

# SHAPE FROM X

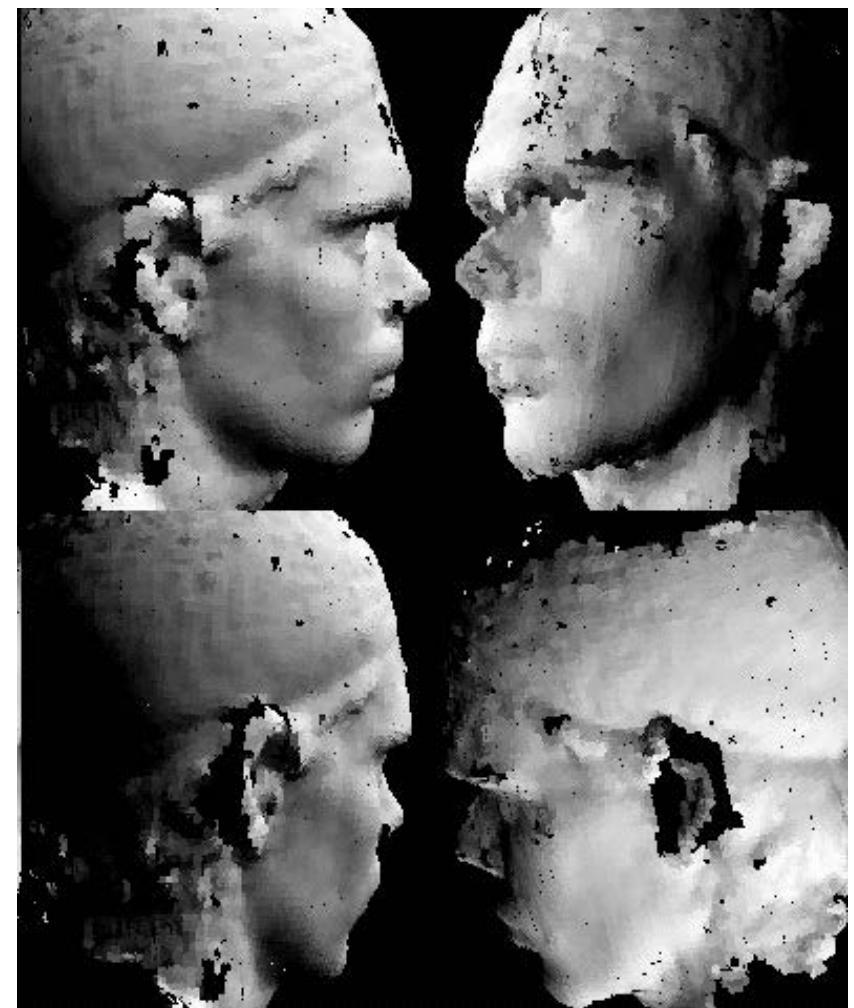
---

One image:

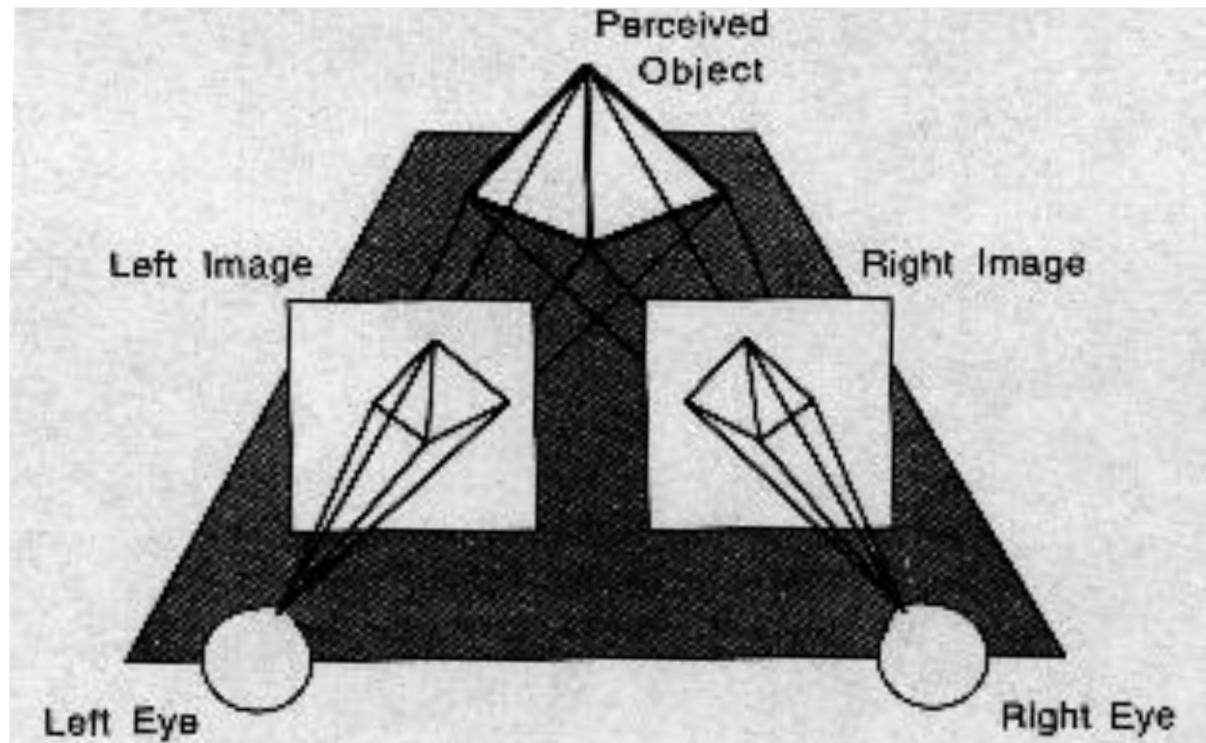
- Texture
- Shading

Two images or more:

- **Stereo**
- Contours
- Motion



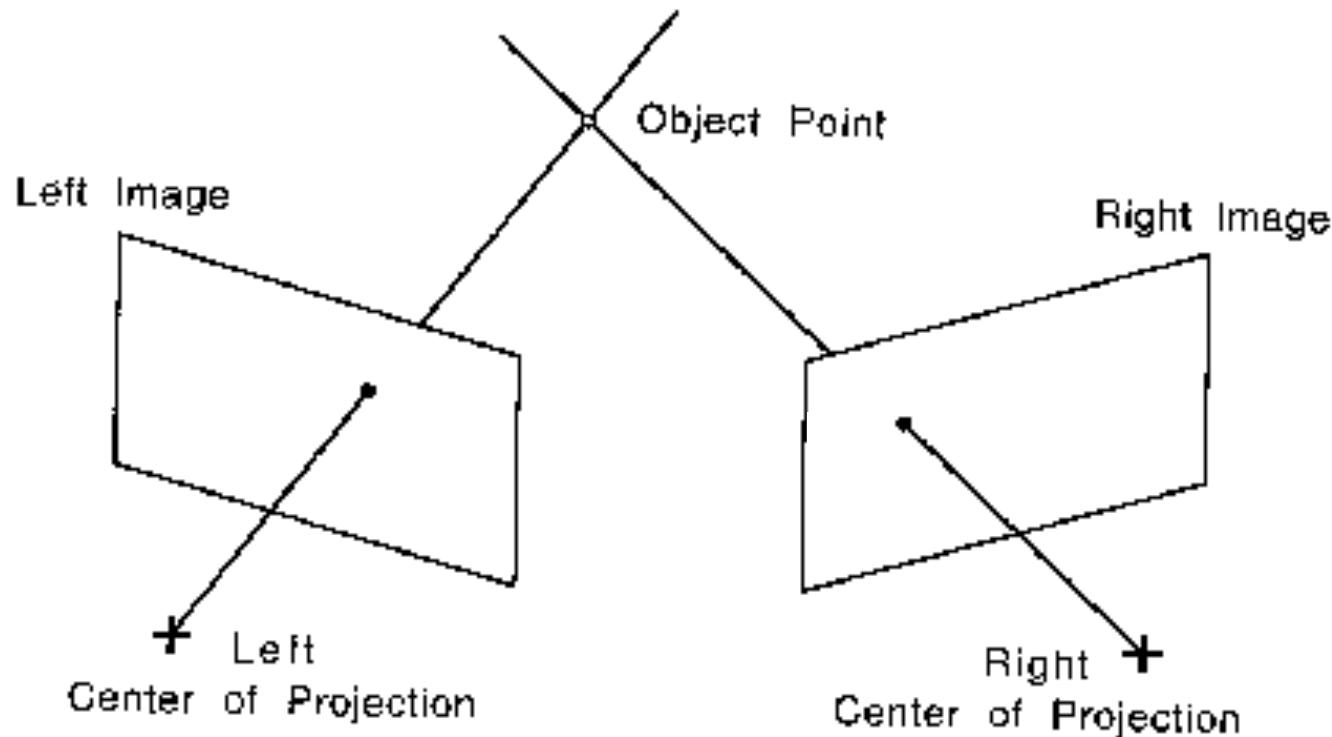
# GEOMETRIC STEREO



Depth from two or more images:

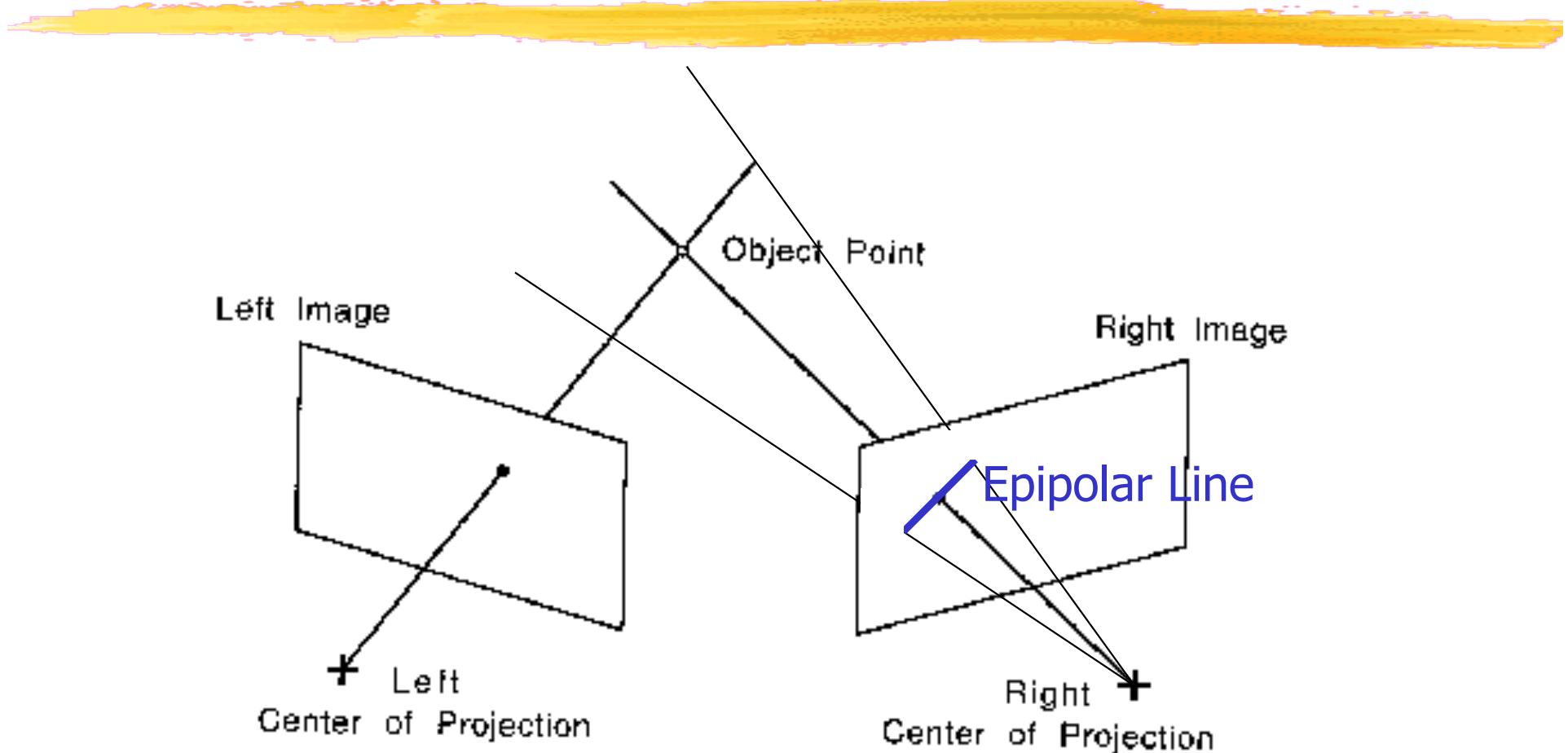
- Geometry of image pairs
- Establishing correspondences

# TRIANGULATION



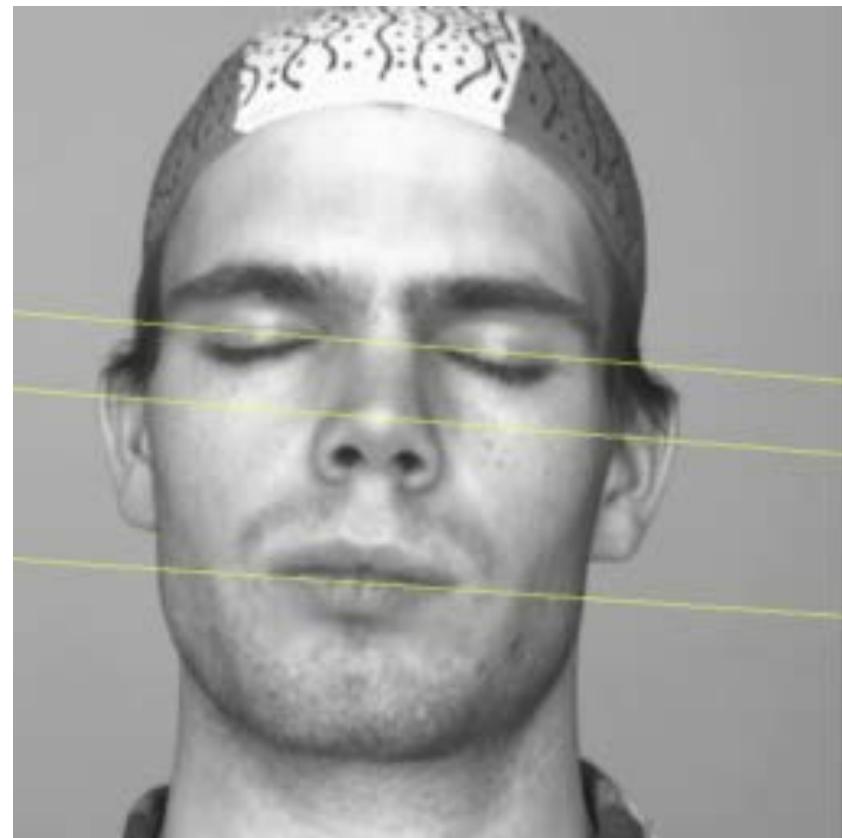
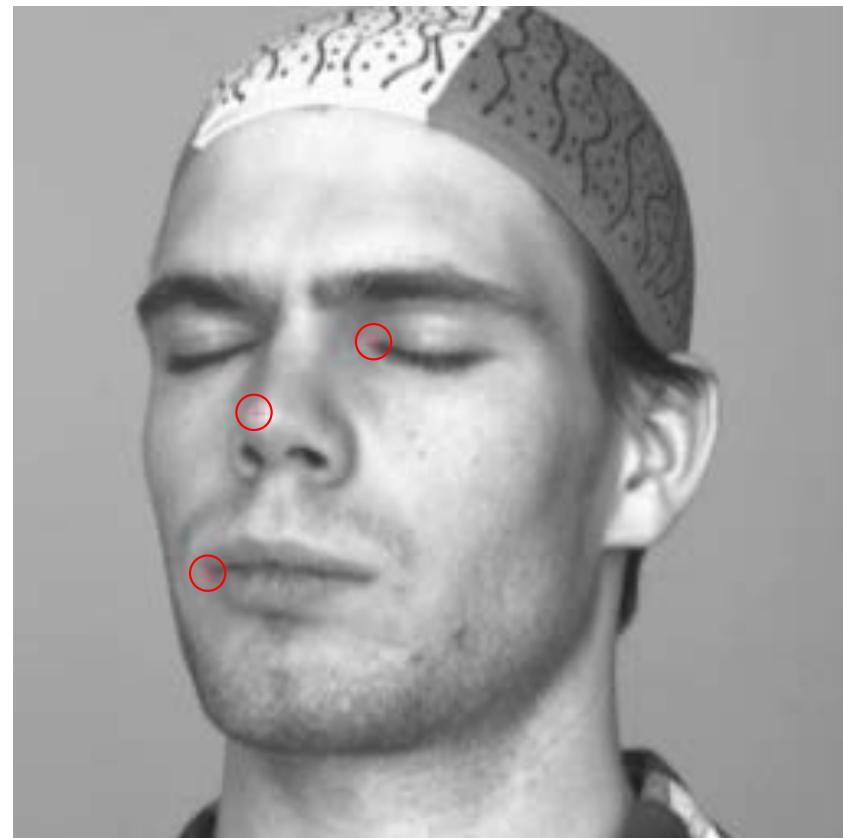
**Geometric Stereo:** Depth from two images

# EPIPOLAR LINE



Line on which the corresponding point must lie.

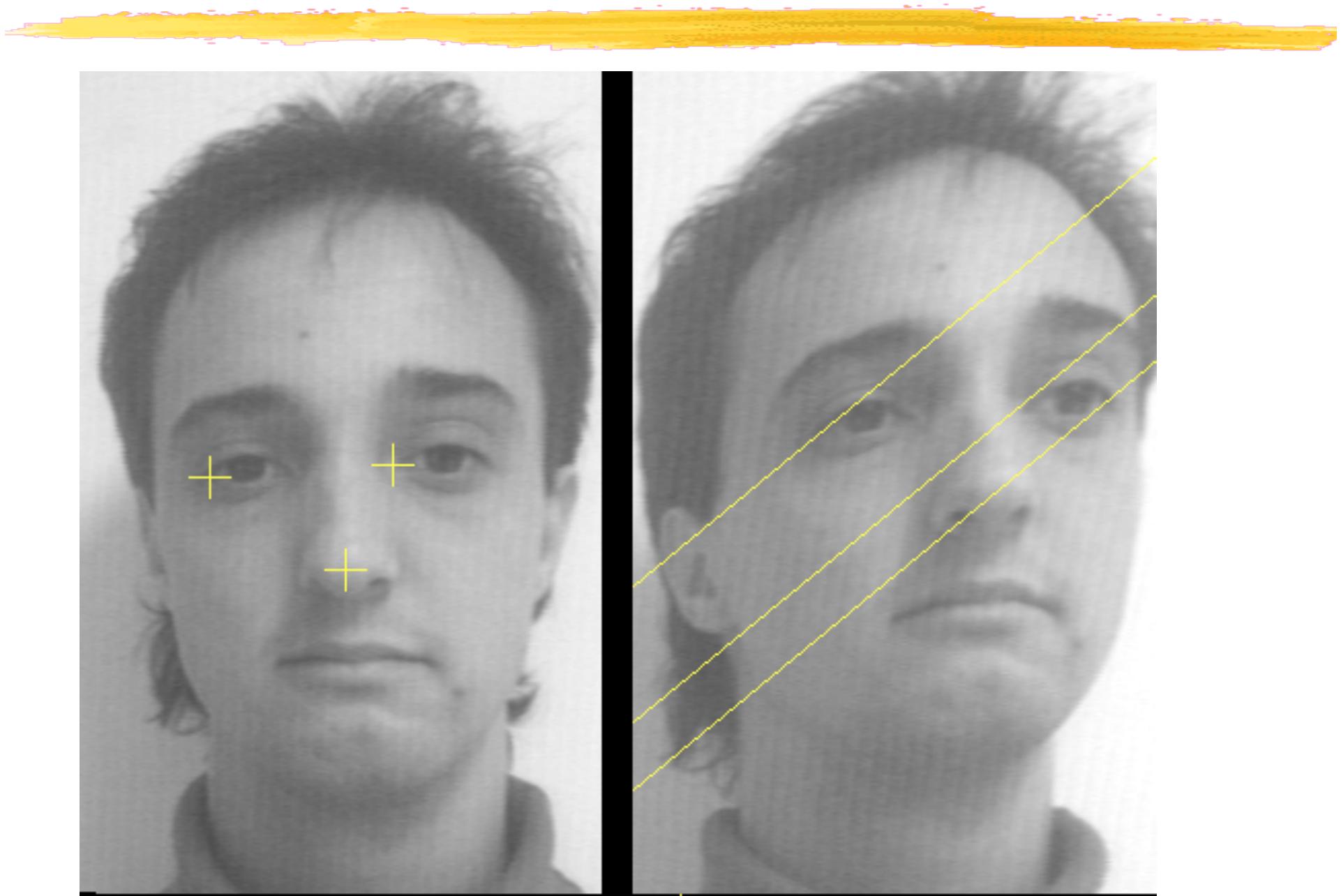
# EPIPOLAR LINES



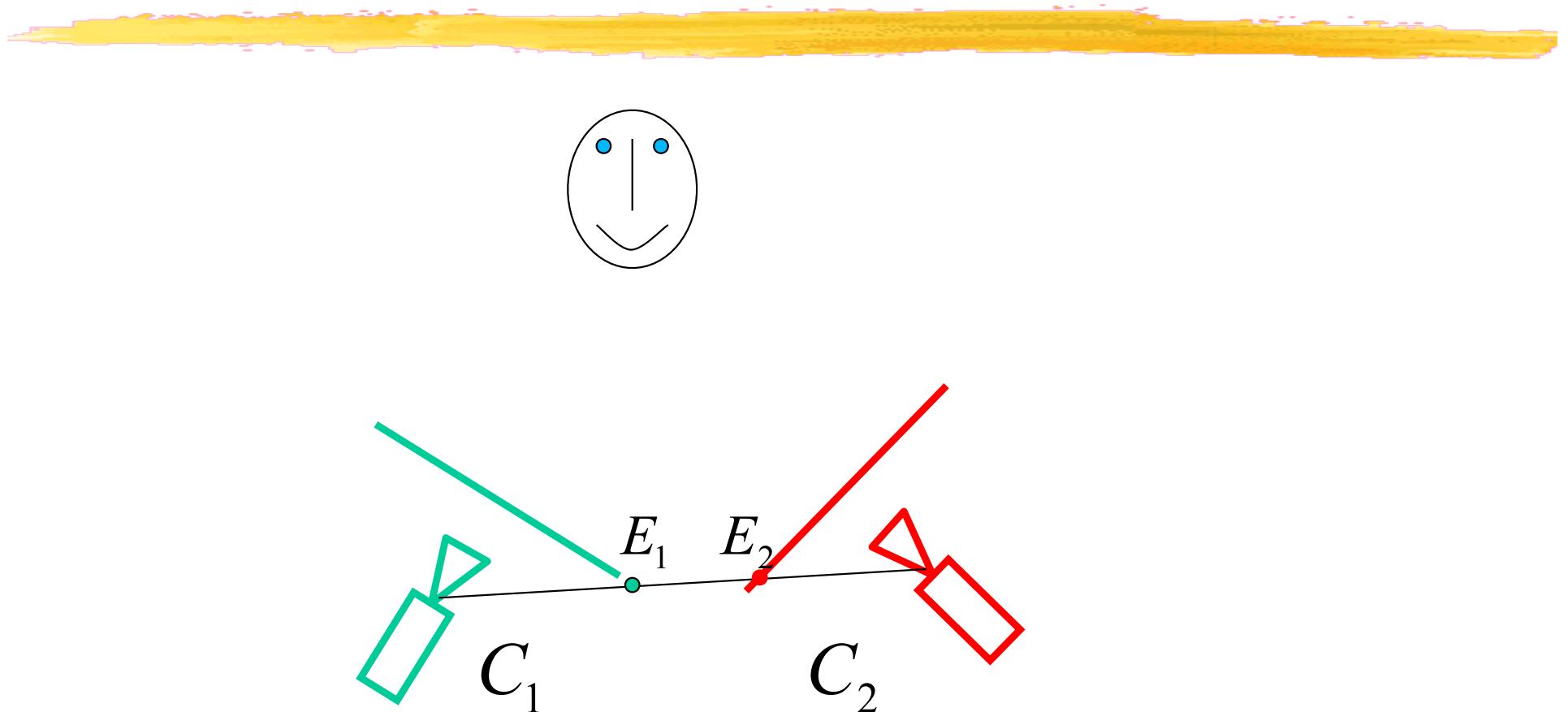
Three points shown as  
red crosses.

