Start Recording

3D Vision, Depth and Stereo Estimation

Harpreet S Sawhney

Microsoft / Vision & Mixed Reality

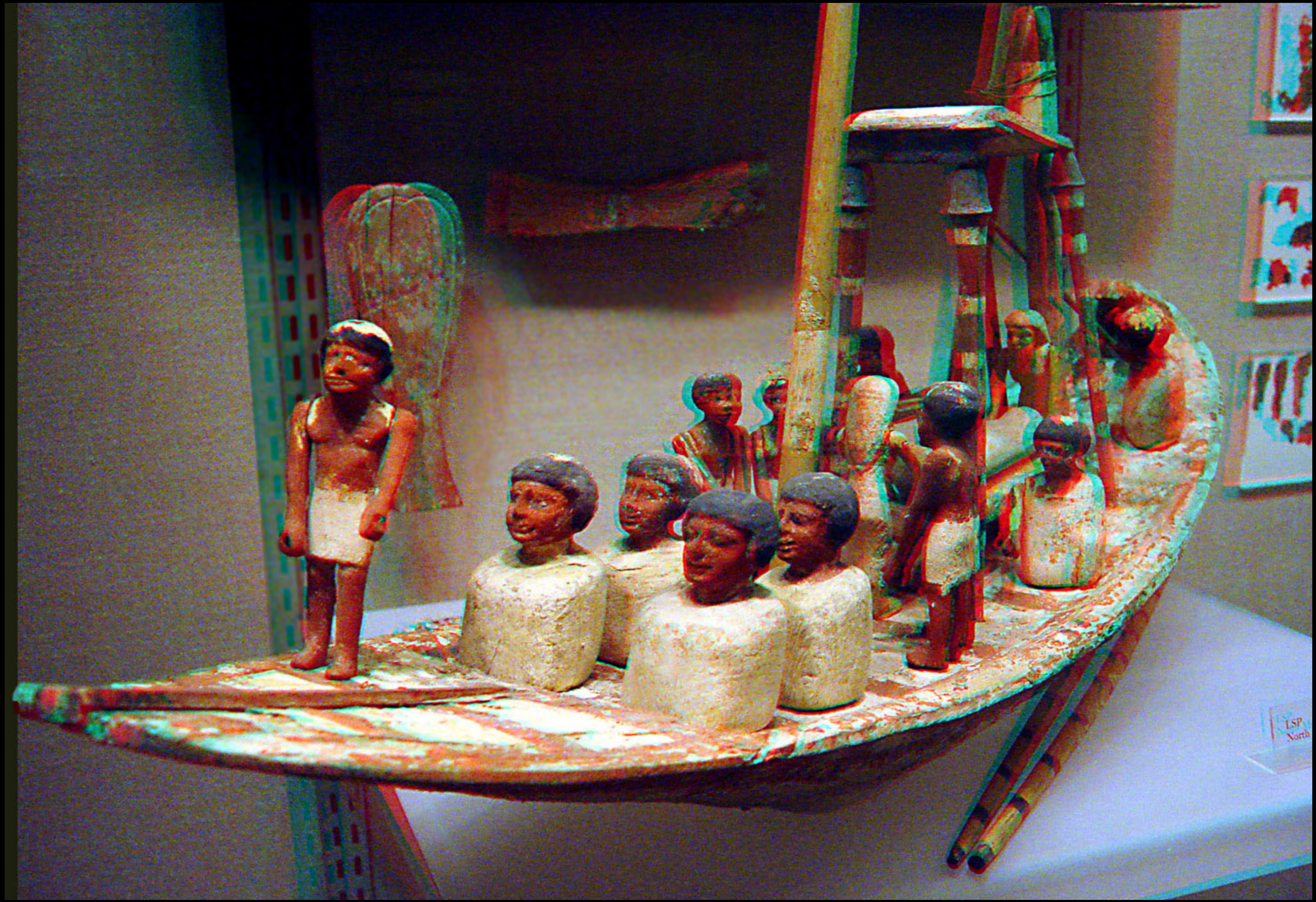
May 19th , 2020

We live, work and play in a 3D World…

…and evolved to perceive in 3D

Cues for 3D / Depth Perception in Our Vision System

Binocular Disparities via Anaglyphs



- Egyptian Ship Model rendered as an Anaglyph to give 3D perception with red-green glasses

Binocular Disparities via Autostereograms

Thinker

Monocular Perspective Cues: Julian Beever's Pavement drawings

Who's Painting Who ? Historians have long wondered just how Michaelangelo managed to paint his famous Mona Lisa on the wall of the Sistine Chapel. Here a possible method is demonstrated.

Batman and Robinto the rescue. (London, UK)

3D Visual Illusions via Texture

https://www.beautifullife.info/urban-design/10-crazy-3d-optical-illusions-will-blow-mind/

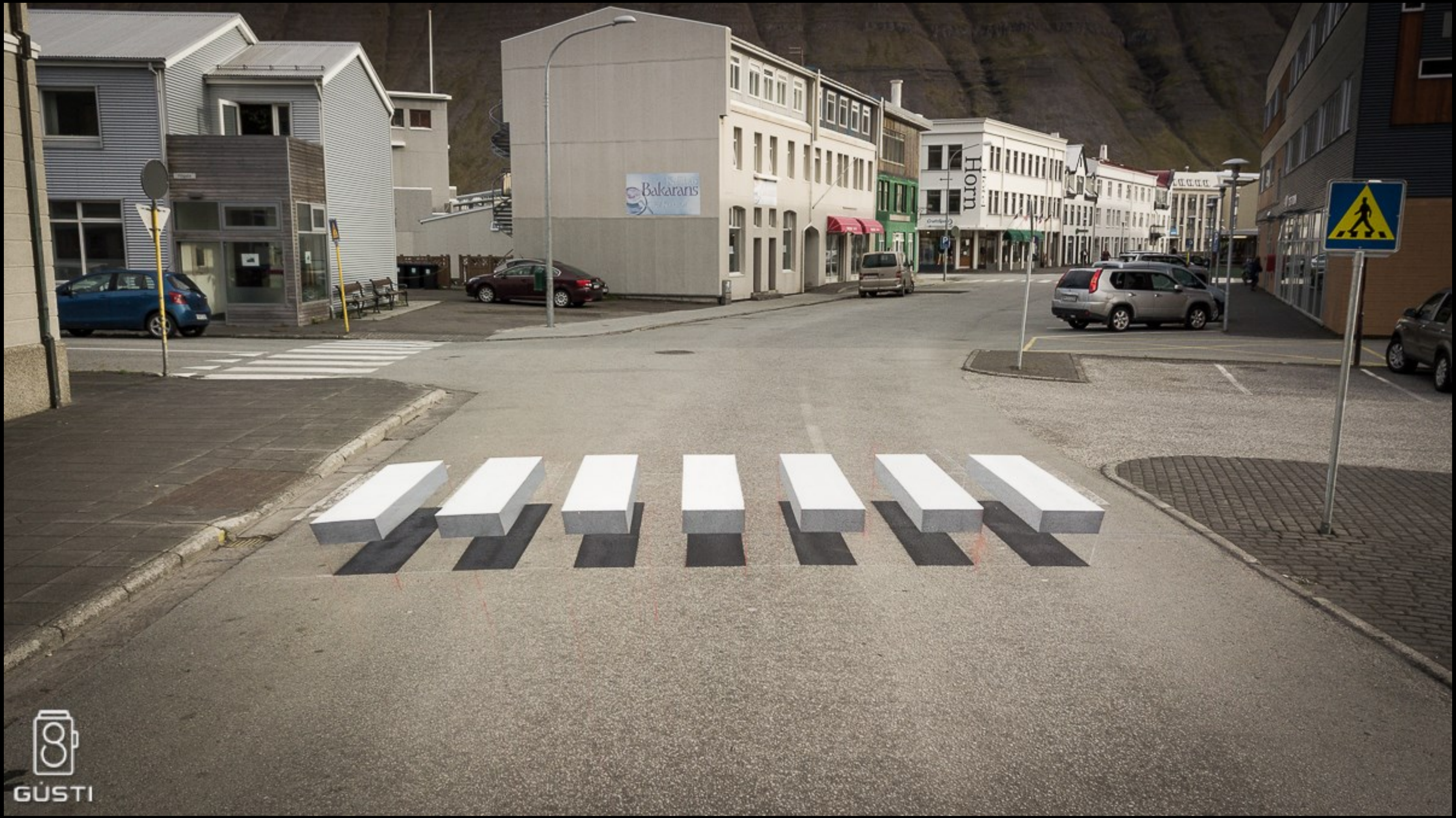
And if you do believe your eyes…

https://www.beautifullife.info/urban-design/10-crazy-3d-optical-illusions-will-blow-mind/

3D Perception via Perspective & Shading

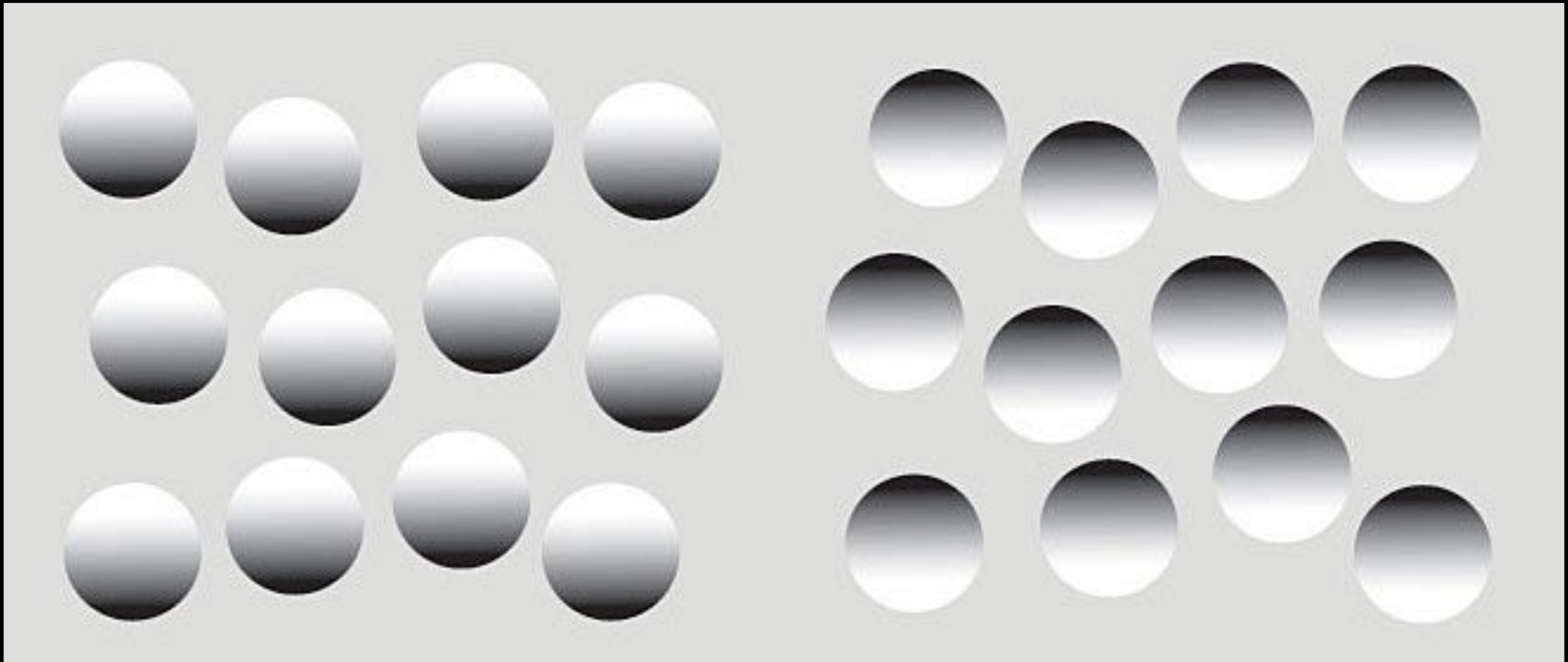
https://www.beautifullife.info/urban-design/10-crazy-3d-optical-illusions-will-blow-mind/

https://youtu.be/szJbz-z7iJw



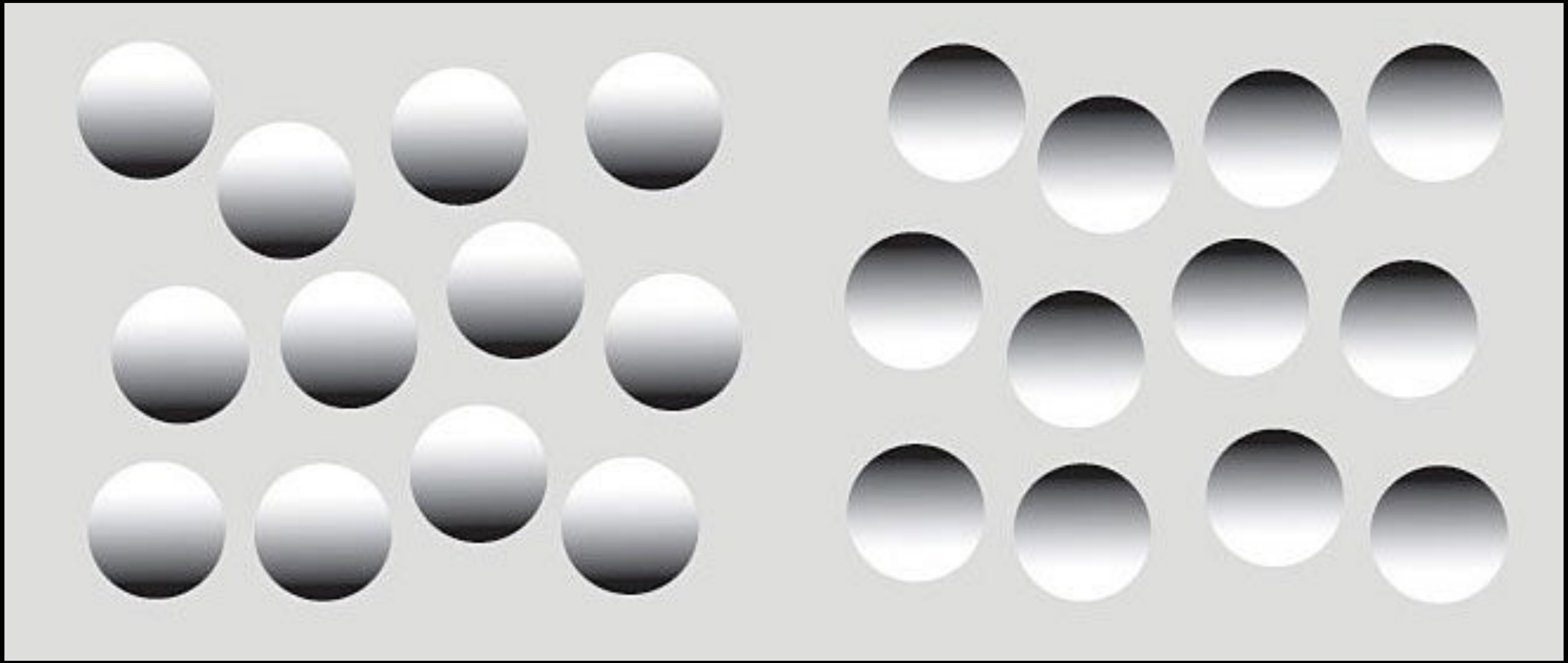
3D Perception via Motion Parallax

3D via Shading



- 3D Perception created

3D via Shading



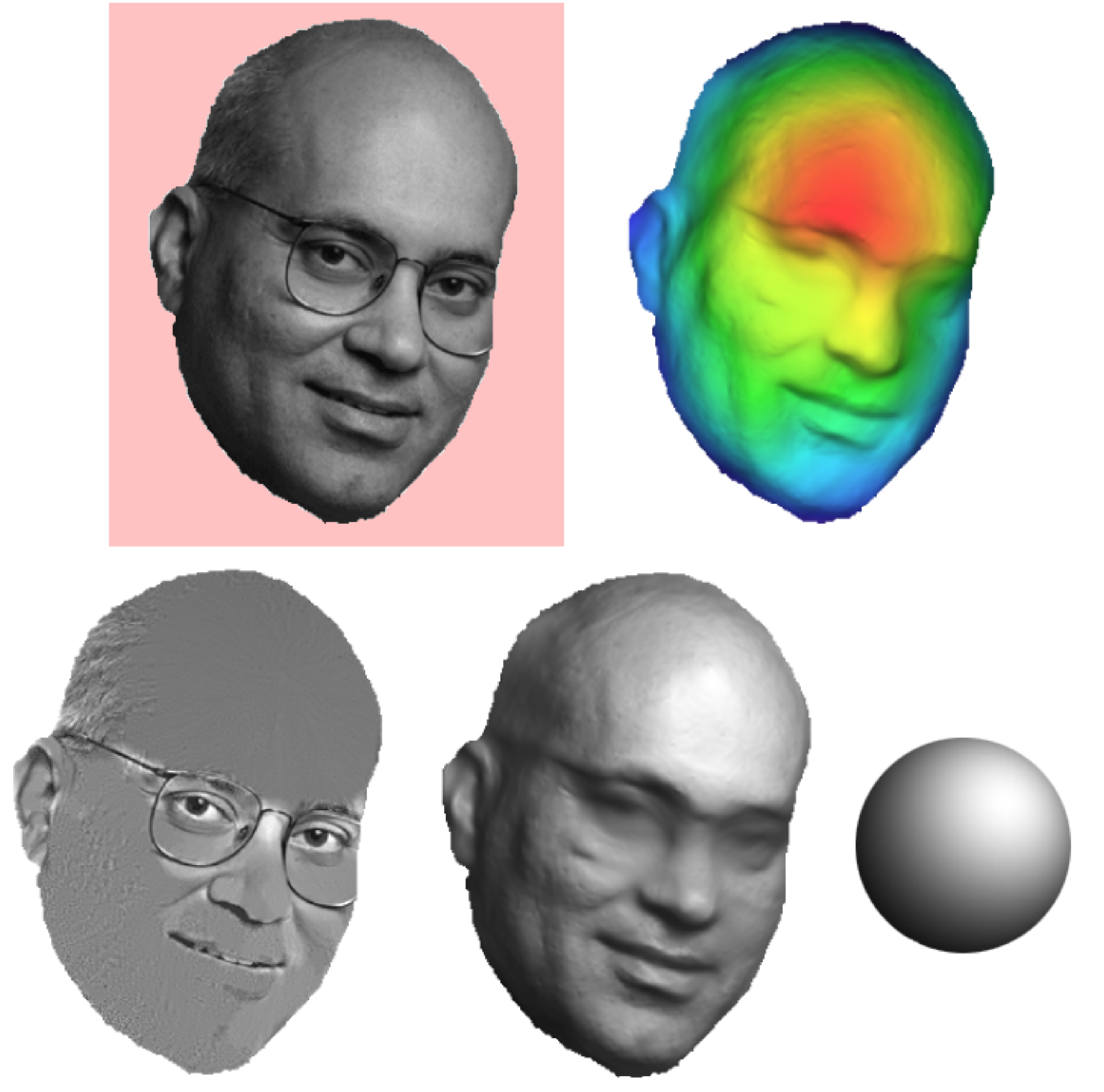
- 3D Perception created

3D via Shading

Shape, Albedo and Illumination

from a

Single Image

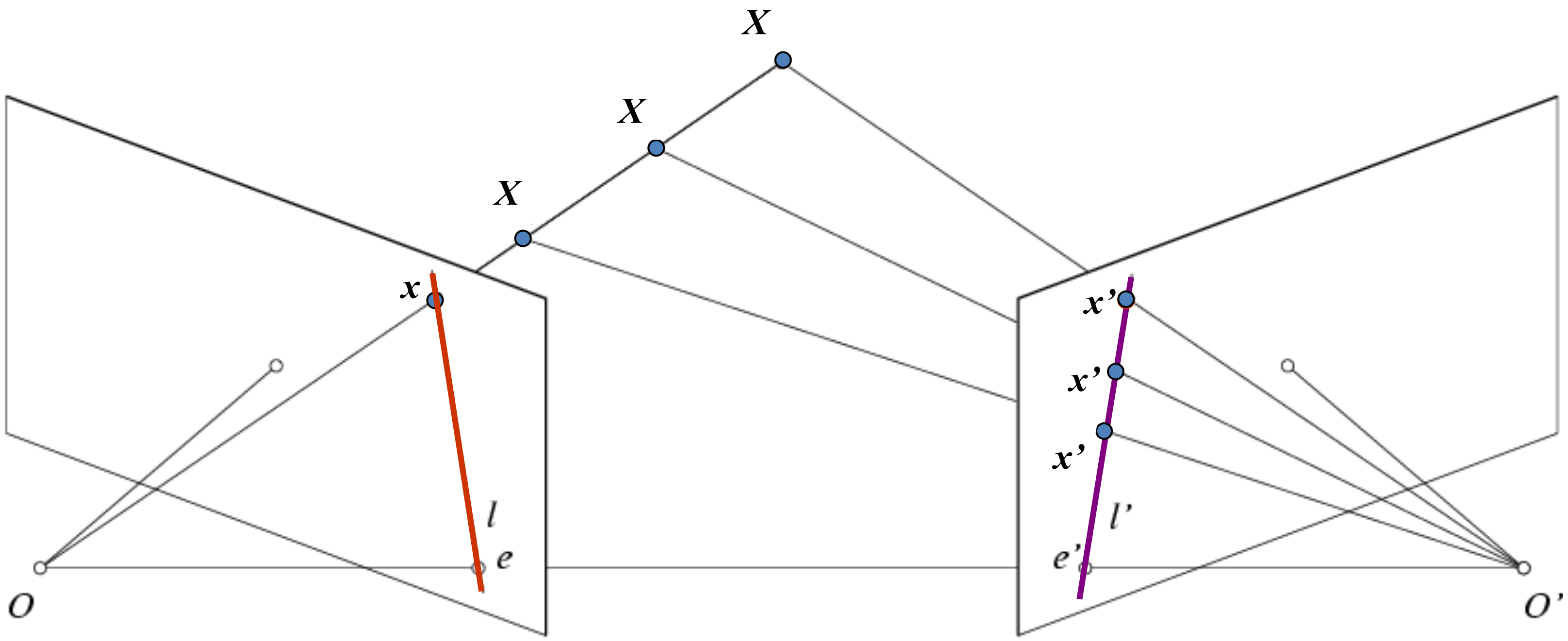


Today’s Lecture

- Two-view Epipolar geometry
  - Relates cameras from two positions
  - Takes us into the realm of Computing 3D from Images
- Binocular Stereo depth estimation
  - Recover depth from two images
- Deep Learning for 3D Estimation
  - Geometric and Photometric Constraints

Via Derek Hoeim

Two View Geometry: Epipolar constraint



Epipolar Plane:

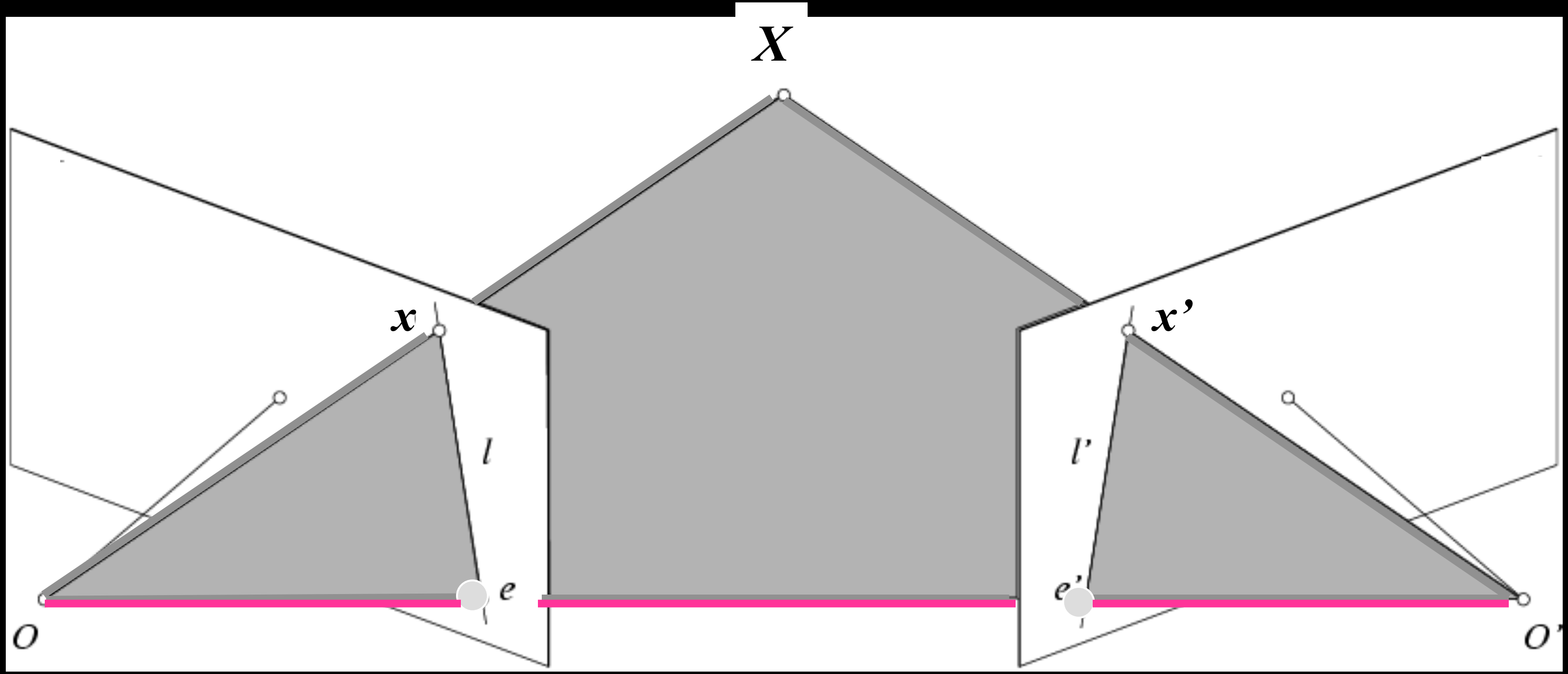
Displacement vector OO', and the two corresponding view rays Ox and Ox' form a plane

Correspondence Constraint between x and x':

Potential matches for *x* have to lie on the corresponding line *l'*.
Potential matches for *x'* have to lie on the corresponding line *l*.

Epipolar Geometry: Notation

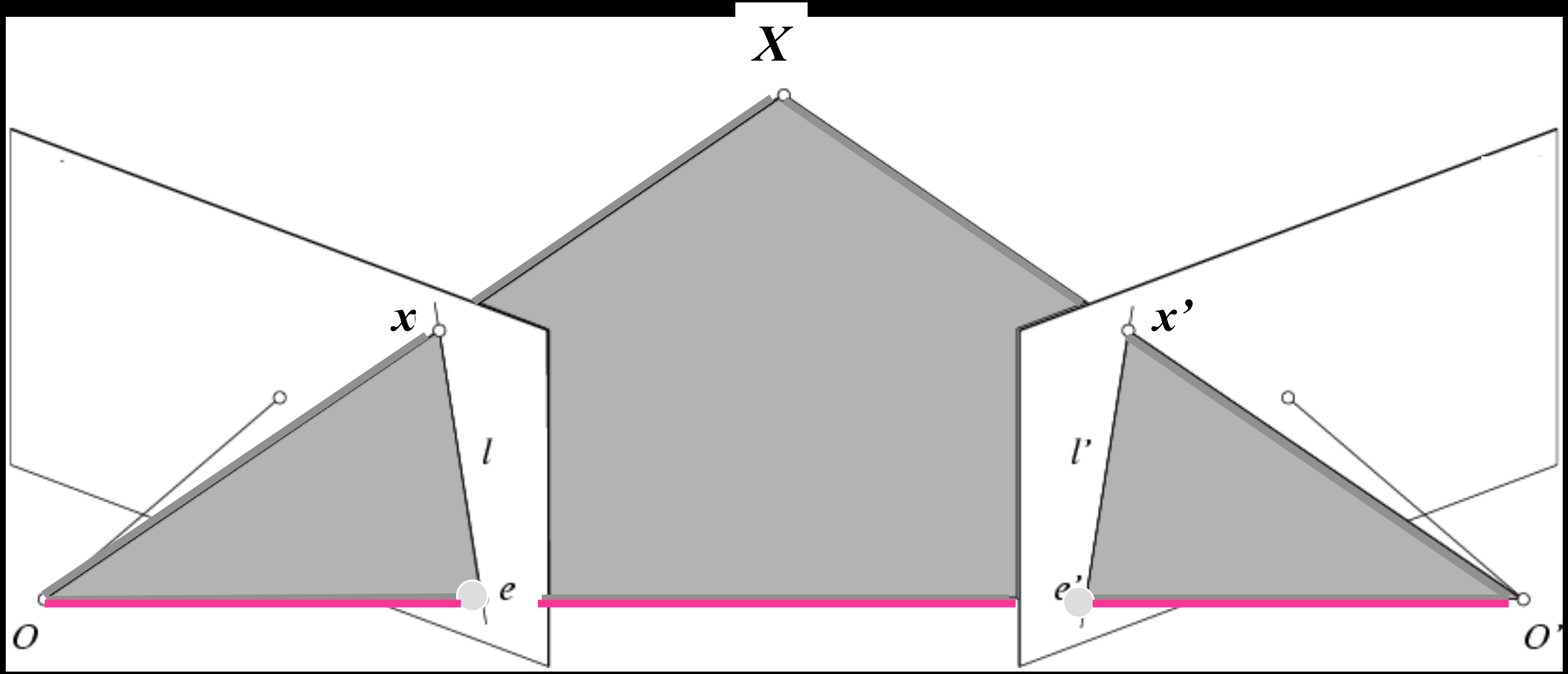
Via Derek Hoeim



- **Baseline** – line connecting the two camera centers
- **Epipoles**
  = intersections of baseline with image planes
  = projections of the other camera center
- **Epipolar Plane** – plane containing baseline (1D family)

Epipolar Geometry: Notation

Via Derek Hoeim



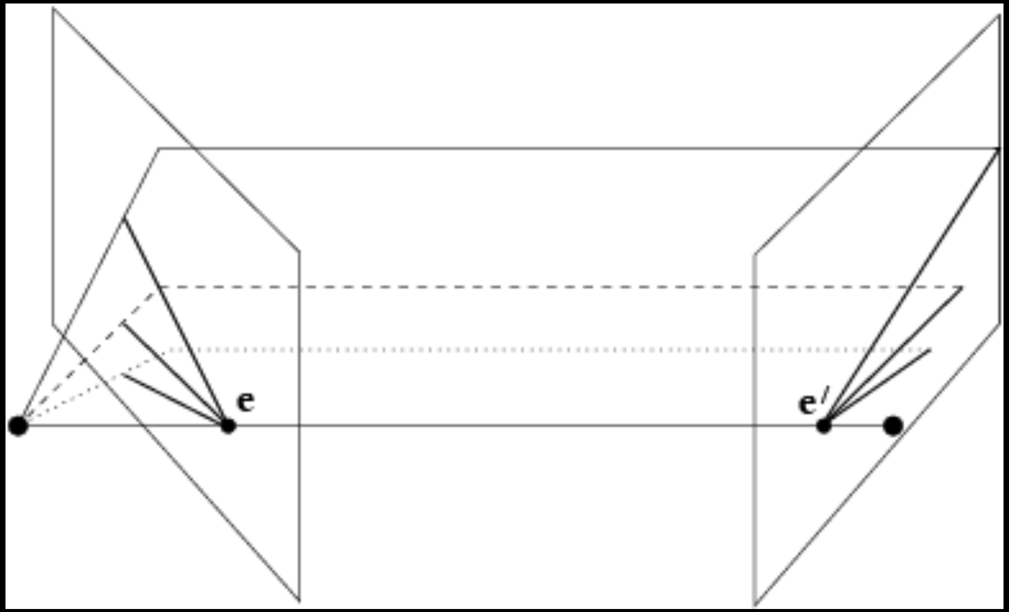
- **Baseline** – line connecting the two camera centers
- **Epipoles**

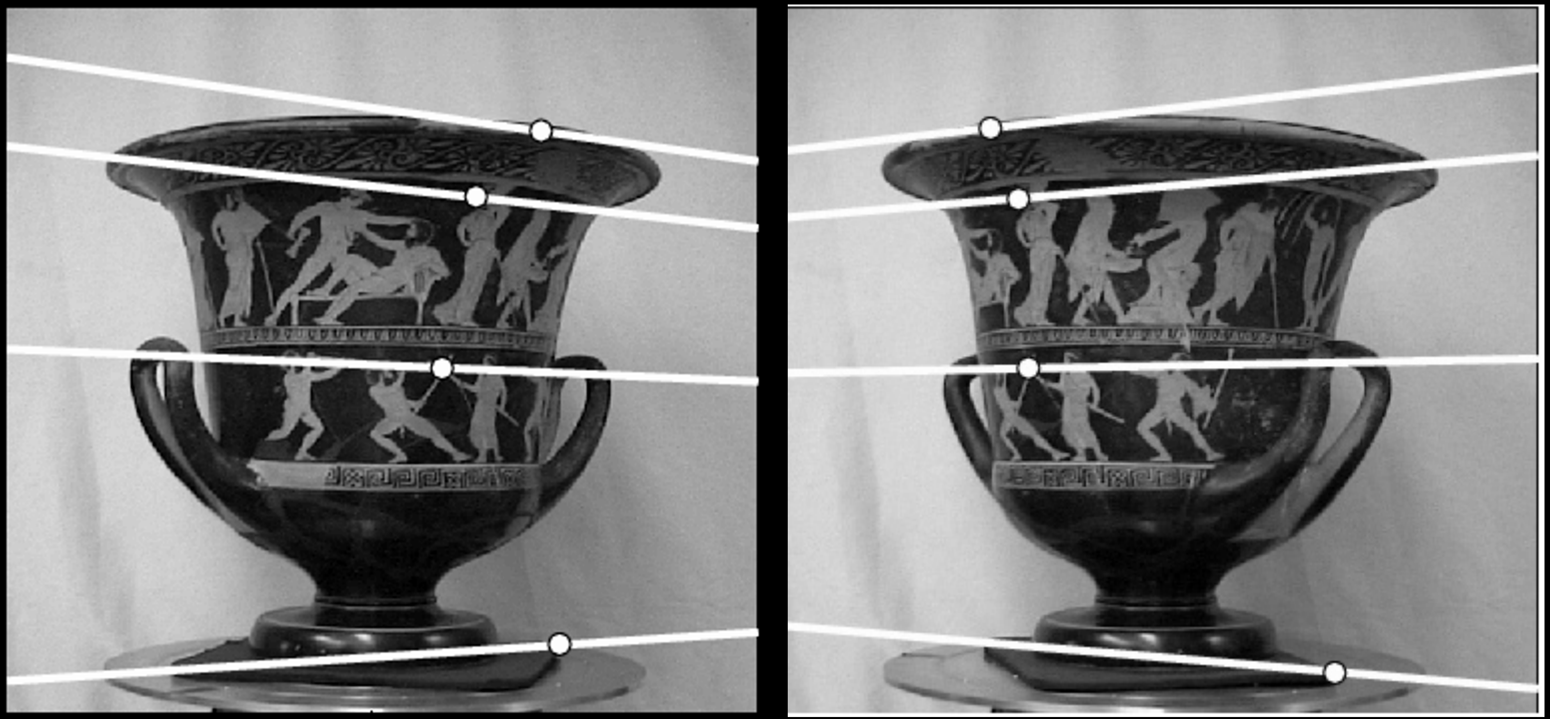
= intersections of baseline with image planes
= projections of the other camera center

- **Epipolar Plane** – plane containing baseline (1D family)
- **Epipolar Lines** - intersections of epipolar plane with image planes (always come in corresponding pairs)

Example: Fixated / Verged "Eyes"

