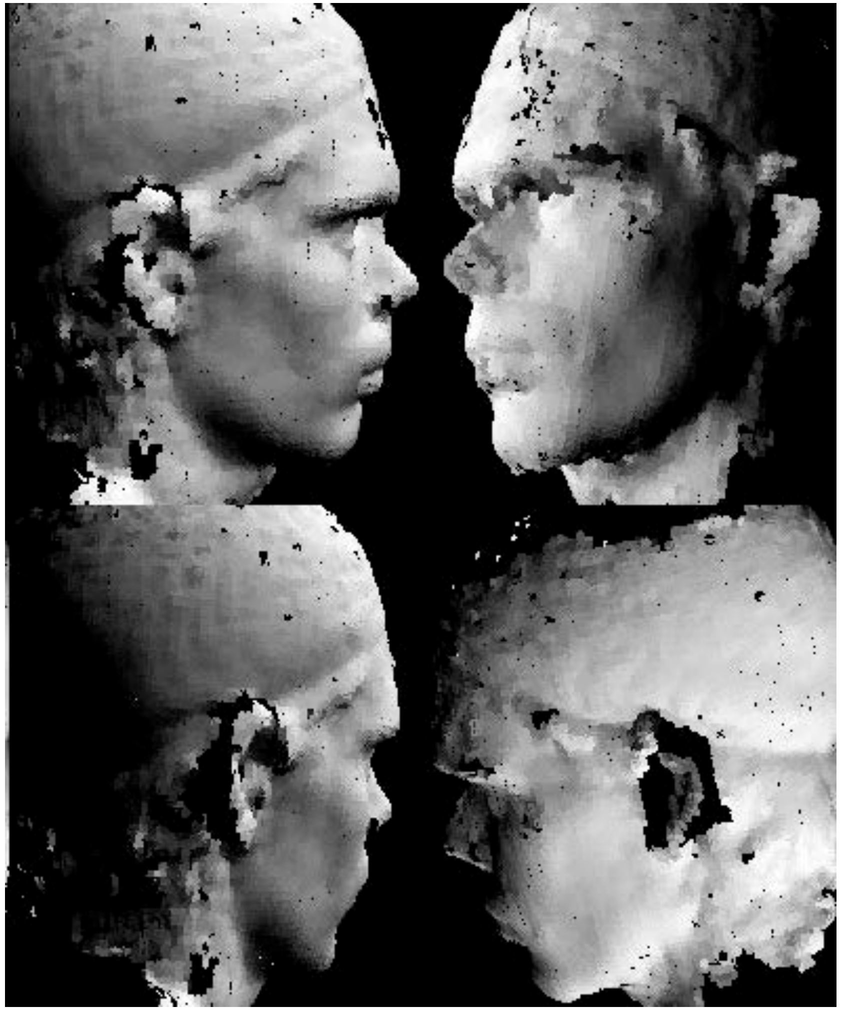
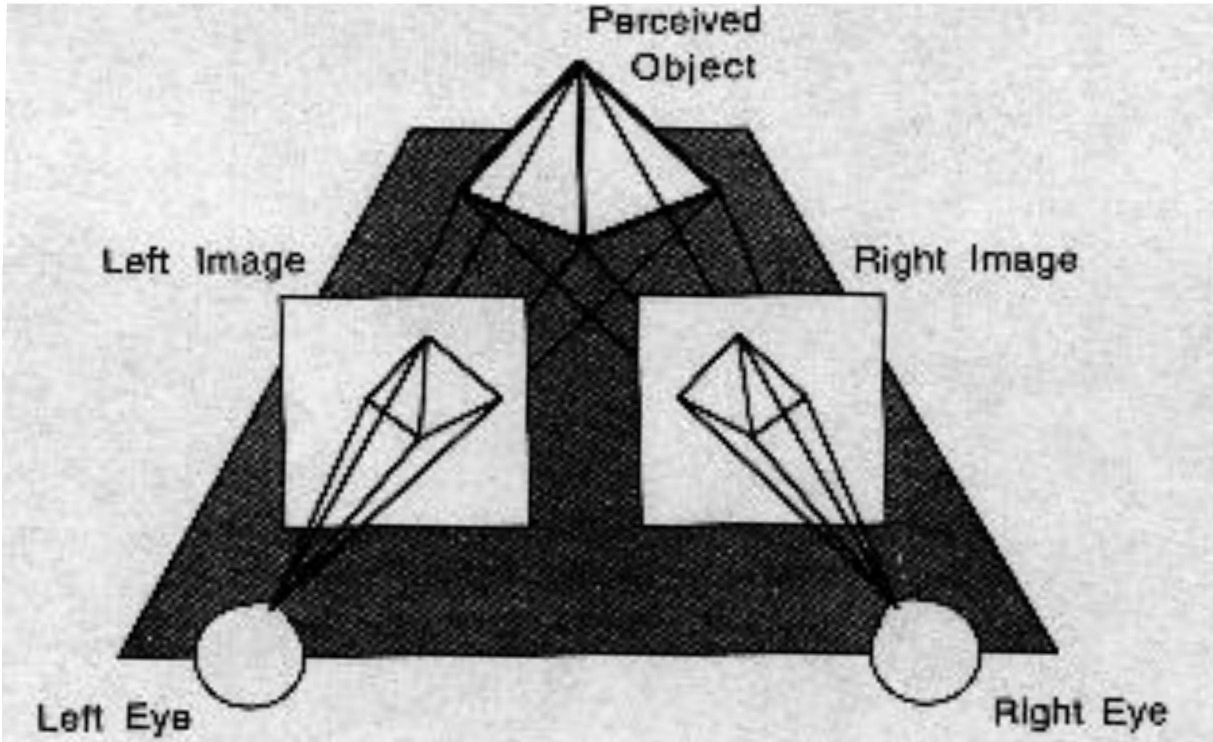
SHAPE FROM X

- One image:
  - Texture
  - Shading
- Two images or more:
  - **Stereo**
  - Contours
  - Motion



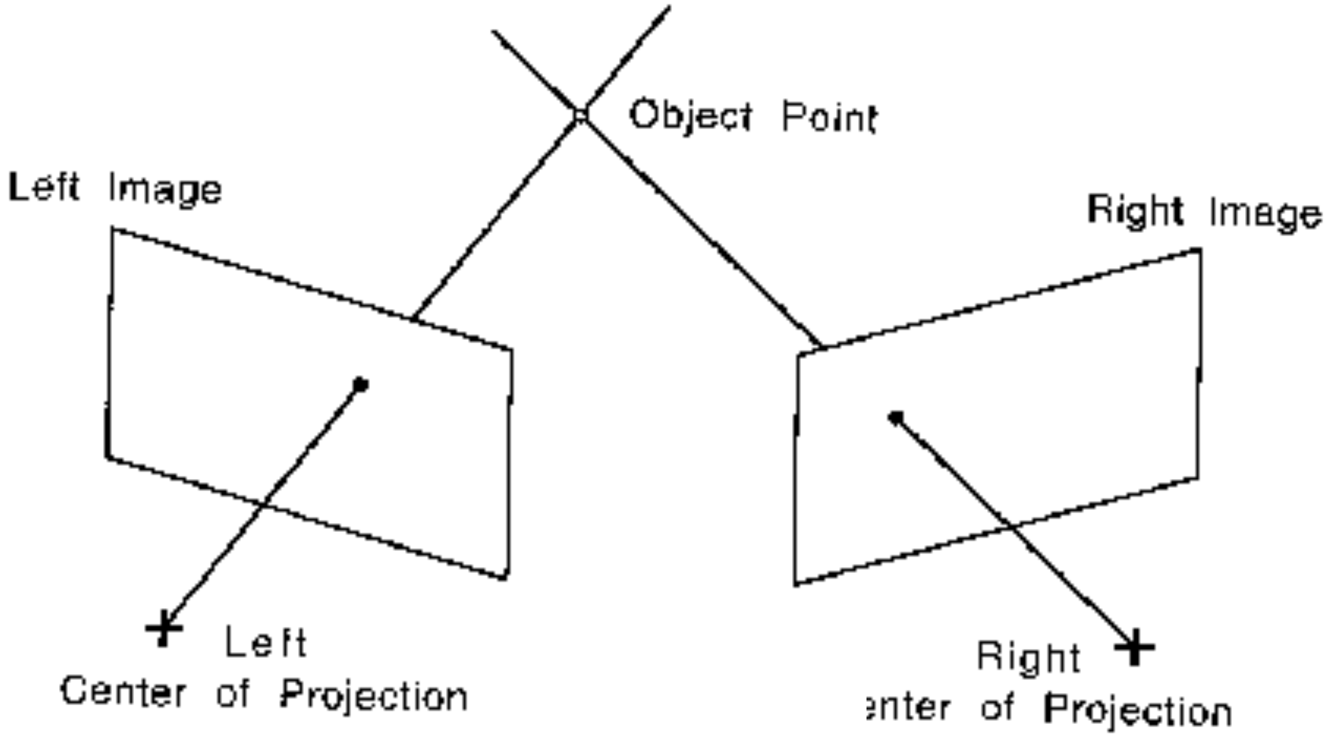
Geometric Stereo



Depth from two or more images:

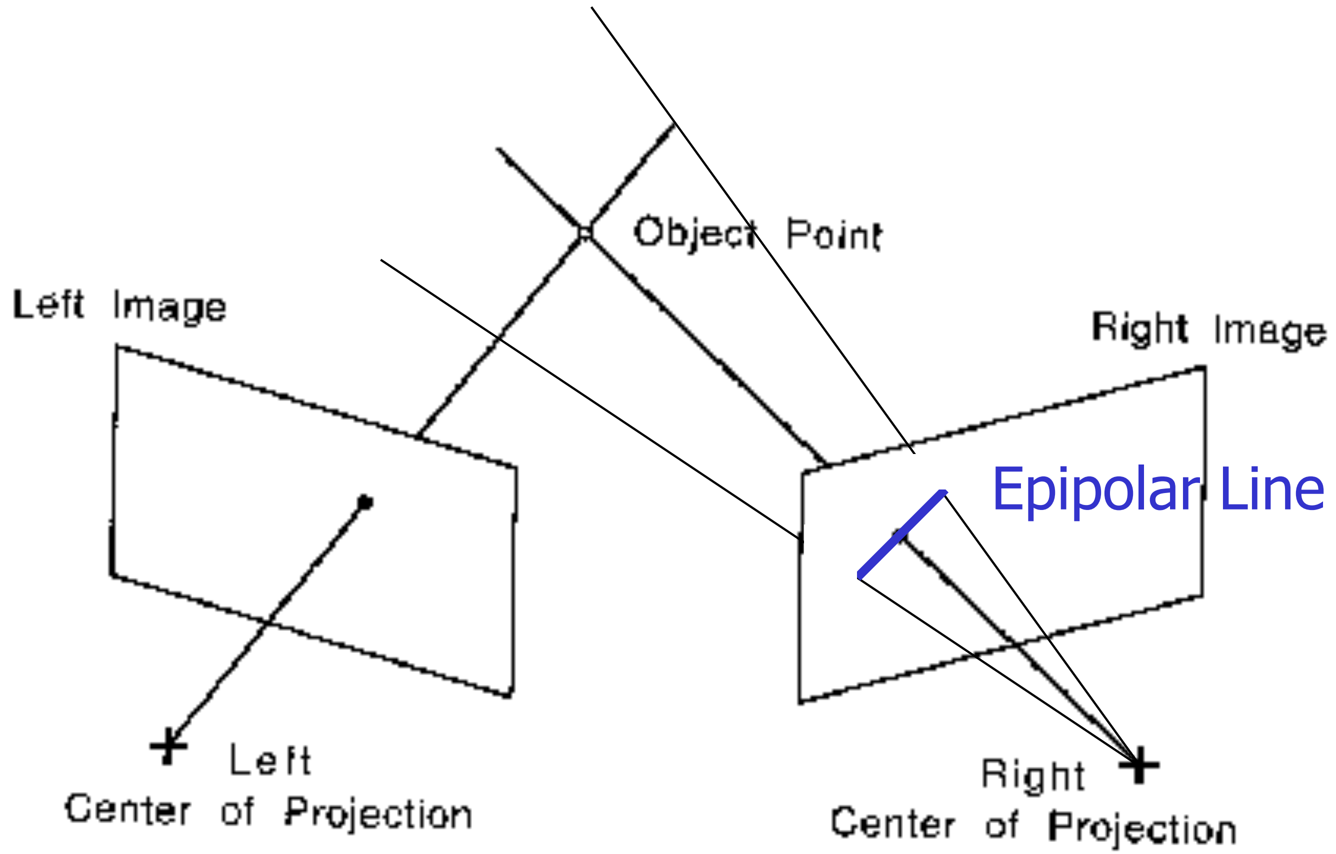
- Geometry of image pairs
- Establishing correspondences

Triangulation



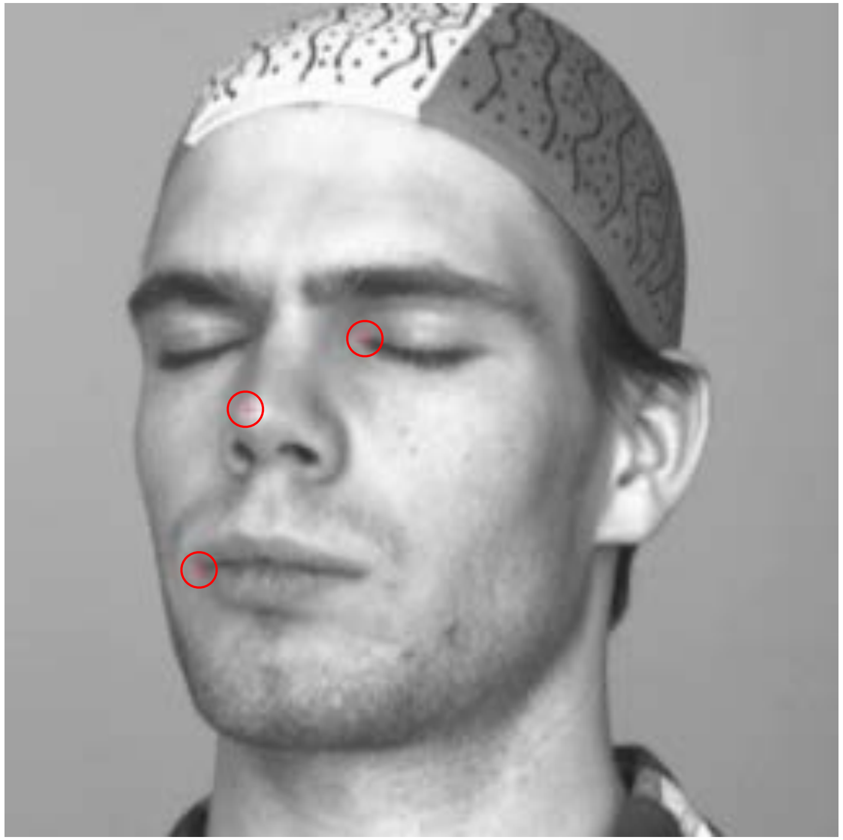
Geometric Stereo: Depth from two images

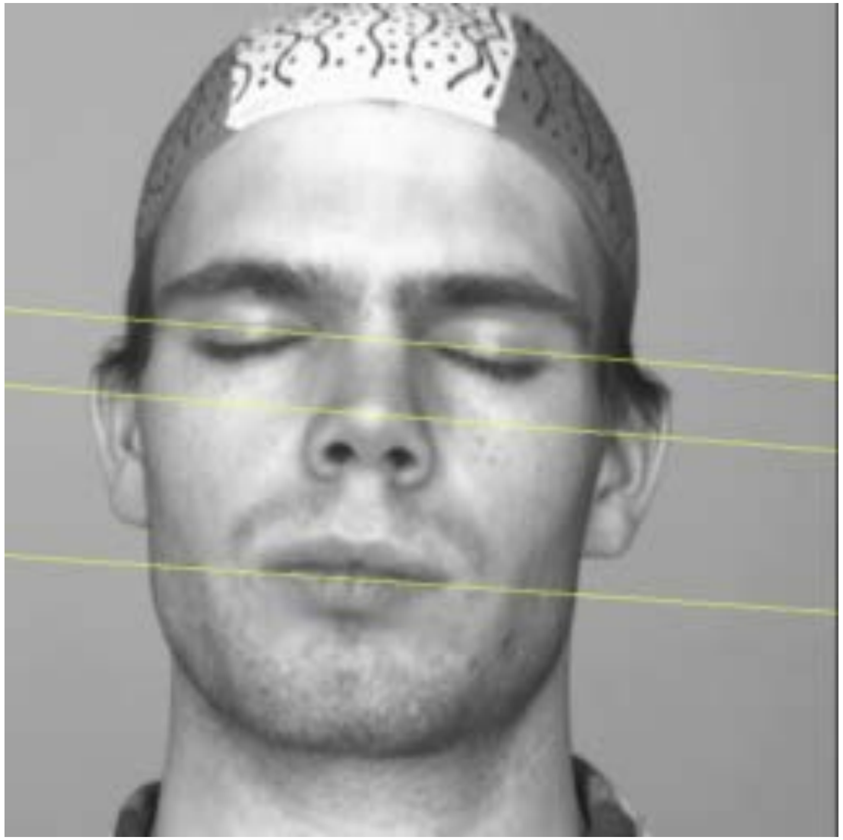
Epipolar Line



Line on which the corresponding point must lie.

Epipolar Lines

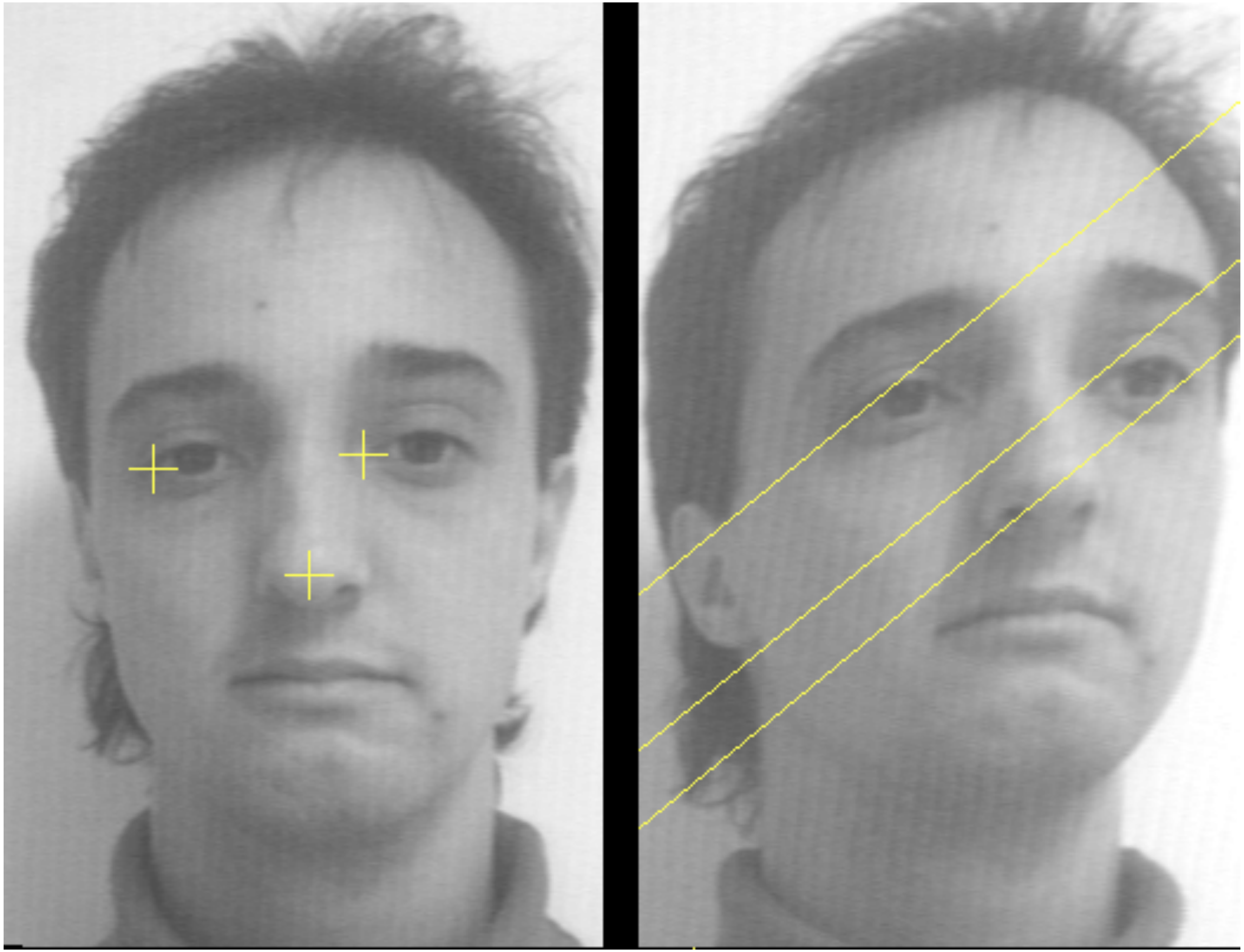




Three points shown as red crosses.

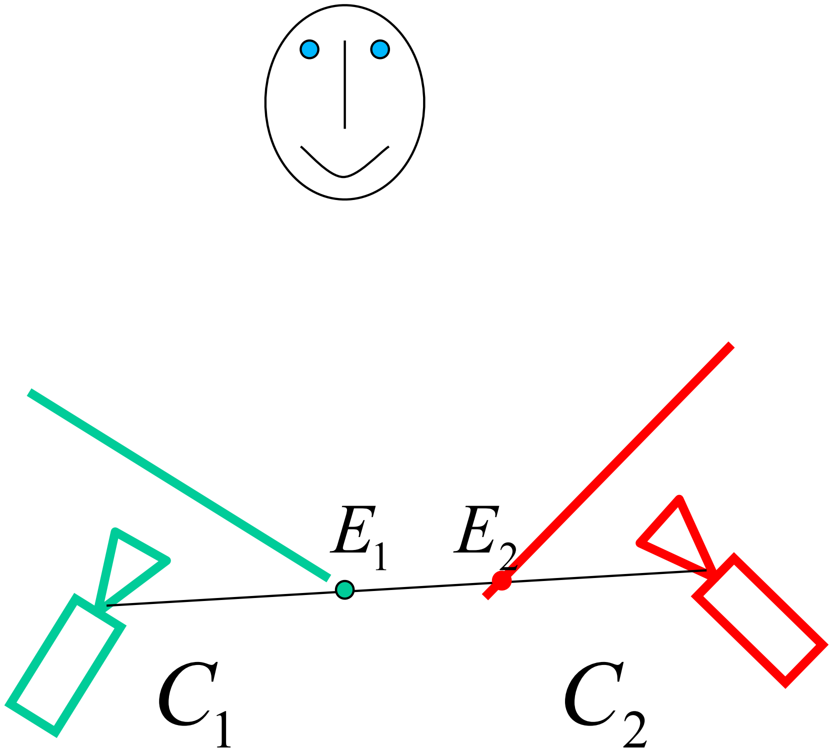
Corresponding epipolar lines.

Epipolar Lines



They can have any orientation.

