat{\mathtt{F}}$.

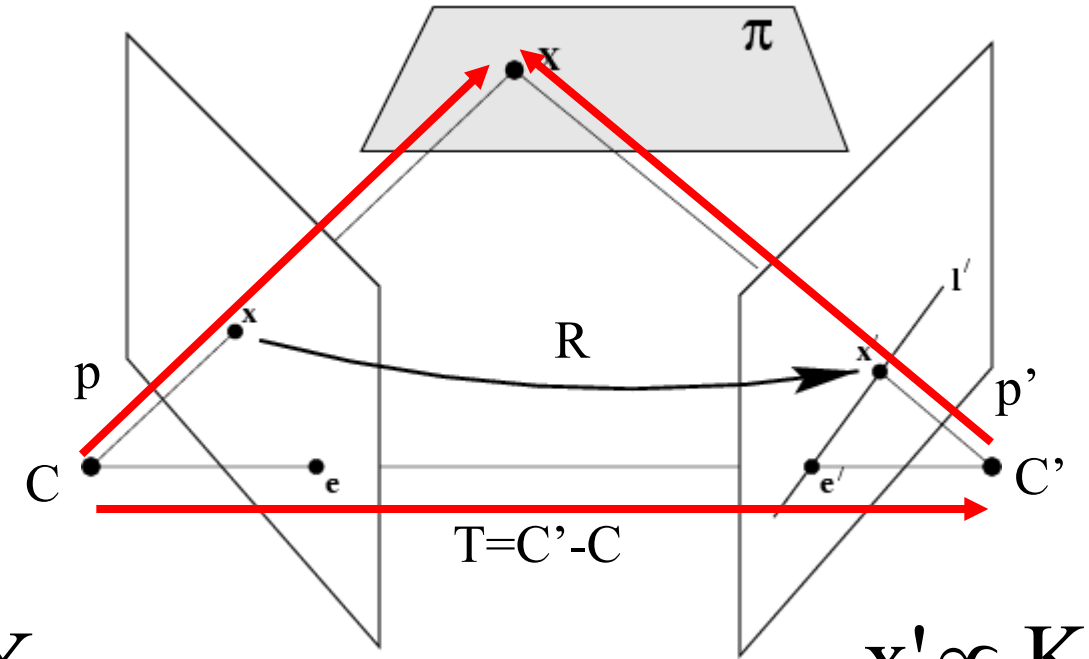
(iv) **Denormalization:** $\mathtt{F} = \mathtt{T}'^{\top}\widehat{\mathtt{F}}'\mathtt{T}$.

Essential Matrix & Fundamental Matrix

- Reading Chapter 9.6 in Multiple-view Geometry in Computer Vision

p: 3 x 1(射线)
K: 3 x 3

The Essential matrix E



$$x \propto KX \qquad x' \propto K'R(X-T)$$

$$p = K^{-1}x \propto X \qquad p' = K'^{-1}x' \propto R(X-T)$$

The equation of the epipolar plane through X is

p'定义在其相机坐标空间中

$$(\mathbf{X}-\mathbf{T})^{\mathrm{T}}(\mathbf{T}\times\mathbf{p})=0 \Rightarrow (\mathbf{R}^{\mathrm{T}}\mathbf{p}')^{\mathrm{T}}(\mathbf{T}\times\mathbf{p})=0$$

p'在world坐标空间中的表达

The Essential Matrix E

$$(\mathbf{R}^{\mathrm{T}}\mathbf{p}')^{\mathrm{T}}(\mathbf{T}\times\mathbf{p})=0$$

$$\mathbf{T}\times\mathbf{p}=\mathbf{S}\mathbf{p}$$

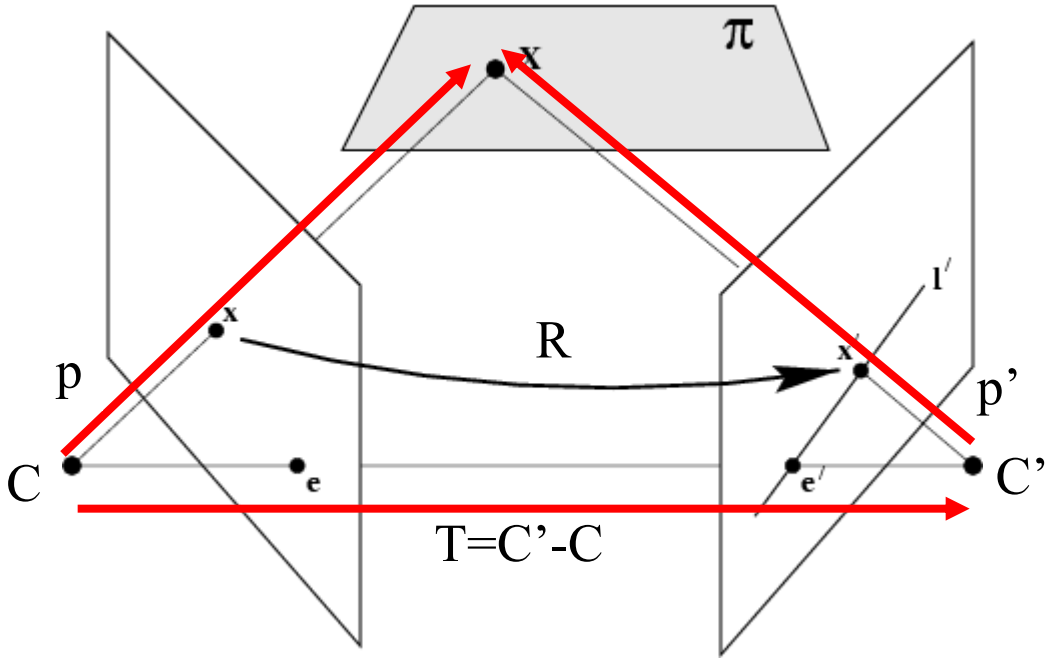
$$\mathbf{S}=\begin{bmatrix}0 & -T_z & T_y\\ T_z & 0 & -T_x\\ -T_y & T_x & 0\end{bmatrix}$$

$$\Rightarrow\quad (\mathbf{R}^{\mathrm{T}}\mathbf{p}')^{\mathrm{T}}(\mathbf{S}\mathbf{p})=0$$

$$\Rightarrow\quad (\mathbf{p}'^{\mathrm{T}}\mathbf{R})(\mathbf{S}\mathbf{p})=0$$

$$\Rightarrow\quad \mathbf{p}'^{\mathrm{T}}\mathbf{E}\mathbf{p}=0$$ Essential matrix

The Essential Matrix E



$$\mathbf{p}'^{T}\,\mathbf{E}\mathbf{p} = 0$$

$$\mathbf{p}'^{\mathrm{T}}\,\mathbf{E}\mathbf{p} = 0$$

Let K and K' be the intrinsic matrices, then

$$\mathbf{p} = \mathbf{K}^{-1}\mathbf{x} \qquad \mathbf{p}' = \mathbf{K}'^{-1}\,\mathbf{x}'$$

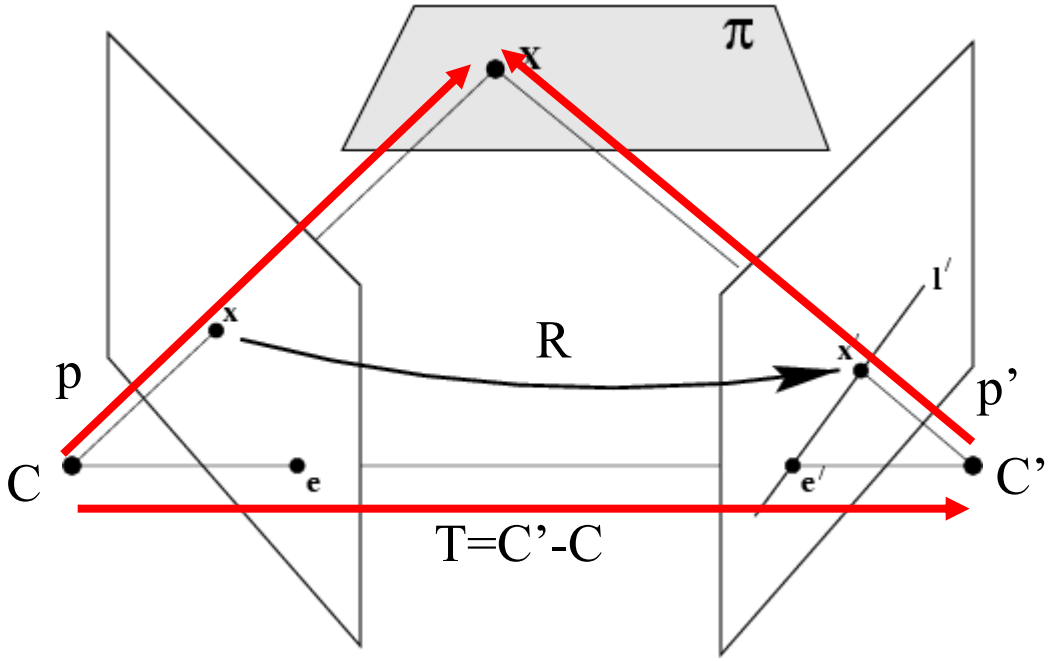
$$\Rightarrow \quad (\mathbf{K}'^{-1}\,\mathbf{x}')^{\mathrm{T}}\,\mathbf{E}(\mathbf{K}^{-1}\mathbf{x}) = 0$$

$$\Rightarrow \quad \mathbf{x}'^{\mathrm{T}}\,\boxed{\mathbf{K}'^{-\mathrm{T}}\,\mathbf{E}\mathbf{K}^{-1}}\mathbf{x} = 0$$

$$\Rightarrow \quad \mathbf{x}'^{\mathrm{T}}\,\boxed{\mathbf{F}}\mathbf{x} = 0$$

Fundamental matrix

F vs E



$$\mathbf{p}'^{T}\,\mathbf{E}\mathbf{p} = 0$$

$$\mathbf{x}'^{T}\,\mathbf{F}\mathbf{x} = 0$$

Factorization of the fundamental matrix

SVD method

(i) Define

$$\mathtt{Z} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

(ii) Compute the SVD

$$\mathtt{F} = \mathtt{U}\mathtt{D}\mathtt{V}^\top \text{ where } \mathtt{D} = \mathrm{diag}r, s, 0$$

(iii) Factorization is

$$\mathtt{F} = (\mathtt{U}\mathtt{Z}\mathtt{U}^\top)(\mathtt{U}\mathtt{Z}\mathtt{D}\mathtt{V}^\top)$$

- Simultaneously corrects $\mathtt{F}$ to a singular matrix.

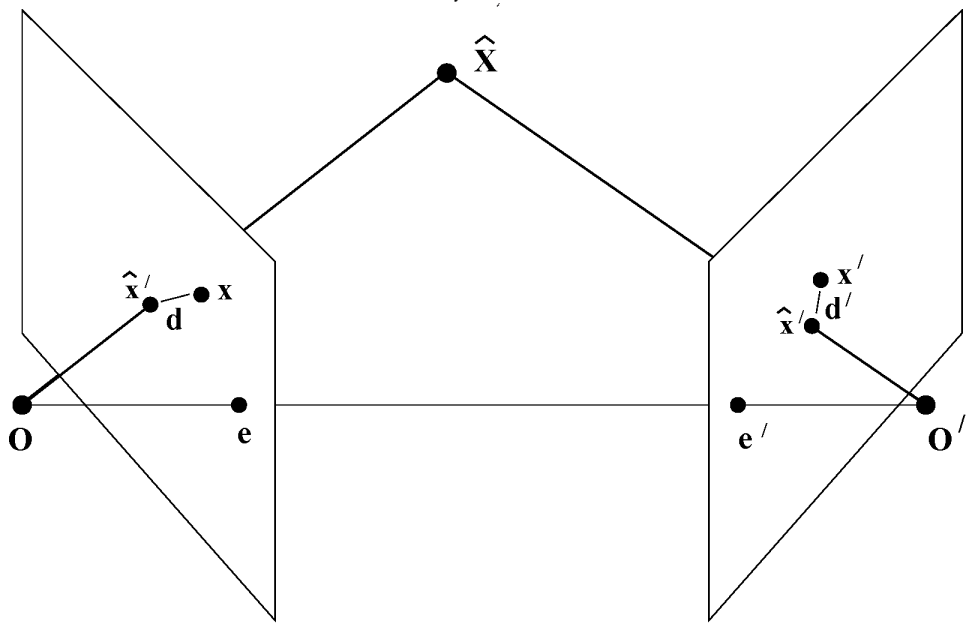
Triangulation

Triangulation

Triangulation :

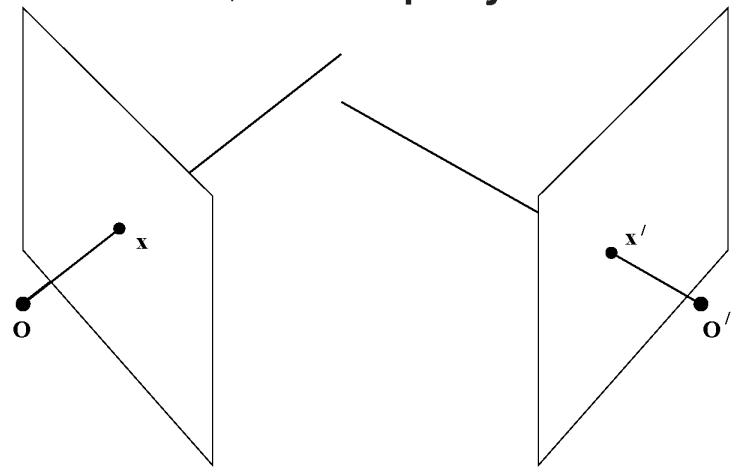
- Knowing $\mathtt{P}$ and $\mathtt{P}'$
- Knowing $\mathbf{x}$ and $\mathbf{x}'$
- Compute $\mathbf{x}'$ such that

$$\mathbf{x} = \mathbf{P}\mathbf{X};\ \mathbf{x}' = \mathbf{P}'\mathbf{X}$$

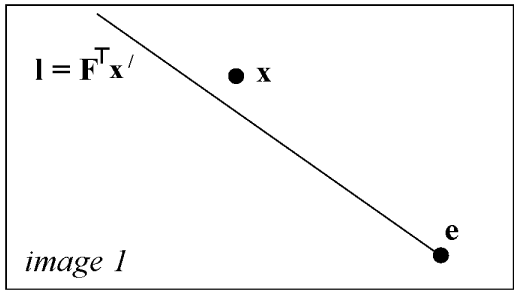


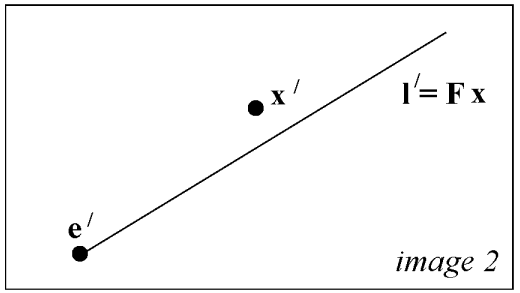
Triangulation in presence of noise

- In the presence of noise, back-projected lines do not intersect.