Via Derek Hoeim

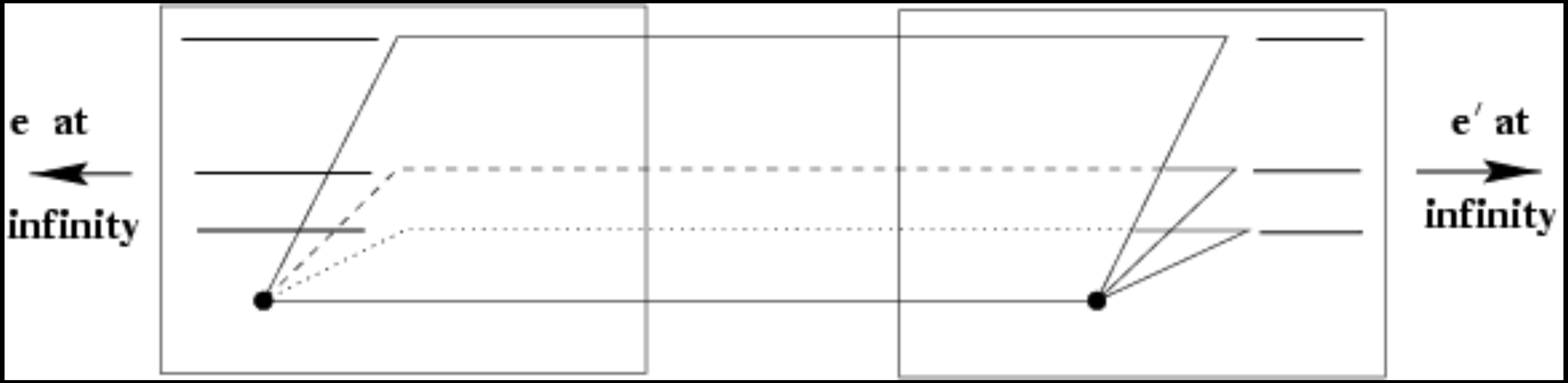




Via Derek Hoeim

Example: Cameras with no displacement in depth

Epipoles at Infinity



Camera Displacement in Depth: Focus of Expansion / Contraction

Via Derek Hoeim

Camera Displacement in Depth: Focus of Expansion / Contraction

Via Derek Hoeim

- Displacement is perpendicular to the image plane
- Epipole is the “focus of expansion” and the principal point

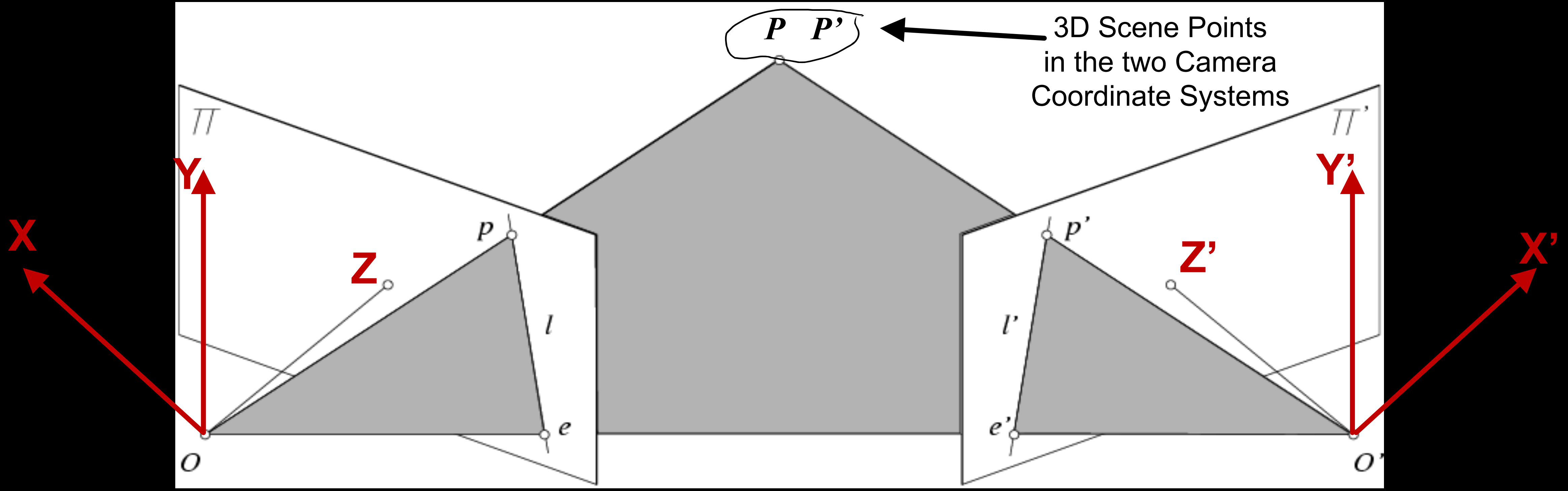
Motion perpendicular to image plane

Via Svetlana L.

http://vimeo.com/48425421

Epipolar Geometry: Calibrated Cameras

Via Derek Hoeim



$p = [x \quad y \quad 1]^T$

$p \approx \lambda P$

$p' \approx \lambda P'$

$p' = [x' \quad y' \quad 1]^T$

$\hat{p} = K^{-1} p$

$\widehat{p'} = K'^{-1} p'$

Homogeneous 2D Pixel Coordinates

3D View Ray in Homogeneous Coordinates

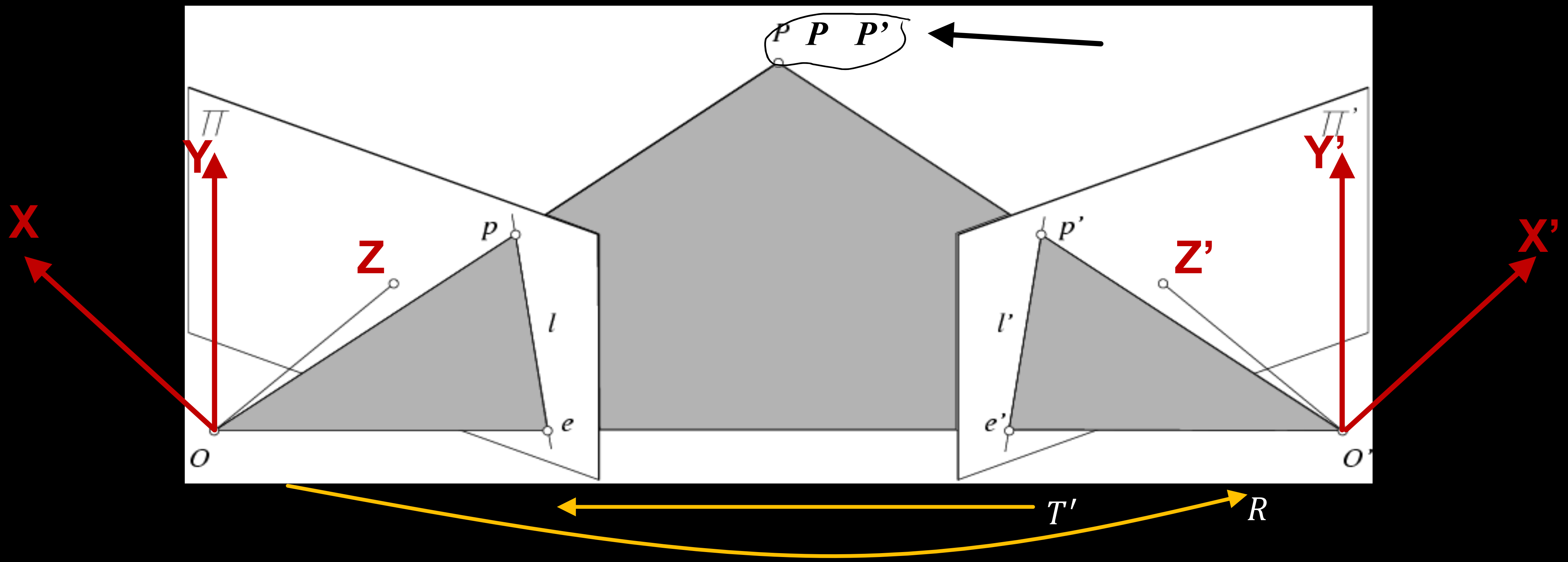
$$\mathrm{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathrm{K}' = \begin{bmatrix} f_x{}' & s' & c_x{}' \\ 0 & f_y{}' & c_y{}' \\ 0 & 0 & 1 \end{bmatrix}$$

Intrinsic Camera Matrices

Epipolar Geometry: Calibrated Cameras

Via Derek Hoeim



$P' = RP + T'$

The vectors $R\hat{p}$, T' and $\widehat{p'}$ are Co-planar: They all lie on the Epipolar Plane.

$\widehat{p'}.[T' \text{x}\ R\hat{p}] = 0 \quad \Leftrightarrow \quad \hat{p}'^T[[T']_x\ R]\hat{p} = 0$

$\hat{p}'^T E\ \hat{p} = 0 \qquad E = [[T']_x\ R]$

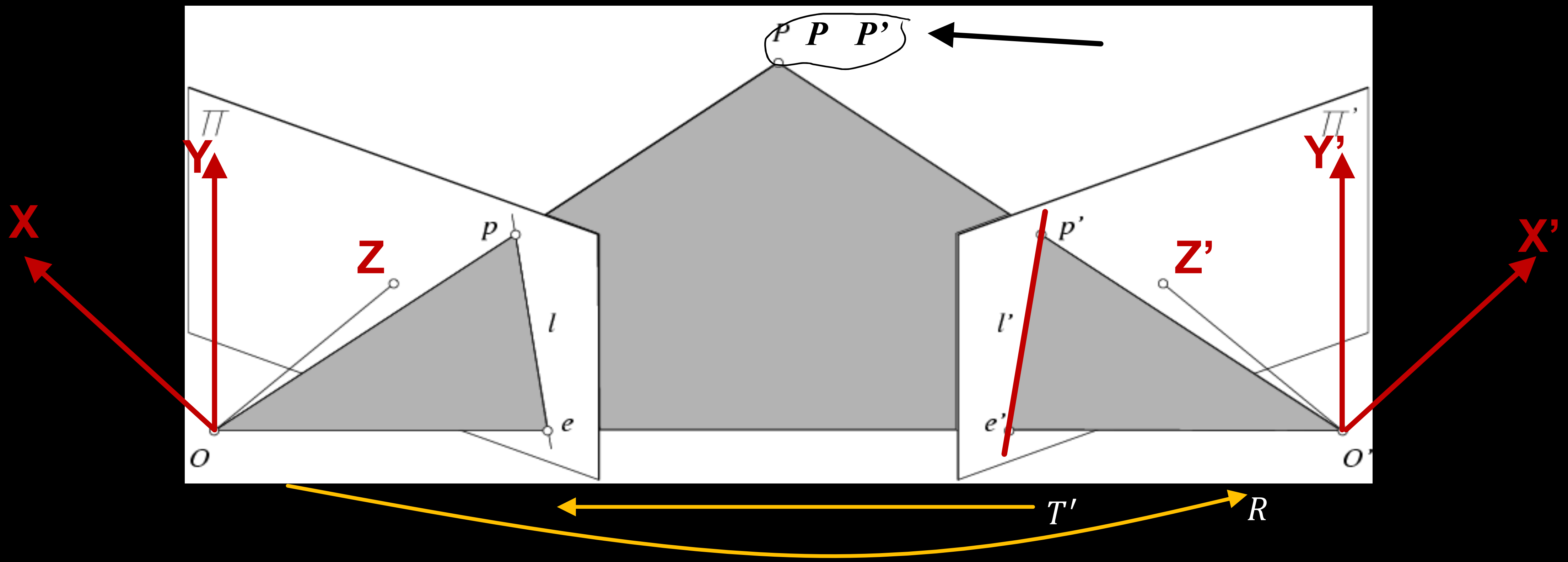
Essential Matrix
(Longuet-Higgins, 1981)

Recall: $\boxed{\mathbf{a}\times\mathbf{b} = \underbrace{\begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix}}\begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = [\mathbf{a}_\times]\mathbf{b}}$

Rank 2 Skew-Symmetric Matrix
Null Vector is the vector **a**

Epipolar Geometry: Properties of the E Matrix

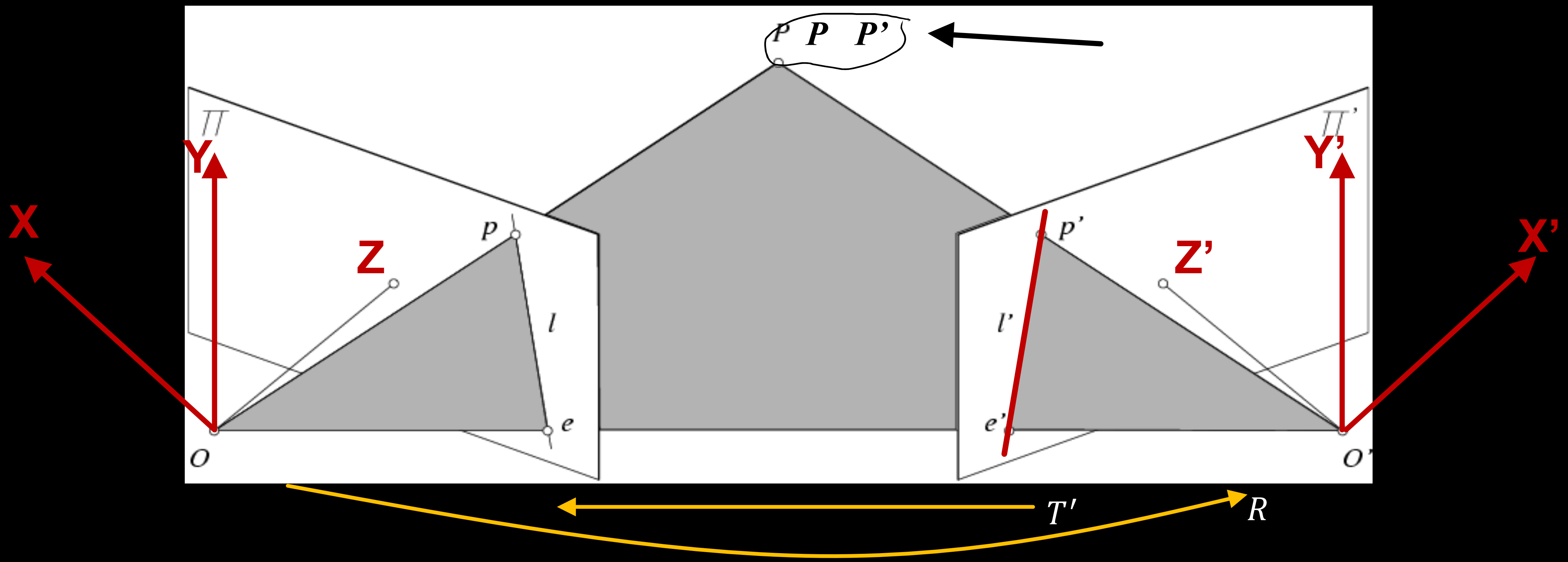
Via Derek Hoeim



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

- $E\, \hat{p}$ is the Epipolar Line (***l'***) in the second image along which the correspondence for $\hat{p}$ lies
- So a point $\hat{p}'$ on this line is co-incident with the line: $\hat{p}'^T\, l' = 0$
- Likewise $E^T\, \hat{p}'$ is the Epipolar Line (***l***) in the first image along which the correspondence for $\hat{p}'$ lies
- Thus $\hat{p}^T\, l = 0$

Epipolar Geometry: Properties of the E Matrix

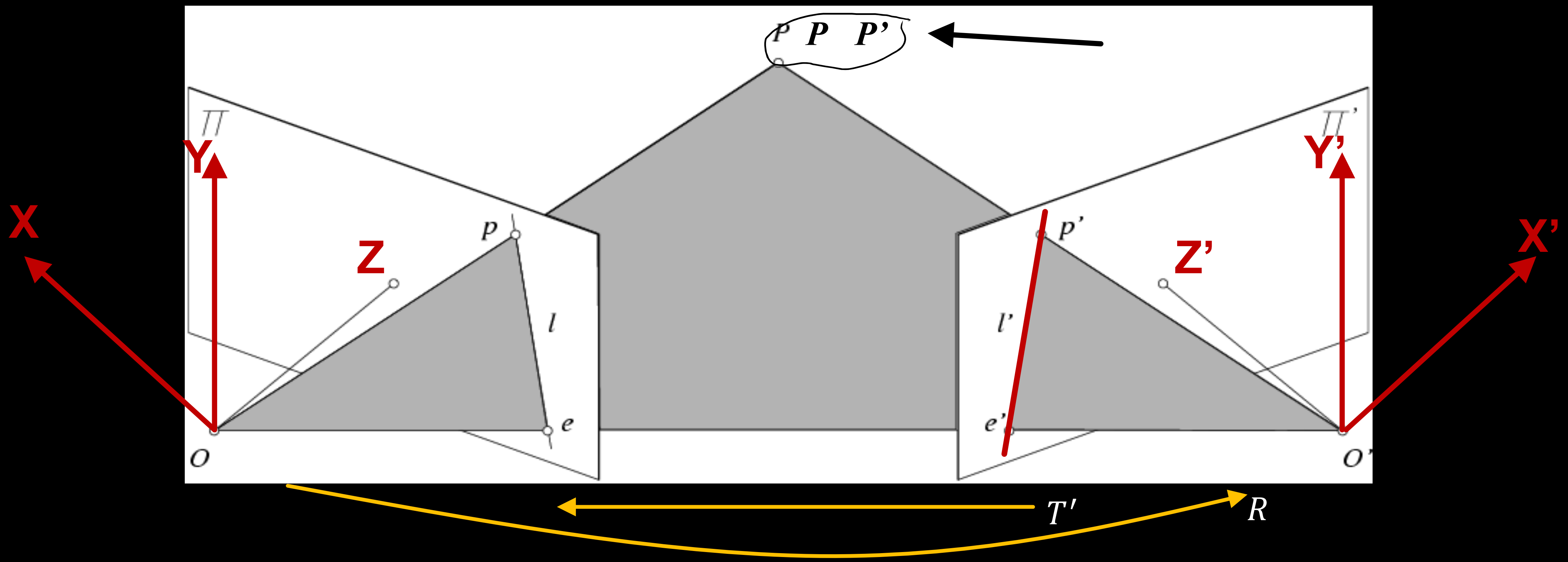


$$\hat{p}'^T E \, \hat{p} = 0$$

- ○ E is Singular with Rank 2 since $[T']_x$ has Rank 2 and R has Rank 3
- ○ $Ee' = 0$ gives the Epipole e' in Image 2 as the projection of first camera's center in the second image (
- ○ $E^T e = 0$ gives the Epipole e in Image 1 as the projection of the second camera's center in the first image
- E has five degrees of freedom (3 for R, 2 for t because it's up to a scale)

Via Derek Hoeim

Epipolar Geometry: Uncalibrated Cameras

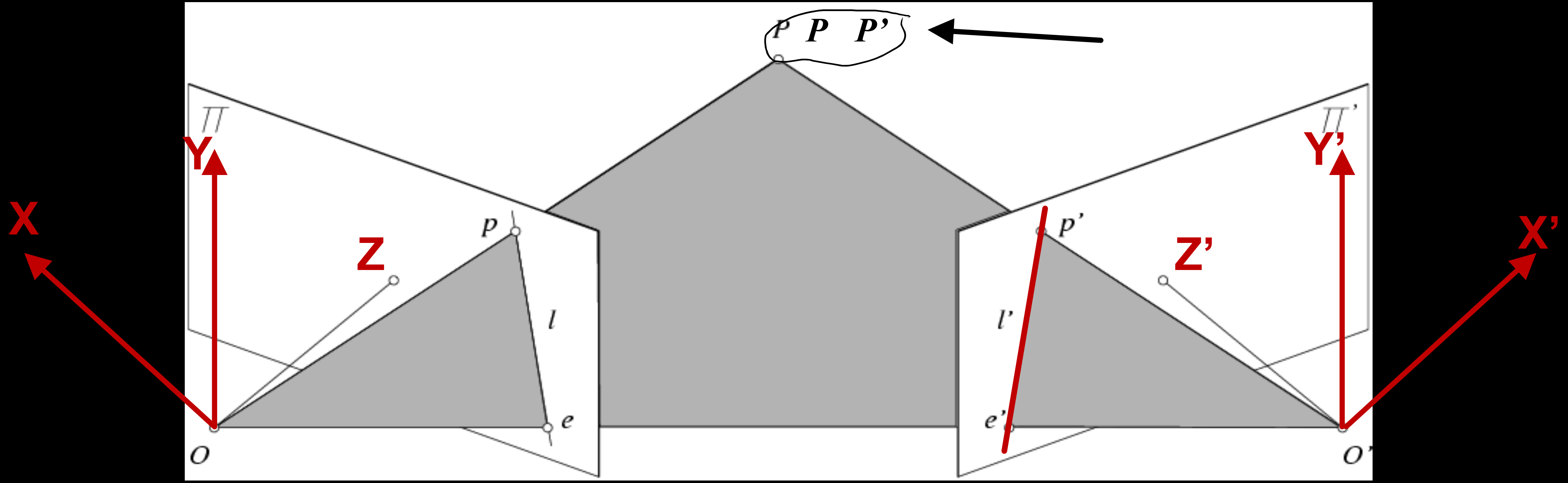


$\hat{p}'^T E \, \hat{p} = 0$ ➔ $p'^T \underbrace{K'^{-T} E \, K^{-1}}_{} p = 0$ ➔ $p'^T F \, p = 0$

Fundamental Matrix
(Faugeras and Luong, 1992)

Epipolar Geometry: Properties of the F Matrix

Via Derek Hoeim



$$p^{\cdot\prime T} K^{\prime -T} E\, K^{-1} p = 0 \quad \rightarrow \quad p^{\cdot\prime T} F\, p = 0$$

- F p is the Epipolar Line (***l'***) in the second image along which the correspondence for p lies
- $F^T\, p'$ is the Epipolar Line (***l***) in the first image along which the correspondence for p' lies
- F is Singular with Rank 2 : Det(F) = 0
- F$e' = 0$ and $F^T e = 0$
- F has seven degrees of freedom: 9 parameters defined upto an arbitrary scale and Det(F) = 0 reduce 2 DoFs

Estimating the Fundamental Matrix

Via Derek Hoeim

- 8-point algorithm
  - Least squares solution using SVD on equations from 8 pairs of correspondences
  - Enforce Det(F)=0 constraint using SVD on F
- 7-point algorithm
  - Use least squares to solve for null space (two vectors) using SVD and 7 pairs of correspondences
  - Solve for linear combination of null space vectors that satisfies Det(F)=0
- Minimize reprojection error
  - Non-linear least squares

Note: estimation of F (or E) is degenerate for a planar scene (Homography)

Estimating the fundamental matrix

Via Svetlana L.

Using 8 Point Correspondences

Via Svetlana L.

The eight-point algorithm: Each correspondence gives one equation

$$\mathrm{x} = (u, v, 1)^T, \quad \mathrm{x}' = (u', v', 1)$$

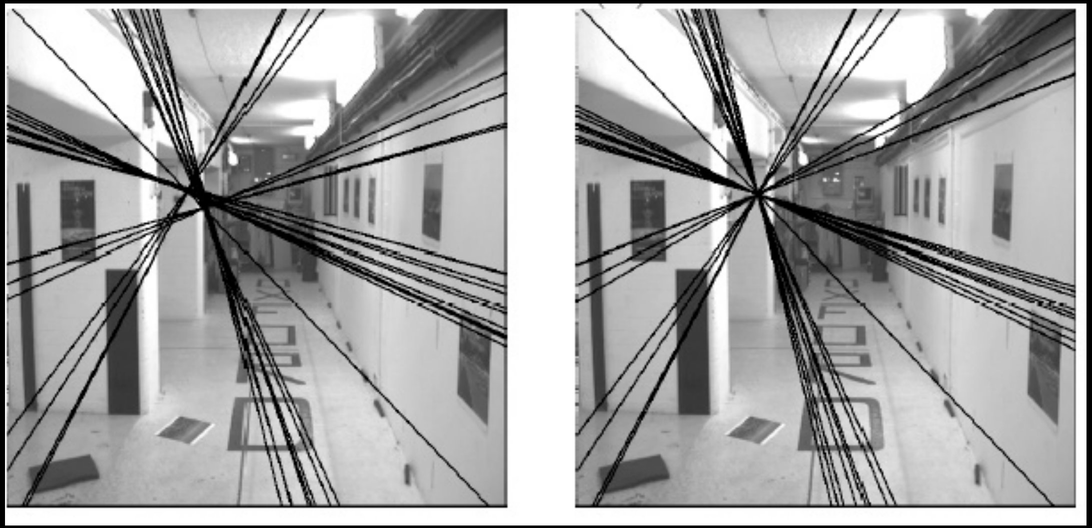
$$[u' \quad v' \quad 1] \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0$$

$$[u'u \quad u'v \quad u' \quad v'u \quad v'v \quad v' \quad u \quad v \quad 1] \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0$$

Solve homogeneous linear system using eight or more matches

Enforce rank-2 constraint

(take SVD of ***F*** and throw out the smallest singular value)



Problem with eight-point algorithm

Via Svetlana L.

$$\begin{bmatrix} u'u & u'v & u' & v'u & v'v & v' & u & v \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \end{bmatrix} = -1$$

F[3][3] has been set to 1 to remove scale ambiguity

Problem with eight-point algorithm

Via Svetlana L.

$$\begin{bmatrix} 250906.36 & 183269.57 & 921.81 & 200931.10 & 146766.13 & 738.21 & 272.19 & 198.81 \\ 2692.28 & 131633.03 & 176.27 & 6196.73 & 302975.59 & 405.71 & 15.27 & 746.79 \\ 416374.23 & 871684.30 & 935.47 & 408110.89 & 854384.92 & 916.90 & 445.10 & 931.81 \\ 191183.60 & 171759.40 & 410.27 & 416435.62 & 374125.90 & 893.65 & 465.99 & 418.65 \\ 48988.86 & 30401.76 & 57.89 & 298604.57 & 185309.58 & 352.87 & 846.22 & 525.15 \\ 164786.04 & 546559.67 & 813.17 & 1998.37 & 6628.15 & 9.86 & 202.65 & 672.14 \\ 116407.01 & 2727.75 & 138.89 & 169941.27 & 3982.21 & 202.77 & 838.12 & 19.64 \\ 135384.58 & 75411.13 & 198.72 & 411350.03 & 229127.78 & 603.79 & 681.28 & 379.48 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \end{bmatrix} = -1$$

- Poor numerical conditioning (Remember very eccentric level curves from Gradient Descent)
  - $\lambda_{max} / \lambda_{min}$ >> 1
- Can be fixed by rescaling the data (“Circling the Ellipses”)

The normalized eight-point algorithm

Via Svetlana L.

(Hartley, 1995)

- Center the image data at the origin, and scale it so the mean squared distance between the origin and the data points is 2 pixels
- Use the eight-point algorithm to compute ***F*** from the normalized points
- Enforce the rank-2 constraint (for example, take SVD of ***F*** and throw out the smallest singular value)
- Transform fundamental matrix back to original units: if ***T*** and ***T'*** are the normalizing transformations in the two images, than the fundamental matrix in original coordinates is $\boldsymbol{T'^T F T}$

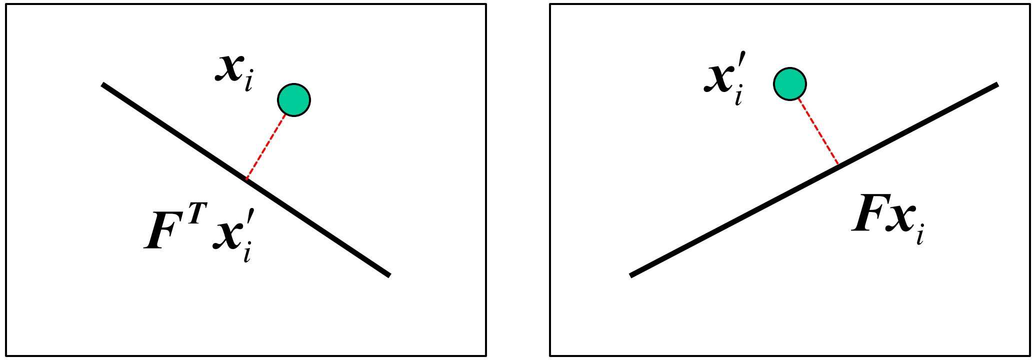
Nonlinear estimation

- Linear estimation minimizes the sum of squared *algebraic* distances between points $\boldsymbol{x}'_i$ and epipolar lines $\boldsymbol{F}\,\boldsymbol{x}_i$ (or points $\boldsymbol{x}_i$ and epipolar lines $\boldsymbol{F}^T\boldsymbol{x}'_i$):

$$\sum_{i=1}^{N} (\boldsymbol{x}_i'^T \boldsymbol{F}\,\boldsymbol{x}_i)^2$$

- Nonlinear approach: minimize sum of squared *geometric* distances

$$\sum_{i=1}^{N} \left[\mathrm{d}^2(\boldsymbol{x}'_i, \boldsymbol{F}\,\boldsymbol{x}_i) + \mathrm{d}^2(\boldsymbol{x}_i, \boldsymbol{F}^T \boldsymbol{x}'_i) \right]$$



Via Svetlana L.

Nonlinear estimation

- Linear estimation minimizes the sum of squared *algebraic* distances between points $\boldsymbol{x}'_i$ and epipolar lines $\boldsymbol{F}\,\boldsymbol{x}_i$ (or points $\boldsymbol{x}_i$ and epipolar lines $\boldsymbol{F}^T\boldsymbol{x}'_i$):

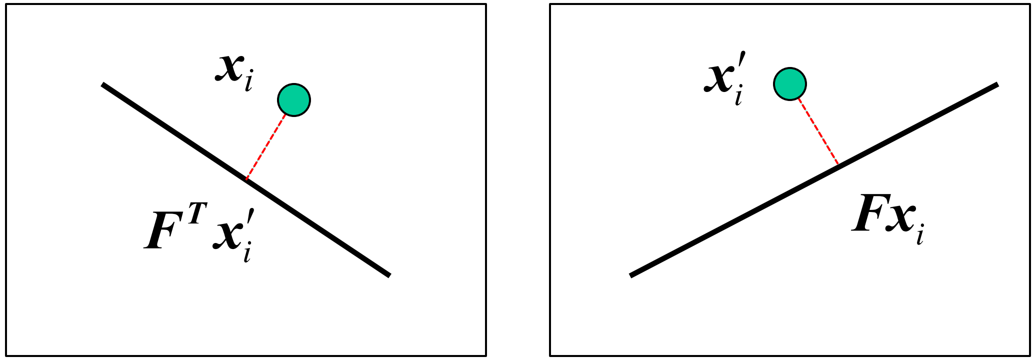
$$\sum_{i=1}^{N}(\boldsymbol{x}_i'^T \boldsymbol{F}\,\boldsymbol{x}_i)^2$$

Initial Guess

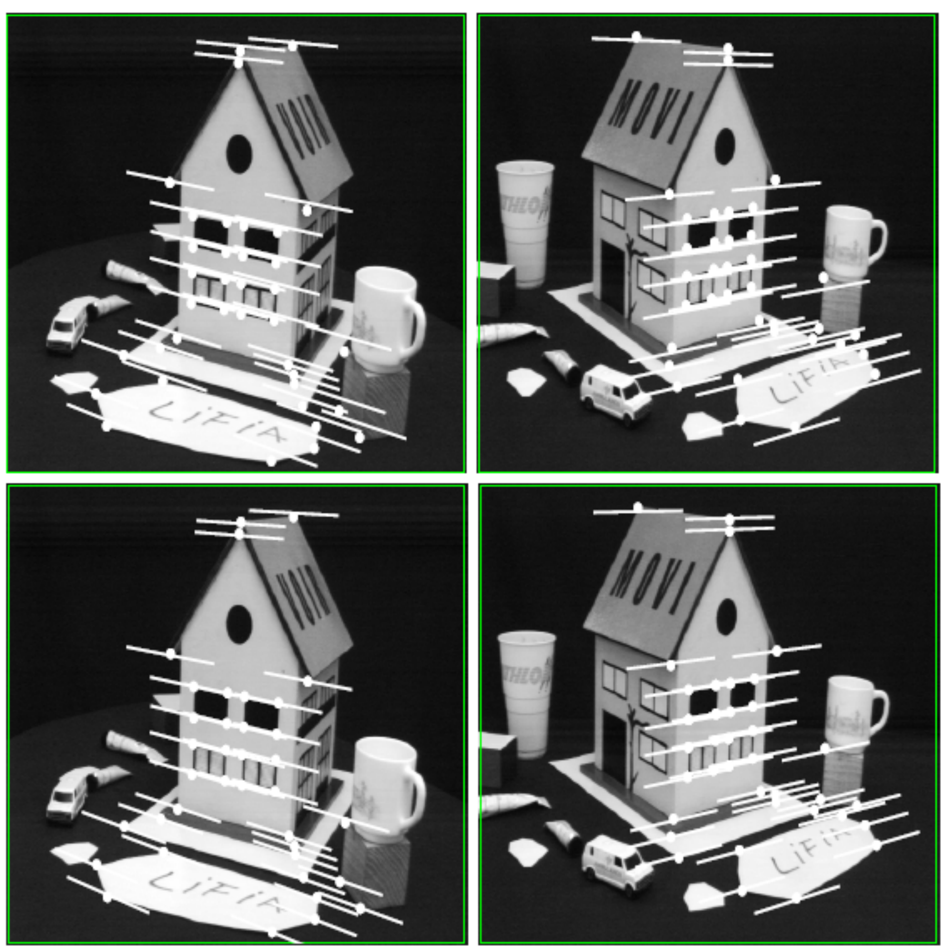
- Nonlinear approach: minimize sum of squared *geometric* distances

$$\sum_{i=1}^{N}\left[\mathrm{d}^2(\boldsymbol{x}'_i, \boldsymbol{F}\,\boldsymbol{x}_i) + \mathrm{d}^2(\boldsymbol{x}_i, \boldsymbol{F}^T\boldsymbol{x}'_i)\right]$$

Remember Levenberg-Marquardt?