Corresponding epipolar  
lines.

# EPIPOLAR LINES



# EPIPOLE

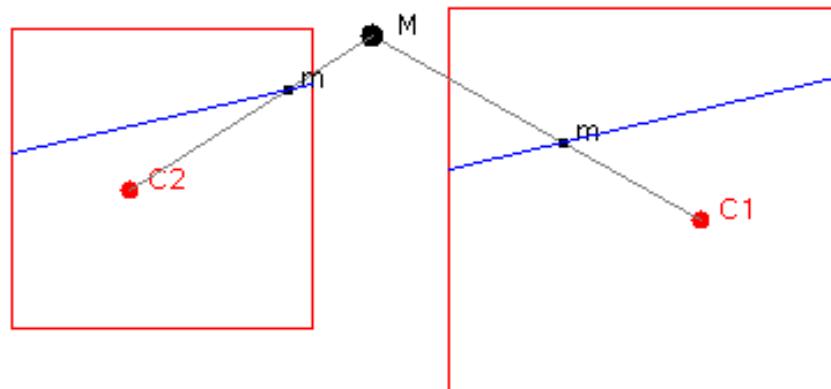
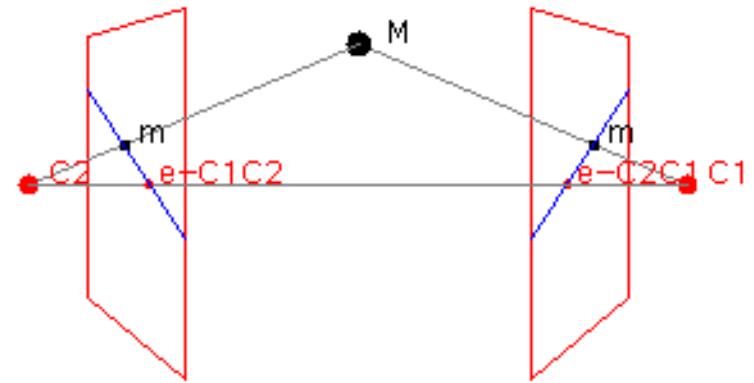


Point at which **all** epipolar lines intersect:

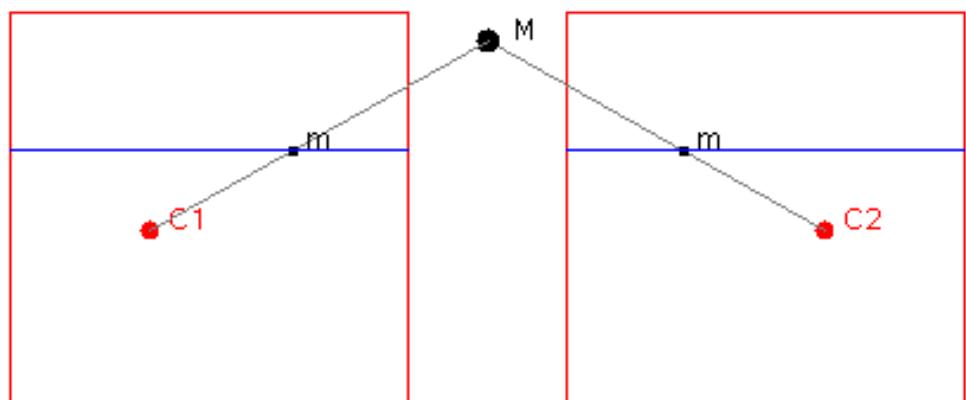
- ▶ Located at the intersection of line joining optical centers and image plane.

# EPIPOLAR GEOMETRY

In general:

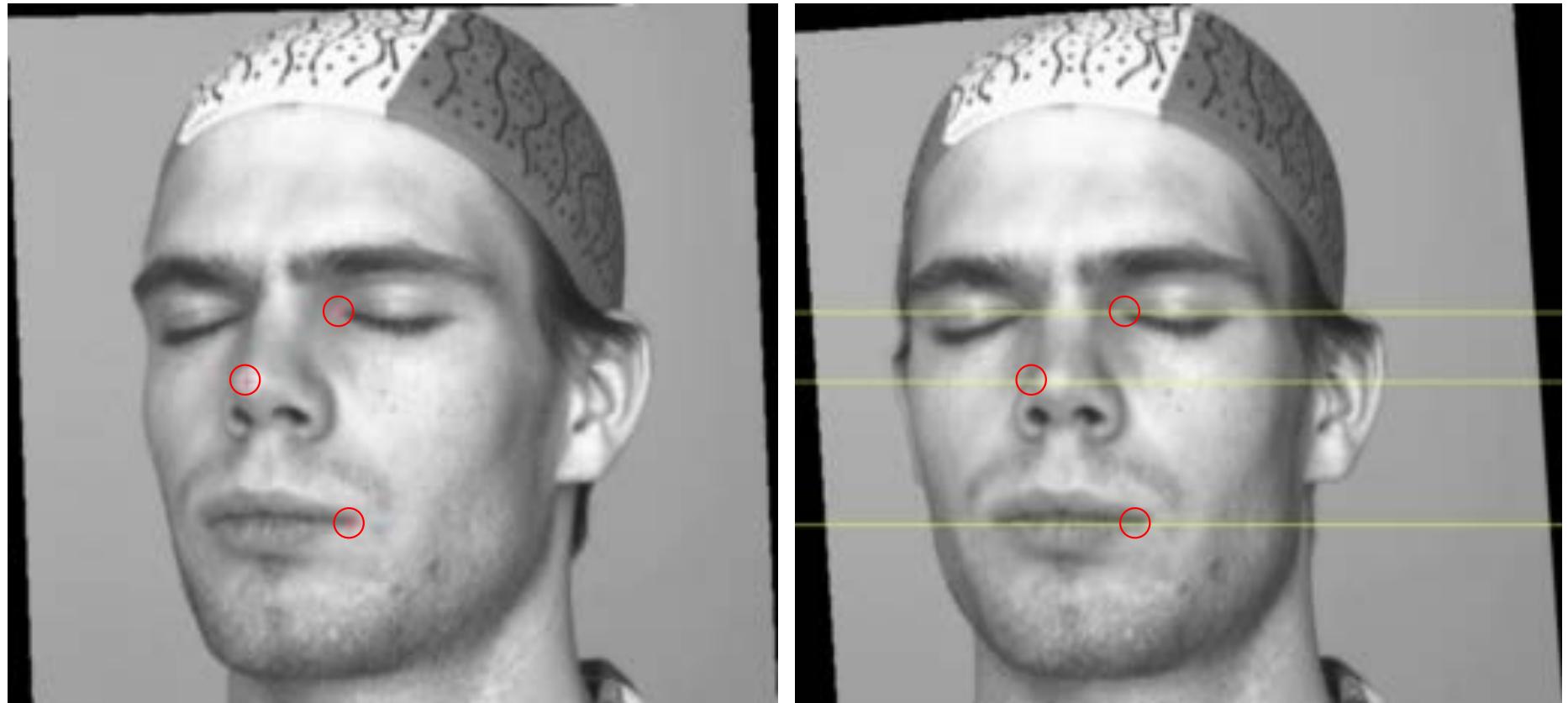


Parallel image planes



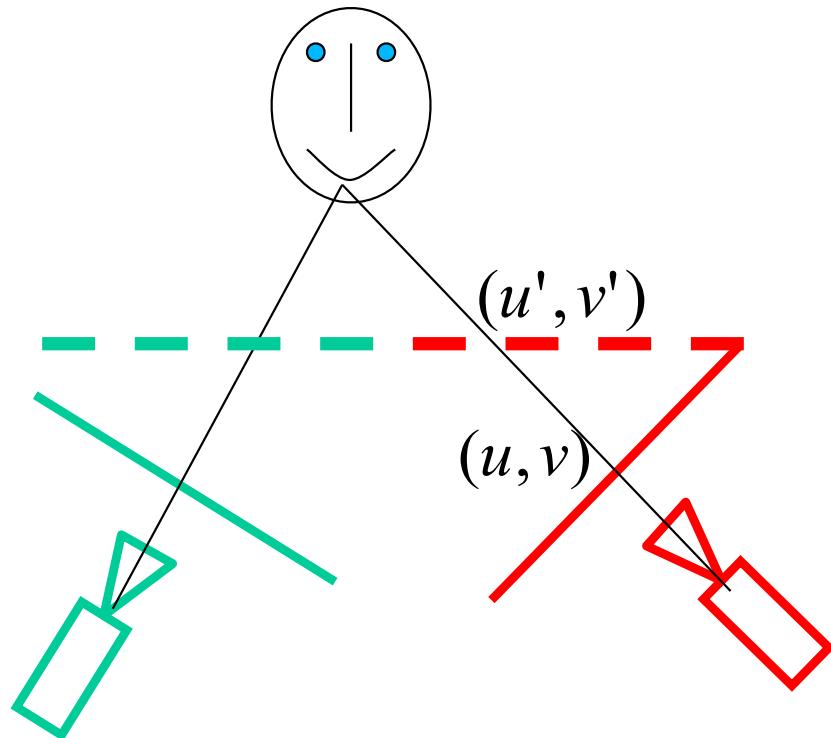
Horizontal baseline

# RECTIFICATION



Parallel epipolar lines

# RECTIFICATION



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

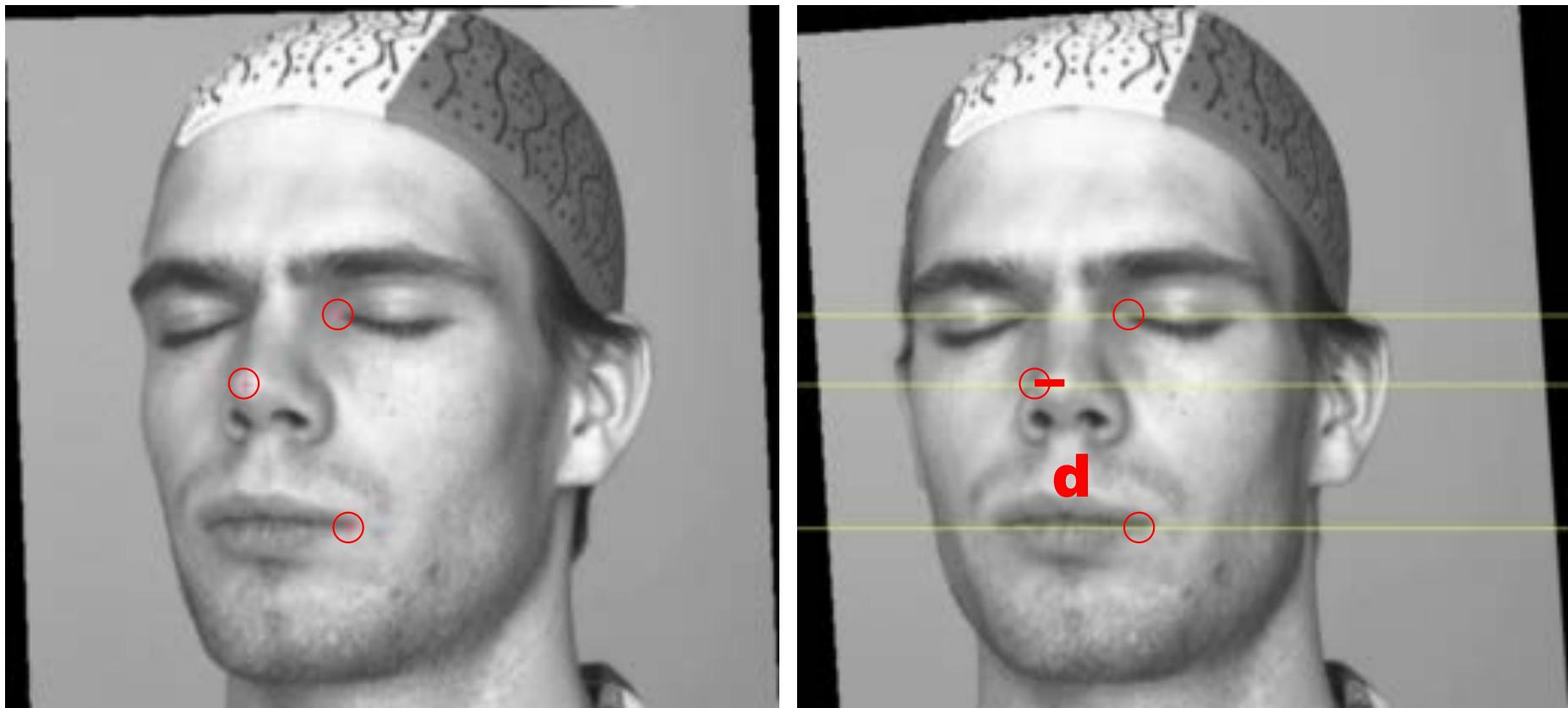
$$u' = U' / W'$$

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

Reprojection into parallel virtual image planes:

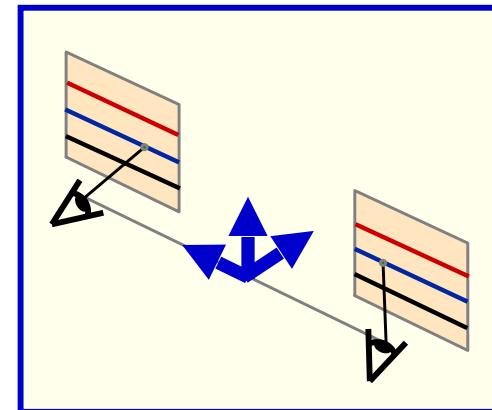
- Linear operation in projective coordinates
- Real-time implementation possible

# DISPARITY



Horizontal shift along 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



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

# FINDING A PATTERN IN AN IMAGE

Straightforward Approach



Pattern

Move pattern everywhere and compare with image.

But how?

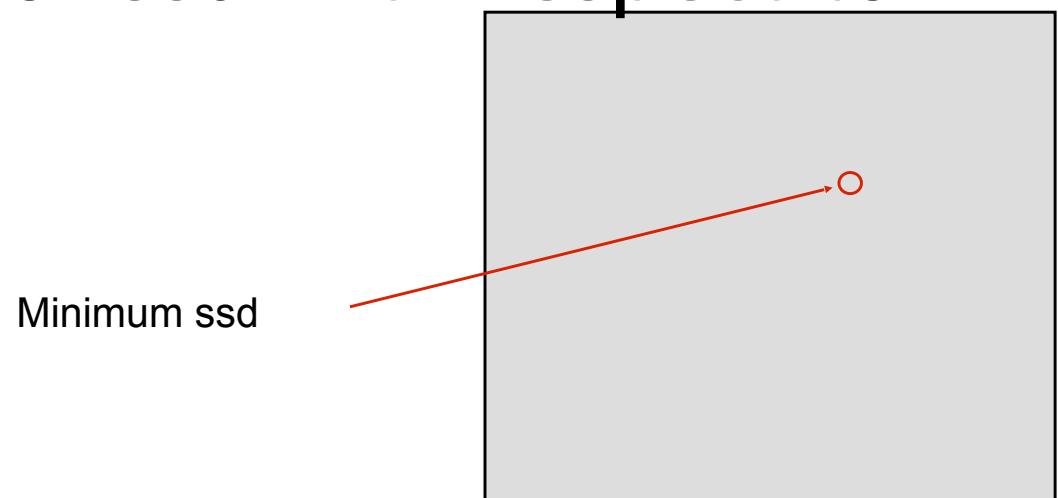
# SUM OF SQUARE DIFFERENCES

Subtract pattern and image pixel by pixel and add squares:

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

If identical  $ssd=0$ , otherwise  $ssd > 0$

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



# CORRELATION

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



Sum of squares  
of the window  
(positive term)



Sum of squares of  
the pattern  
(CONSTANT term)



Correlation

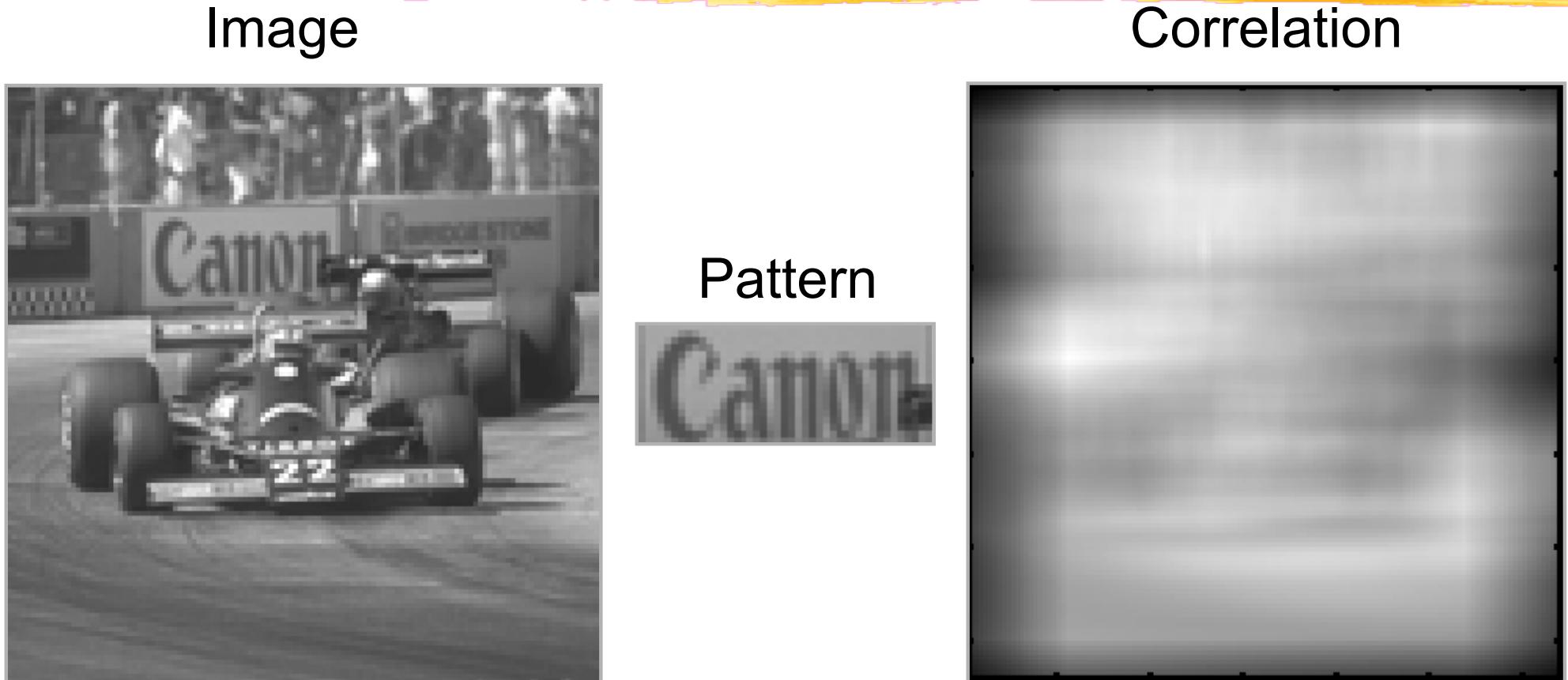
$ssd(u,v)$  is minimized when correlation is largest  
→ Correlation measures similarity

# SIMPLE EXAMPLE

The diagram illustrates a convolution operation. On the left, a large black square labeled **I** contains a smaller white square. In the center, a smaller black square labeled **P** contains a white square. Between them is a multiplication symbol ( $*$ ). To the right of the multiplication symbol is an equals sign (=). To the right of the equals sign is a large black square with a bright, localized white spot in the upper-right quadrant, labeled **I correlated with P**.

$$I \star P = I \text{ correlated with } P$$

# NOT SO SIMPLE EXAMPLE



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

# NORMALIZED CROSS CORRELATION

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

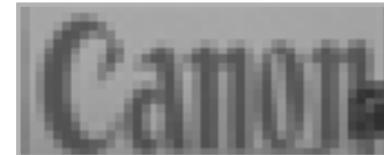
- Between -1 and 1
- Invariant to linear transforms
- Independent of the average gray levels of the pattern and the image window

# NORMALIZED EXAMPLE

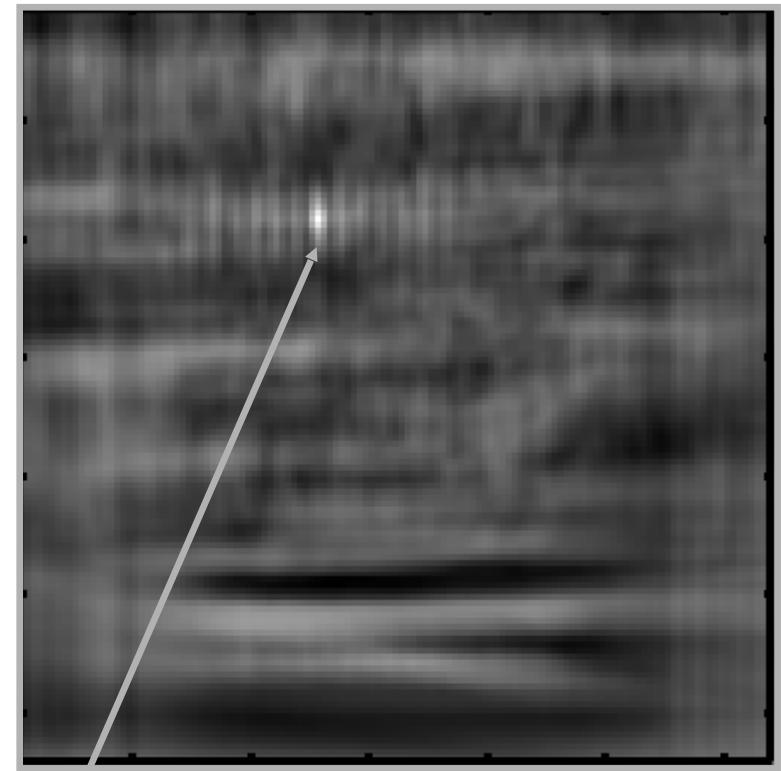
Image



Pattern

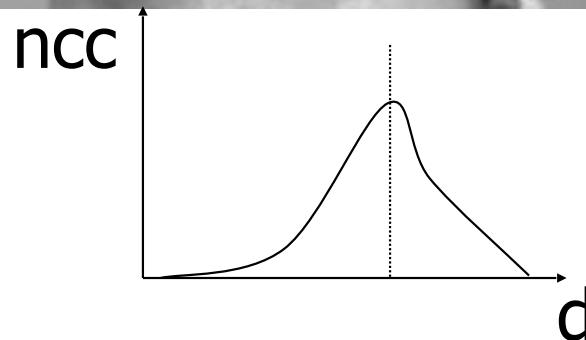
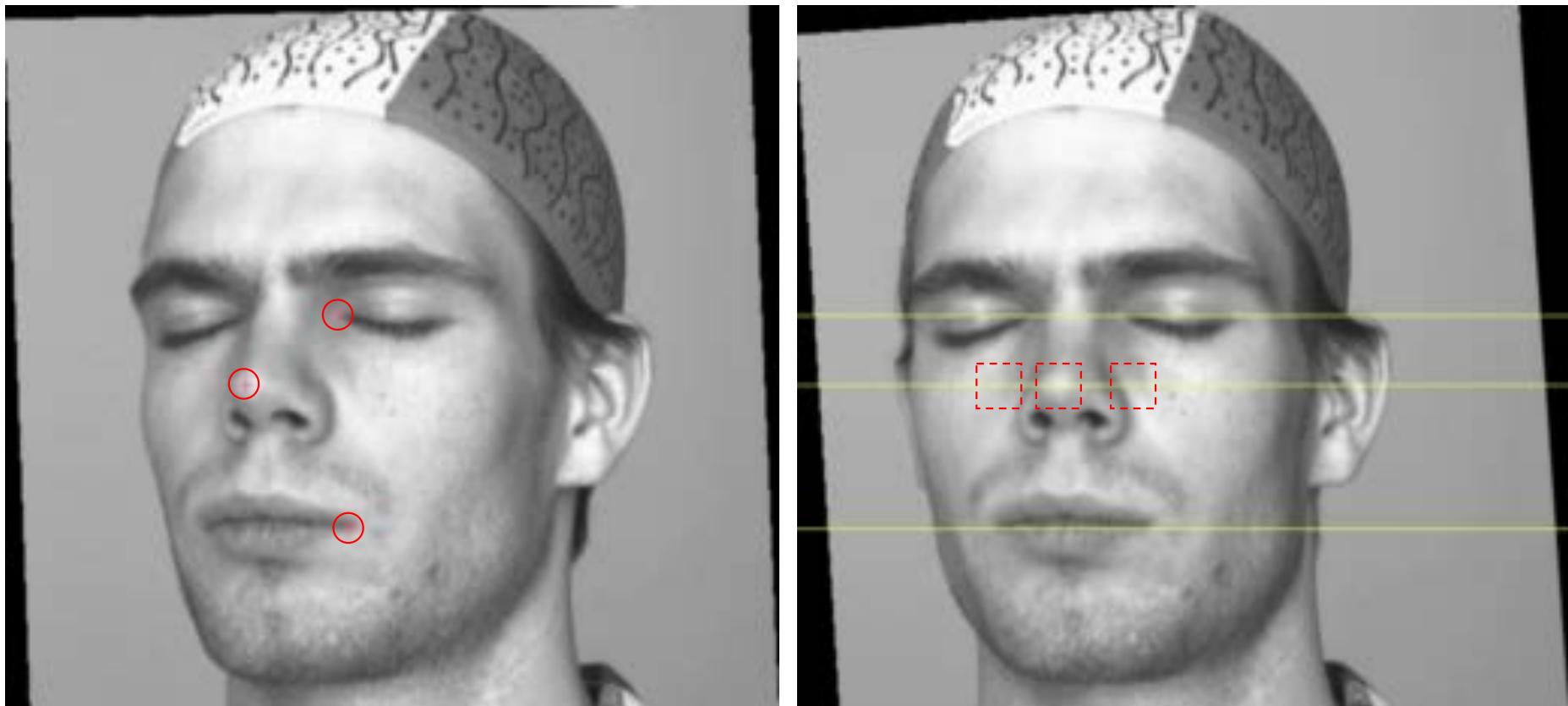