Rays do not intersect in space





Measured points do not lie on corresponding epipolar lines

Problem statement

- Assume camera matrices are given without error, up to projective distortion.
- Hence `F` is known.
- A pair of matched points in an image are given.
- Possible errors in the position of matched points.
- Find 3D point that minimizes suitable error metric.
- Method must be invariant under 3D projective transformation.

Linear triangulation methods

通过在两个相机中的像素位置确定实际世界坐标点

- Direct analogue of the linear method of camera resectioning.
- Given equations

$$\mathbf{x} = \mathbf{P}\mathbf{X};\ \mathbf{x}' = \mathbf{P}'\mathbf{X}$$

- $\mathbf{p}^{i\top}$ are the rows of $\mathtt{P}$.
- Write as linear equations in $\mathbf{X}$

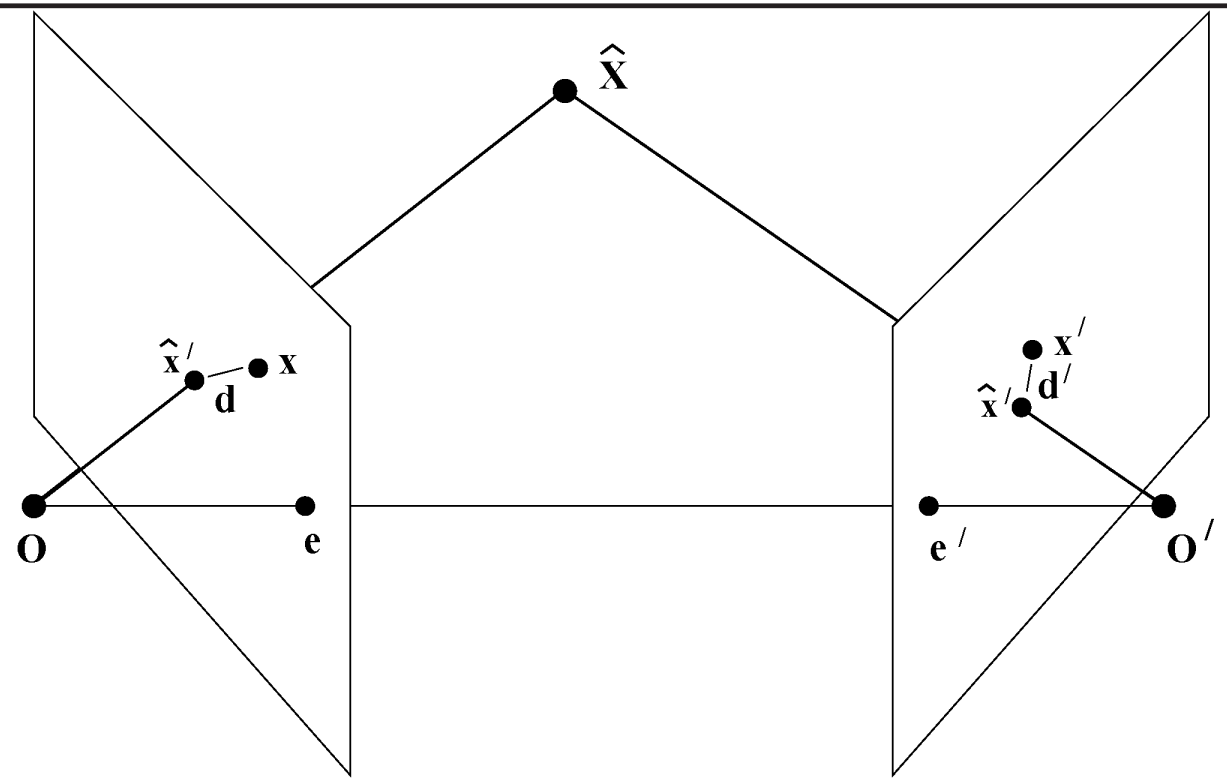
$$\begin{bmatrix} x\mathbf{p}^{3\top} - \mathbf{p}^{1\top} \\ y\mathbf{p}^{3\top} - \mathbf{p}^{2\top} \\ x'\mathbf{p}'^{3\top} - \mathbf{p}'^{1\top} \\ y\mathbf{p}'^{3\top} - \mathbf{p}'^{2\top} \end{bmatrix} \mathbf{X} = 0$$

- Solve for $\mathbf{X}$.
- Generalizes to point match in several images.
- Minimizes no meaningful quantity – not optimal.

Minimizing geometric error

- Point x in space maps to projected points $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ in the two images.
- Measured points are $\mathbf{x}$ and $\mathbf{x}'$.
- Find $\mathbf{X}$ that minimizes difference between projected and measured points.

Geometric error . . .



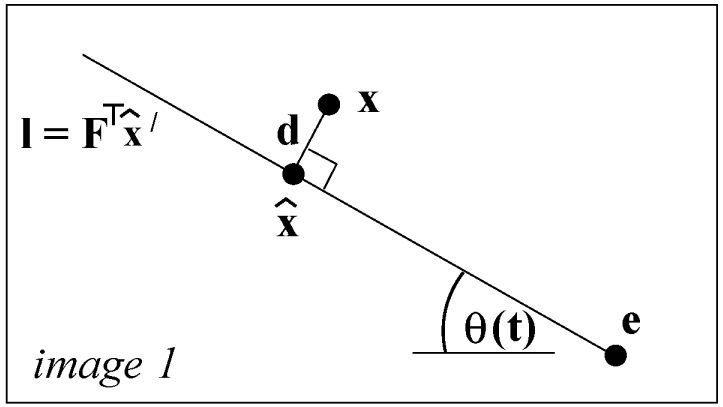
Cost function

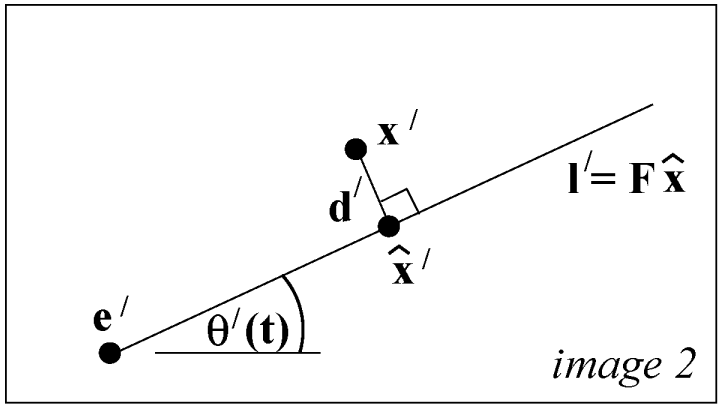
$$\mathcal{C}(\mathrm{x}) = d(\mathbf{x}, \hat{\mathbf{x}})^2 + d(\mathbf{x}', \hat{\mathbf{x}}')^2$$

Different formulation of the problem

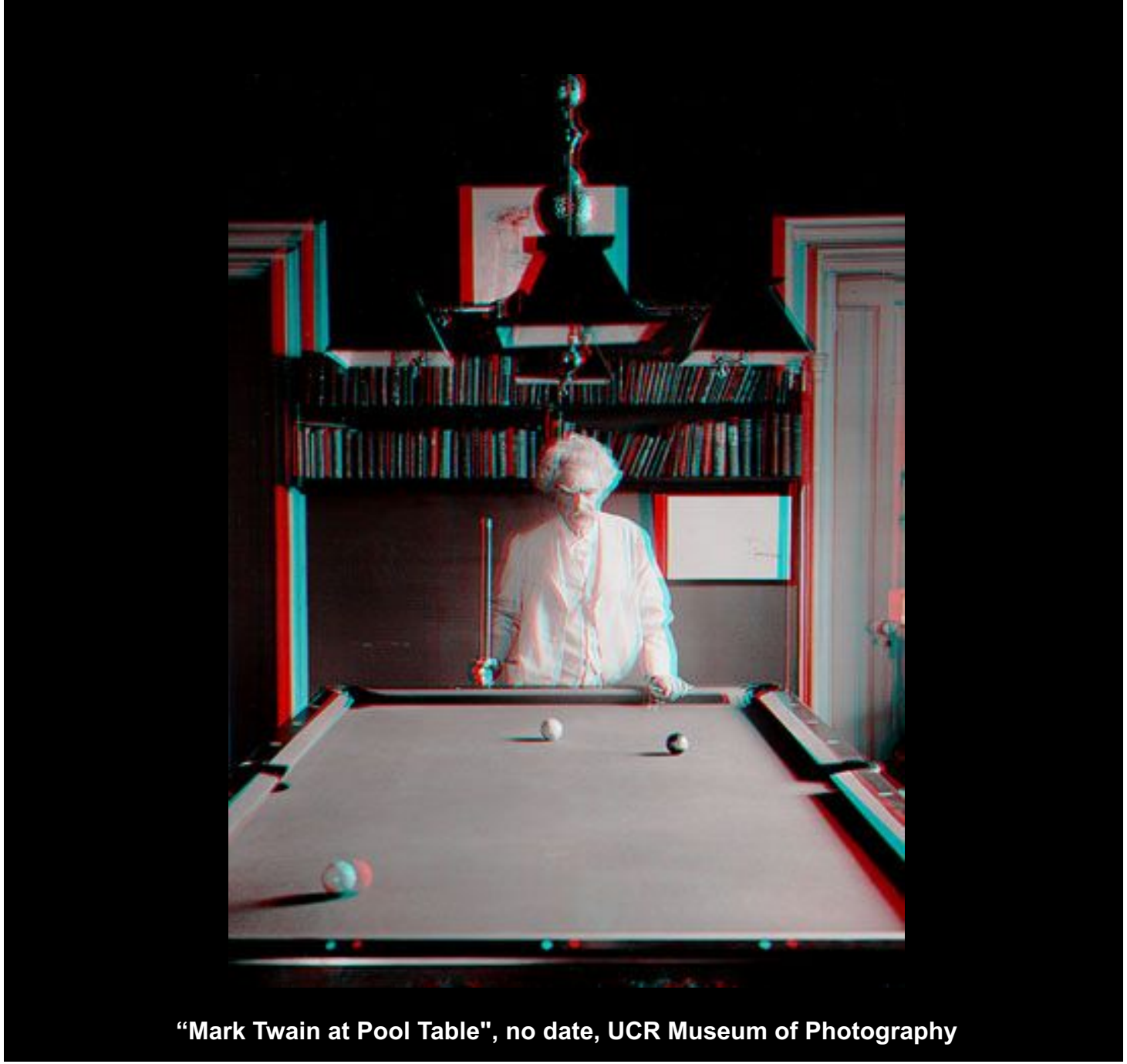
Minimization problem may be formulated differently:

- Minimize
$$d(\mathbf{x}, \mathbf{l})^2 + d(\mathbf{x}', \mathbf{l}')^2$$
- $\mathbf{l}$ and $\mathbf{l}'$ range over all choices of corresponding epipolar lines.
- $\hat{\mathbf{x}}$ is the closest point on the line $\mathbf{l}$ to $\mathbf{x}$.
- Same for $\hat{\mathbf{x}}'$.





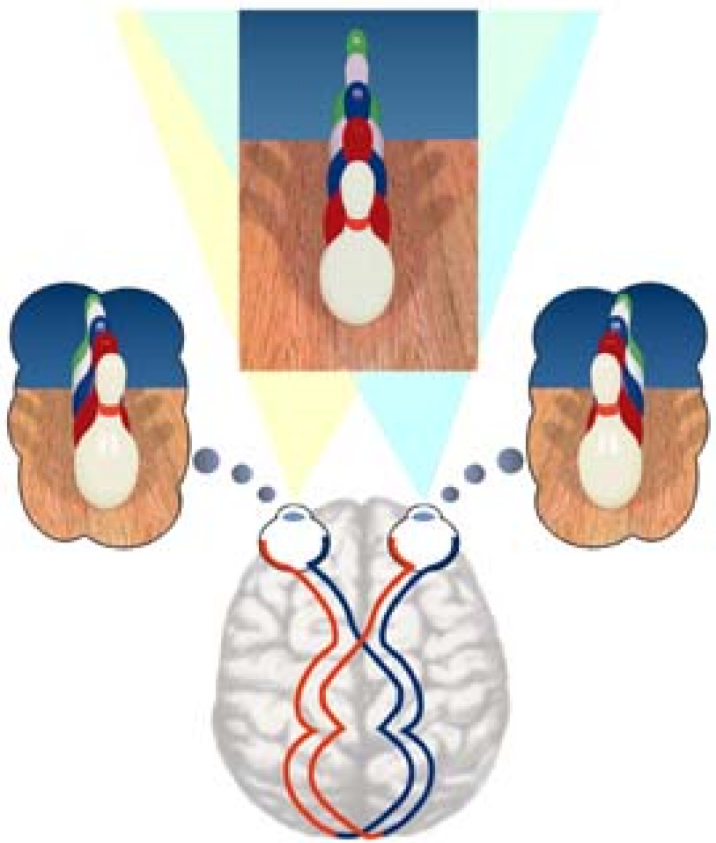
Stereo Vision



“Mark Twain at Pool Table", no date, UCR Museum of Photography

Credit: Noah Snavely

Binocular Stereo Vision



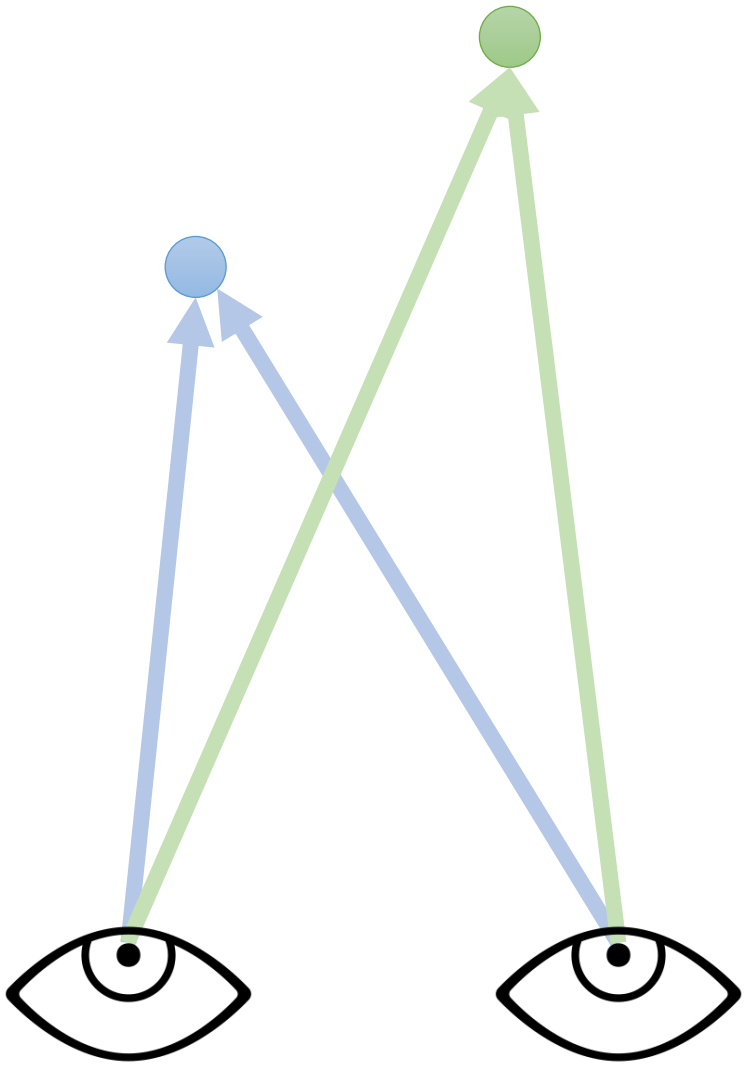
Public Library, Stereoscopic Looking Room, Chicago, by Phillips, 1923

