Image Based Rendering

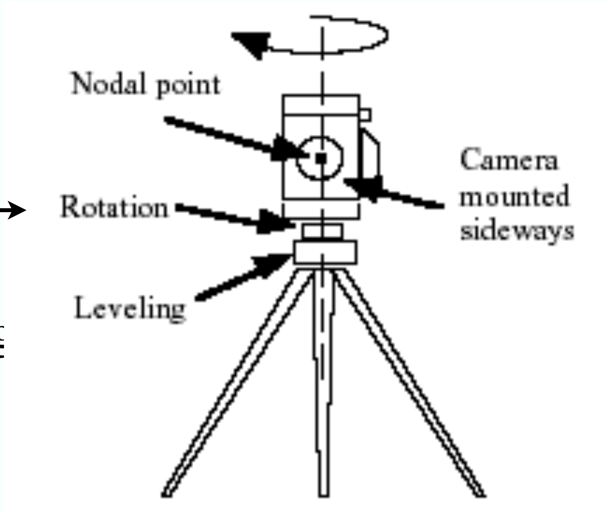
D.A. Forsyth, with slides from John Hart

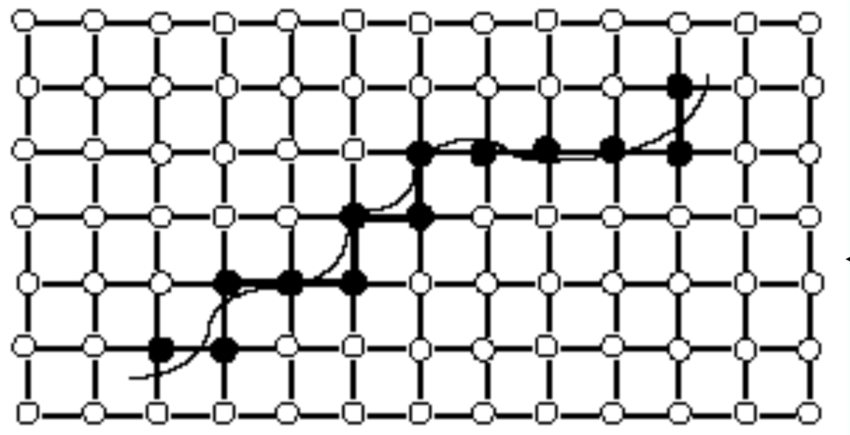
Topics

- Mosaics
  - translating cameras reveal extra information, break occlusion
- Optical flow
  - for very small movements of the camera
- Explicit image based rendering
  - multiple calibrated cameras yield a system of rays that models objects
- Camera calibration
  - postrender things into pictures
- Stereopsis
  - two cameras reveal a lot of geometry
- Structure from motion
  - more cameras yield even more geometry

Implicit example: Quicktime VR

- Construct a mosaic that can provide various camera views at various points
- Issues:
  - recovering the mosaics
    - specialised hardware
    - correlation based mosaicing
  - structuring the representation for fast rendering

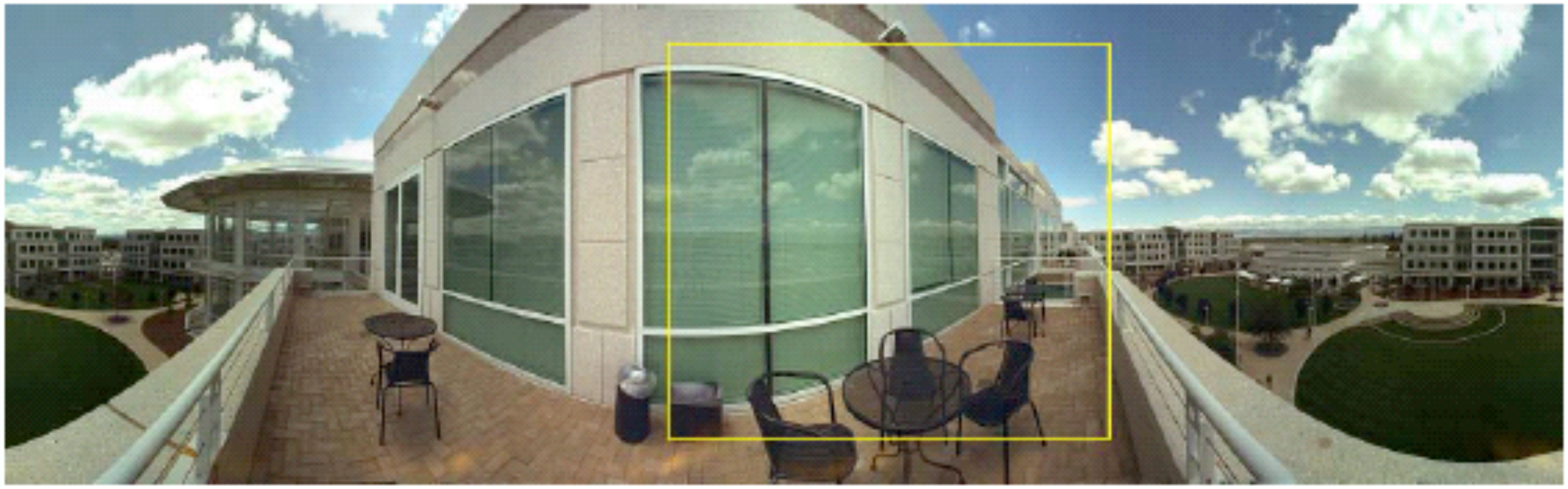


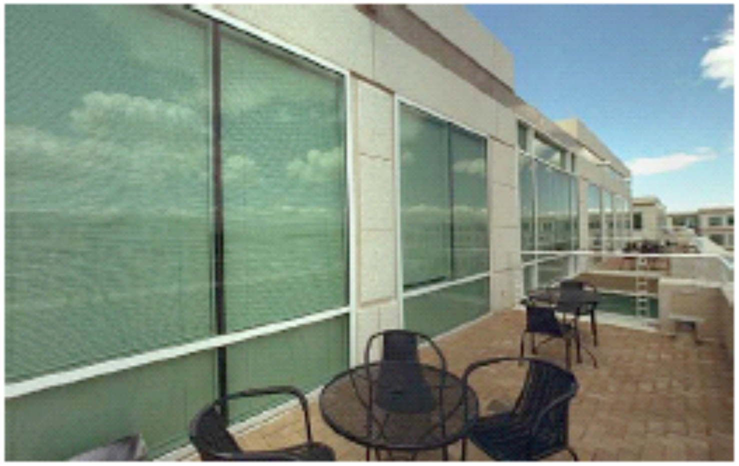


One view per point

Figures from "QuickTime VR – An Image-Based Approach to Virtual Environment Navigation", Shenchang Eric Chen, SIGGRAPH 95