Normalized Correlation

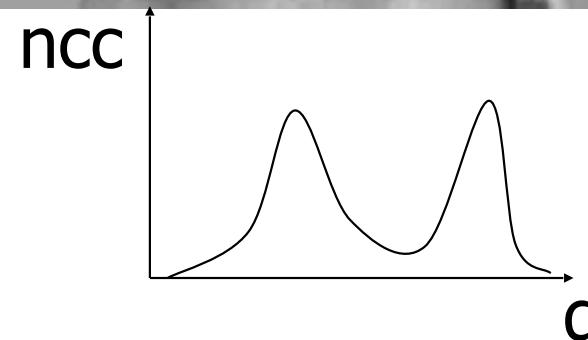


Point of maximum correlation

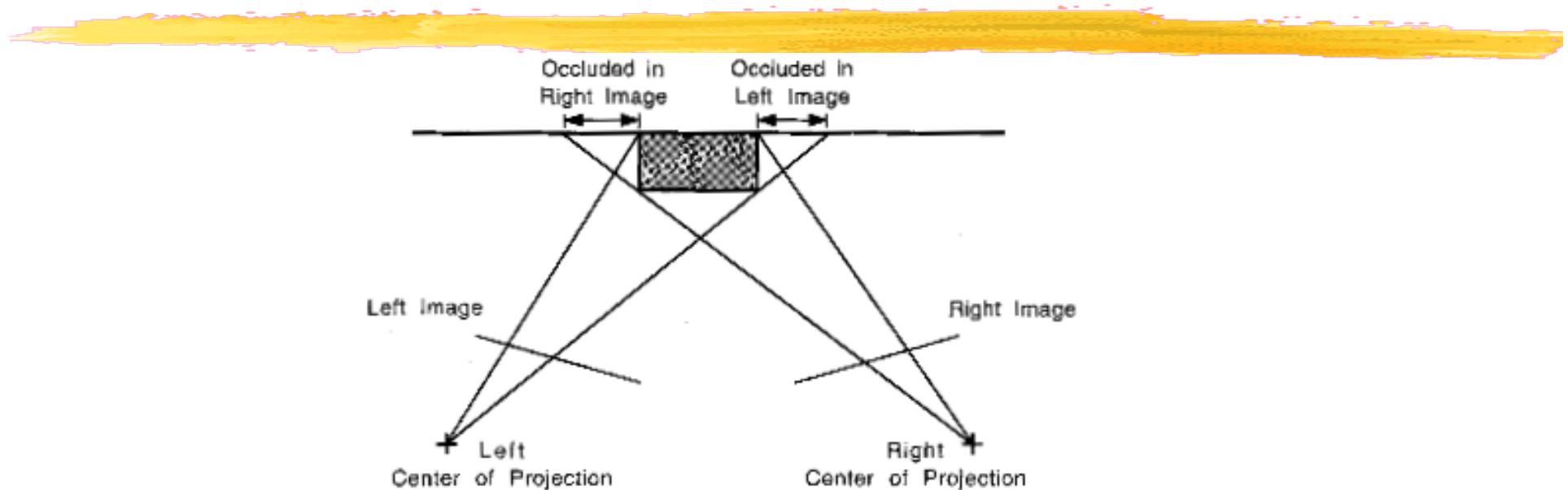
# SEARCHING ALONG EPIPOLAR LINES



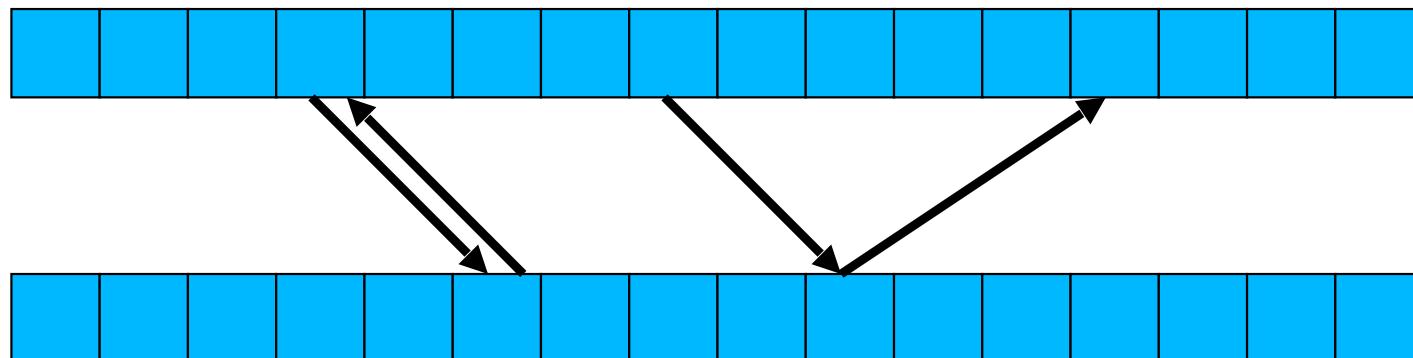
or



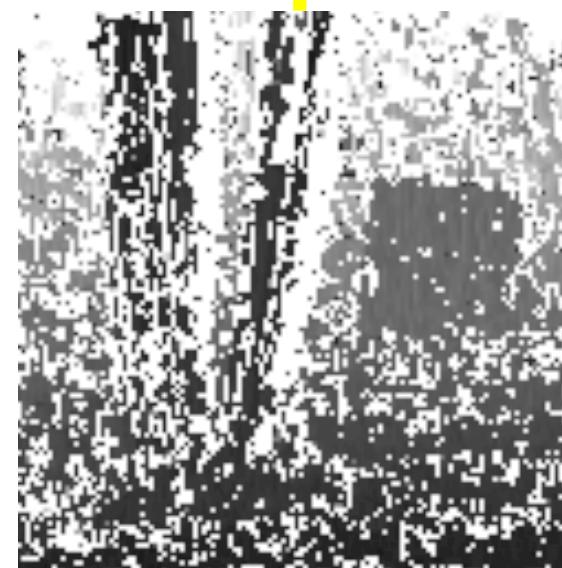
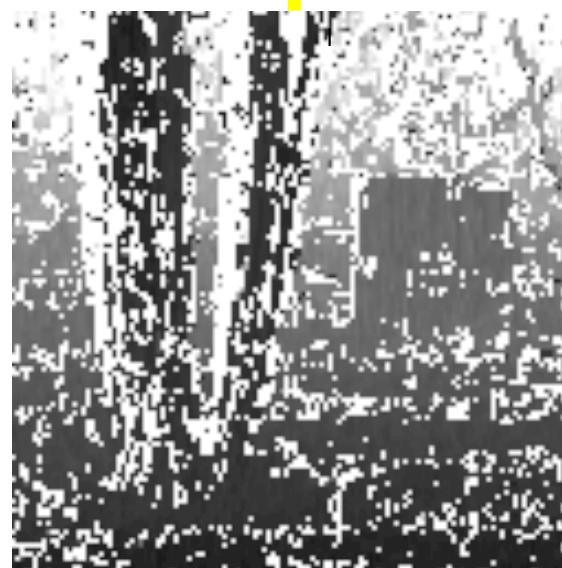
# OCCLUSIONS



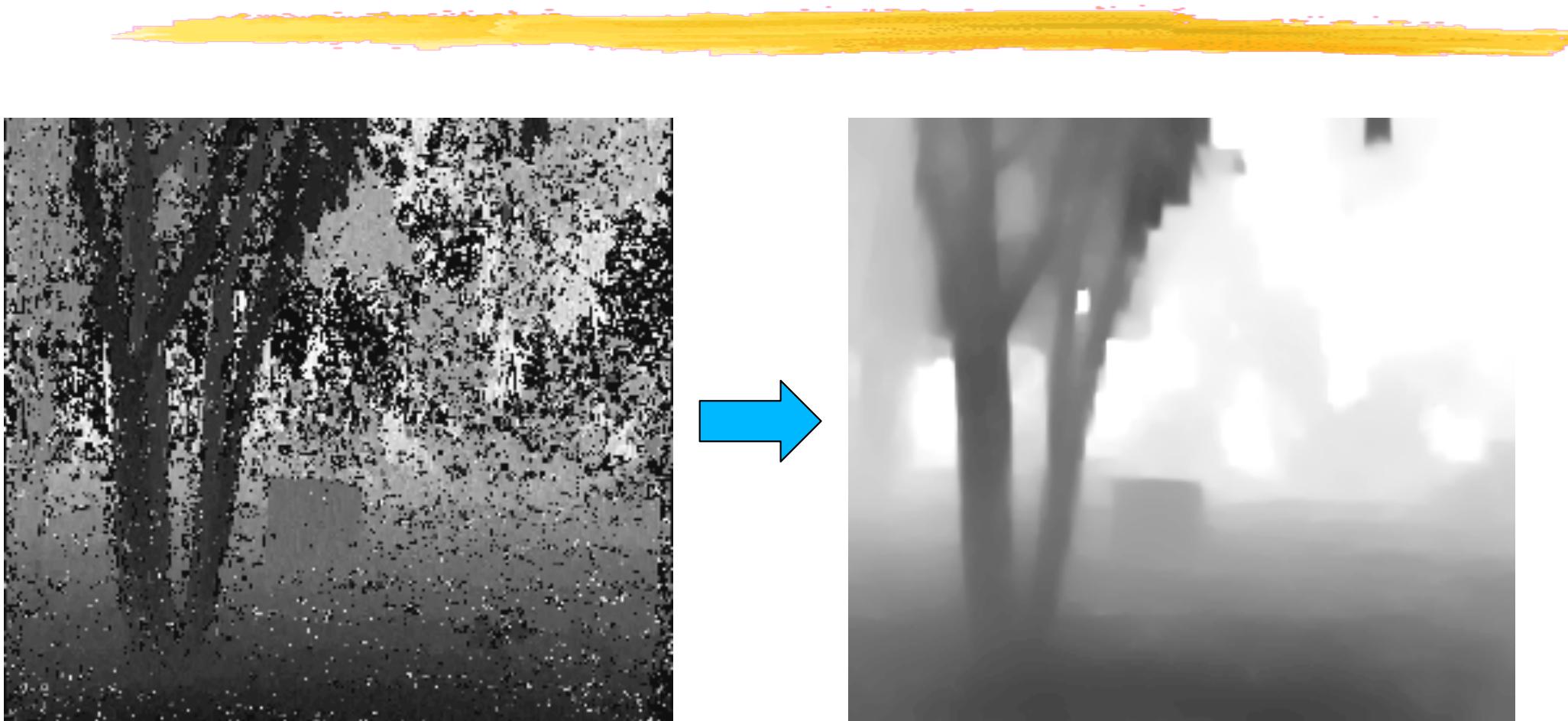
→ Consistency test



# GROUND LEVEL STEREO

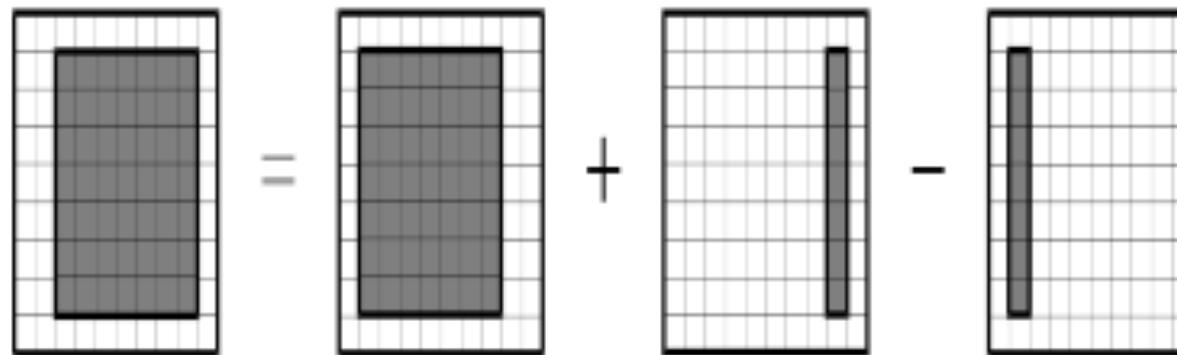


# COMBINING DISPARITY MAPS



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

# REAL-TIME IMPLEMENTATION



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

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

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

# VARIATIONAL APPROACH

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



$s$  = Correlation score if  $w_0$  has been measured, 0 otherwise.

$$\lambda_x = c_x f\left(\frac{\partial I}{\partial x}\right)$$

$$\lambda_y = c_y f\left(\frac{\partial I}{\partial y}\right)$$

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

# DISCRETIZATION

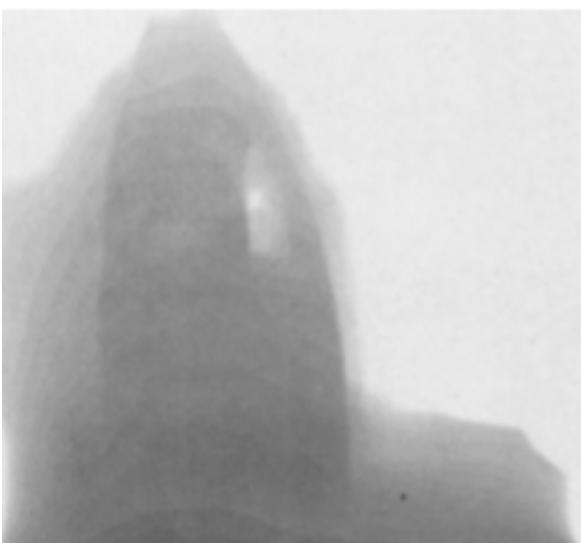
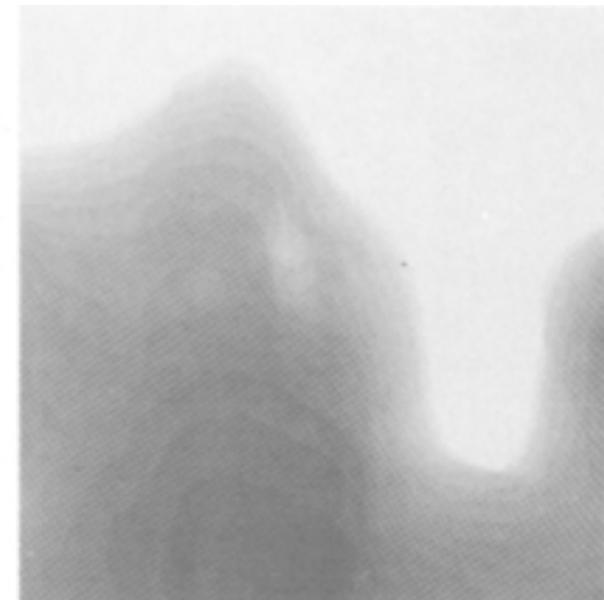
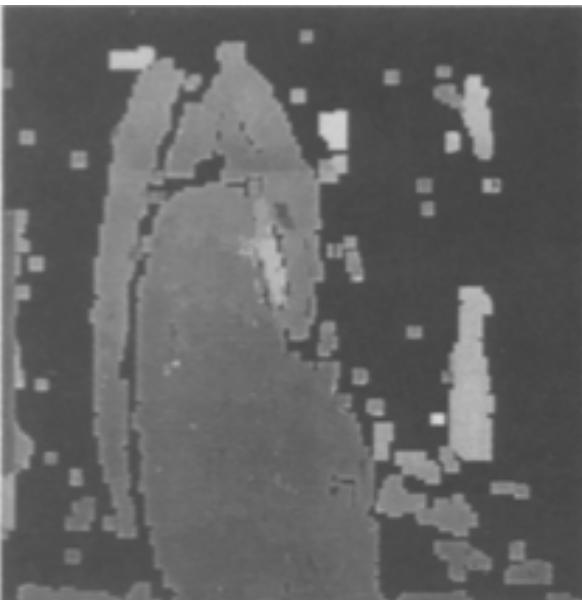


$$\begin{aligned}\mathcal{C} &= \sum_{ij} s_{ij} (w_{ij} - w_{0ij})^2 + \lambda_x \sum_{ij} (w_{i+1,j} - w_{i,j})^2 + \lambda_y \sum_{ij} (w_{i,j+1} - w_{i,j})^2 \\ &= (W - W_0)^t S (W - W_0) + W^t K W\end{aligned}$$

$$\Rightarrow \frac{\partial \mathcal{C}}{\partial W} = 0$$

$$\Rightarrow (K + S)W = SW_0$$

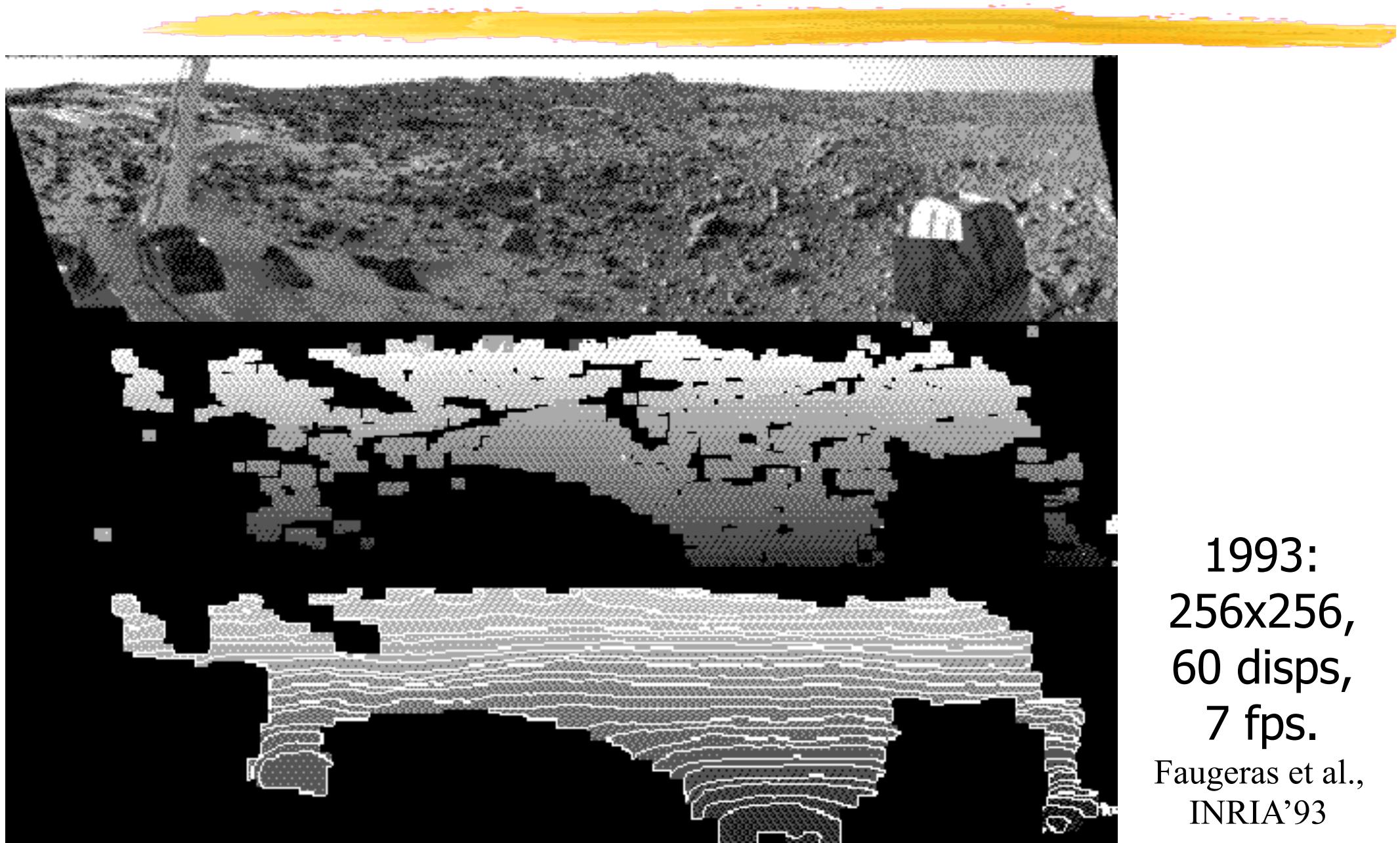
# PRESERVING DISCONTINUITIES



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

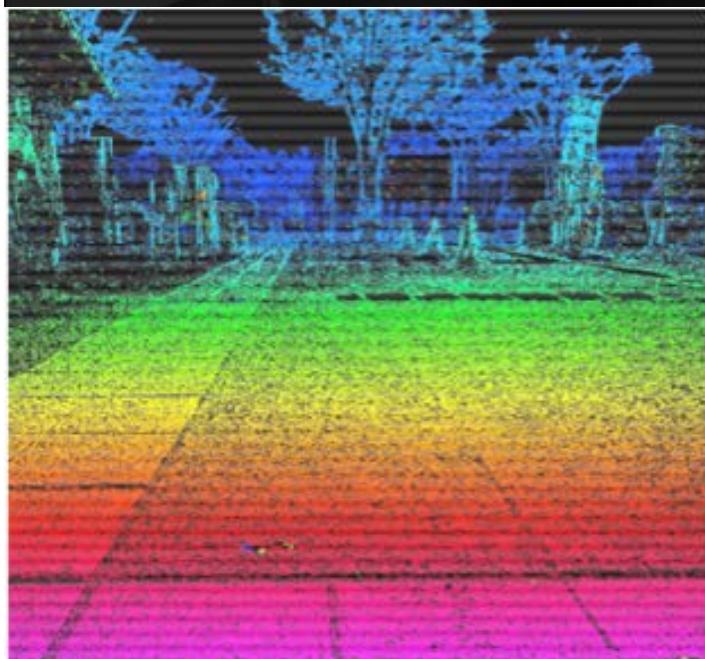
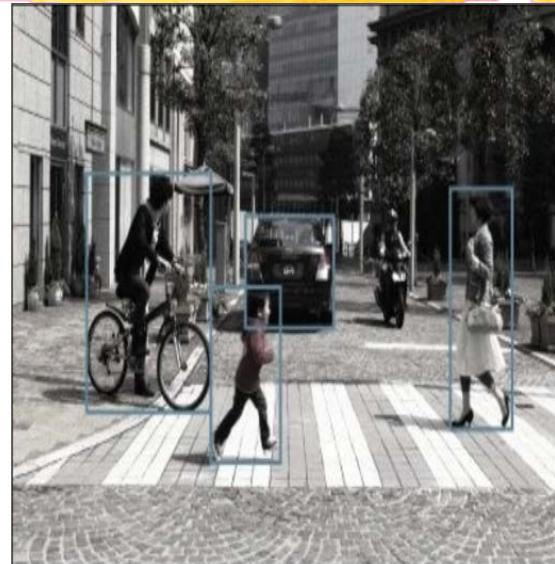
$$\lambda_y = f\left(\frac{\partial I}{\partial y}\right) f\left(\frac{\partial w}{\partial y}\right)^2$$

# THEN ....



1993:  
256x256,  
60 disps,  
7 fps.  
Faugeras et al.,  
INRIA'93

# ... AND MORE RECENTLY



Subaru's EyeSight System

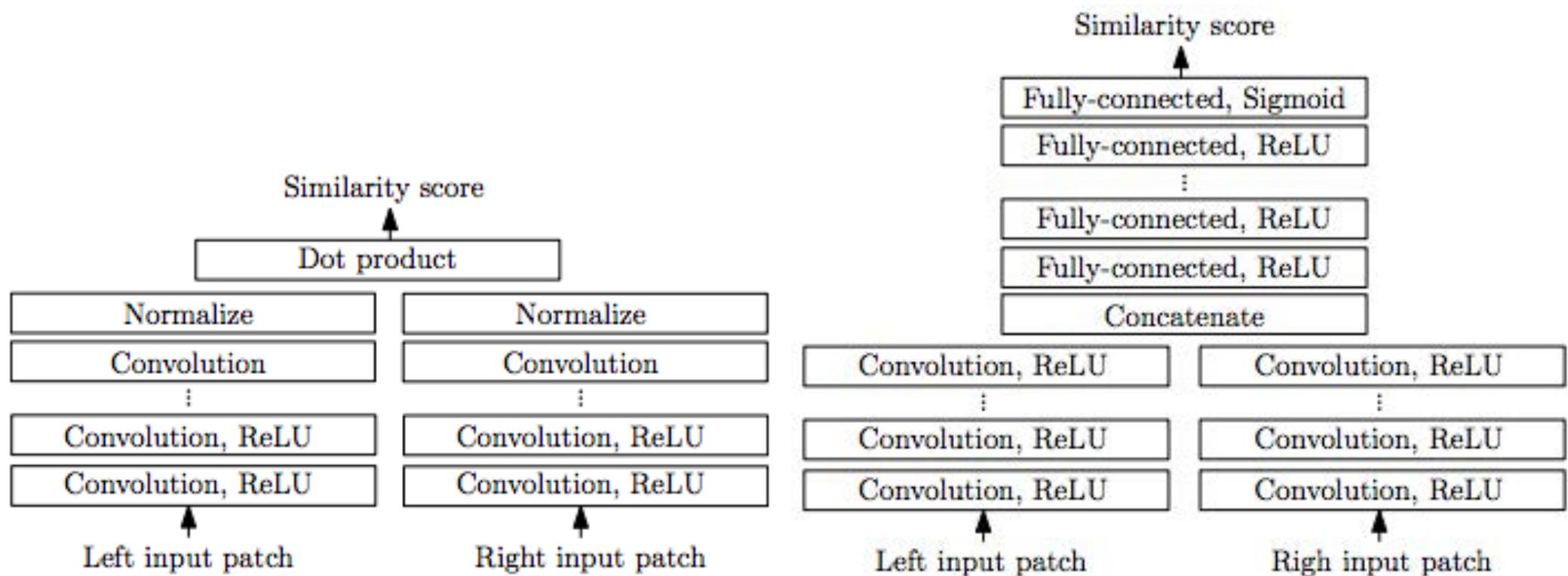
<http://www.gizmag.com/subaru-new-eyesight-stereoscopic-vision-system/14879/>

2011:  
1312x688,  
176 disps,  
160 fps.