https://giphy.com/gifs/wigglegram-706pNfSKyaDug

Credit: Noah Snavely

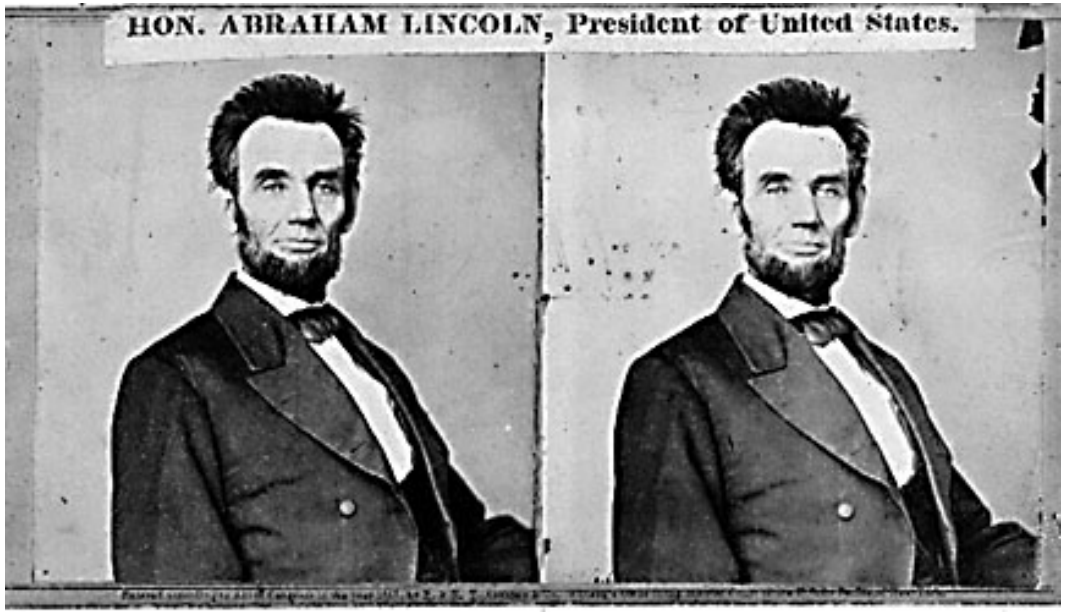
Stereo Vision as Localizing Points in 3D

- An object point will project to some point in our image
- That image point corresponds to a ray in the world
- Two rays intersect at a single point, so if we want to localize points in 3D we need 2 eyes



Credit: Noah Snavely

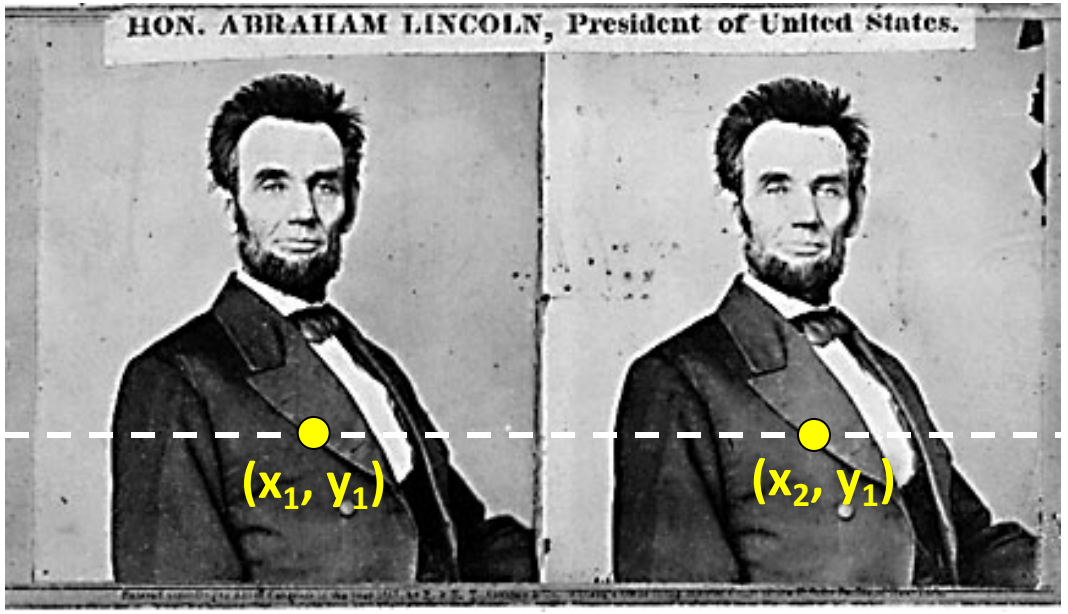
Stereo



- Given two images from different viewpoints
  - How can we compute the depth of each point in the image?
  - Based on *how much each pixel moves* between the two images

Credit: Noah Snavely

Epipolar geometry



Two images captured by a purely horizontal translating camera (*rectified* stereo pair)

$x_2 - x_1$ = the *disparity* of pixel (x_1, y_1)

Credit: Noah Snavely

Disparity = inverse depth

http://stereo.nypl.org/view/41729

(Or, hold a finger in front of your face and wink each eye in succession.)

3D Perception Using Stereo Vision

- Humans do it well from a *single* image and very well through *stereo* images
- Mechanism is not well understood:
  - Biological components: some understanding
  - Algorithm: little understanding
- Goal of computer vision is to mimic the *functionality* of the biological system
  - *Not* to mimic its mechanics

Stereo Vision

- Stereo matching computes depth information from two images
- Depth information can be used to…
  - Differentiate objects from the background
  - Differentiate objects from one another
  - Expose camouflaged objects
  - Navigate in environment avoiding obstacles

Stereo Vision: two sub-problems

Correspondence Problem

- The problem of measuring the disparity of each point in the two eye (camera) projections

Interpretation Problem

- The use of disparity information to recover the orientation and distance of surfaces in the scene

Stereo Vision: Algorithmic Steps

- Basic steps to be performed in any stereo imaging system:
  1. Image Acquisition
  2. Camera Modelling
  3. Feature Extraction
  4. Image Matching
  5. Depth Interpolation

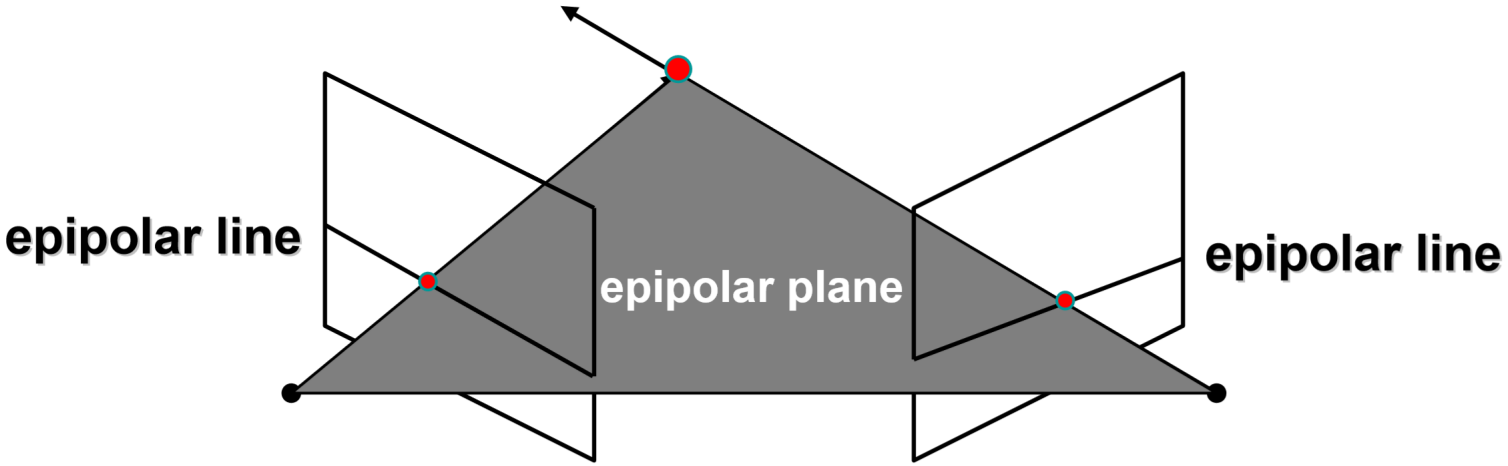
1. Stereo Image Acquisition

- As the name implies
- Capturing two images with a very specific camera geometry

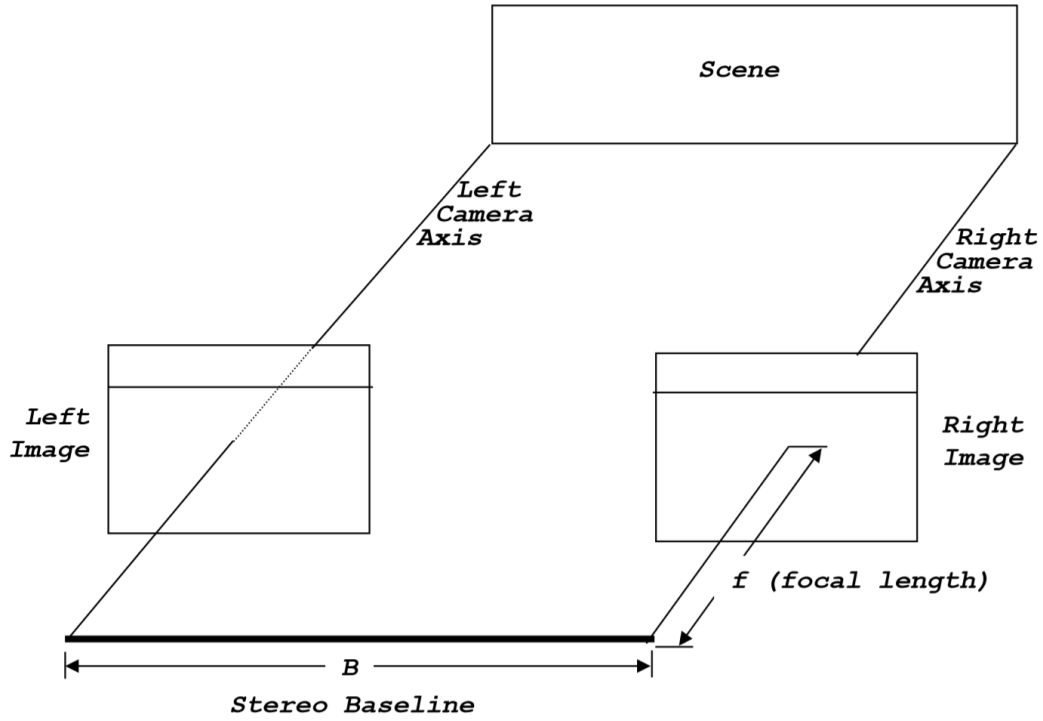
2. Camera Modelling

- Related to image acquisition
- For accurate depth results the camera parameters must be known
  - Intrinsic parameters
  - Focal length, pixel skew/shape, distortion, etc
- And the relationship between the two cameras must be known
  - Extrinsic parameters
  - Rotation, translation