Window of cylindrical mosaic

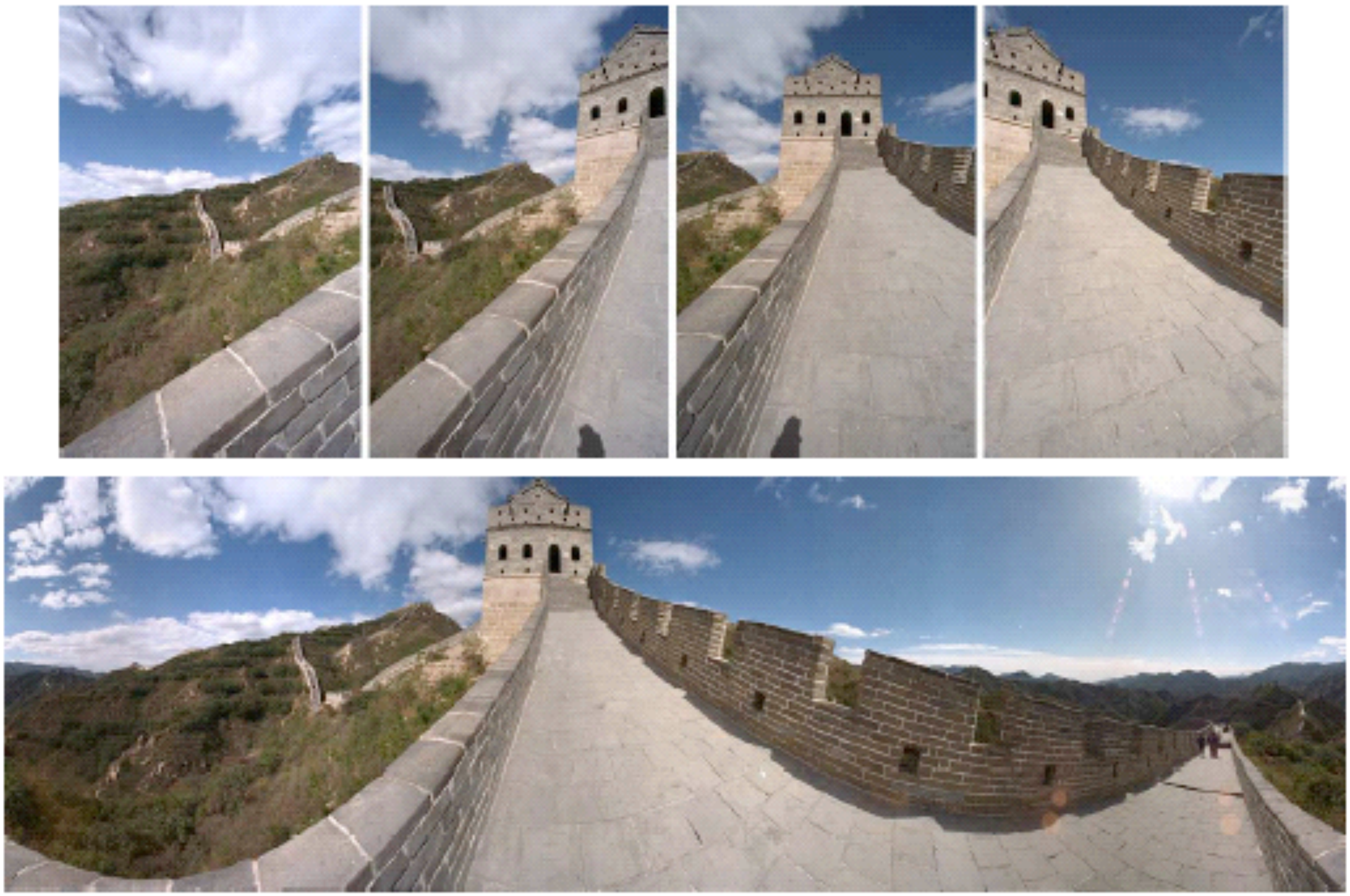




Rectified to flat camera

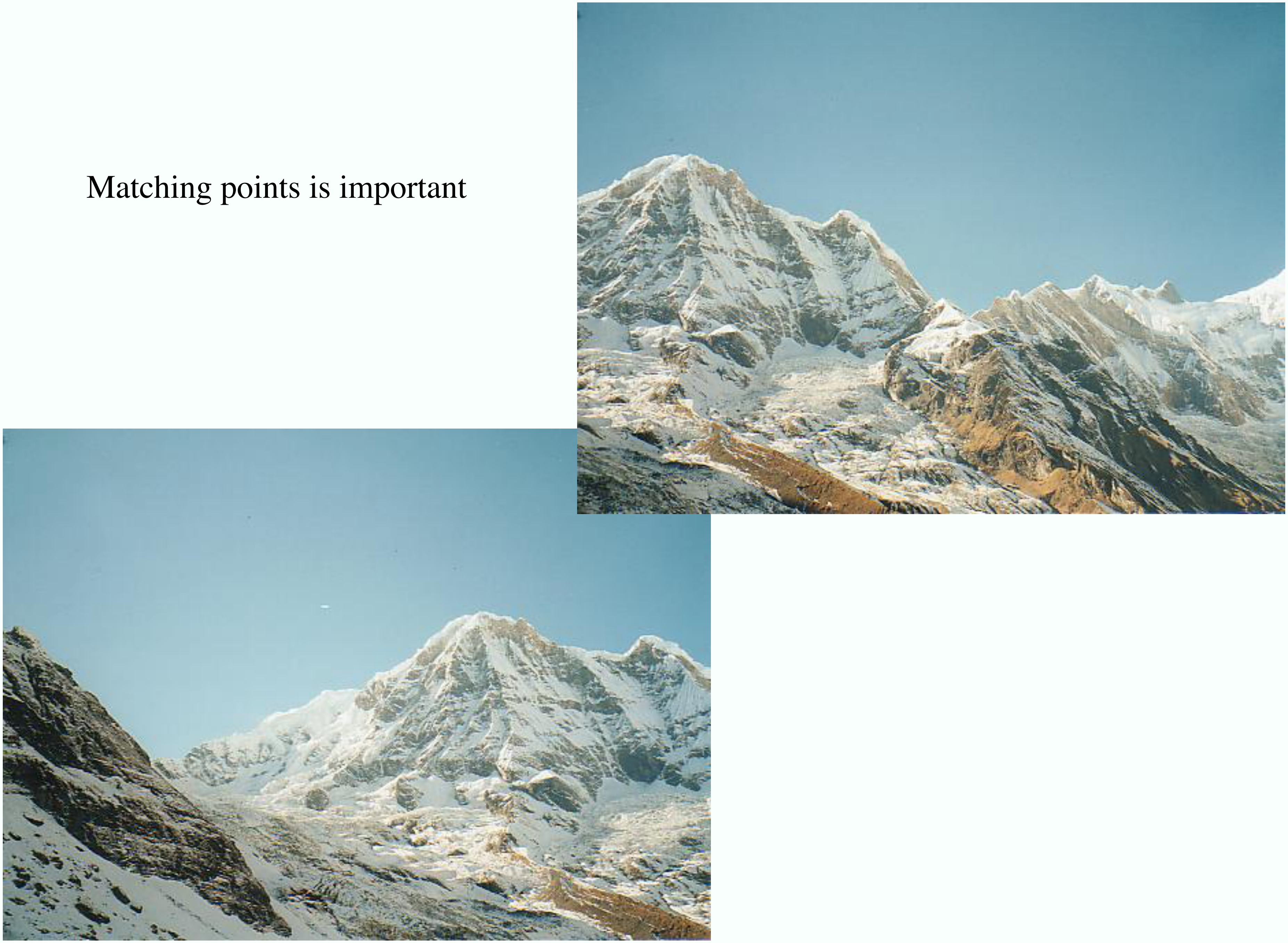
Figures from “QuickTime VR – An Image-Based Approach to Virtual Environment Navigation”, Shenchang Eric Chen, SIGGRAPH 95

But a cylindrical camera is hard to get - make mosaic from flat images



Figures from “QuickTime VR – An Image-Based Approach
to

Matching points is important



M. Brown and D. Lowe, “Recognising Panoramas”, ICCV 2003

Matching points

- A description of tiny gradients near point is distinctive
  - Lowe’s SIFT feature