Epipole

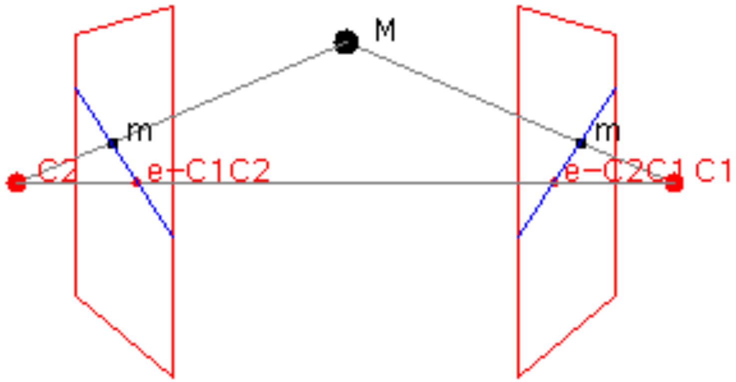


Point at which **all** epipolar lines intersect:

➡ Located at the intersection of line joining optical centers and image plane.

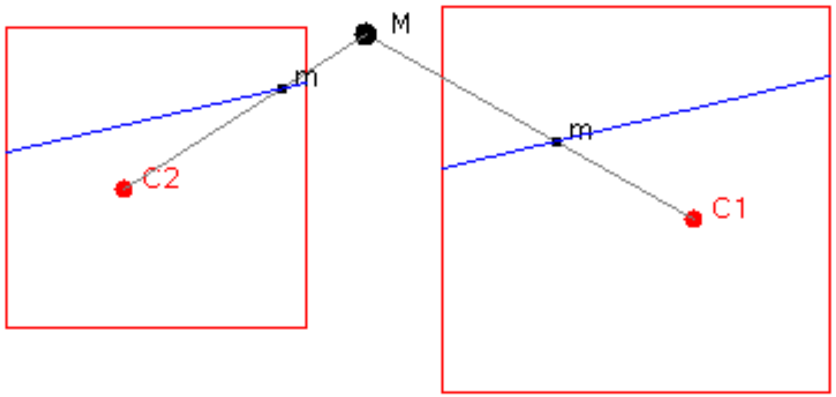
Epipolar Geometry

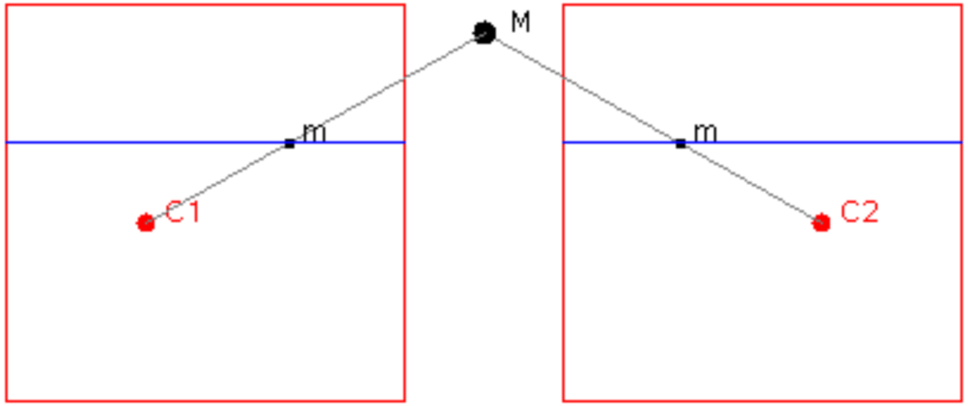
In general:



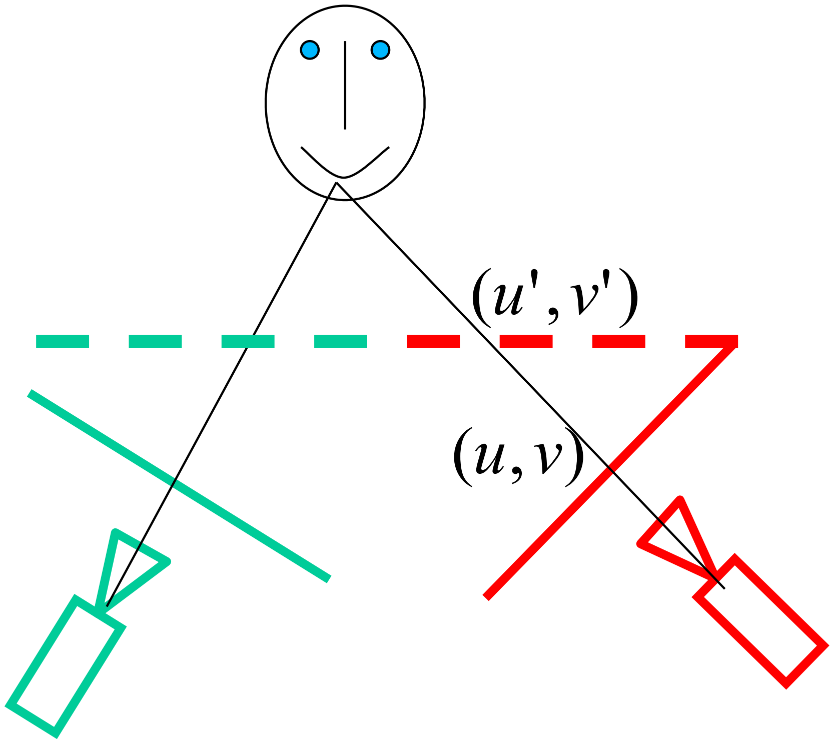
Parallel image planes

Horizontal baseline





Rectification



$$\begin{bmatrix} U' \\ V' \\ W' \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

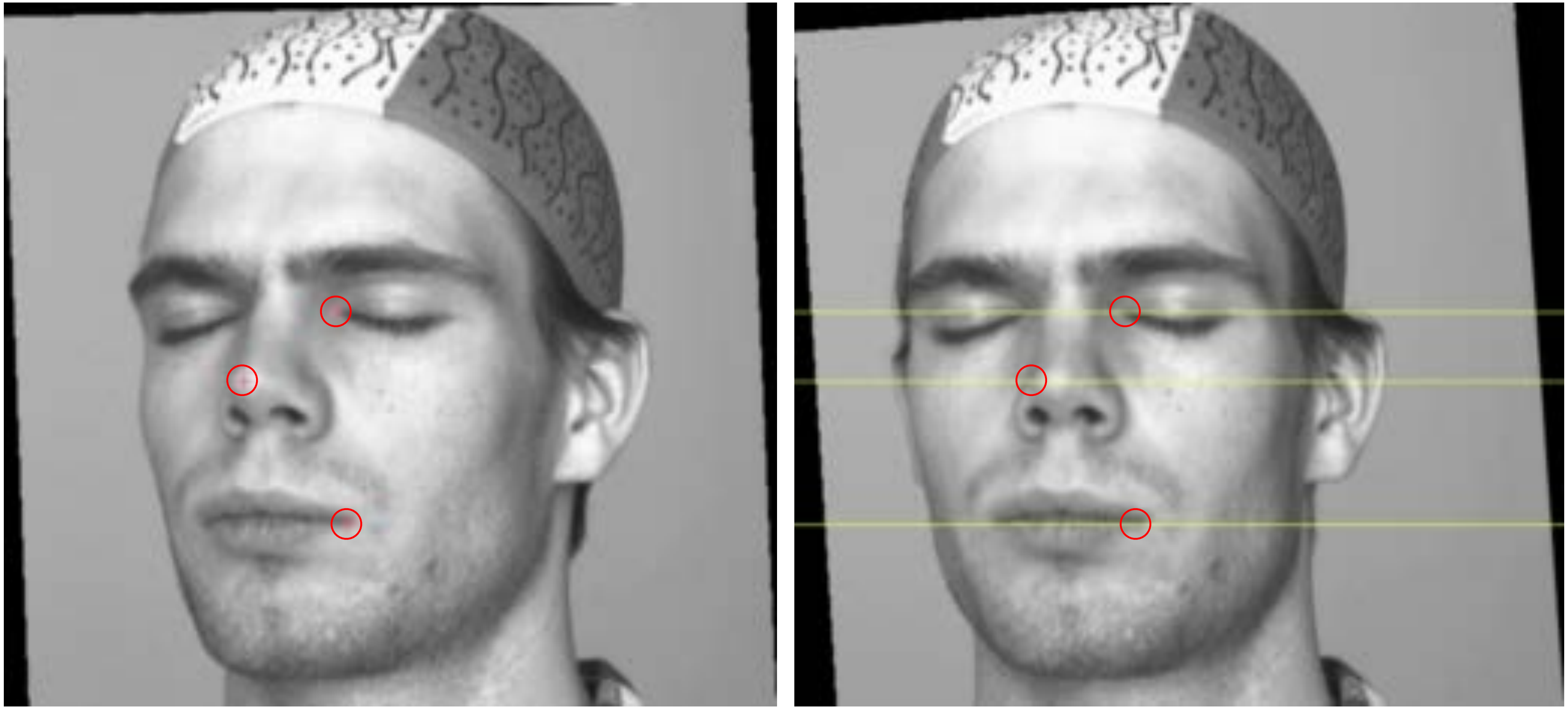
$$u' = U'/W'$$

$$v' = V'/W'$$

Reprojection into parallel virtual image planes:

- Linear operation in projective coordinates
- Real-time implementation possible

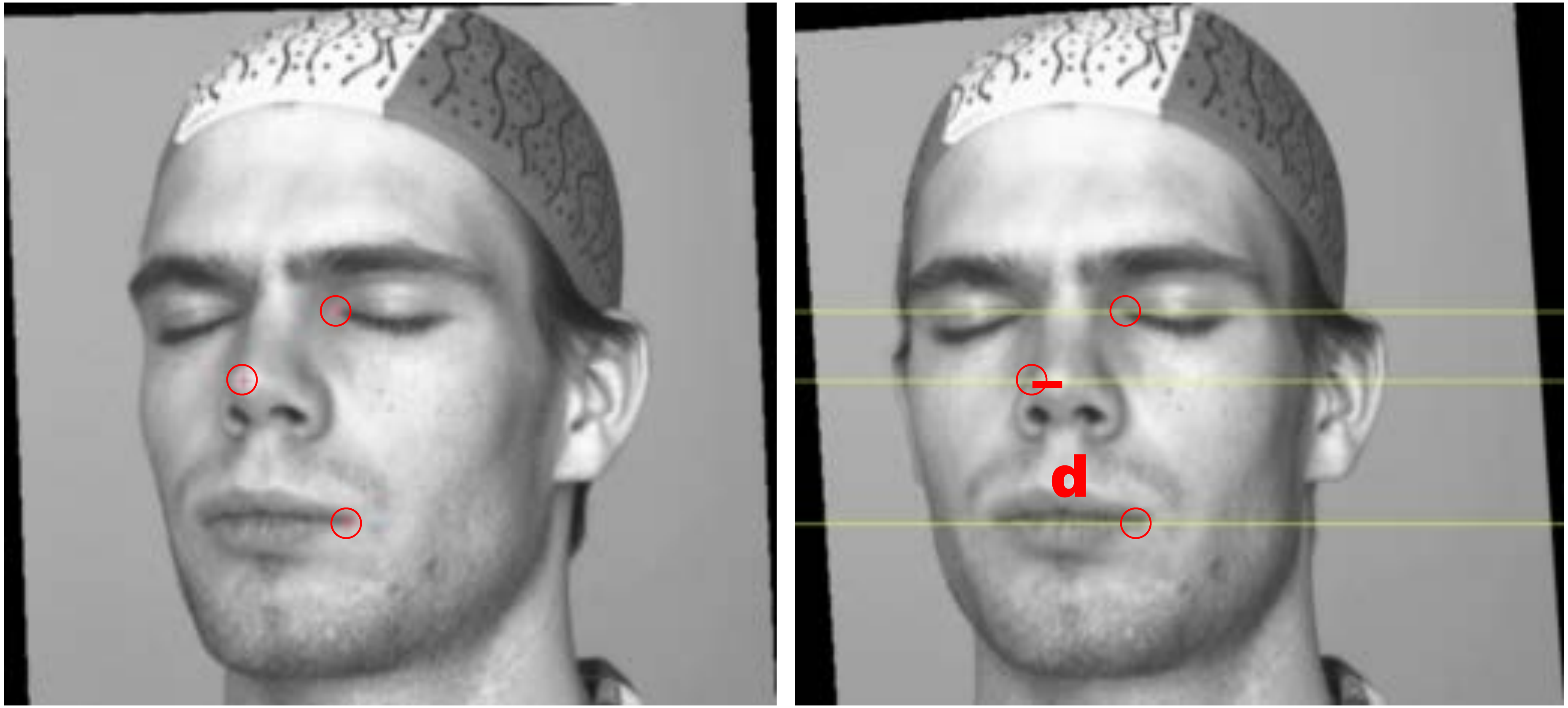
Rectification



From intersecting epipolar lines …

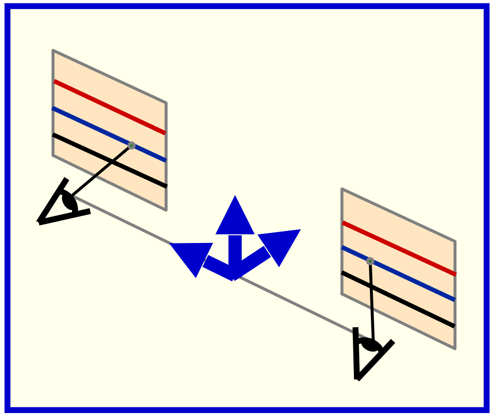
… to parallel ones.

Disparity



The horizontal shift along an epipolar line, inversely proportional to distance.

Disparity vs Depth



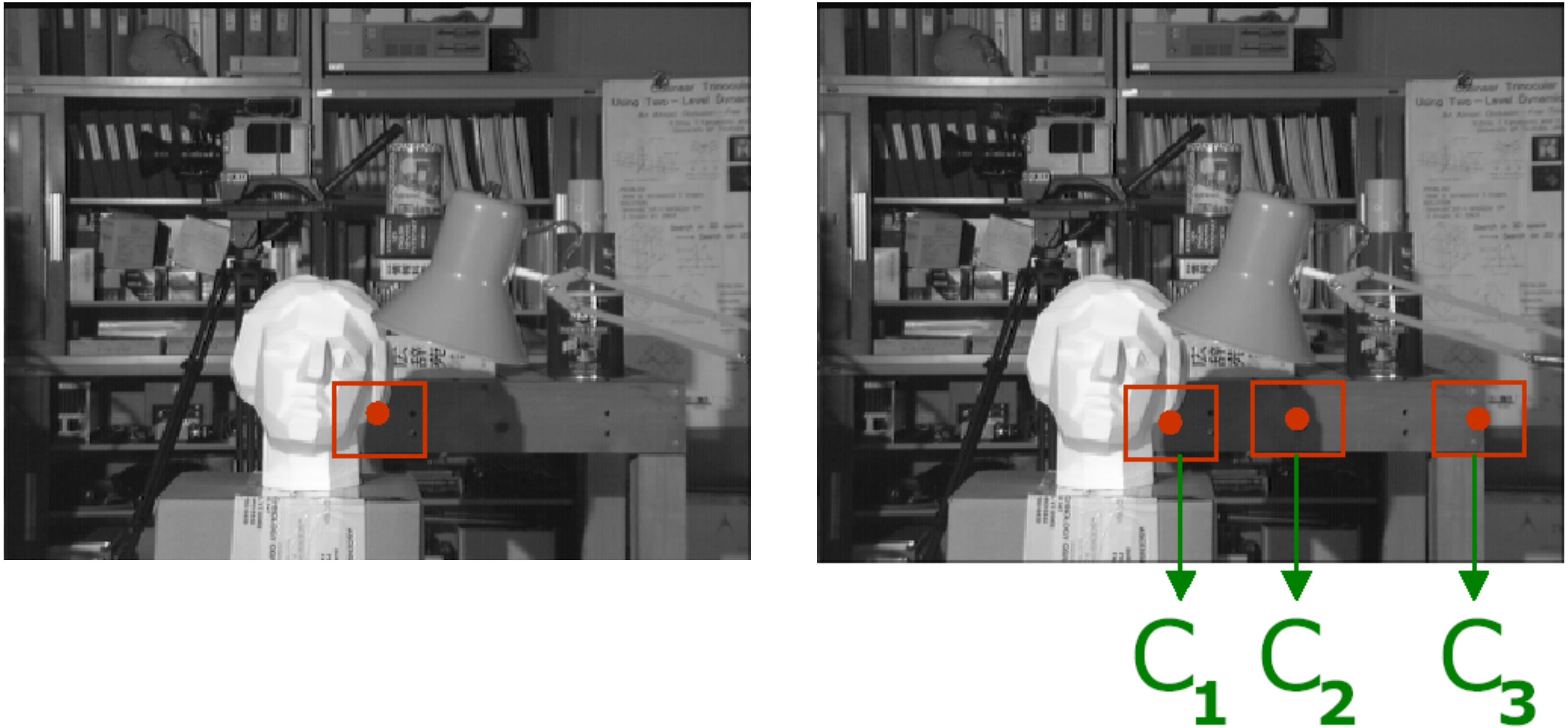
$$u_l = \frac{f(X - b/2)}{Z}, \; v_l = \frac{fY}{Z}$$

$$u_r = \frac{f(X + b/2)}{Z}, \; v_l = \frac{fY}{Z}$$

$$d = f\frac{b}{Z}$$

→ Disparity is inversely proportional to depth.

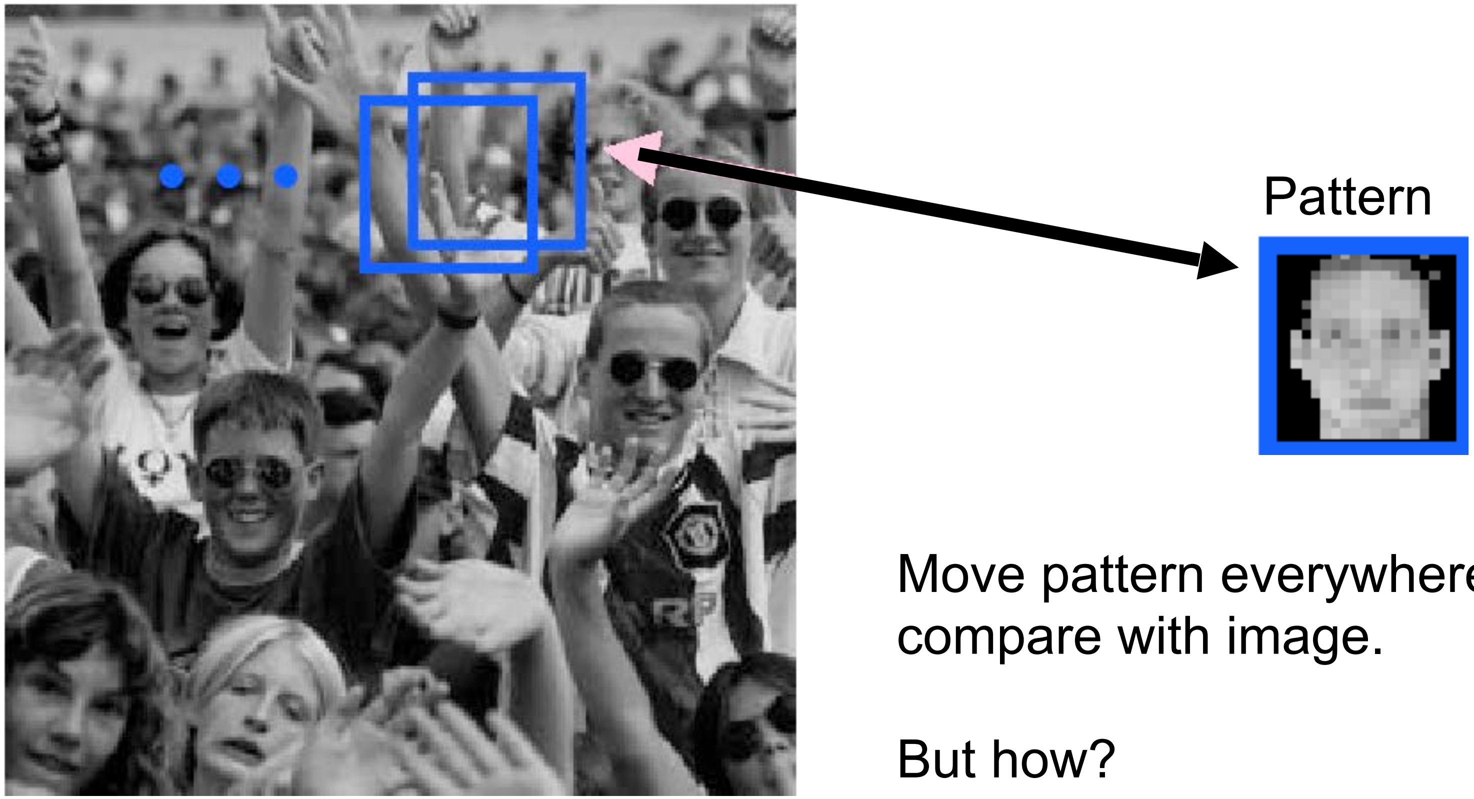
Window Based Approach to Establishing Correspondences



- Compute a cost for each C_n location.
- Pick the lowest cost one.

Finding a Pattern in an Image

Straightforward approach:



Move pattern everywhere and compare with image.

But how?

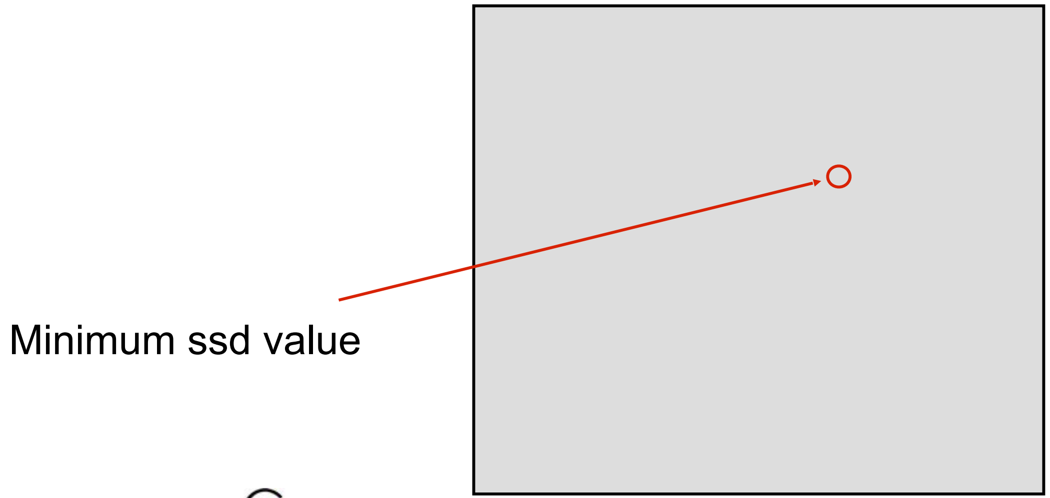
Sum of Square Differences

- Subtract pattern and image pixel by pixel and add squares:

$$ssd(u,v) = \sum_{(x,y)\in N} \left[I(u+x, v+y) - P(x,y) \right]^2$$

- If identical ssd=0, otherwise ssd >0

→Look for minimum of ssd with respect to u and v.