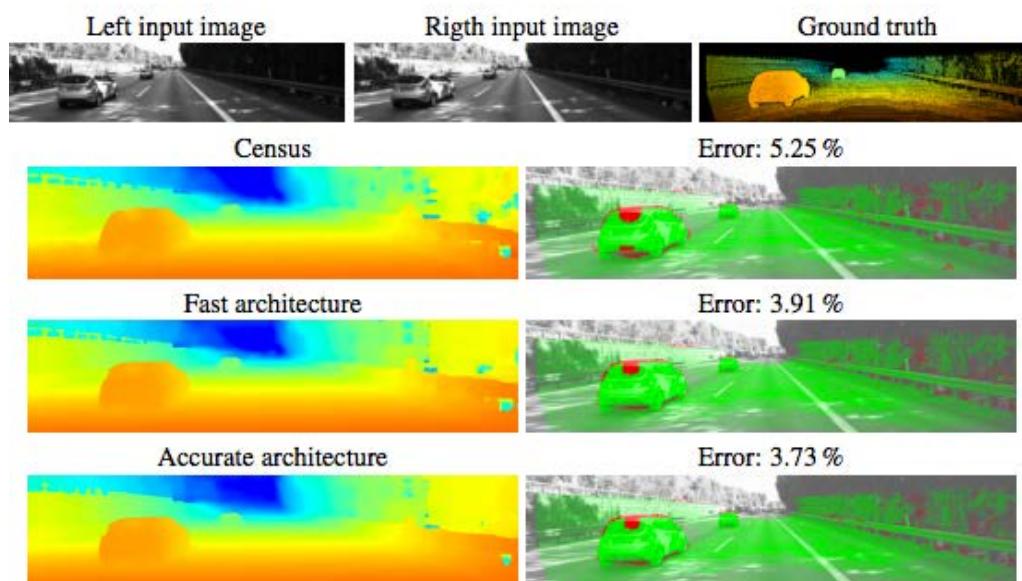
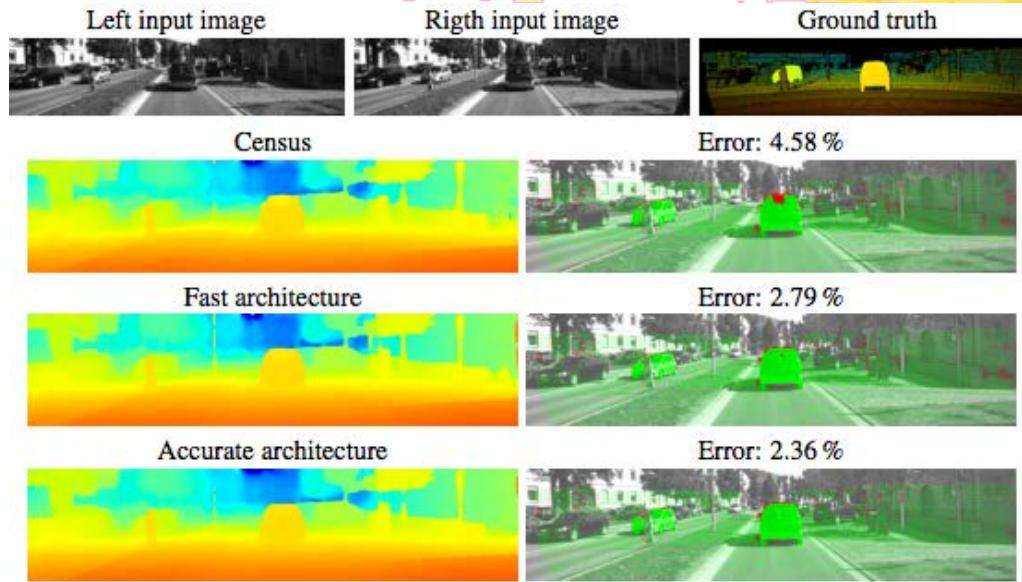
Saneyoshi, CMVA'11

**... AND EVEN MORE RECENTLY**

Train Siamese nets to return a similarity score.



# 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.

# XKCD'S TAKE ON DEEP LEARNING



# 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

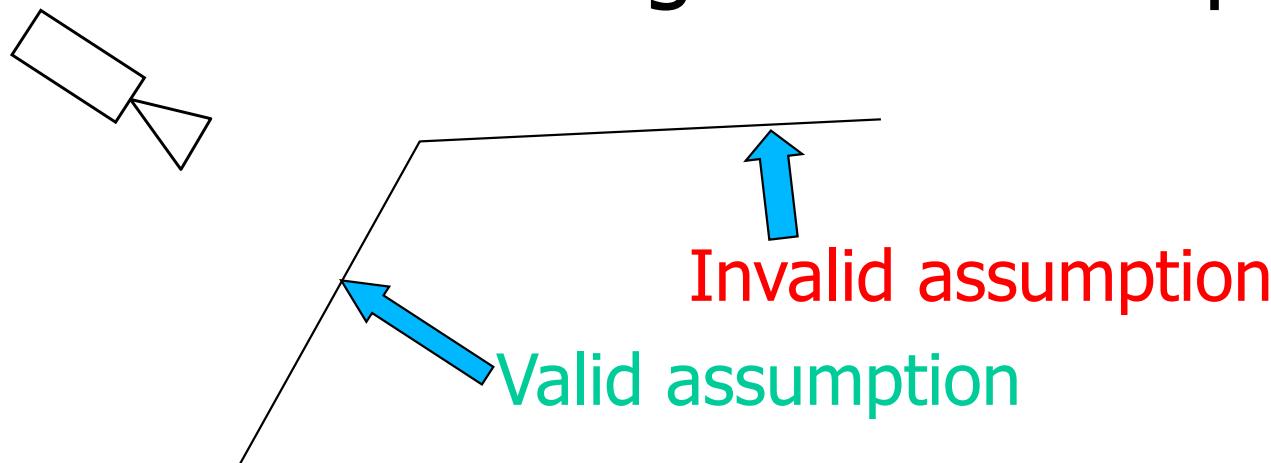


Difference of  
Gaussians

- 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 in the whole correlation window, which is equivalent to assuming constant depth.



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

# MULTI-VIEW STEREO



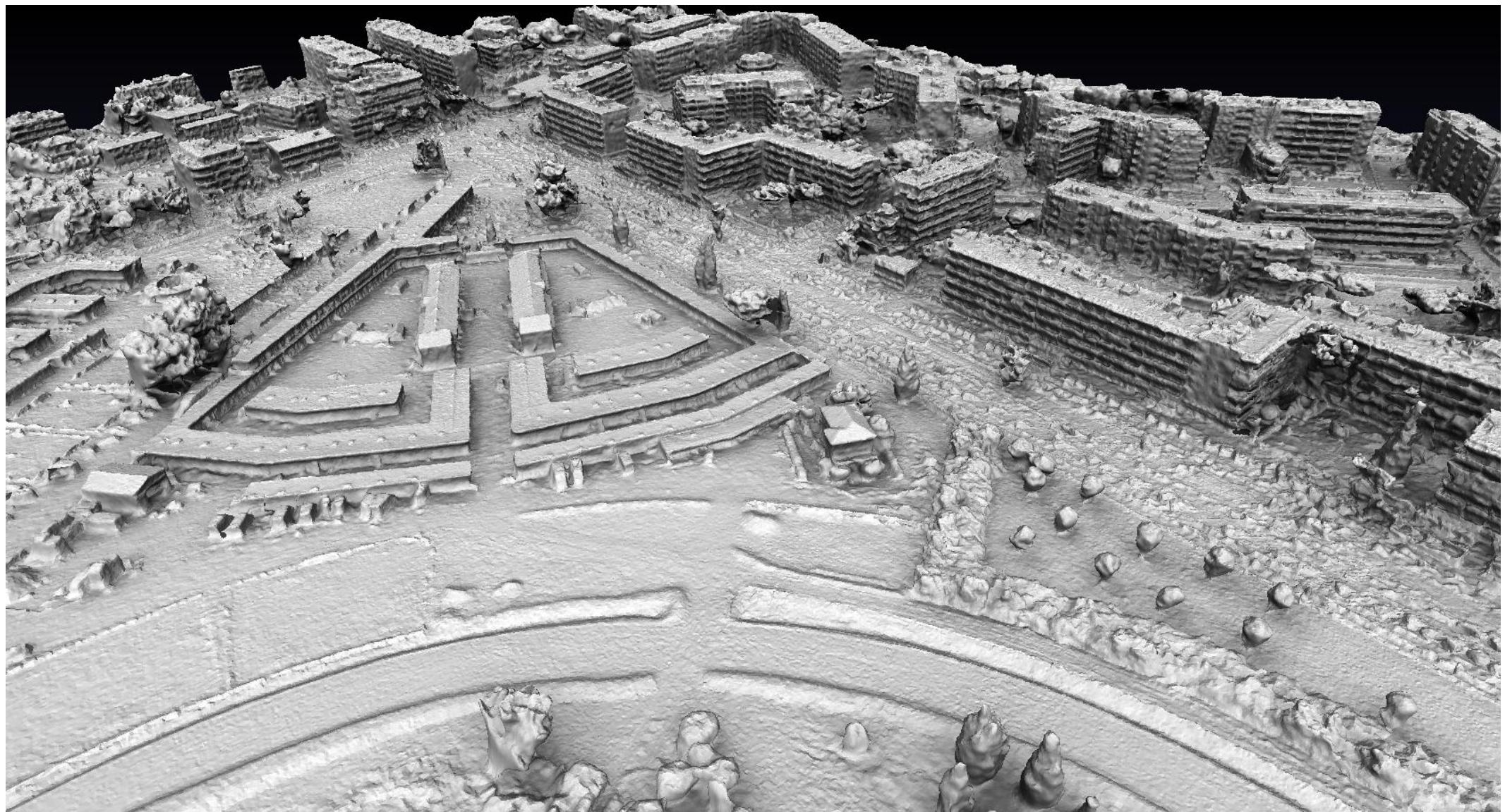
Multi-view reconstruction setup

- Adjust correlation window shapes to handle orientation.



Textured Mapped 3D Model

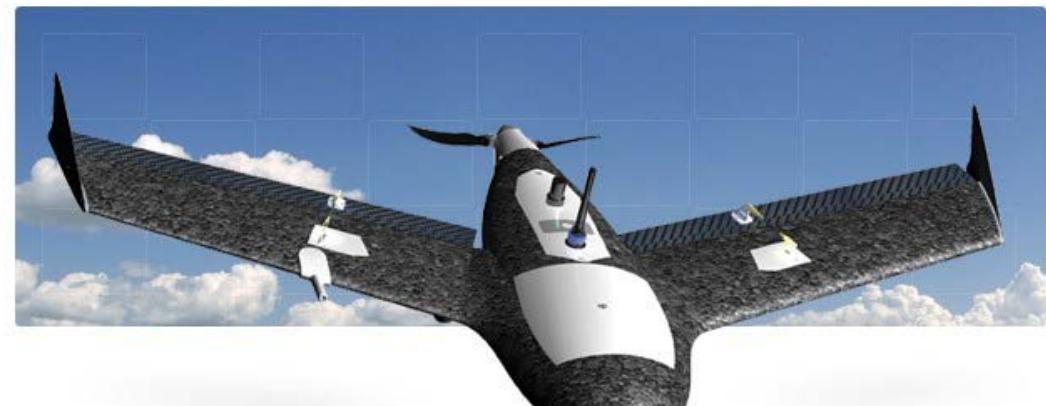
# MULTI-VIEW STEREO



# SMALL DRONES



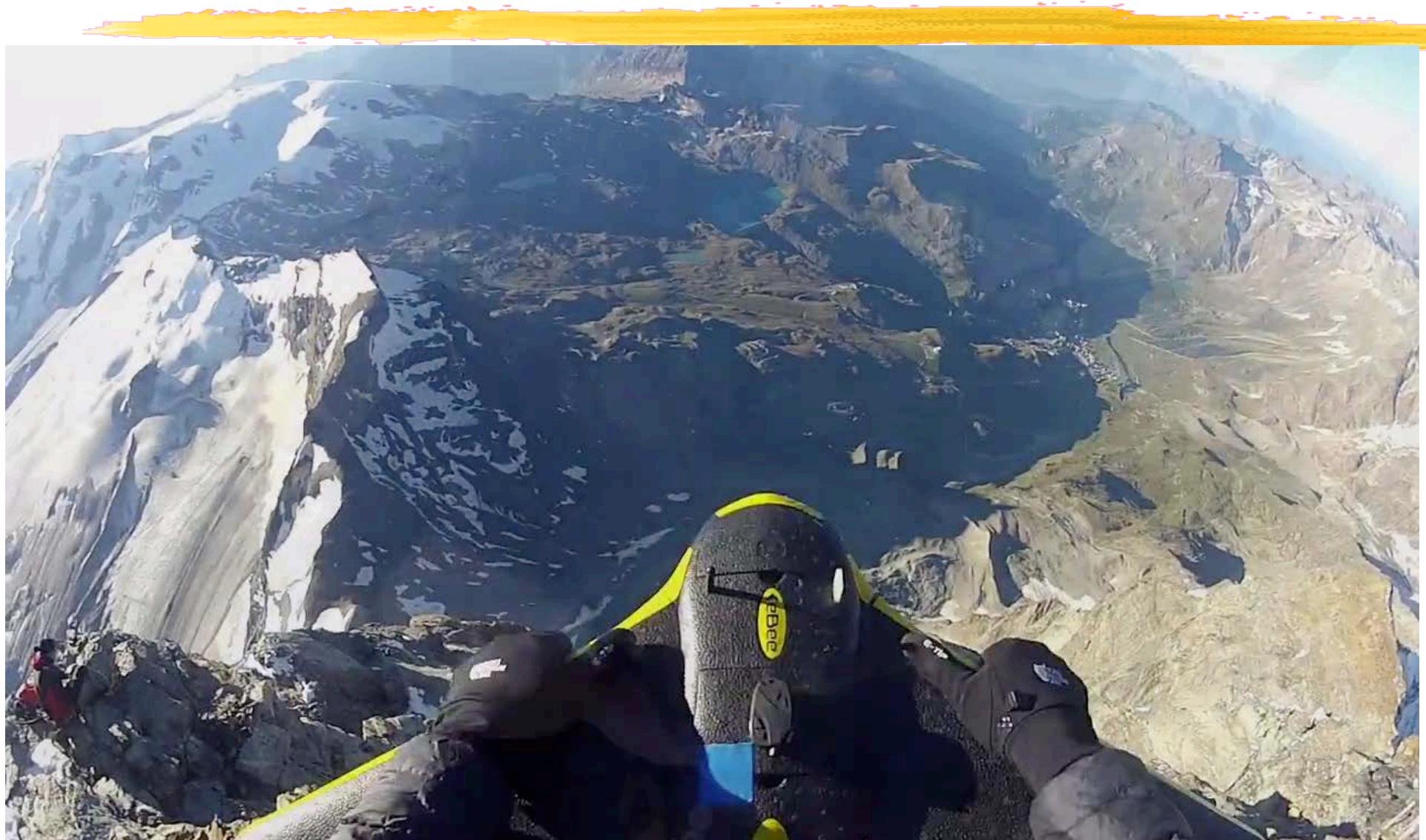
SenseFly:  
[www.sensefly.com](http://www.sensefly.com)



The X100  
revolutionary mapping.  
PATENT PENDING

Gatewing:  
[www.gatewing.com](http://www.gatewing.com)

# MATTERHORN



Drone: [www.sensefly.com](http://www.sensefly.com)

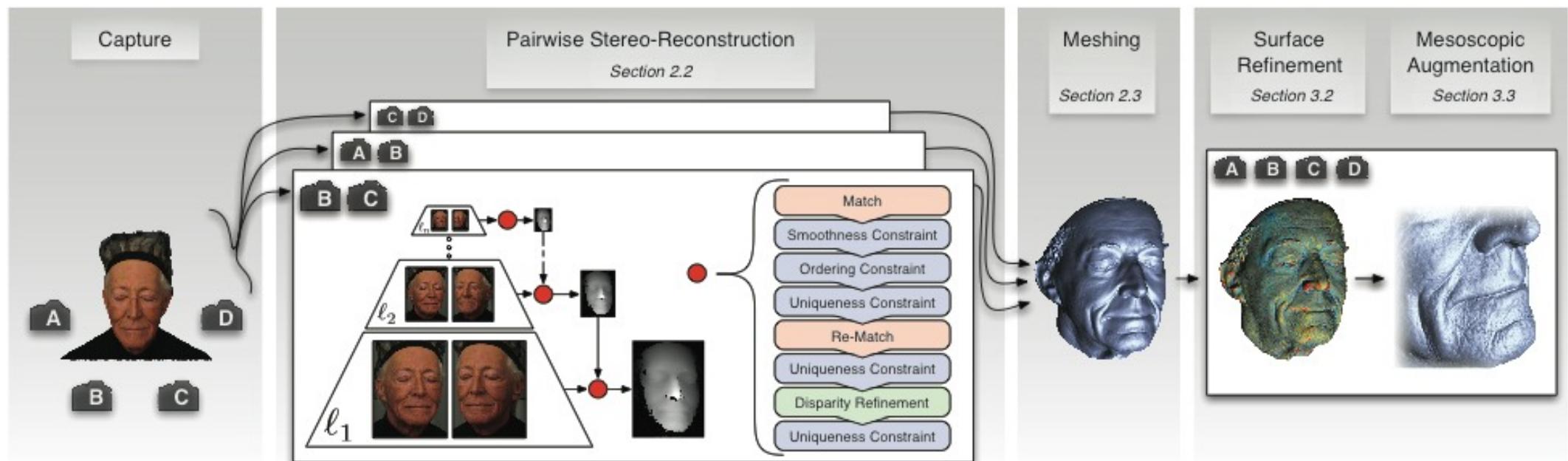
Mapping: [www.pix4d.com](http://www.pix4d.com)

# FACE RECONSTRUCTION



Beeler et al. SIGGRAPH'10

# FACE RECONSTRUCTION



# DYNAMIC SHAPE



## Lightweight Binocular Facial Performance Capture under Uncontrolled Lighting

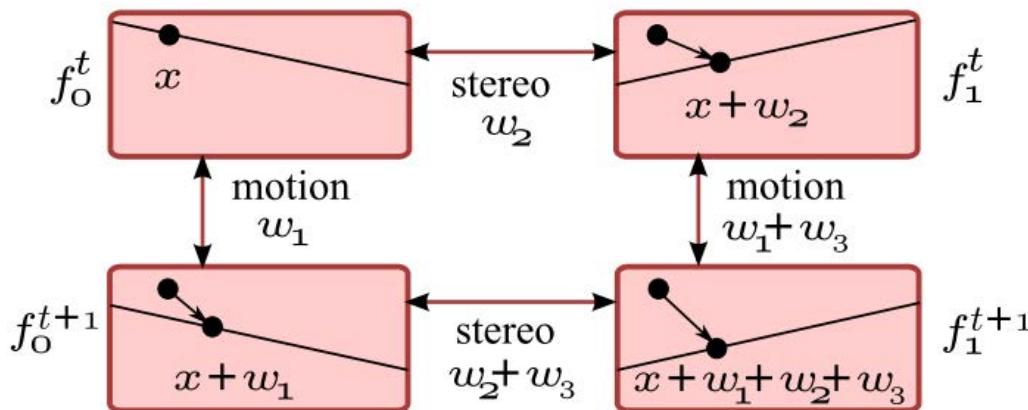
Levi Valgaerts<sup>1</sup>   Chenglei Wu<sup>1,2</sup>   Andrés Bruhn<sup>3</sup>  
Hans-Peter Seidel<sup>1</sup>   Christian Theobalt<sup>1</sup>

<sup>1</sup> MPI for Informatics

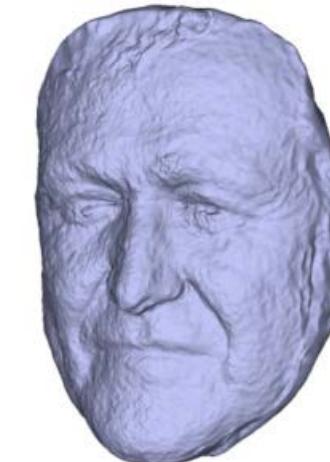
<sup>2</sup> Intel Visual Computing Institute