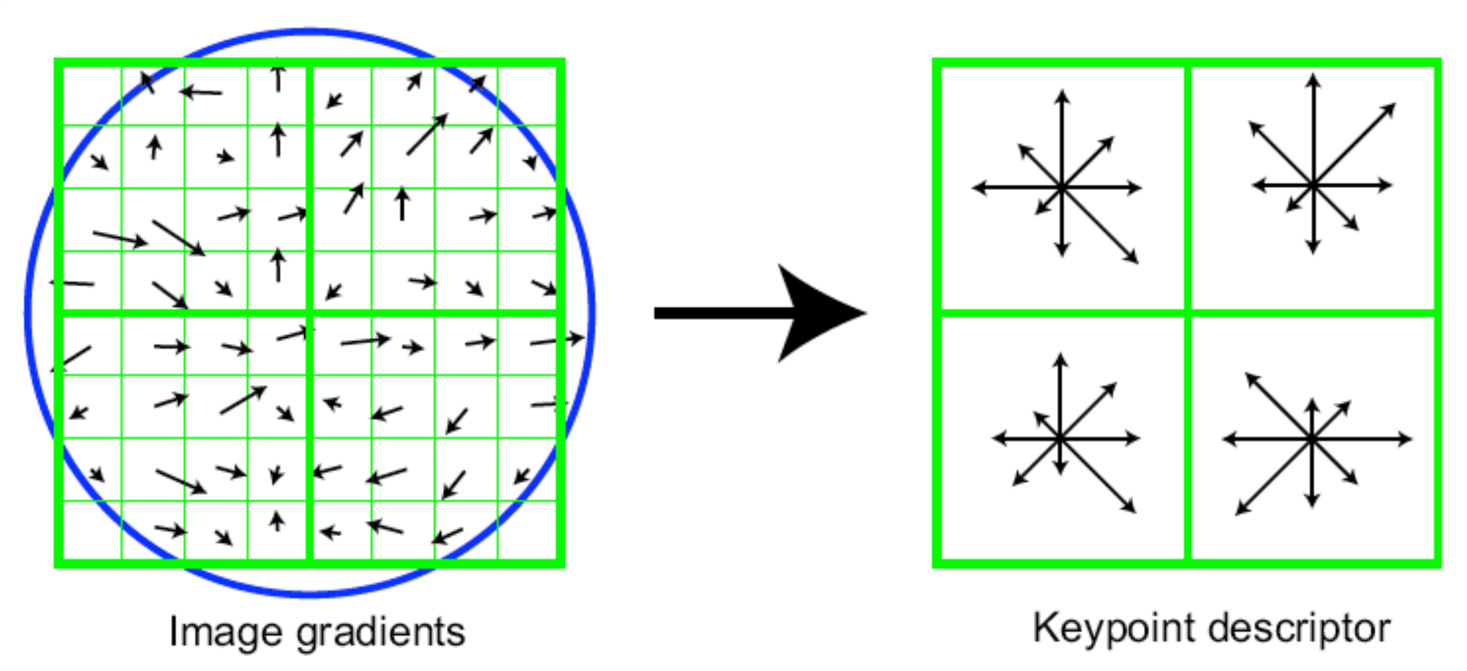
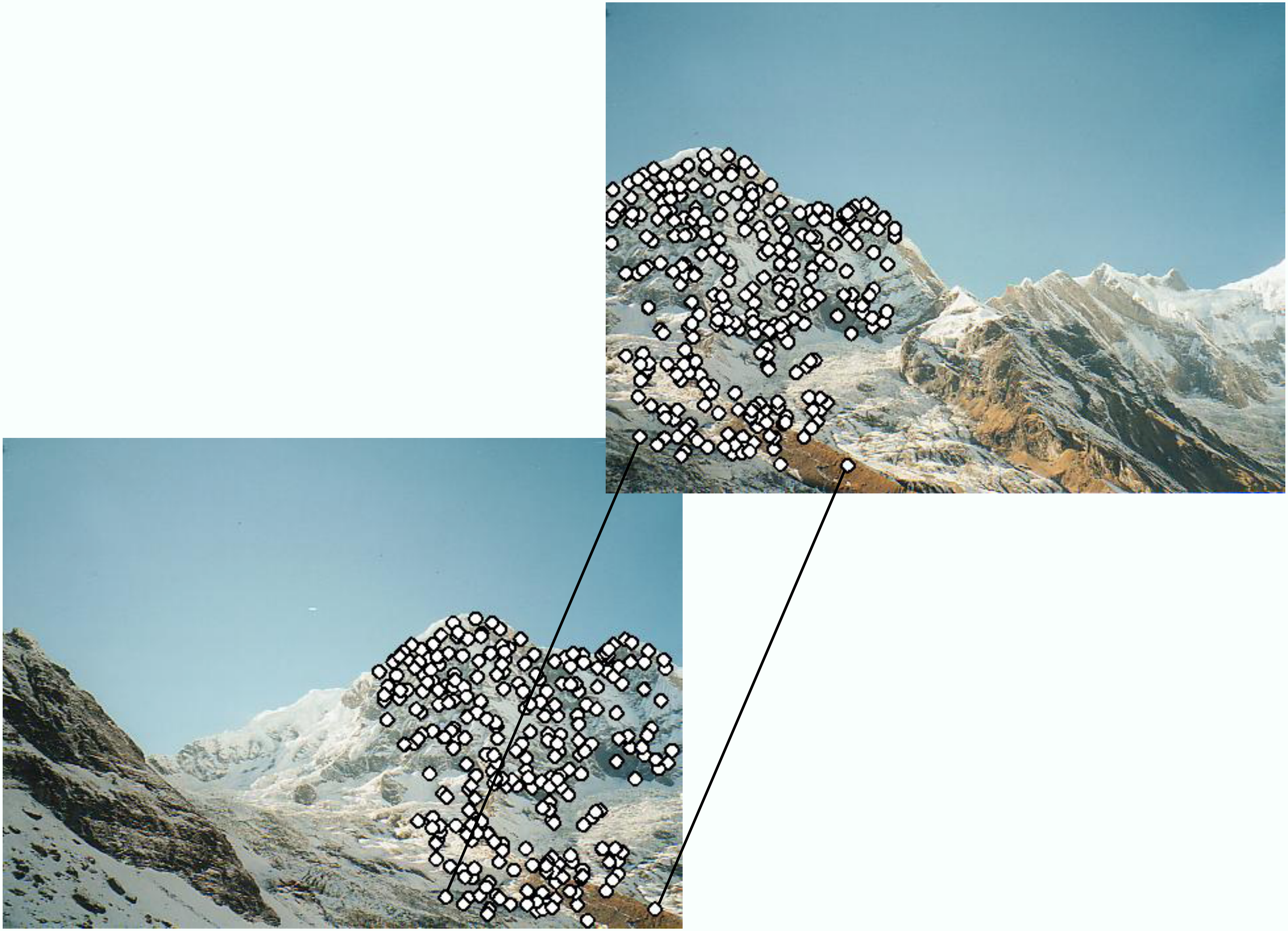
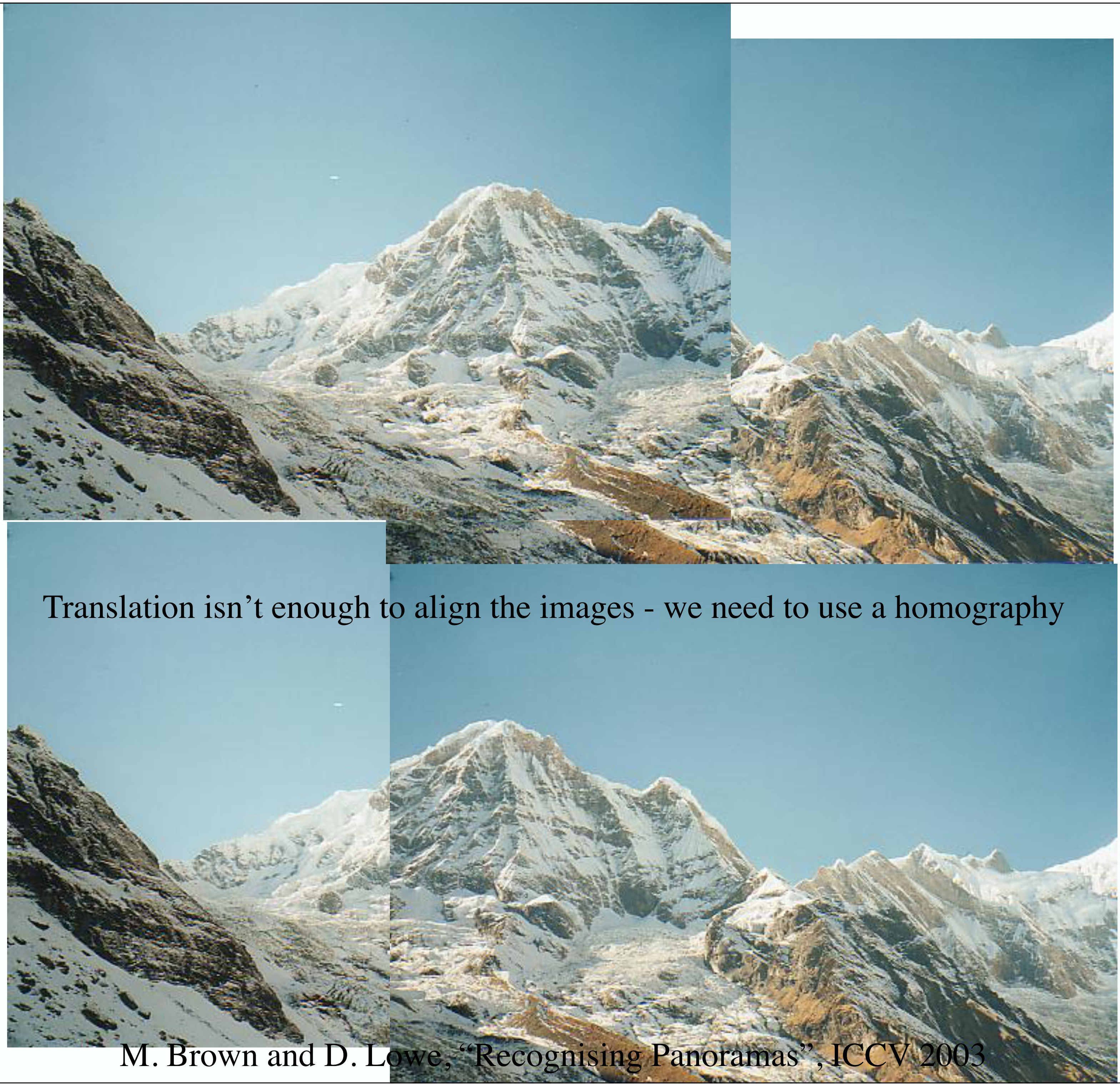


Fig 7 from:
Distinctive image features from scale-invariant keypoints
David G. Lowe, International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.



M. Brown and D. Lowe, “Recognising Panoramas”, ICCV 2003



Translation isn’t enough to align the images - we need to use a homography

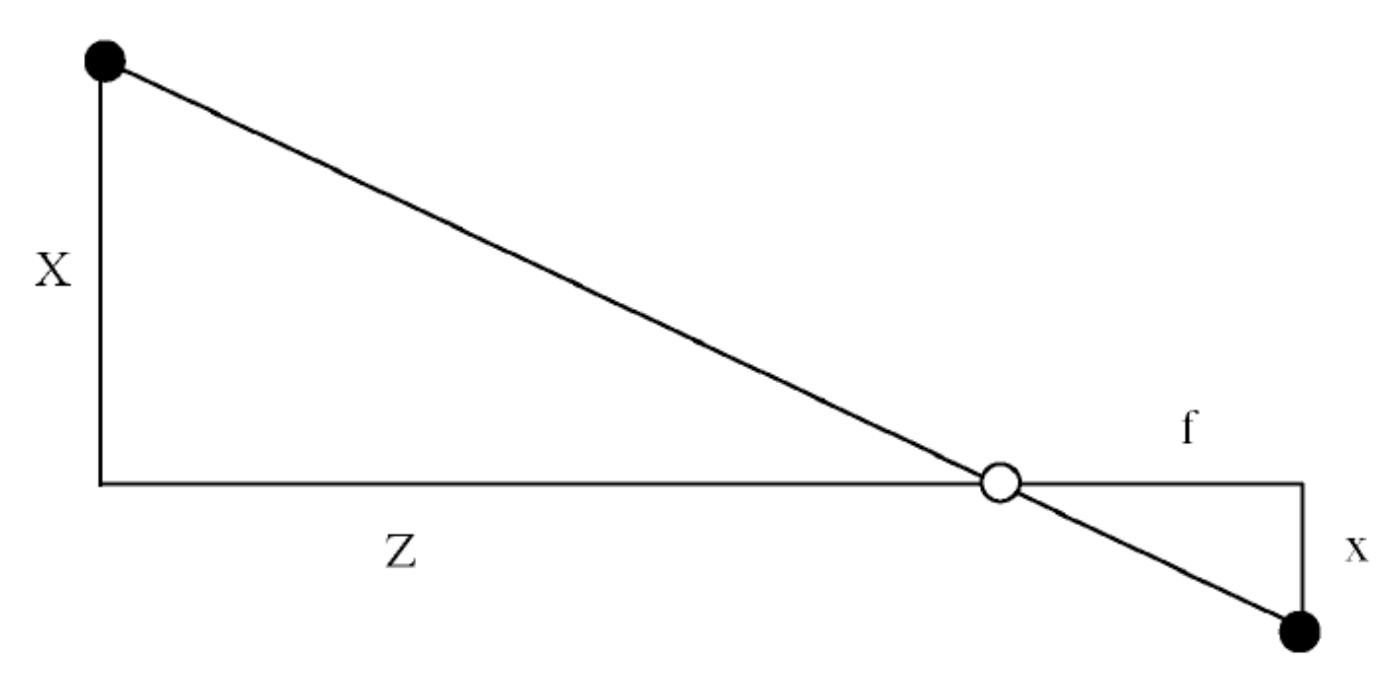
M. Brown and D. Lowe, “Recognising Panoramas”, ICCV 2003

Homographies

- Assume camera rotates about focal point
  - what happens to the image?
    - write camera as matrix, assume infinite image plane at z=-f

Projection in Coordinates

- From the drawing, we have X/Z = -x/f
- Generally



A perspective camera as a matrix

- Turn previous expression into HC's
  - HC's for 3D point are (X,Y,Z,T)
  - HC's for point in image are (U,V,W)

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

$$\mathcal{C}$$

A general perspective camera - I

- Can place a perspective camera at the origin, then rotate and translate coordinate system
- In homogeneous coordinates, rotation, translation are:

$$\mathcal{E} = \begin{pmatrix} \mathcal{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix}$$

- So rotated, translated camera is:

$$\mathcal{C}\mathcal{E}$$

A general perspective camera - II

- In the camera plane, there can be a change of coordinates
  - choice of origin
    - there is a “natural” origin --- the camera center
      - where the perpendicular passing through the focal point hits the image plane
  - rotation
  - pixels may not be square
  - scale

- Camera becomes

$$\mathcal{MCE}$$

Intrinsics - typically come with the camera

Extrinsics - change when you move around

What are the transforms?