Correlation

$$ssd(u,v) = \sum_{(x,y)\in N} \left[I(u+x,v+y) - P(x,y) \right]^2$$

$$= \sum_{(x,y)\in N} \cancel{I(u+x,v+y)^2} + \sum_{(x,y)\in N} \cancel{P(x,y)^2} - 2 \sum_{(x,y)\in N} I(u+x,v+y)P(x,y)$$

Sum of squares of
the window
(slow varying)

Sum of squares of
the pattern
(constant)

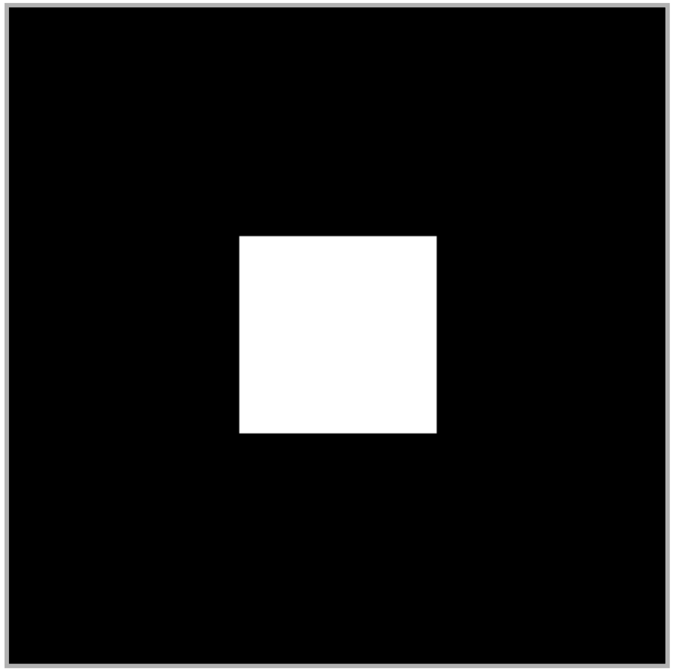
Correlation

ssd(u,v) is smallest when correlation is largest

→ Correlation measures similarity

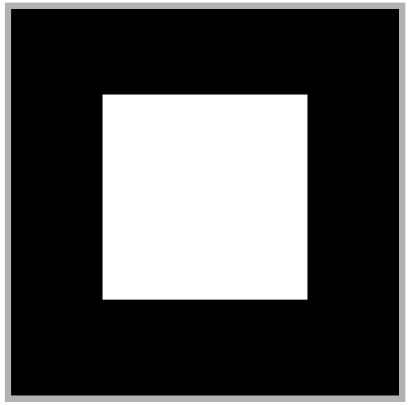
Synthetic Example

I



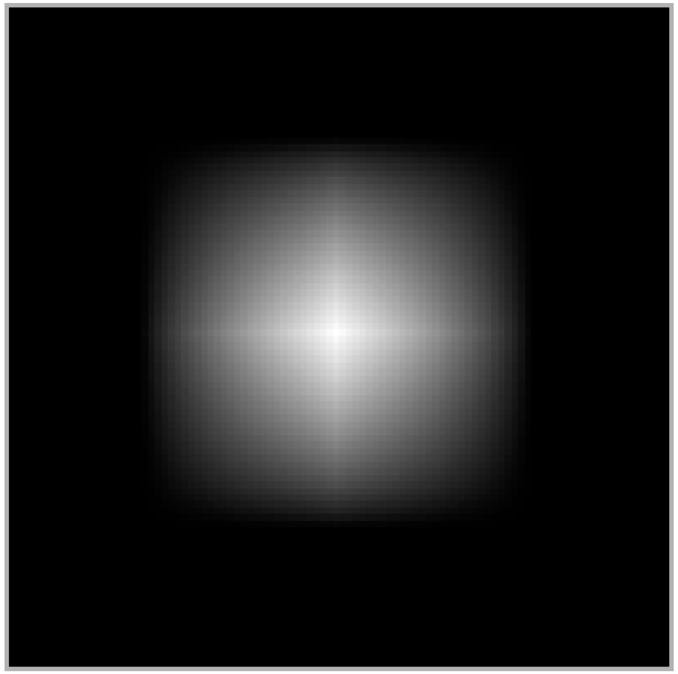
$*$

P



$=$

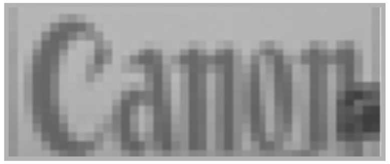
I correlated with P



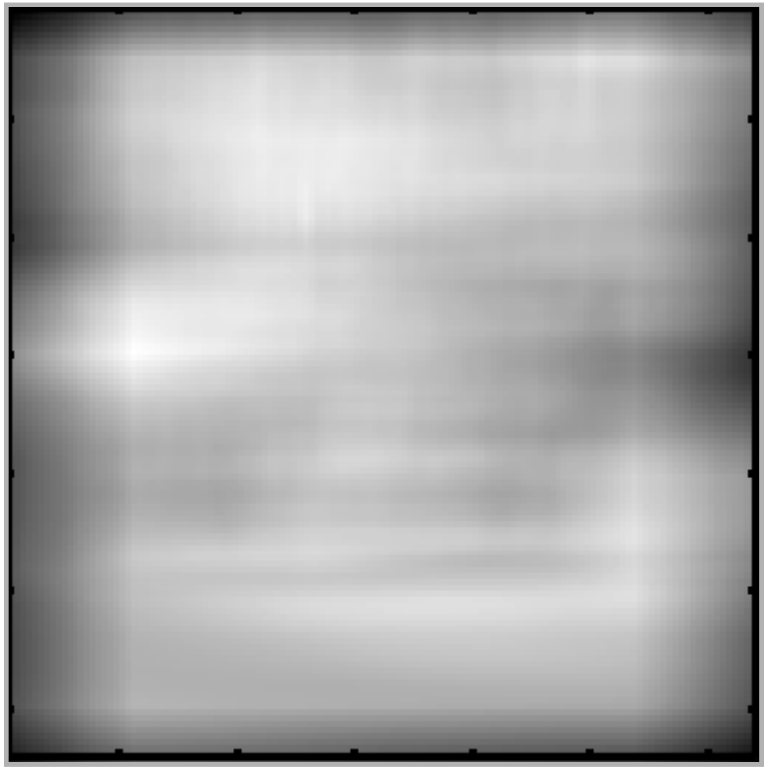
Rea World Example

Image

Pattern



Correlation



- The correlation value depends on the local gray levels of the pattern and image window.
- Need to normalize.

Normalized Cross Correlation

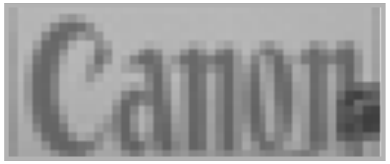
$$ncc(u,v) = \frac{\sum_{(x,y)\in N} \left[I(u+x,v+y) - \bar{I}\right]\left[P(x,y) - \bar{P}\right]}{\sqrt{\sum_{(x,y)\in N} \left[I(u+x,v+y) - \bar{I}\right]^2 \sum_{(x,y)\in N} \left[P(x,y) - \bar{P}\right]^2}}$$

- Between -1 and 1
- Invariant to linear transforms
- Independent of the average gray levels of the pattern and the image window

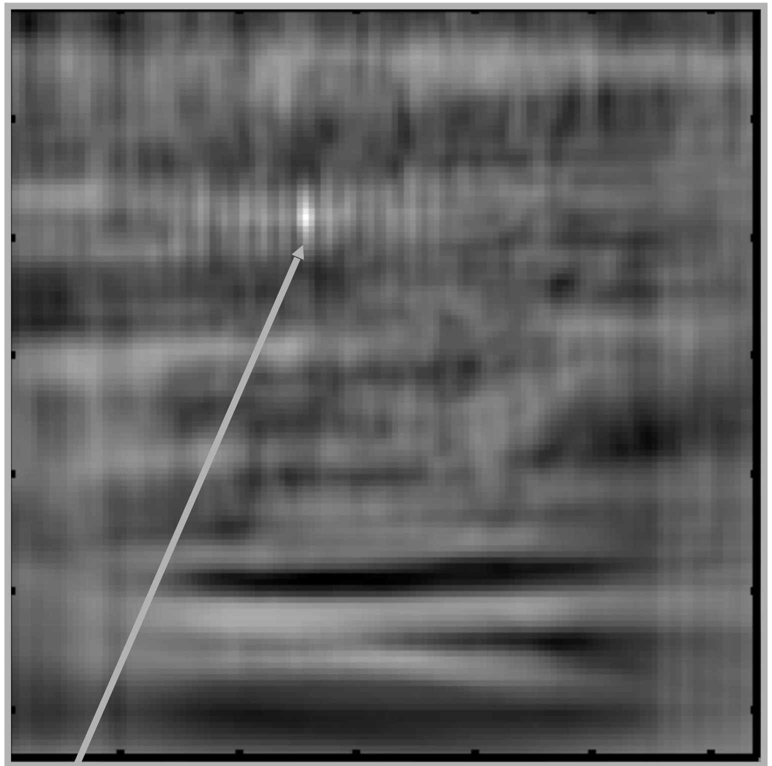
Normalized Example

Image

Pattern

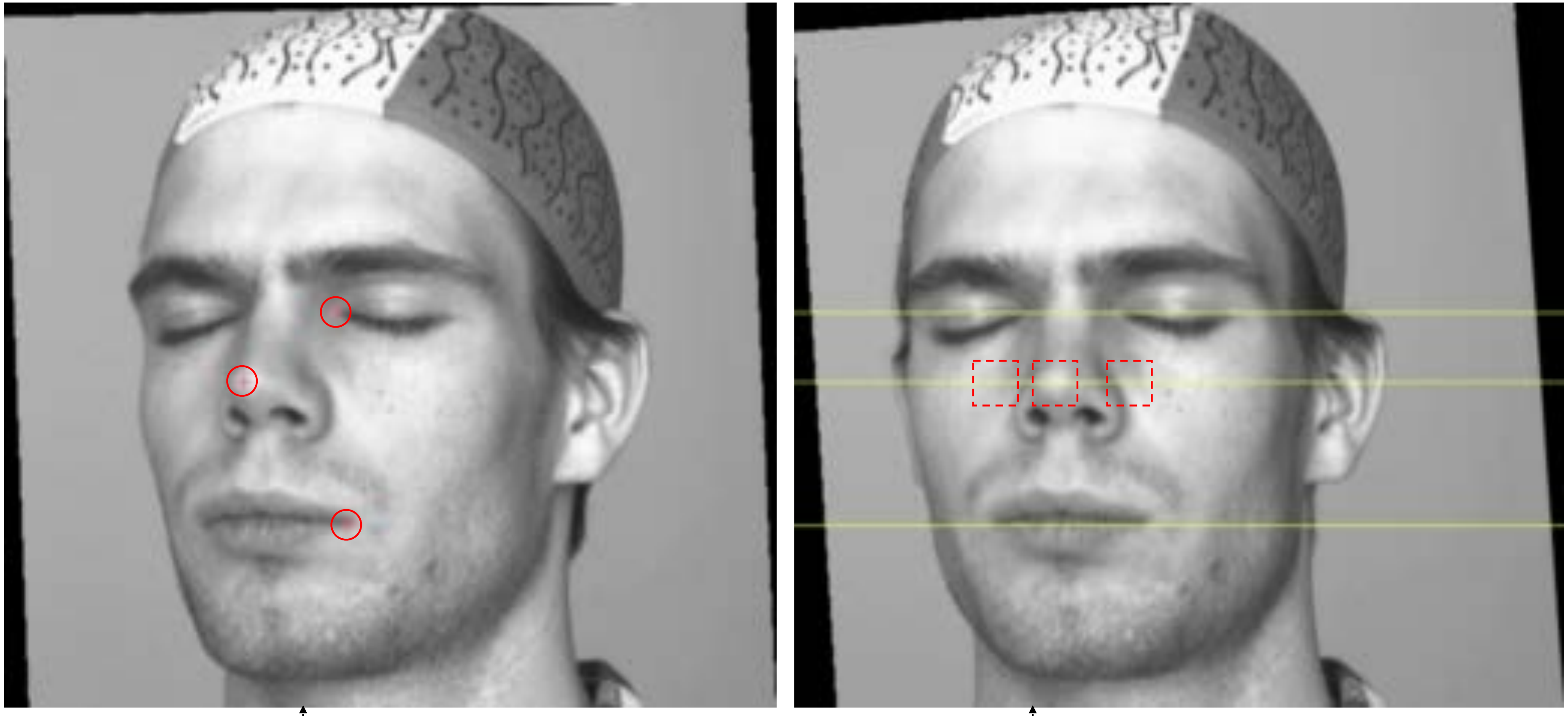


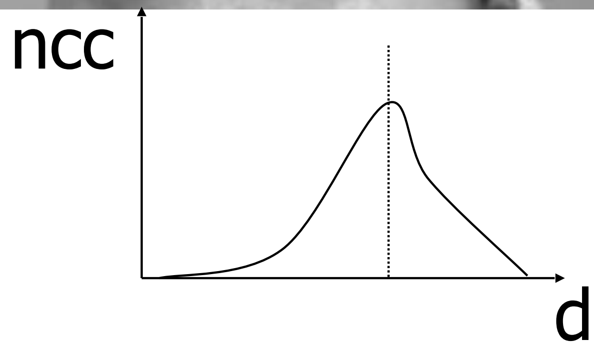
Normalized Correlation



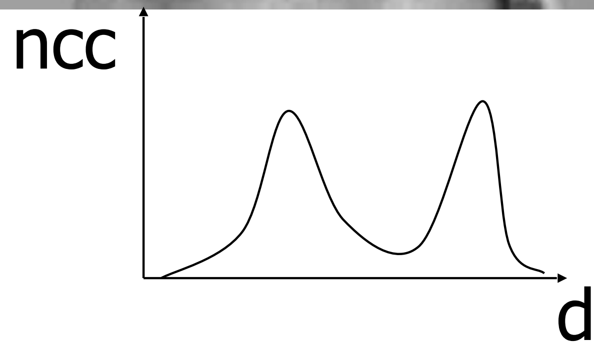
Point of maximum correlation

Searching along Epipolar Lines



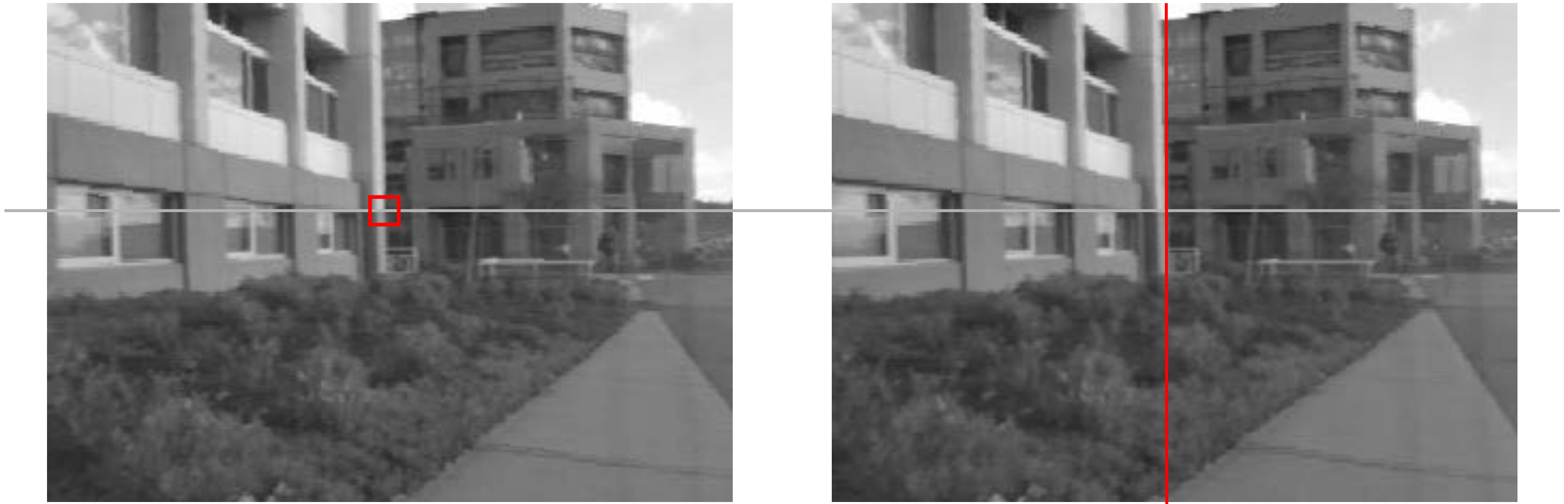


or



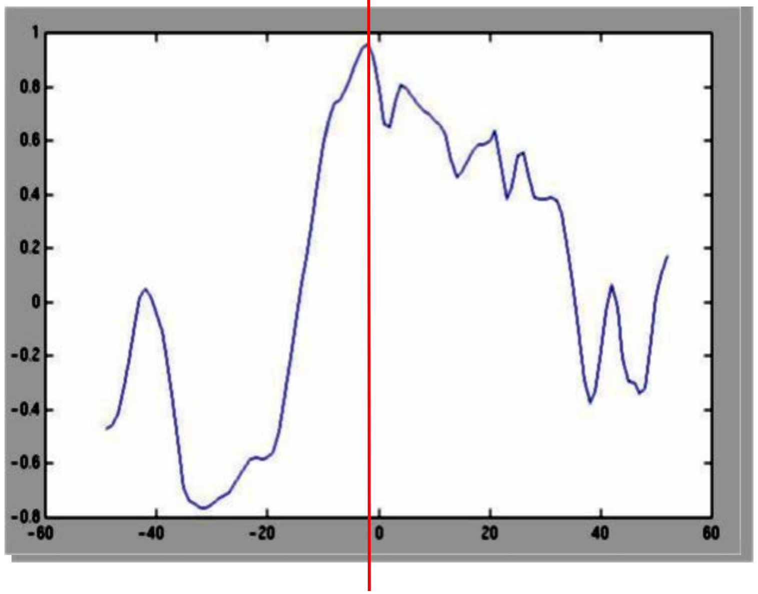
Outdoor Scene

scanline

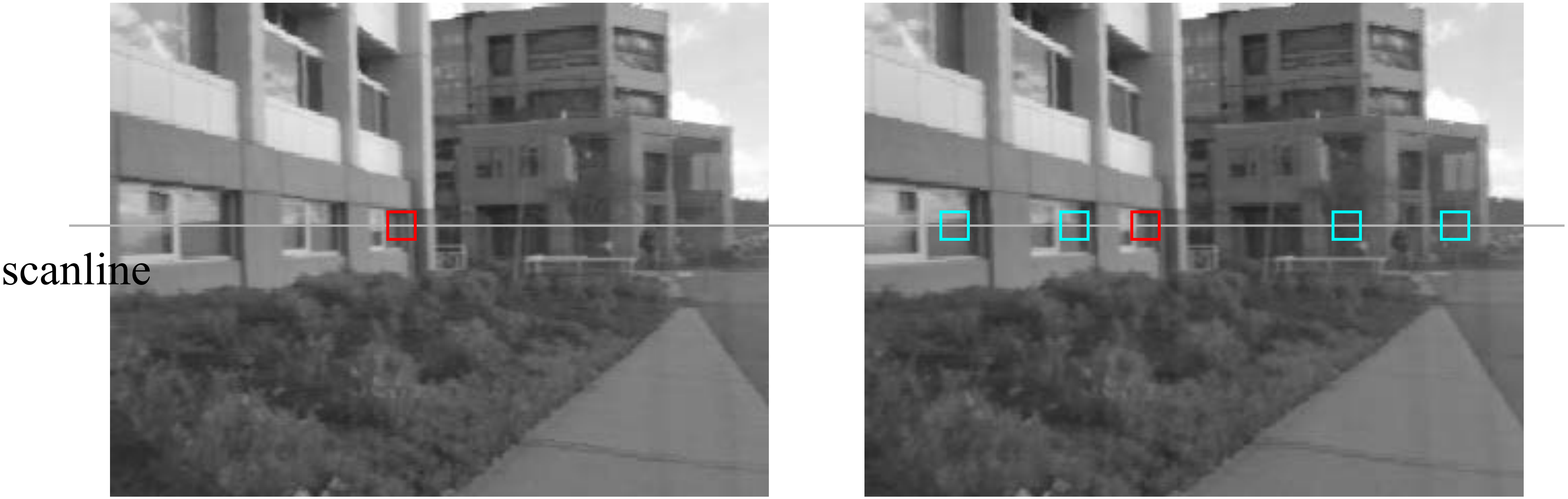


SSD

NCCR

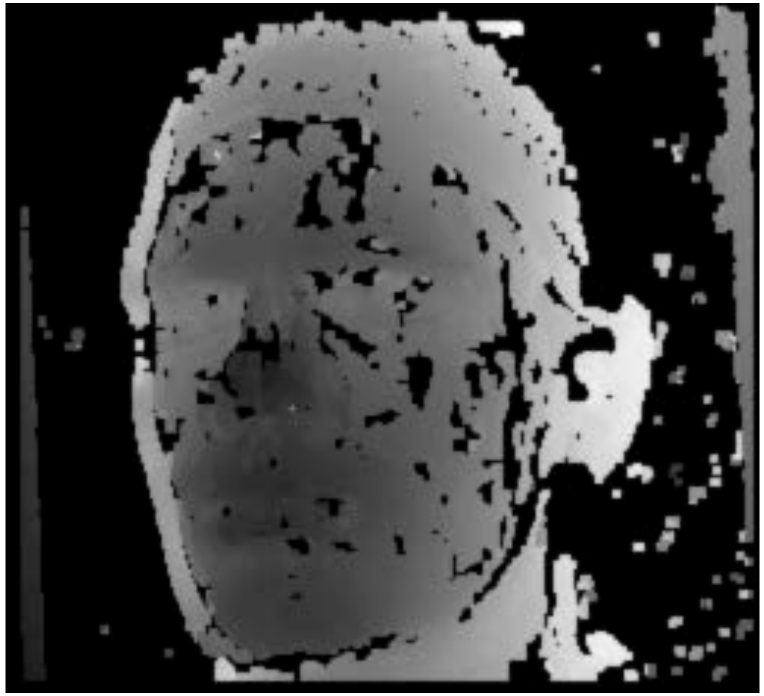


Ambiguities



—> Repetitive patterns, textureless areas, and occlusions can cause problems.