<sup>3</sup> University of Stuttgart

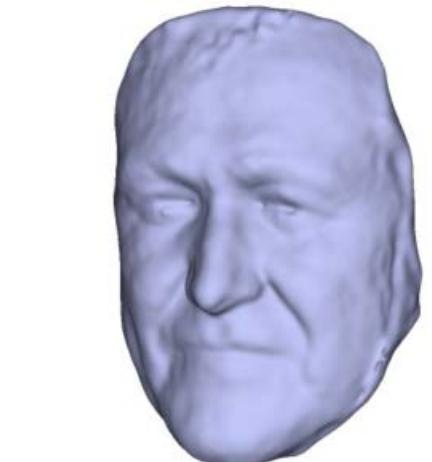
# SCENE FLOW



Correspondences across cameras and across time

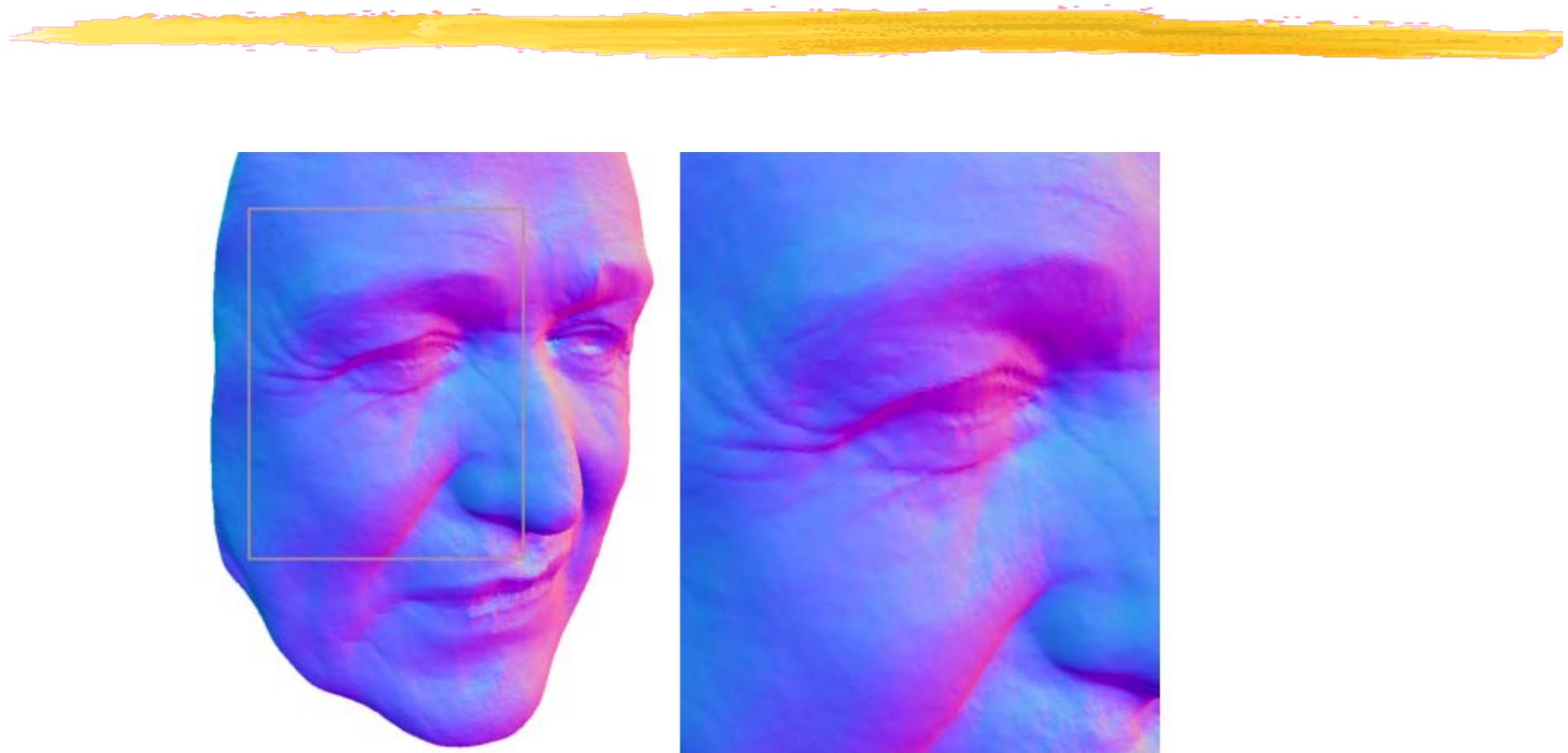


Stereo Only



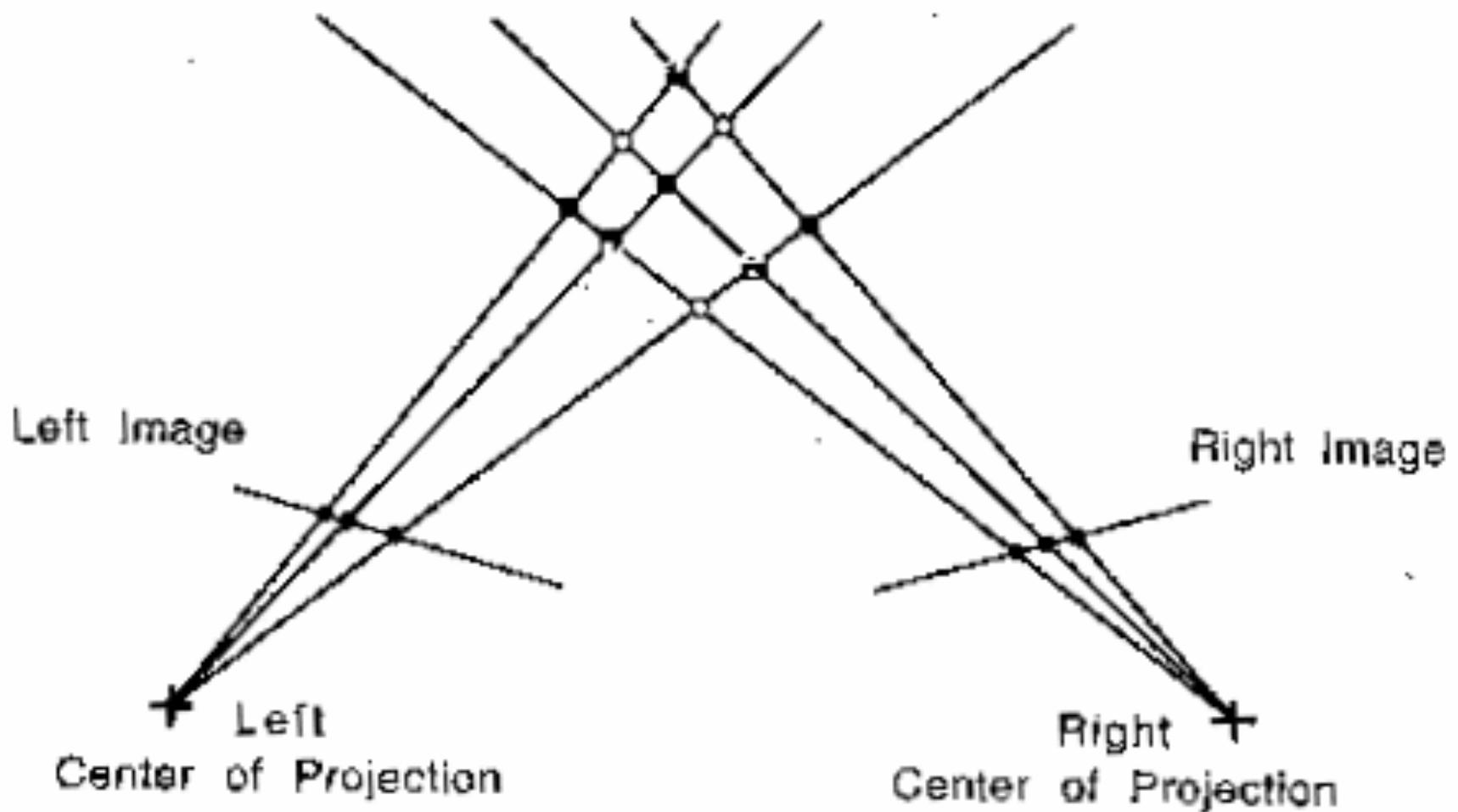
Stereo + Flow

# SHAPE FROM SHADING

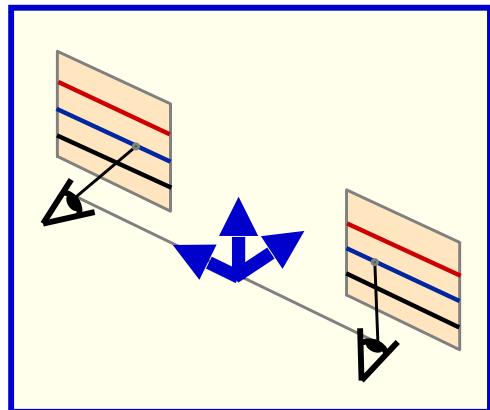


Shape-from-shading is used to refine the shape and provide high-frequency details.

# UNCERTAINTY



# PRECISION vs BASELINE



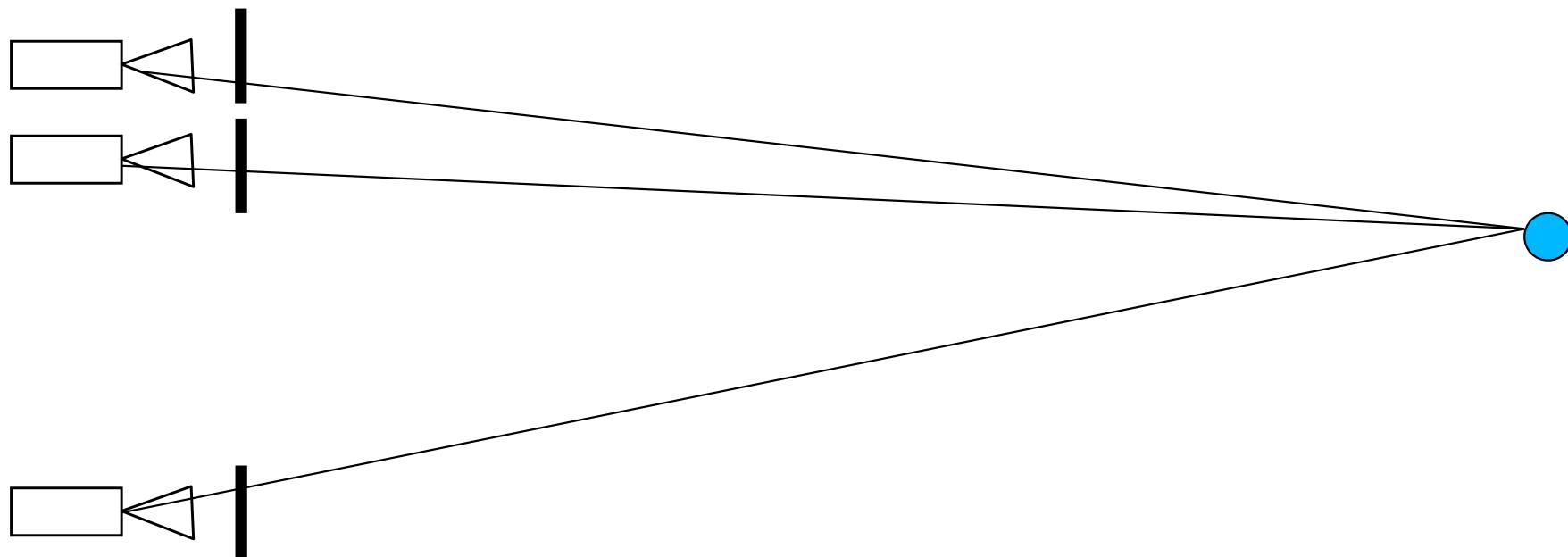
$$d = f \frac{b}{Z}$$

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

$$\Rightarrow \frac{\partial Z}{\partial d} = -f \frac{b}{d^2} = -\frac{Z^2}{fb}$$

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

# SHORT vs LONG BASELINE



Long baseline:

- Harder to match
- More occlusions
- Better precision

Short baseline:

- Good matches
- Few occlusions
- Poor precision

# MARS ROVER



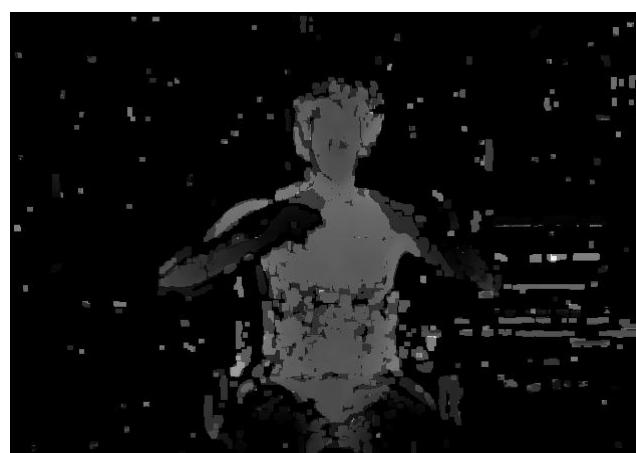
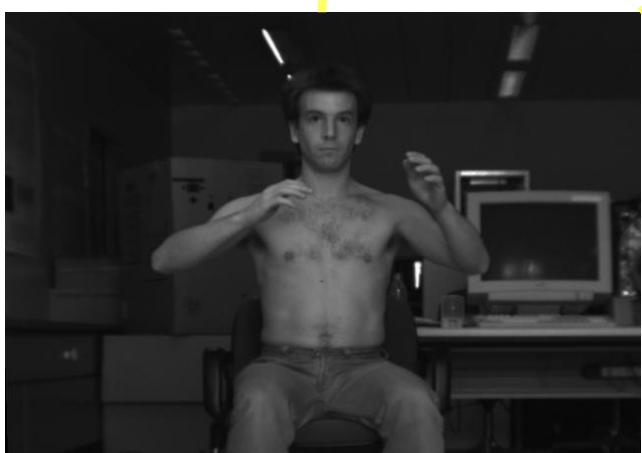
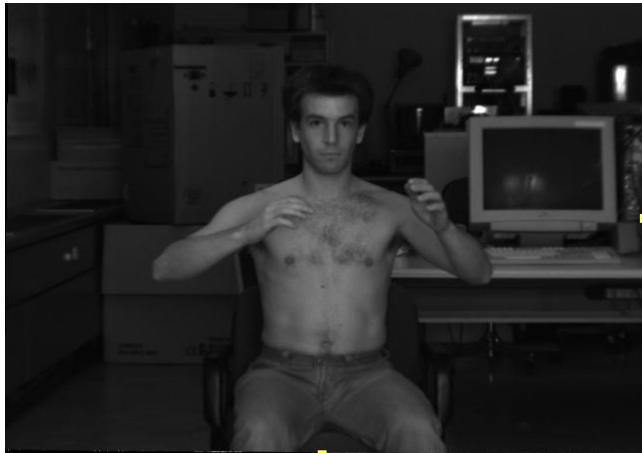
There are four cameras!

# VIDEO-BASED MOTION CAPTURE

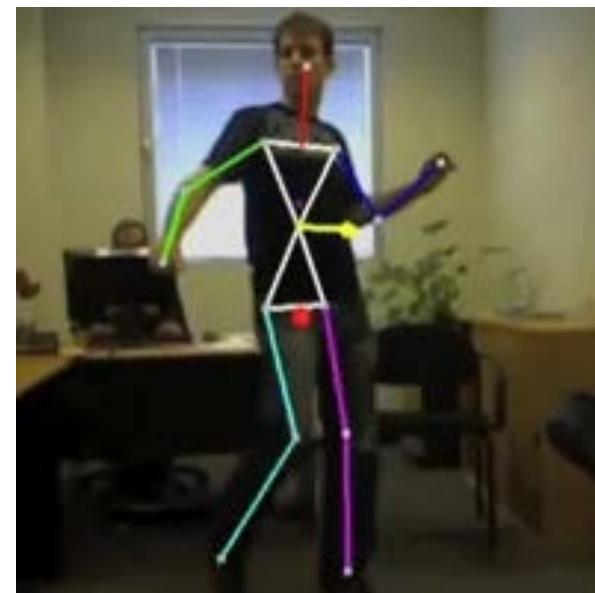


Fitting an articulated body model to stereo data.

# TRINOCULAR STEREO

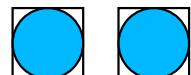


# KINECT: STRUCTURED LIGHT

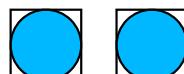
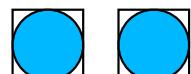


- The Kinect camera projects a IR pattern and measures depth from its distortion.
- Same principle but the second camera is replaced by the projector.

# MULTI-CAMERA CONFIGURATIONS



3 cameras give both robustness and precision



4 cameras give additional redundancy



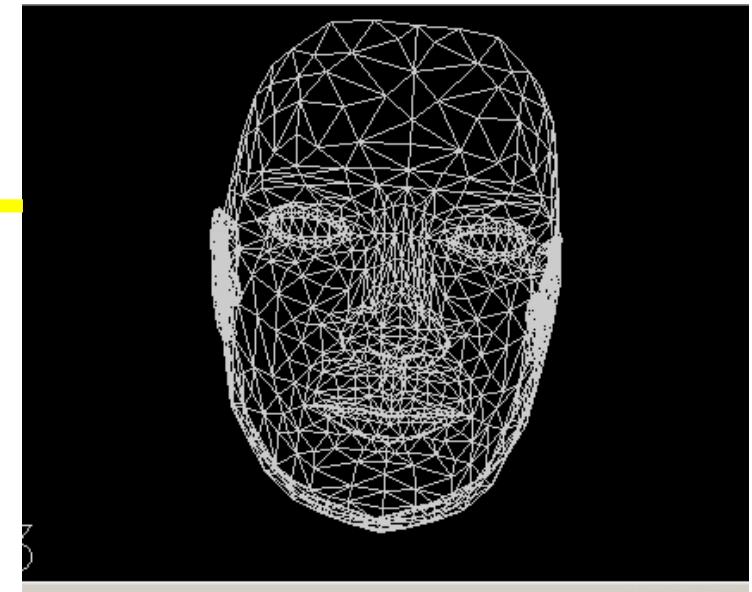
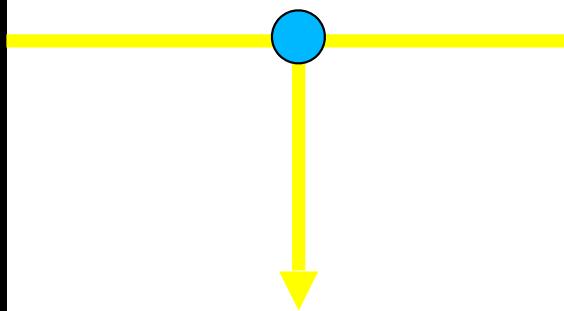
3 cameras in a T arrangement allow the system to see vertical lines.

# FACES FROM LOW-RESOLUTION VIDEOS

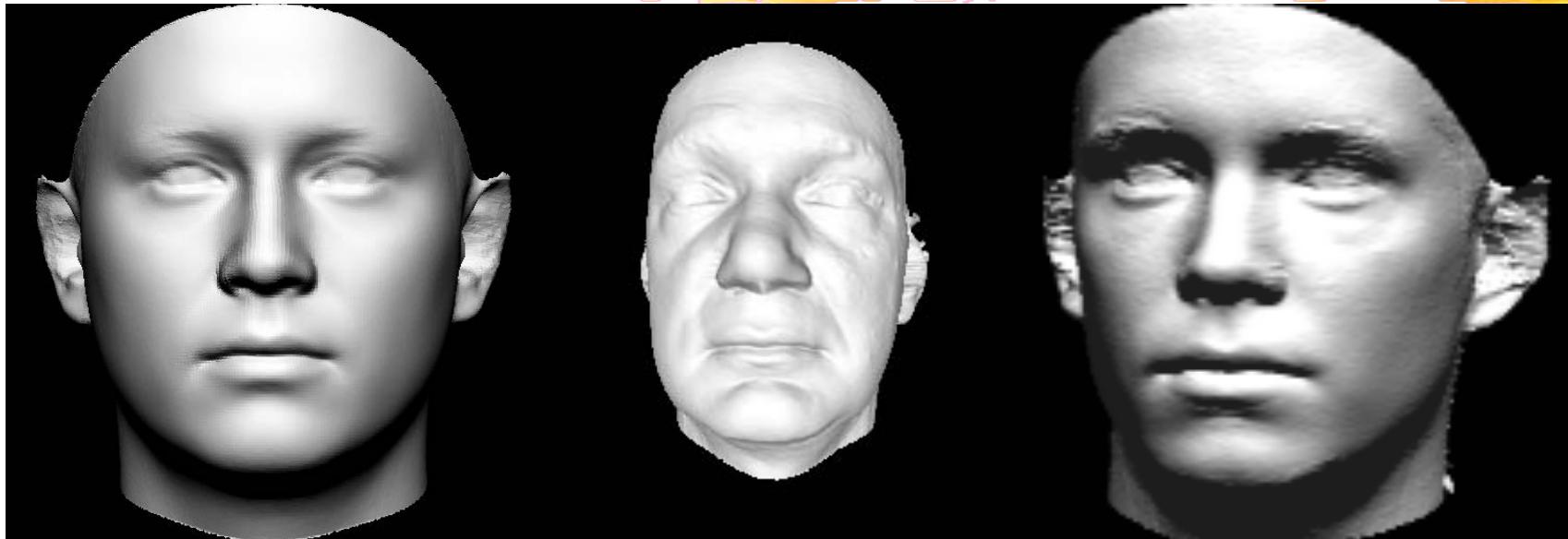


- No calibration data
- Relatively little texture
- Difficult lighting

# SIMPLE FACE MODEL



# PCA FACE MODEL

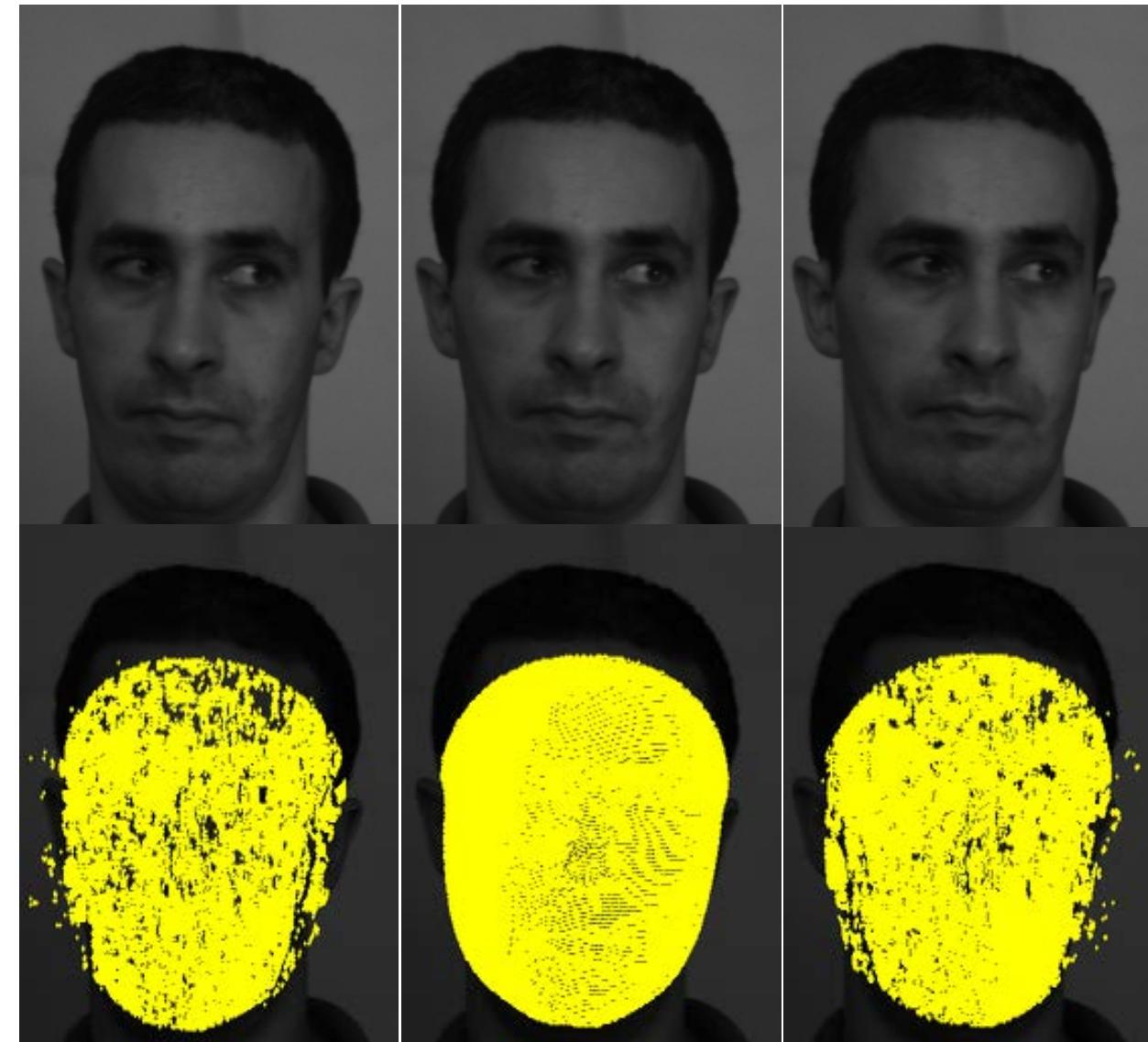


$$S = \bar{S} + \sum_{i=1}^{99} a_i S_i$$

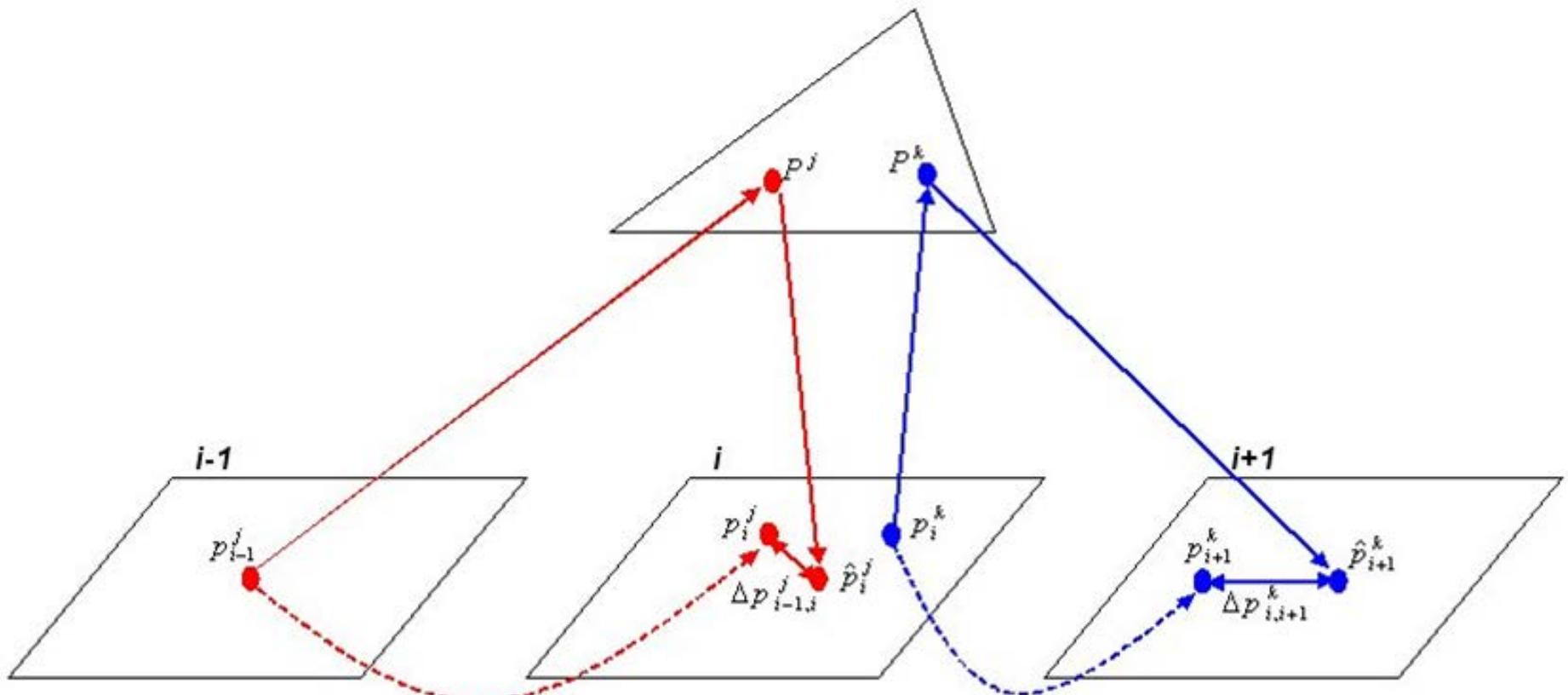
$\bar{S}$ : Average shape  
 $S_i$ : Shape vector  
 $a_i$ : Shape coefficients

V. Blanz and T. Vetter, "A Morphable Model for the Synthesis of 3-D Faces" in Computer Graphics, SIGGRAPH Proceedings, Los Angeles, CA, August 1999.