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} \text{Transform} \\ \text{representing} \\ \text{intrinsic parameters} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \text{Transform} \\ \text{representing} \\ \text{extrinsic parameters} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

$$\begin{pmatrix} s & 0 & c_x \\ 0 & sa & c_y \\ 0 & 0 & s/f \end{pmatrix}$$

cx, cy - location of camera center

s - scale

a - aspect ratio

f - focal length

Homographies

- Camera 1 is

$$\begin{pmatrix} s & 0 & c_x \\ 0 & sa & c_y \\ 0 & 0 & s/f \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

- Camera 2 is

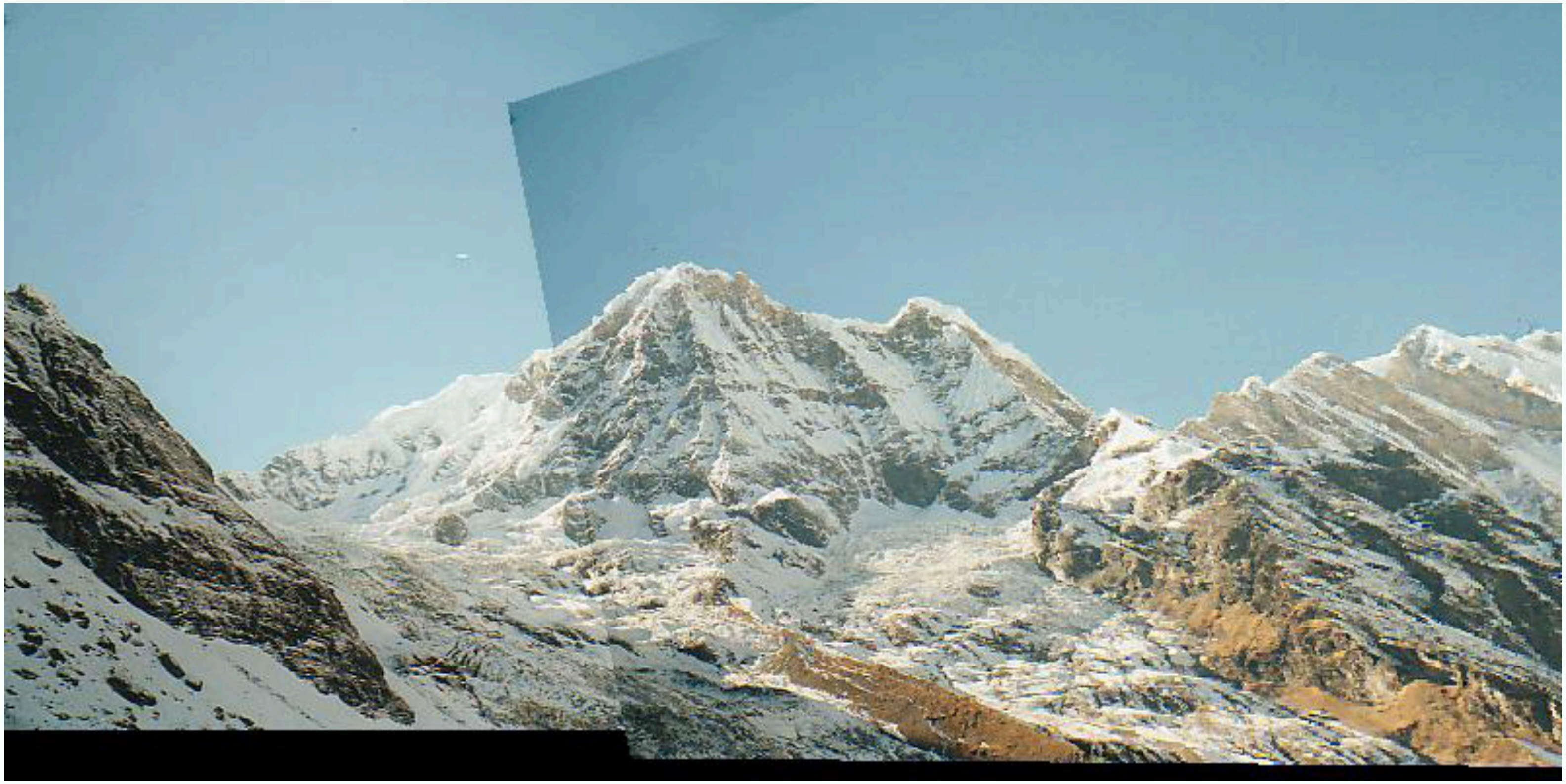
$$\begin{pmatrix} s & 0 & c_x \\ 0 & sa & c_y \\ 0 & 0 & s/f \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \mathcal{R} & \mathbf{t} \\ 0 & 1 \end{pmatrix}$$

Homographies

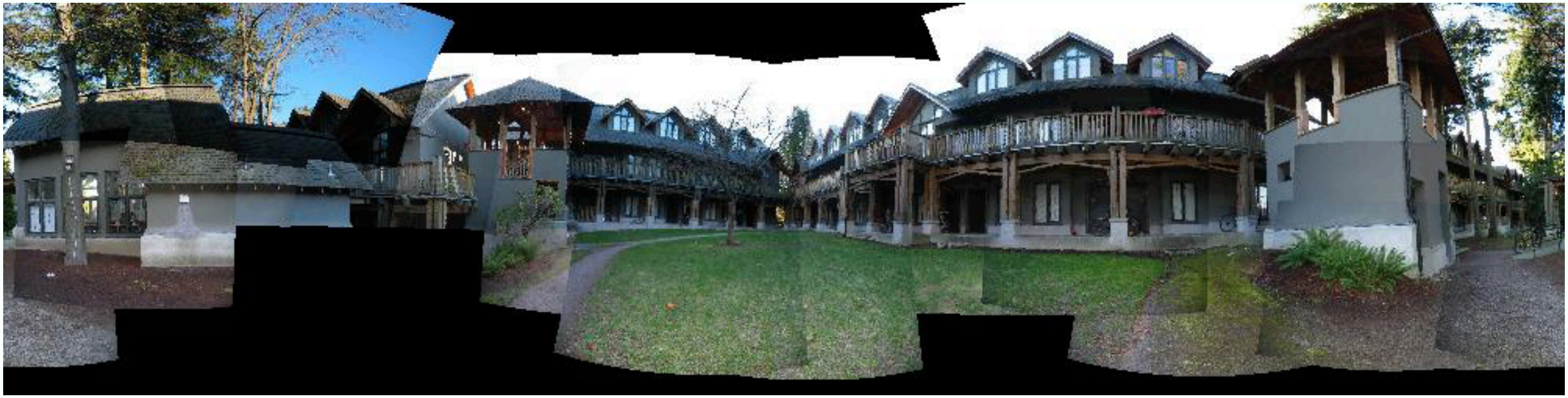
- There isn’t any translation, so 1 -> 2 is

$$\begin{pmatrix} s & 0 & c_x \\ 0 & sa & c_y \\ 0 & 0 & s/f \end{pmatrix} \mathcal{R} \begin{pmatrix} s & 0 & c_x \\ 0 & sa & c_y \\ 0 & 0 & s/f \end{pmatrix}^{-1}$$

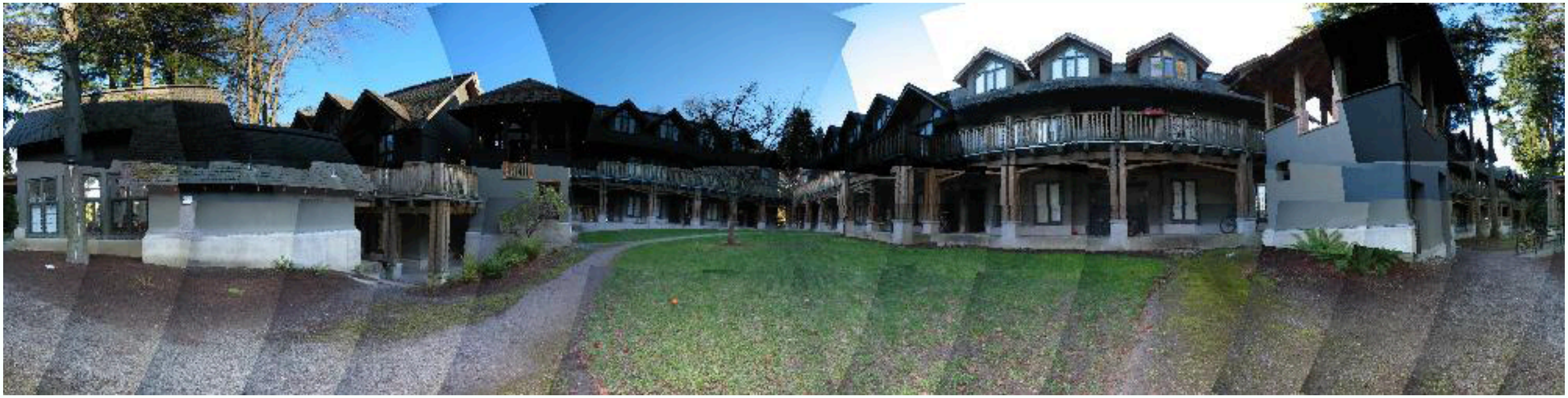
- How do we estimate?
  - linear least squares, followed by nonlinear least squares