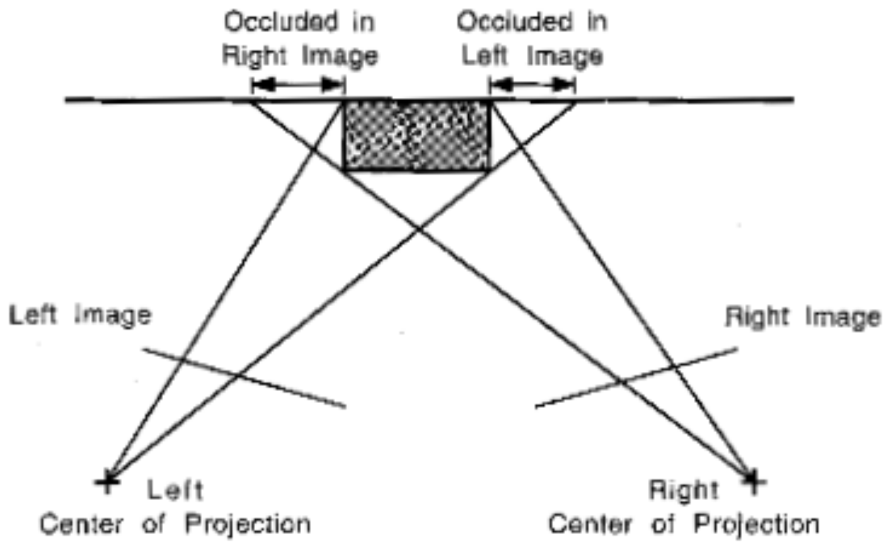
Disparity Map



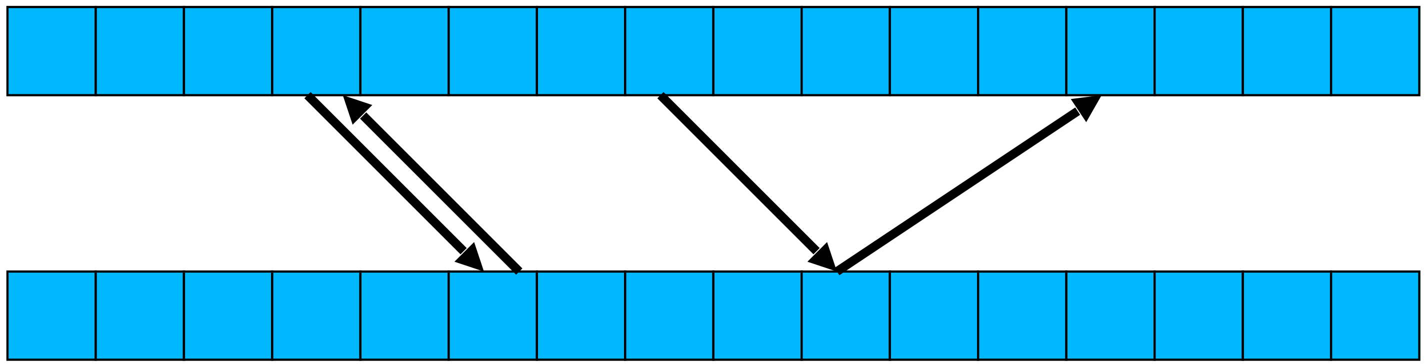
Black pixels: No disparity.

Occlusions

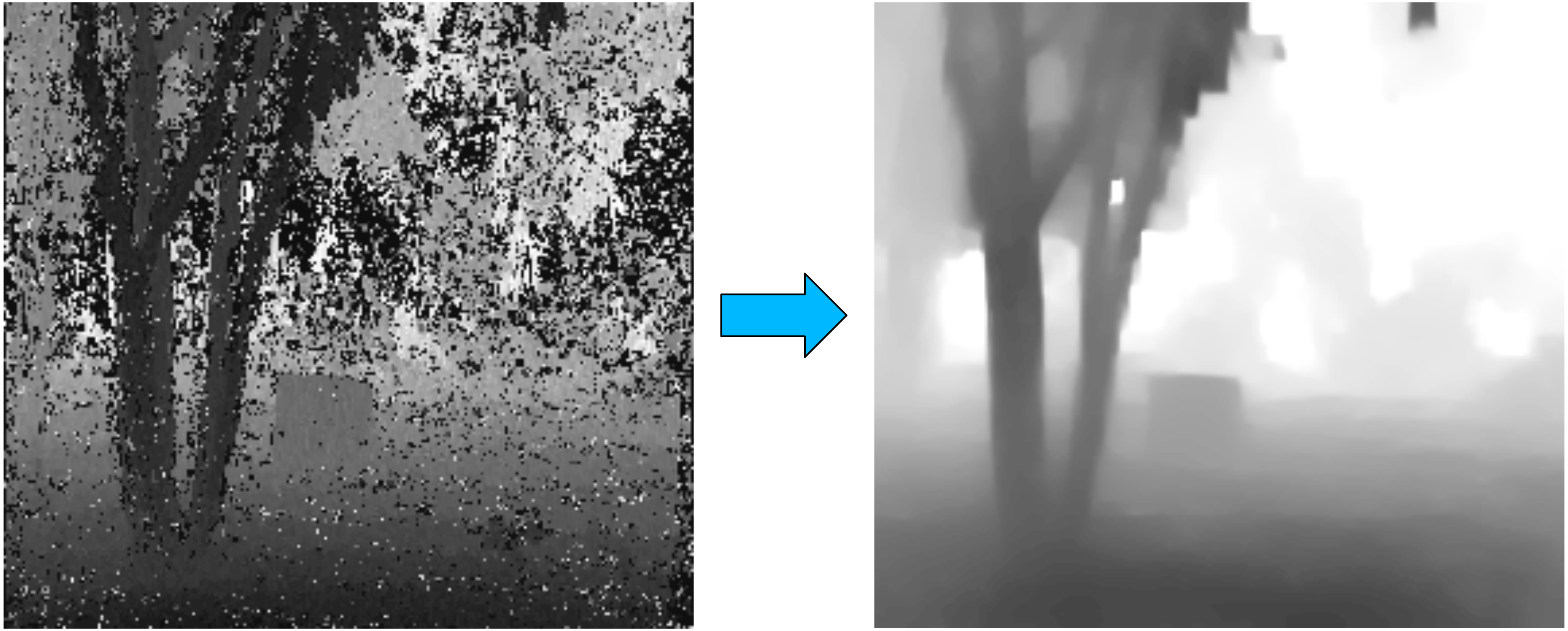
Some pixels have no corresponding pixel in the other image:



Left right consistency test:



Combining Disparity Maps



- Merging several disparity maps.
- Smoothing the resulting map.

Fua, MVA'91

Variational Approach

$$\mathcal{C} = \int s(w - w_0)^2 + \lambda_x\left(\frac{\partial w}{\partial x}\right)^2 + \lambda_y\left(\frac{\partial w}{\partial y}\right)^2$$

$$
\begin{aligned}
s &= \text{Correlation score if } w_0 \text{ has been measured}, 0 \text{ otherwise.} \\
\lambda_x &= c_x f\left(\frac{\partial I}{\partial x}\right) \\
\lambda_y &= c_y f\left(\frac{\partial I}{\partial y}\right)
\end{aligned}
$$

$$f(x) = \begin{cases} 1 & \text{if} \quad x < x_0 \\ \frac{x_1 - x}{x_1 - x_0} & \text{if} \quad x_0 < x < x_1 \\ 0 & \text{if} \quad x_1 < x \end{cases}$$

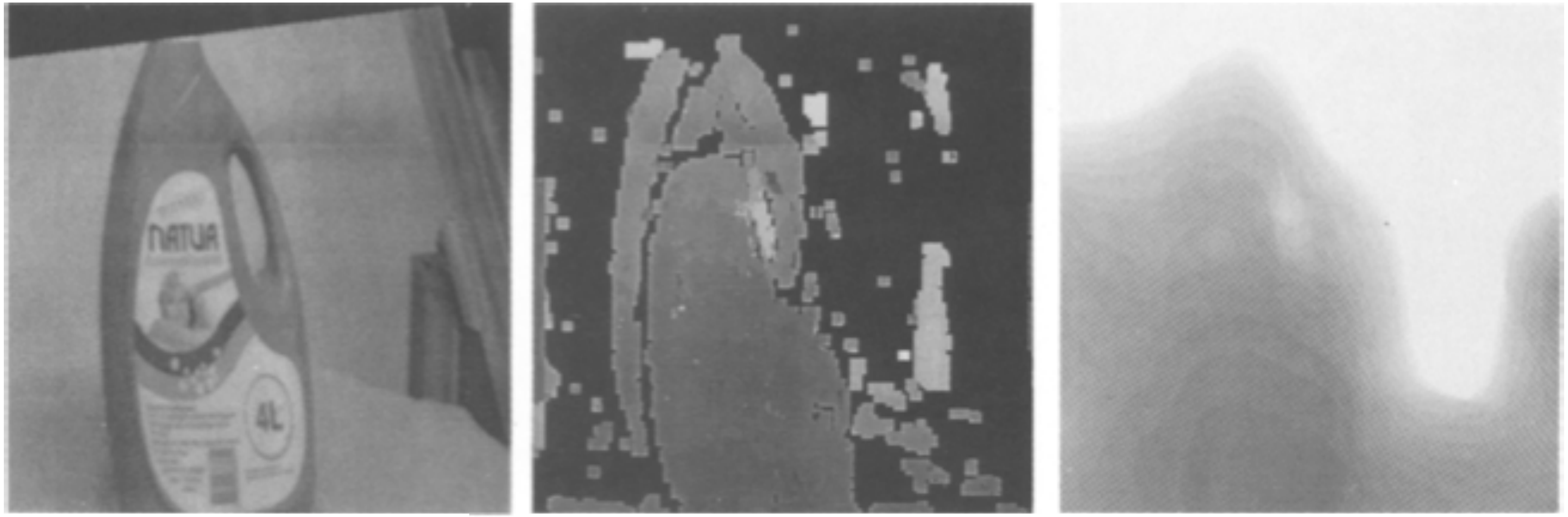
Solving the Variational Problem

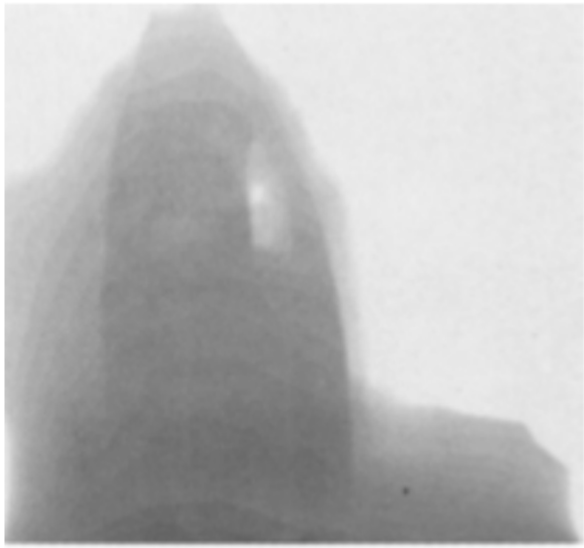
Discretize the integral and solve a linear problem:

$$\begin{aligned}
\mathcal{C} &= \sum_{ij} s_{ij}(w_{ij} - w_{0ij})^2 + \lambda_x \sum_{ij}(w_{i+1,j} - w_{i,j})^2 + \lambda_y \sum_{ij}(w_{i,j+1} - w_{i,j})^2 \\
&= (W - W_0)^t S(W - W_0) + W^t K W
\end{aligned}$$

$$\begin{aligned}
\Rightarrow \quad \frac{\partial \mathcal{C}}{\partial W} &= 0 \\
\Rightarrow \quad (K + S)W &= SW_0
\end{aligned}$$

Preserving Discontinuities

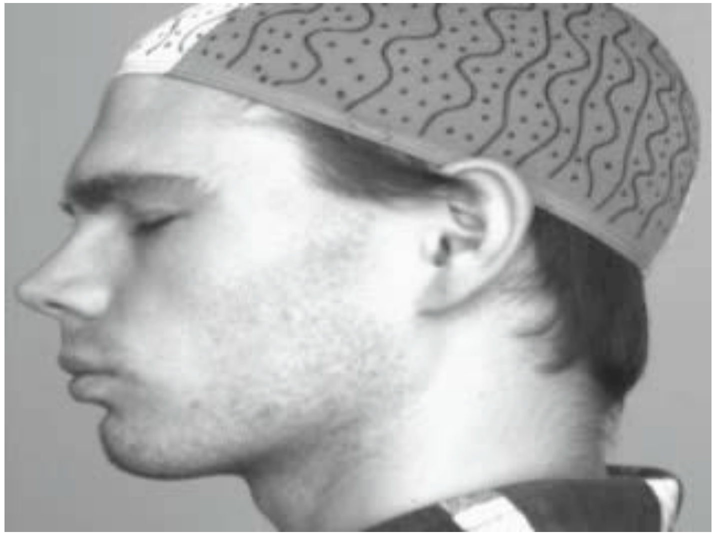


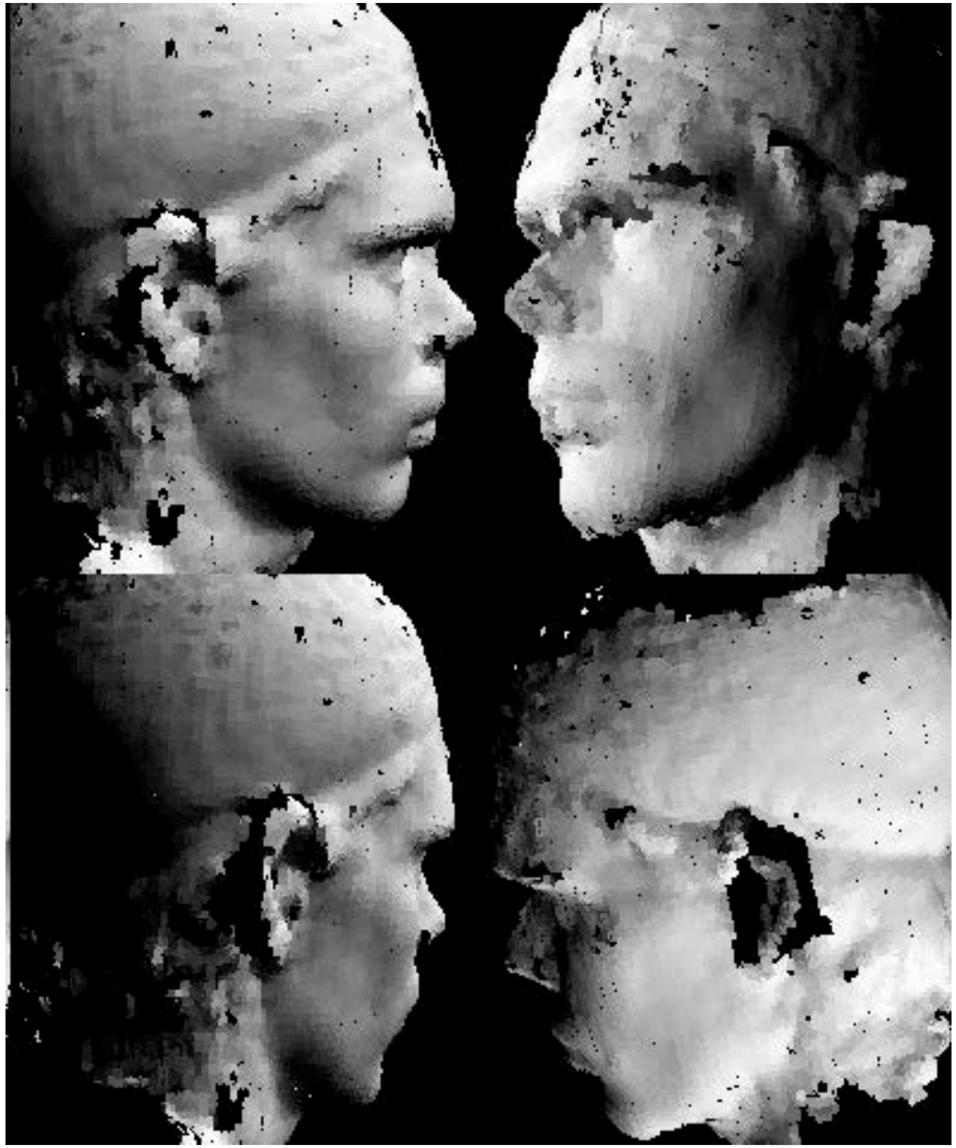


$$\lambda_x = f(\frac{\partial I}{\partial x}) f(\frac{\partial w}{\partial x})^2$$

$$\lambda_y = f(\frac{\partial I}{\partial y}) f(\frac{\partial w}{\partial y})^2$$

Shape From Video



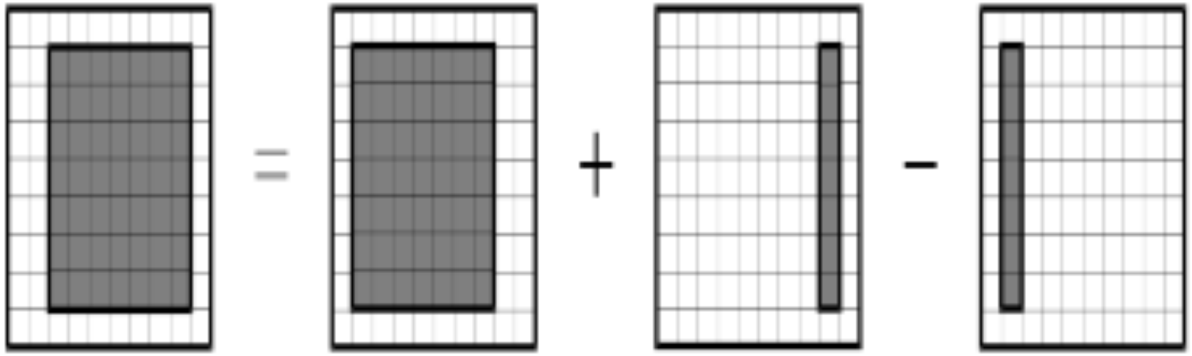


Treat consecutive images as stereo pairs.

1. Compute disparity maps.
2. Merge 3-D point clouds.
3. Represent as small patches.

Fua. IJCV'97

Real-Time Implementation



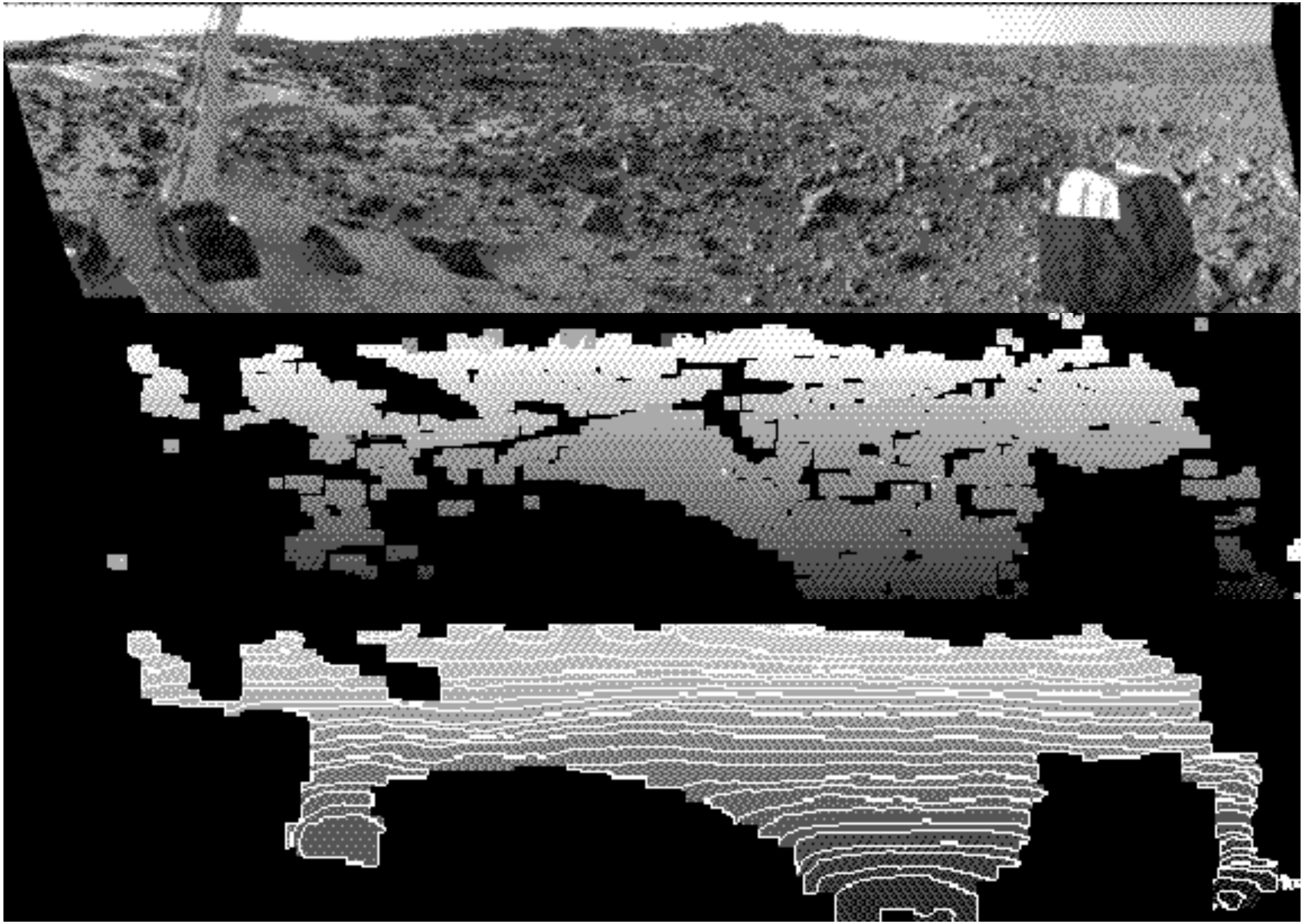
$$C(x,y,d) \propto \frac{\sum_{i,j} I_1(x+i,y+j) \times I_2(x+d+i,y+j)}{\sqrt{\sum_{i,j} I_2(x+d+i,y+j)^2}}$$

$$C(x+1,y,d) \propto \frac{\sum_{i,j} I_1(x+1+i,y+j) \times I_2(x+1+d+i,y+j)}{\sqrt{\sum_{i,j} I_2(x+1+d+i,y+j)^2}}$$

$$\propto \frac{\sum_{i',j} I_1(x+i',y+j) \times I_2(x+d+i',y+j)}{\sqrt{\sum_{i,j} I_2(x+d+i',y+j)^2}}$$

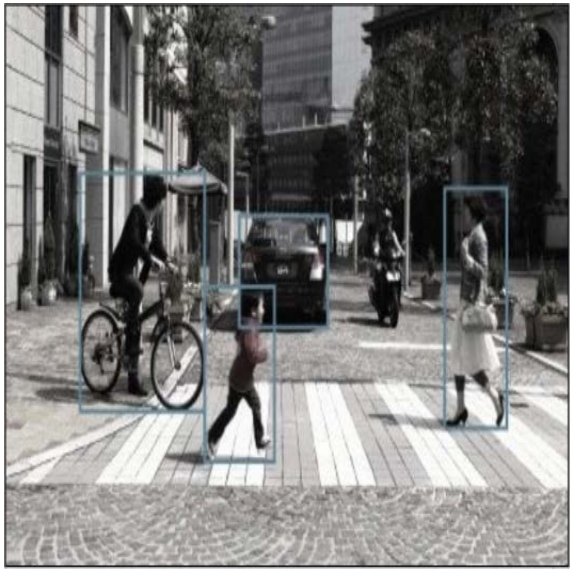
- Many duplicated computations.
- Can be implemented so that it is fast.
- Speed is independent from window size.