General Epipolar Constraint



Simple (Ideal) Stereo Geometry



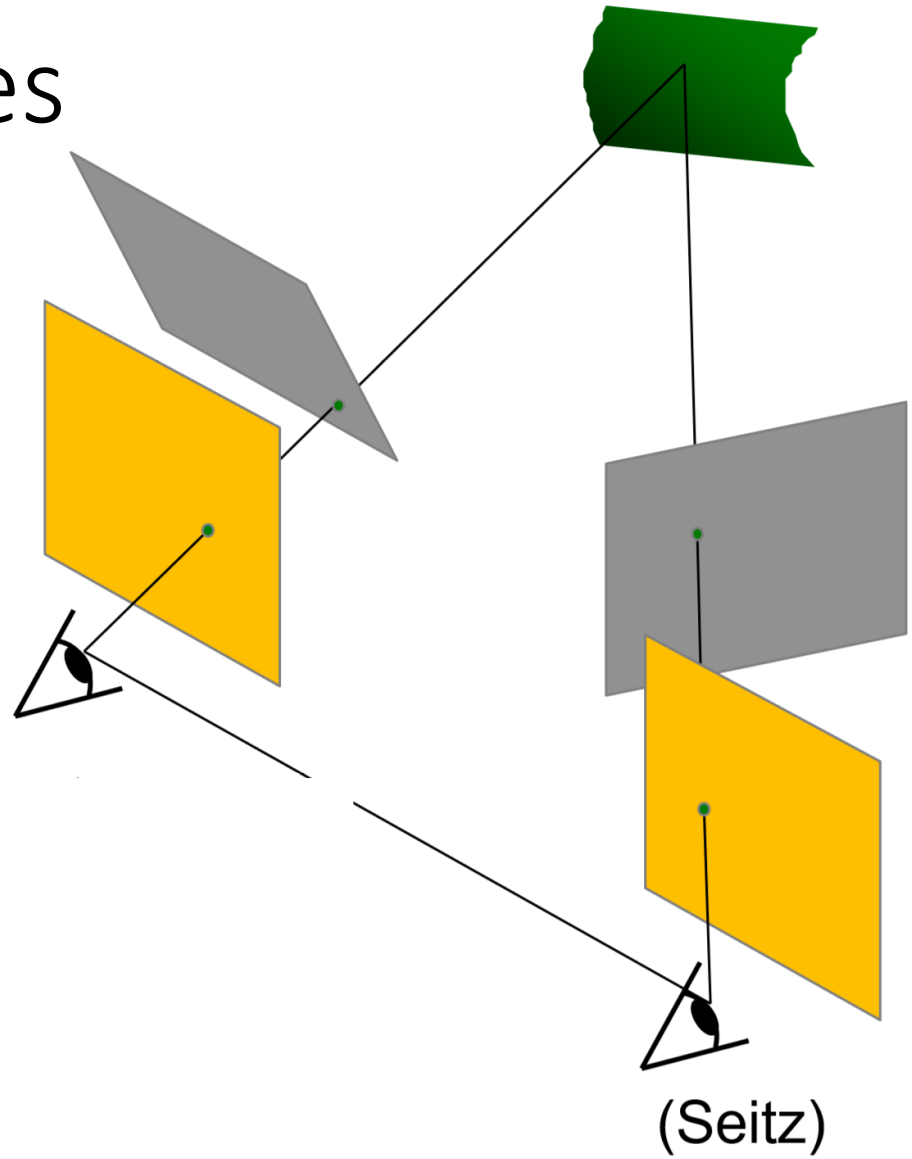
- Two slightly different images

Simplest Case: Rectified Stereo Images

- Assumes:
  - Image planes of cameras are parallel
  - Focal points are at same height
  - Identical focal lengths
- Then epipolar lines fall along the horizontal scan lines of the images
- We will assume images have been rectified so that epipolar lines correspond to scan lines:
  - Simplifies algorithms
  - Improves efficiency

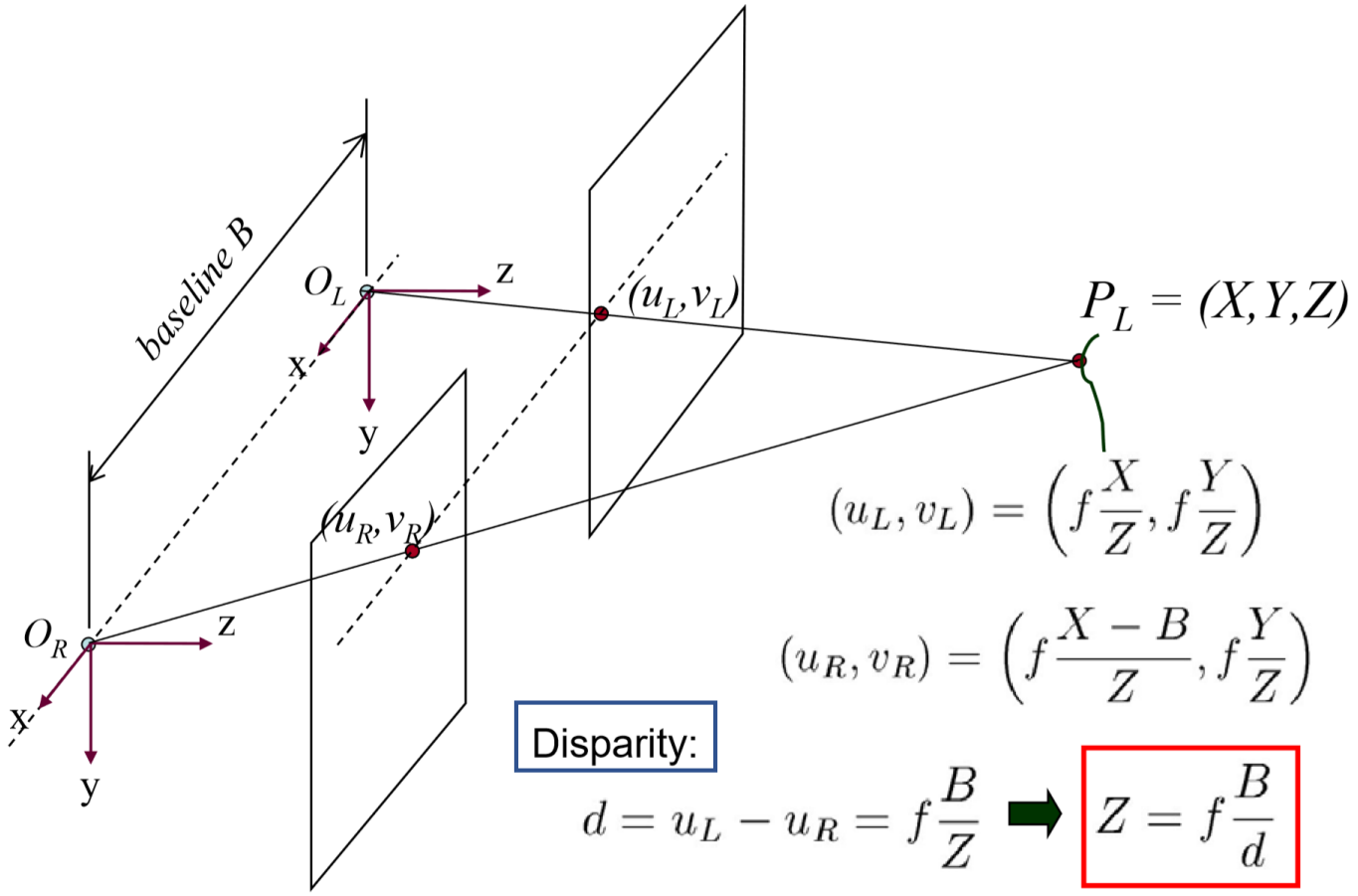
Simplest Case: Rectified Stereo Images

- We can always achieve this *ideal* geometry with stereo-rectification
- Image Reprojection:
  - Reproject image planes onto a common plane parallel to line between optical centres using suitable planar homographies



(Seitz)

Disparity Computation

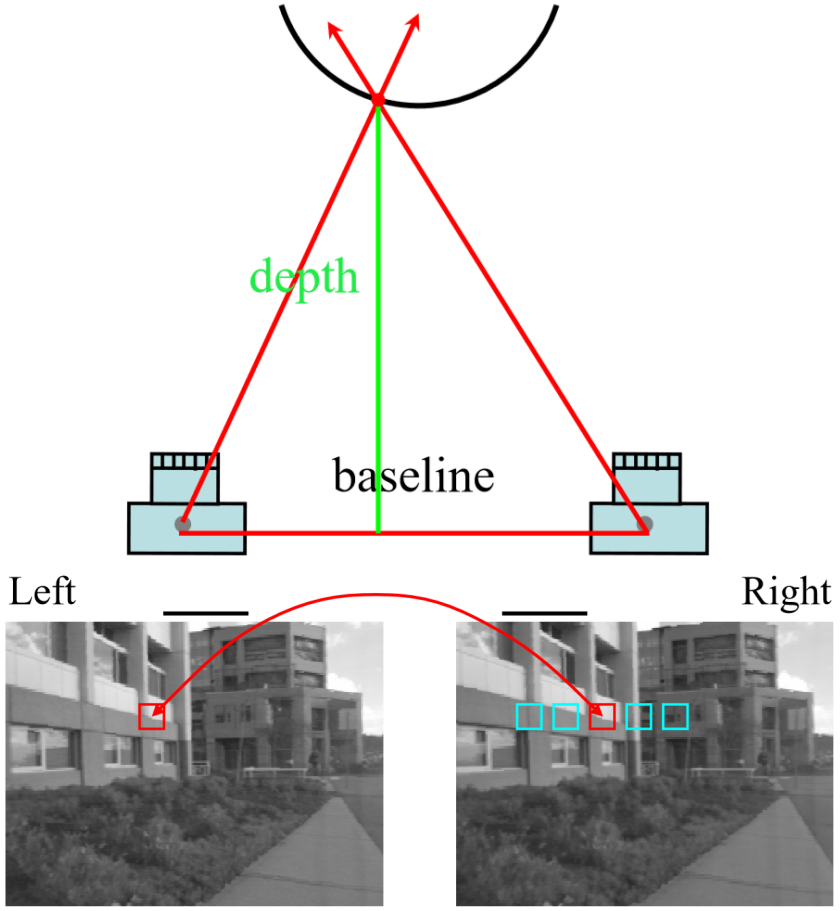


3-4. The Correspondence Problem

- What should we match?

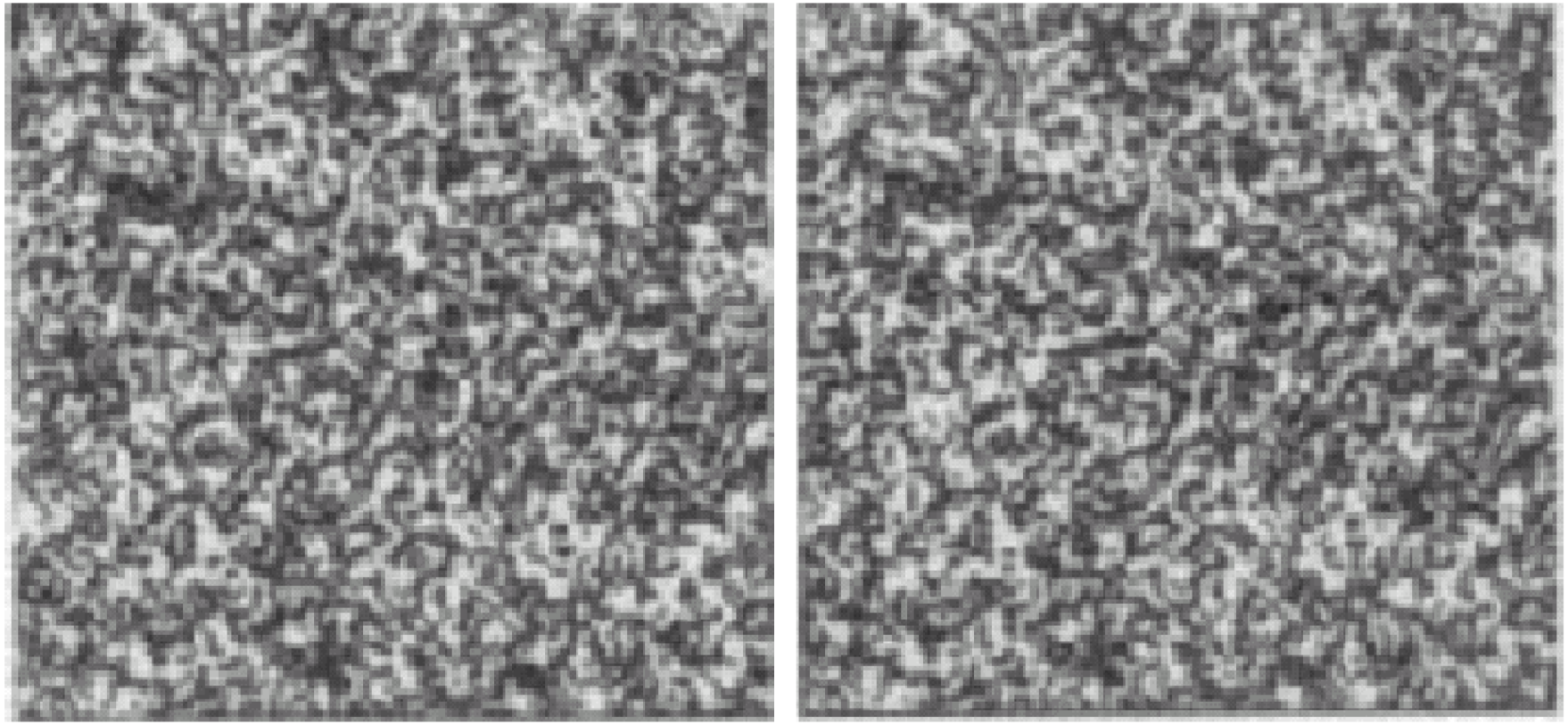
- Objects?
- Edges?
- Pixels?
- Super-pixels (set of pixels)?