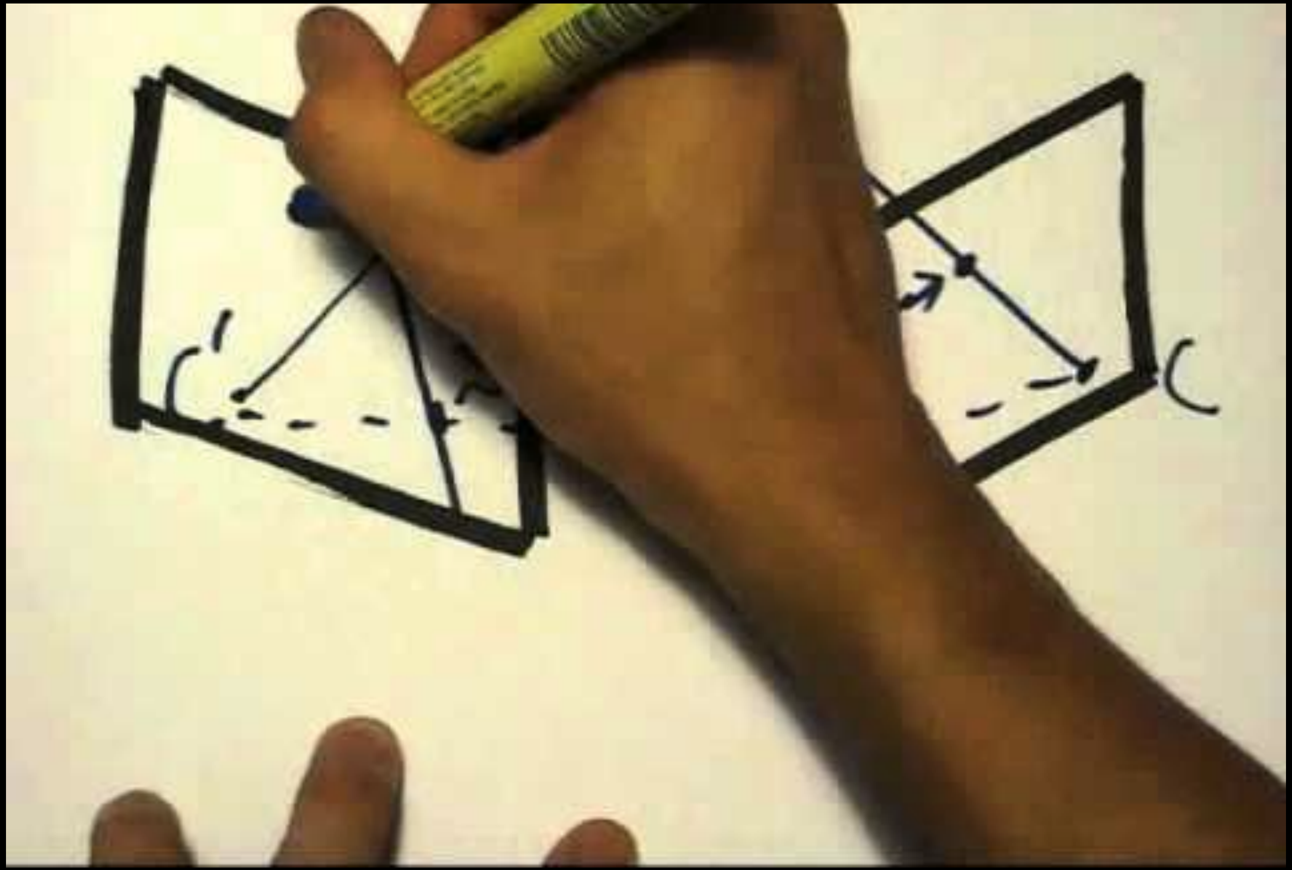


Comparison of estimation algorithms



	8-point	Normalized 8-point	Nonlinear least squares
Av. Dist. 1	2.33 pixels	0.92 pixel	0.86 pixel
Av. Dist. 2	2.18 pixels	0.85 pixel	0.80 pixel

The Fundamental Matrix Song

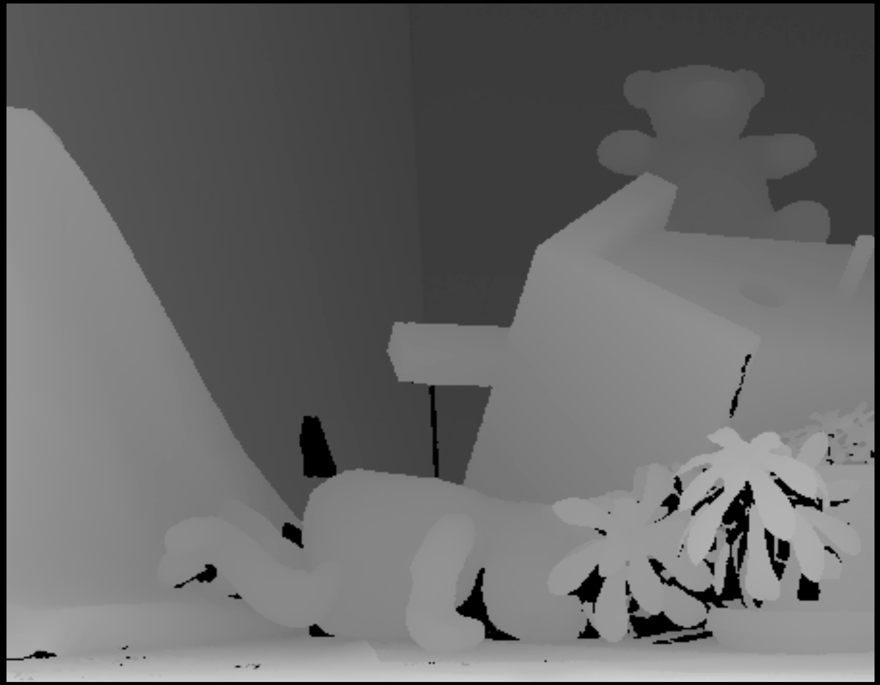


http://danielwedge.com/fmatrix/

Two-View Stereo : 3D Depth from two Views

Many slides adapted from Steve Seitz

Disparity Map



Stereograms

- Humans can fuse pairs of images to get a sensation of depth

Autostereograms: www.magiceye.com

Stereograms

- Humans can fuse pairs of images to get a sensation of depth

Autostereograms: www.magiceye.com

Problem formulation

Via Derek Hoeim

- Given a calibrated binocular stereo pair, fuse it to produce a depth image

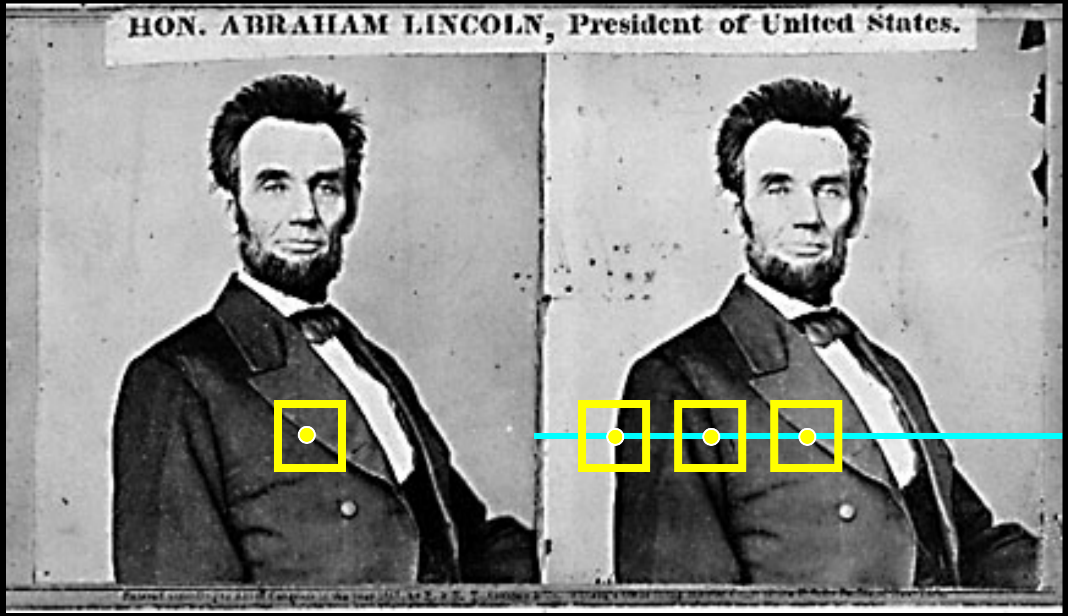
image 1

image 2

Dense depth map

Basic stereo matching algorithm

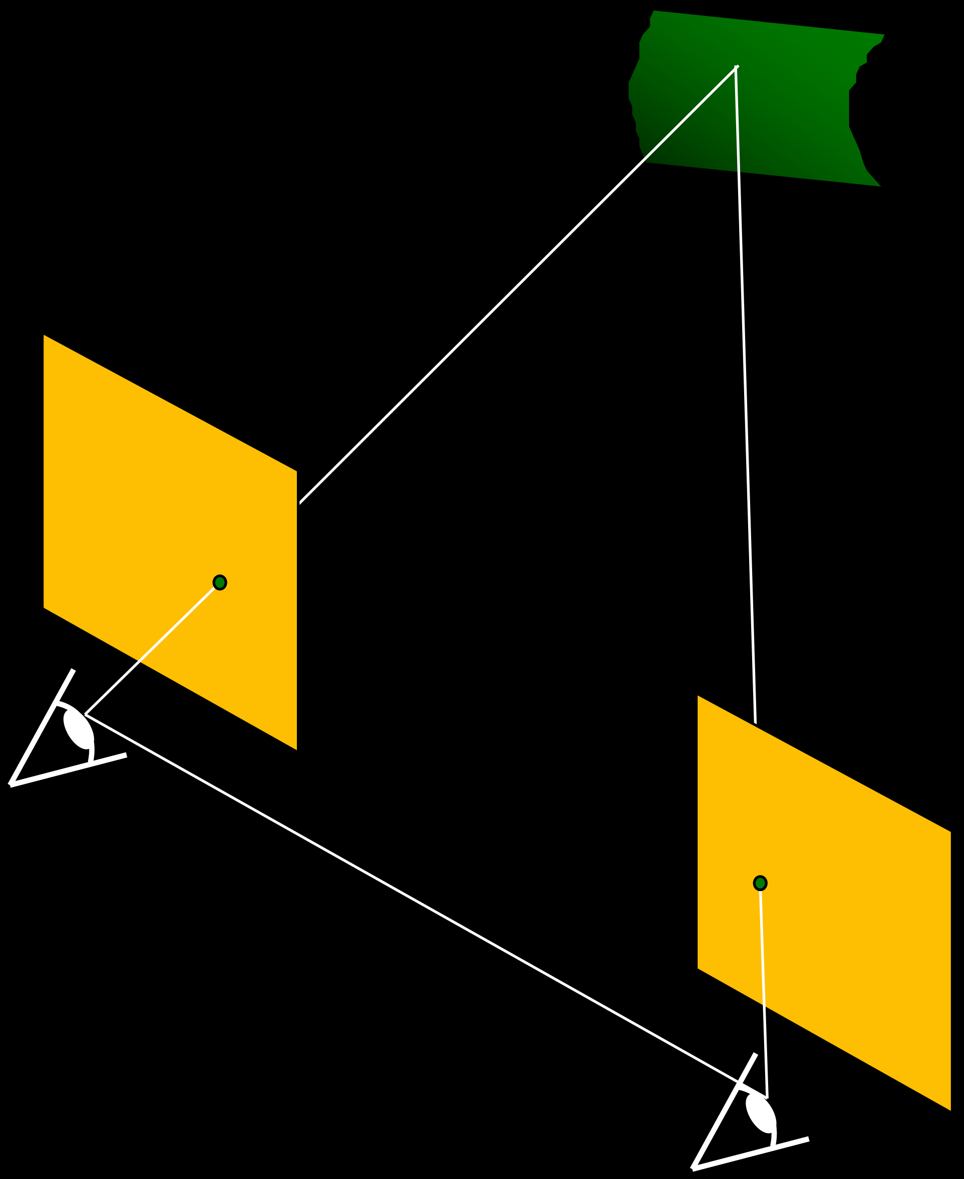
Via Derek Hoeim



- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Examine all pixels on the epipolar line and pick the best match
  - Triangulate the matches to get depth information
- Simplest case: epipolar lines are corresponding scanlines
  - When does this happen?

Simplest Case: Parallel images

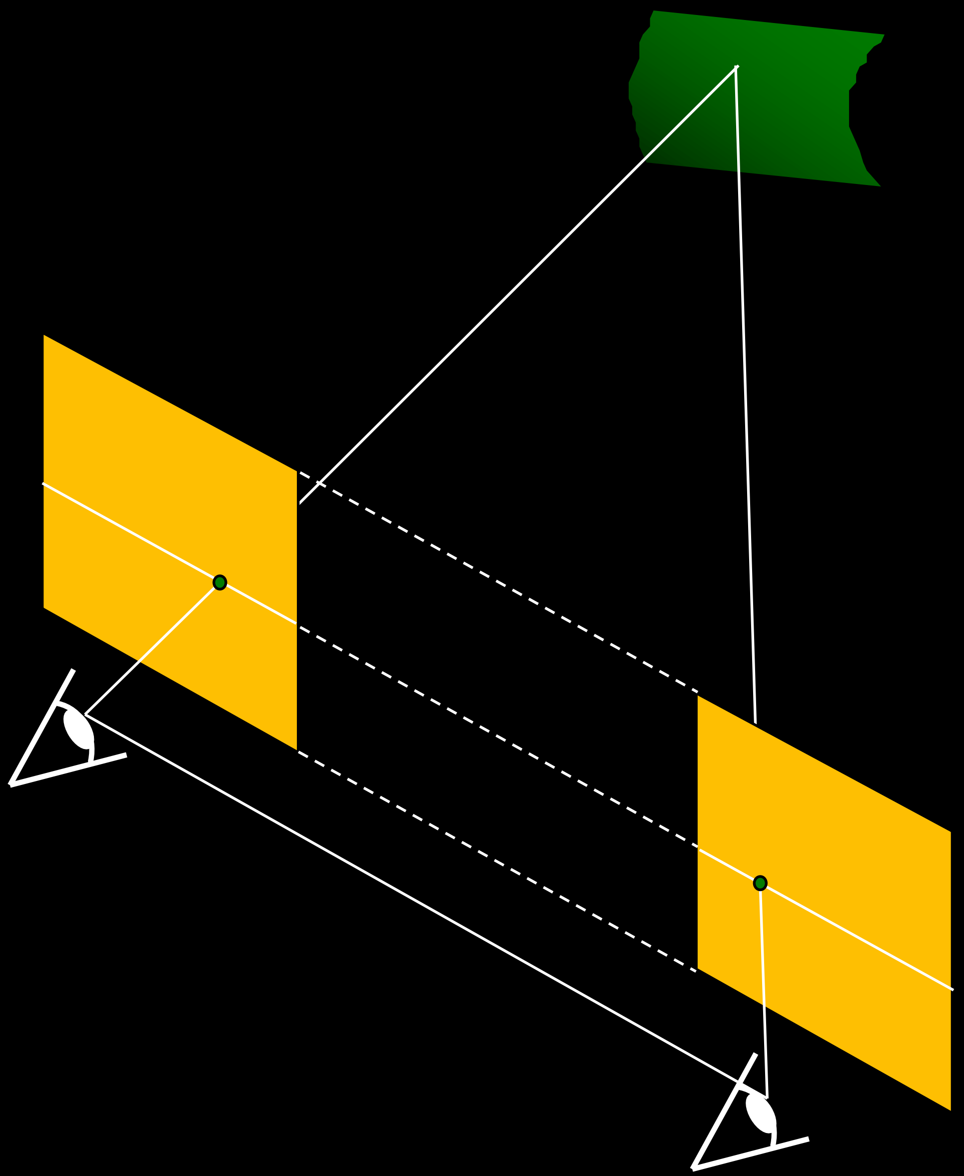
Via Derek Hoeim



- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at same height
- Focal lengths are the same

Simplest Case: Parallel images

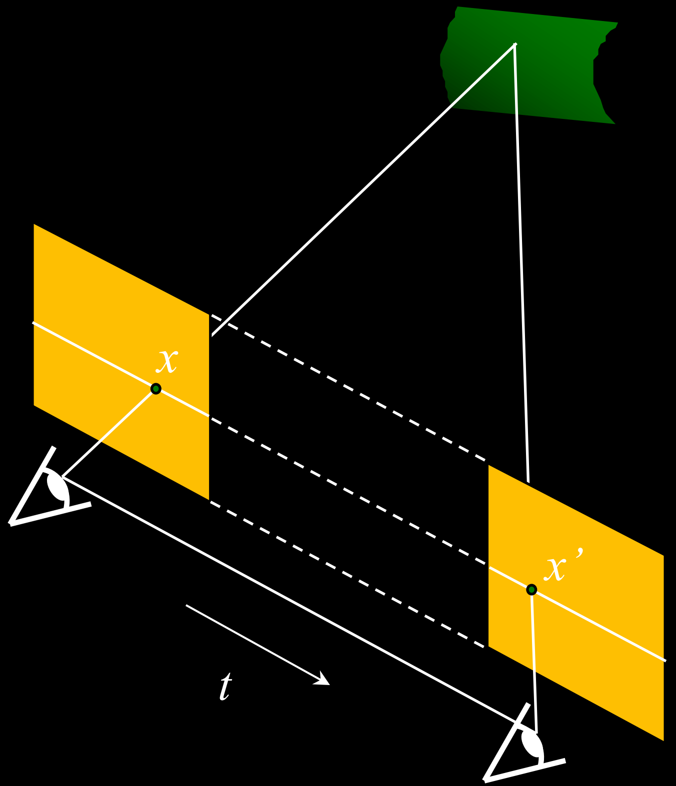
Via Derek Hoeim



- Image planes of cameras are parallel to each other and to the baseline
- Camera centers are at same height
- Focal lengths are the same
- Then epipolar lines fall along the horizontal scan lines of the images

Via Derek Hoeim

Essential matrix for parallel images



Epipolar constraint:

$$\mathrm{x}'^{T}\mathrm{E}\mathrm{x} = 0, \quad \mathrm{E} = [\mathrm{t}_\times]\mathrm{R}$$

$$\boldsymbol{R} = \boldsymbol{I} \qquad \boldsymbol{t} = (T, 0, 0)$$

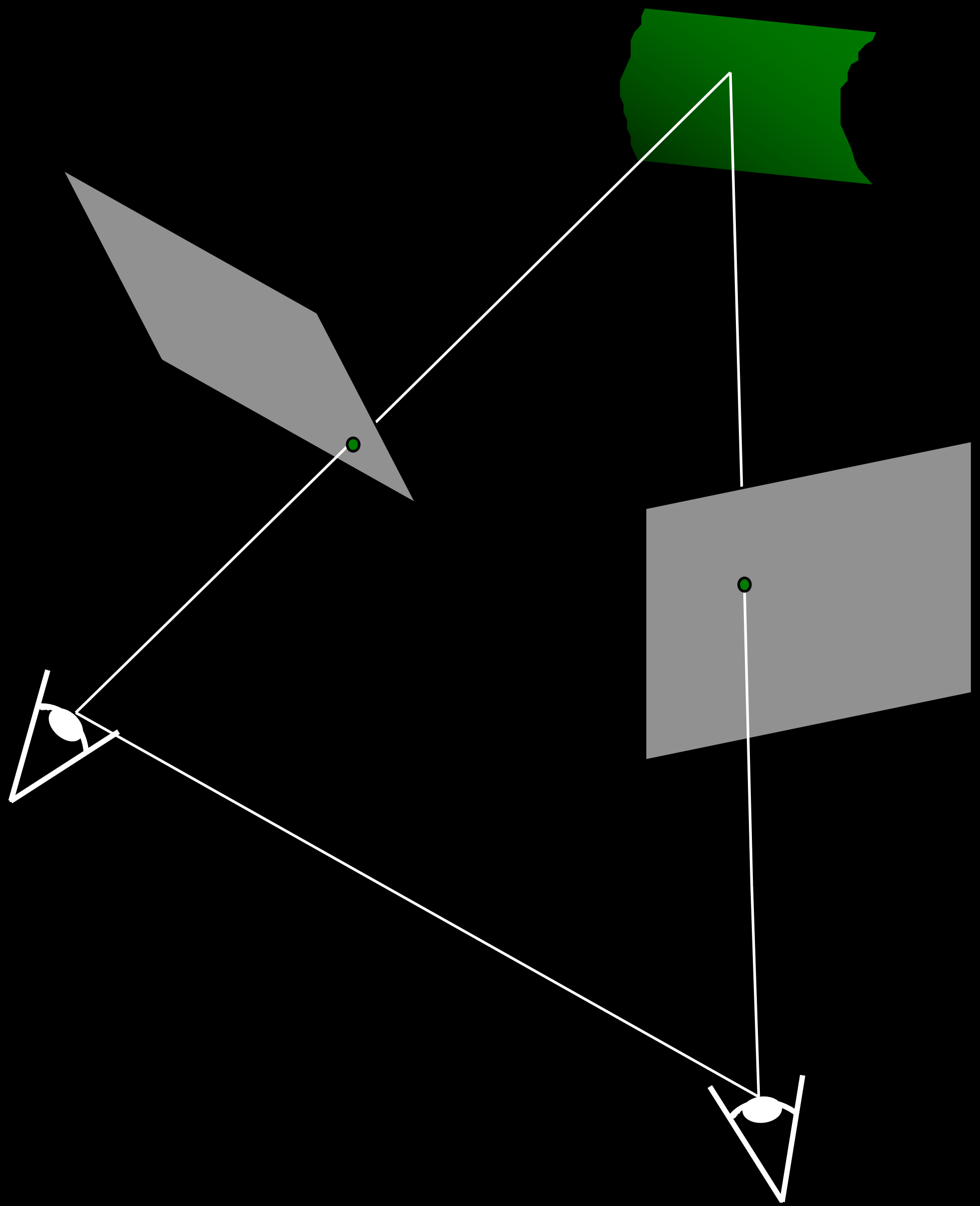
$$\mathrm{E} = [\mathrm{t}_\times]\mathrm{R} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix}$$

$$\begin{pmatrix} u' & v' & 1 \end{pmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -T \\ 0 & T & 0 \end{bmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = 0 \qquad \begin{pmatrix} u' & v' & 1 \end{pmatrix} \begin{pmatrix} 0 \\ -T \\ Tv \end{pmatrix} = 0 \qquad Tv' = Tv$$

The y-coordinates of corresponding points are the same!

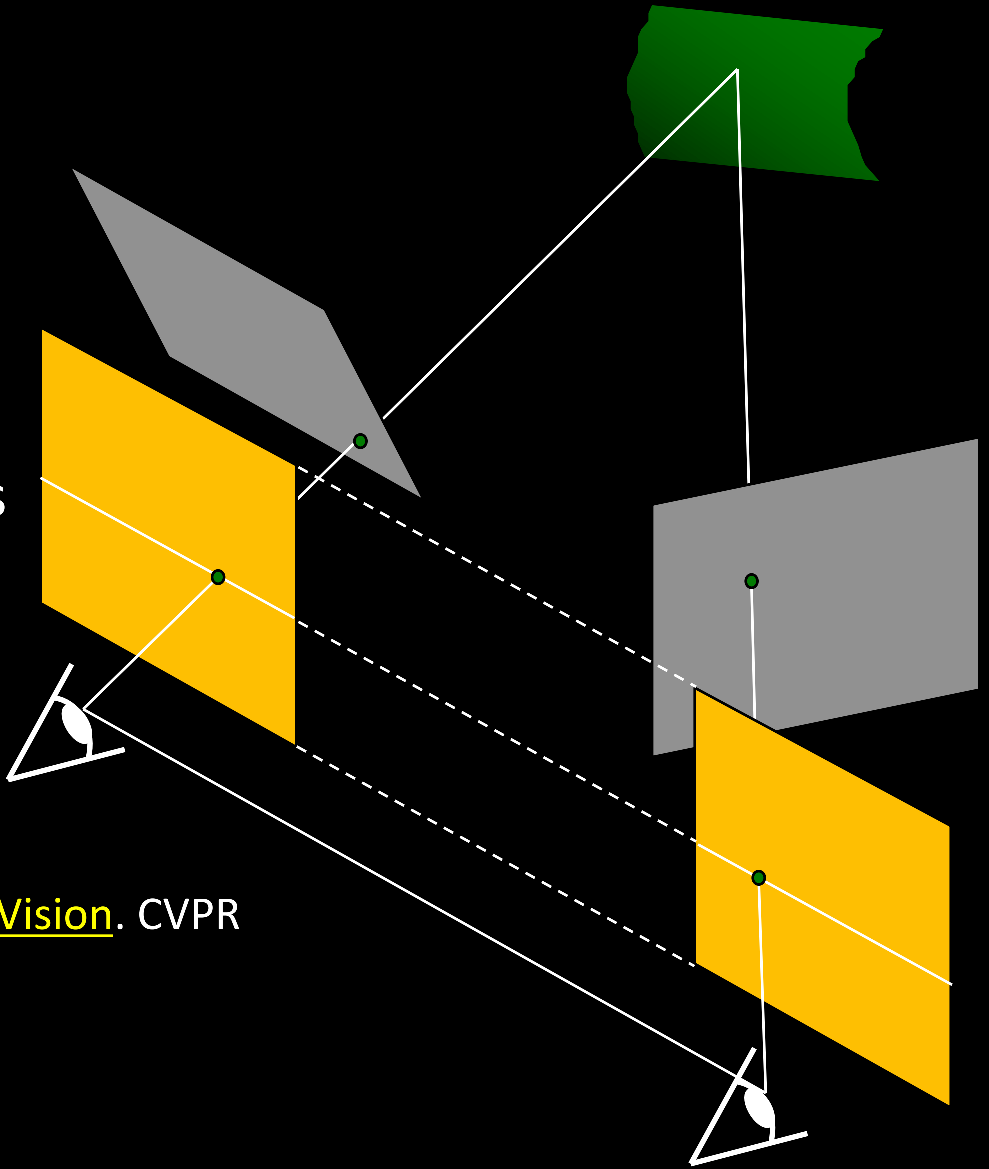
Stereo image rectification

Via Derek Hoeim



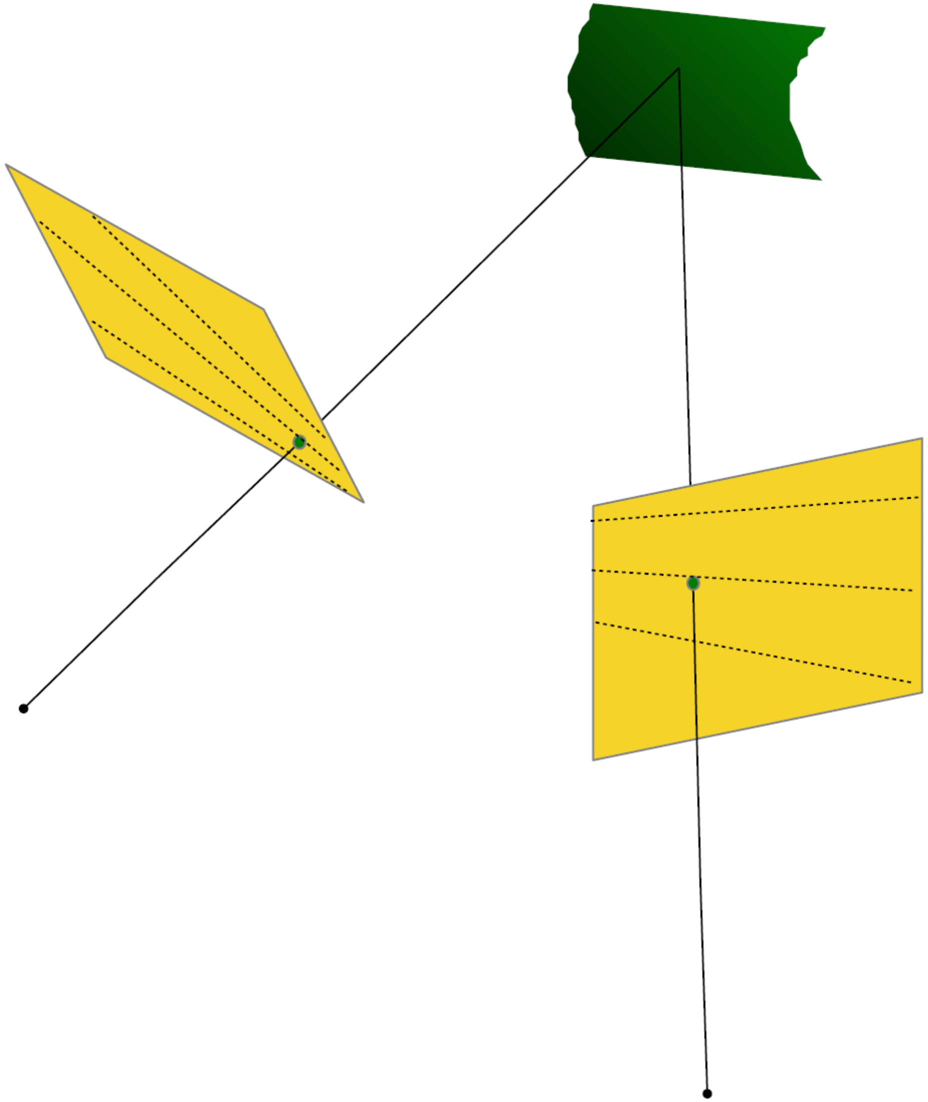
Stereo image rectification

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



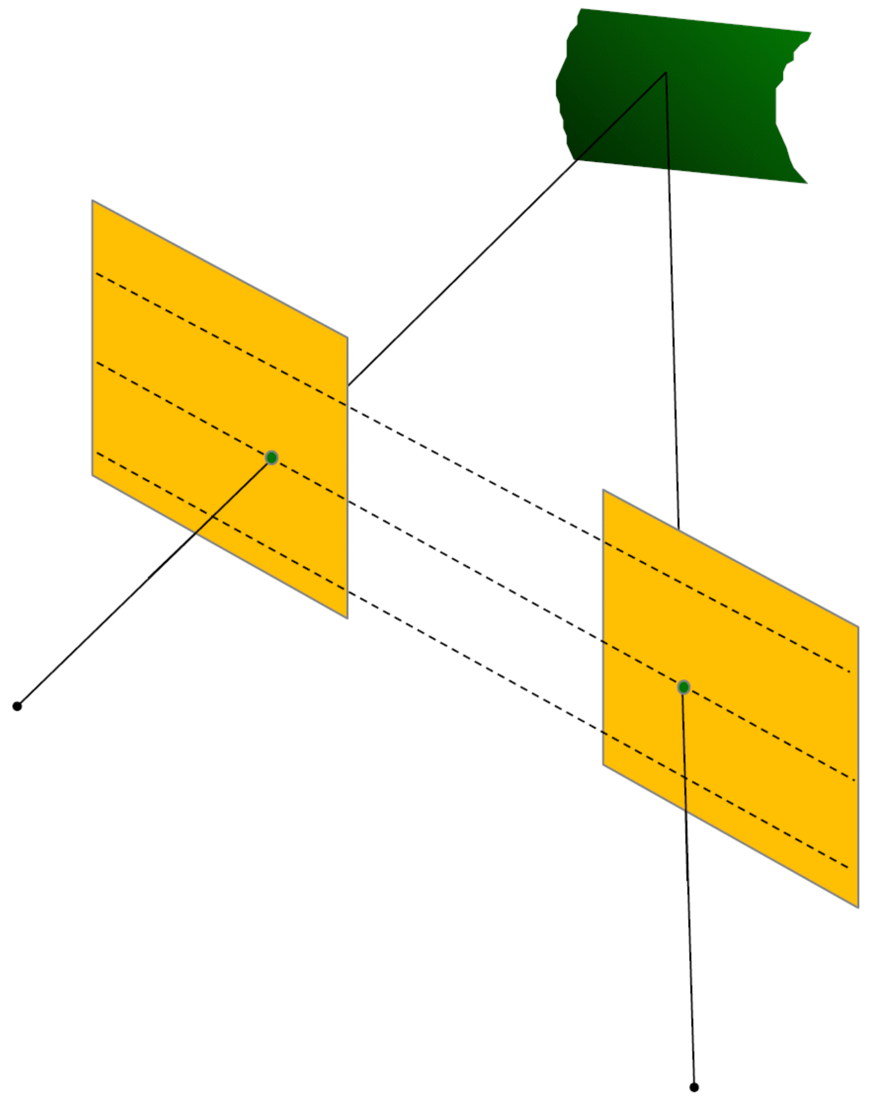
C. Loop and Z. Zhang. [Computing Rectifying Homographies for Stereo Vision](). CVPR 1999

Stereo Rectification:



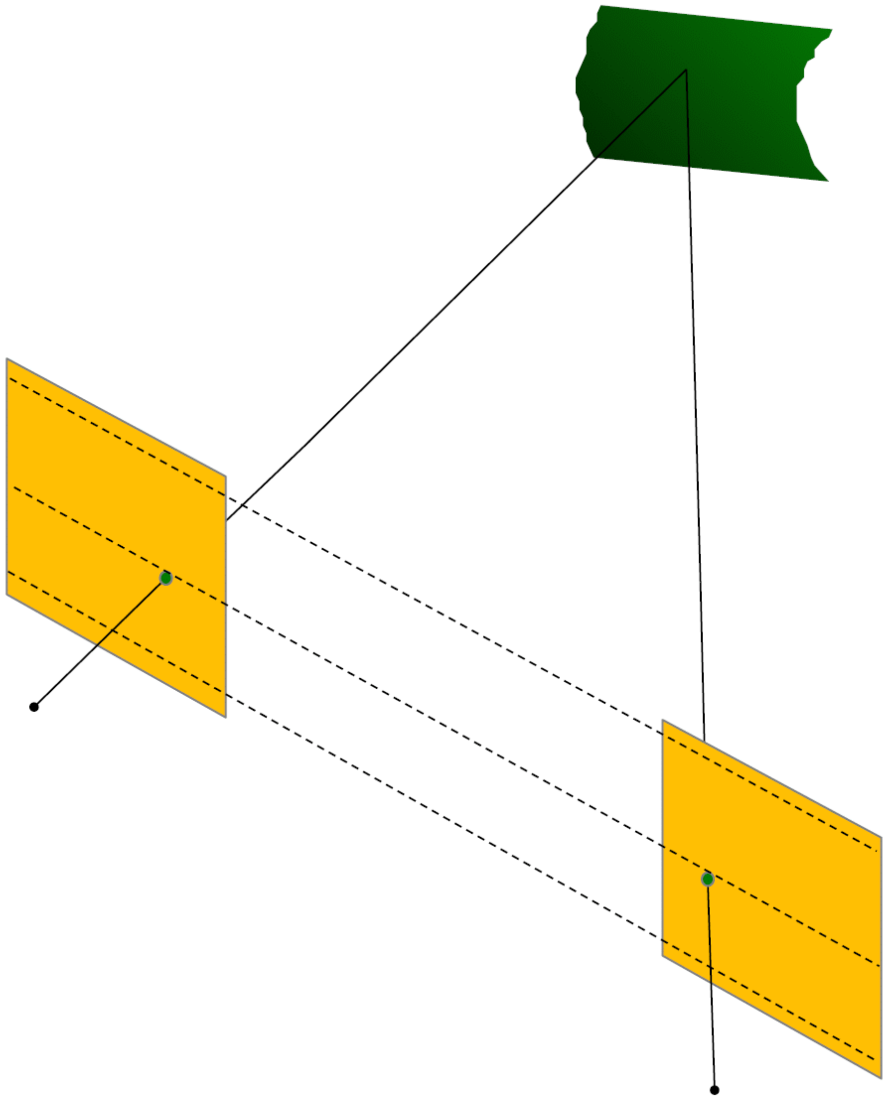
1. Compute **E** to get **R**
2. Rotate right image by **R**
3. Rotate both images by $\mathbf{R}_{\mathbf{rect}}$
4. Scale both images by **H**

Stereo Rectification:



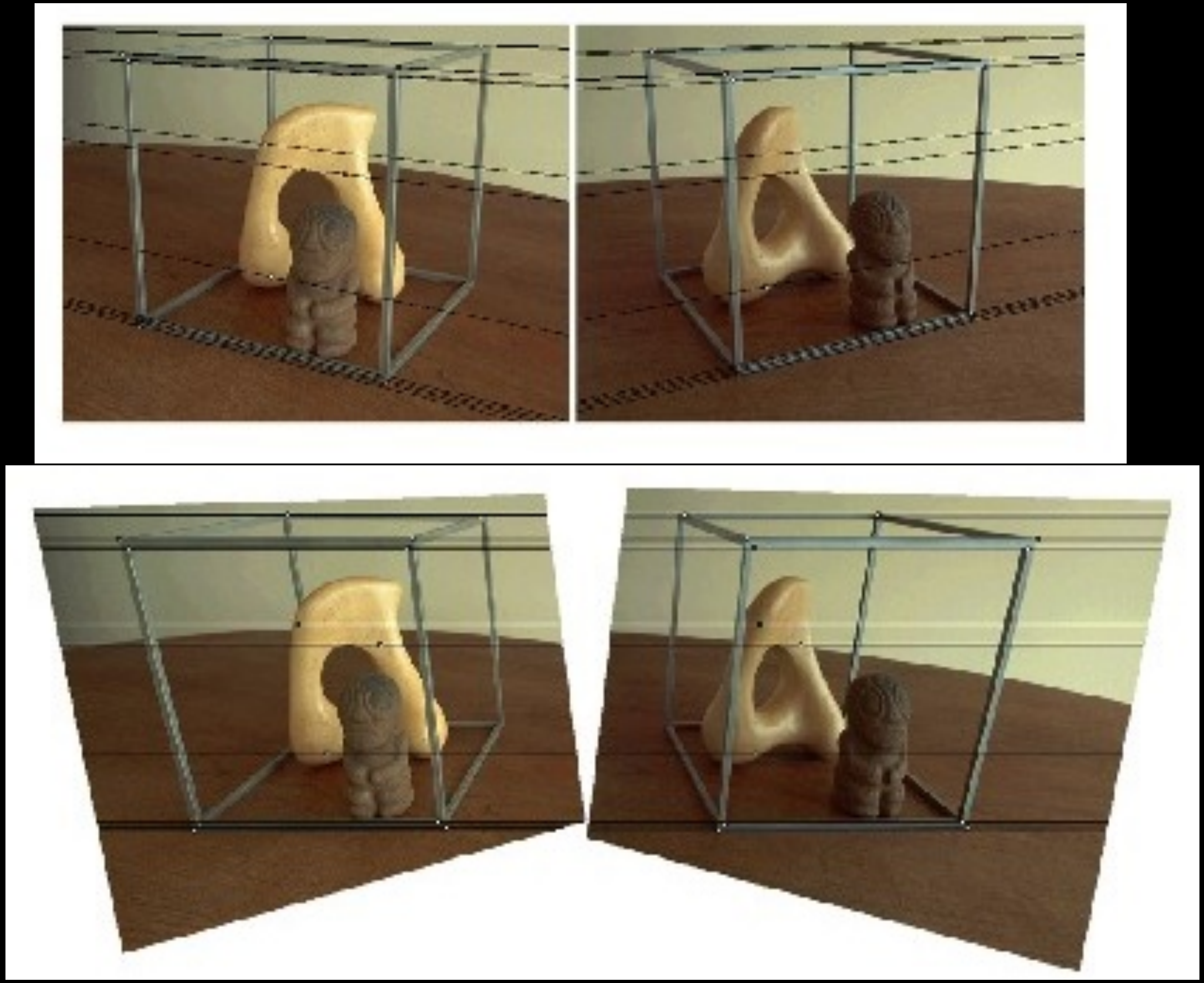
1. Compute **E** to get **R**
2. Rotate right image by **R**
3. Rotate both images by $\mathbf{R_{rect}}$
4. Scale both images by **H**

Stereo Rectification:



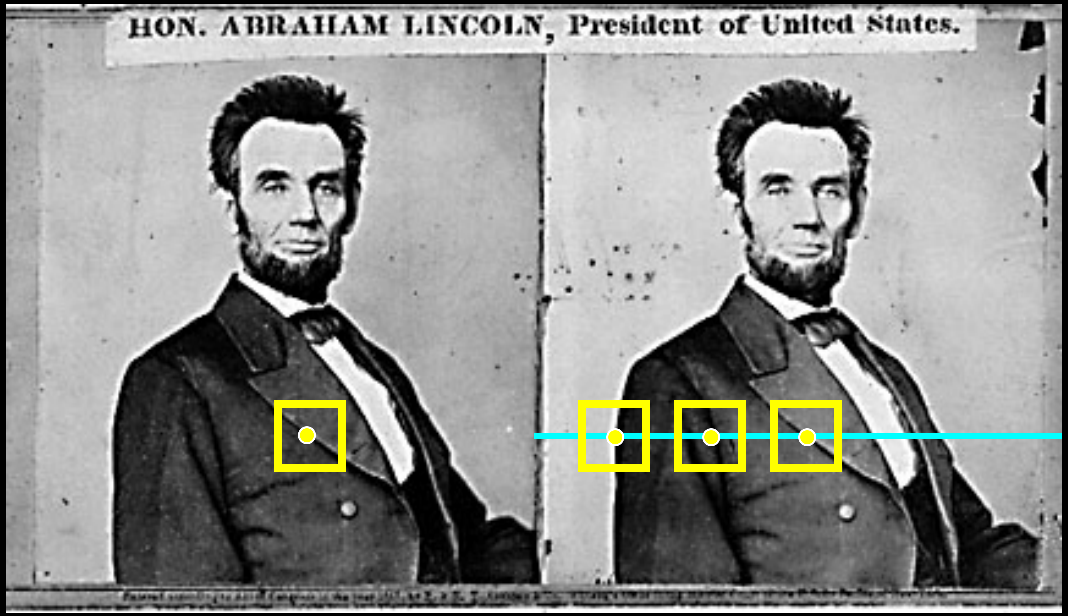
1. Compute **E** to get **R**
2. Rotate right image by **R**
3. Rotate both images by $\mathbf{R_{rect}}$
4. Scale both images by **H**

Rectification example



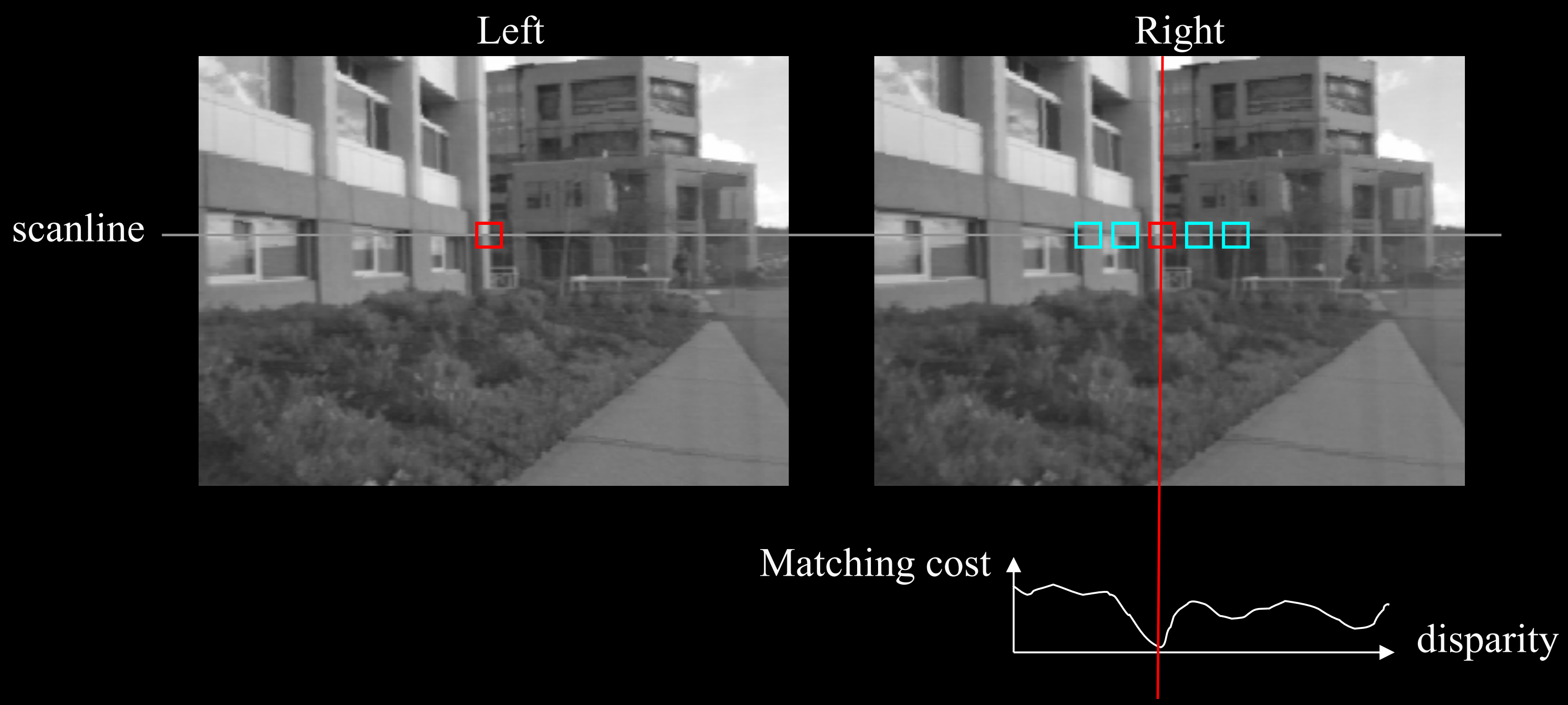
Another rectification example

Basic stereo matching algorithm



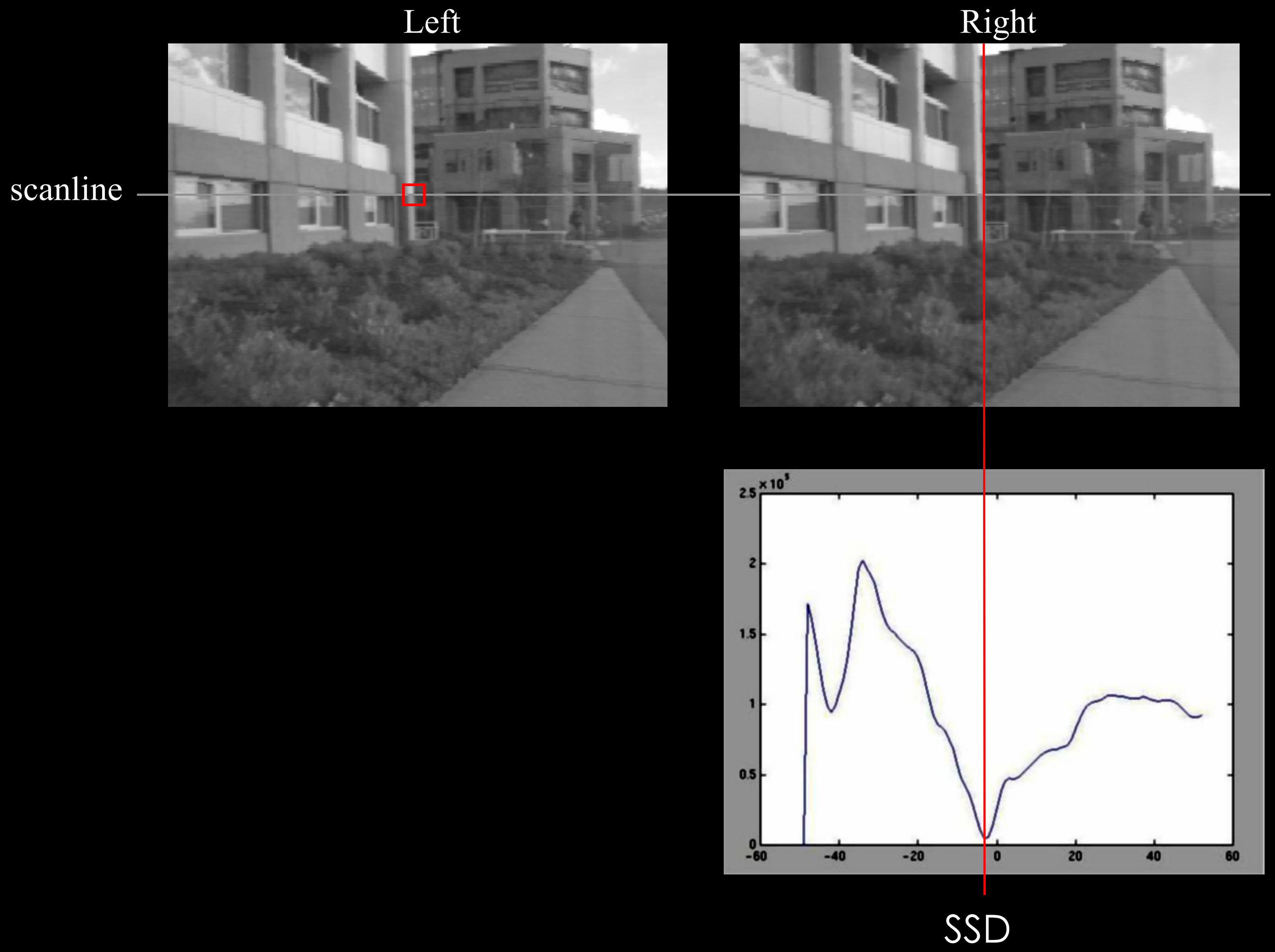
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel in the first image
  - Find corresponding epipolar line in the right image
  - Examine all pixels on the epipolar line and pick the best match

Correspondence search

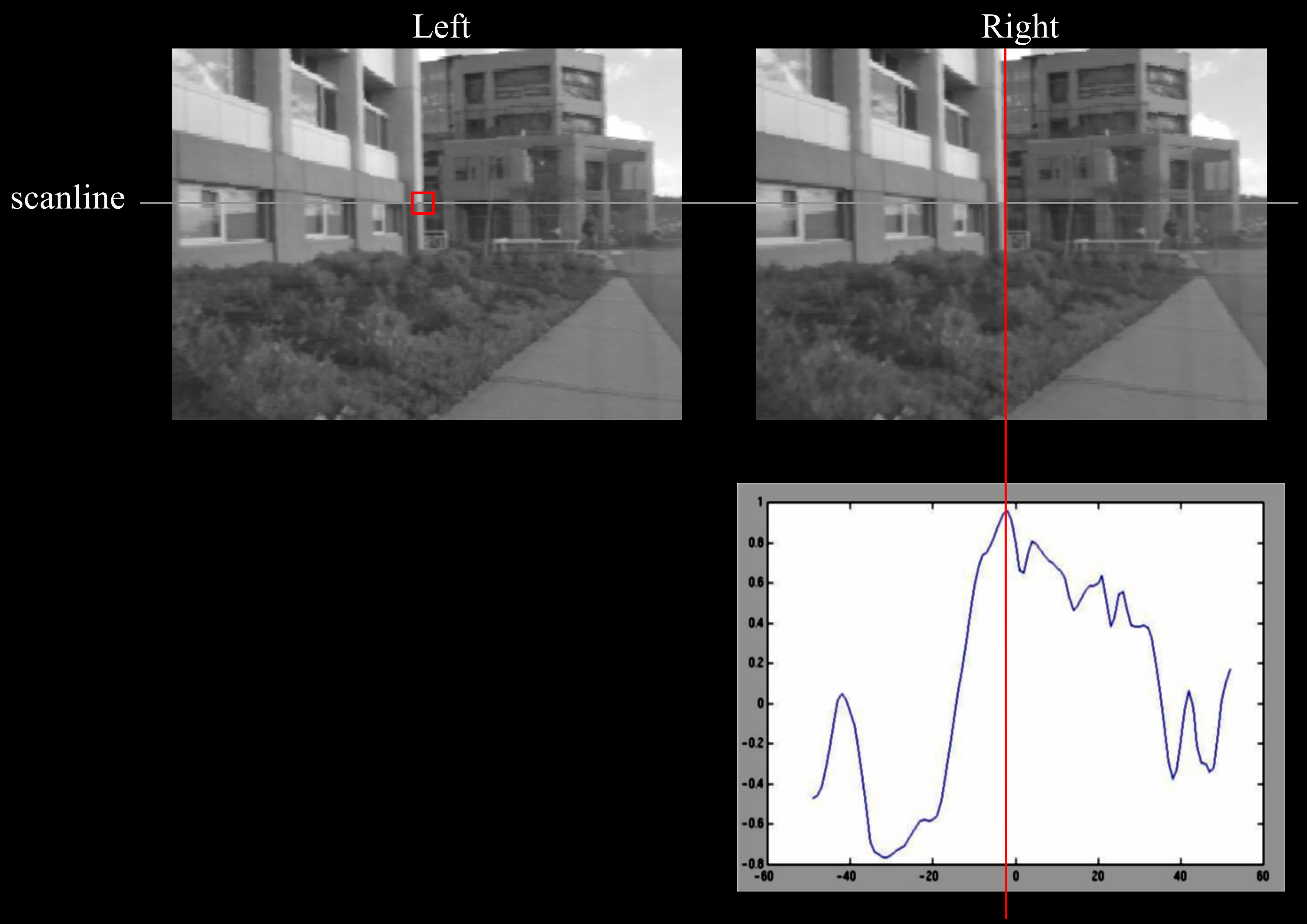


- Slide a window along the right scanline and compare contents of that window with the reference window in the left image
- Matching cost: SSD or normalized correlation

Correspondence search

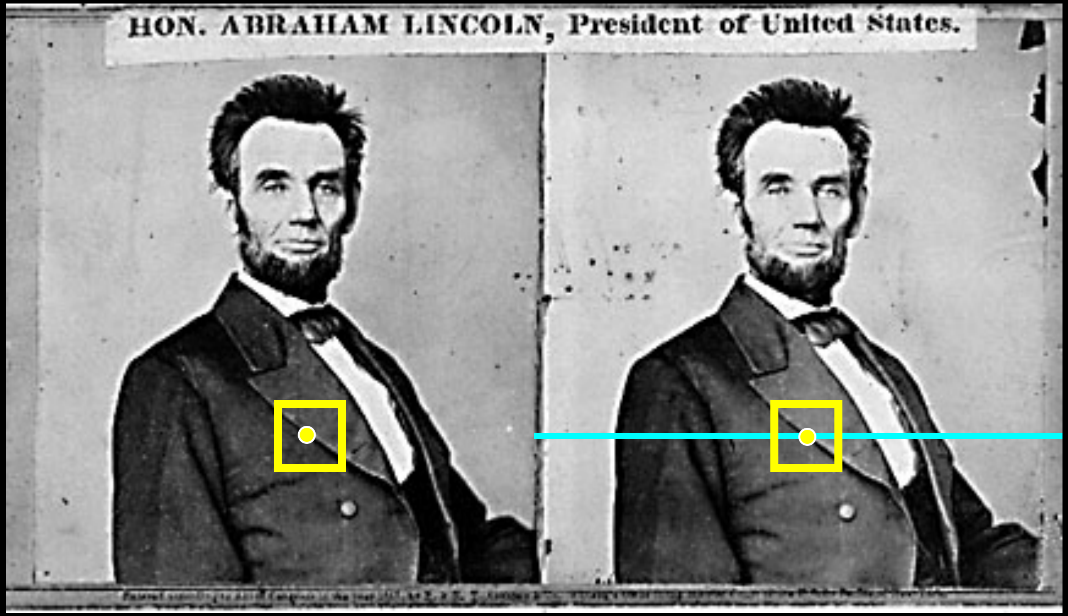


Correspondence search



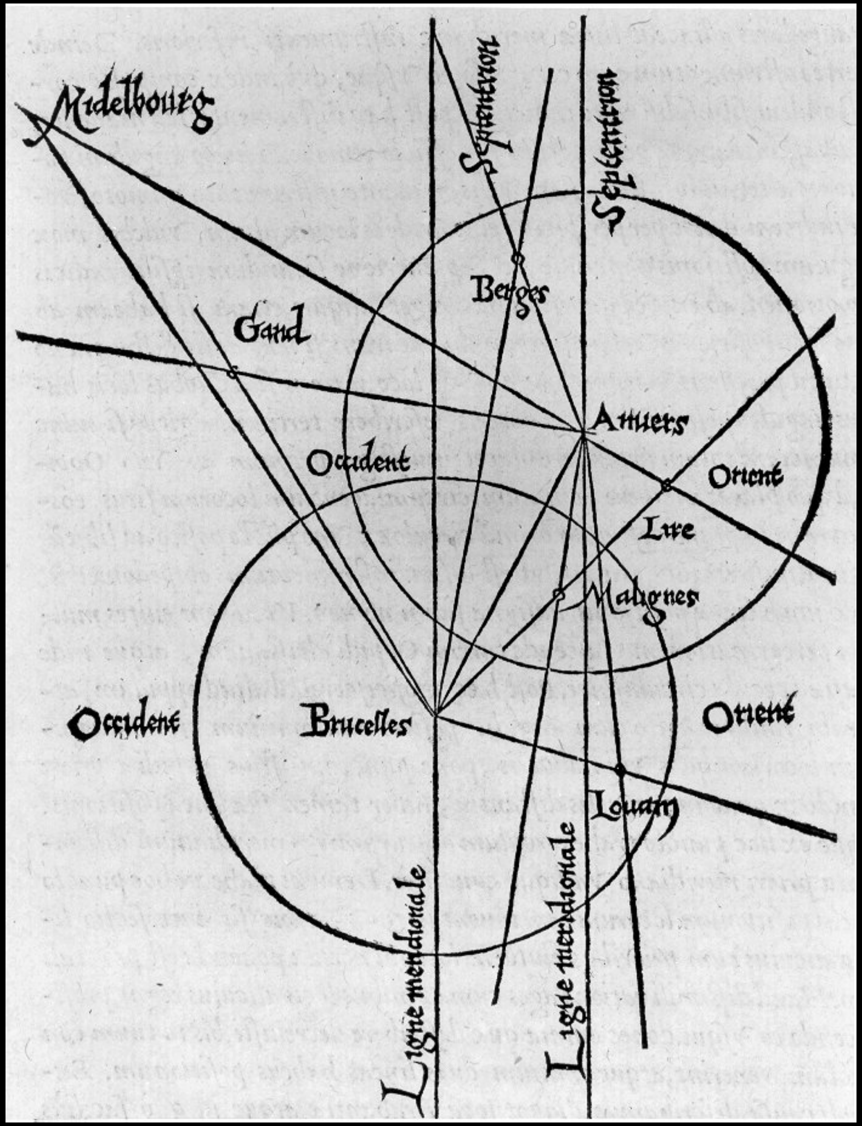
Norm. corr

Basic stereo matching algorithm



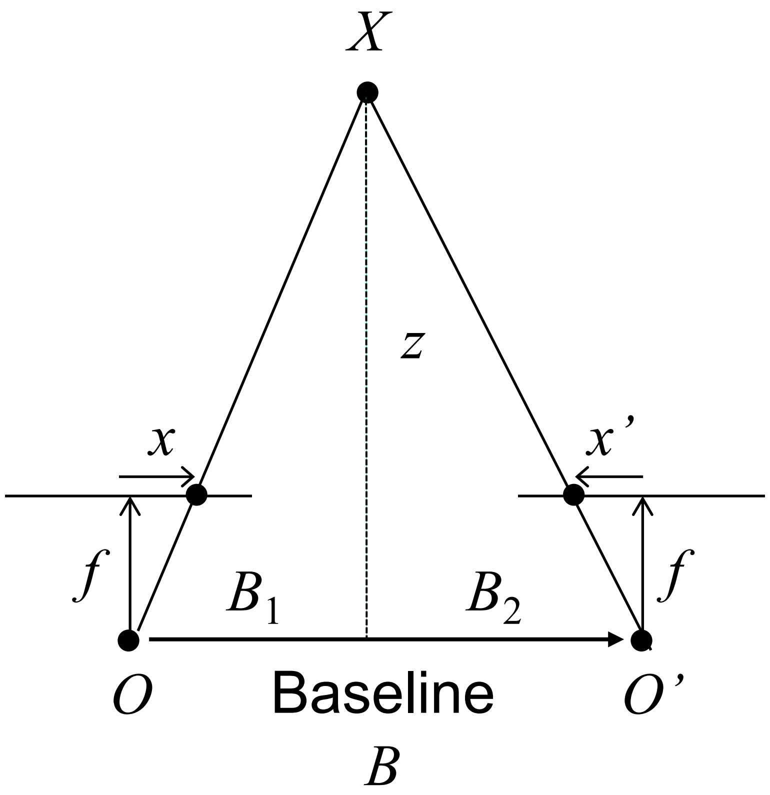
- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match x'
  - Triangulate the matches to get depth information

Triangulation: History



- From Wikipedia: *Gemma Frisius's 1533 diagram introducing the idea of triangulation into the science of surveying. Having established a baseline, e.g. the cities of Brussels and Antwerp, the location of other cities, e.g. Middelburg, Ghent etc., can be found by taking a compass direction from each end of the baseline, and plotting where the two directions cross. This was only a theoretical presentation of the concept — due to topographical restrictions, it is impossible to see Middelburg from either Brussels or Antwerp. Nevertheless, the figure soon became well known all across Europe.*

Depth from disparity



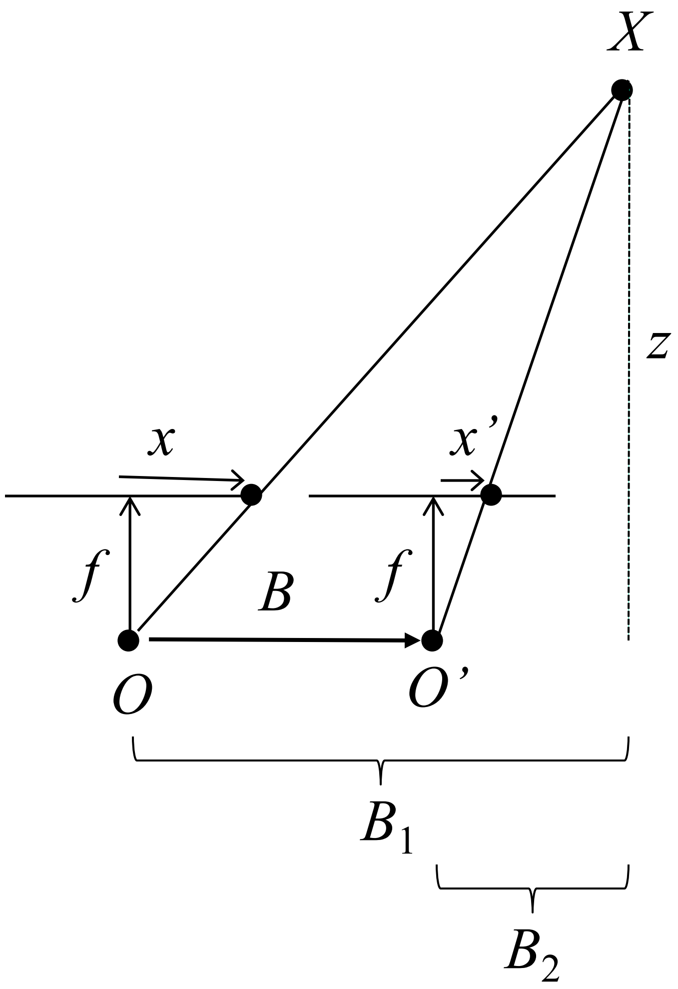
$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{-x'}{f} = \frac{B_2}{z}$$

$$\frac{x - x'}{f} = \frac{B_1 + B_2}{z}$$

$$\boxed{disparity = x - x' = \frac{B \cdot f}{z}}$$

Disparity is inversely proportional to depth!

Depth from disparity

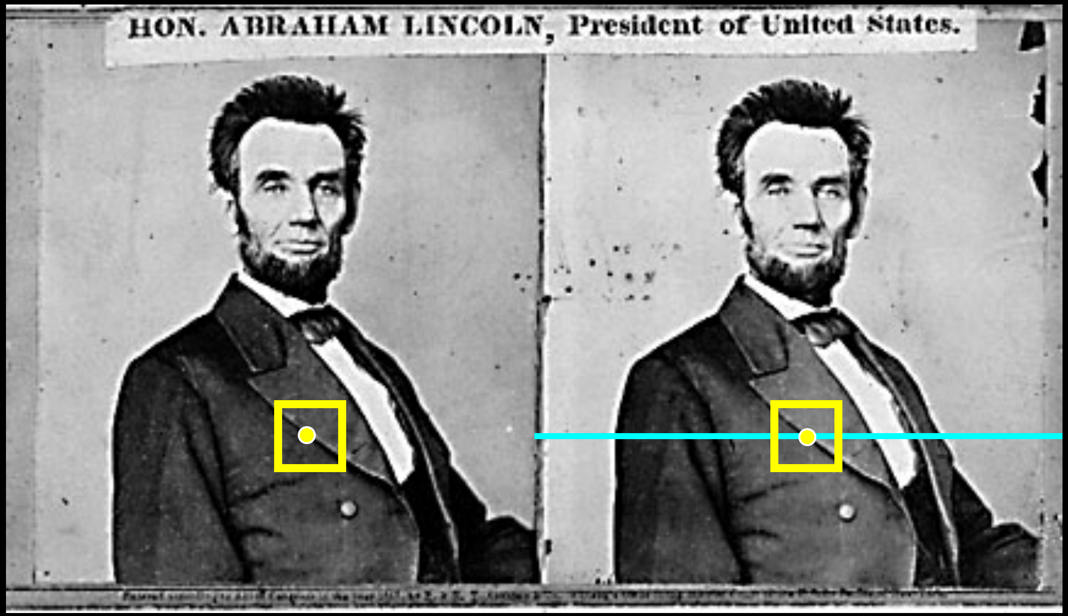


$$\frac{x}{f} = \frac{B_1}{z} \qquad \frac{x'}{f} = \frac{B_2}{z}$$

$$\frac{x - x'}{f} = \frac{B_1 - B_2}{z}$$

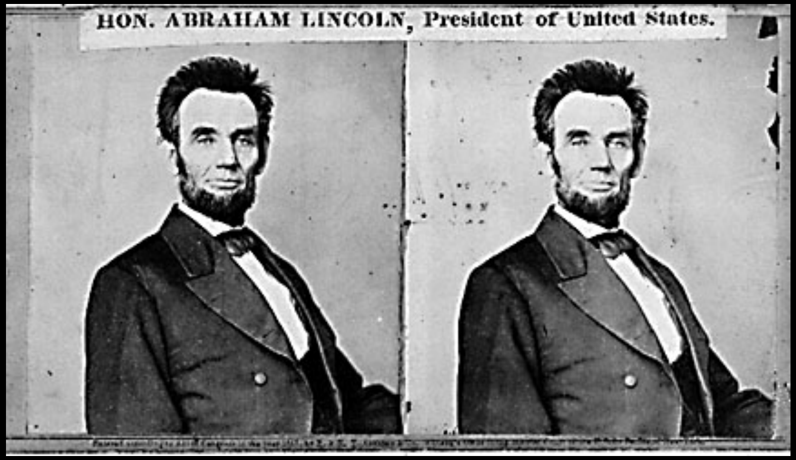
$$\boxed{disparity = x - x' = \frac{B \cdot f}{z}}$$

Basic stereo matching algorithm



- If necessary, rectify the two stereo images to transform epipolar lines into scanlines
- For each pixel x in the first image
  - Find corresponding epipolar scanline in the right image
  - Examine all pixels on the scanline and pick the best match x'
  - Compute disparity $x-x'$ and set $\text{depth}(x) = B*f/(x-x')$

Failures of correspondence search

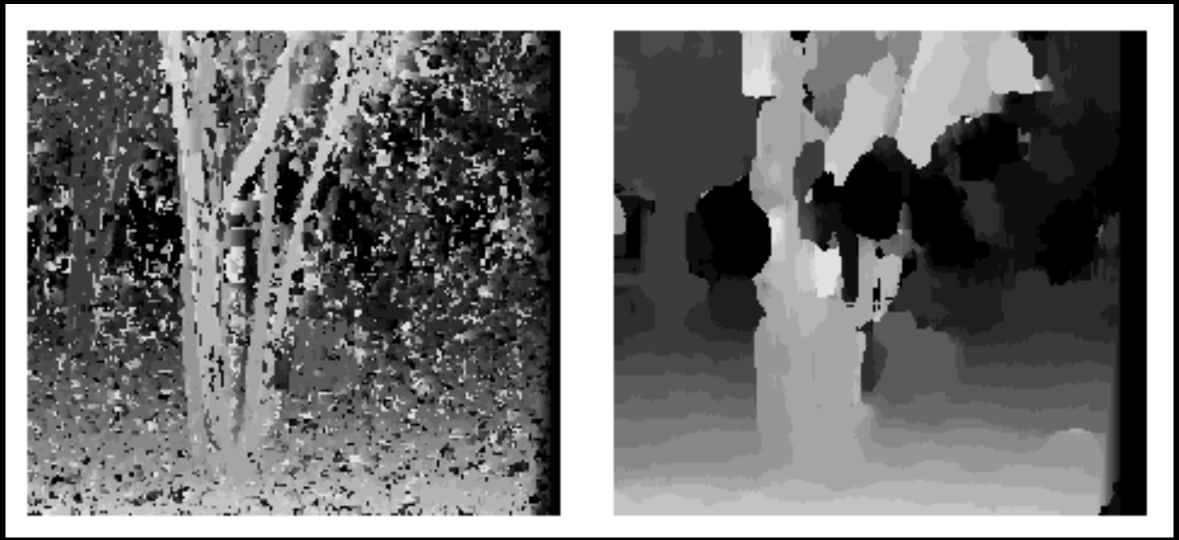


Textureless surfaces

Occlusions, repetition

Non-Lambertian surfaces, specularities

Effect of window size



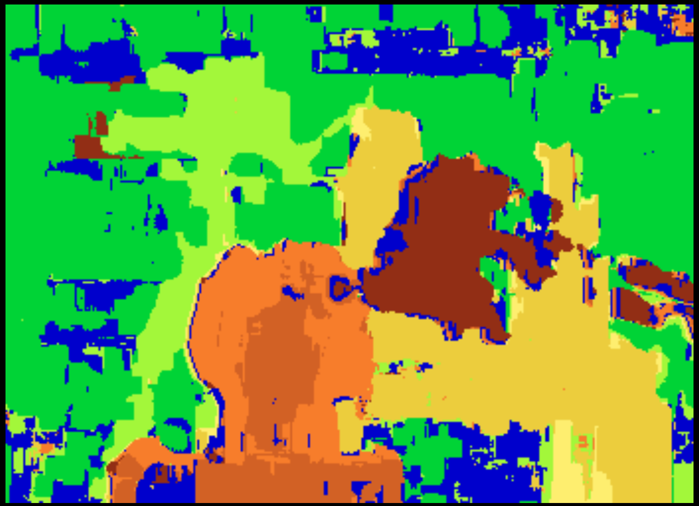
W = 3 W = 20

- Smaller window
  - + More detail
  - ○ More noise
- Larger window
  - + Smoother disparity maps
  - ○ Less detail

Results with window search

Data

Window-based matching



Ground truth

How can we improve window-based matching?

- The similarity constraint is **local** (each reference window is matched independently)
- Need to enforce **non-local** correspondence constraints

Spatial / Temporal Window Selection

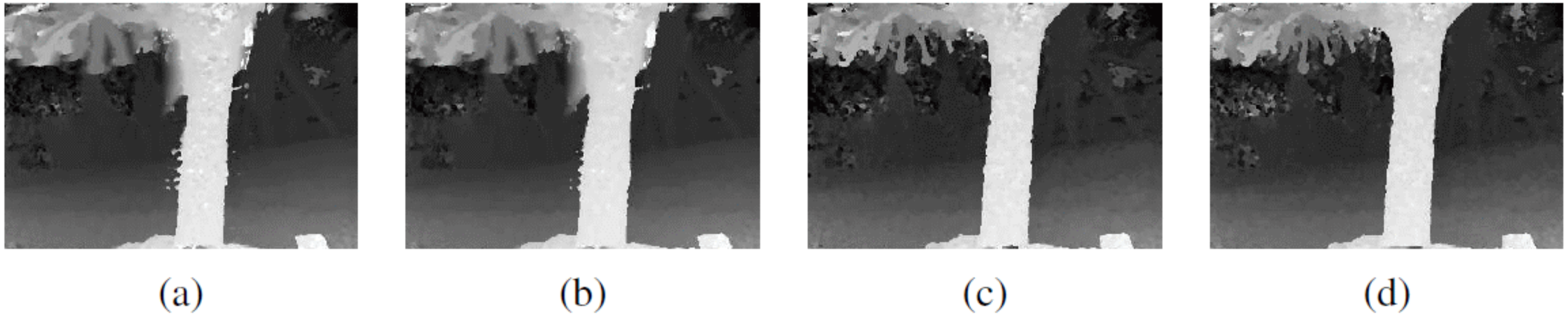
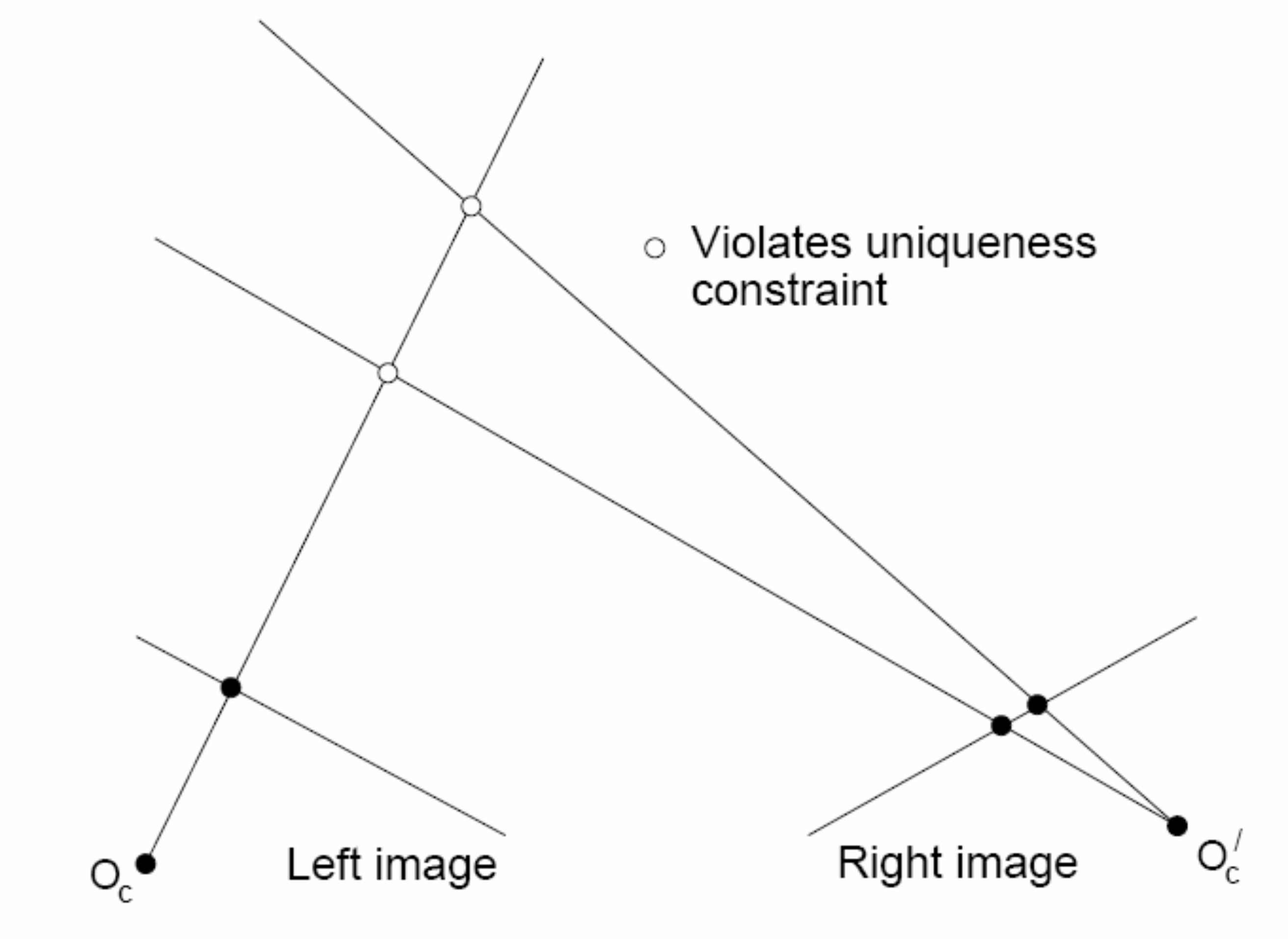


Figure 12.17 *Local (5 × 5 window-based) matching results (Kang, Szeliski, and Chai 2001) © 2001 IEEE: (a) window that is not spatially perturbed (centered); (b) spatially perturbed window; (c) using the best five of 10 neighboring frames; (d) using the better half sequence. Notice how the results near the tree trunk are improved using temporal selection.*