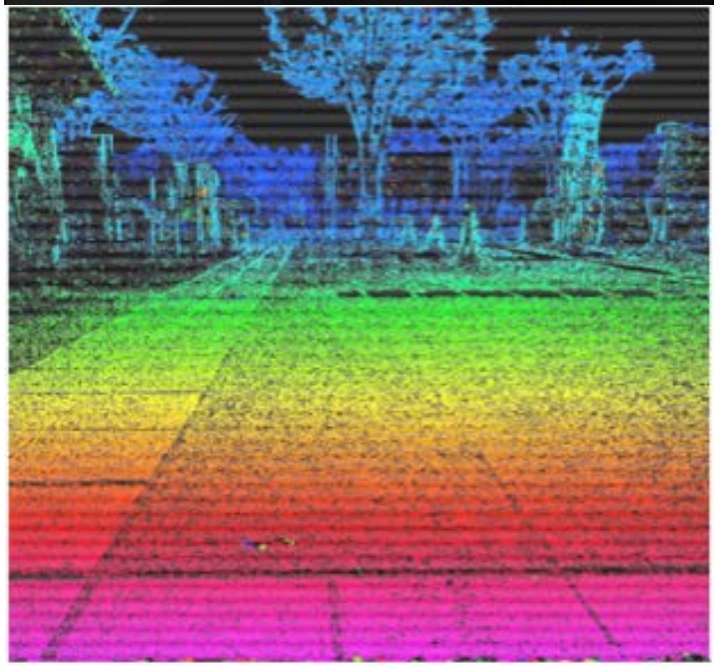
Then ....



1993:
256x256,
60 disps,
7 fps.

… And More Recently





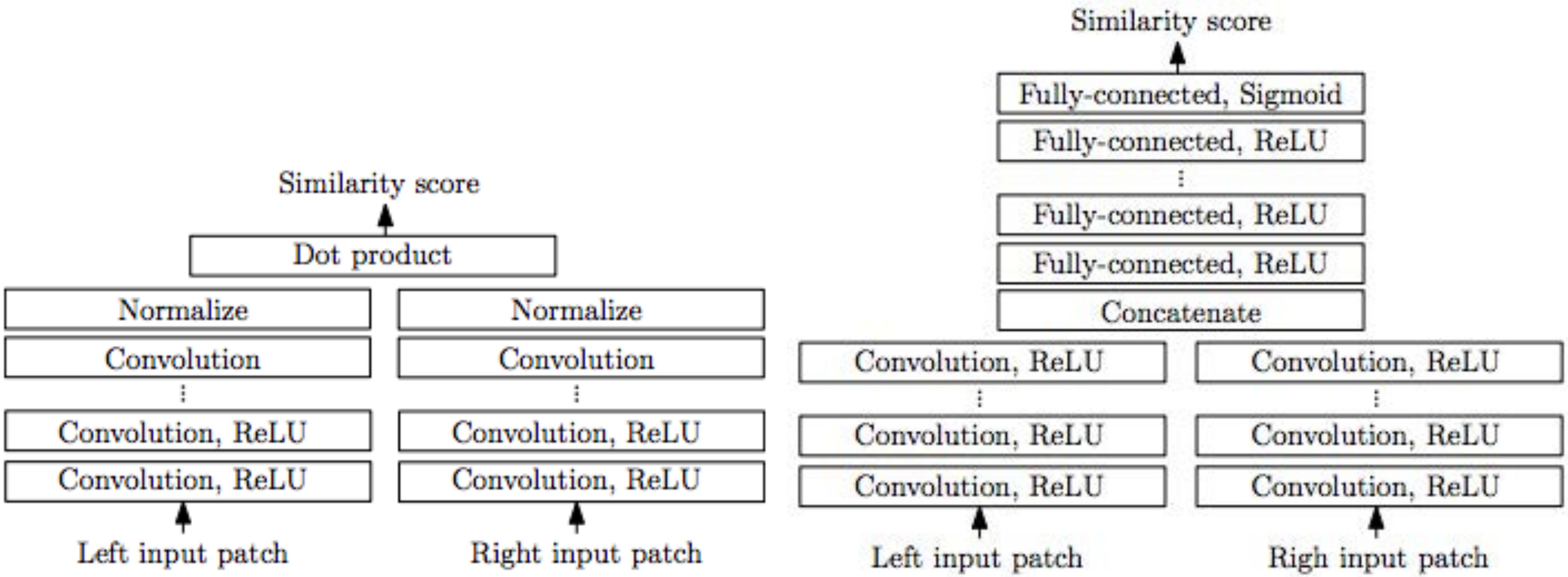
Subaru's EyeSight System

http://www.gizmag.com/subaru-new-eyesight-stereoscopic-vision-system/14879/

2011:
1312x688,
176 disps,
160 fps.

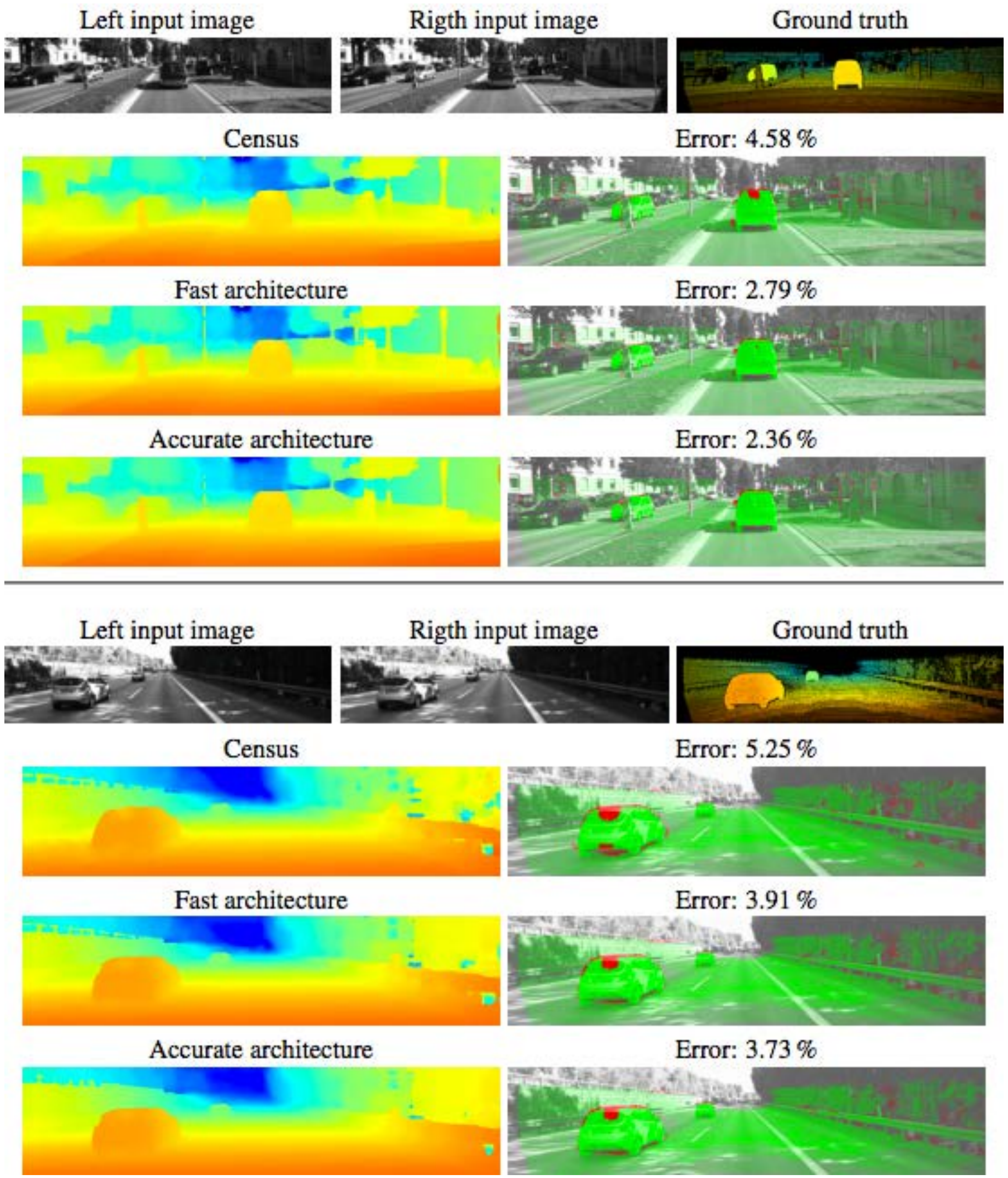
... and even More Recently

Replace Normalized Cross Correlation by Siamese nets designed to return a similarity score for potentially matching patches.



Zbontar and Lecun, 2015

Comparative Results



Improved performance on test data but

- How well will it generalize to unseen images?

- Is it worth the much heavier computational load?

Time will tell.

Tesla's non LiDar Approach

https://www.therobotreport.com/researchers-back-teslas-non-lidar-approach-to-self-driving-cars/

Window Size

Small windows:

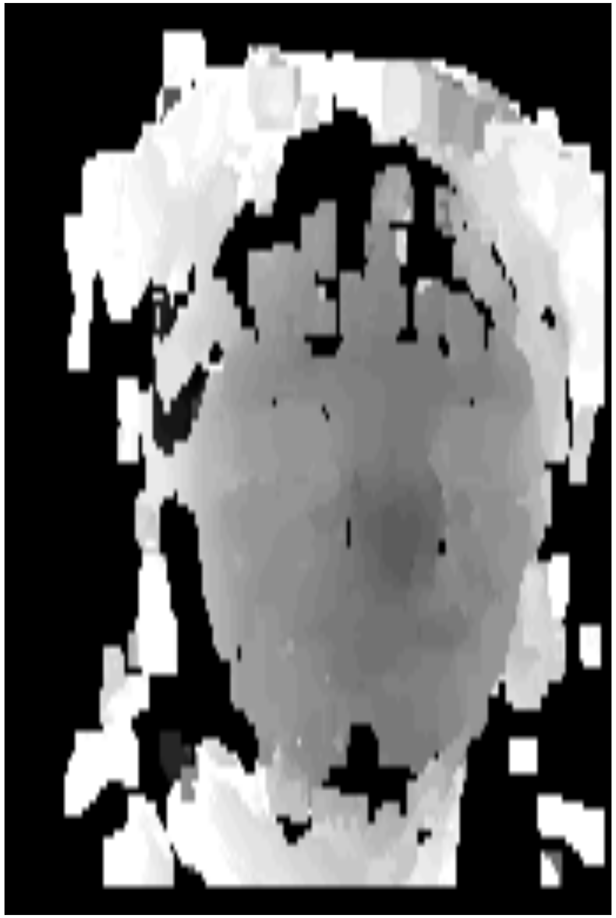
- Good precision
- Sensitive to noise

Large windows:

- Diminished precision
- Increased robustness to noise

→ Same kind of trade-off as for edge-detection.

Window Size

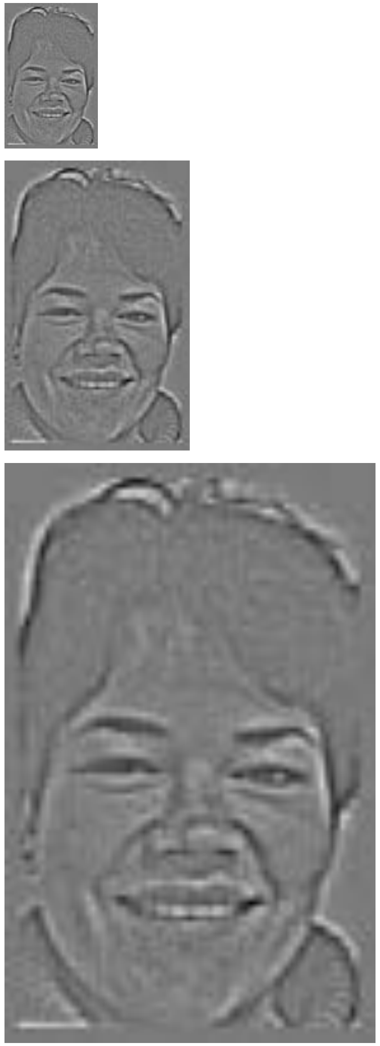


15x15

7x7

Scale-Space Revisited

Gaussian pyramid



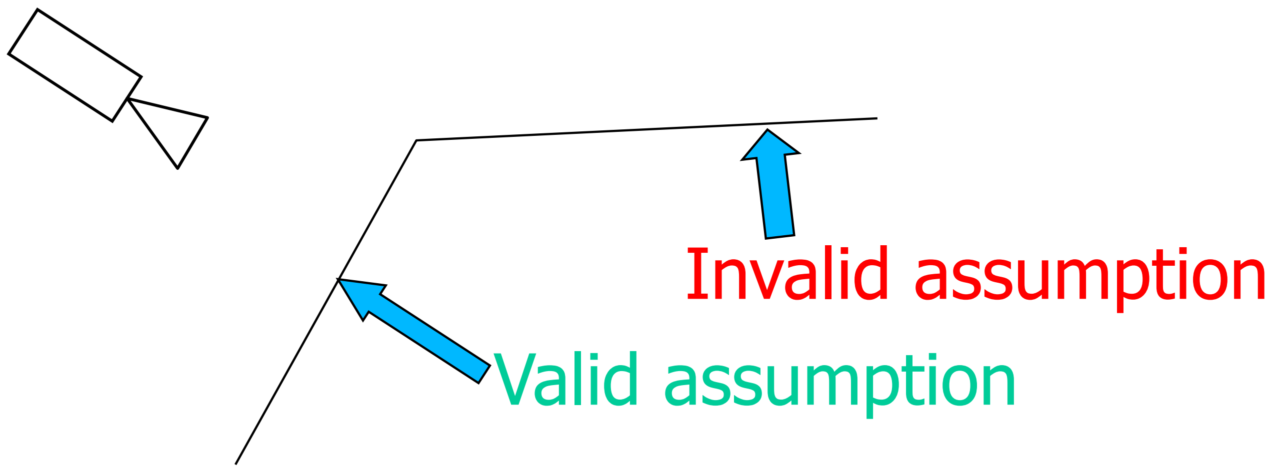
Difference of Gaussians

- Using a small window on a reduced image is equivalent to using a large one on the original image.
- Using difference of Gaussian images is an effective way of achieving normalization.

→It becomes natural to use results obtained using low resolution images to guide the search at higher resolution.

Fronto-Parallel Assumption

- The disparity is assumed to be the same over the entire correlation window, which is equivalent to assuming constant depth.



→ Ok when the surface faces the camera but breaks down otherwise.

Multi-View Stereo

Multi-view reconstruction setup

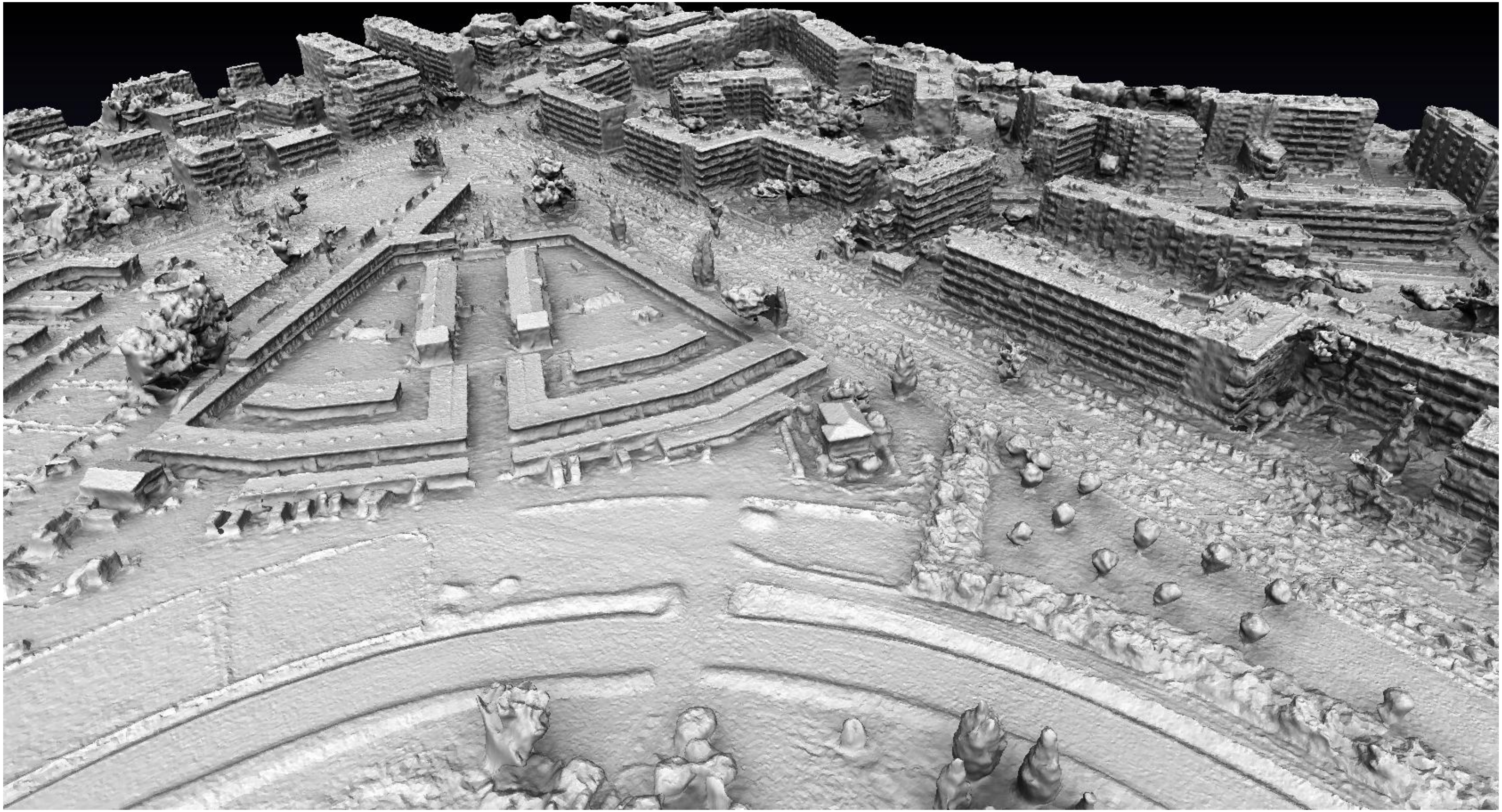
—> Adjust correlation window shapes to handle orientation.

Textured-Mapped 3D Model

Shaded 3D Model

Furukawa&Ponce ECCV'06

MULTI-VIEW STEREO



Aganj et al. ICCV'09

Small Drones

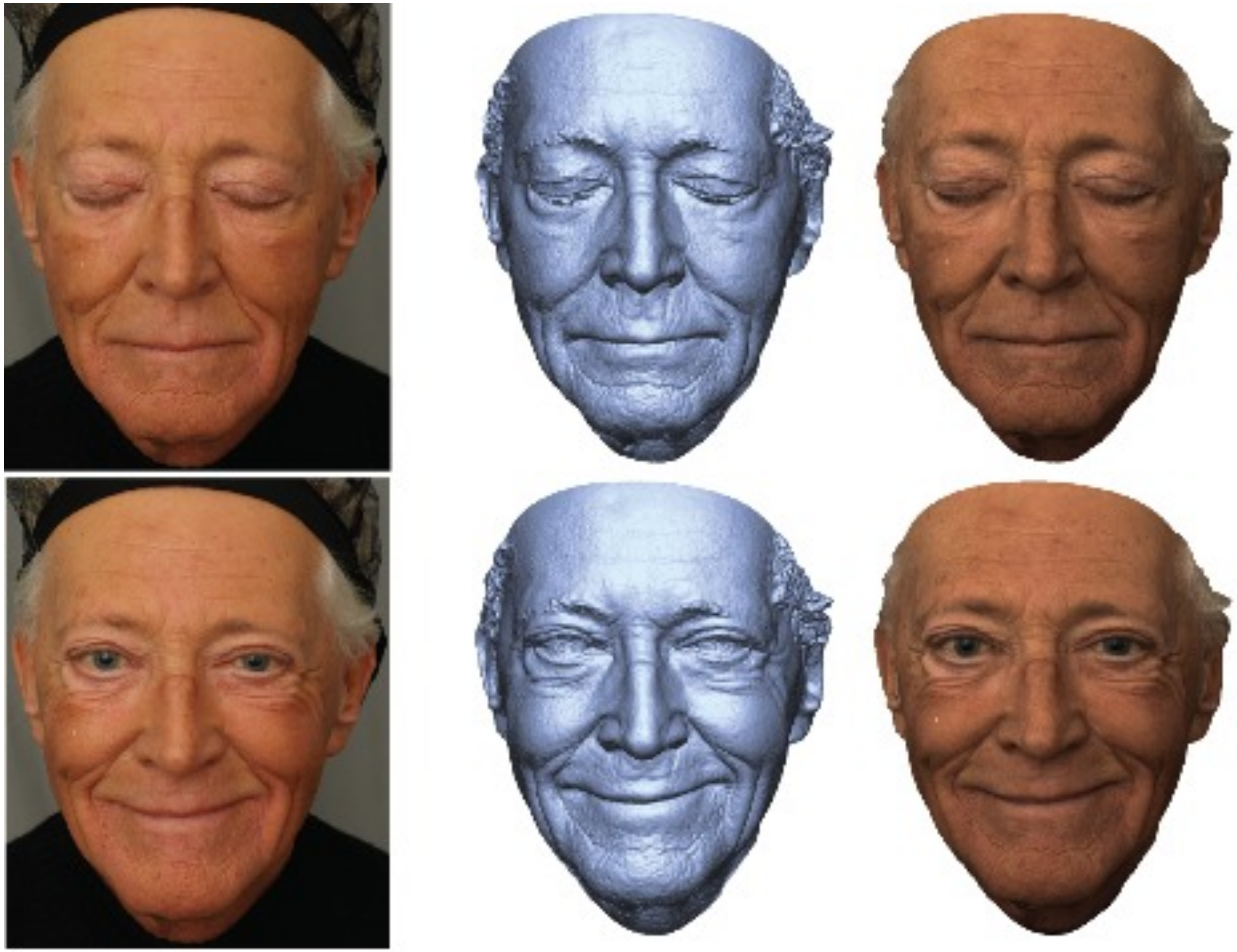
SenseFly:
www.sensefly.com

Gatewing:
www.gatewing.com

Matterhorn

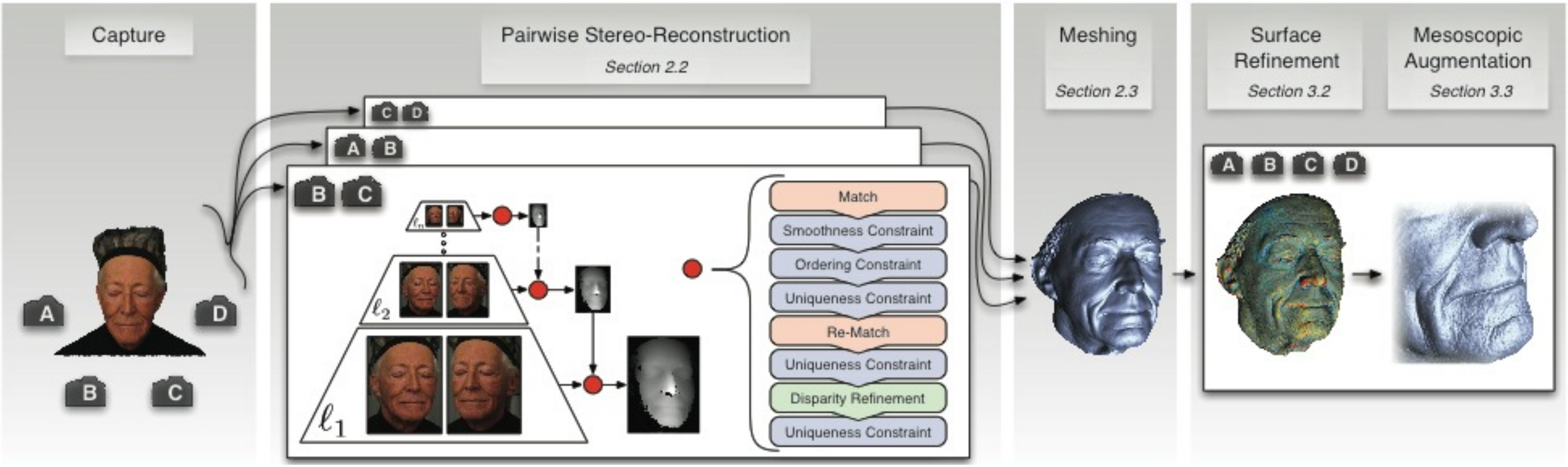
Drone: www.sensefly.com Mapping: www.pix4d.com