The Correspondence Problem



- Triangulate on two images of the same point to recover depth
  - Feature matching across views
  - Calibrated cameras
- Matching correlation windows across scan lines

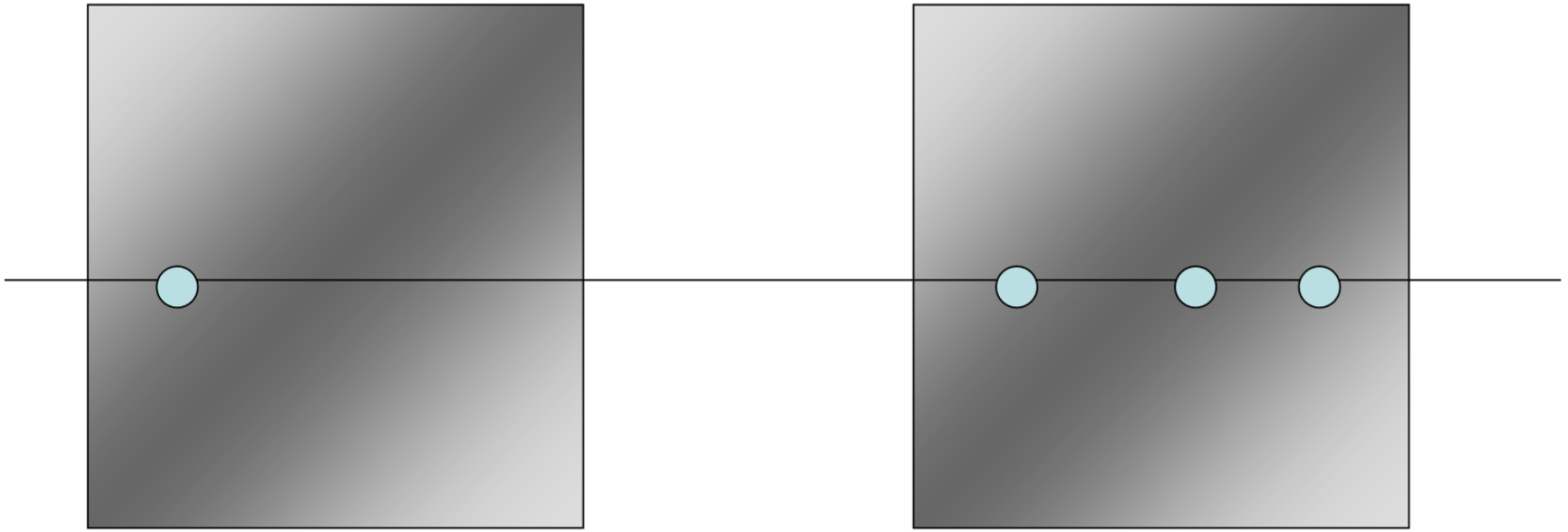
Random Dot Stereograms



- Julesz: showed that recognition is not needed for stereo

The Correspondence Problem: Epipolar Constraint

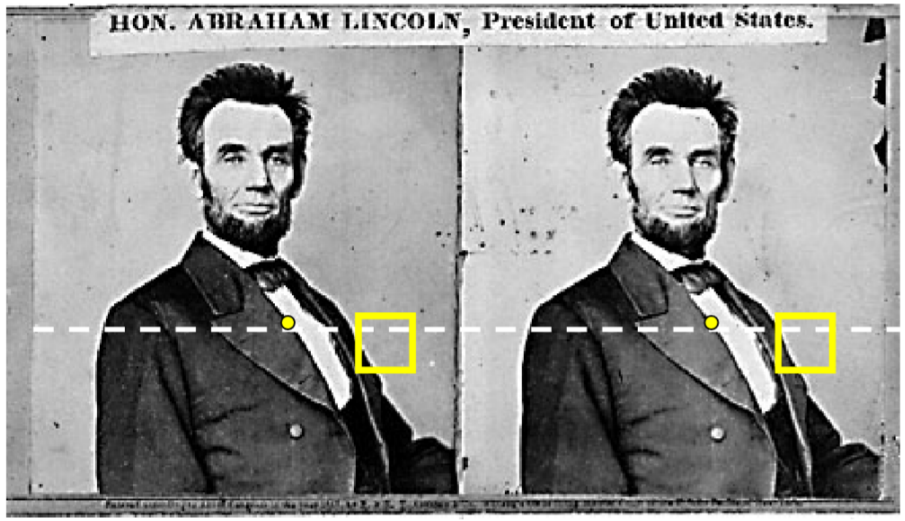
- The epipolar constraint removes some ambiguity



The Correspondence Problem: Colour Constancy Constraint

- Same world point has same intensity (or colour) in both images
  - True for Lambertian surfaces
    - A Lambertian surface has a brightness that is independent of viewing angle
  - Violations:
    - Noise
    - Specularity
    - Non-Lambertian materials
    - Pixels that contain multiple surfaces

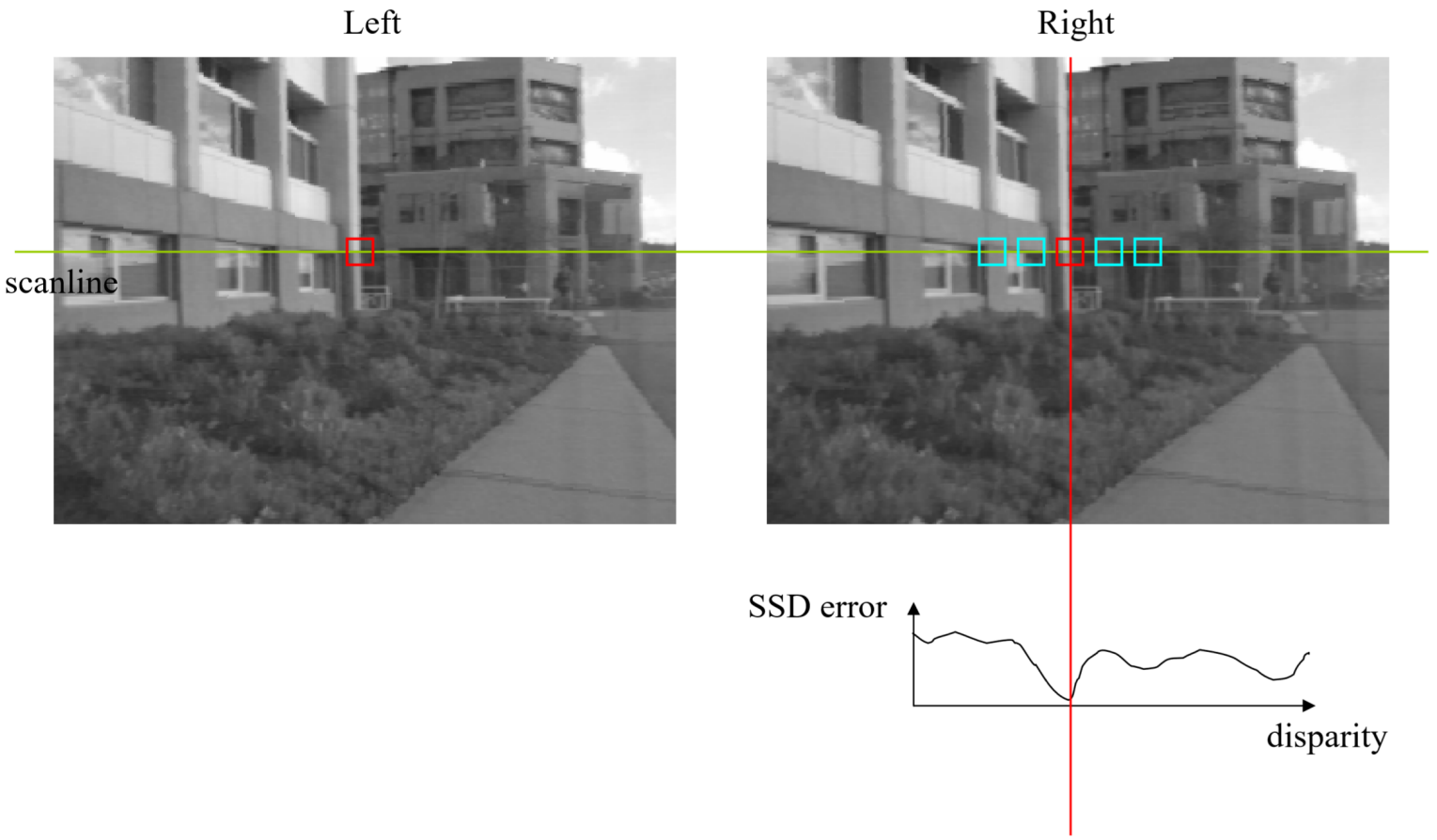
The Correspondence Problem: Pixel Matching



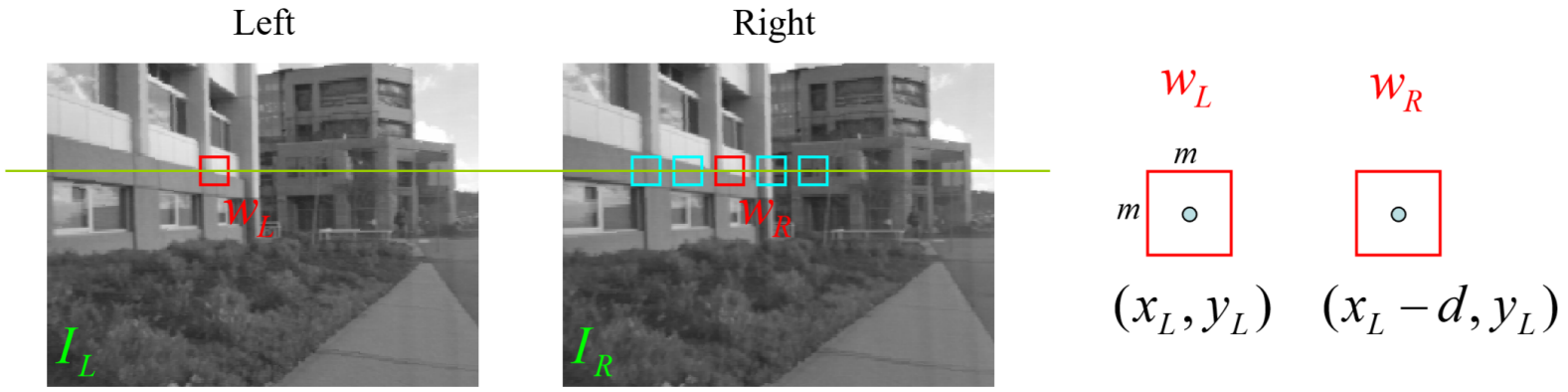
- For each epipolar line:
  - For each pixel in the left image:
    - Compare with every pixel on same epipolar line in right image
    - Pick pixel with minimum match cost
- Still too much ambiguity
  - Improvement: match windows

(Seitz)

The Correspondence Problem: Correspondence Using Correlation



Sum of Squared (Pixel) Differences



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

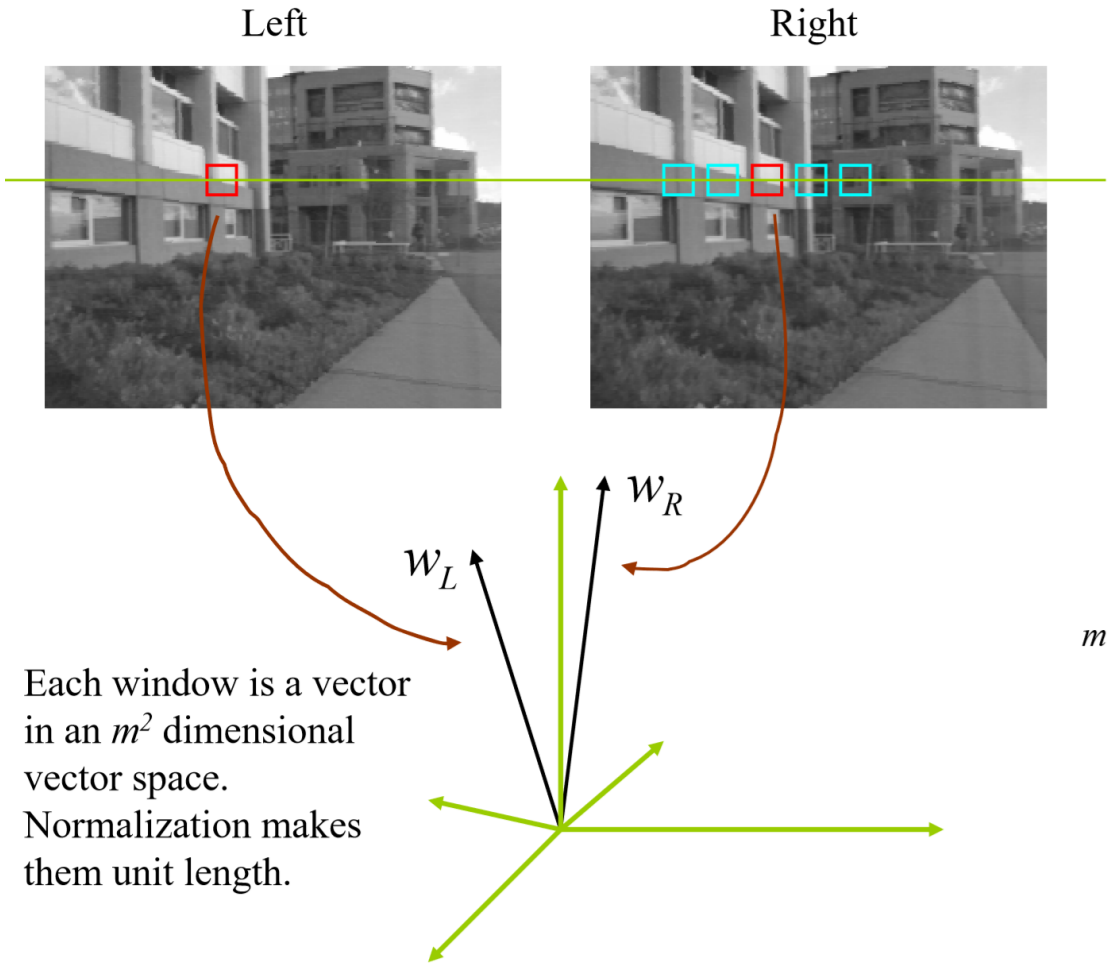
We define the window function :

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

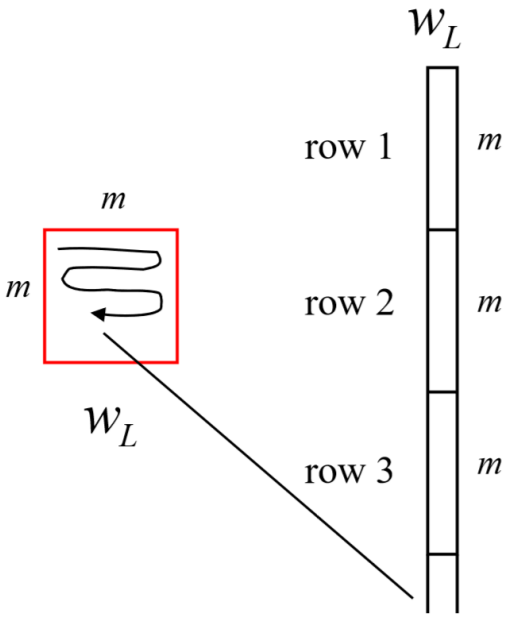
The SSD cost measures the intensity difference as a function of disparity :