# CORRESPONDENCES

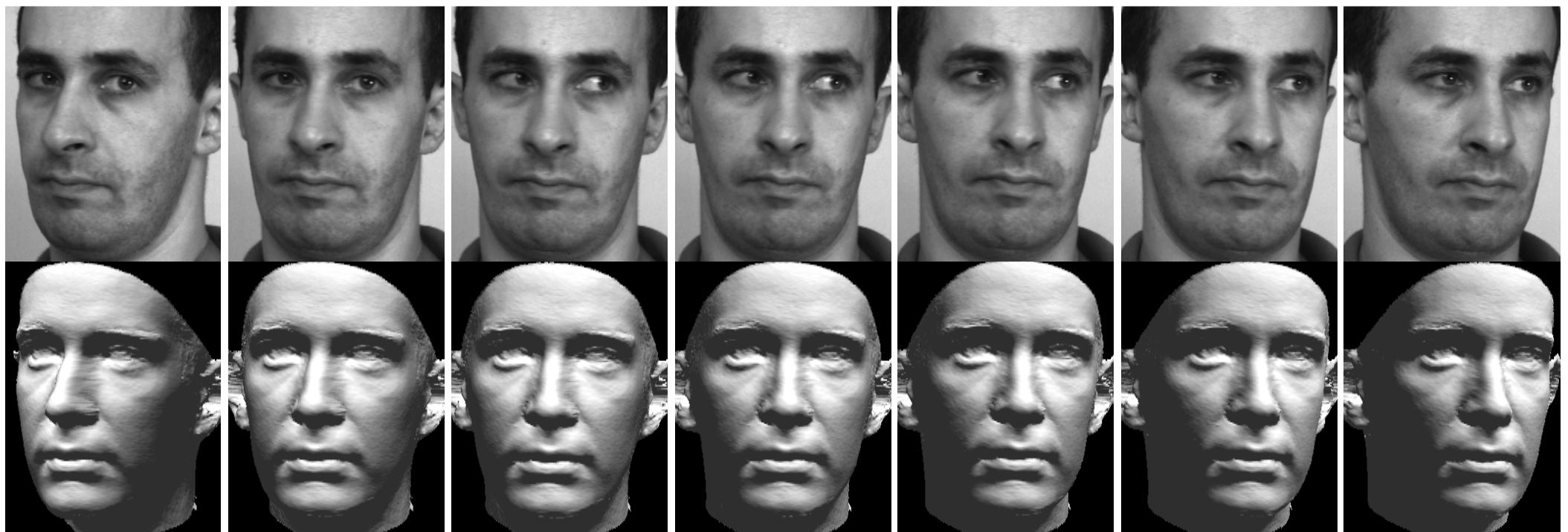


# TRANSFER FUNCTION



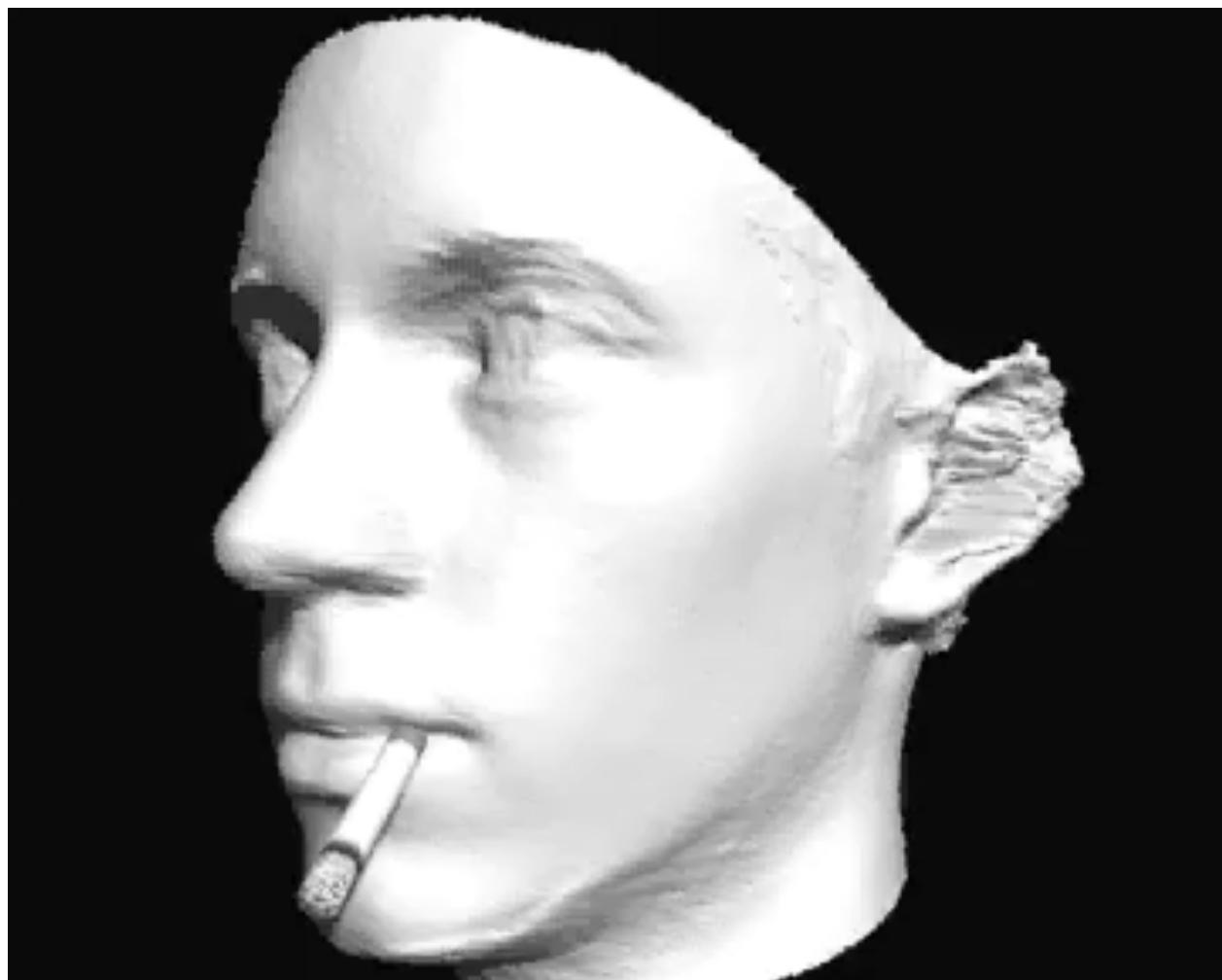
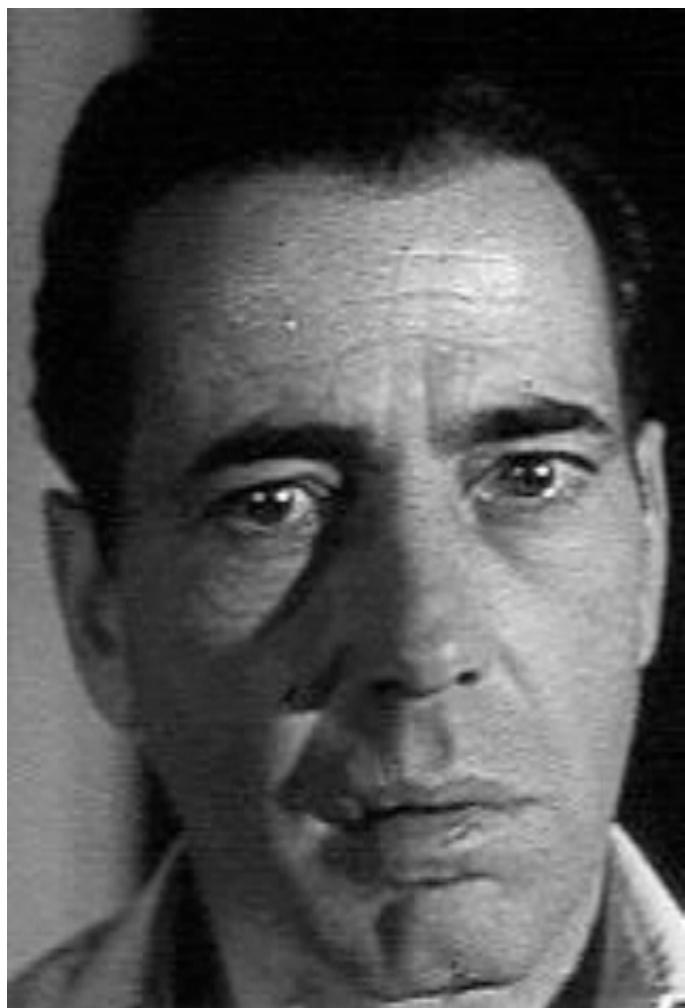
$$F_3(A, C_{i-1}, C_i, C_{i+1}) = \sum_{j \in Q_{i-1}} \left\| \Delta p_{i-1,i}^j \right\|^2 + \sum_{k \in Q_i} \left\| \Delta p_{i,i+1}^k \right\|^2$$

# MODEL BASED BUNDLE ADJUSTMENT

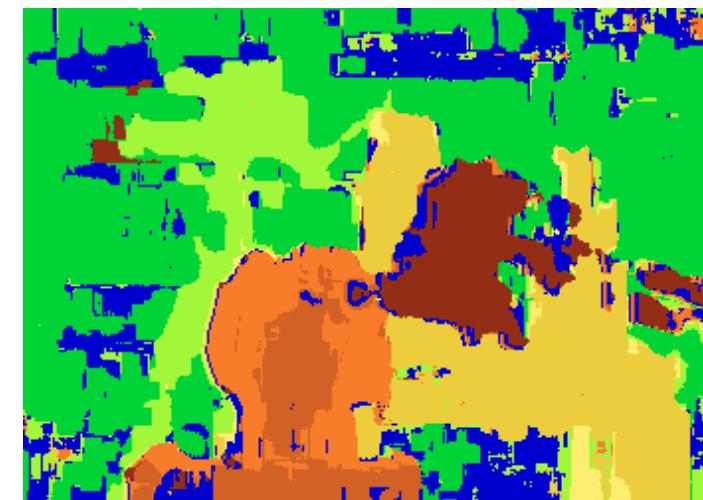
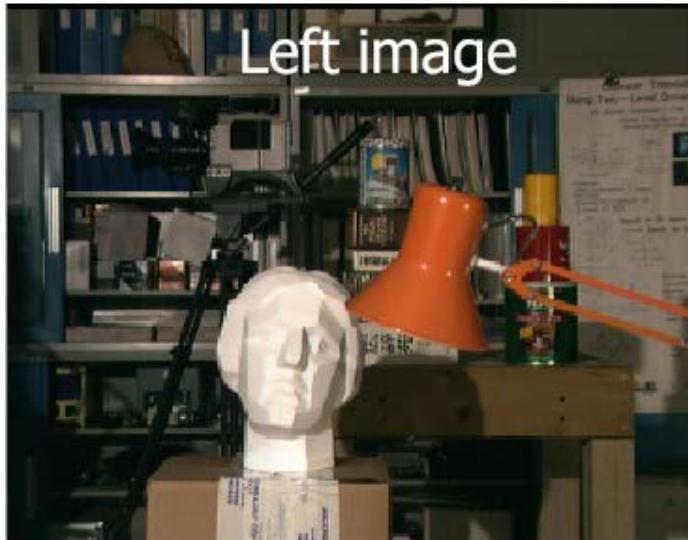


→ Median accuracy greater than 0.5mm

# MODEL FROM OLD MOVIE

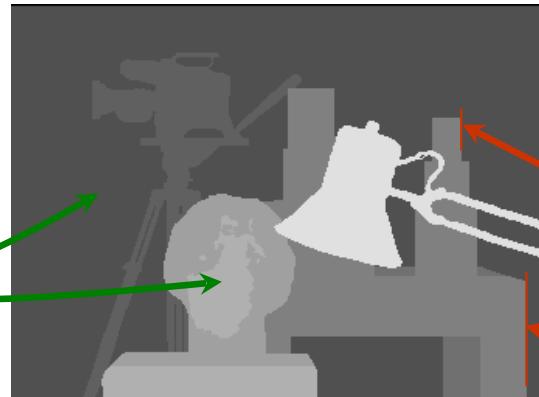


# LIMITATIONS OF WINDOW BASED METHODS



# ENERGY MINIMIZATION

Disparity  
continuous in  
most places,



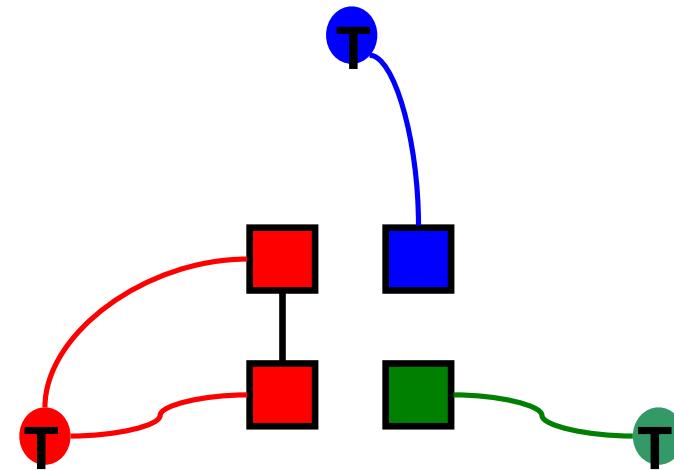
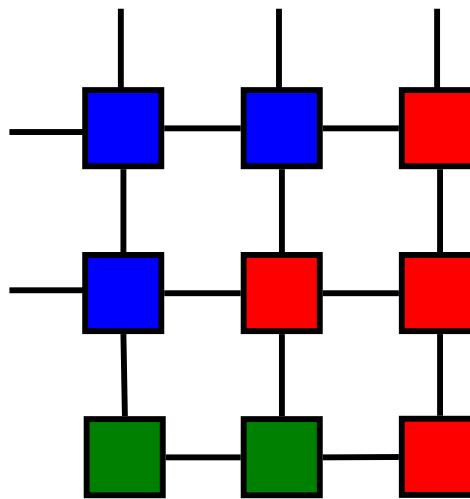
except at  
depth  
discontinuities

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

→ Minimize

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

# GRAPH CUTS



1. Stereo is a labeling problem
  2. Graph cut corresponds to a labeling.
- **Assign edge weights cleverly so that the min-weight cut gives the minimum energy!**

# GRAPH CUT OPTIMIZATION

Construct a graph including

Nodes:

Pixels (in first image)

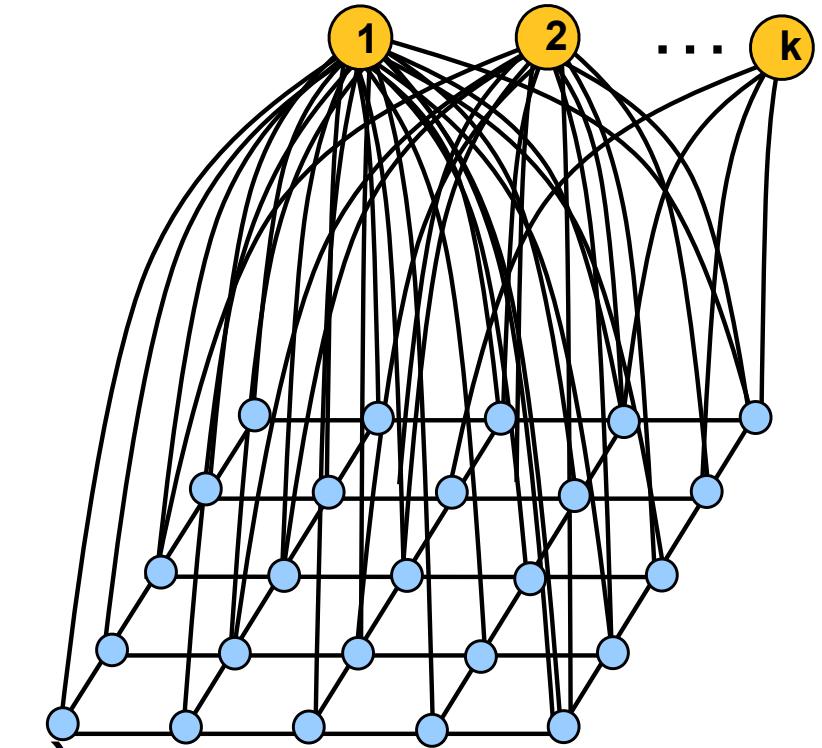
$k$  discrete disparity values

Edges:

From every pixel node to a  
depth node (data term)

Neighboring nodes (smoothness)

Assign weights corresponding to  
pixel intensities to get a global cost  
function



● depths

● pixels

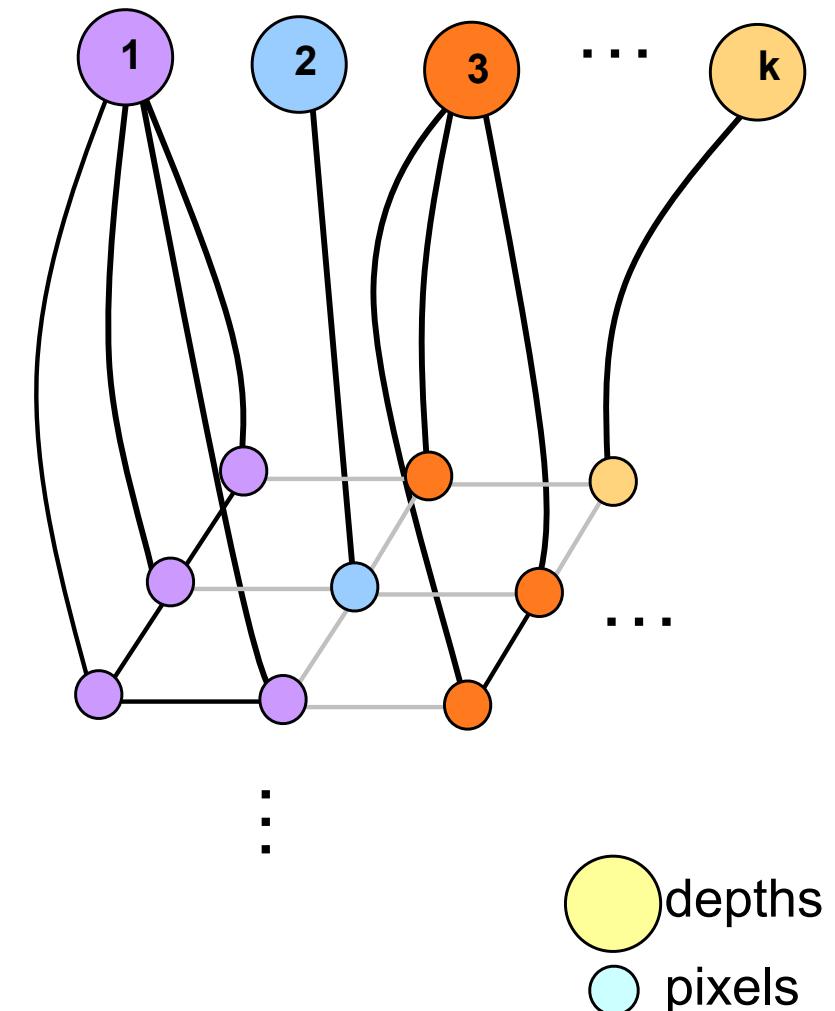
# MULTI WAY CUT

Goal:

Every pixel remains connected to one depth node.

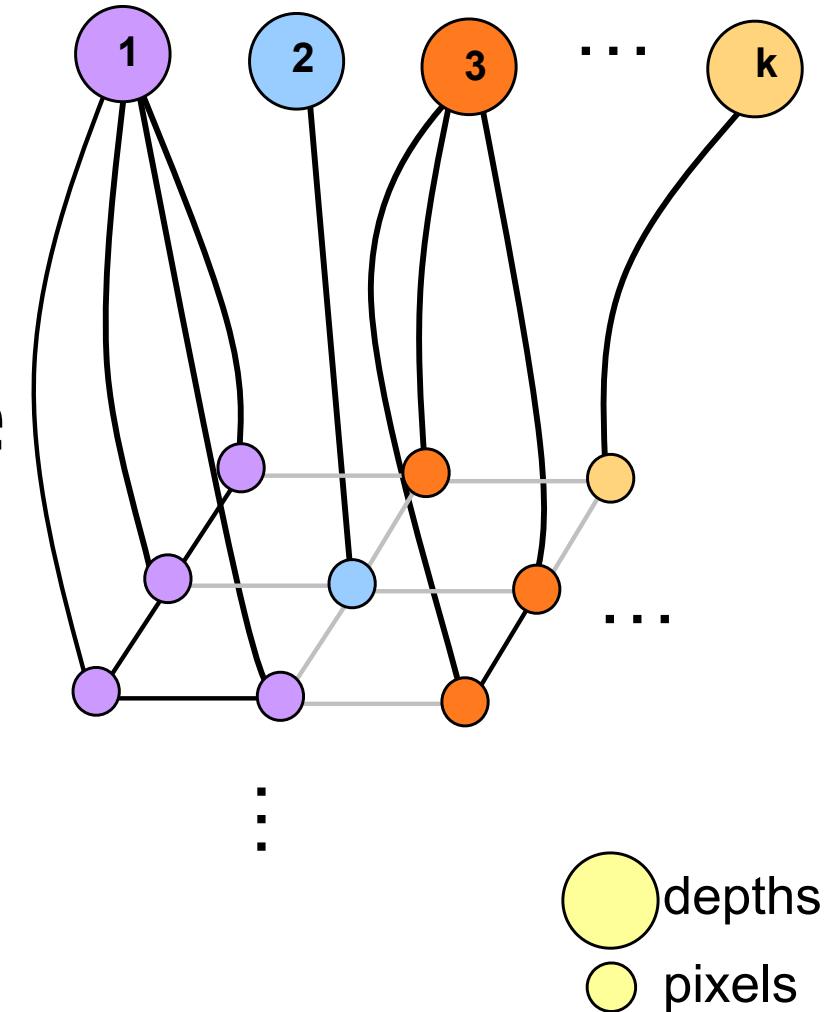
Edges between neighboring nodes only if they are connected to same depth node.