M. Brown and D. Lowe, “Recognising Panoramas”, ICCV 2003



M. Brown and D. Lowe, “Recognising Panoramas”, ICCV 2003



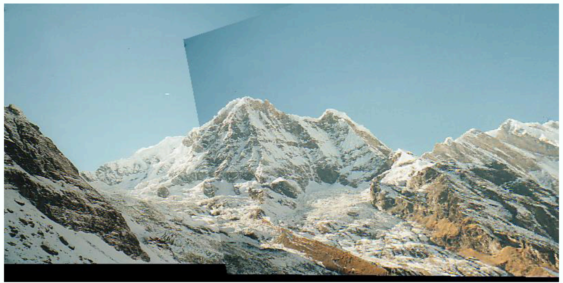
M. Brown and D. Lowe, “Recognising Panoramas”, ICCV 2003

Bundle adjustment

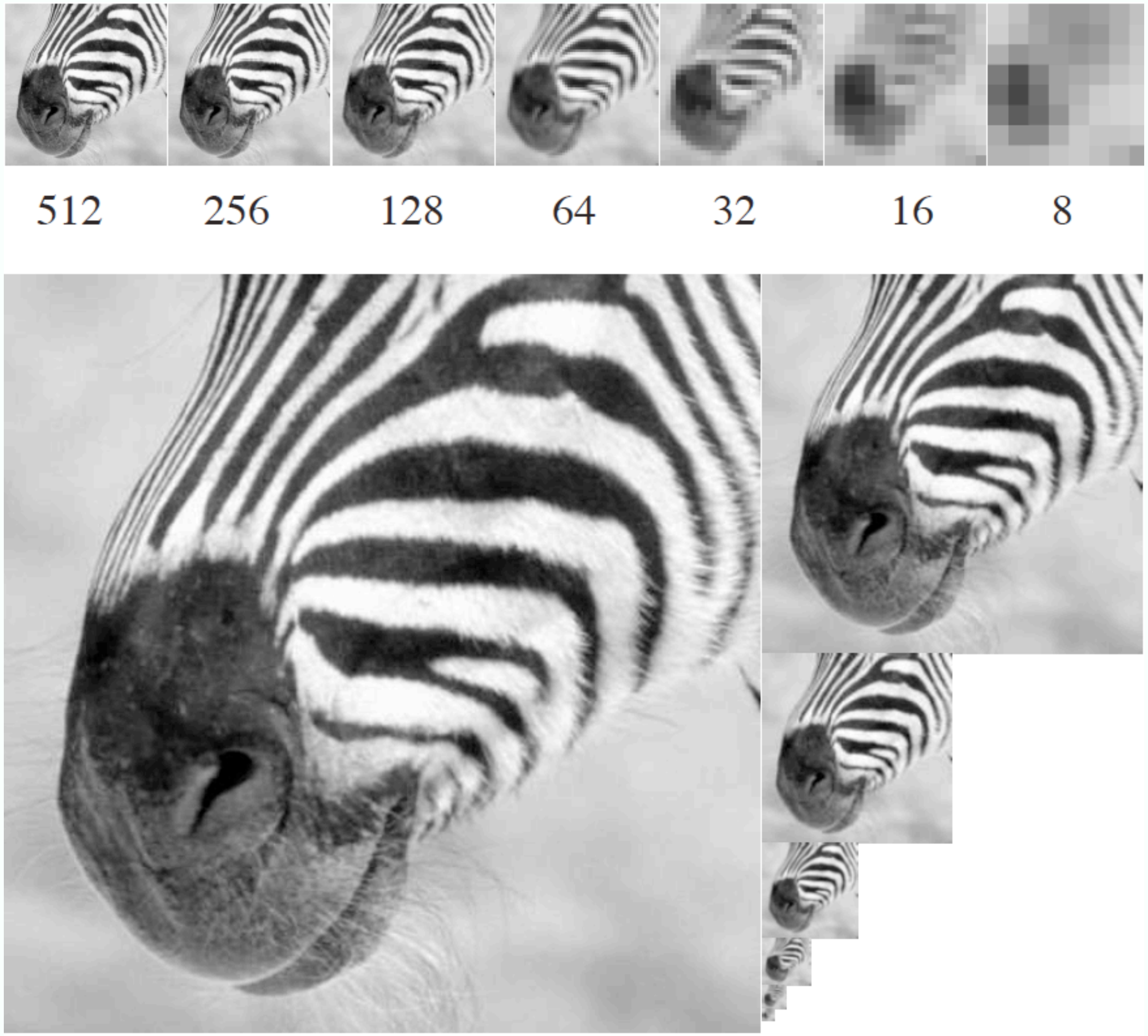
- Errors accumulate
  - so pairwise homographies will not join up to make a cylindrical mosaic
- Minimize all errors for all pairs of corresponding points
  - as a function of all parameters
  - start with pairwise estimates, use newton's method

Blending

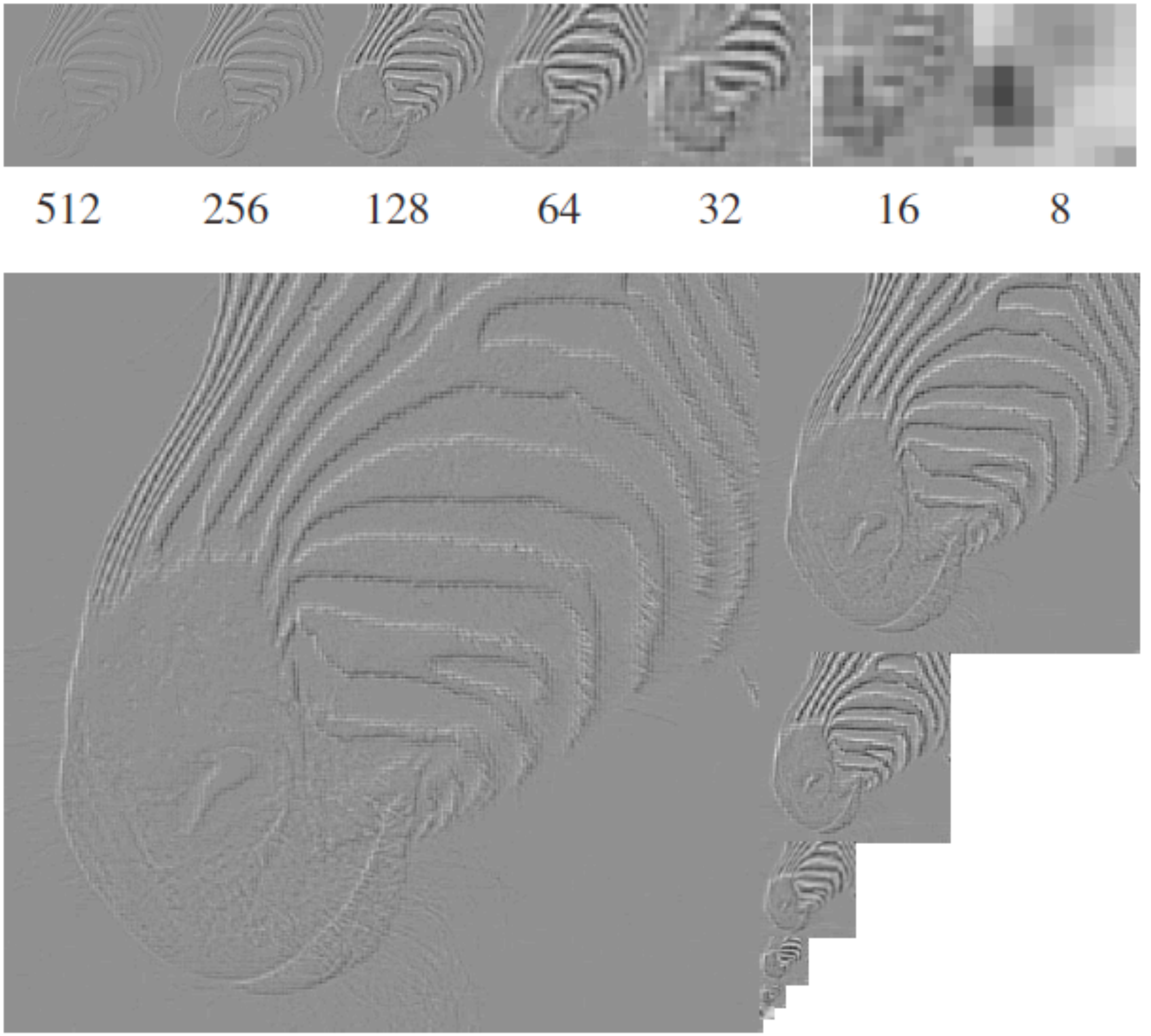
- Corresponding pixels aren’t always same color
  - aperture, sensitivity, etc., etc.
- Blend for consistency
  - pixels “far” from camera center are less reliable
  - Strategy:
    - weight with distance from camera center, then blend
      - fuzzes out small details
  - Strategy
    - separate bands
    - blend low spatial frequencies like this
    - high spatial frequencies from image with most weight