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

Images as Vectors

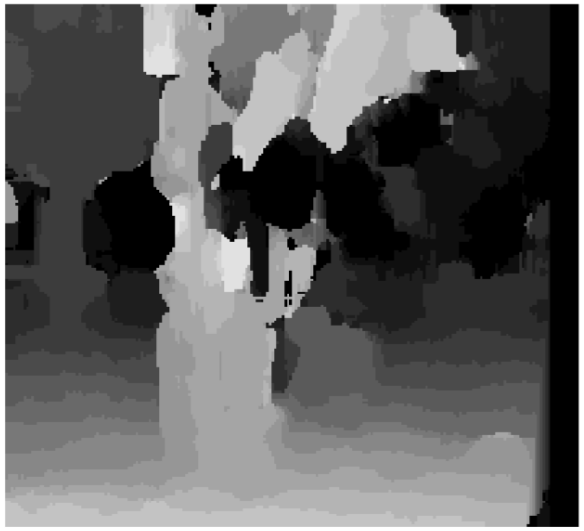


"Unwrap" image to form vector, using raster scan order



Effect of Window Size

W = 3



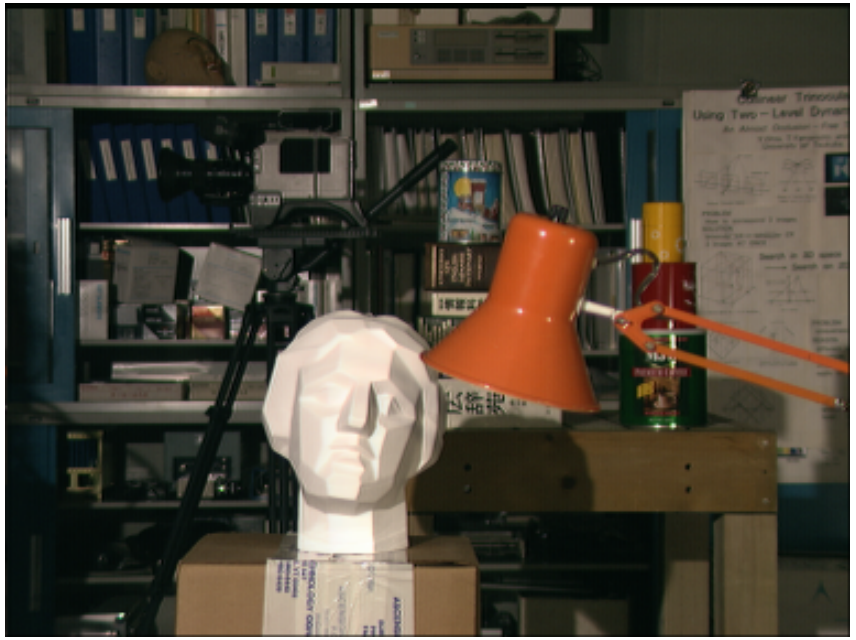
W = 20

- Improvement: use an adaptive window size (try multiple sizes and select best match)

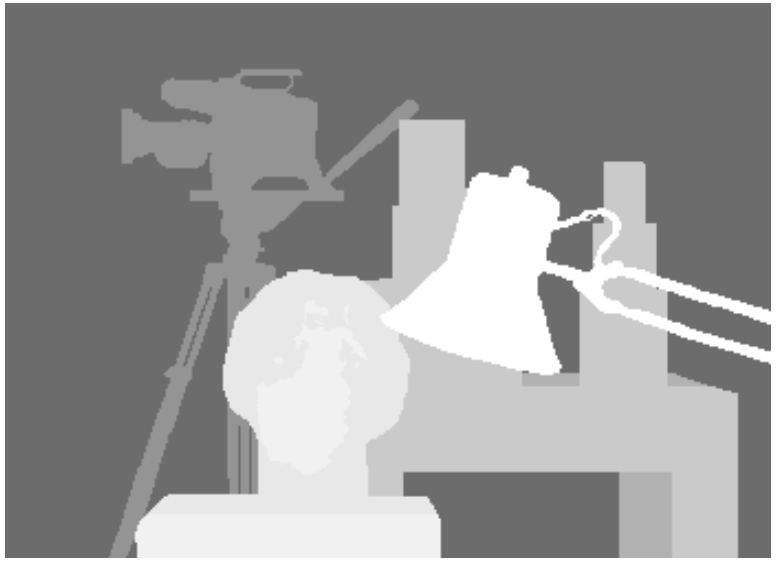
(Seitz)

Stereo results

- Data from University of Tsukuba
- Similar results on other images without ground truth

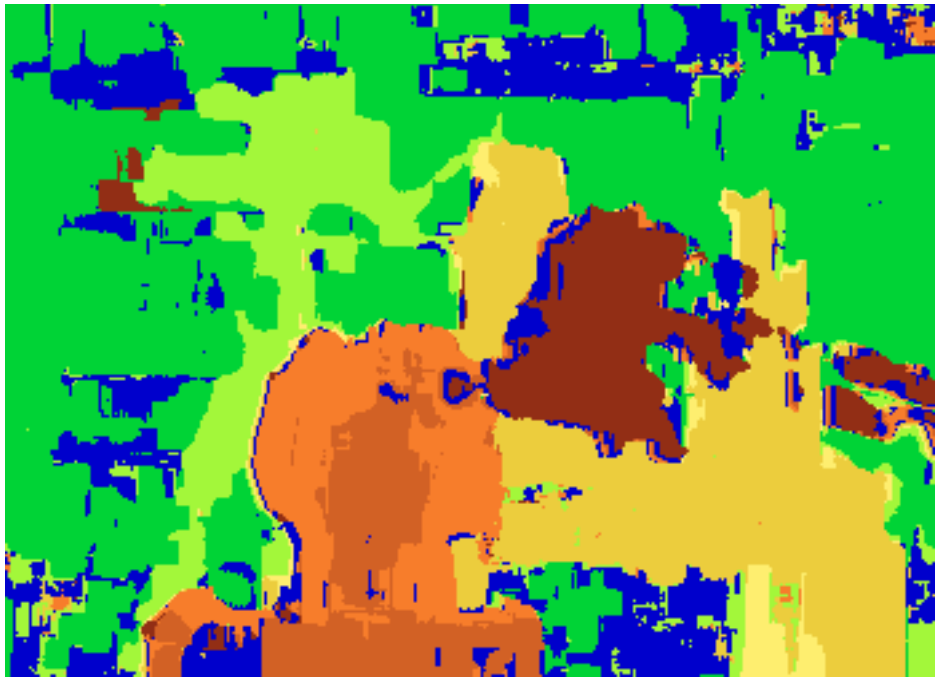


Scene



Ground truth

Results with window search



Window-based matching
(best window size)

Ground truth

Better methods exist…

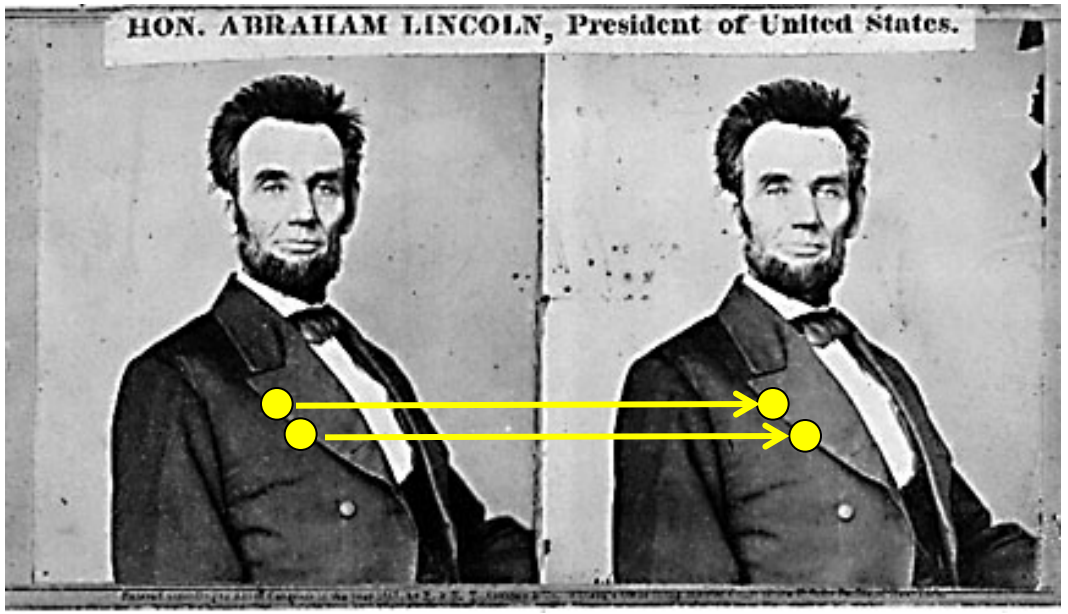
Graph cuts-based method

Ground truth

Boykov et al., Fast Approximate Energy Minimization via Graph Cuts, International Conference on Computer Vision 1999.

For the latest and greatest: http://www.middlebury.edu/stereo/

Stereo as energy minimization



- What defines a good stereo correspondence?
  1. Match quality
     - Want each pixel to find a good match in the other image
  2. Smoothness
     - If two pixels are adjacent, they should (usually) move about the same amount

Stereo as energy minimization

- Find disparity map *d* that minimizes an *energy function*

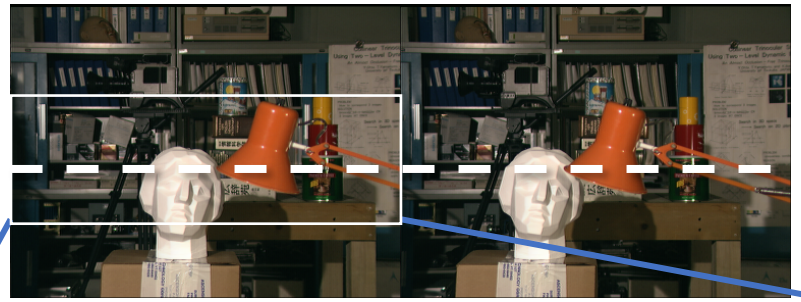
$$E(d)$$

- Simple pixel / window matching

$$E(d) = \sum_{(x,y) \in I} C(x, y, d(x, y))$$

$C(x, y, d(x, y))$ = SSD distance between windows $I(x, y)$ and $J(x + d(x,y), y)$

Stereo as energy minimization



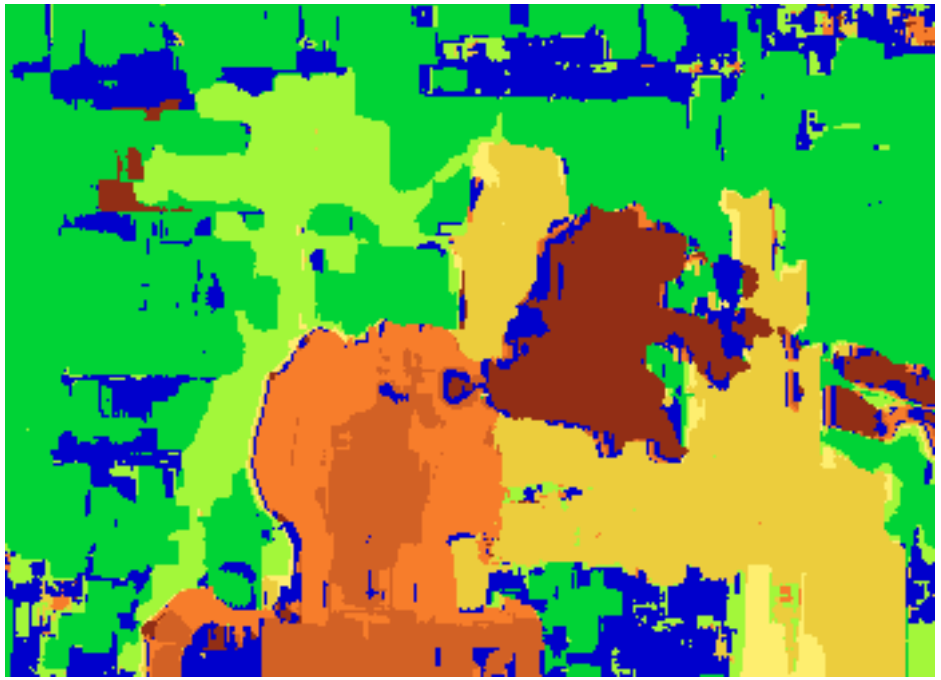
$C(x, y, d)$; the *disparity space image* (DSI)

Stereo as energy minimization

Simple pixel / window matching: choose the minimum of each column in the DSI independently:

$$d(x, y) = \arg\min_{d'} C(x, y, d')$$

Greedy selection of best match



Stereo as energy minimization

- Better objective function

$$E(d) = \underbrace{E_d(d)}_{\text{match cost}} + \lambda \underbrace{E_s(d)}_{\text{smoothness cost}}$$

Want each pixel to find a good match in the other image

Adjacent pixels should (usually) move about the same amount

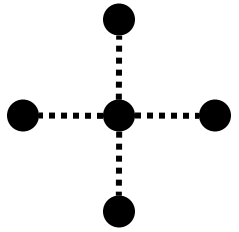
Stereo as energy minimization

$$E(d) = E_d(d) + \lambda E_s(d)$$

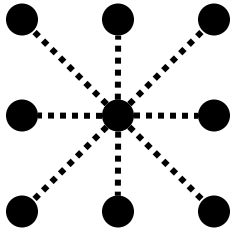
match cost: $$E_d(d) = \sum_{(x,y)\in I} C(x, y, d(x, y))$$

smoothness cost: $$E_s(d) = \sum_{(p,q)\in\mathcal{E}} V(d_p, d_q)$$

$\mathcal{E}$: set of neighboring pixels



4-connected neighborhood



8-connected neighborhood

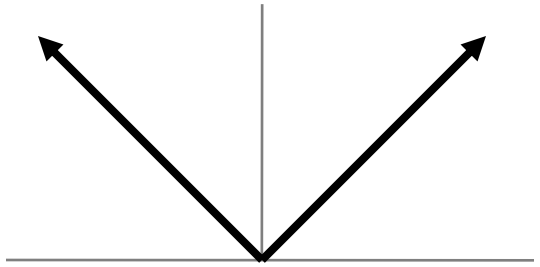
Smoothness cost

$$E_s(d) = \sum_{(p,q)\in\mathcal{E}} V(d_p, d_q)$$

How do we choose *V*?