- Nodes are assigned the depth that they are connected to.
- Multiway cut is NP-complete, solve iteratively.

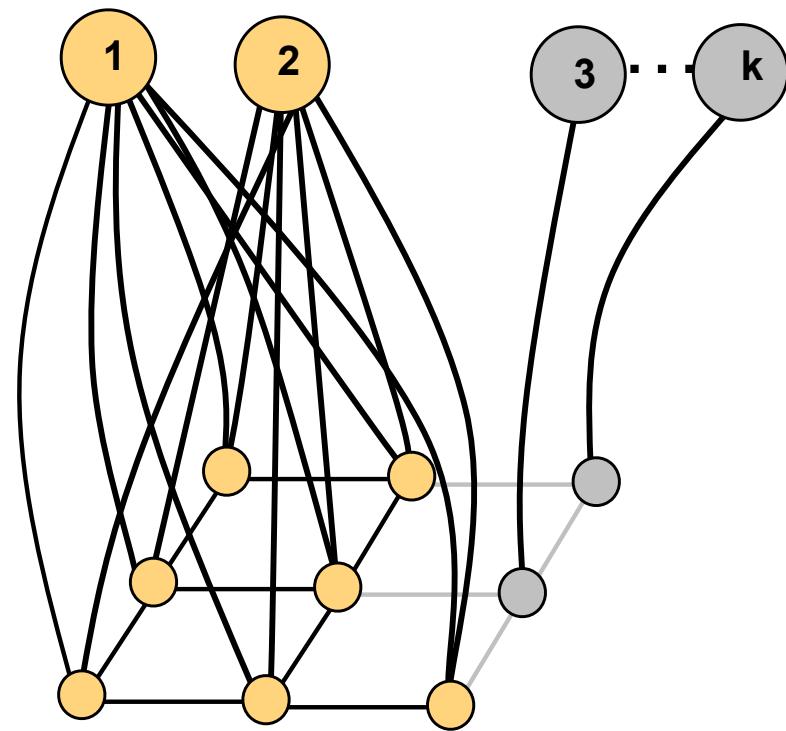
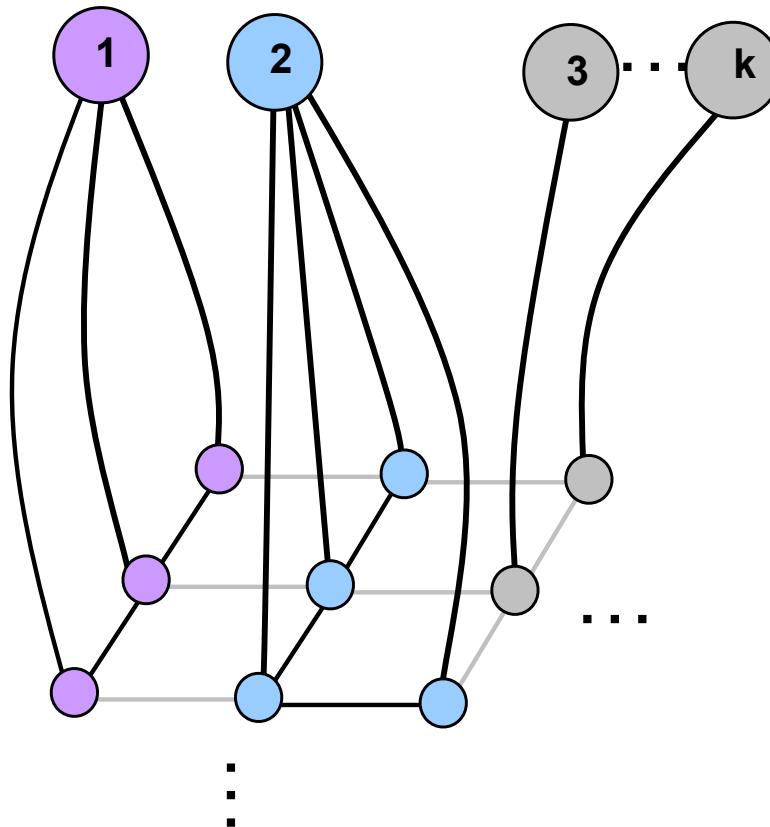


# $\alpha$ - $\beta$ SWAP

- Nodes labeled  $\alpha$  or  $\beta$  can switch their labels.
- Edges between neighbors are updated according to the new labeling.
- Other edges remain unchanged.

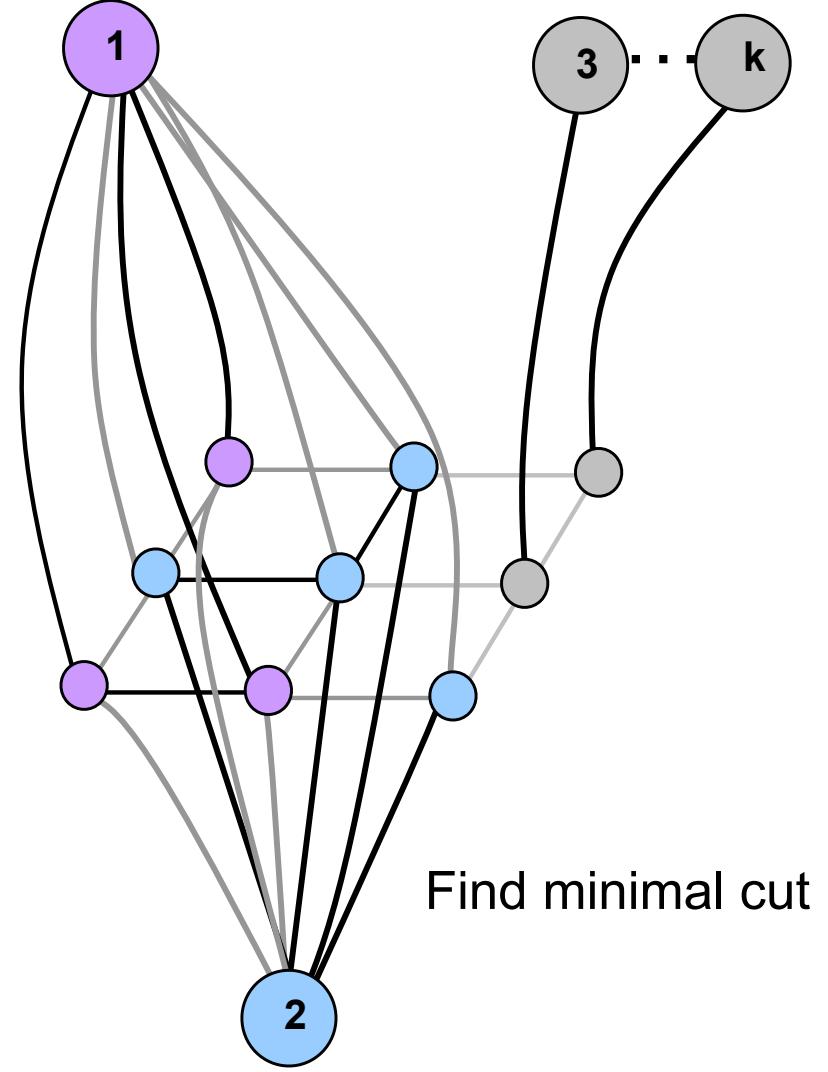
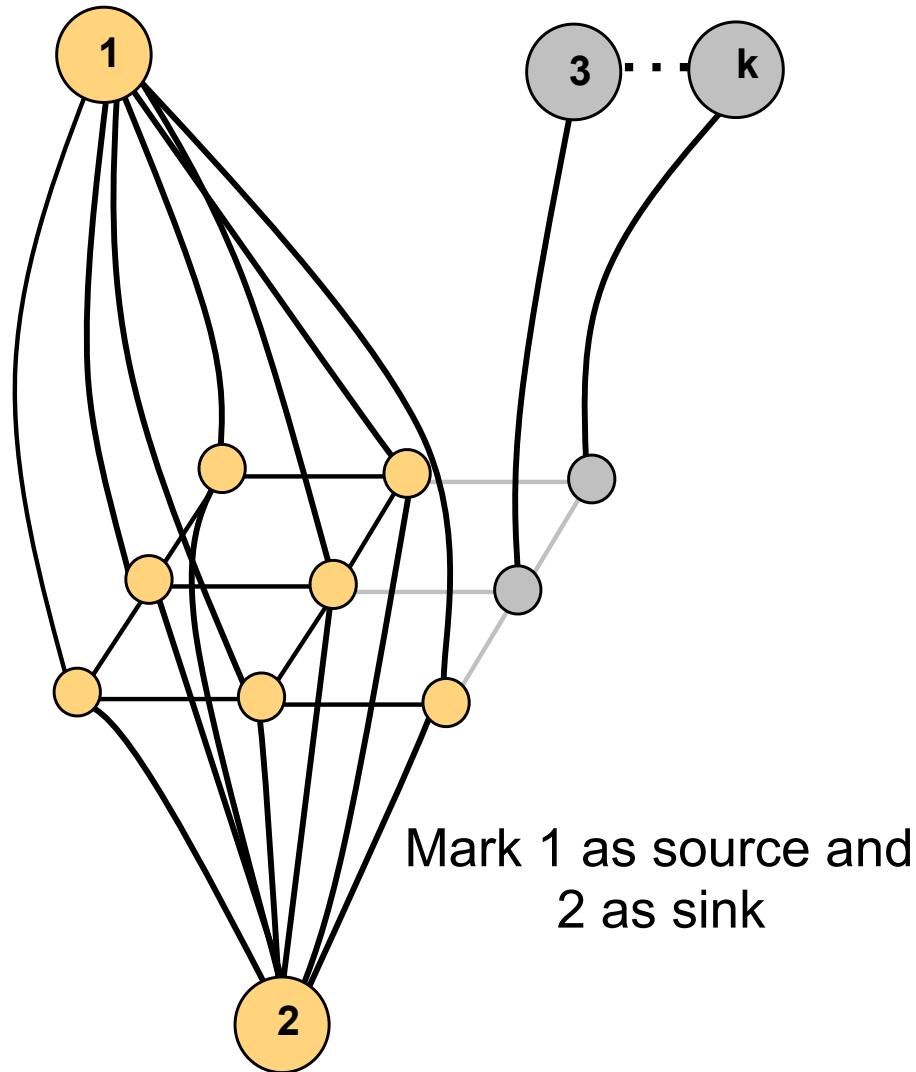


# EXAMPLE: 1-2 SWAP

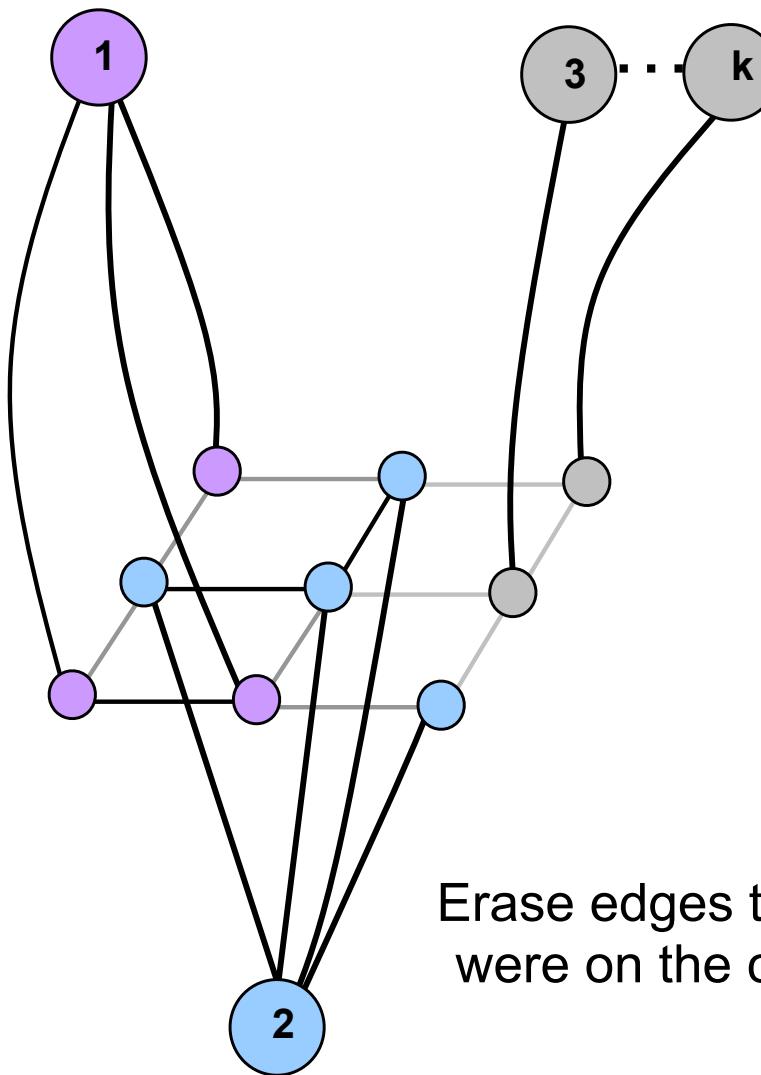


Connect the nodes  
labeled 1 or 2 to both  
labels

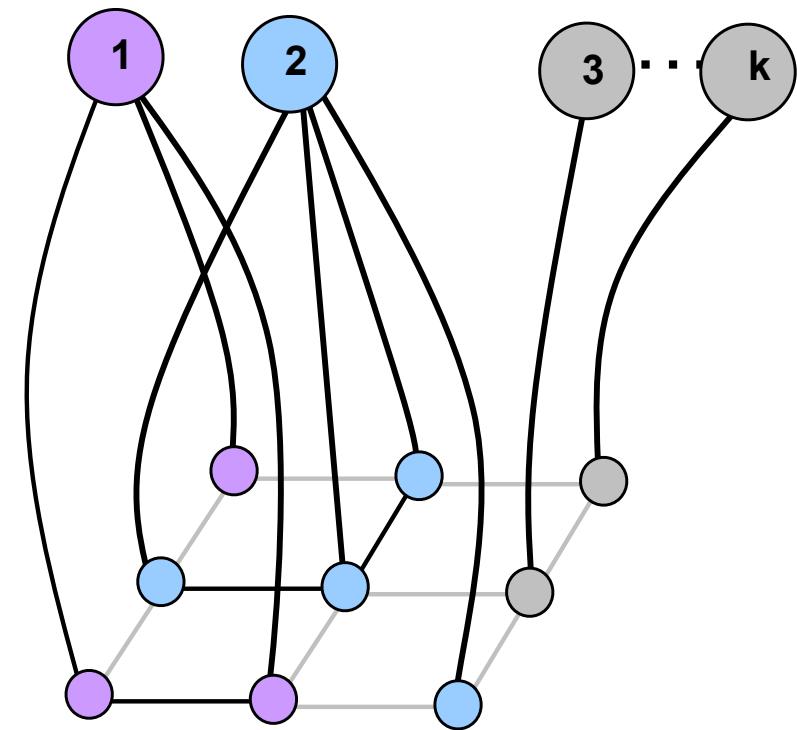
# EXAMPLE: 1-2 SWAP



# EXAMPLE: 1-2 SWAP

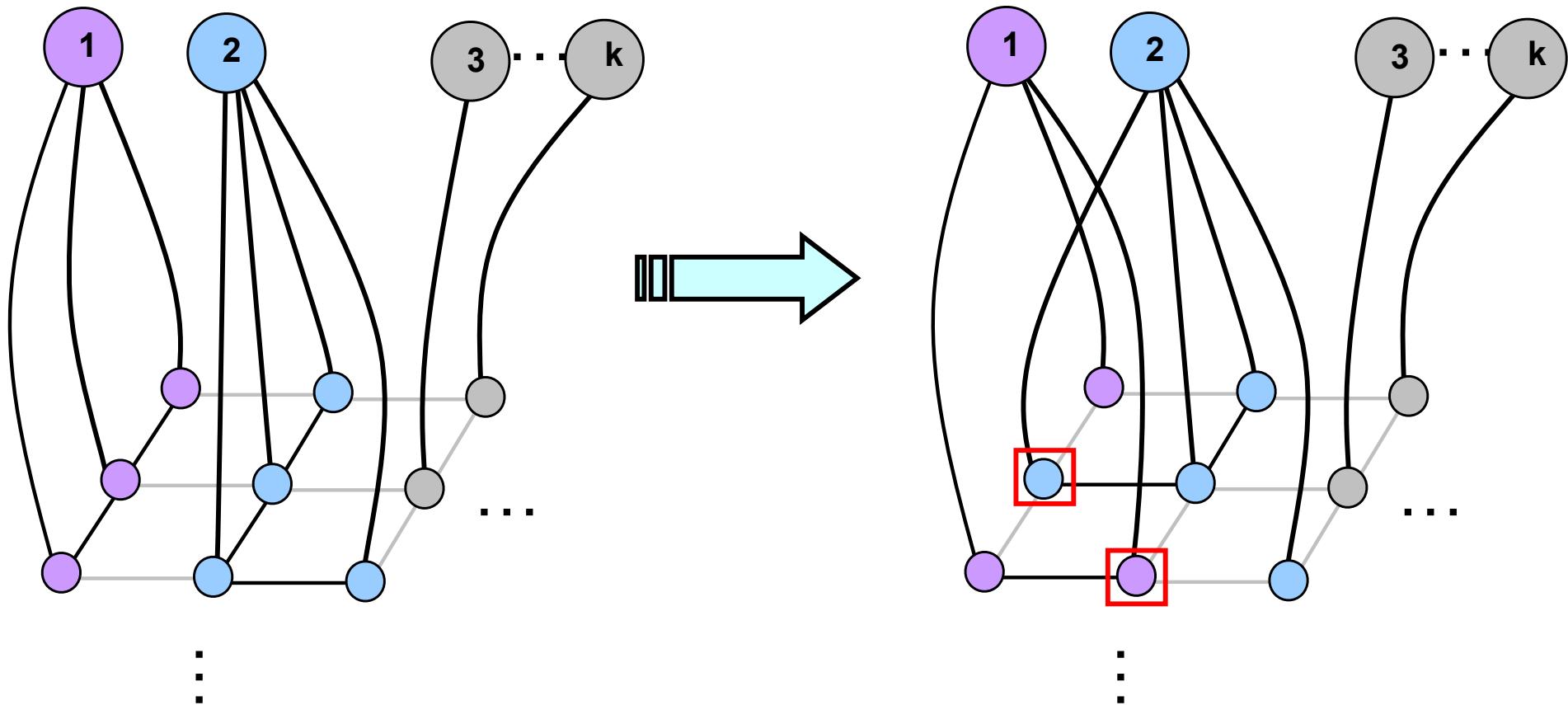


Erase edges that  
were on the cut



Result: a new labeling of  
the 1,2 nodes

## EXAMPLE: 1-2 SWAP



# GRAPH CUT ALGORITHM

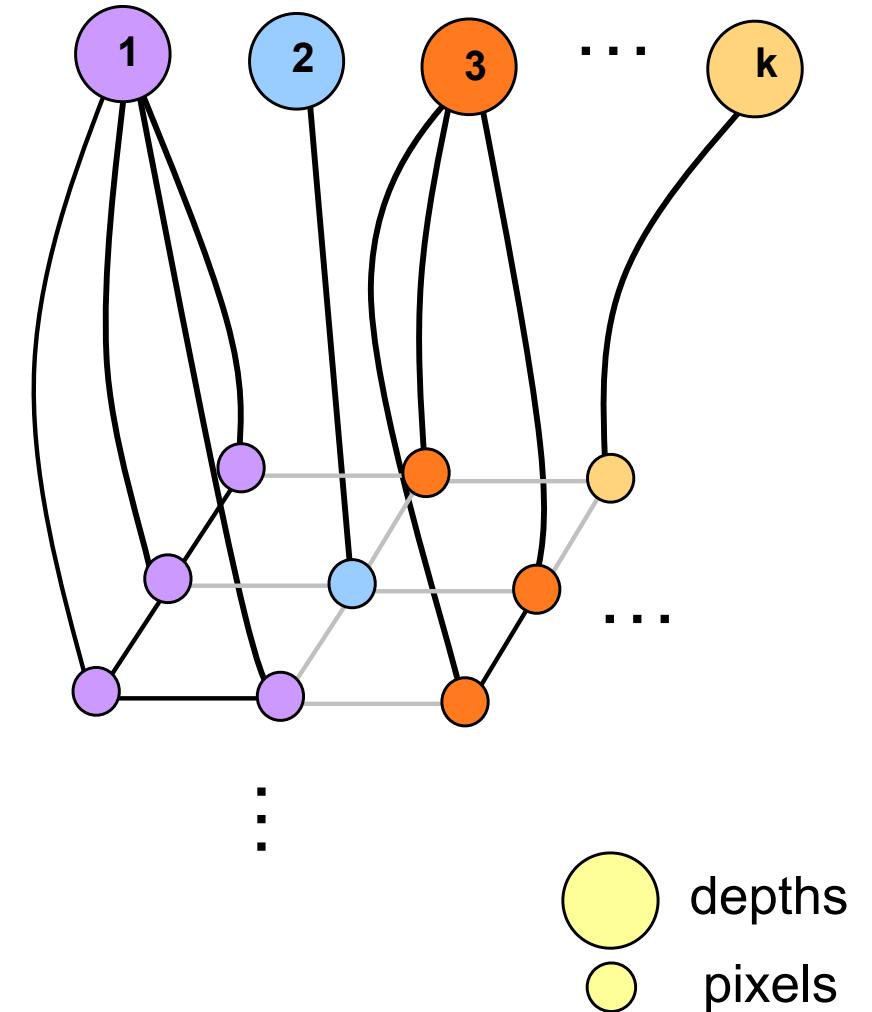


1. Start with an arbitrary labeling
2. For every pair  $\{\alpha, \beta\}$  in  $\{1, \dots, k\}$ 
  - Find the  $\alpha$ - $\beta$  swap that minimizes the function
  - Update the graph by adding and erasing edges
3. Quit when no pair improves the cost function
4. Induce pixel labels

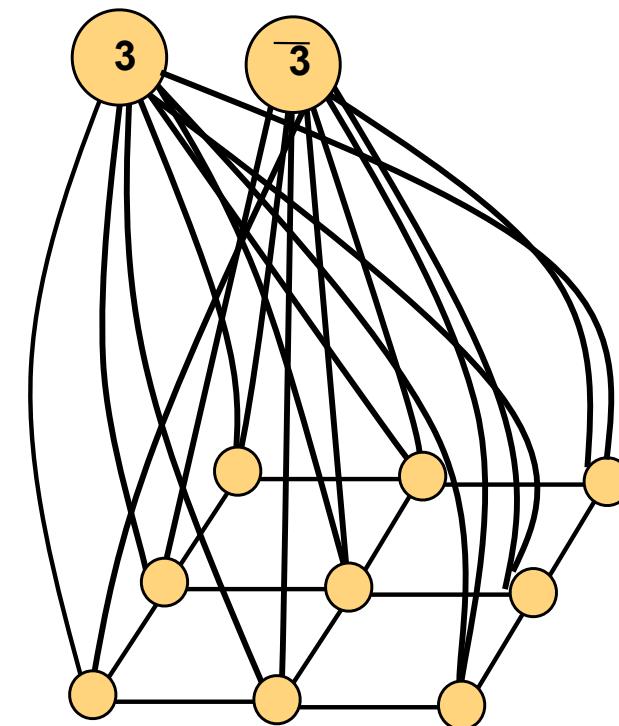
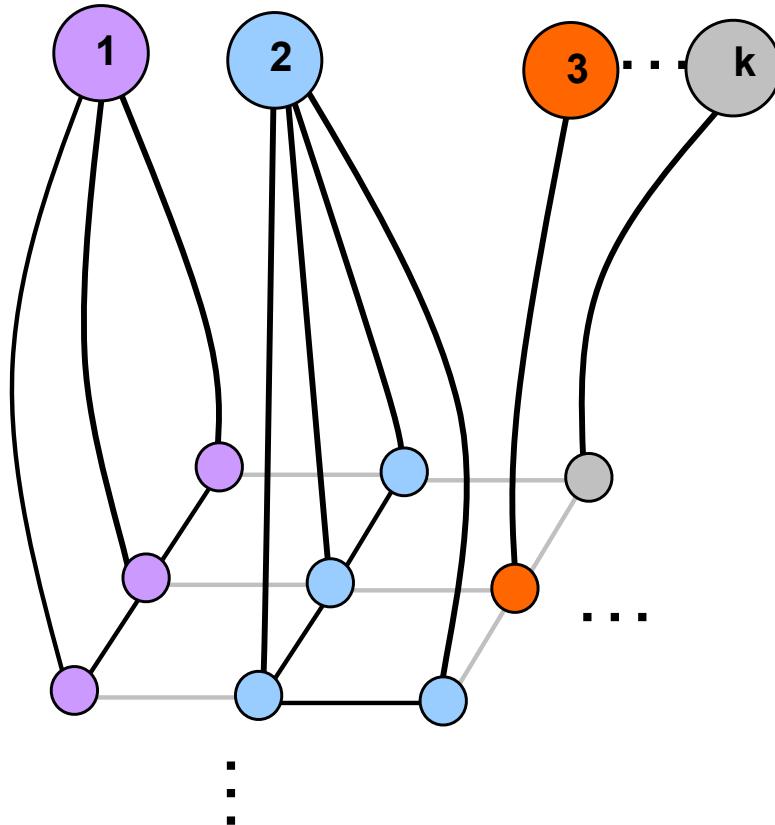
# $\alpha$ -Expansion



- Nodes having a label different than  $\alpha$  can either keep it or switch to  $\alpha$ .
- Edges between neighbors are updated according to the new labeling.
- Other edges remain unchanged.