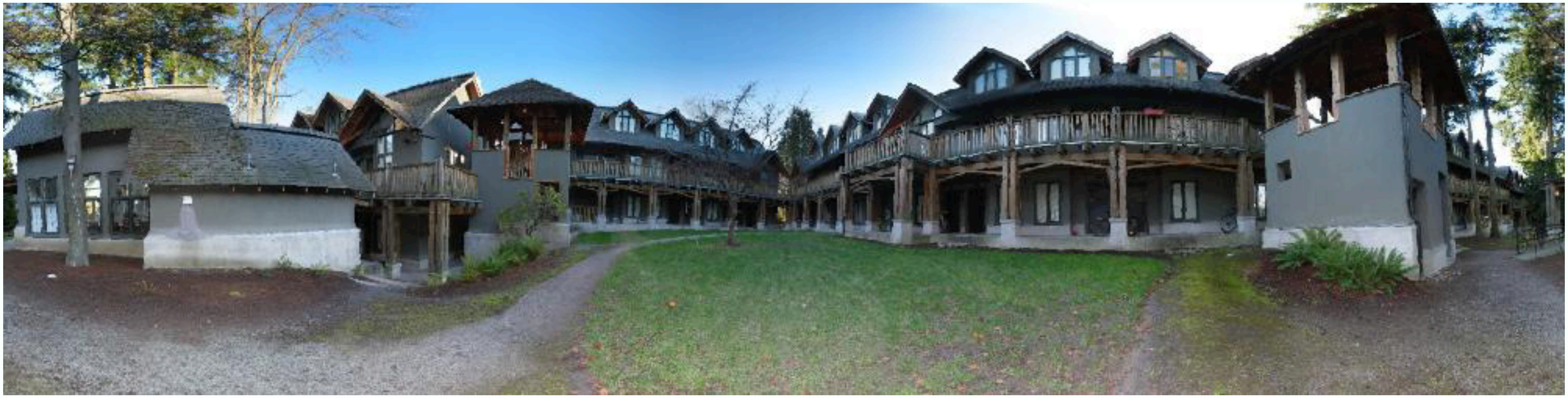


Gaussian Pyramid



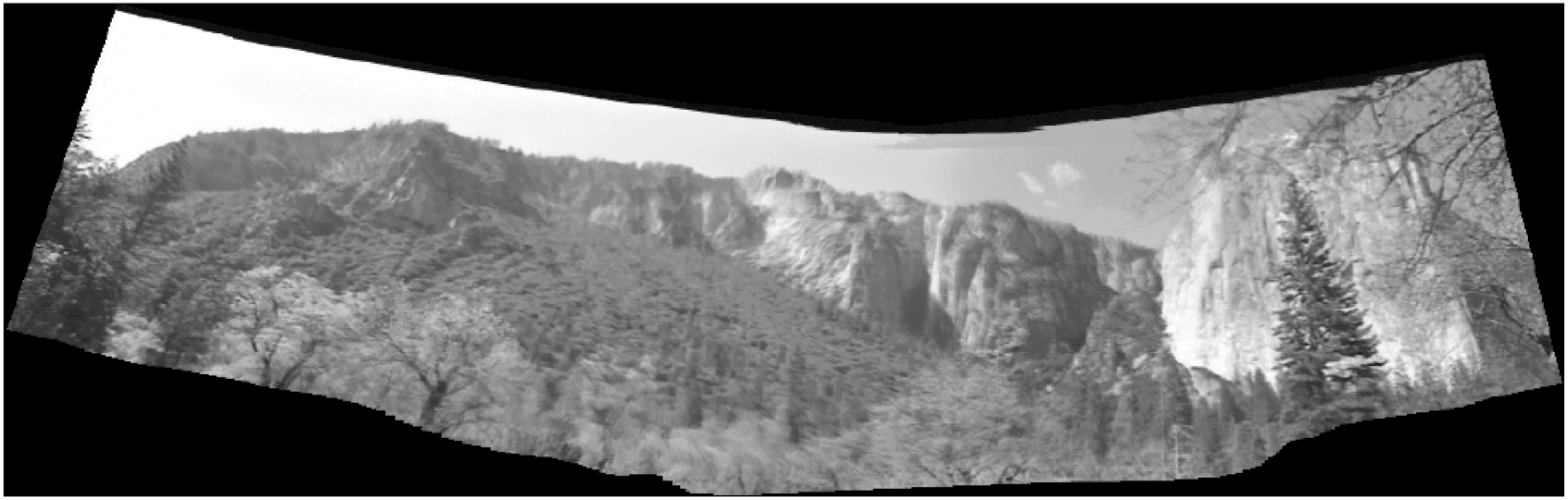
Laplacian pyramid

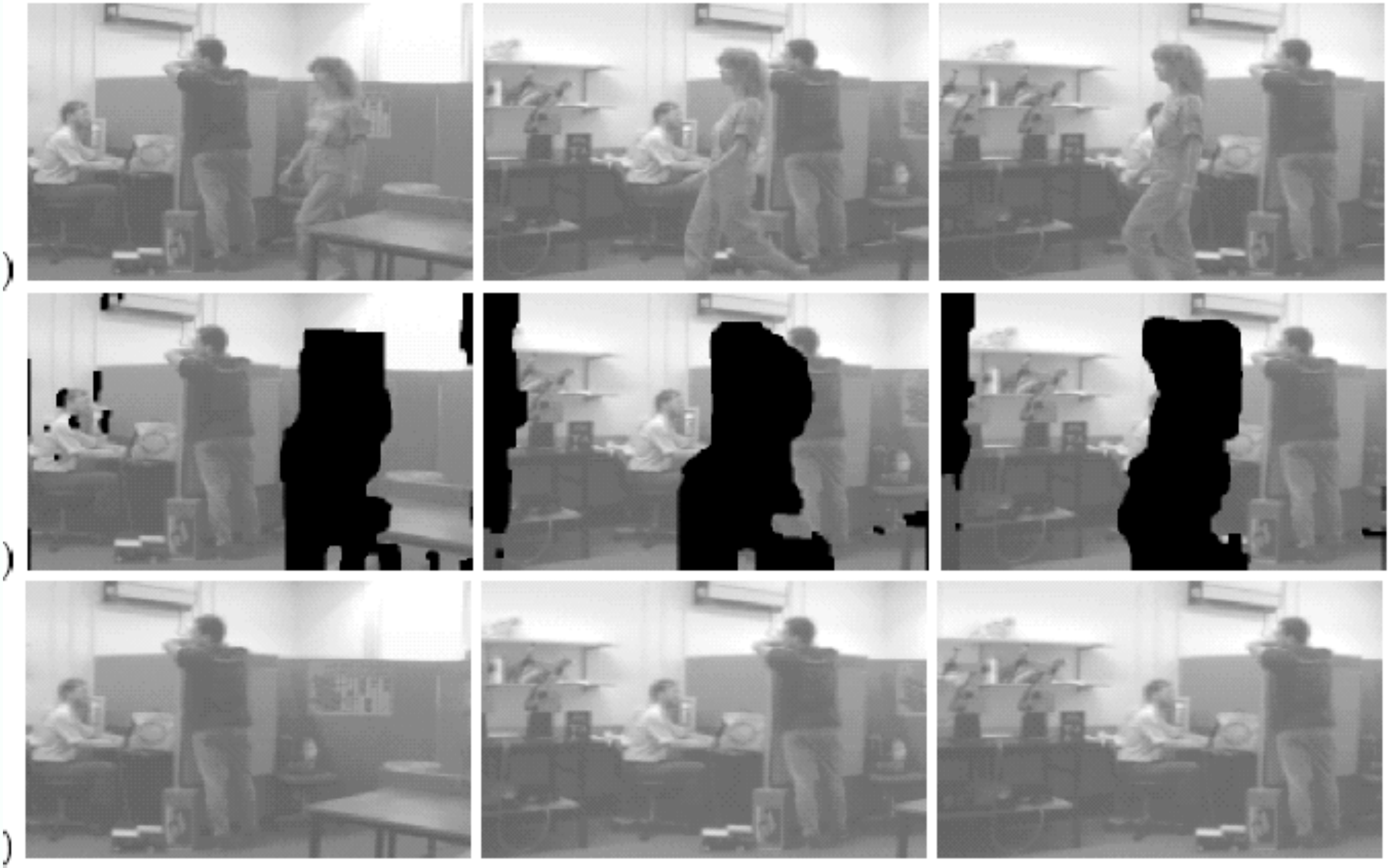




M. Brown and D. Lowe, “Recognising Panoramas”, ICCV 2003

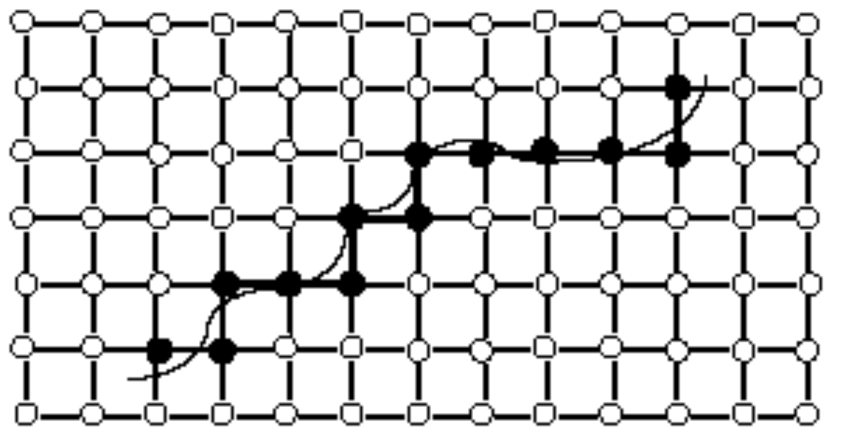
MAY 19 1995





Optical Flow

- Local motion “at a pixel”
  - Arrow joins pixel in this frame to corresponding pixel in next frame
    - hard to estimate accurately from images
      - but easy to predict for small movements of the head, known geom