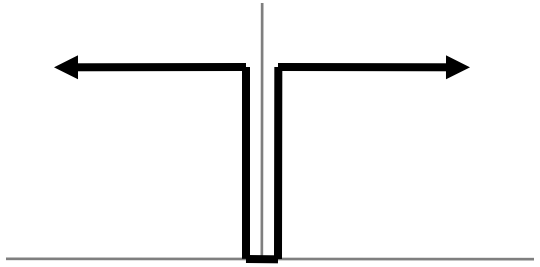
$$V(d_p, d_q) = |d_p - d_q|$$

L_1 distance



$$V(d_p, d_q) = \begin{cases} 0 & \text{if } d_p = d_q \\ 1 & \text{if } d_p \neq d_q \end{cases}$$

"Potts model"



Smoothness cost

$$E(d) = E_d(d) + \lambda E_s(d)$$

- If λ = infinity, then we only consider smoothness
- Optimal solution is a surface of constant depth/disparity
  - *Fronto-parallel* surface

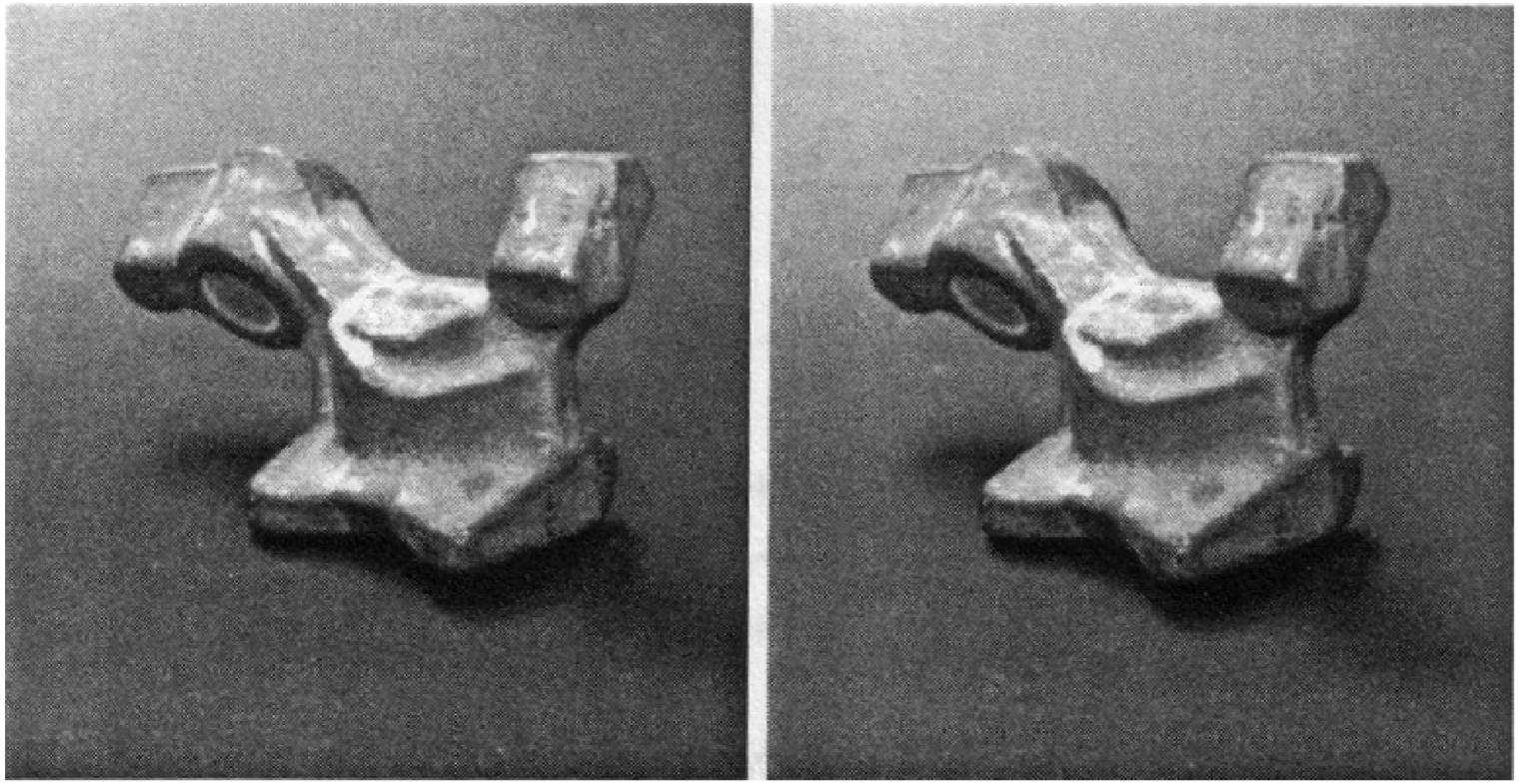
- In practice, want to balance data term with smoothness term

Dynamic programming

$$E(d) = E_d(d) + \lambda E_s(d)$$

- Can minimize this independently per scanline using dynamic programming (DP)