Face Reconstruction

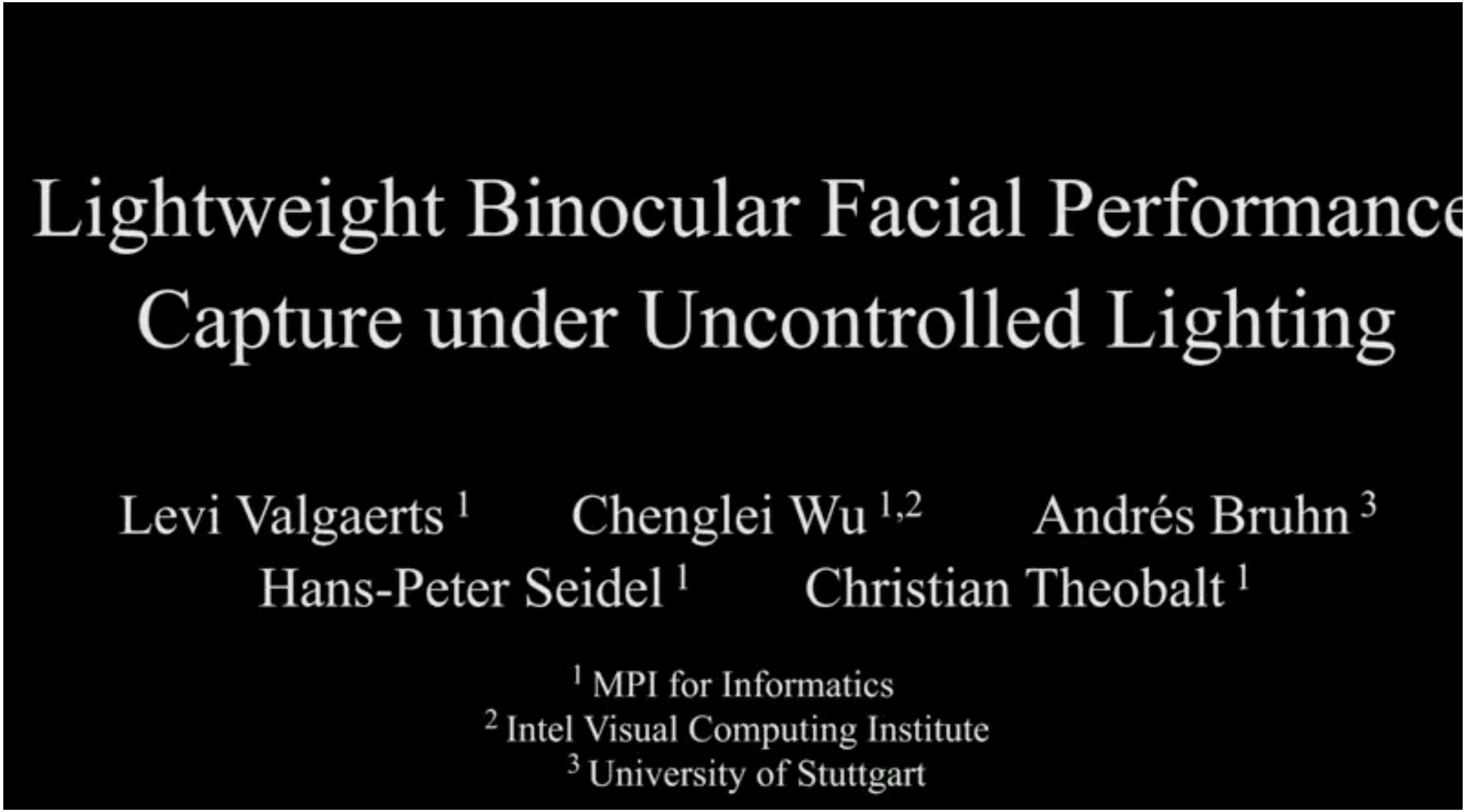


Beeler et al. SIGGRAPH'10

Face Reconstruction

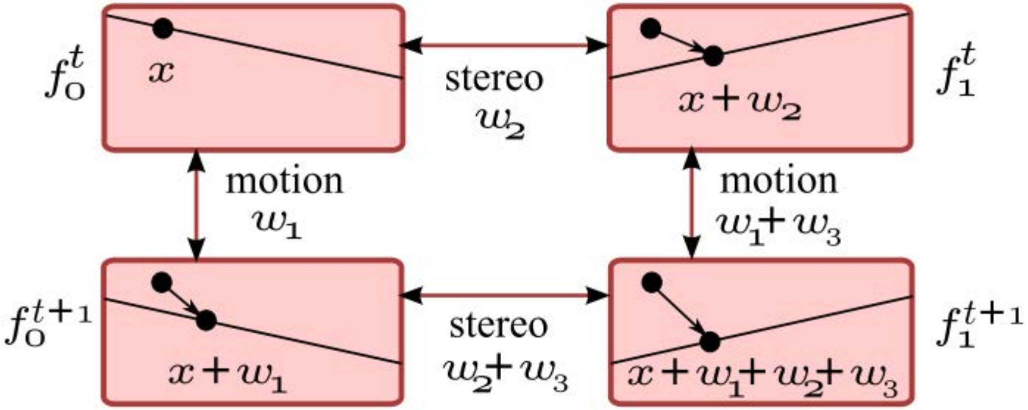


Dynamic Shape

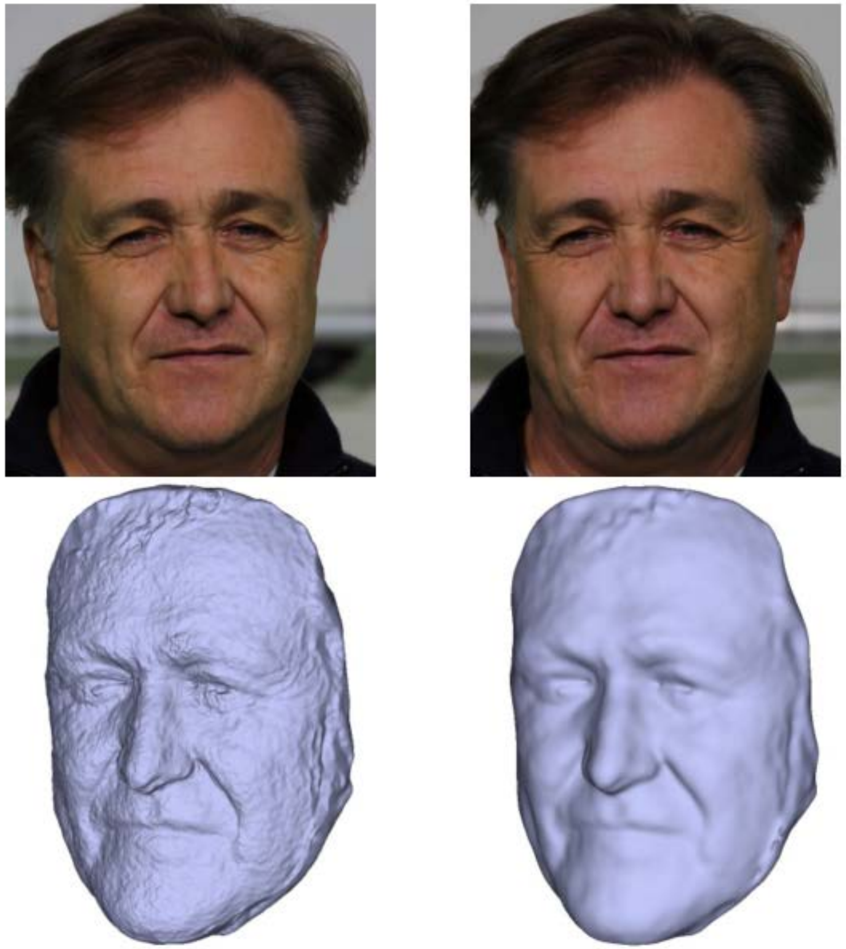


Valgaerts et al. SIGGRAPH Asia'12

Scene Flow



Correspondences across cameras and across time

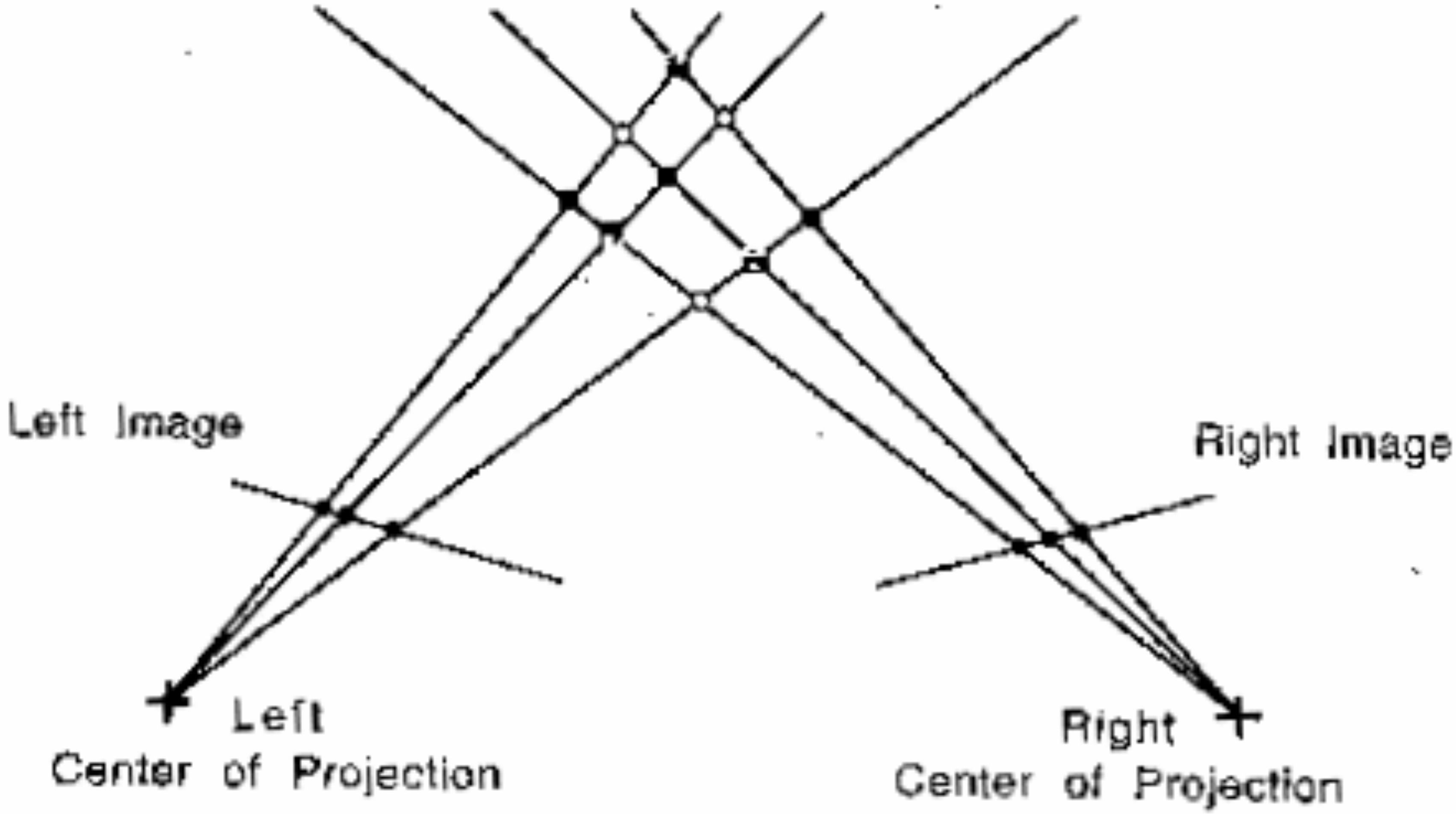


Stereo Only Stereo + Flow

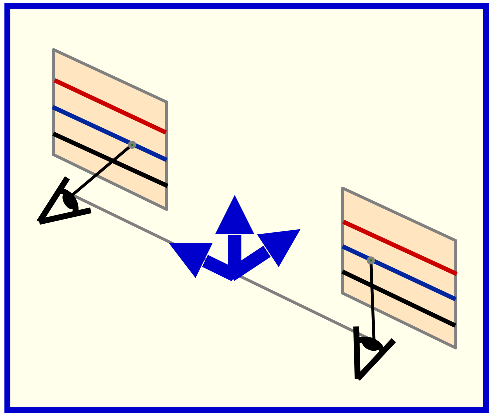
Refining using Shape From Shading

Shape-from-shading can be used to refine the shape and provide high-frequency details.

Uncertainty



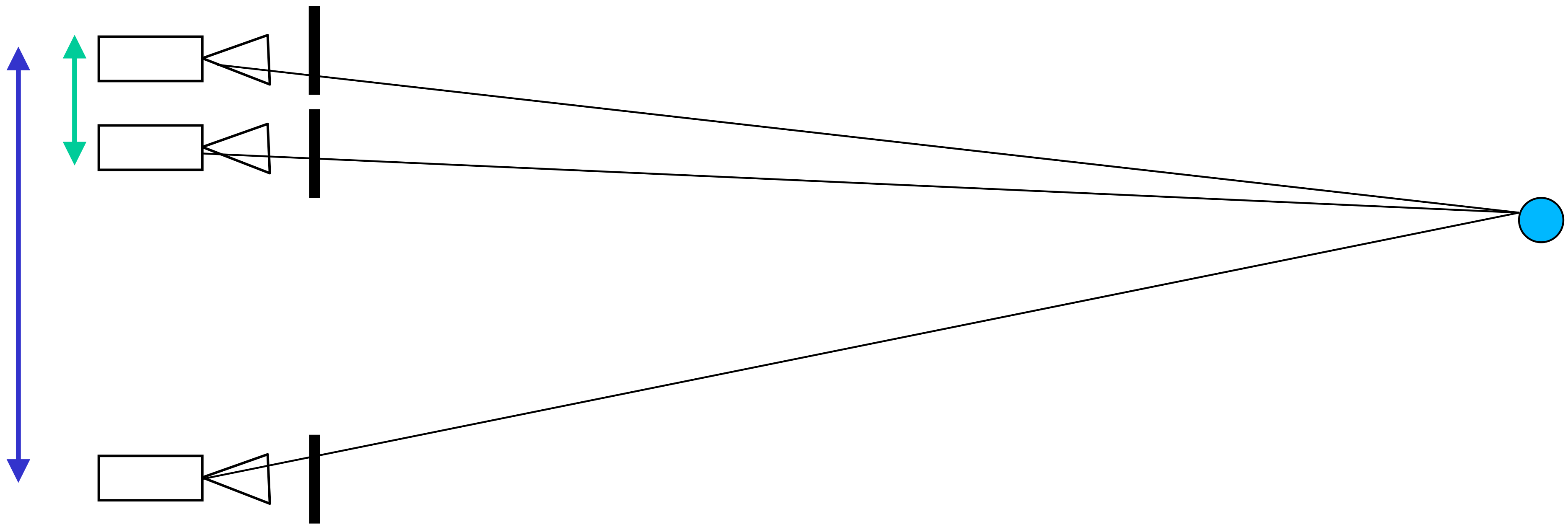
Precision vs Baseline



$$
\begin{aligned}
d &= f\frac{b}{Z} \\
\Rightarrow Z &= f\frac{b}{d} \\
\Rightarrow \frac{\delta Z}{\delta d} &= -f\frac{b}{d^2} = -\frac{Z^2}{fb}
\end{aligned}
$$

- Beyond a certain depth stereo stops being useful.
- Precision is inversely proportional to baseline length.

Short vs Long Baseline



Short baseline:

- Good matches
- Few occlusions
- Poor precision

Long baseline:

- Harder to match
- More occlusions
- Better precision

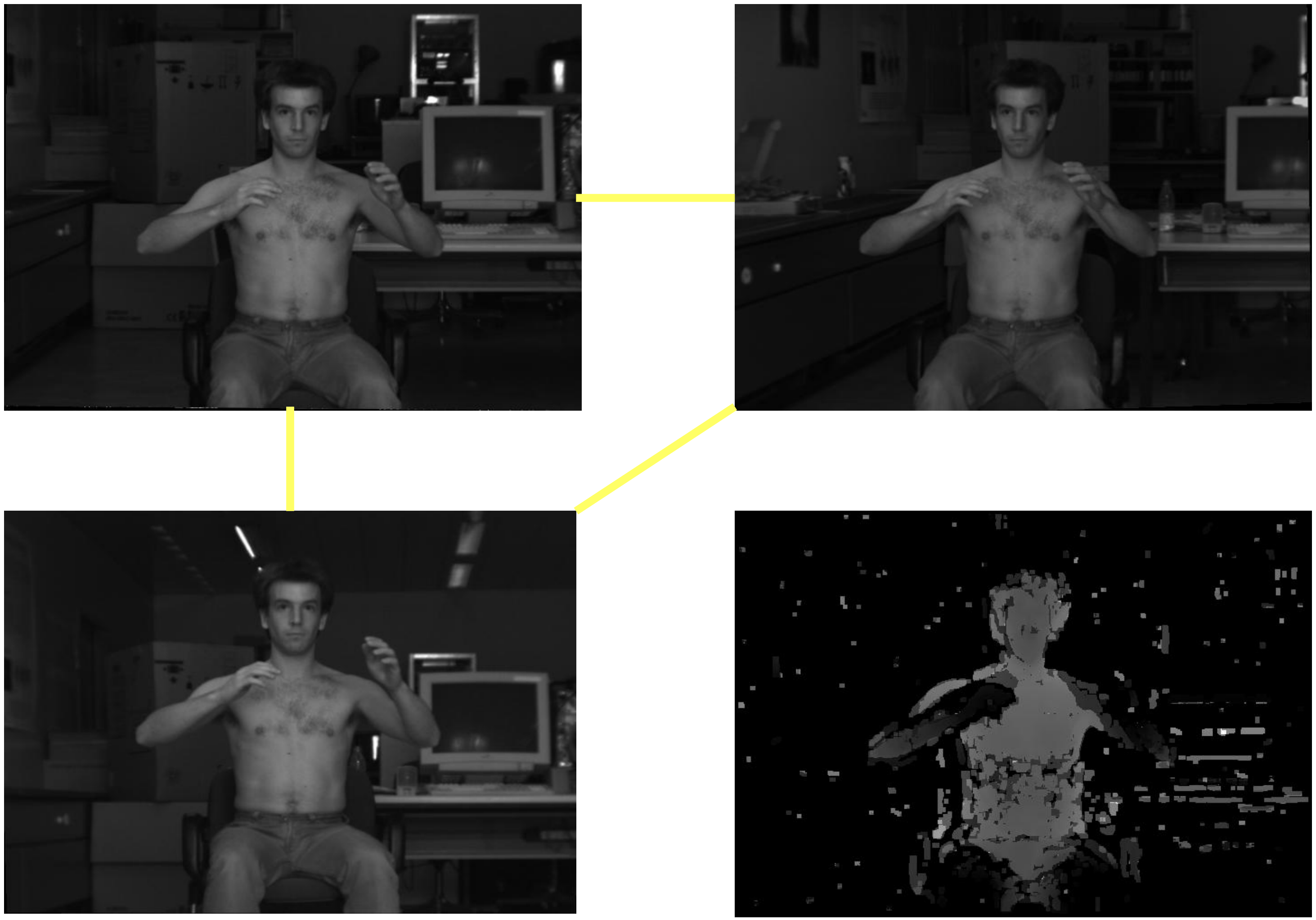
Mars Rover

There are four cameras!

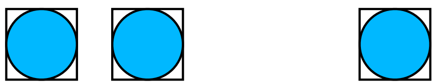
Video-Based Motion Capture

Fitting an articulated body model to stereo data.

Trinocular Stereo

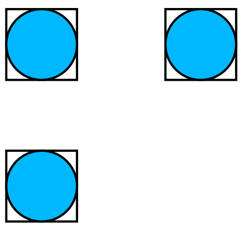


Multi-Camera Configurations



3 cameras give both robustness and precision

4 cameras give additional redundancy

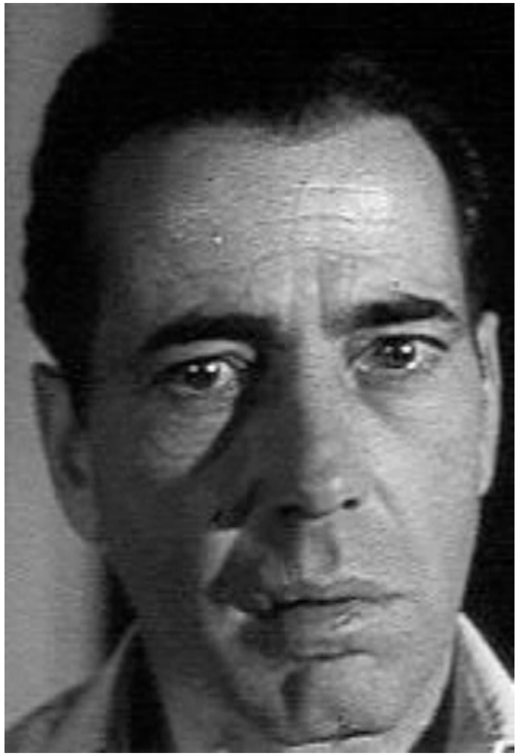


3 cameras in a T arrangement allow the system to see vertical lines.