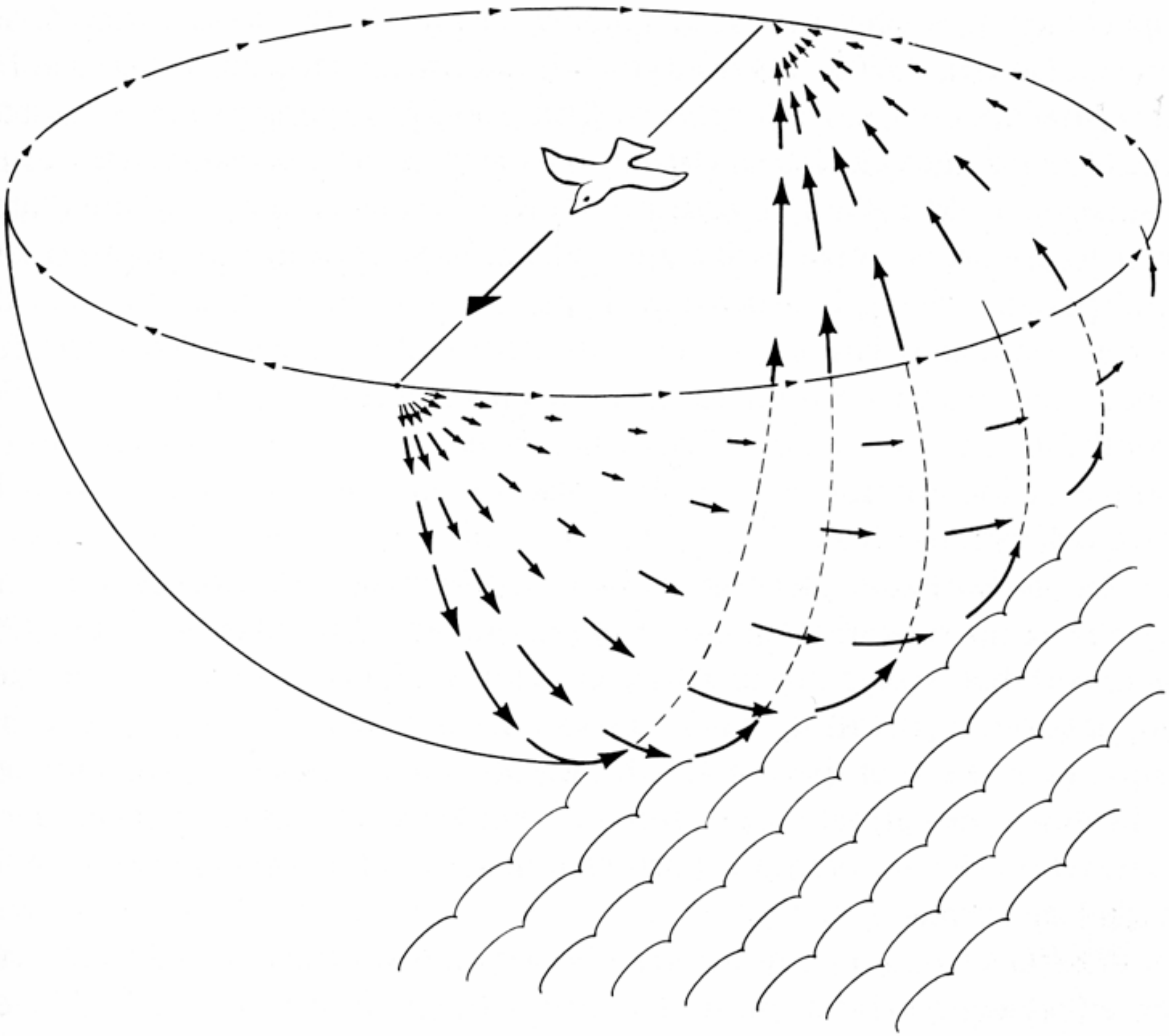
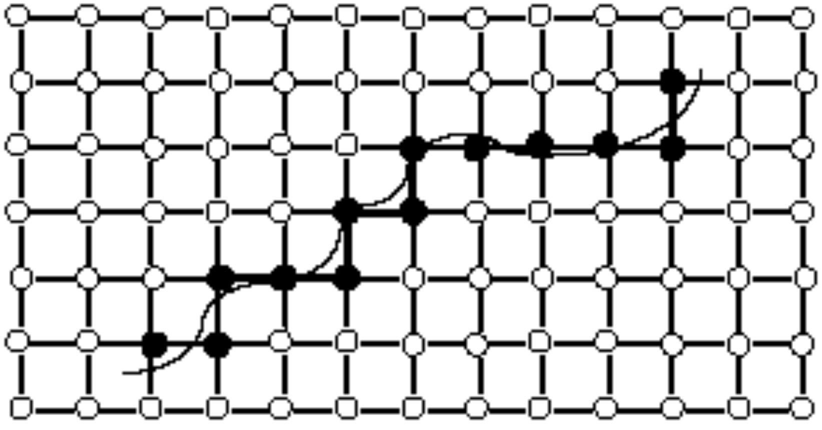


Figure 3–55. Gibson's example of flow induced by motion. The arrows represent angular velocities, which are zero directly ahead and behind. (Reprinted from J. J. Gibson, *The Senses Considered as Perceptual Systems,* Houghton Mifflin, Boston, 1966, fig. 9.3. Copyright © 1966 Houghton Mifflin Company. Used by permission.)

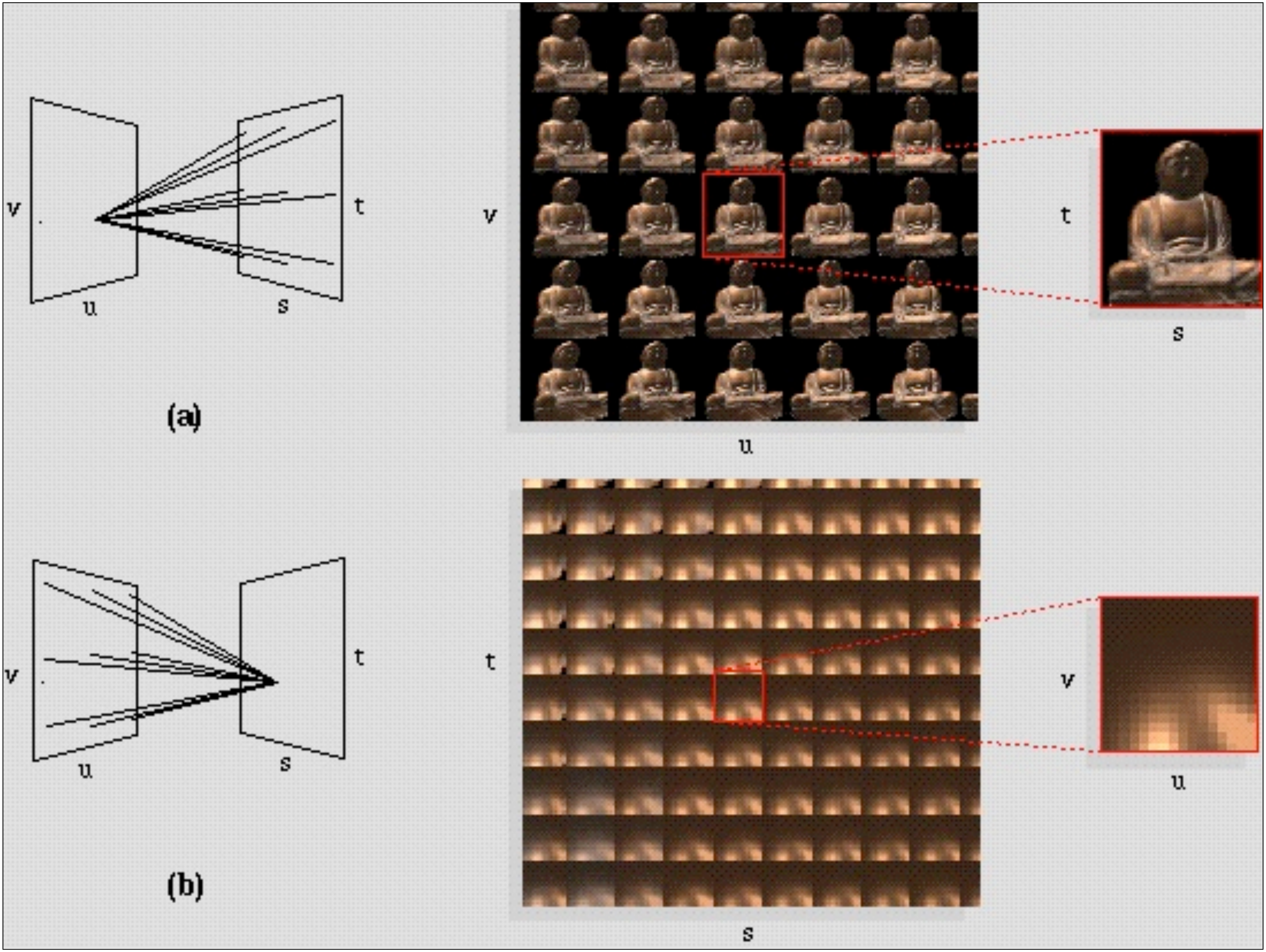
Optical flow



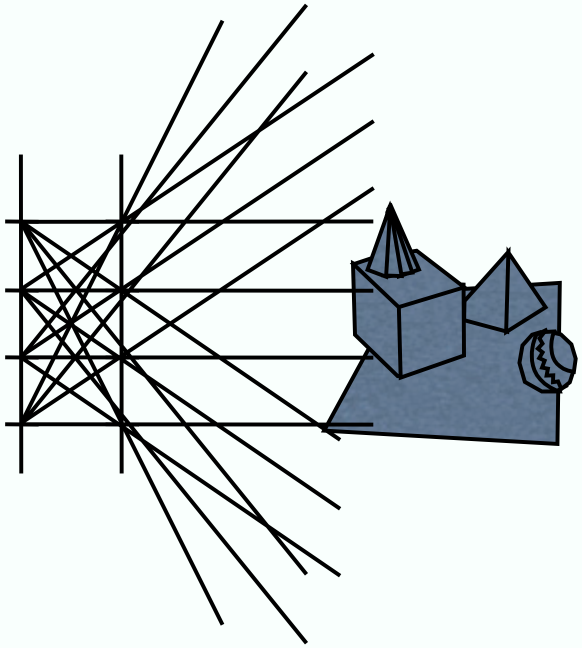
- Compute flows produced by moving
  - with vision methods, using geometry constraints we haven't done yet
  - interpolate along flow to produce intermediate images