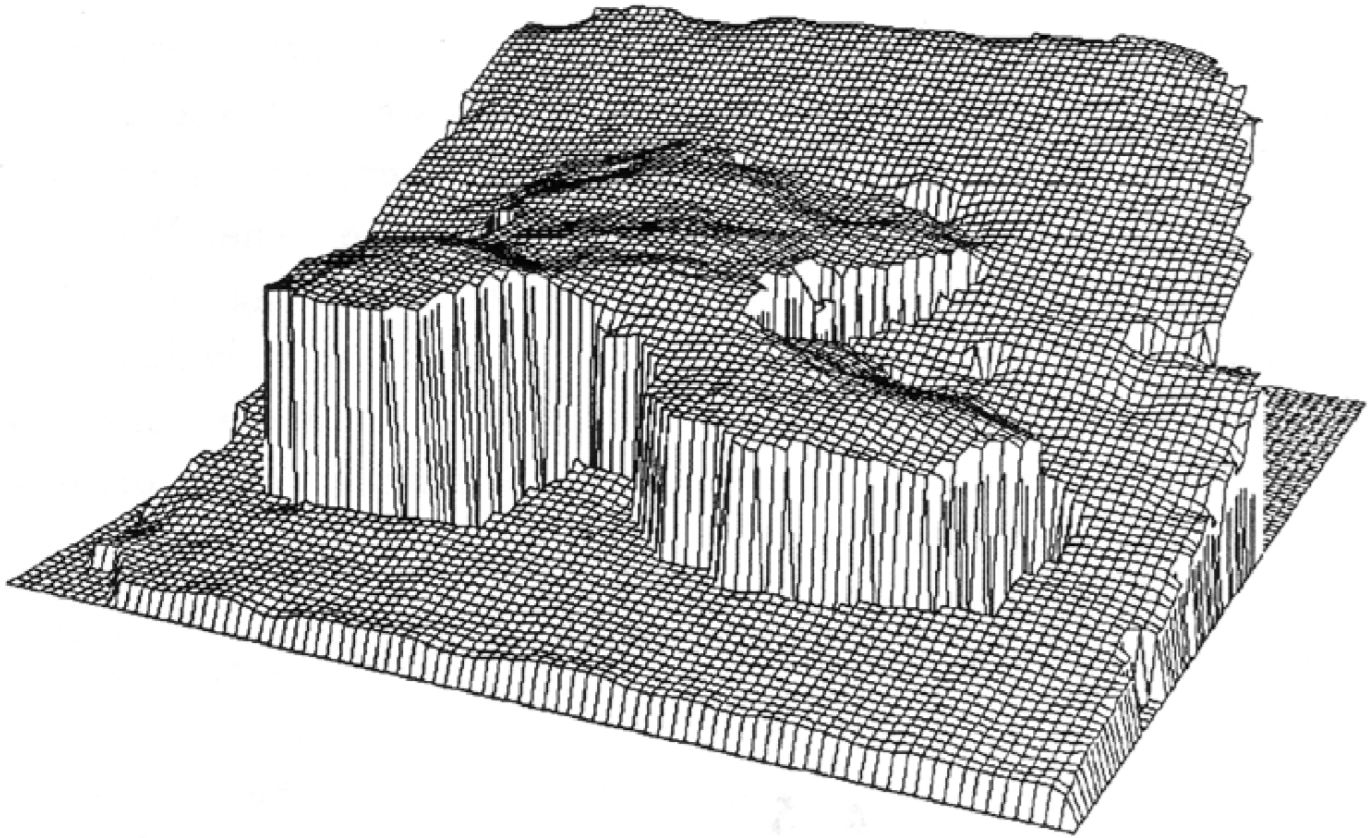
Example Stereo Pair



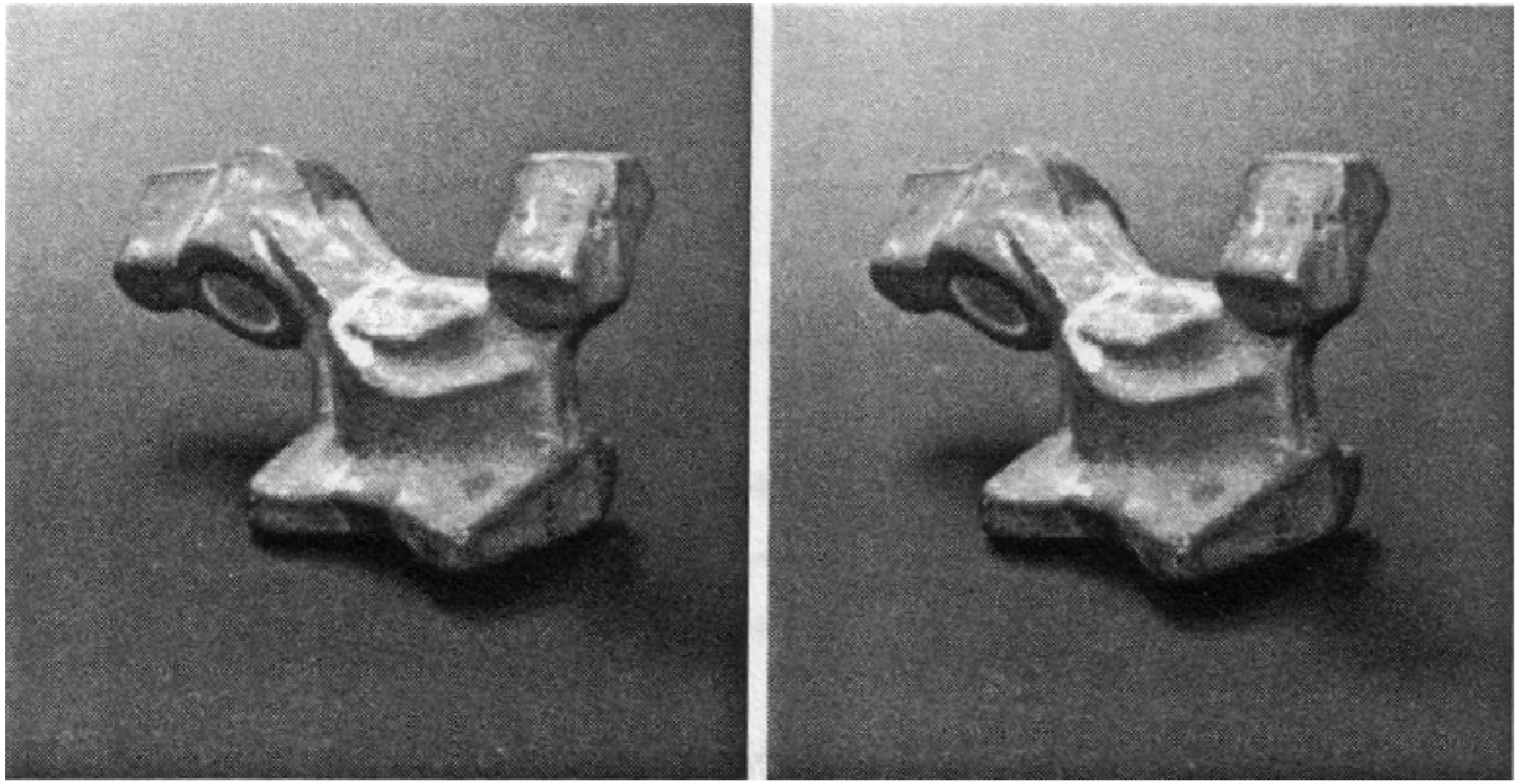
Left Camera

Right Camera

Obtained Depth Map in 3D



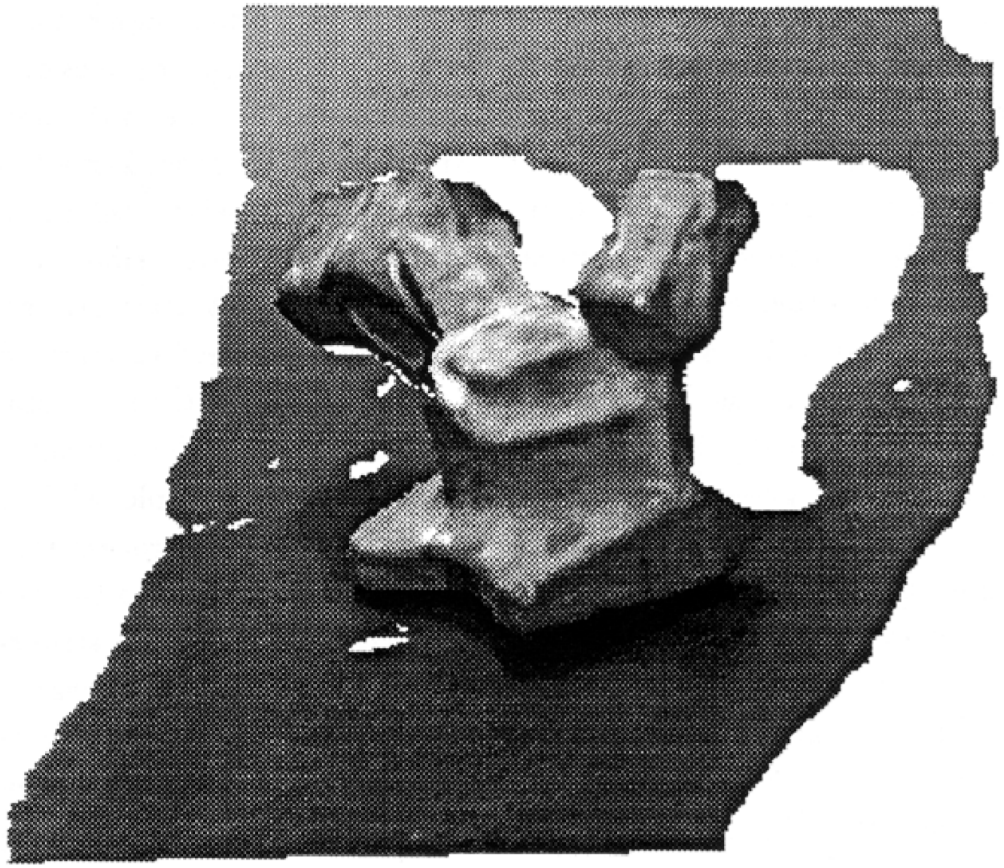
Example Stereo Pair



Left Camera

Right Camera

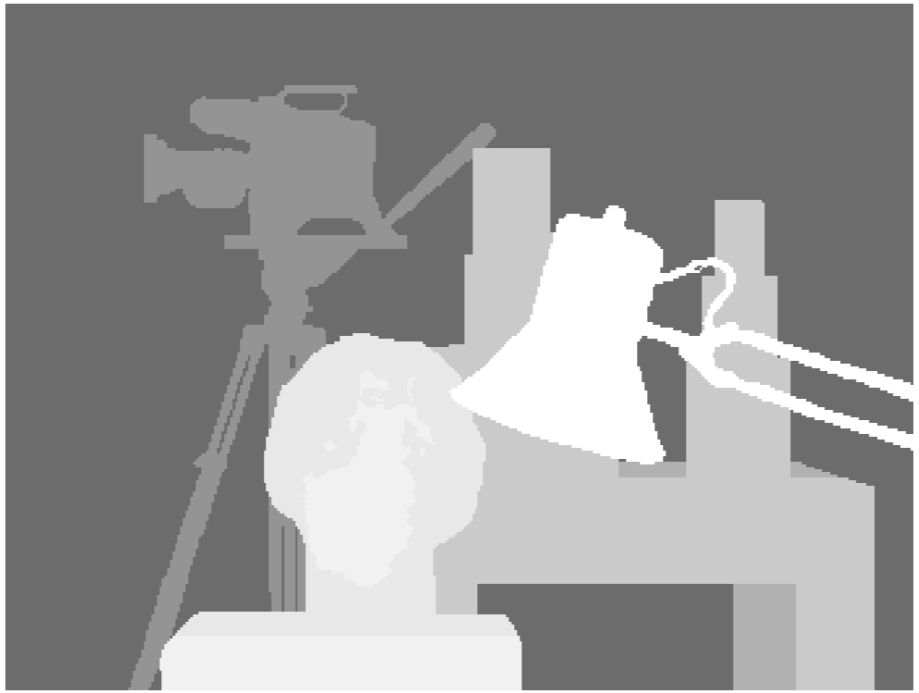
Novel View Synthesis



A Taxonomy of Stereo Algorithms

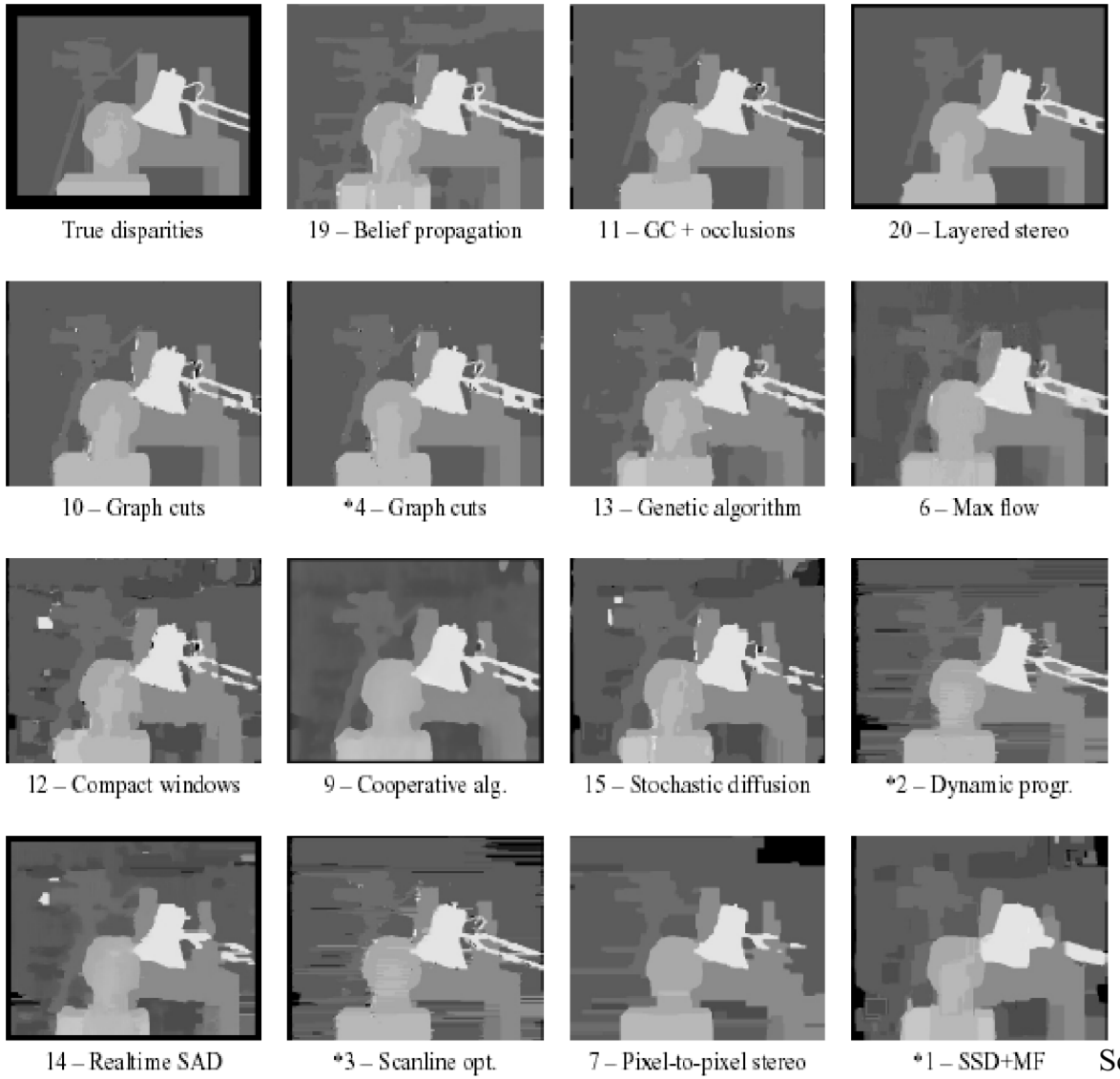
- D. Scharstein and R. Szeliski. "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms," *International Journal of Computer Vision*, 47 (2002), pp. 7-42.

Scene



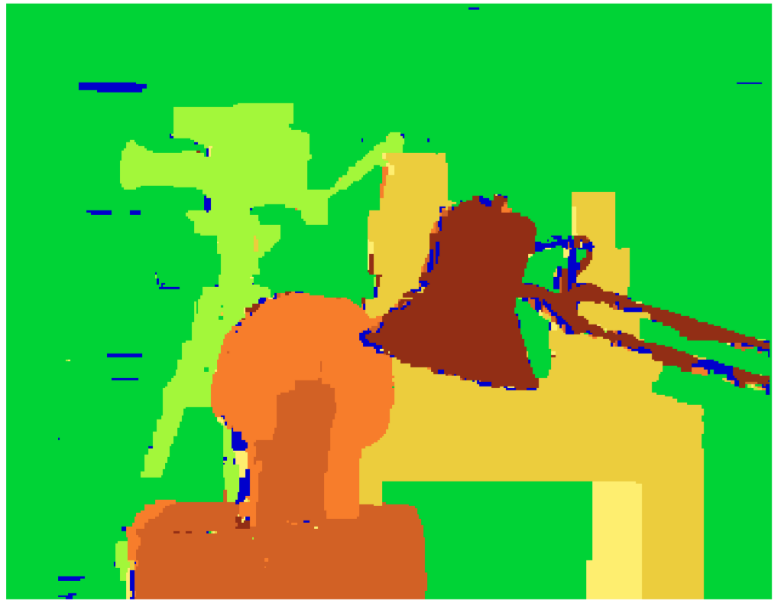
Ground truth

A Taxonomy of Stereo Algorithms



Scharstein and Szeliski

Graph-cut-based Stereo Matching

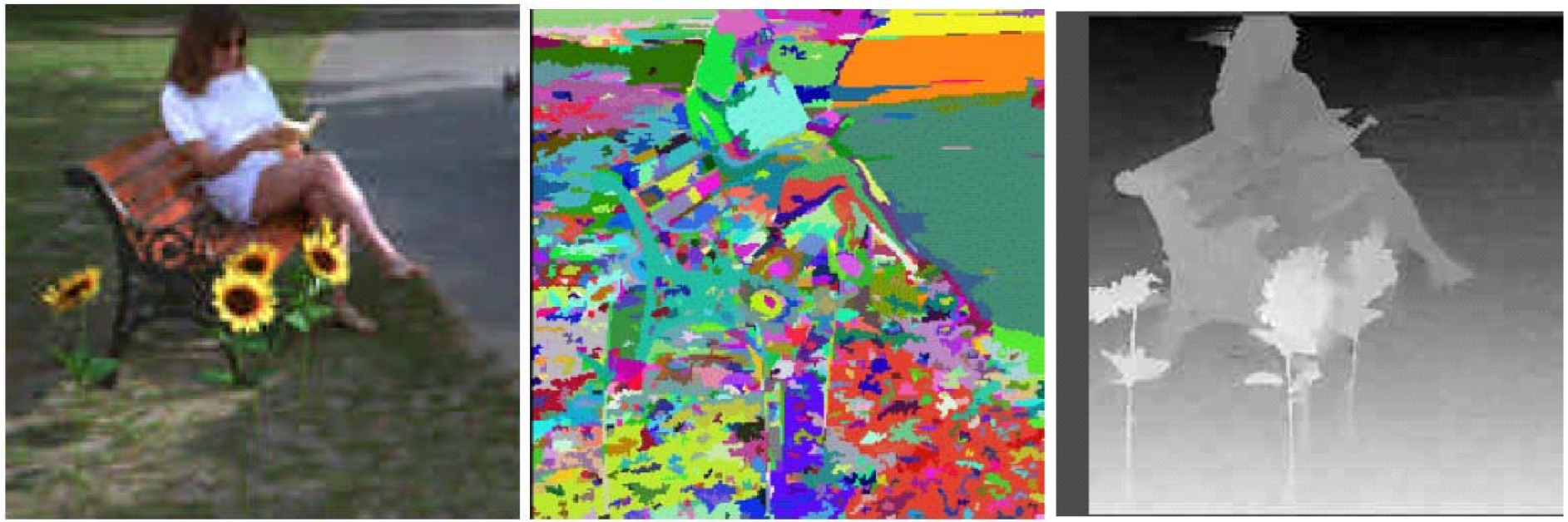


State of the art method: Graph cuts

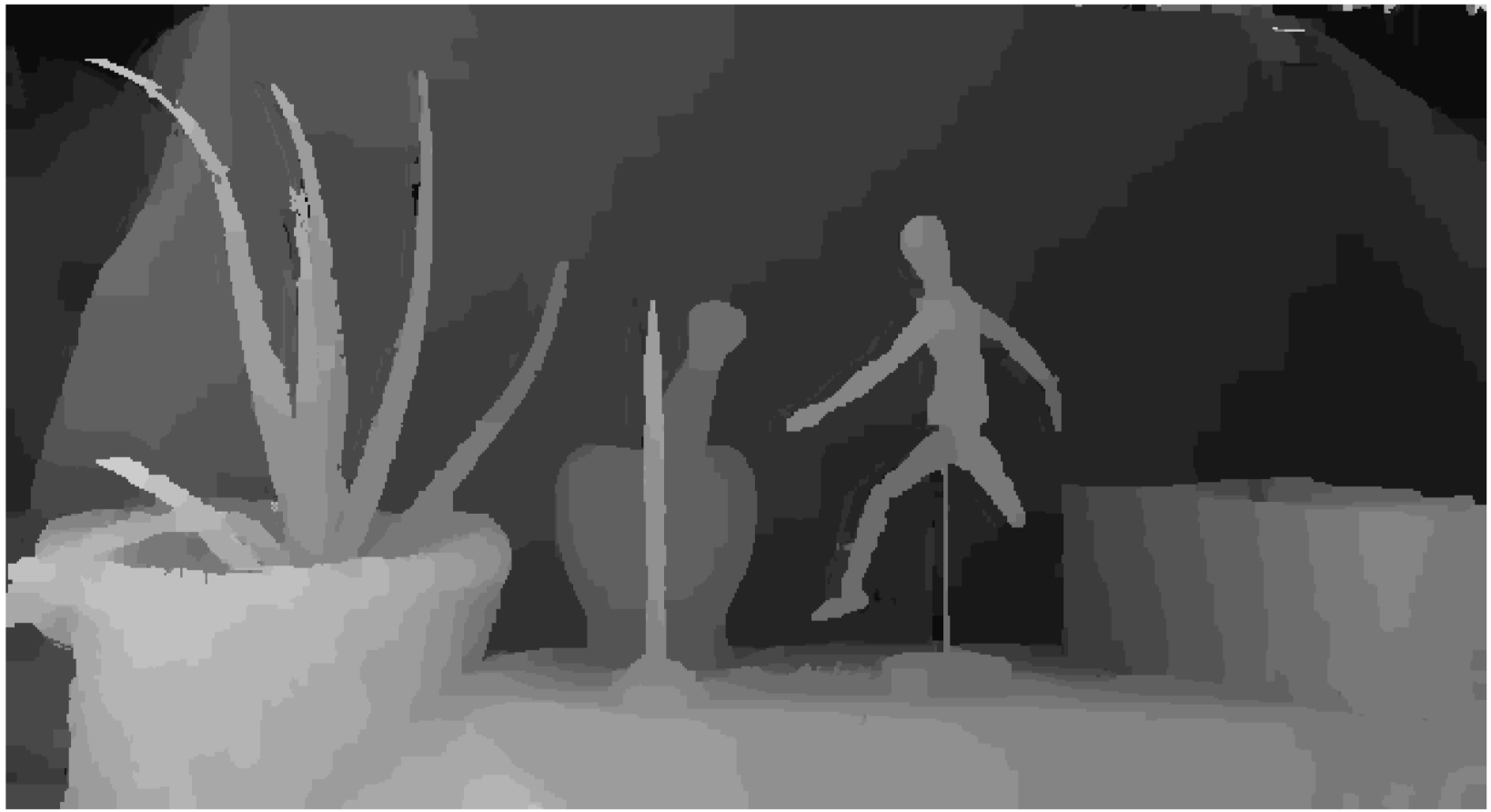
Ground truth

(Seitz)

Segmentation-based Stereo



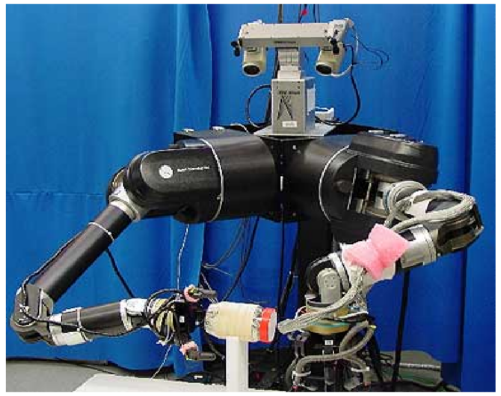
State-of-the-art (before 2015)



Summary of Stereo Vision Methods

- Constraints:
  - Geometry: epipolar constraint
  - Photometric: brightness constancy constraint
  - Ordering: only partially true
  - Smoothness of depth map: only partly true.

Applications of Stereo Vision in Robotics



Reading

- Computer Vision: Algorithms and Applications, Chapter 12