Non-local constraints

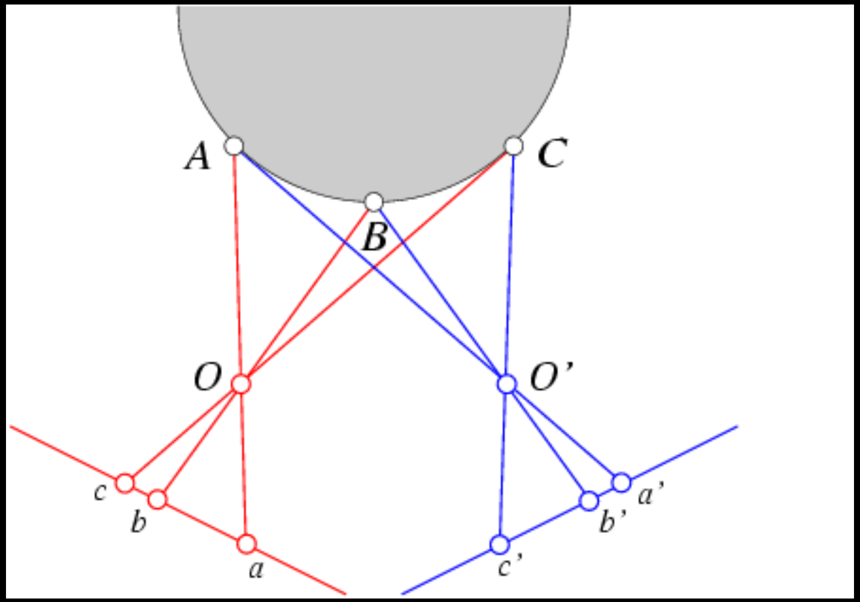
Uniqueness

- For any point in one image, there should be at most one matching point in the other image



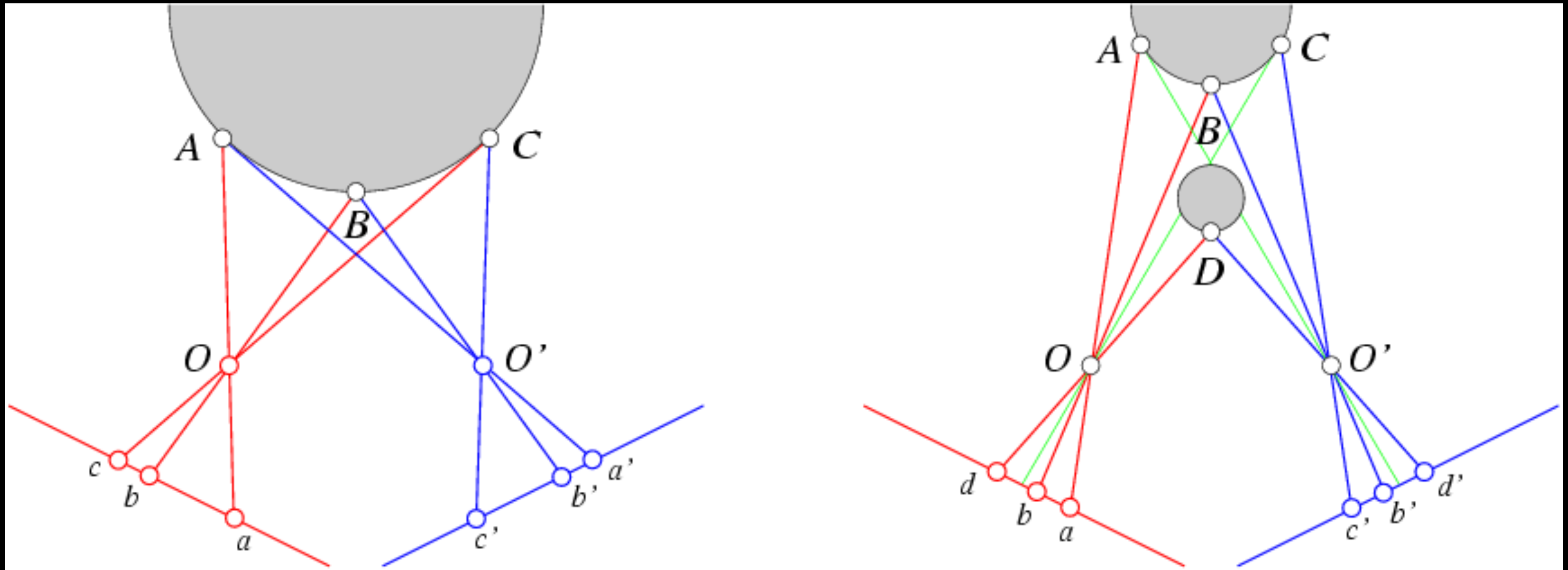
Non-local constraints

- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views



Non-local constraints

- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views



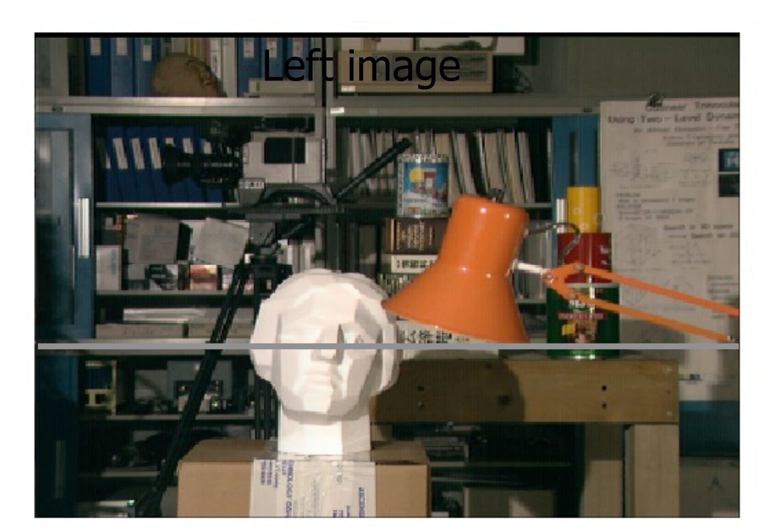
Ordering constraint doesn't hold

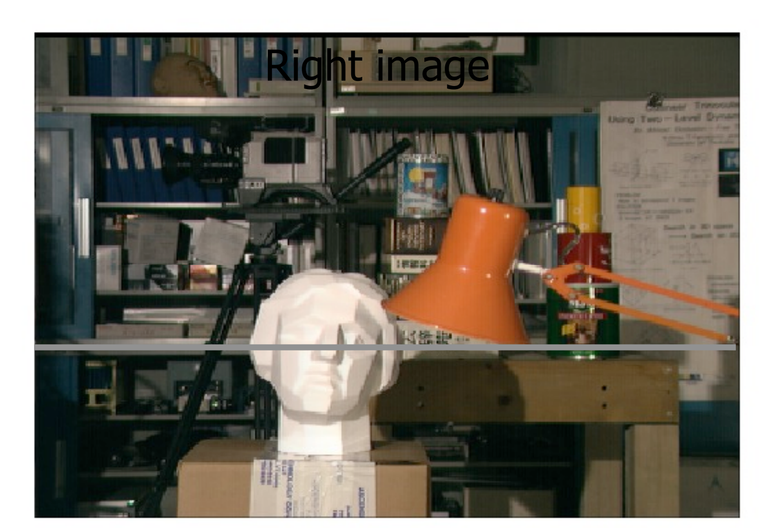
Non-local constraints

- Uniqueness
  - For any point in one image, there should be at most one matching point in the other image
- Ordering
  - Corresponding points should be in the same order in both views
- Smoothness
  - We expect disparity values to change slowly (for the most part)

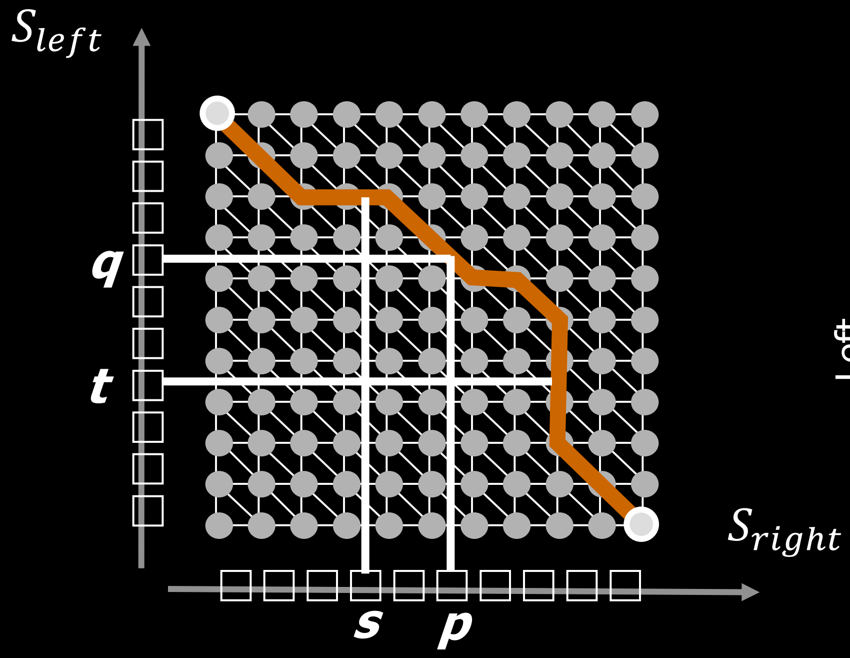
Scanline stereo

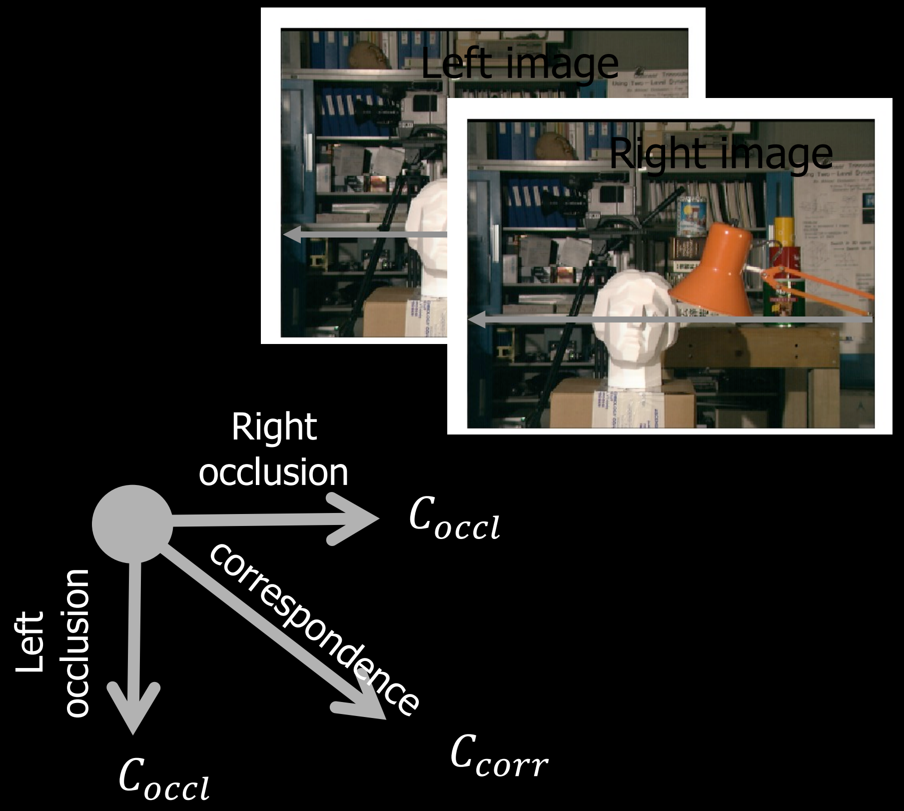
- Try to coherently match pixels on the entire scanline
- Different scanlines are still optimized independently





"Shortest paths" for scan-line stereo





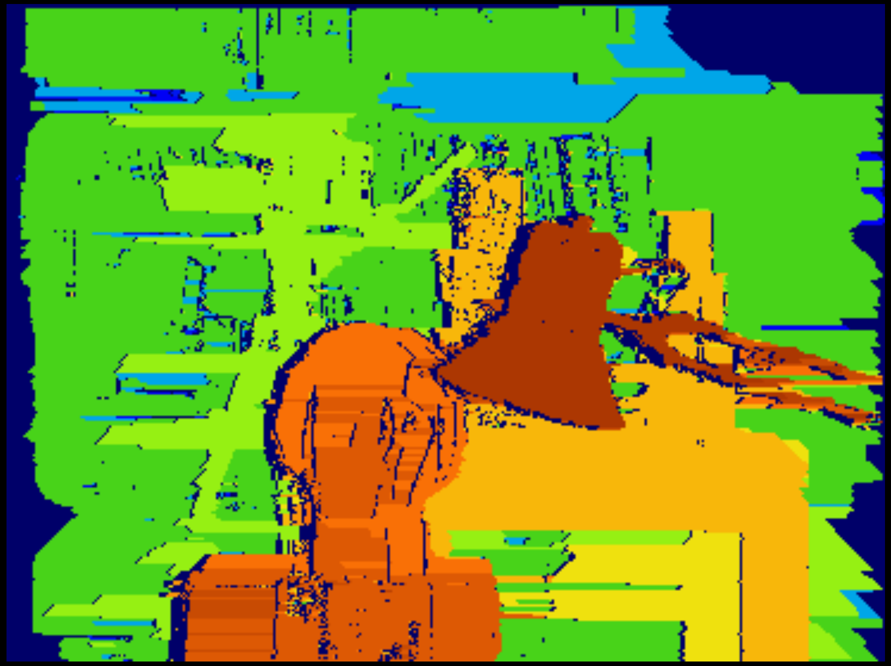
Can be implemented with dynamic programming

Ohta & Kanade '85, Cox et al. '96

Slide credit: Y. Boykov

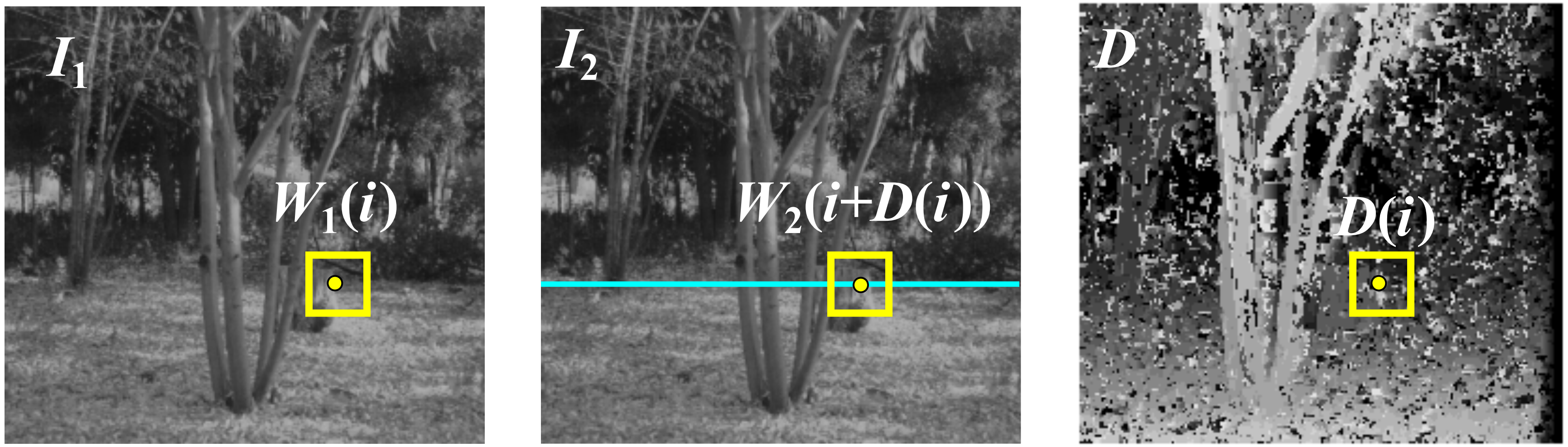
Coherent stereo on 2D grid

- Scanline stereo generates streaking artifacts



- Can't use dynamic programming to find spatially coherent disparities/ correspondences on a 2D grid

Stereo matching as global optimization



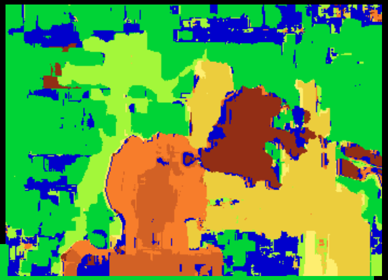
$$E(D) = \underbrace{\sum_i \left(W_1(i) - W_2(i + D(i))\right)^2}_{\textit{data term}} + \lambda \underbrace{\sum_{\text{neighbors } i,j} \rho\left(D(i) - D(j)\right)}_{\textit{smoothness term}}$$

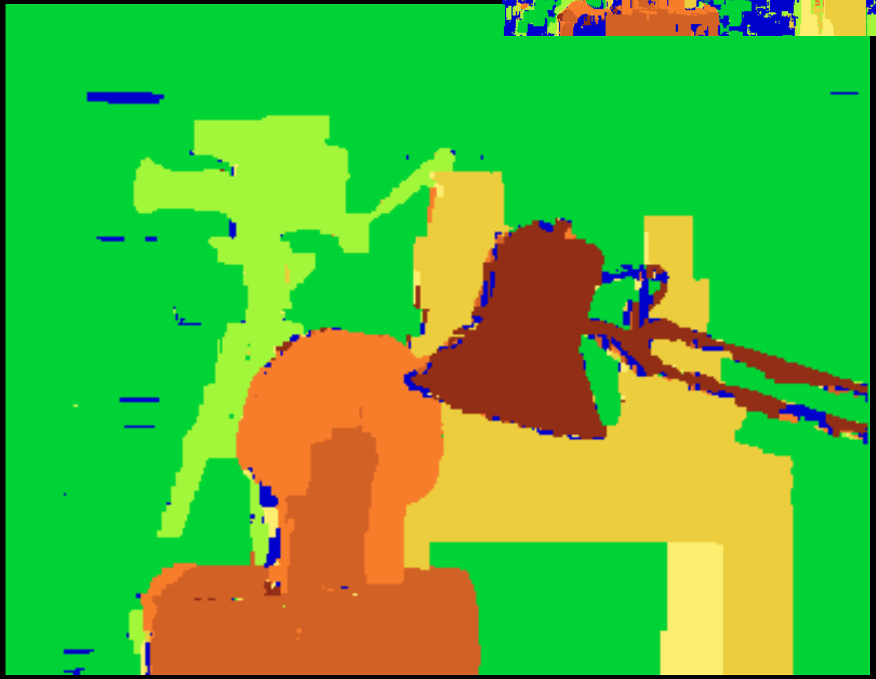
- Energy functions of this form can be minimized using *graph cuts*

Y. Boykov, O. Veksler, and R. Zabih, [Fast Approximate Energy Minimization via Graph Cuts](), PAMI 2001

Many of these constraints can be encoded in an energy function and solved using graph cuts

Before





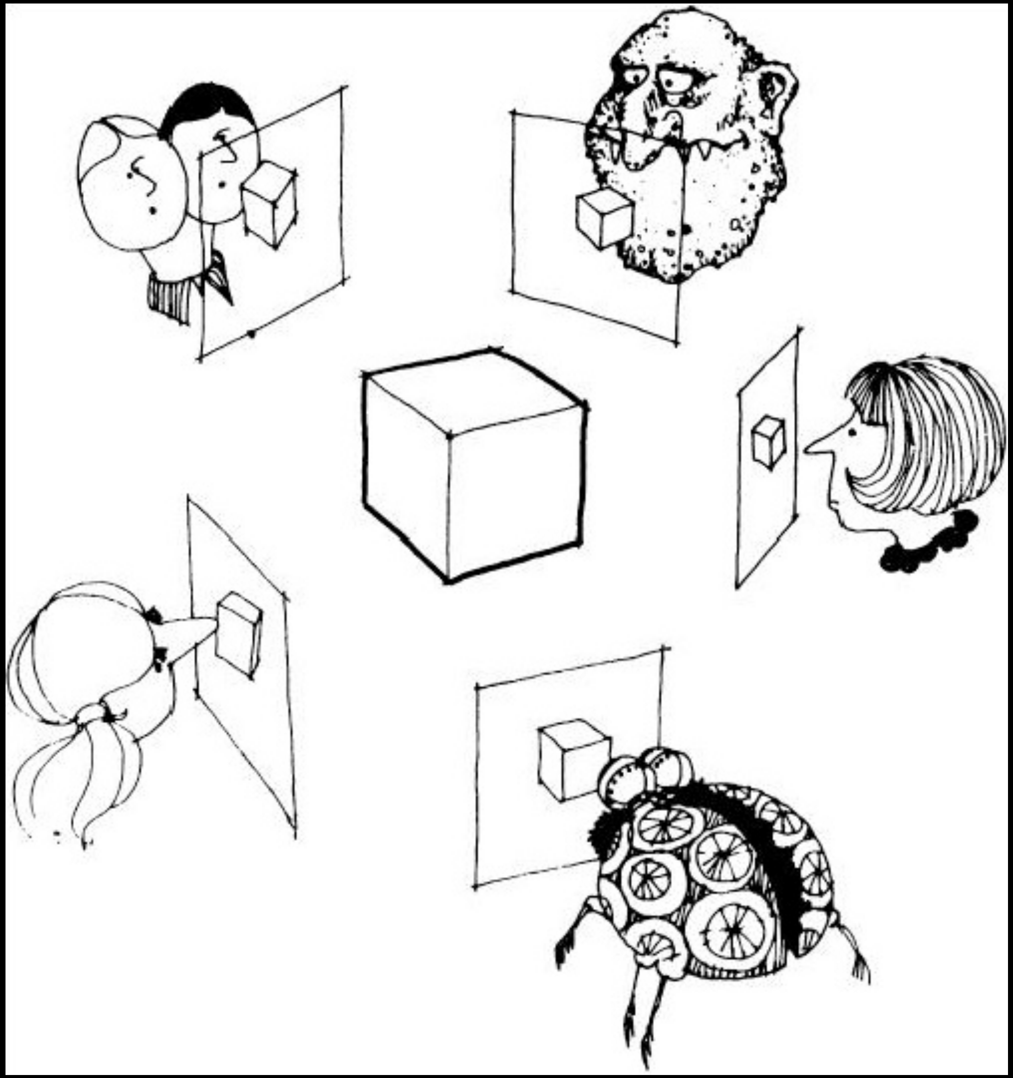
Graph cuts

Ground truth

Y. Boykov, O. Veksler, and R. Zabih, Fast Approximate Energy Minimization via Graph Cuts, PAMI 2001

For good data set comparisons: http://www.middlebury.edu/stereo/

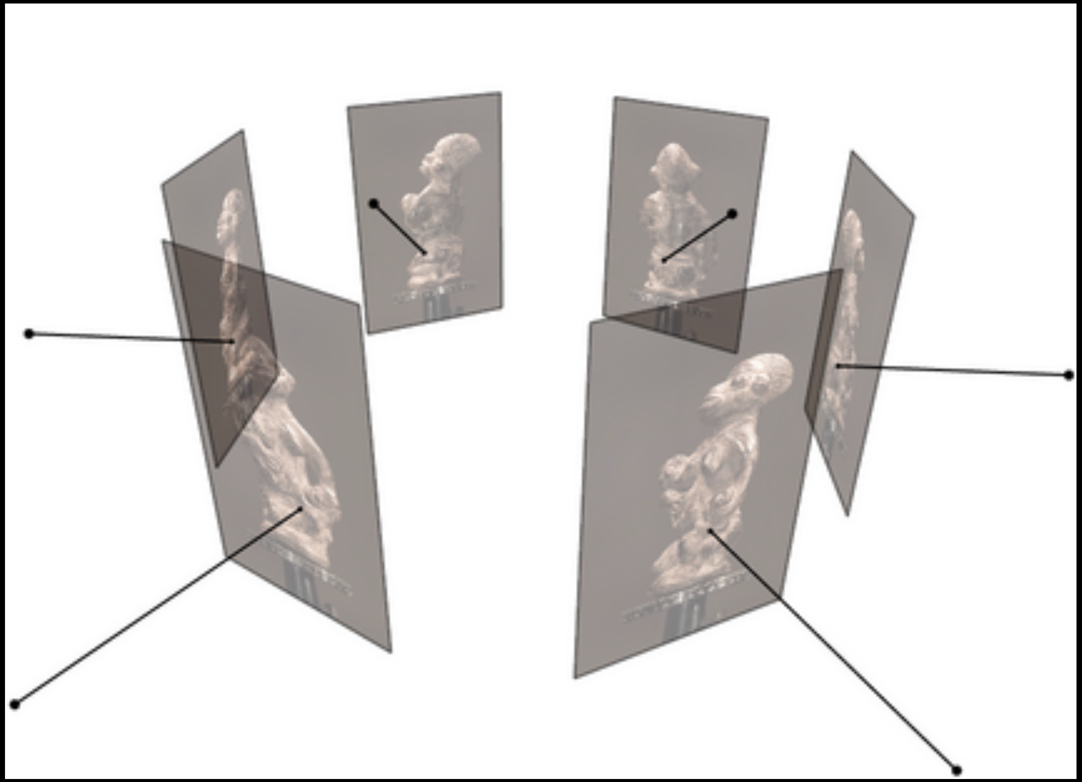
Multi-view stereo



Many slides adapted from S. Seitz, Y. Furukawa, N. Snavely

Multi-view stereo

- Generic problem formulation: given several images of the same object or scene, compute a representation of its 3D shape



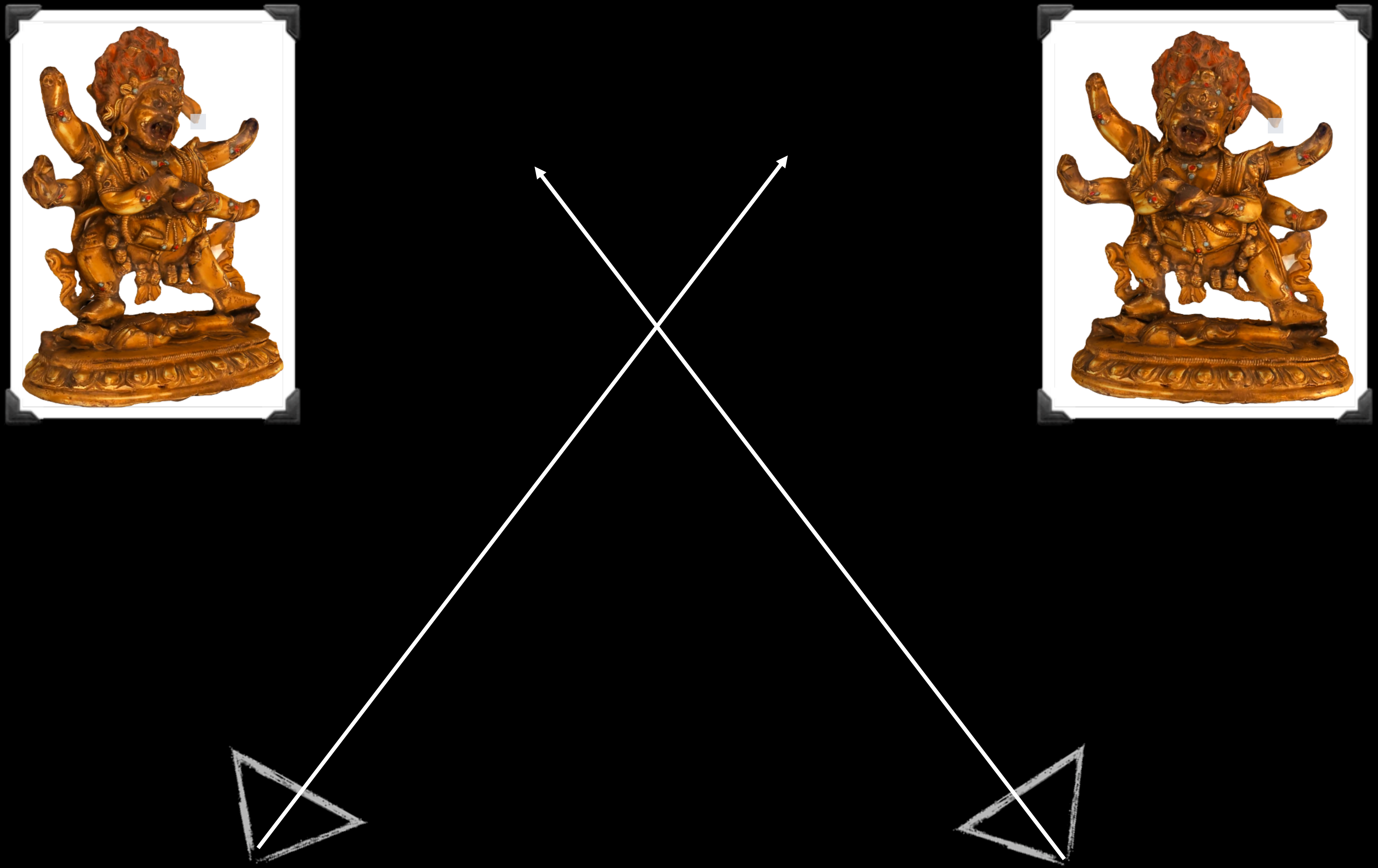
Multi-view stereo

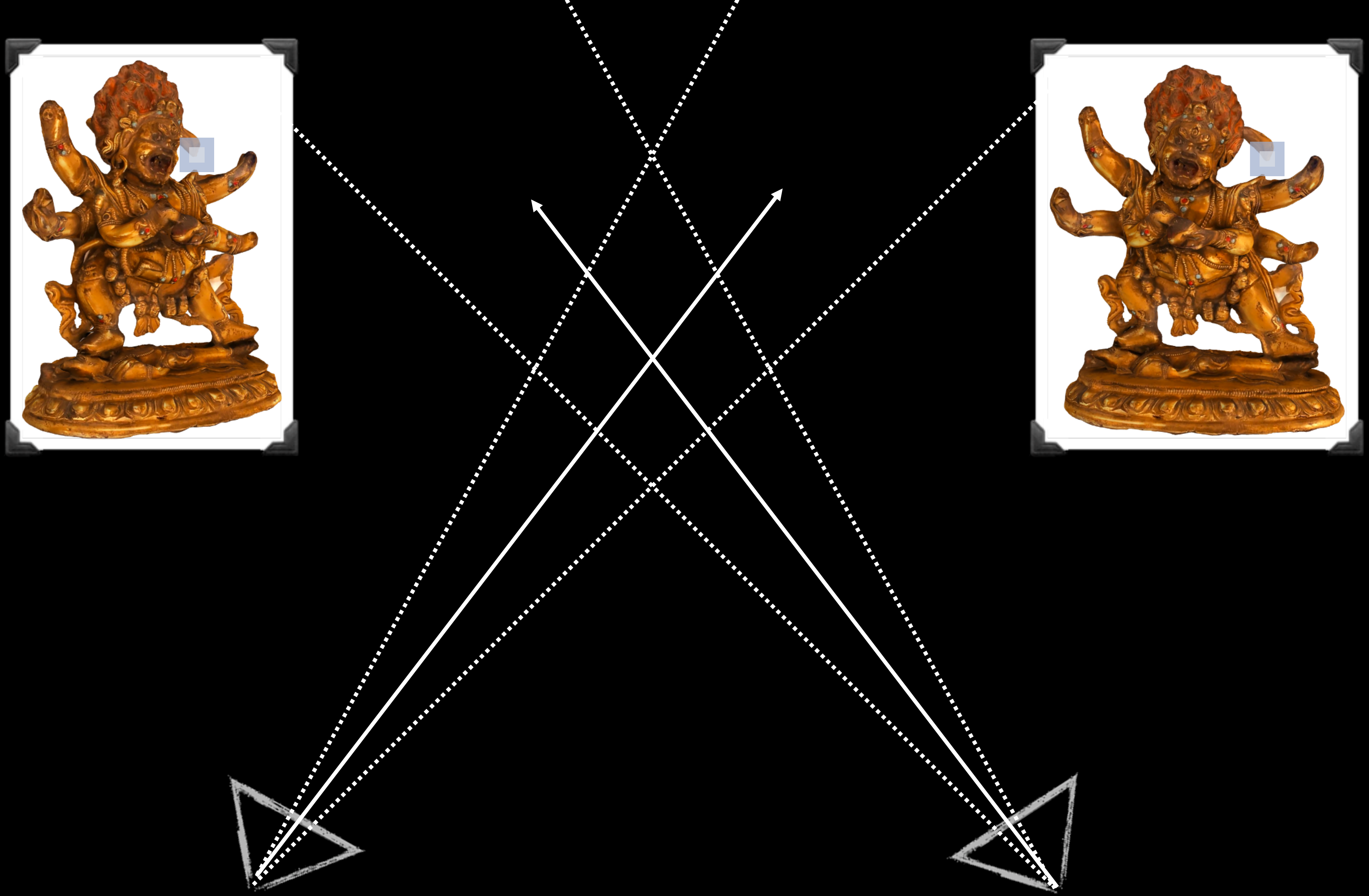
- Generic problem formulation: given several images of the same object or scene, compute a representation of its 3D shape
- “Images of the same object or scene”
  - Arbitrary number of images (from two to thousands)
  - Arbitrary camera positions (special rig, camera network or video sequence)
  - Calibration may be known or unknown
- “Representation of 3D shape”
  - Depth maps
  - Meshes
  - Point clouds
  - Patch clouds
  - Volumetric models
  - ….

Why MVS?