Kinect: Structured Light

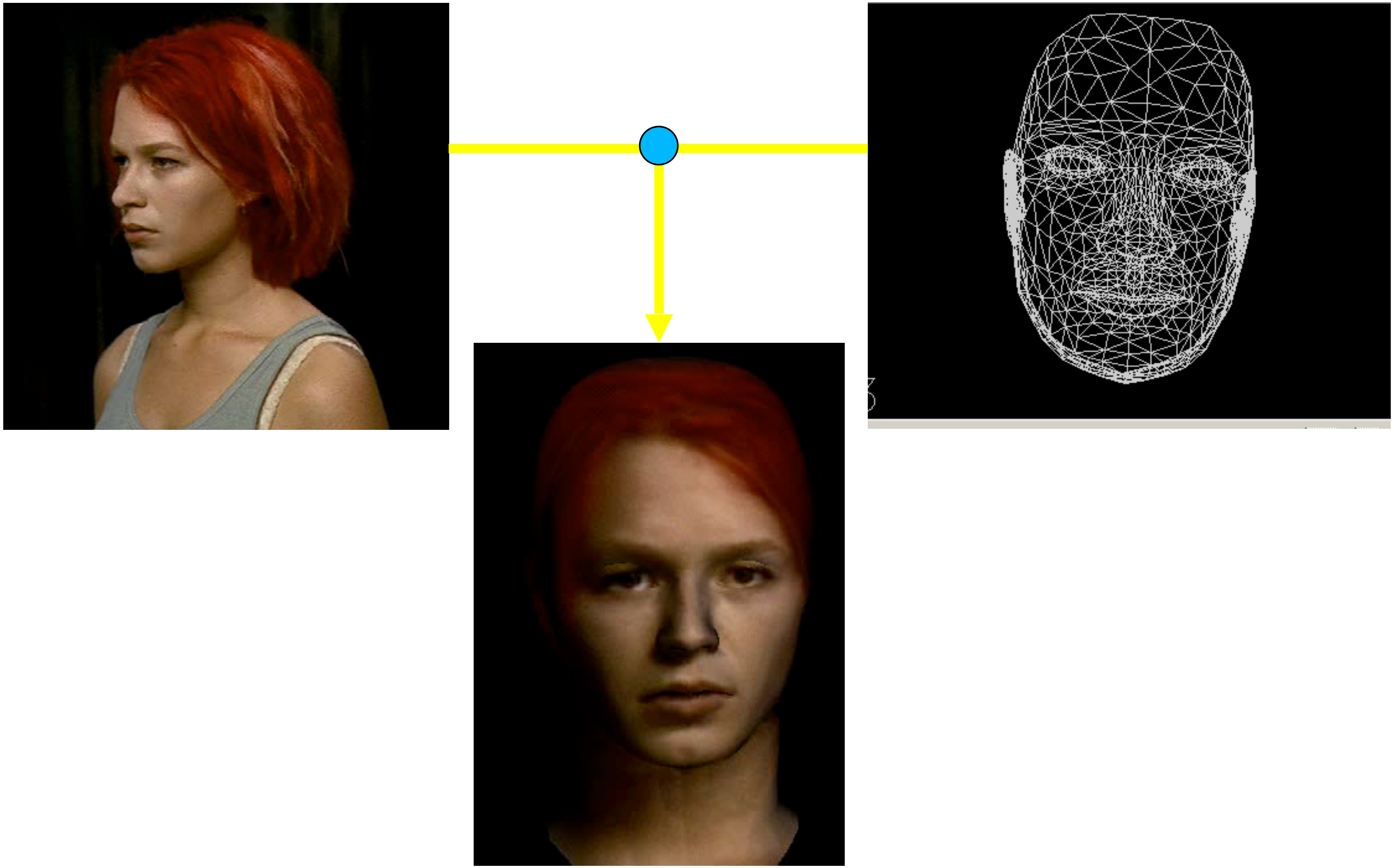
- The Kinect camera projects a IR pattern and measures depth from its distortion.
- Same principle but the second camera is replaced by the projector.

Faces from Low-Resolution Videos

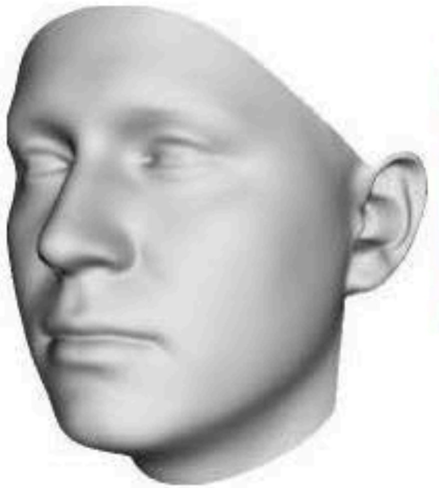


- No calibration data
- Relatively little texture
- Difficult lighting

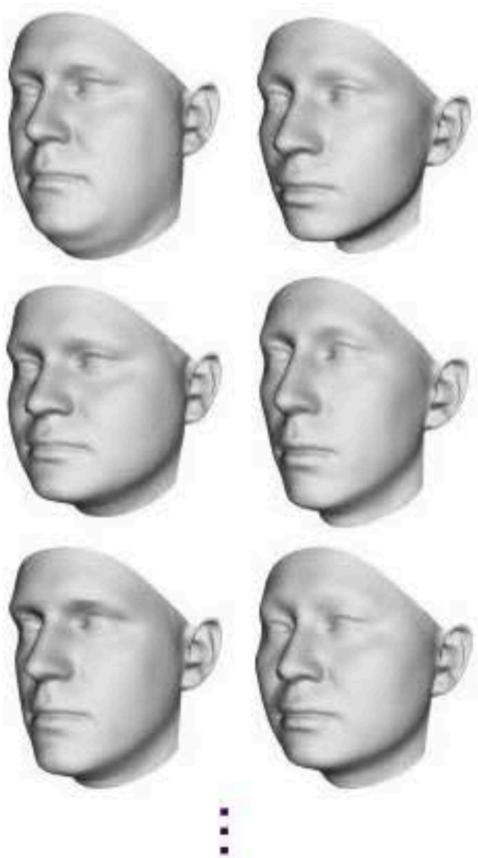
Simple Face Model



PCA Face Model



-2σ +2σ



$$S = \bar{S} + \sum_{i=1}^{99} \alpha_i S_j$$

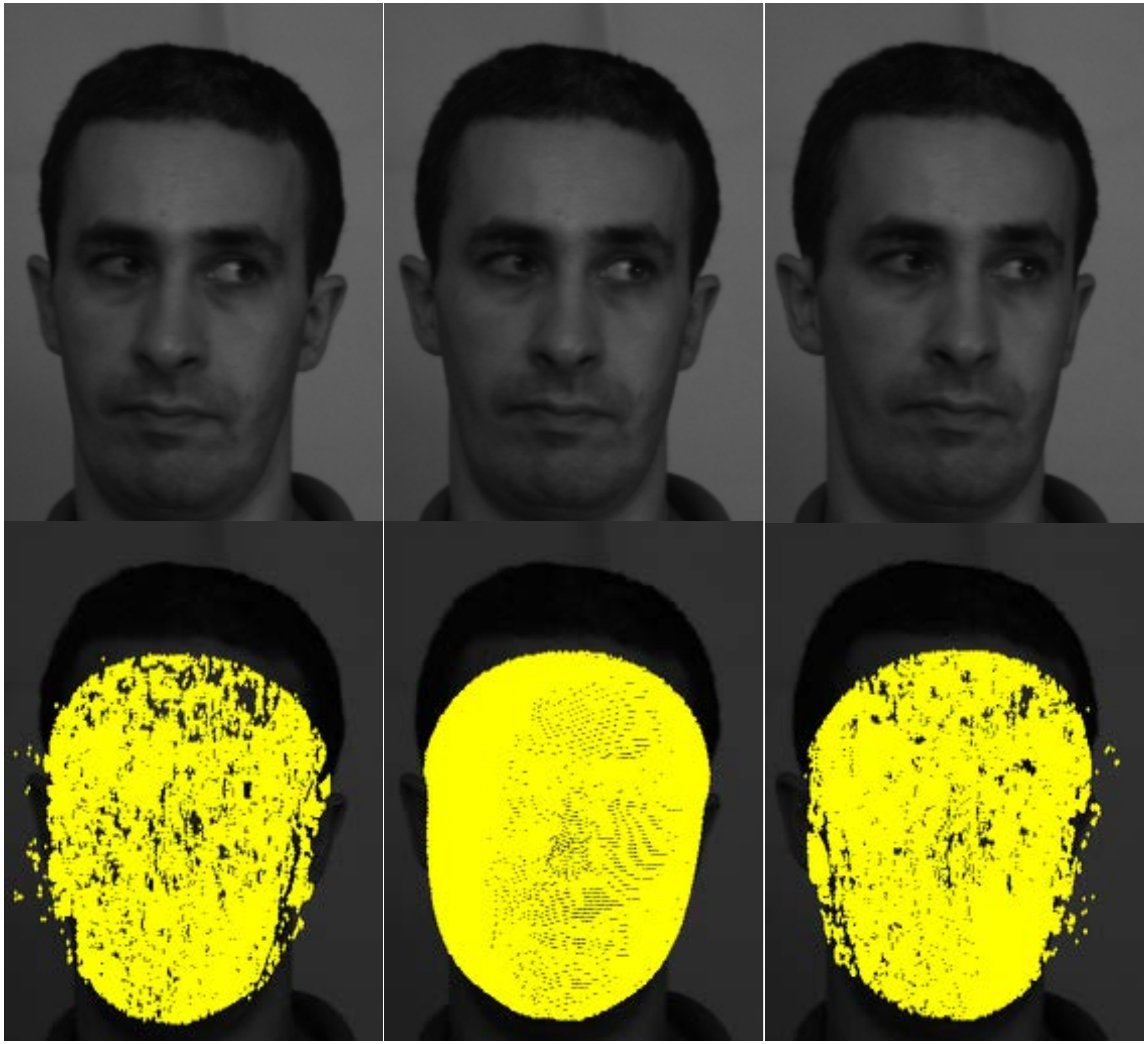
$\bar{S}$: Average shape

S_i : Shape vector

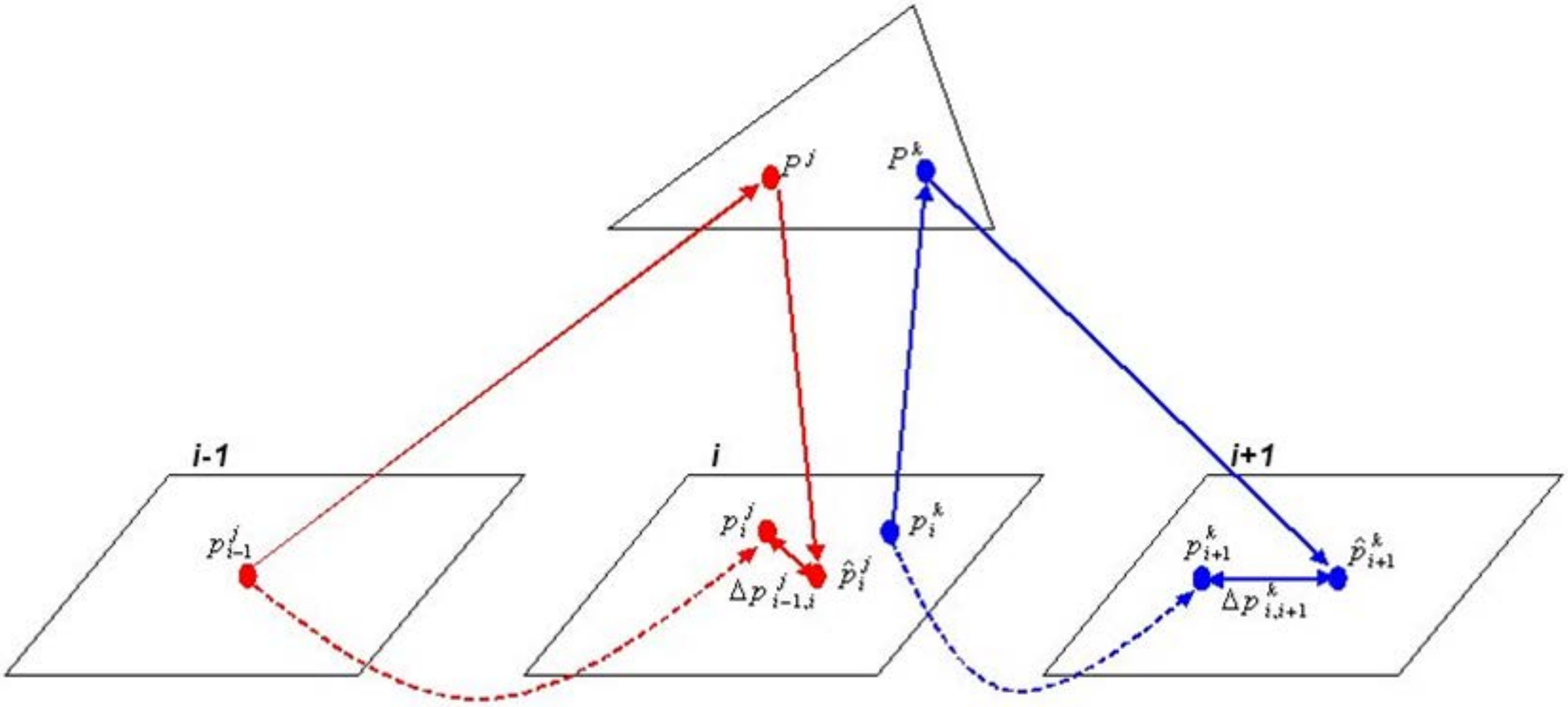
α_i : Shape coefficients

Blanz & Vetter, SIGGRAPH'99

Correspondences

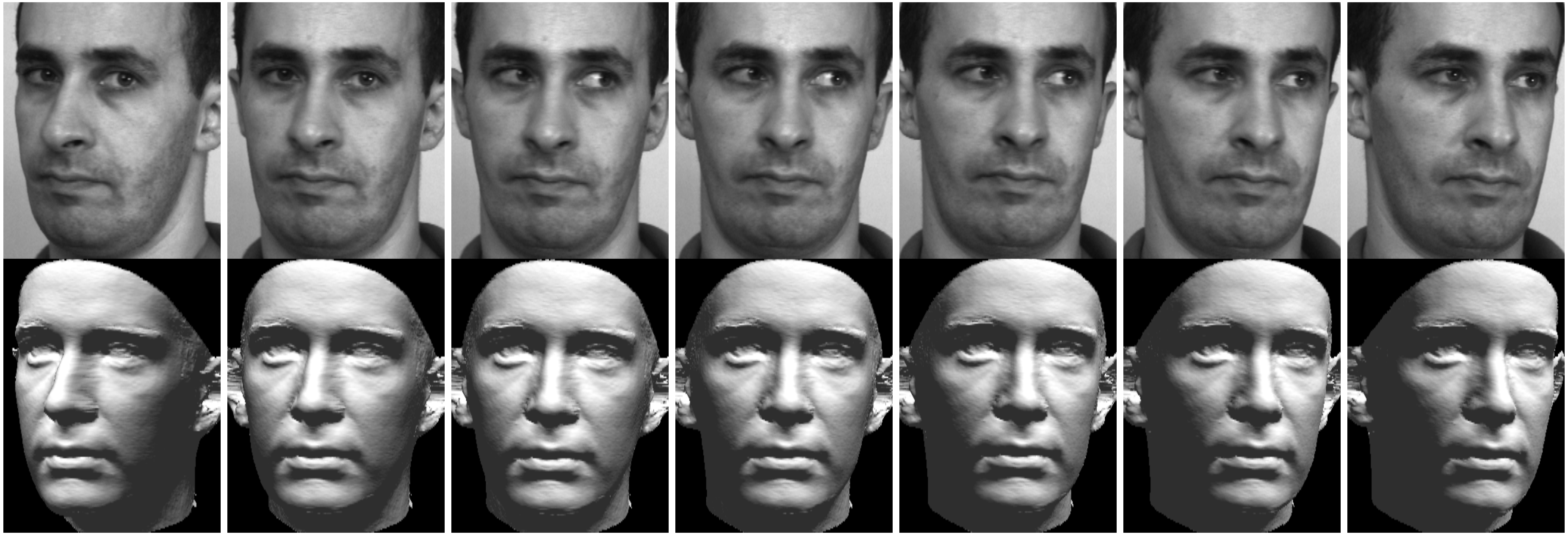


Transfer Function



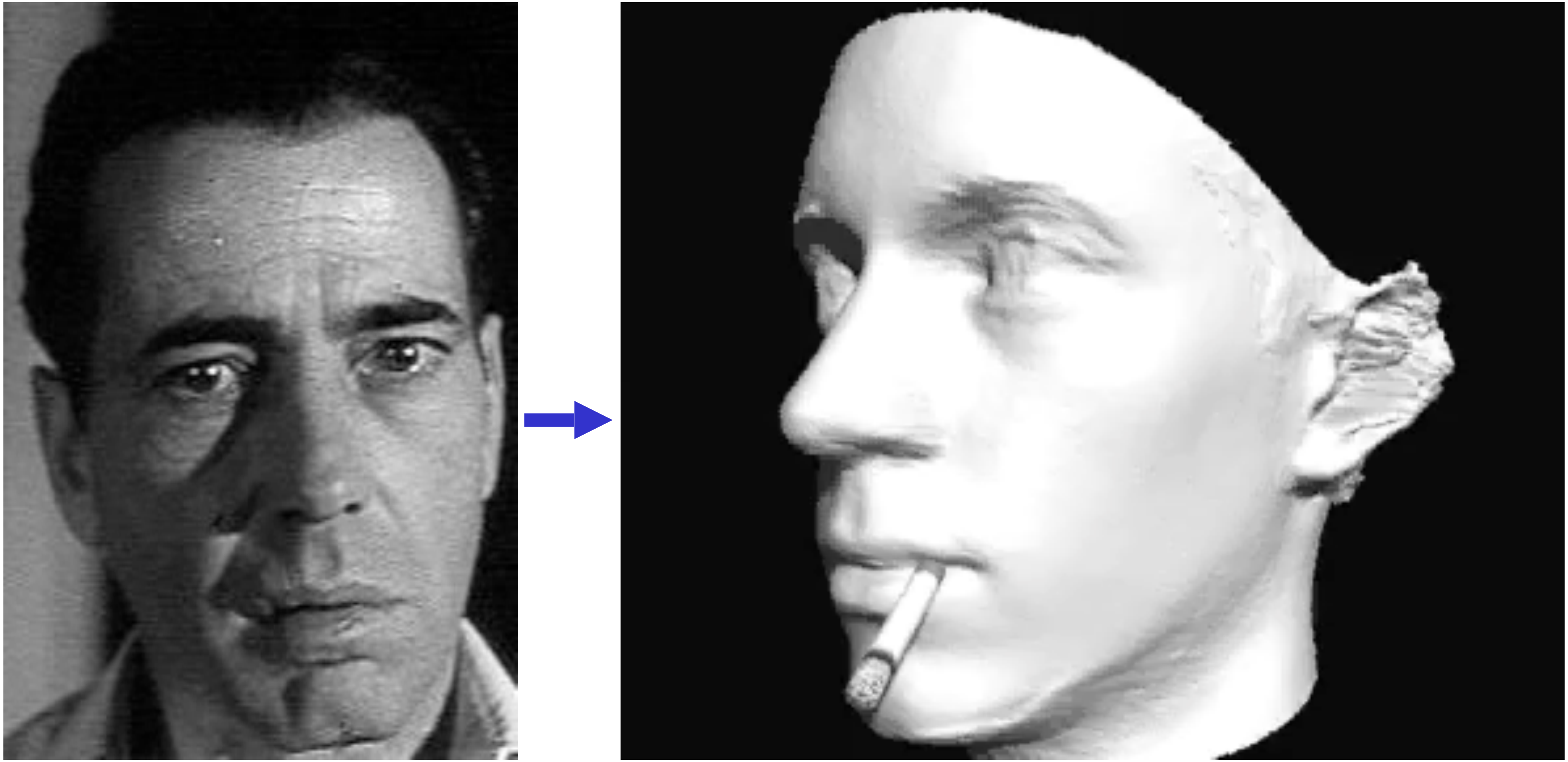
$$F_3(A, C_{i-1}, C_i, C_{i+1}) = \sum_{j \in Q_{i-1}} \left\| \Delta p^j_{i-1,i} \right\|^2 + \sum_{k \in Q_i} \left\| \Delta p^k_{i,i+1} \right\|^2$$

Model Based Bundle Adjustment



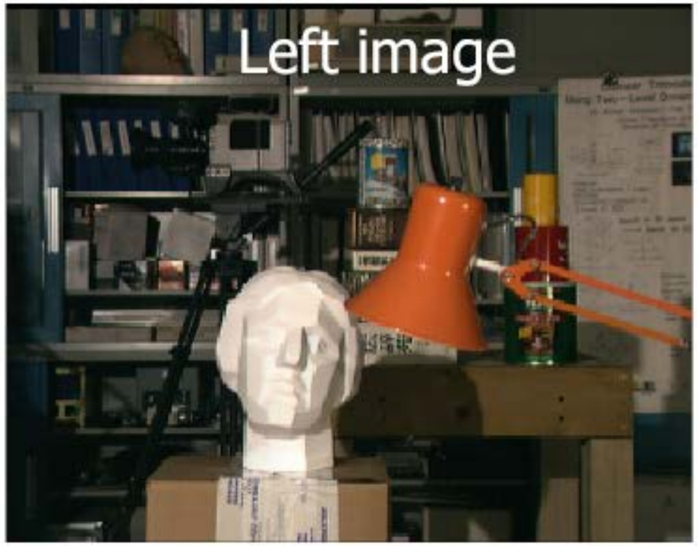
Adjusting the PCA coefficients to minimize the objective function yields an accurate face reconstruction from low-resolution images.

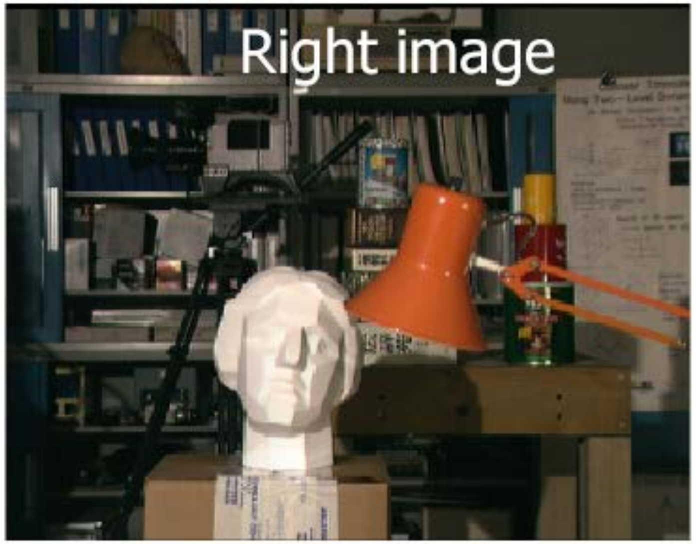
Model from Old Movie



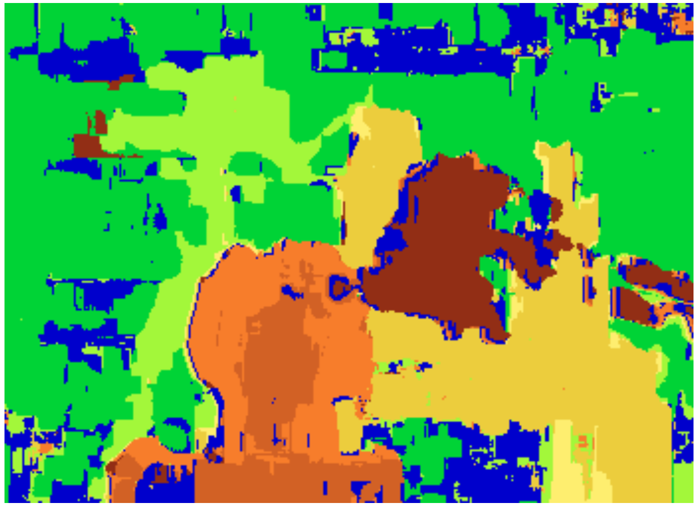
Adjusting the PCA coefficients to minimize the objective function yields an accurate face reconstruction from low-resolution images.