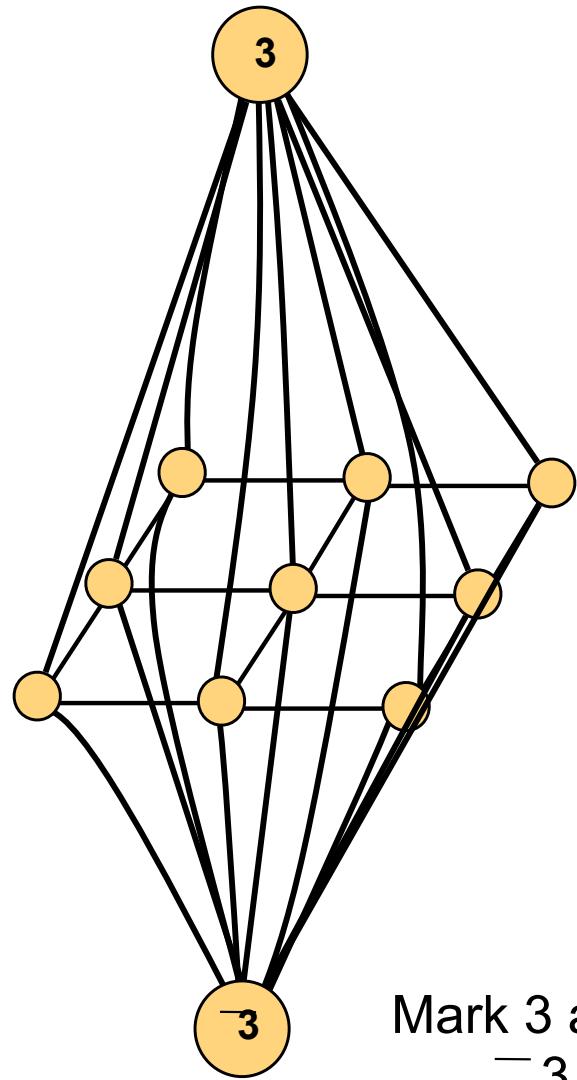


# EXAMPLE: 3-EXPANSION

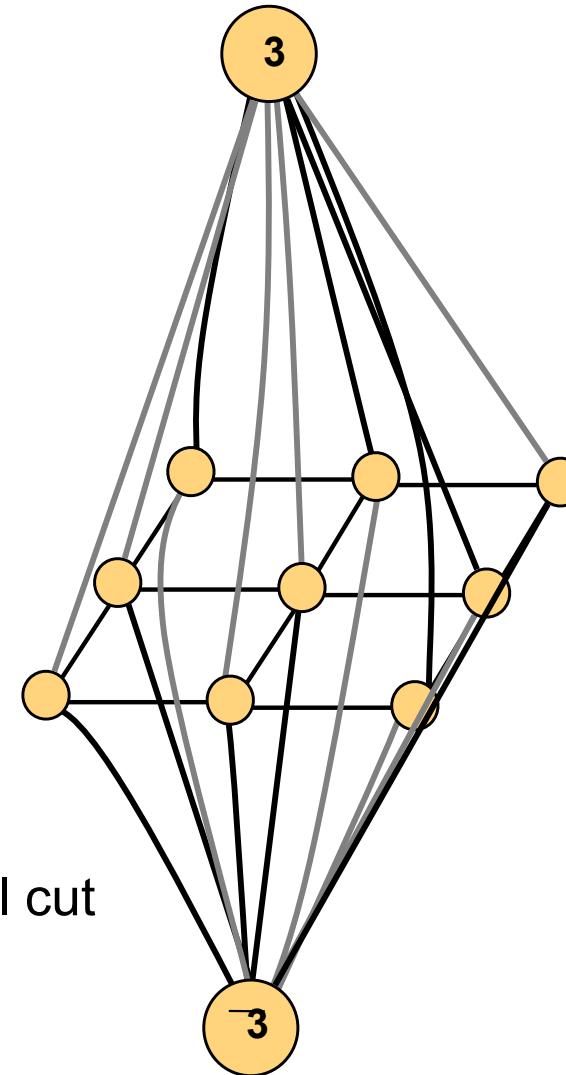


Connect all nodes to  
both 3 and  $\bar{3}$

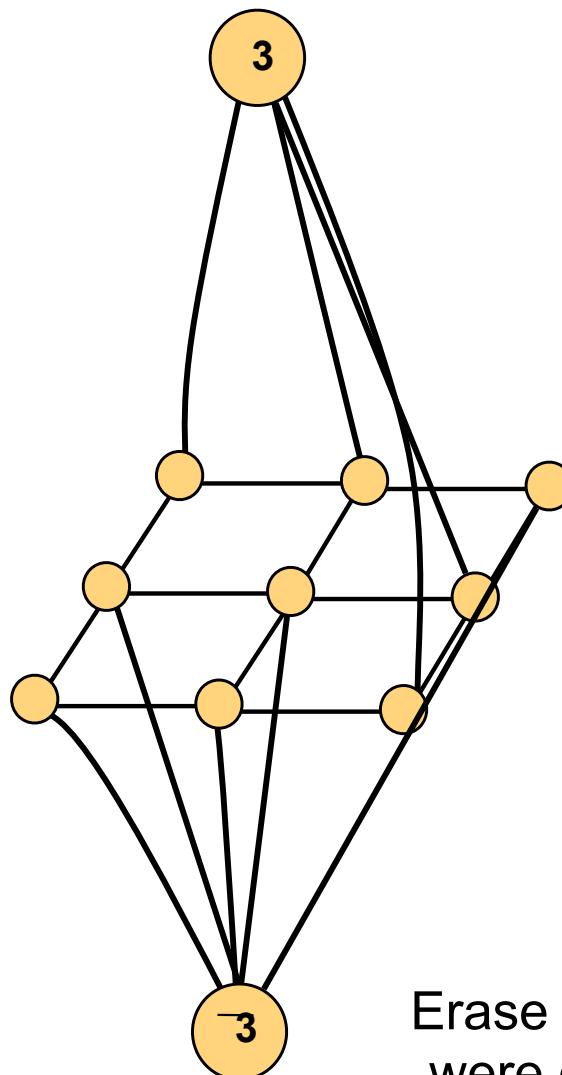
# EXAMPLE: 3-EXPANSION



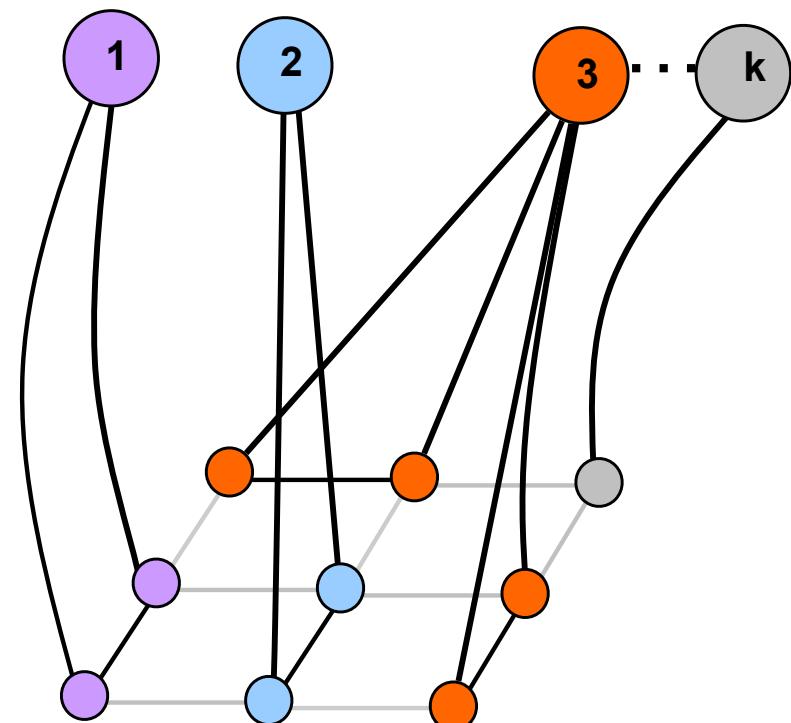
Find minimal cut



# EXAMPLE: 3-EXPANSION

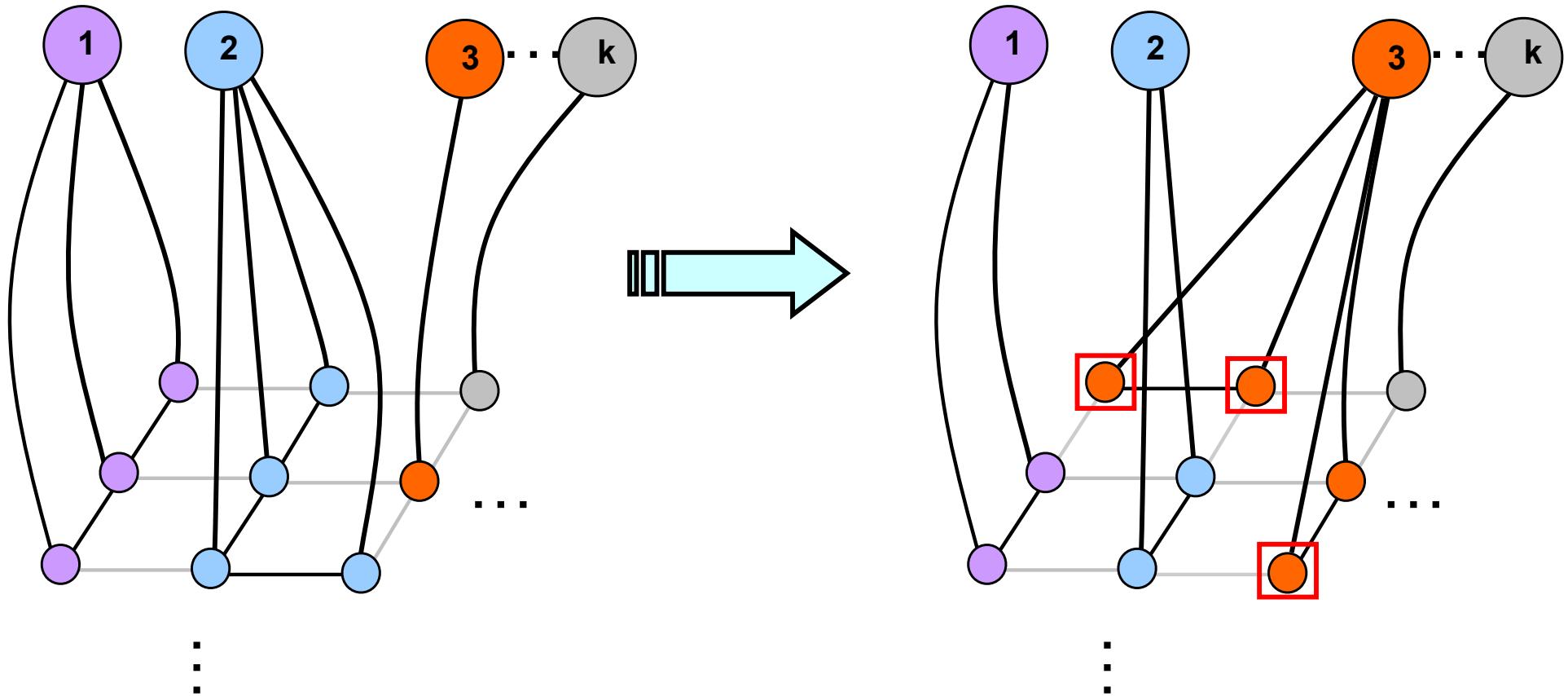


Erase edges that  
were on the cut



Result: 3-expansion

# EXAMPLE: 3-EXPANSION



# GRAPH CUT ALGORITHM



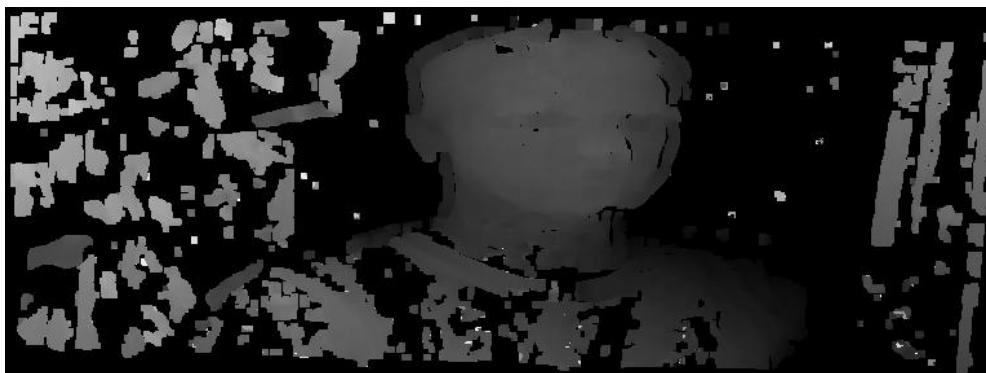
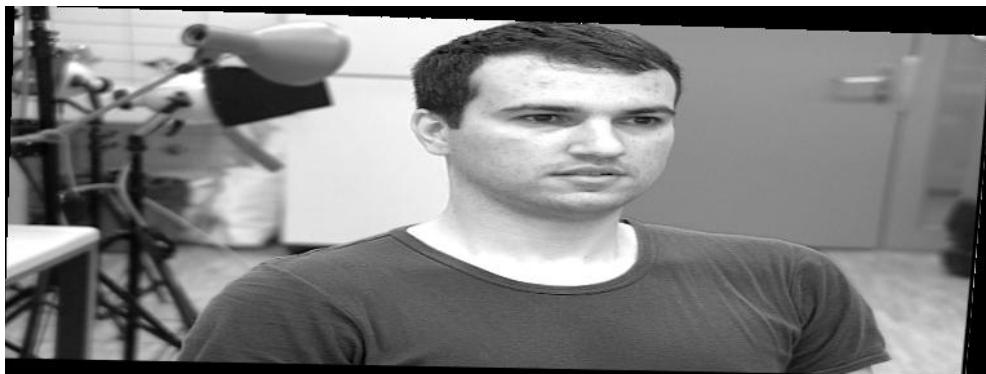
1. Start with an arbitrary labeling
2. For every label  $\alpha$  in  $\{1, \dots, L\}$ 
  - Find the  $\alpha$ -Expansion that minimizes the function
  - Update the graph by adding and erasing edges
3. Quit when no expansion improves the cost
4. Induce pixel labels

## $\alpha$ - $\beta$ SWAP vs $\alpha$ -EXPANSION

	Pair-Wise Penalty	Optimality
$\alpha$ - $\beta$ Swap	Semi-metric	No guarantee
$\alpha$ -Expansion	Metric	Twice global optimum

- $\alpha$ -Expansion guarantees a solution whose energy is at most twice the global optimum but requires the pairwise term to satisfy the triangular inequality.
- $\alpha$ - $\beta$  Swap offers no such guarantee but can deal with more generic pairwise terms.

# NCC vs GRAPH CUTS



Normalized correlation

Graph Cuts

# NCC vs GRAPH CUTS

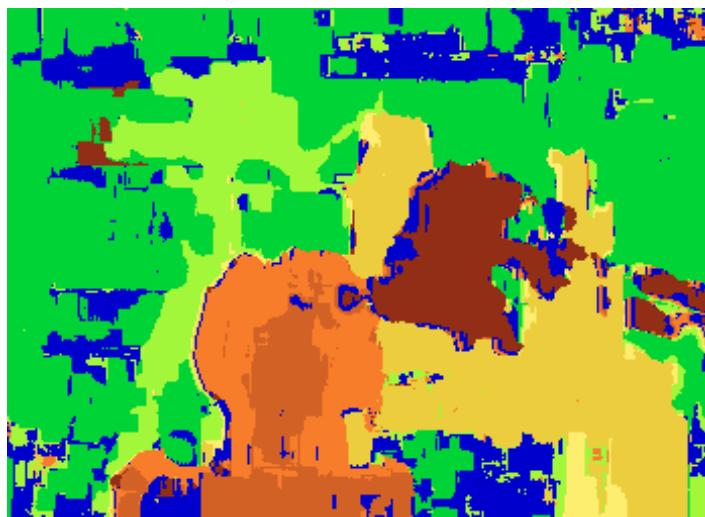
left image



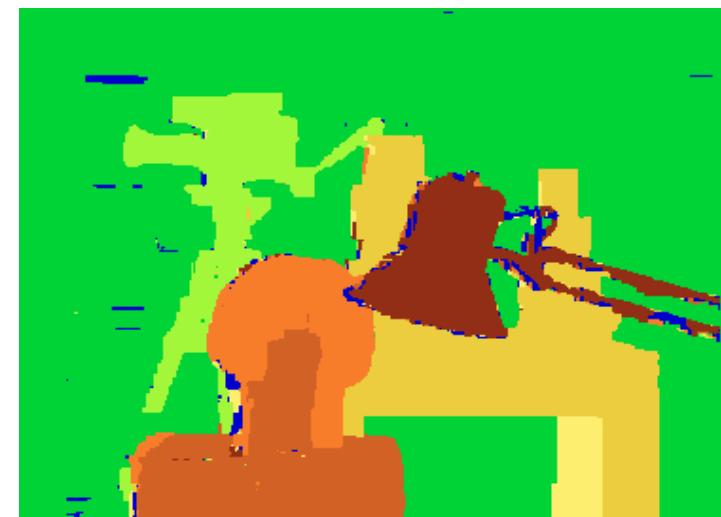
true disparities



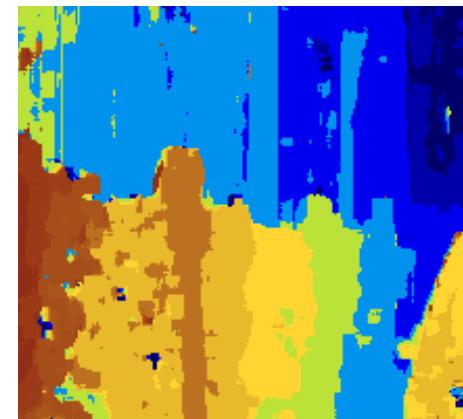
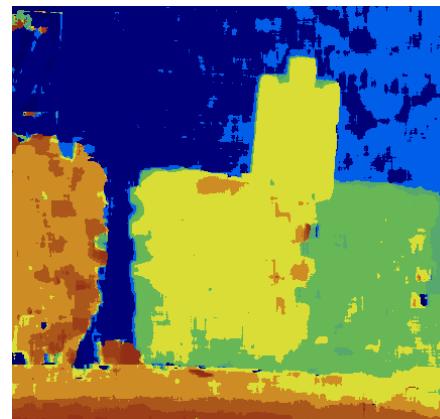
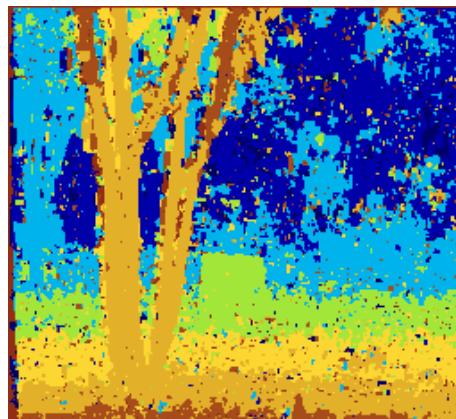
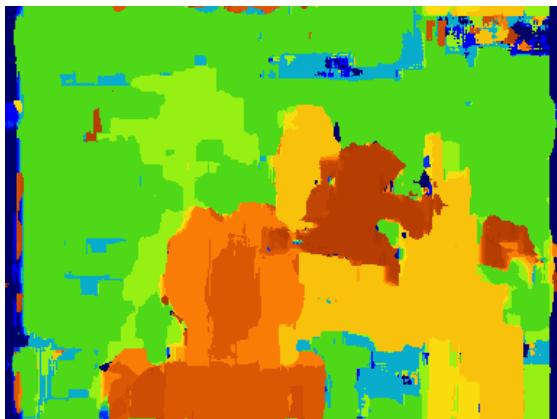
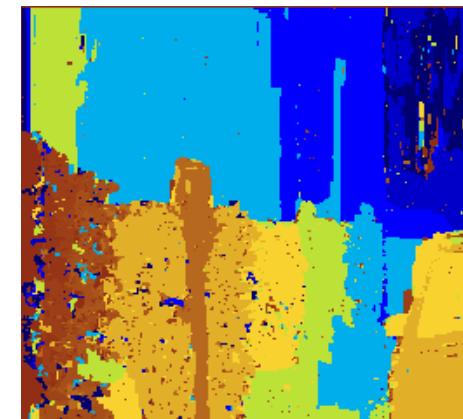
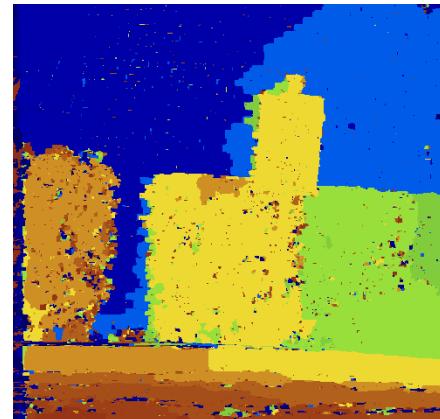
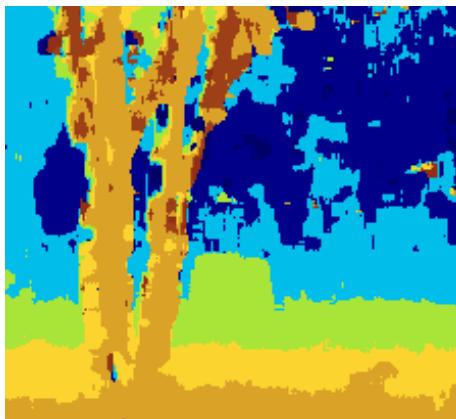
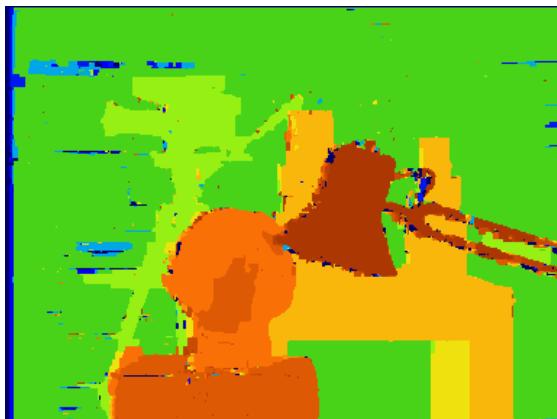
Normalized correlation



Graph Cuts



# GRAPH CUT RESULTS



# STRENGTHS AND LIMITATIONS



## Strengths:

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

## Limitations:

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