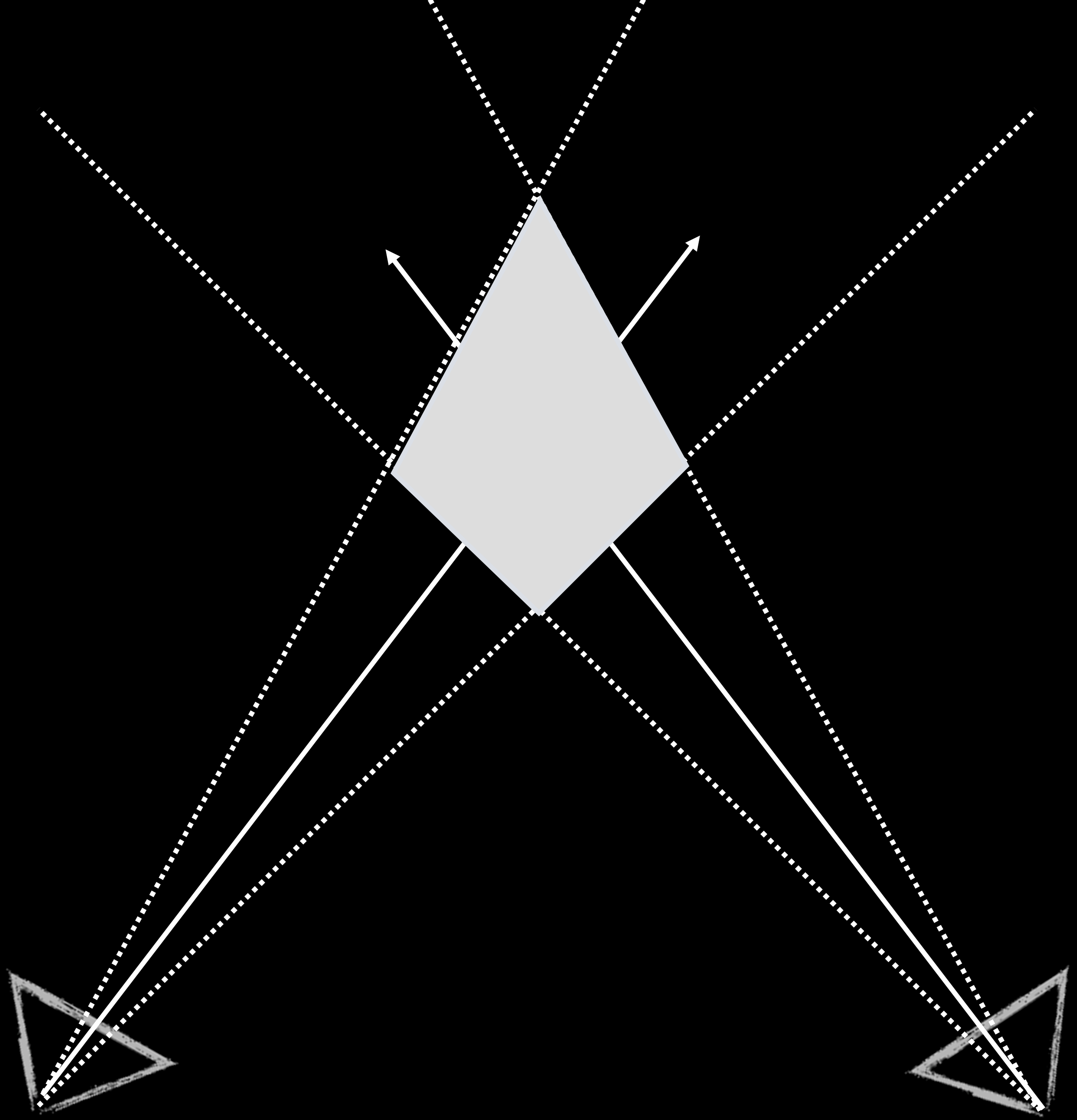
- Different points on the object's surface will be more clearly visible in some subset of cameras
  - Could have high-res closeups of some regions
  - Some surfaces are foreshortened from certain views
  - Some points may be occluded entirely in certain views
- More measurements per point can reduce error

Source: N. Snavely

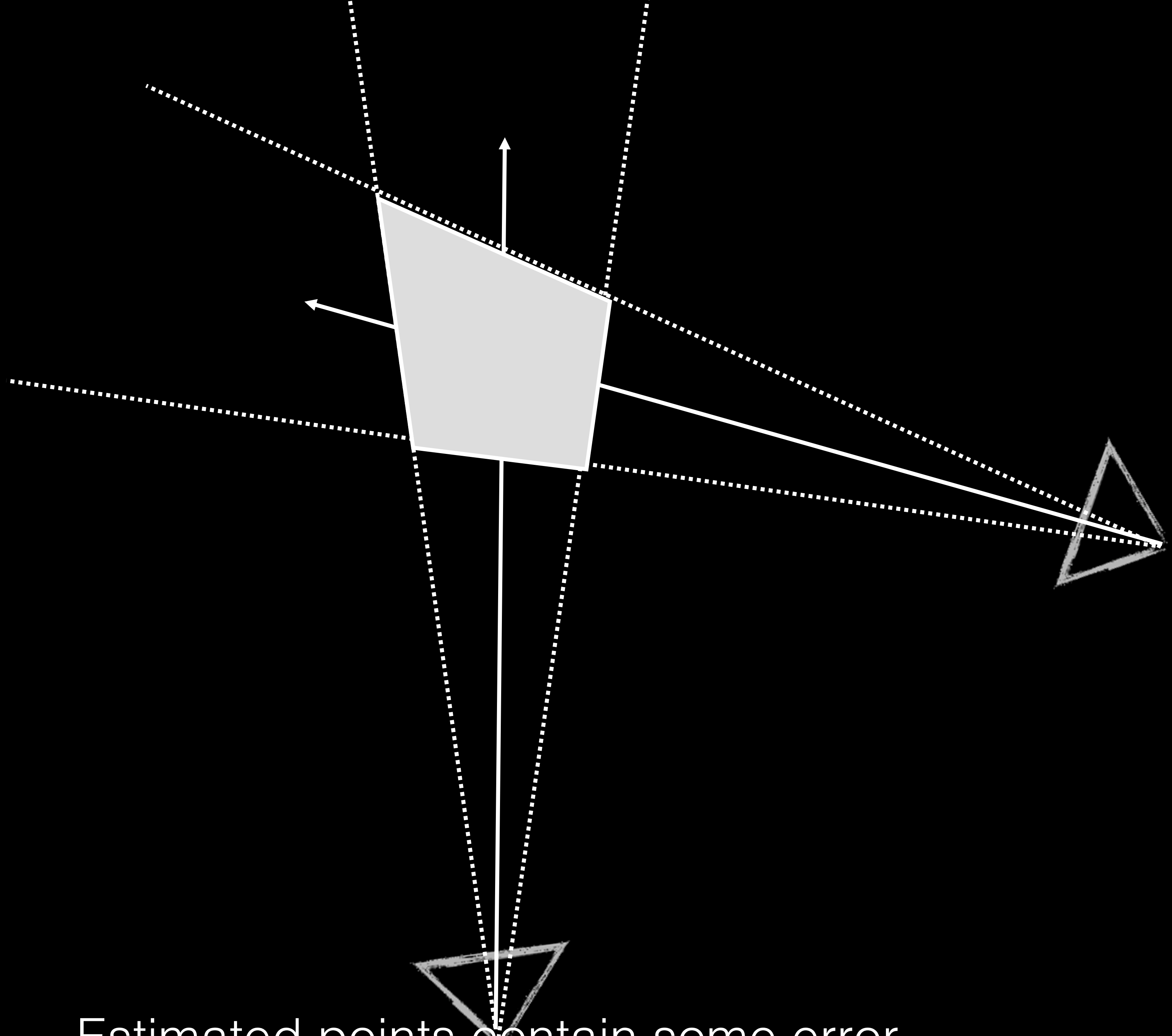




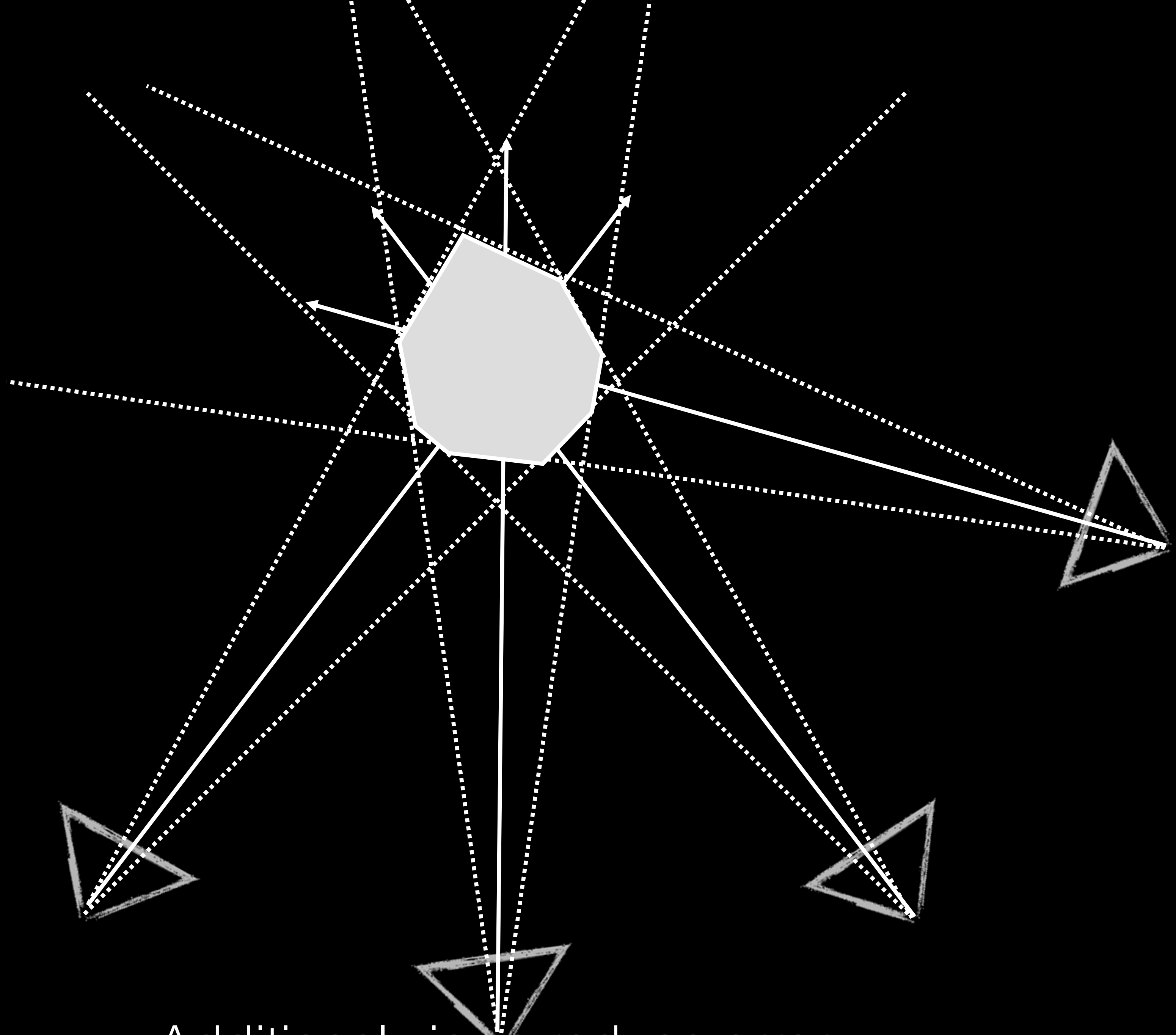
Estimated points contain some error.



Estimated points contain some error.



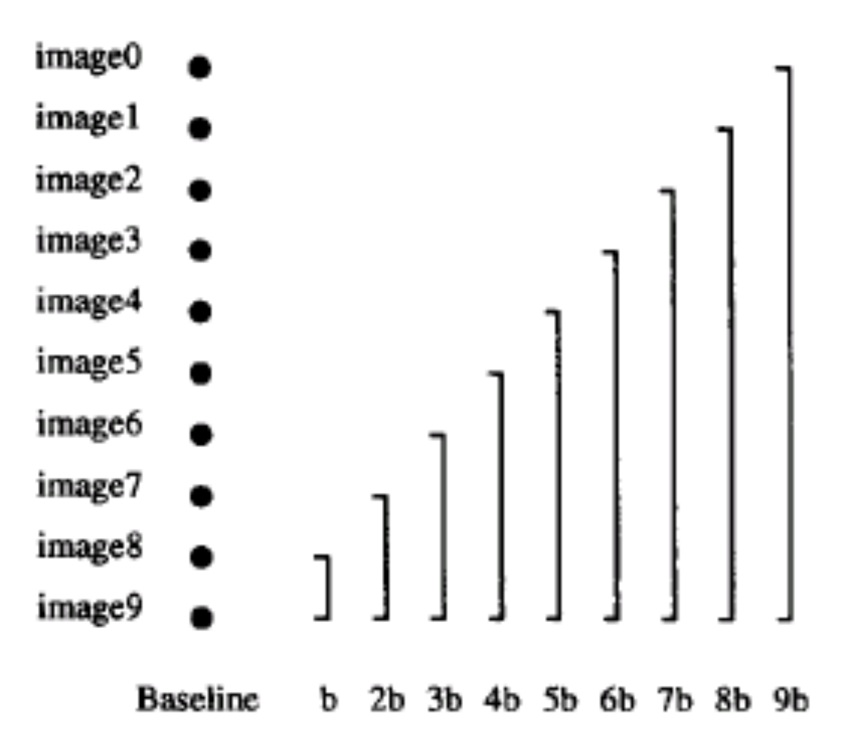
Estimated points contain some error.



Additional views reduce error.

Multiple-baseline stereo

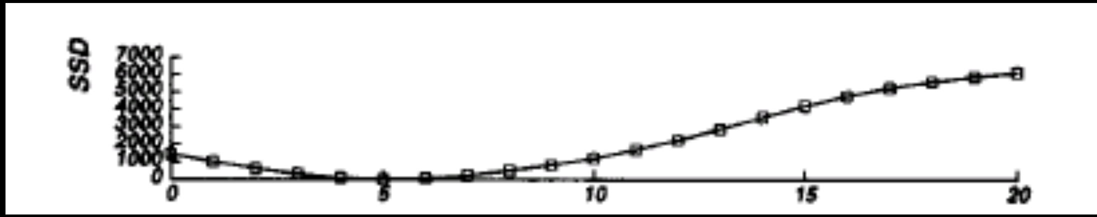
Figure 2: An example scene. The grid pattern in the background has ambiguity of matching.



M. Okutomi and T. Kanade, [A Multiple-Baseline Stereo System](), PAMI 1993

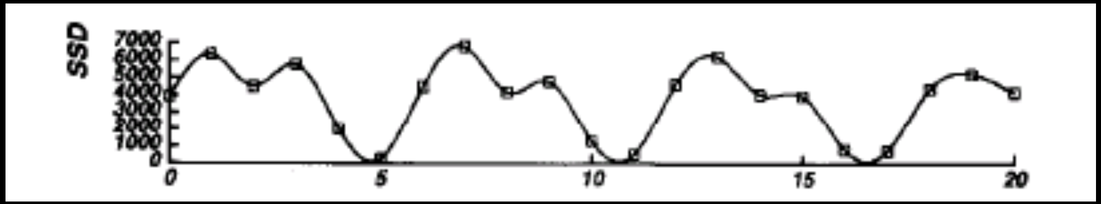
Multiple-baseline stereo

- For each pixel in reference image, simultaneously compute matching scores w.r.t. all the other images

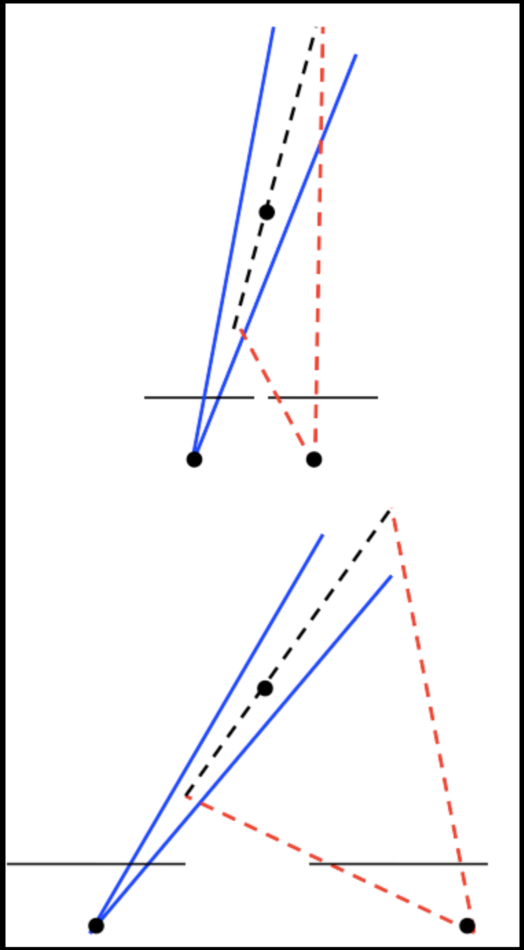


1/z

pixel matching score



1/z



M. Okutomi and T. Kanade, A Multiple-Baseline Stereo System, PAMI 1993

Multiple-baseline stereo

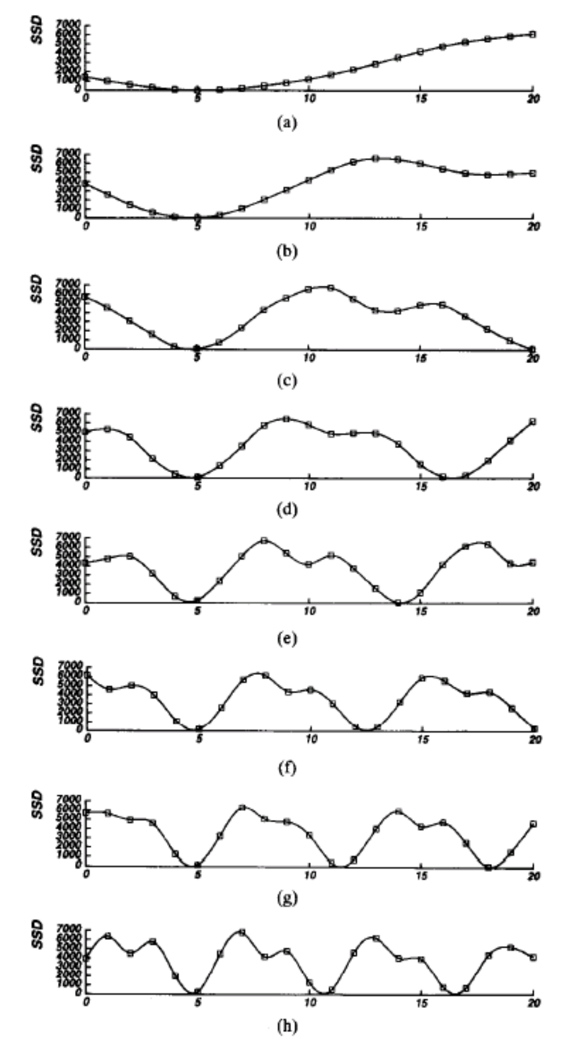


Fig. 5. SSD values versus inverse distance: (a) $B = b$; (b) $B = 2b$; (c) $B = 3b$; (d) $B = 4b$; (e) $B = 5b$; (f) $B = 6b$; (g) $B = 7b$; (h) $B = 8b$. The horizontal axis is normalized such that $8bF = 1$.

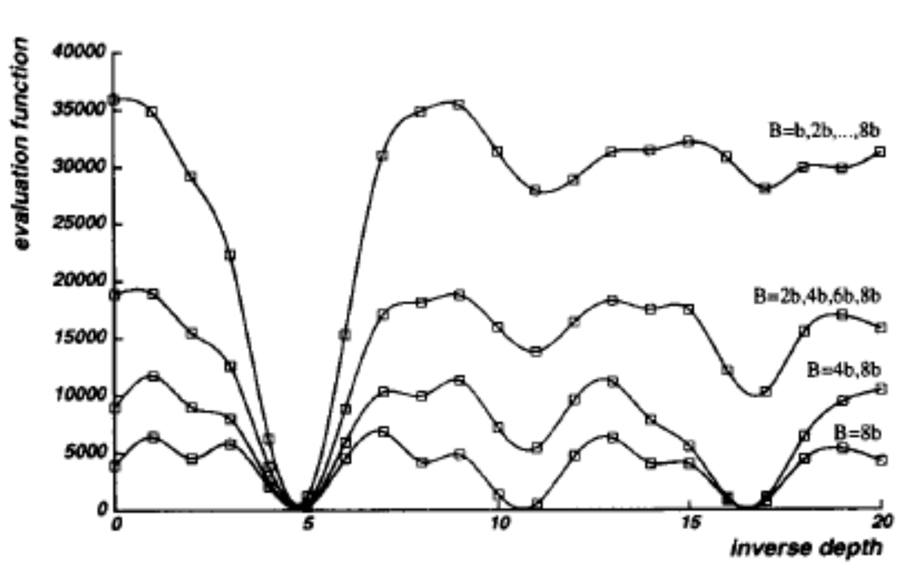
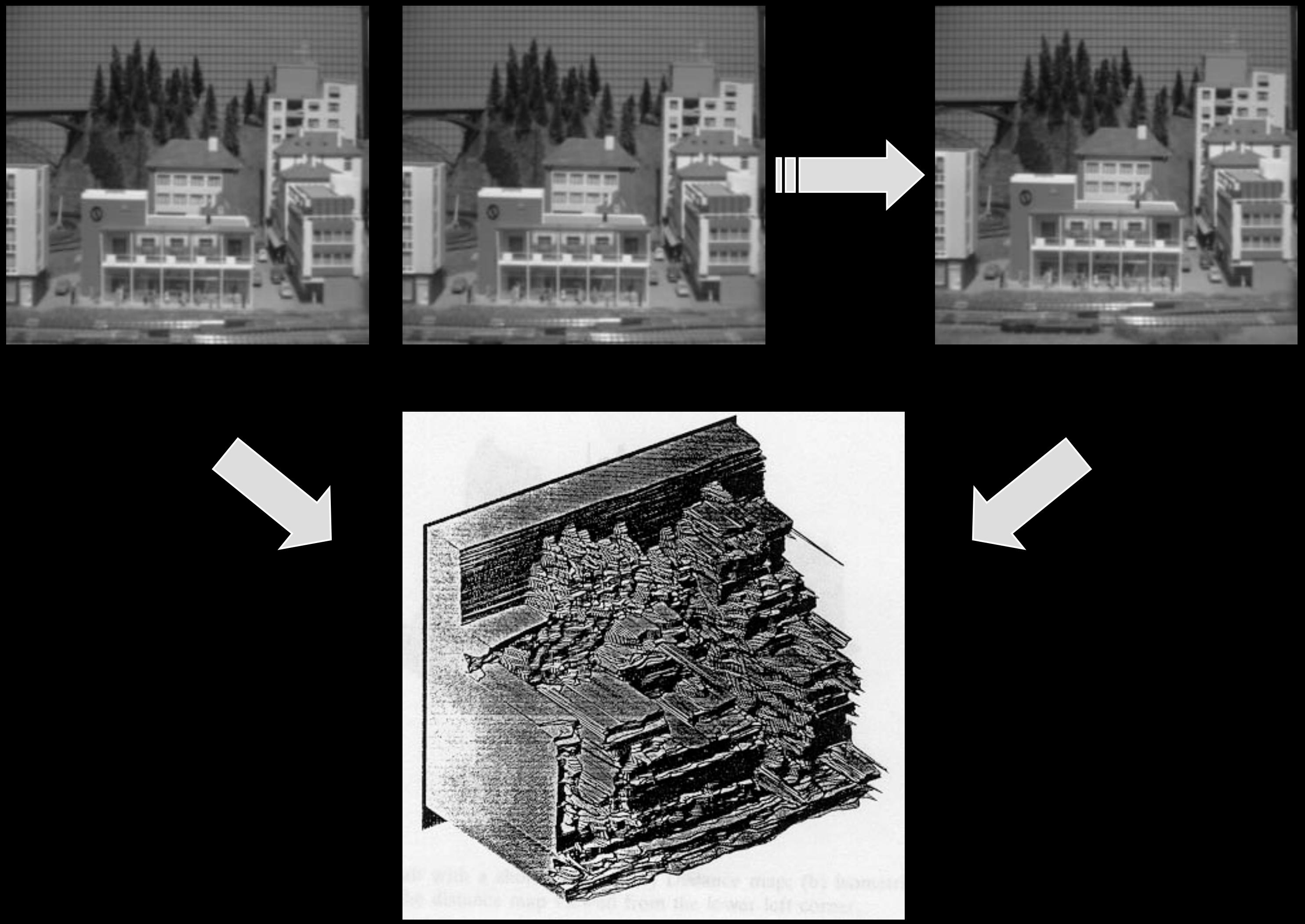


Fig. 7. Combining multiple baseline stereo pairs.

M. Okutomi and T. Kanade, A Multiple-Baseline Stereo System, PAMI 1993

Multiple-baseline stereo



M. Okutomi and T. Kanade, [A Multiple-Baseline Stereo System](), PAMI 1993

Deep Learning Approaches to Stereo Depth Estimation

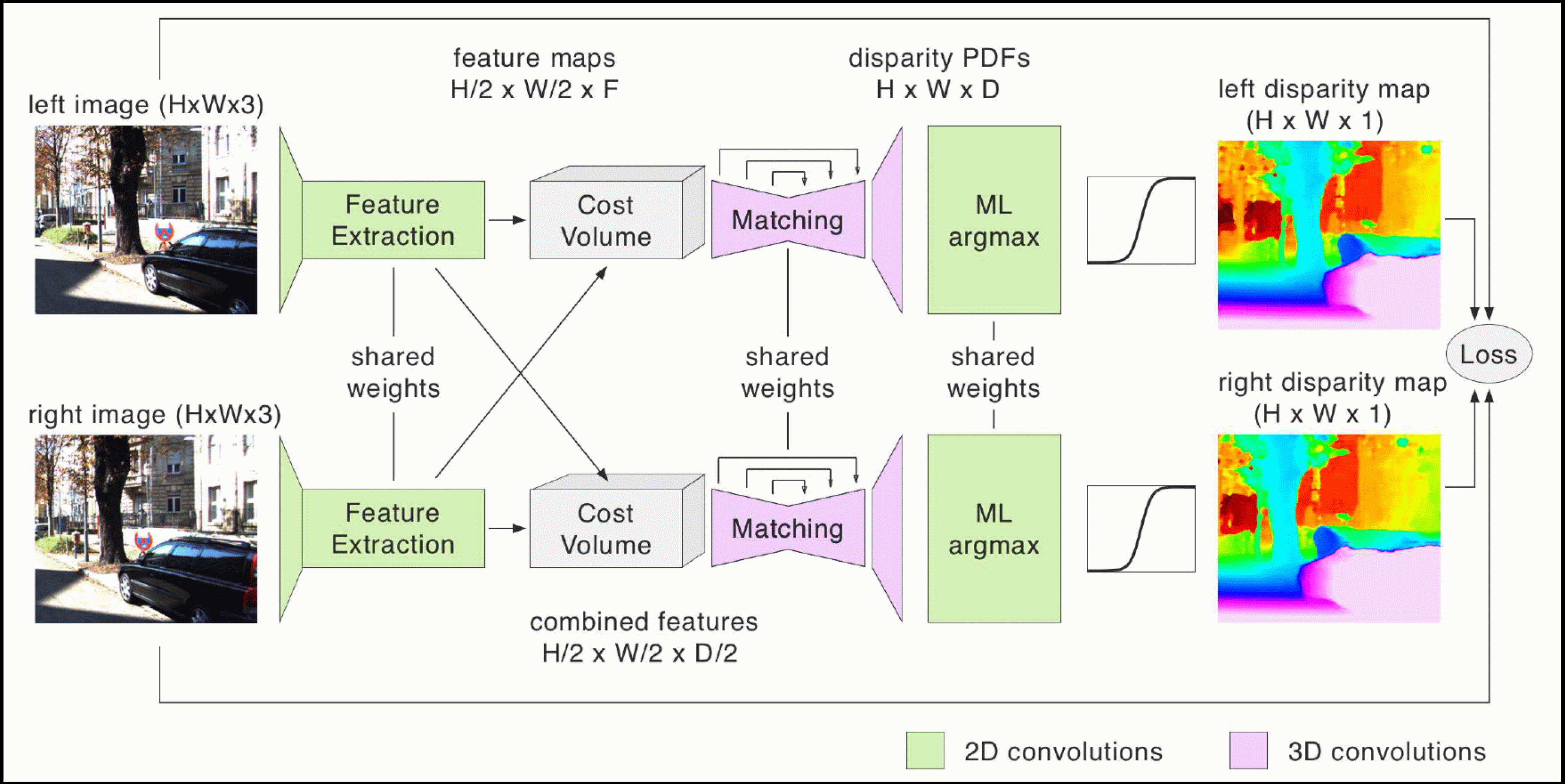
CVPR 2018

On the Importance of Stereo for Accurate Depth Estimation: An Efficient Semi-Supervised Deep Neural Network Approach

Nikolai Smolyanskiy   Alexey Kamenev   Stan Birchfield
NVIDIA

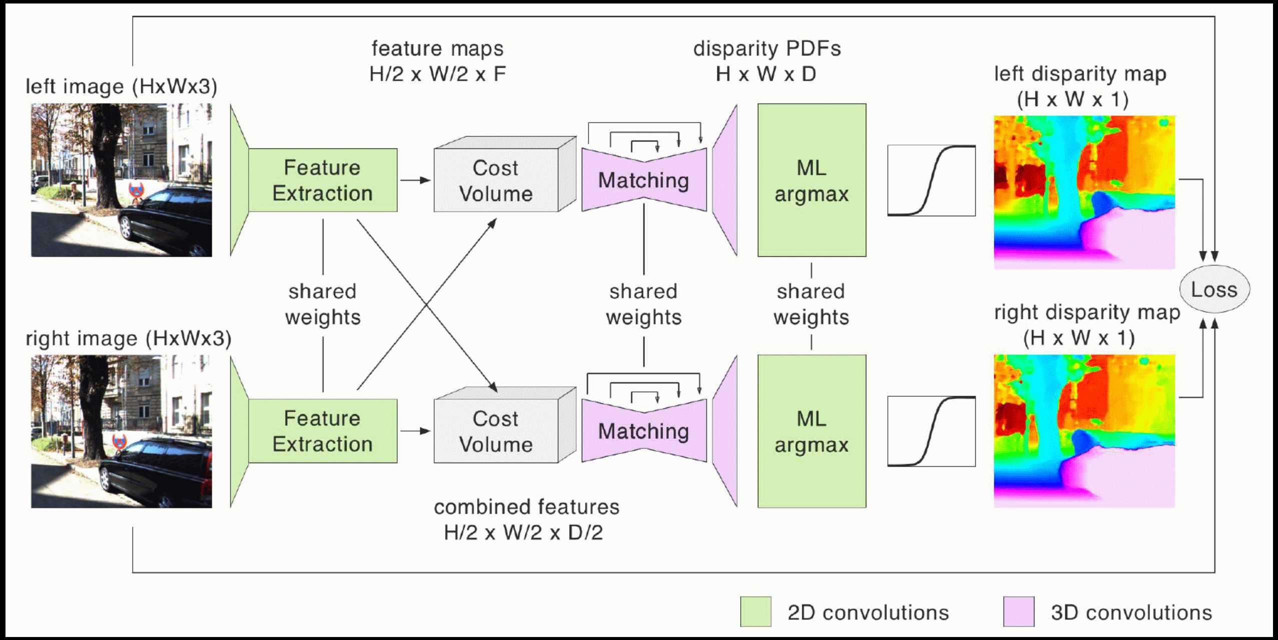
- A novel semi-supervised learning approach to training a deep stereo neural network
- A novel architecture containing a machine-learned argmax layer
- A custom runtime that enables a smaller version of our stereo DNN to run on an embedded GPU
- Competitive results are shown on the KITTI 2015 stereo dataset
- Evaluate the recent progress of stereo algorithms: measure impact on accuracy of various design criteria

Binocular Stereo Network



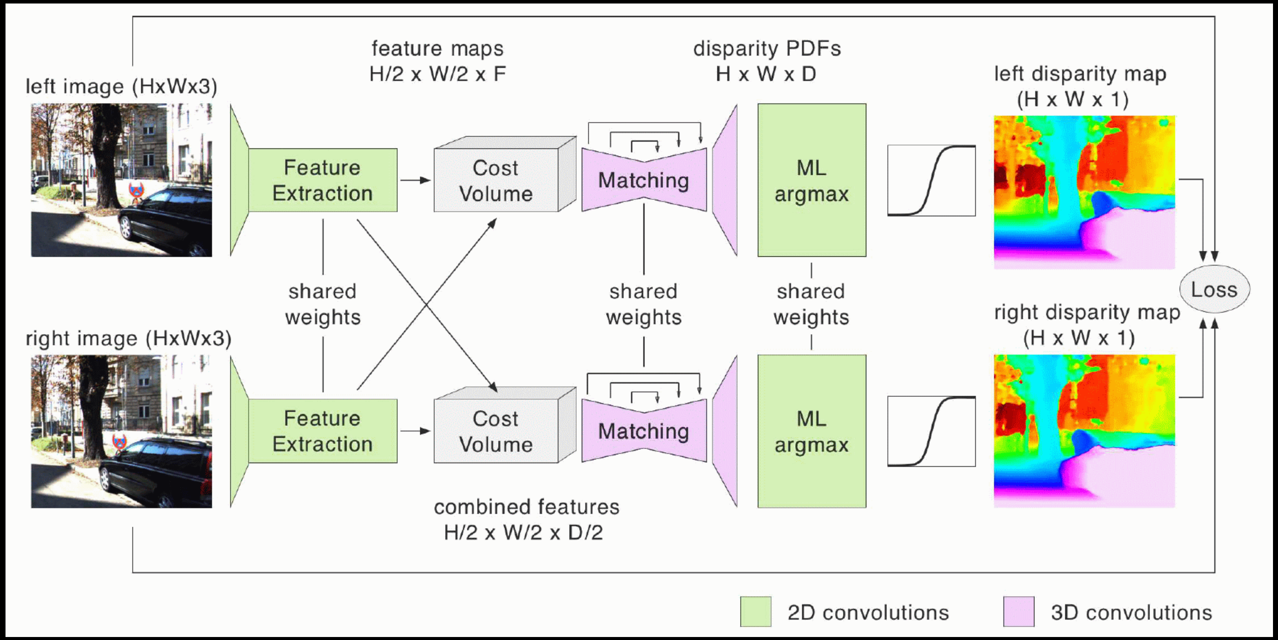
- Similar to traditional Global Matching methods but with Learned Features
- Semi-supervised because Photometric error between left-right images is used in the Loss function

Binocular Stereo Network : Highlights



- Feature Extraction:
  - ResNet-18 working on half resolution inputs to create an H/2 x W/2 x F (=32) Tensor
- Cost Volume:
  - Sweep the disparity range with left-to-right and right-to-left disparity matching
  - Creates H/2 x W/2 x 2F x D/2 cost volumes
- Matcher:
  - 3D Coder/Decoder network to compute matches while sharing weights for L-R and R-L matching
  - Upsampling to produce H x W x D x 1 Left and Right Tensors as the final matching costs between the two images

Binocular Stereo Network : Highlights



- ML-Argmax:
  - Do not use Soft argmax because semi-local context may not have been fully exploited by previous layers
  - Sequence of 2D Convolutions with shared weights for L-R and R-L matching costs
  - Produce a single value for each pixel after passing through a sigmoid
- Claim that ML-Argmax is better at handling uniform or multimodal probability distributions than soft argmax.
- Yields more stable convergence during training

Binocular Stereo Network : Loss Function

- Supervised with Sparse Lidar Ground Truth + Self-supervised via L-R / R-L Photometric consistency with D-warps

$$L = \lambda_1 E_{image} + \lambda_2 E_{lidar} + \lambda_3 E_{lr} + \lambda_4 E_{ds}$$

- Photometric Consistency: $E_{image} = E^l_{image} + E^r_{image}$ $$E^l_{image} = \frac{1}{n}\sum_{i,j} \alpha \frac{1 - SSIM(I^l_{ij}, \tilde{I}^l_{ij})}{2} + (1-\alpha)|I^l_{ij} - \tilde{I}^l_{ij}|$$

- Warp R with left disparity and L with right disparity : $\tilde{I}^l = w_{rl}(I_r, d_l)$ $\quad w_{rl}(I, d) = (x, y) \mapsto I(x + d(x,y), y)$

- Correlation between L and Warped-R and R & Warped-L : $$SSIM(x, y) = \left(\frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}\right)\left(\frac{2\sigma_{xy} + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}\right)$$

- Minimize errors between estimates and sparse Lidar data: $E_{lidar} = |d_l - \bar{d}_l| + |d_r - \bar{d}_r|$

- Left-Right disparity consistency check: $$E_{lr} = \frac{1}{n}\sum_{ij}|d^l_{ij} - \tilde{d}^l_{ij}| + \frac{1}{n}\sum_{ij}|d^r_{ij} - \tilde{d}^r_{ij}|$$ $$\tilde{d}^l = w_{rl}(d_r, d_l) \qquad \tilde{d}^r = w_{lr}(d_l, d_r)$$

- Prefer piecewise smooth disparities: $E_{ds} = E^l_{ds} + E^r_{ds}$ $$E^l_{ds} = \frac{1}{n}\sum_{i,j}|\partial_x d^l_{ij}|e^{-\|\partial_x I^l_{i,j}\|} + |\partial_y d^l_{ij}|e^{-\|\partial_y I^l_{i,j}\|}$$

Improvements vis-à-vis Mono and Lidar only

model	lidar	photo	lidar+photo
MonoDepth [9]	-	32.8%	-
no bottleneck	21.3%	18.6%	14.5%
correlation	14.6%	13.3%	12.9%
baseline (ours)	15.0%	12.9%	8.8%

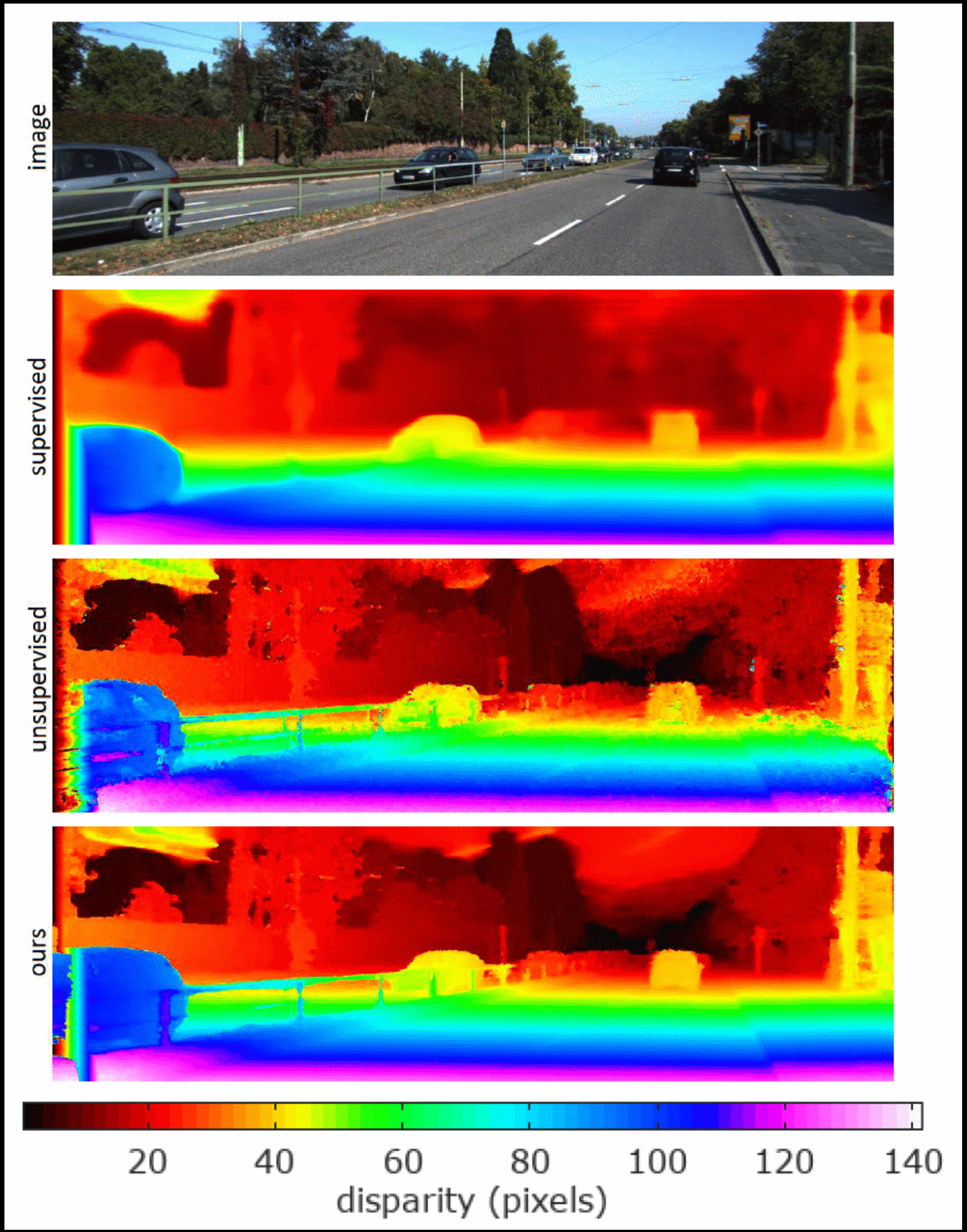
- %age of Outliers: Error at least 3 disparity levels or at least 5% [Lower is better]
- Best results are obtained by combining Supervised Lidar Loss with Self-Supervised Photometric Loss
- Photometric and LIDAR data complement each other:
  - LIDAR is accurate at all depths, but its sparsity leads to blurrier results, and it misses the fine structure
  - Photometric consistency allows the network to recover fine-grained surfaces but suffers from loss in accuracy as depth increases