Limitations Of Window Based Methods



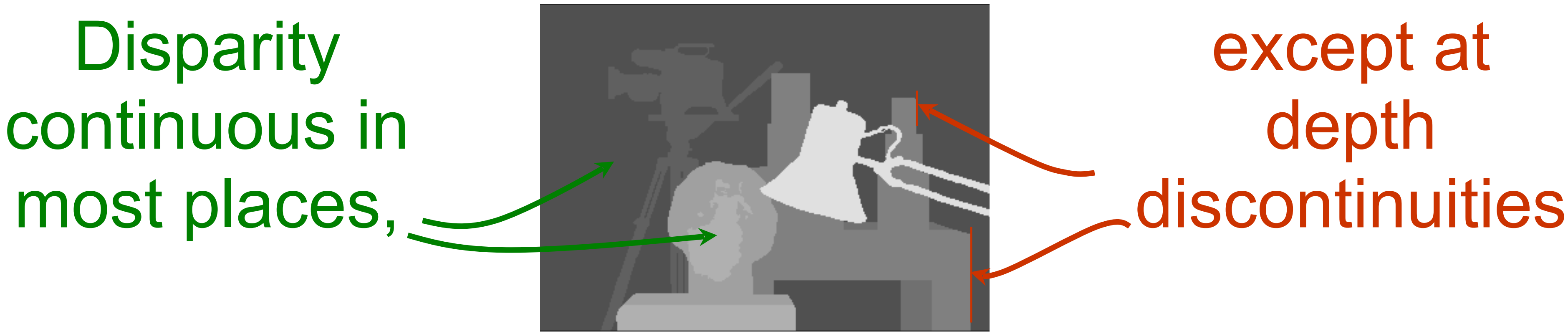


Ground truth



Correlation result

Energy Minimization

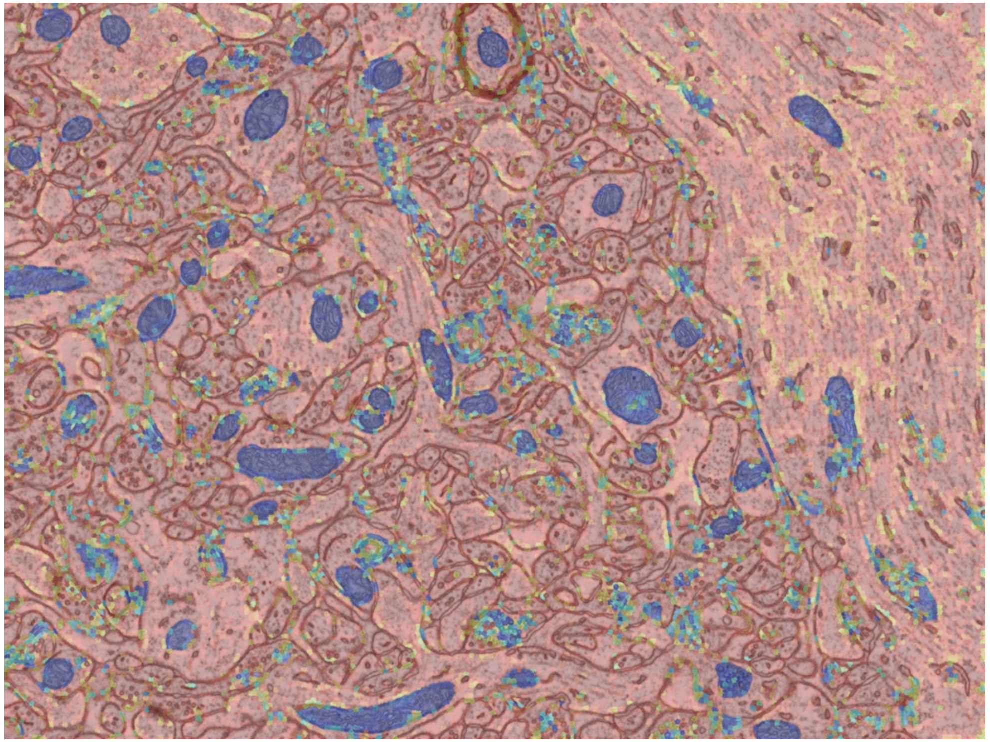


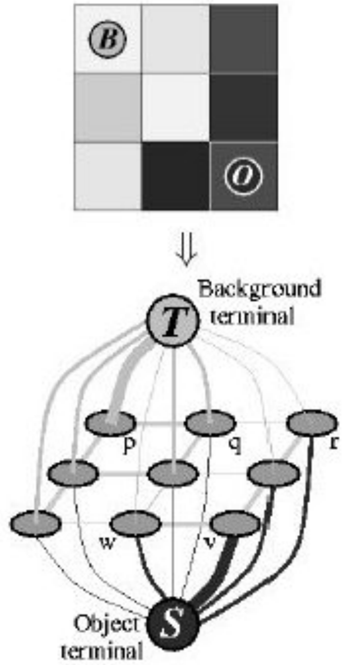
1. Matching pixels should have similar intensities.
2. Most nearby pixels should have similar disparities

→ Minimize

$$\sum [I_2(x+D(x,y),y)-I_1(x,y)]^2+\lambda\sum [D(x+1,y)-D(x,y)]^2+\mu\sum [D(x,y+1)-D(x,y)]^2$$

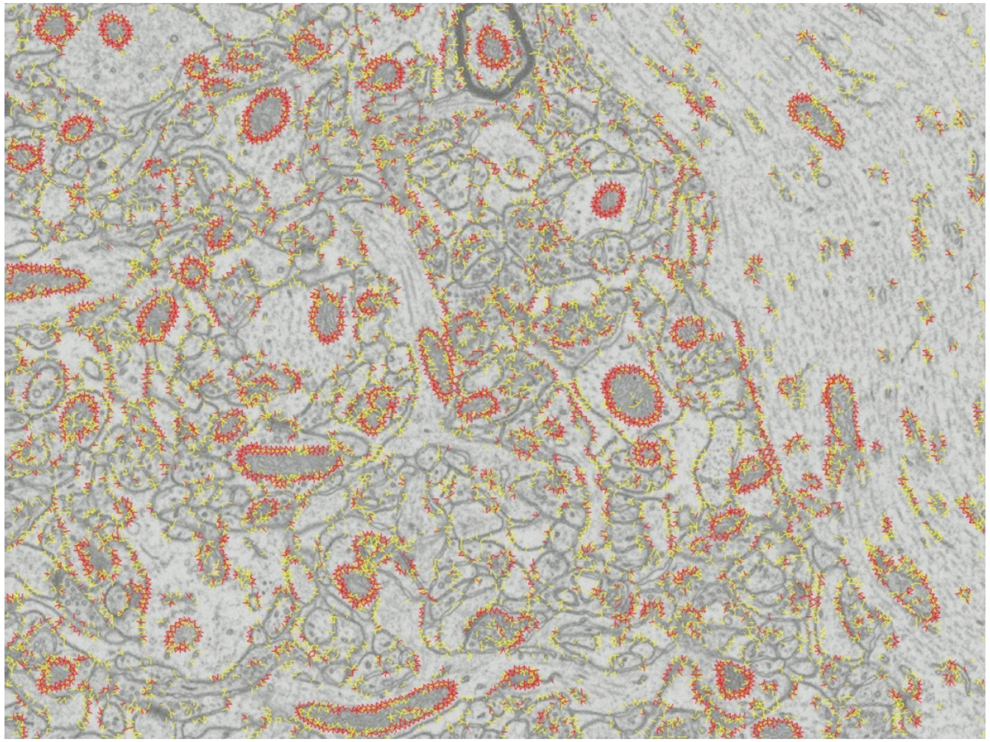
Reminder: Graph-Based Segmentation

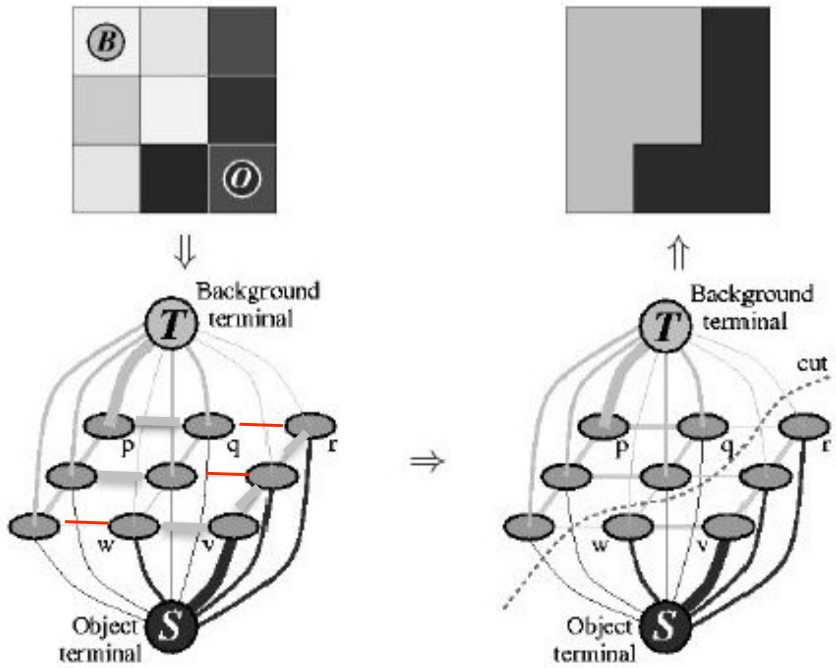




- A high probability of being a mitochondria can be represented by a strong edge connecting a supervoxel to the source and a weak one to the sink.
- And conversely for a low probability.

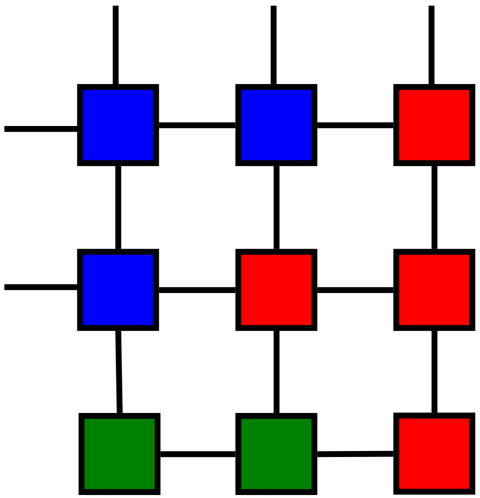
Reminder: Graph-Based Segmentation

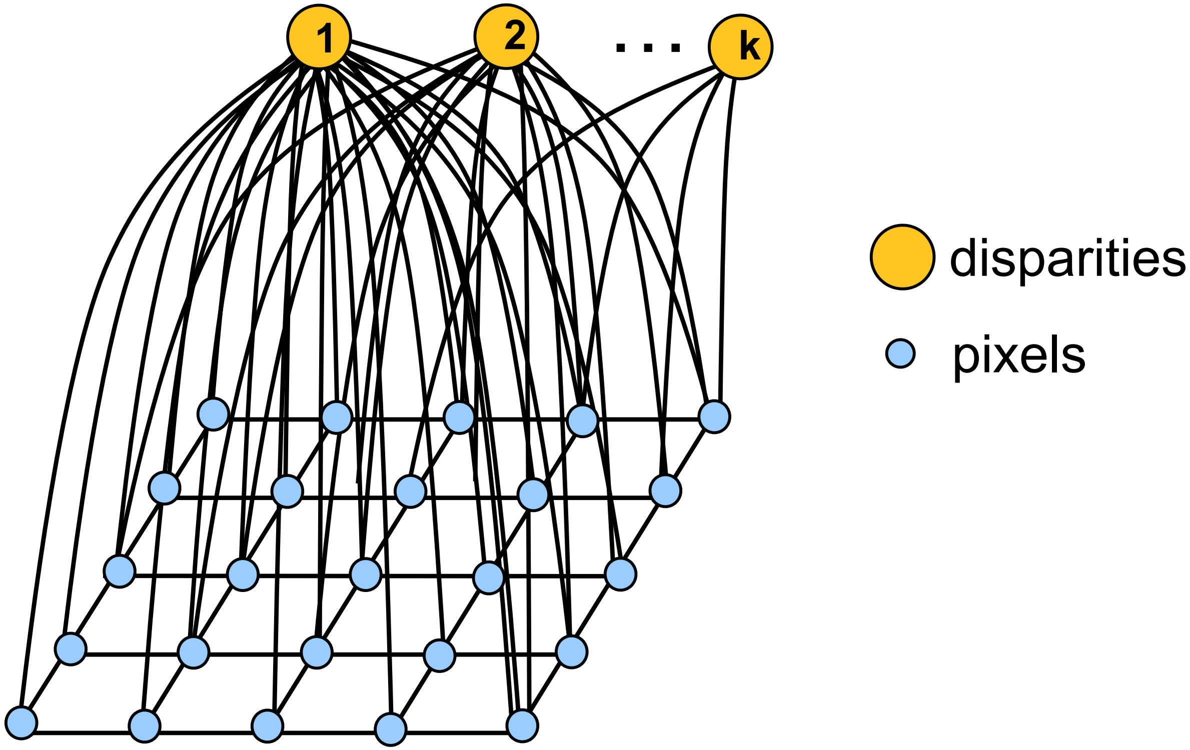




- Another classifier can be trained to assign a high-weight to edges connecting supervoxels belonging to the same class and a low one to others.
- Graph-cut can then be used to partition the pixels into separate regions.

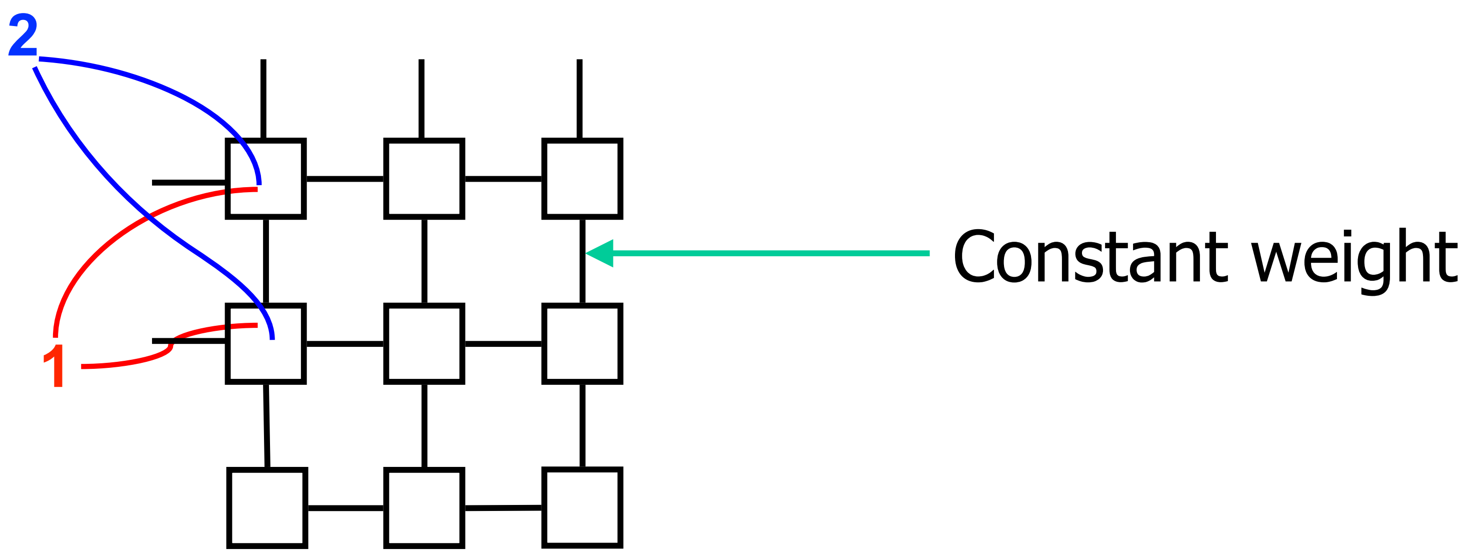
Graph Cut for Stereo





1. Stereo is a labeling problem. —> Use graph cut.
2. Connect each pixel to each possible disparity value.

Assigning Edge Weights



Assign a weight that is inversely proportional to |I2(x+1,y)-I1(x,y)|

Assign a weight that is inversely proportional to |I2(x+2,y)-I1(x,y)|

……

Minimizing the Objective Function

Minimize:

$$\sum [I_2(x+D(x,y),y)-I_1(x,y)]^2+\lambda\sum [D(x+1,y)-D(x,y)]^2+\mu\sum [D(x,y+1)-D(x,y)]^2$$

Graph cut algorithm:

- Guarantees an absolute minimum only when there are only two possible disparities.
- Effective heuristics (α-expansion, α–β swap) otherwise.

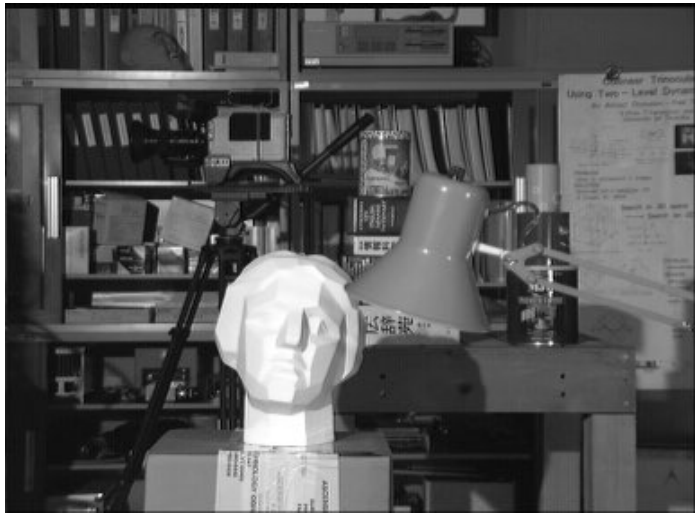
NCC vs Graph-Cut

Normalized correlation

Graph Cut

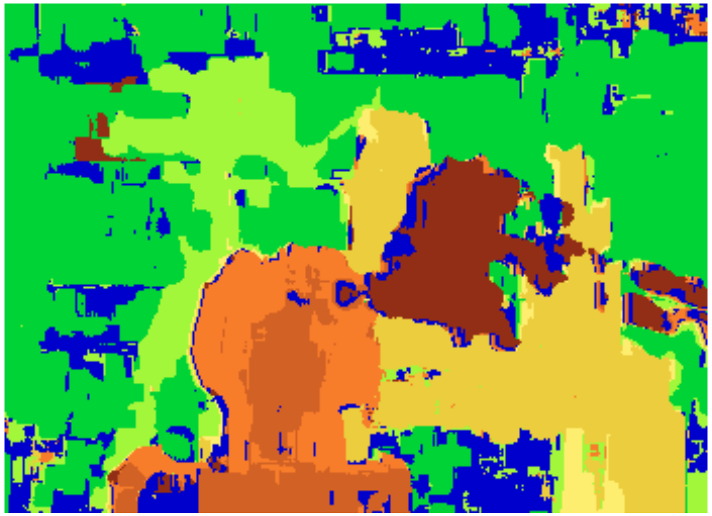
NCC vs Graph Cut

left image



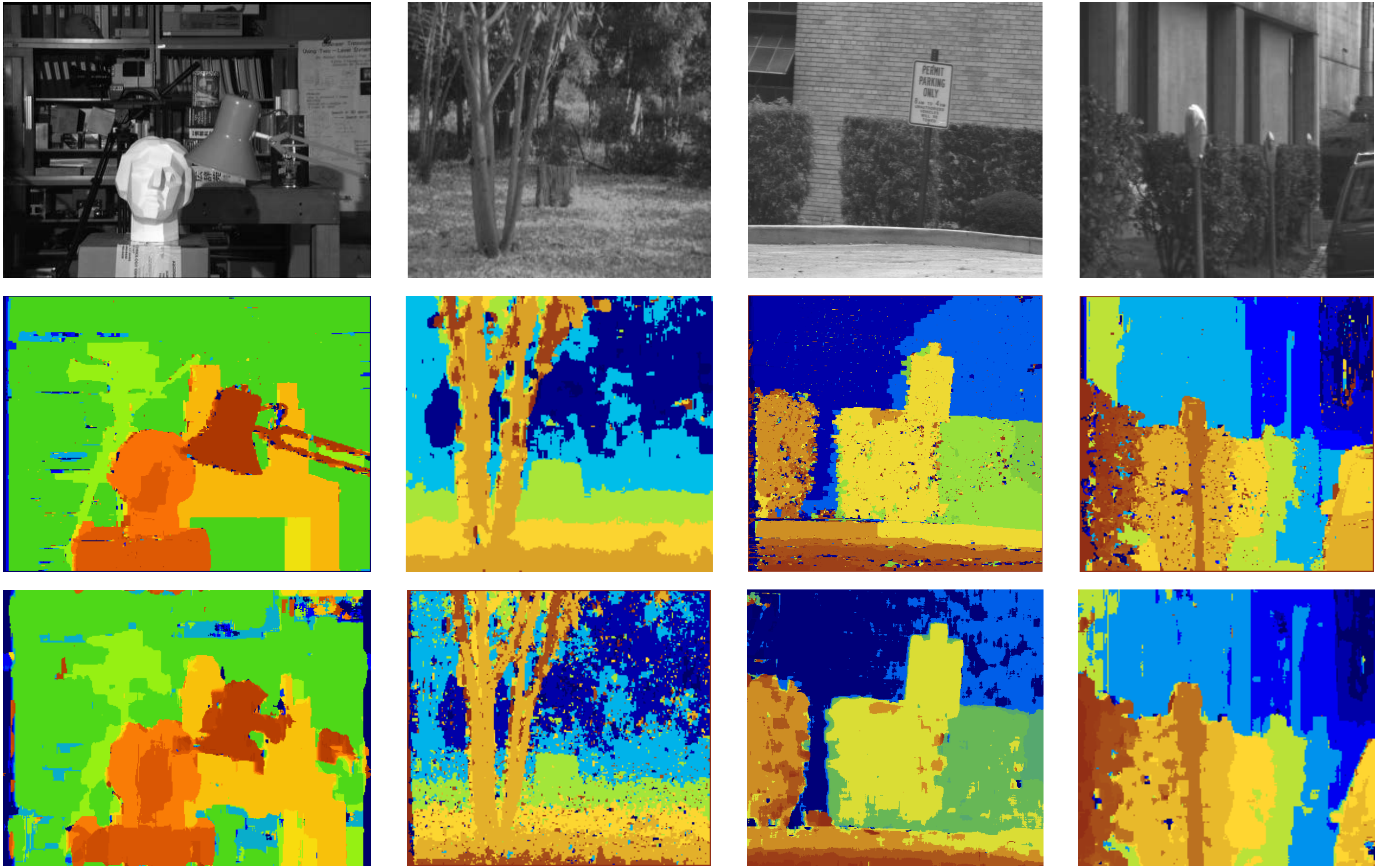
true disparities

Normalized correlation



Graph Cuts

NCC vs Graph Cut



Strengths and Limitations

Strengths:

- Practical method for depth recovery.
- Runs in real-time on ordinary hardware.

Limitations:

- Requires multiple views.
- Only applicable to reasonably textured objects.