Improvements vis-à-vis Mono and Lidar only

Supervised (LIDAR) : Too Smooth

Unsupervised (Photometric) : Noisy with Distance but Preserves Thins Structures

Hybrid (Both)



ECCV 2018

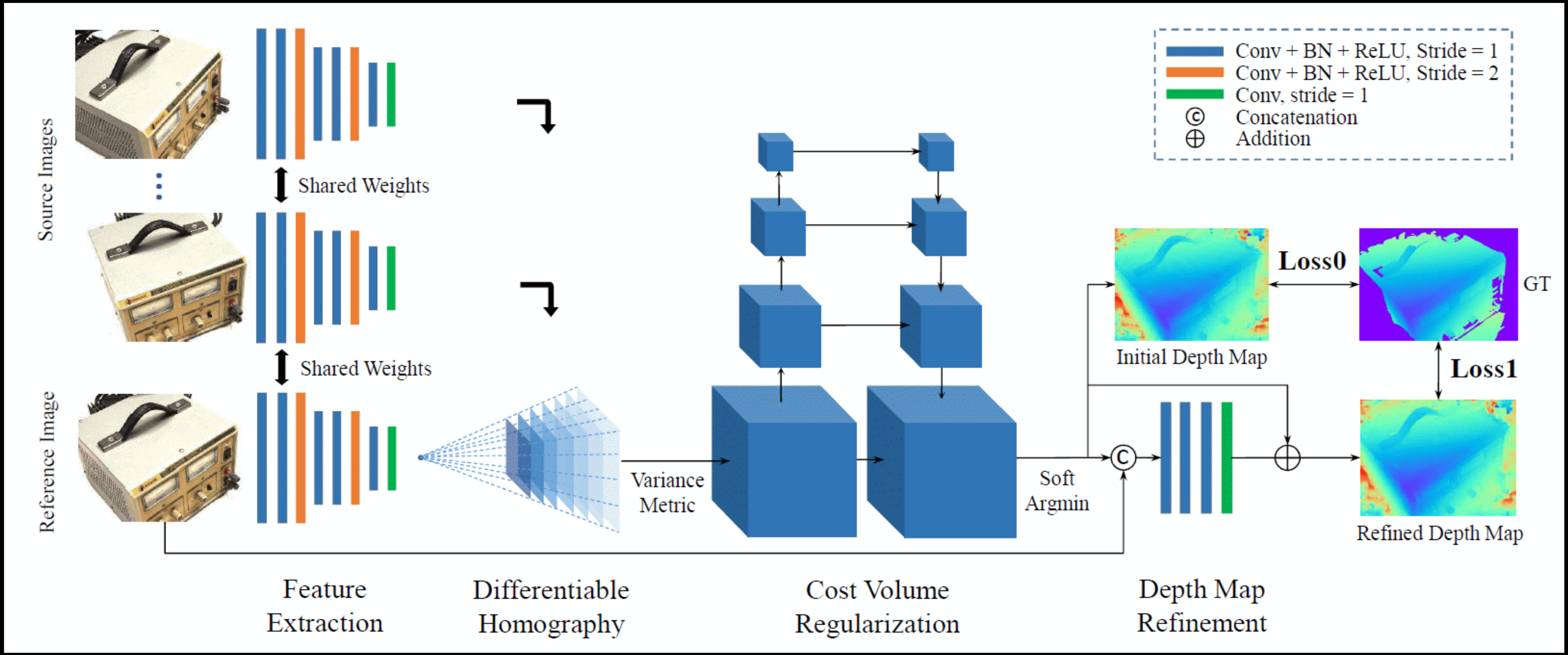
MVSNet: Depth Inference for Unstructured Multi-view Stereo

Yao Yao[1], Zixin Luo[1], Shiwei Li[1], Tian Fang[2], and Long Quan[1]

[1] The Hong Kong University of Science and Technology,
{yyaoag, zluoag, slibc, quan}@cse.ust.hk
[2] Shenzhen Zhuke Innovation Technology (Altizure),

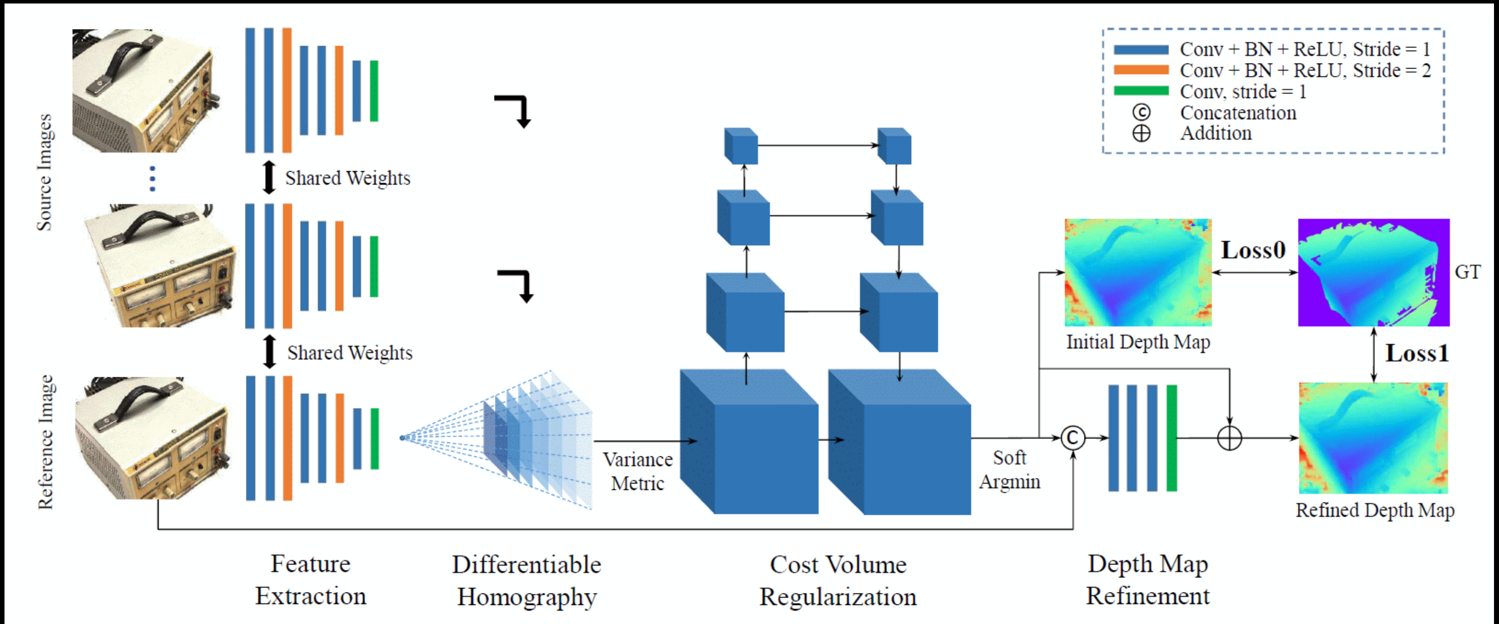
- End-to-end deep learning architecture for depth map inference from multi-view images
- Framework flexibly adapts arbitrary N-view inputs using a variance-based cost metric that maps multiple features into one cost feature

MVSNet



- Input : One reference image and several source images
- Infers the depth map for the reference image.
- Key insight : Differentiable homography warping operation that implicitly encodes camera geometries in the network to build the 3D cost volumes from 2D image features and enables the end-to-end training
- To adapt arbitrary number of source images, variance-based metric to map multiple features into one cost feature

MVSNet : Components

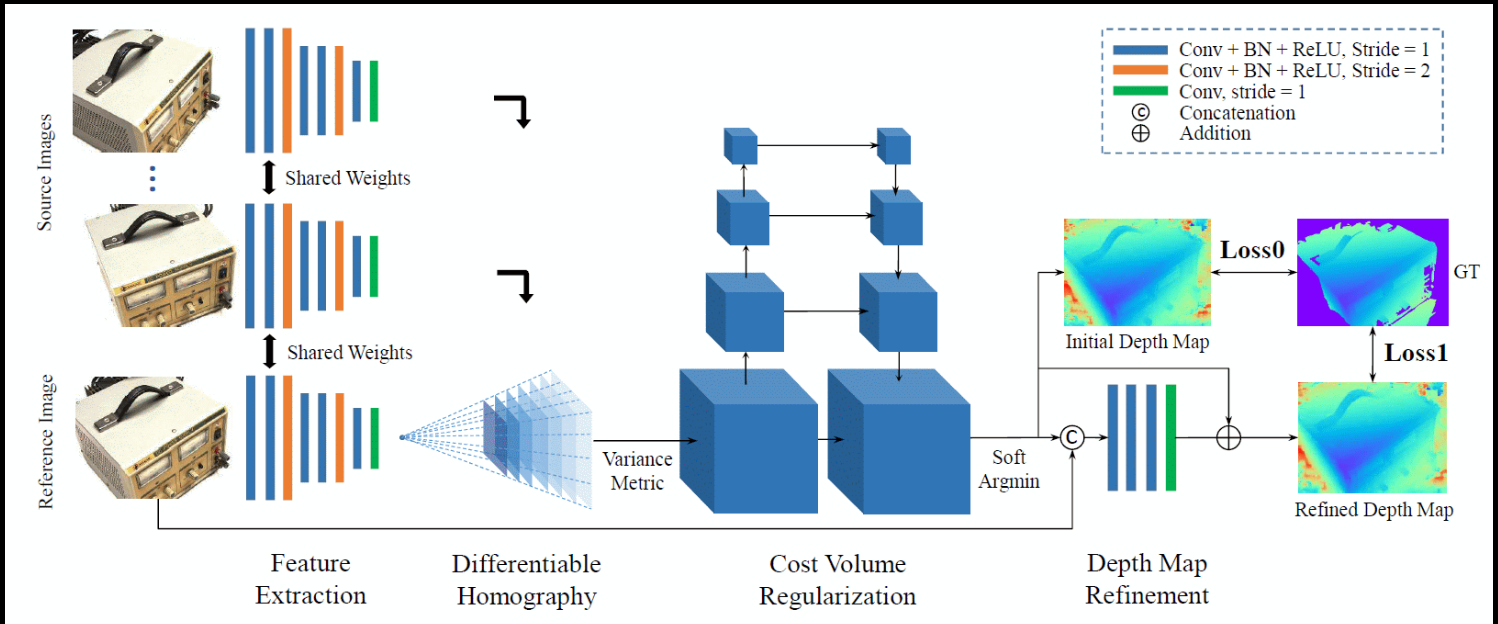


- Image Features: Extract deep features of the N input images for dense matching.
  - Eight-layer 2D CNN : Outputs are N 32-channel feature maps downsized by four in each dimension vs. input images.
- Cost Volume: All feature maps are warped into different fronto-parallel planes to form N feature volumes
  - Warping is for the corresponding Homography between the Reference and the ith image
- Cost Metric: Aggregate multiple feature volumes into one cost volume C
  - V = W/4 x H/4 x D x F

$$\mathbf{C} = \mathcal{M}(\mathbf{V}_1, \cdots, \mathbf{V}_N) = \frac{\sum_{i=1}^{N} (\mathbf{V}_i - \overline{\mathbf{V}_i})^2}{N}$$

- Cost Volume Regularization: Multi-scale 3D CNN similar to UNet ➔ Softmaxed 1-channel probability volume

MVSNet : Components



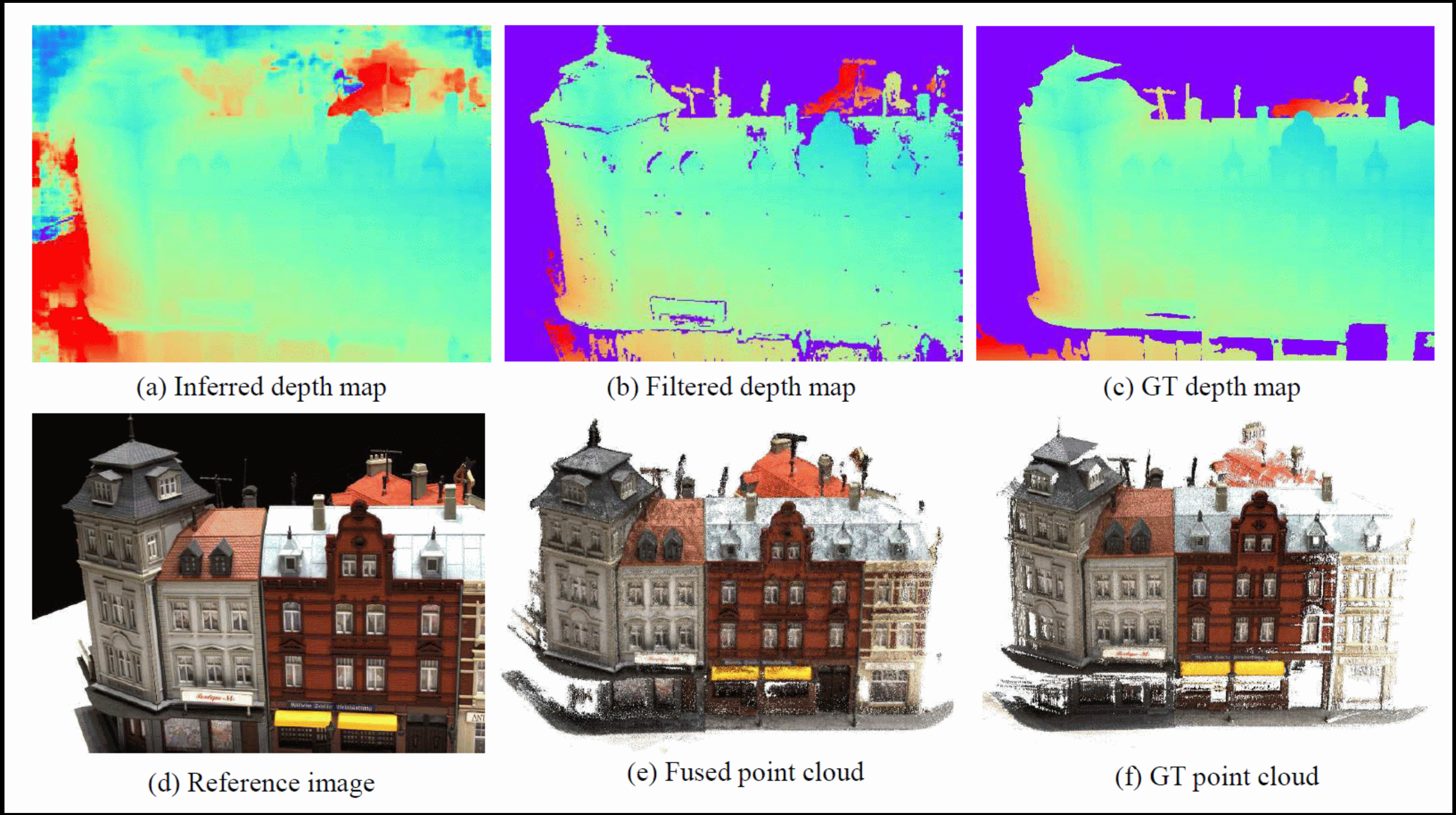
- Depth Map : Probability weighted sum over all hypotheses

$$\mathbf{D} = \sum_{d=d_{min}}^{d_{max}} d \times \mathbf{P}(d)$$

- Quality of depth: Probability sum over the four nearest depth hypotheses
- Depth Map Refinement: Depth residual learning network at the end of MVSNet
- Loss:

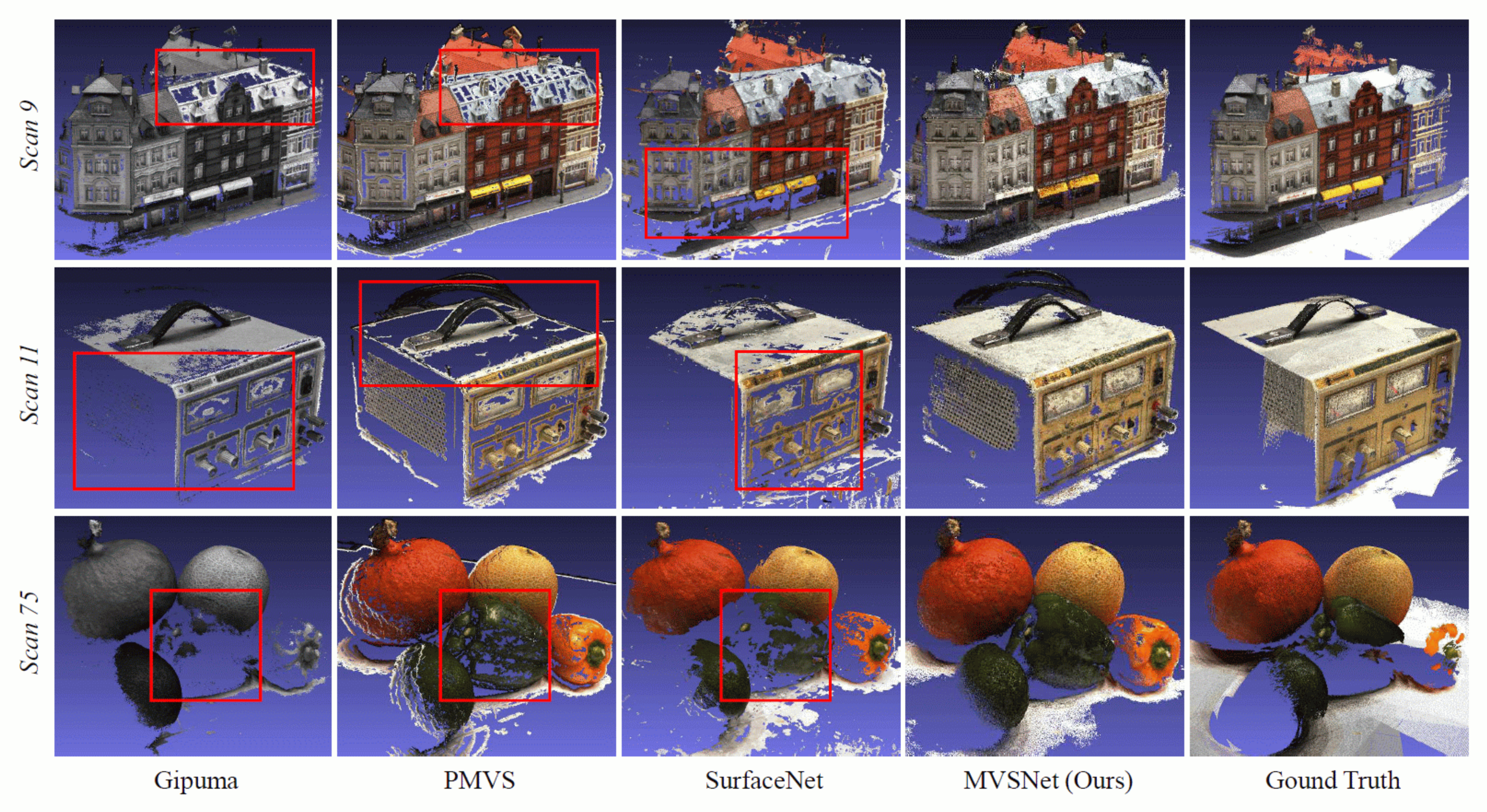
$$Loss = \sum_{p \in \mathbf{p}_{valid}} \underbrace{\|d(p) - \hat{d}_i(p)\|_1}_{Loss0} + \lambda \cdot \underbrace{\|d(p) - \hat{d}_r(p)\|_1}_{Loss1}$$

MVSNet : Results



- Depth Refinement using Photo Consistency and L-R R-L Disparity consistency check
- Depth Map Fusion

MVSNet : Results

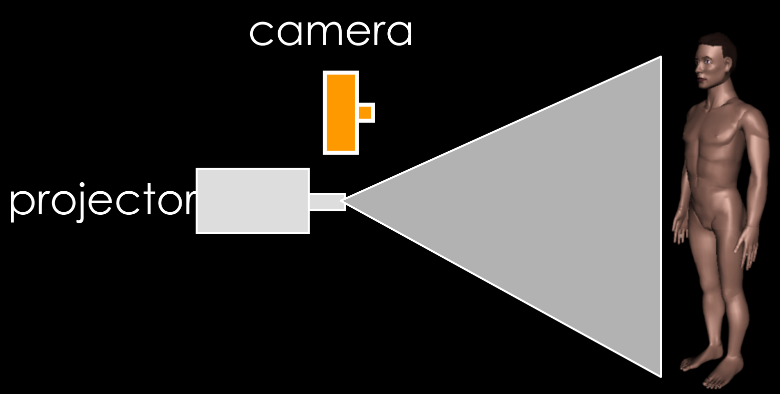


Stereo datasets

- Middlebury stereo datasets
- KITTI
- Synthetic data?

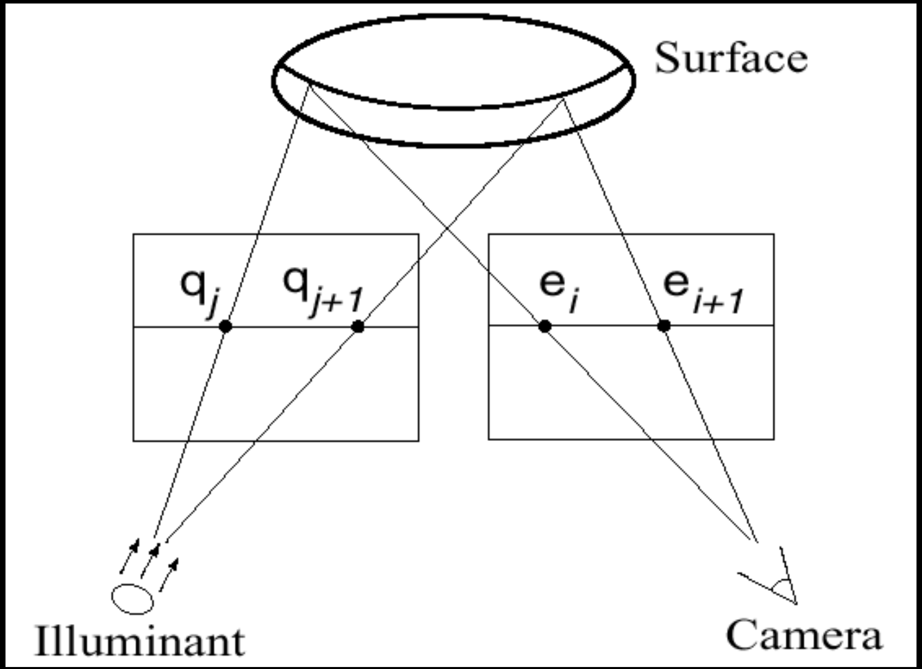
Active stereo with structured light

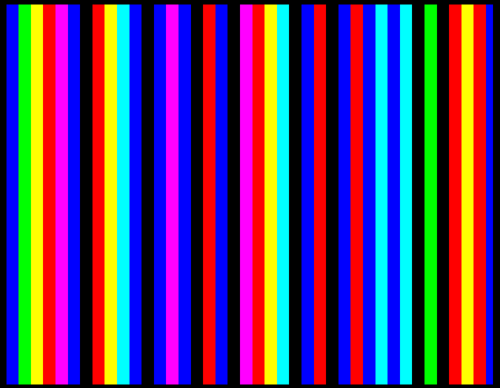
- Project "structured" light patterns onto the object
  - Simplifies the correspondence problem
  - Allows us to use only one camera

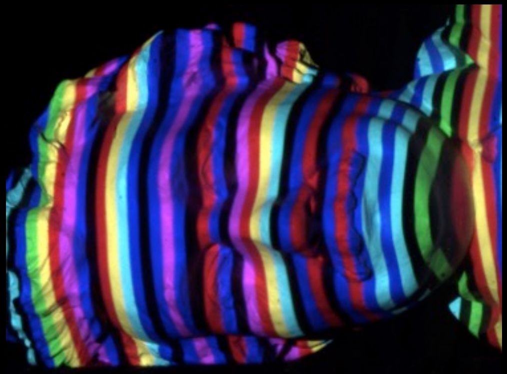


L. Zhang, B. Curless, and S. M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. *3DPVT* 2002

Active stereo with structured light

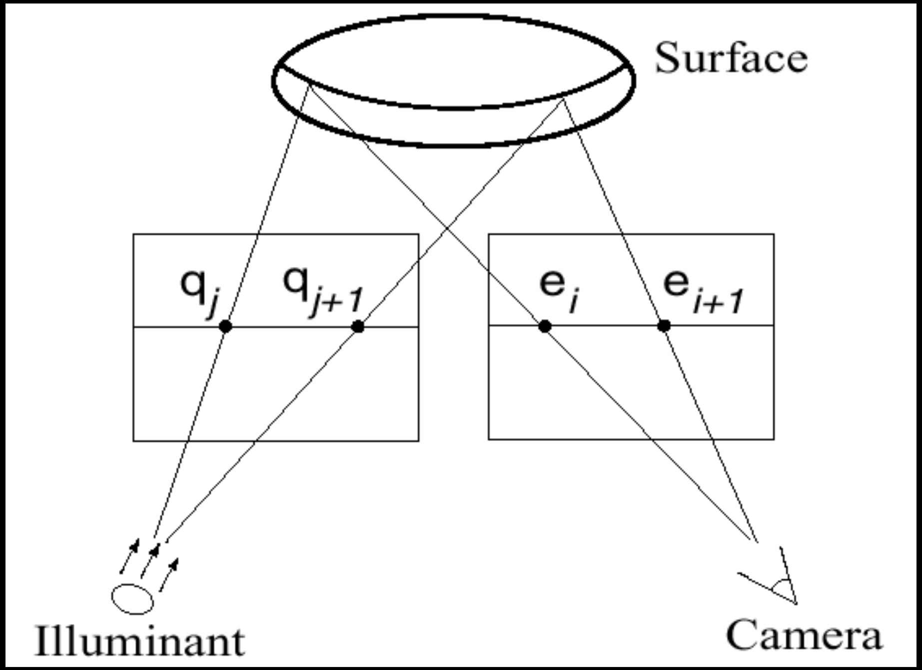


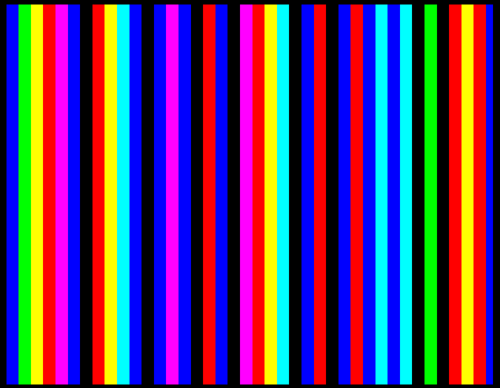


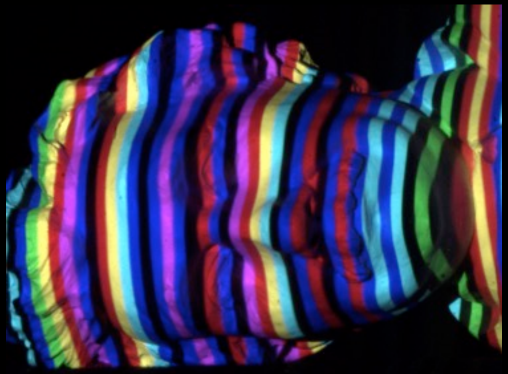


L. Zhang, B. Curless, and S. M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. *3DPVT* 2002

Active stereo with structured light





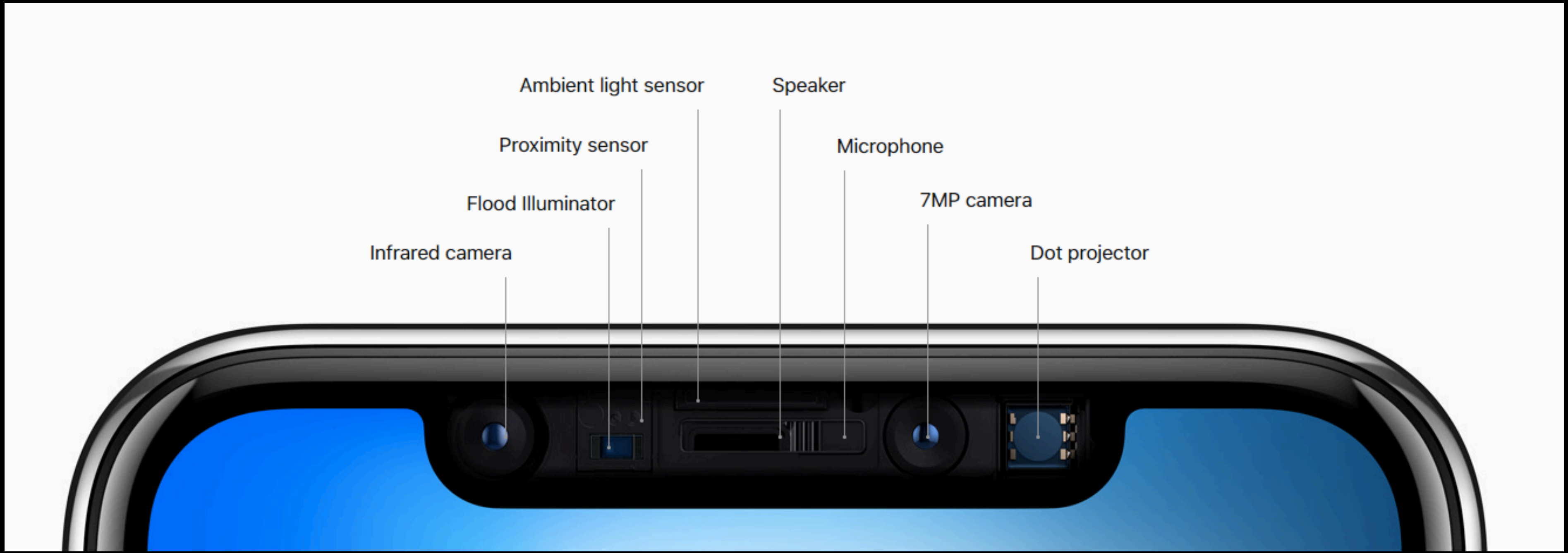


http://en.wikipedia.org/wiki/Structured-light_3D_scanner

Kinect: Structured infrared light

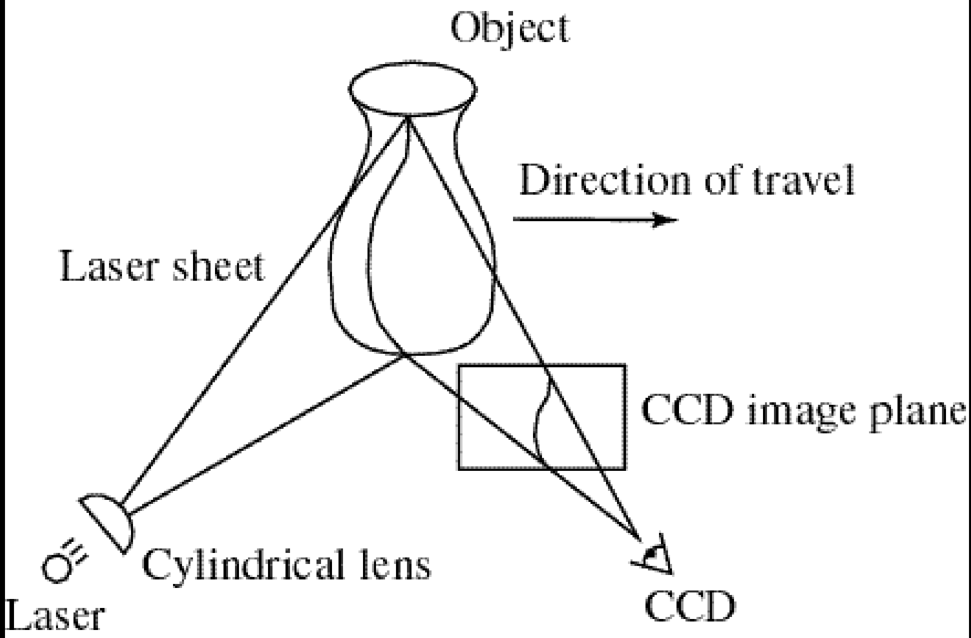
http://bbzippo.wordpress.com/2010/11/28/kinect-in-infrared/

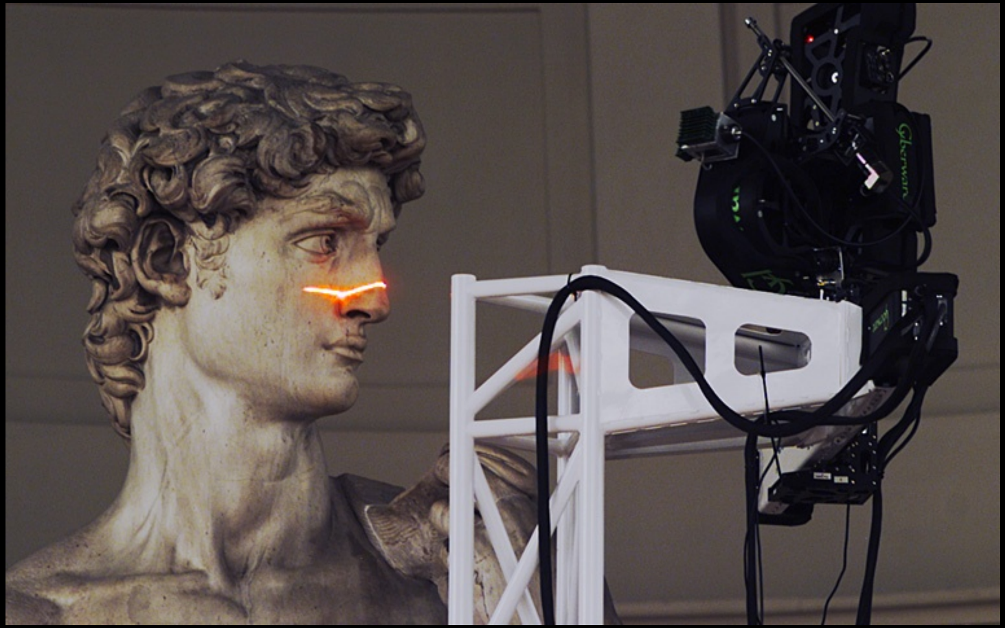
Apple TrueDepth



https://www.cnet.com/news/apple-face-id-truedepth-how-it-works/

Laser scanning





Digital Michelangelo Project
Levoy et al.
http://graphics.stanford.edu/projects/mich/

- Optical triangulation
  - Project a single stripe of laser light
  - Scan it across the surface of the object
  - This is a very precise version of structured light scanning

Source: S. Seitz

Laser scanned models

The Digital Michelangelo Project, Levoy et al.

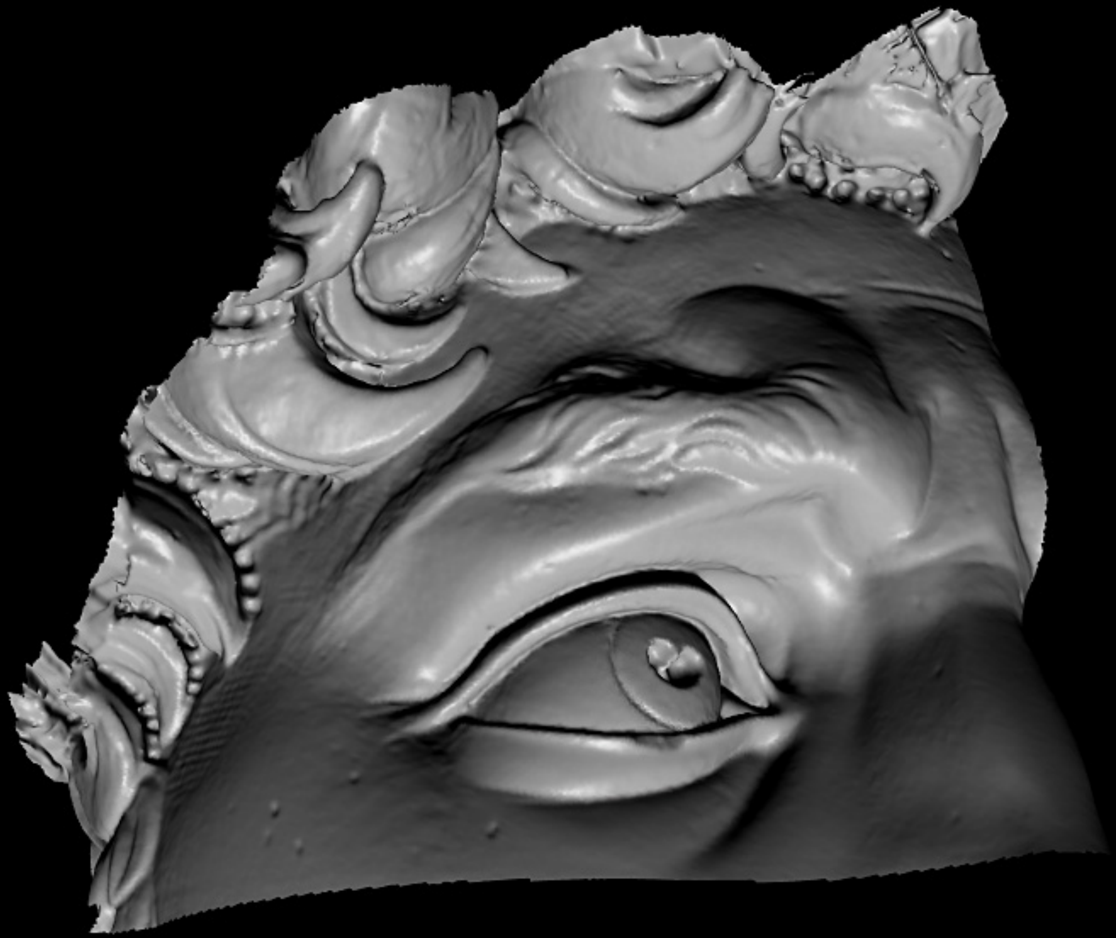
Source: S. Seitz

Laser scanned models

The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

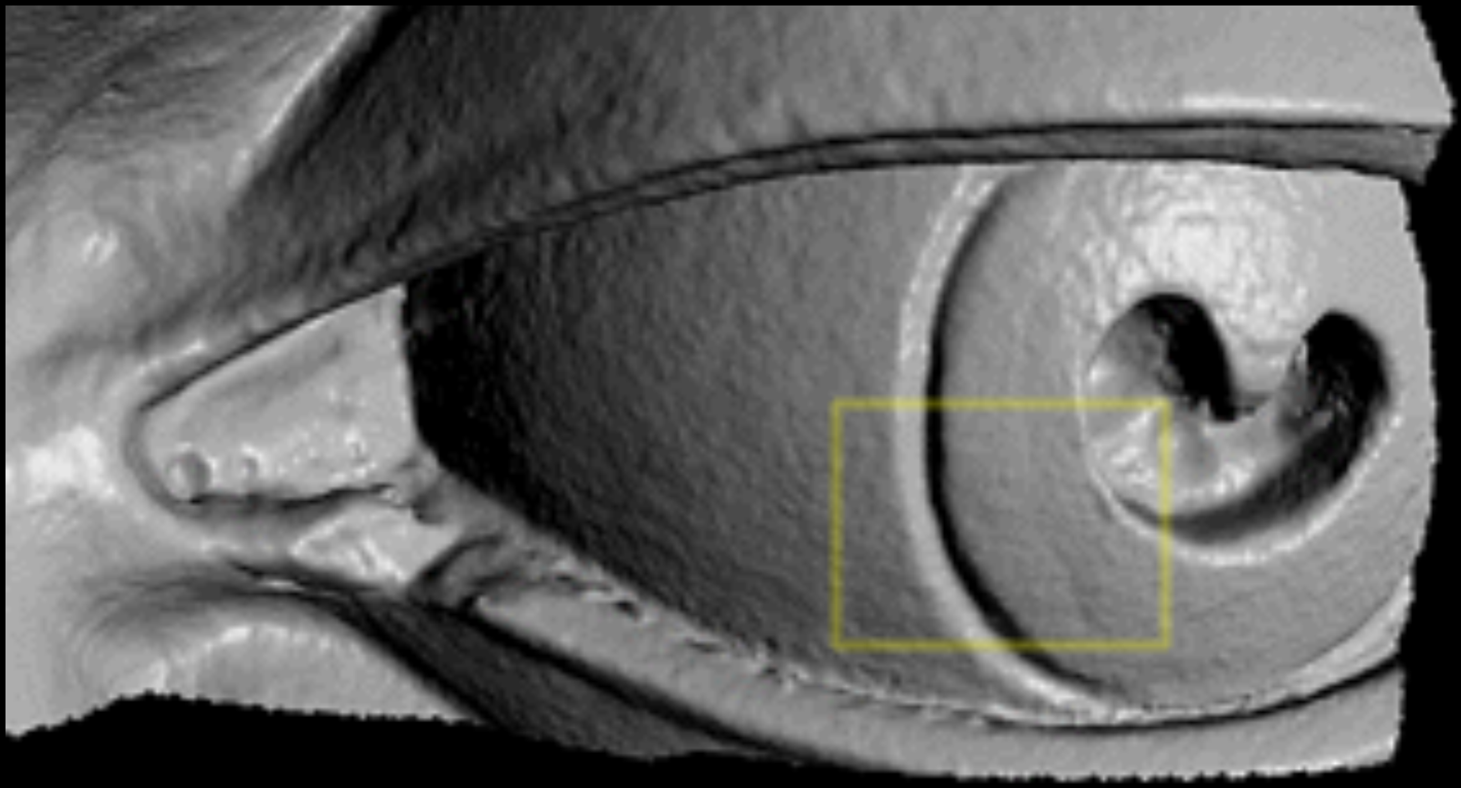
Laser scanned models



The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

Laser scanned models

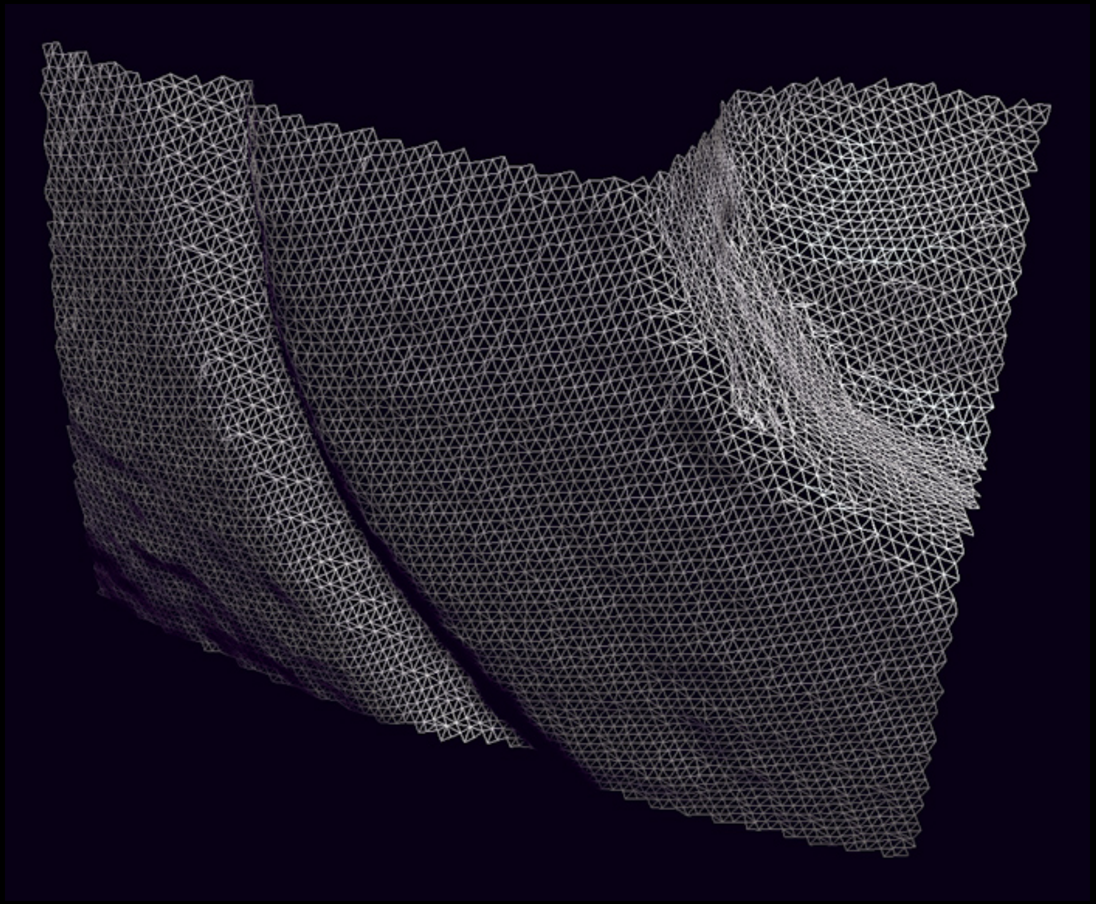


The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

Laser scanned models

1.0 mm resolution (56 million triangles)

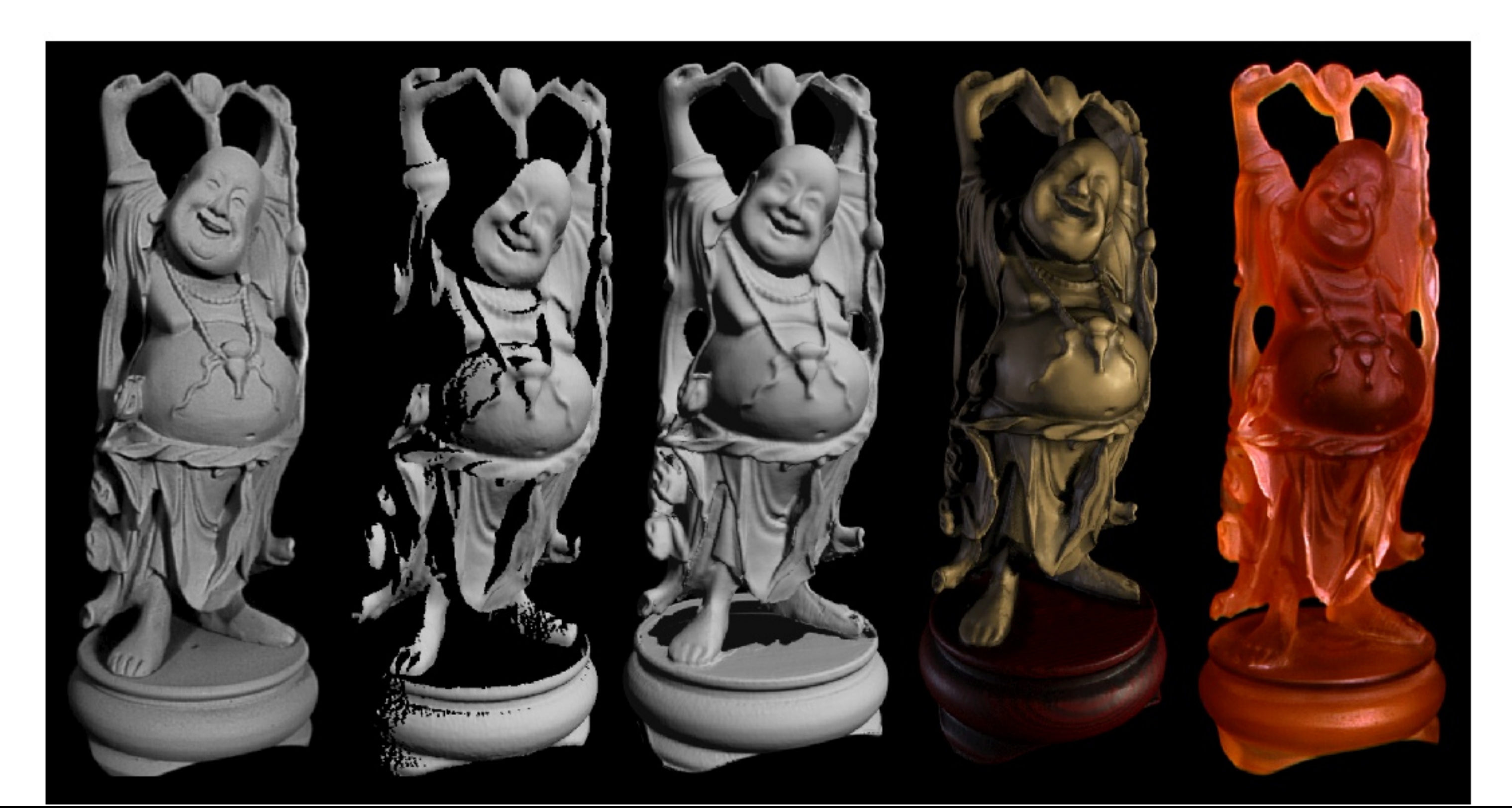


The Digital Michelangelo Project, Levoy et al.

Source: S. Seitz

Aligning range images

- A single range scan is not sufficient to describe a complex surface
- Need techniques to register multiple range images



B. Curless and M. Levoy, A Volumetric Method for Building Complex Models from Range Images, SIGGRAPH 1996

Hybrid Stereo Camera

An IBR Approach for Synthesis of Very High Resolution Stereo Image Sequences

Limitations on IMAX 3D Content Creation

Live Action Content

- Camera is very large.
- Requires two strips of large format film.
- Size of camera and cost of film limits production.

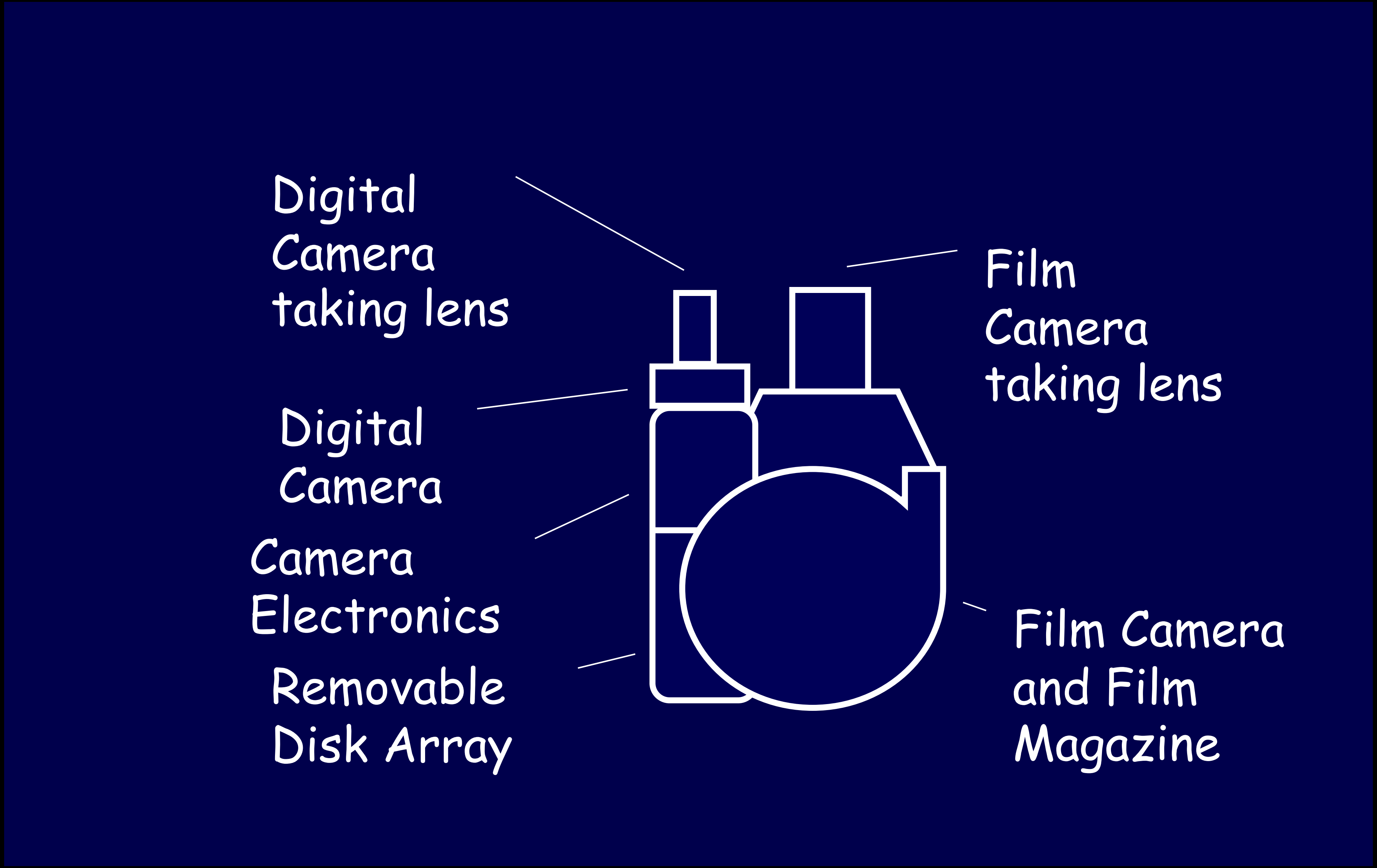
70 mm

15 perforations

CG Content

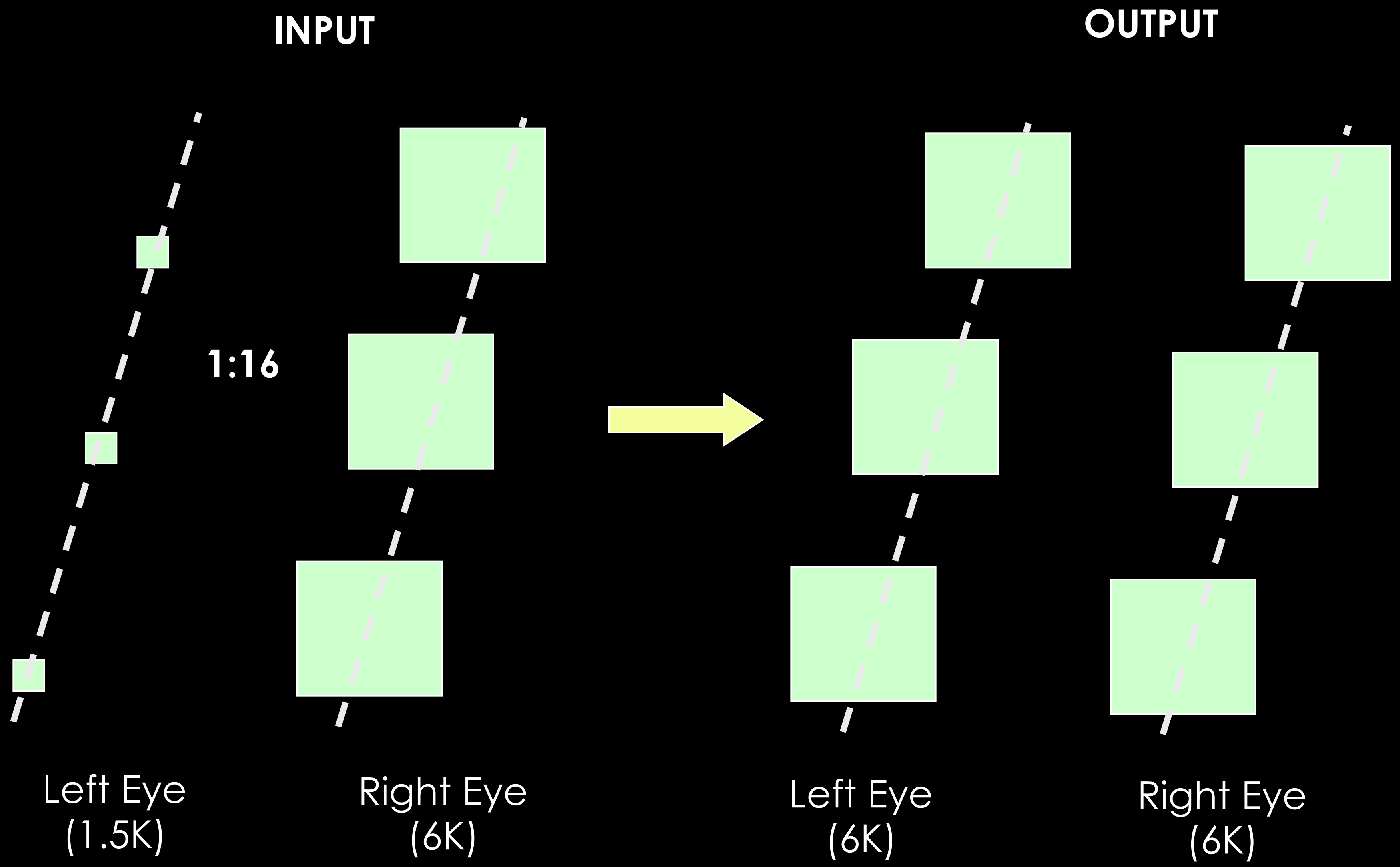
- 6-14 hours rendering time per frame !

Solution: Hybrid Stereo Camera



Hybrid Stereo Camera

... pure upsampling is not an option ...

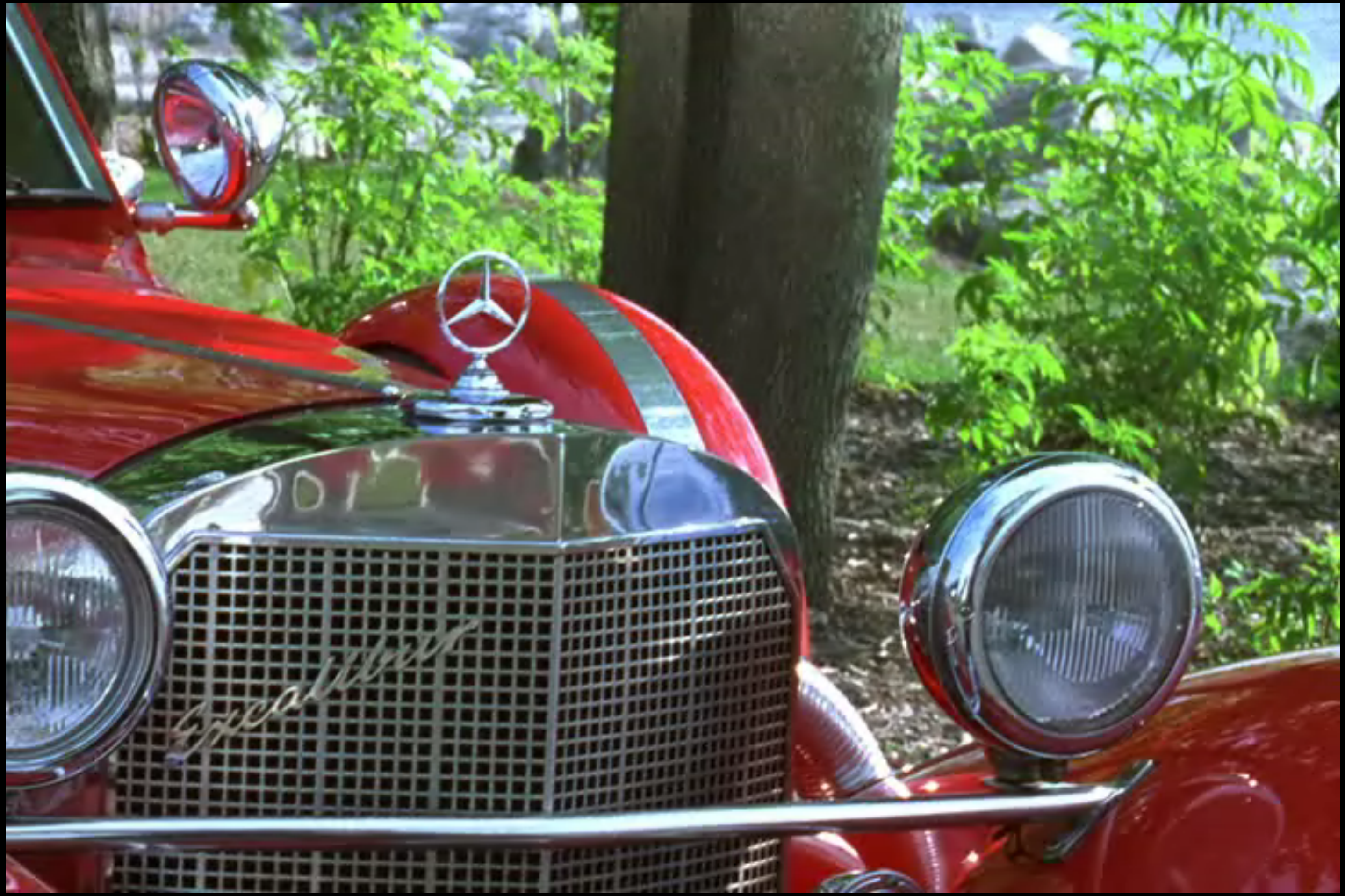


Live Action Sequence

Live Action : Hybrid Input

Left

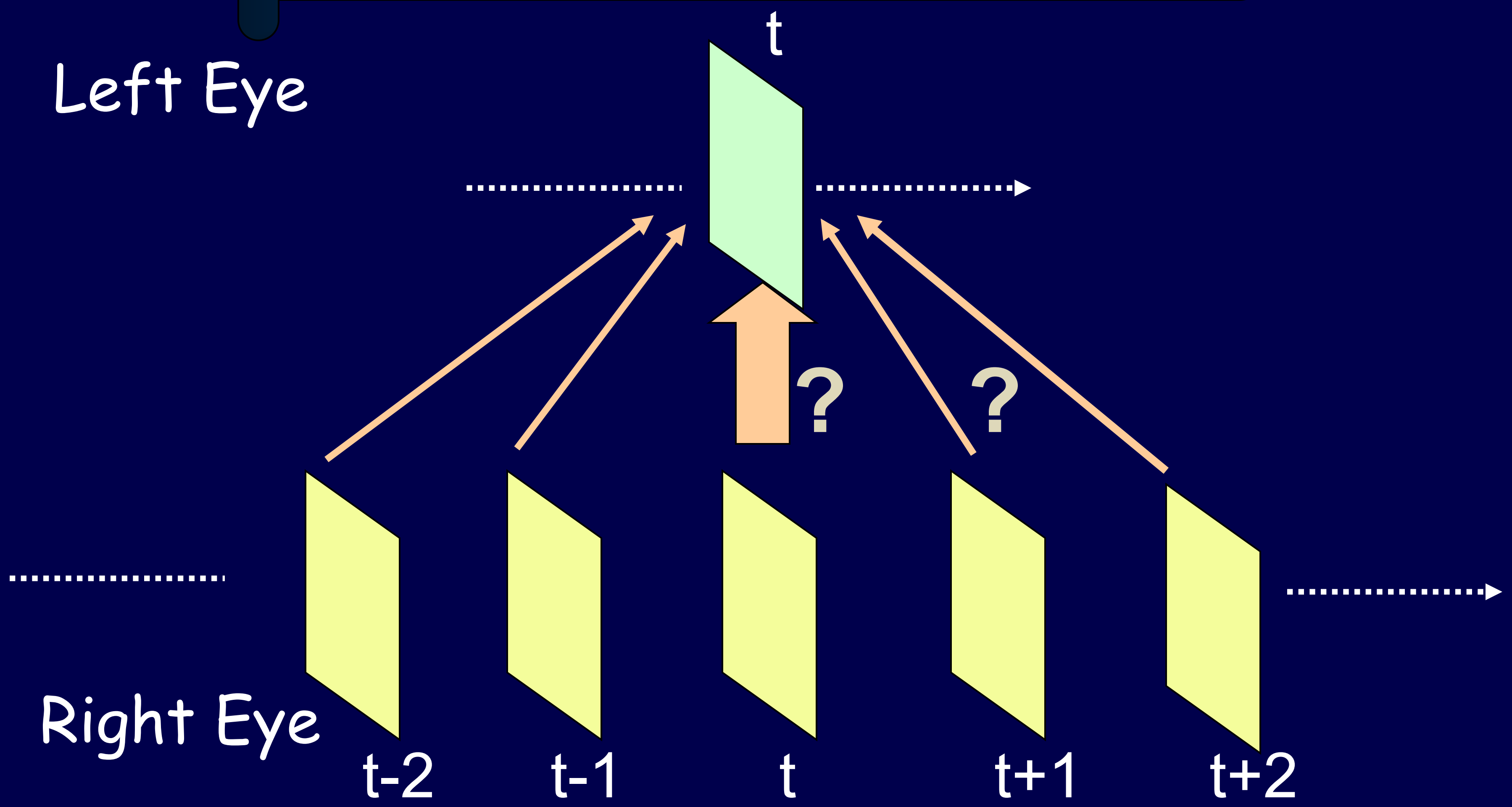
Right



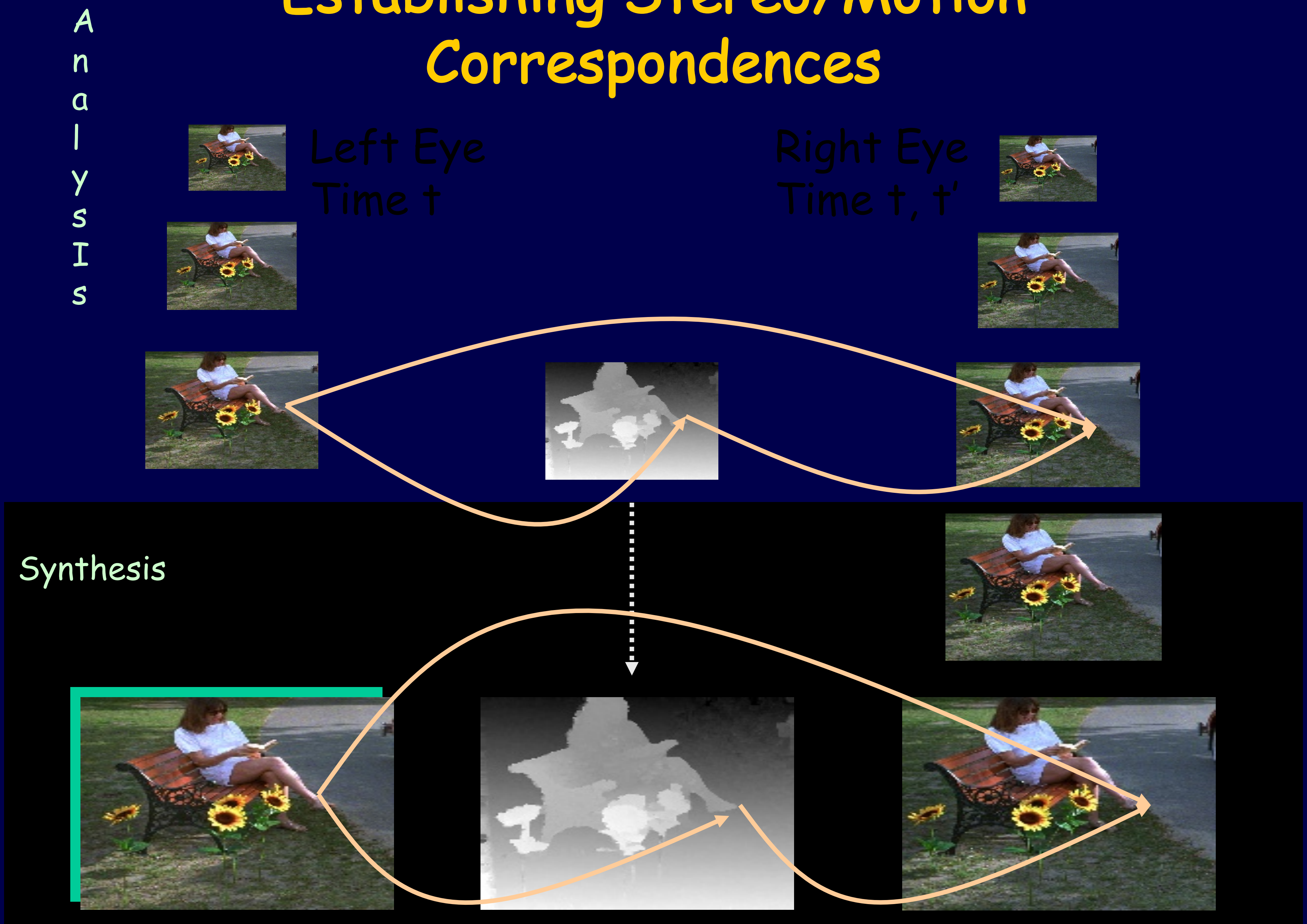
Live Action : Hybrid Input

How can the Hybrid Camera be Realized ?

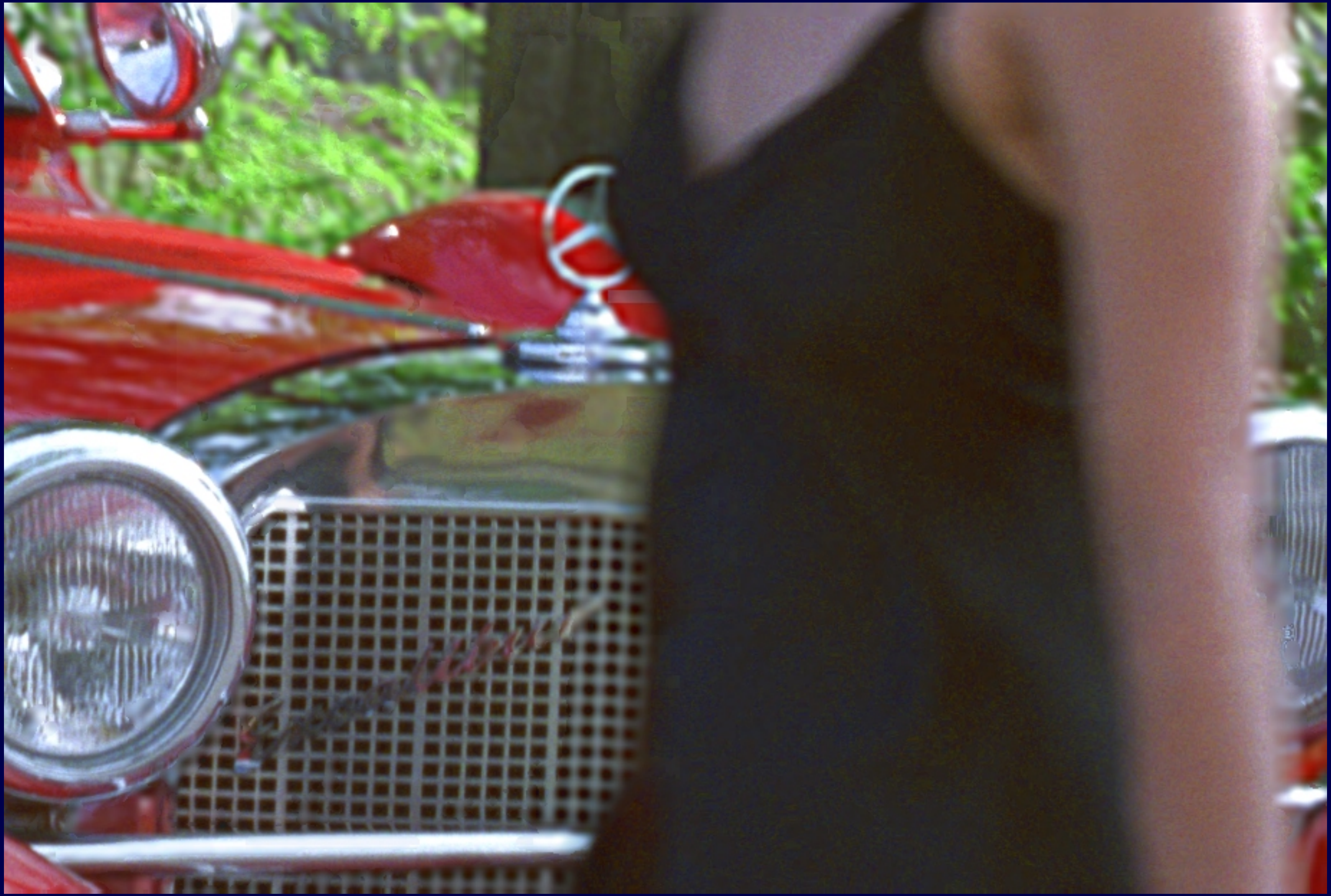
Render the High-Res content
into the coordinate system
of the Low-Res Frame !



Establishing Stereo/Motion Correspondences



Synthesis vs. Up-resing : Live Action



Synthesis vs. Up-resing : CG Animation

Thanks !

This was fun!