Explicit image based rendering

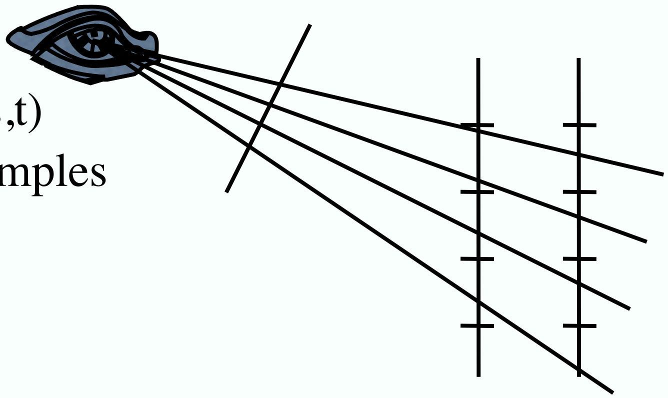
- Put object “in a box”
- Evaluate every light ray through the box
  - four dimensional family
  - by taking lots of photographs
- Render
  - query this structure
  - using any ray tracing alg we know


v
t
u
s
(a)
v
u
t
s
v
t
u
s
(b)
t
s
v
u

Rendering and light fields



- Rendering into a light field
  - Cast rays between all pairs of points in panes
  - Store resulting radiance at (u,v,s,t)

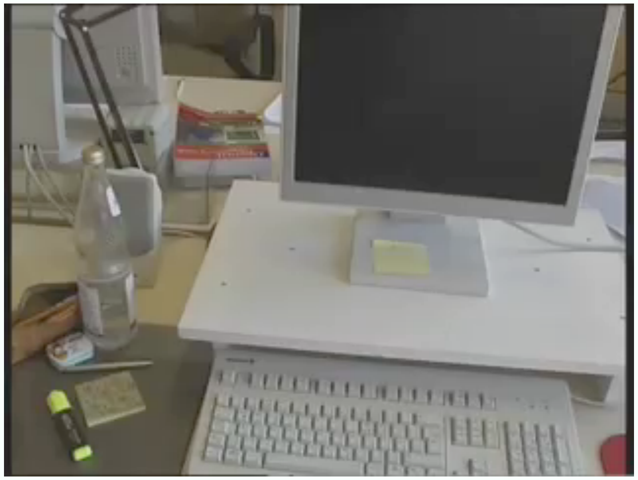


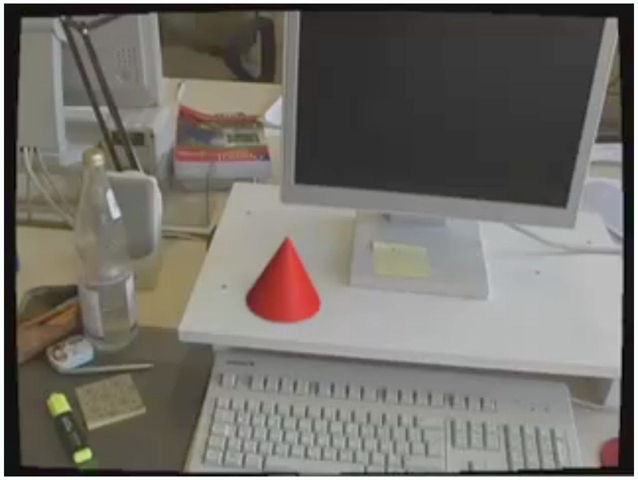
- Rendering from a light field
  - Cast rays through pixels into light field
  - Compute two ray-plane intersections to find (u,v,s,t)
  - Interpolate u,v and s,t to find radiance between samples
  - Plot radiance in pixel

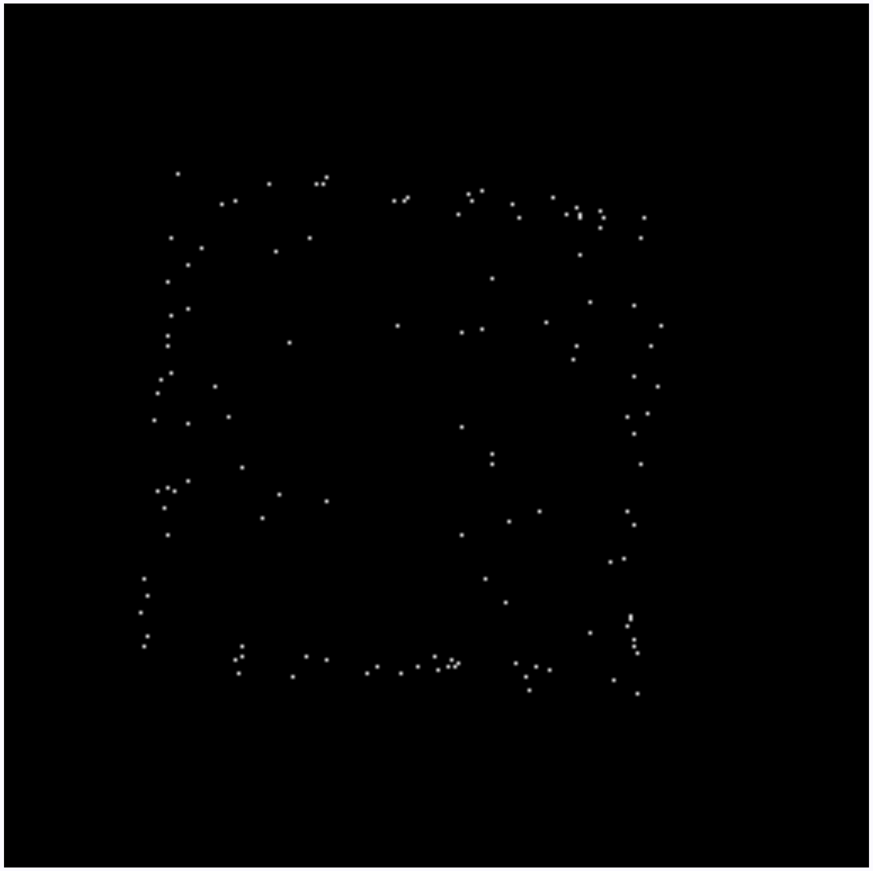
Ren Ng’s camera

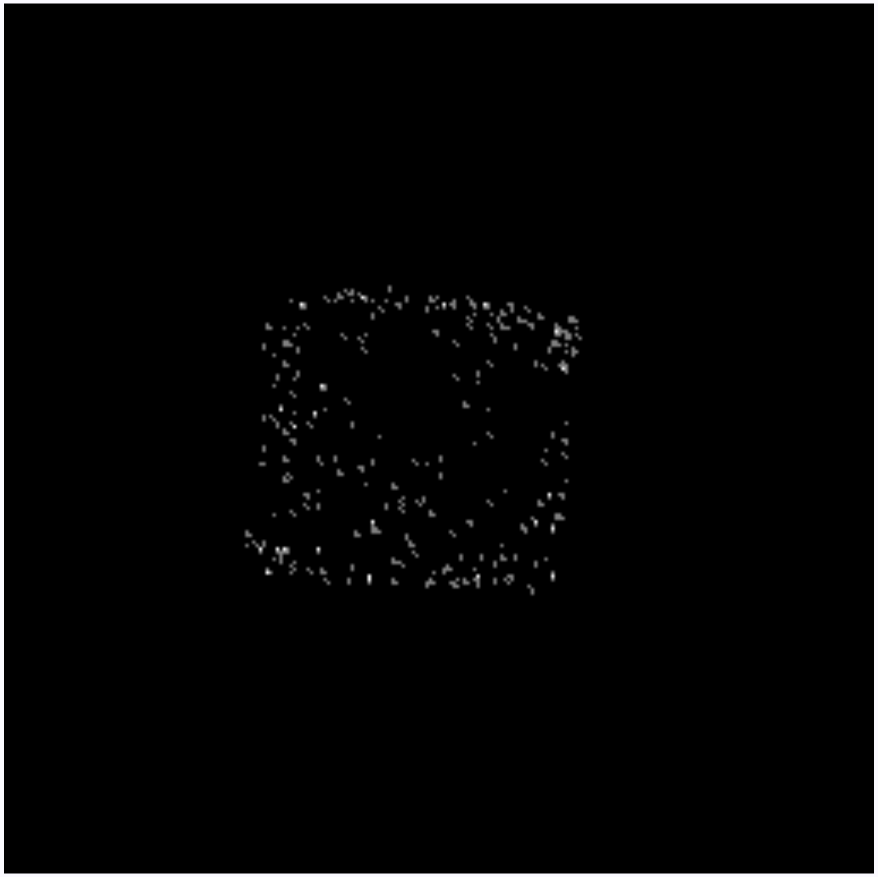
Postrendering into images, video

- Options
  - Insert a calibration object, calibrate camera, use this info to render
    - problem: calibration object in picture
  - (video) reconstruct world points, camera, render using camera
    - we’ll discuss this shortly









Reconstruction from more than two views

- Further geometric constraint on triples of views
  - all other constraints are redundant, given these
- Simplest case:
  - assume an orthographic camera
  - assume we see all points in all views
  - assume we know all point-point correspondences

Orthographic cameras

- Model is:

Homogeneous
coordinates
in 3D

$$\mathbf{x} = \begin{pmatrix} s & 0 & t_x \\ 0 & s & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathcal{R} & \mathbf{t} \\ 0 & 1 \end{pmatrix} \mathbf{X}$$

Homogeneous
coordinates
in image

Intrinsics:
scale and translation

Projection

Extrinsics:
rotation and translation

Simplify

- Place the 3D origin at center of gravity of points
  - ie mean of x is zero, mean of y is zero, mean of z is zero
- Place the image origin at center of gravity of image points
  - we see all of them, so we can compute this
  - this is the projection of 3D mean
- Now camera becomes

$$\mathbf{x} = \begin{pmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathcal{R} & 0 \\ 0 & 1 \end{pmatrix} \mathbf{X}$$

Drop HC's, simplify

- Model

$$x_{im} = \mathbf{v}_x^T \mathbf{X}_{3D}$$
$$y_{im} = \mathbf{v}_y^T \mathbf{X}_{3D}$$

- constraints

$$\mathbf{v}_x^T \mathbf{v}_y = 0$$
$$\mathbf{v}_x^T \mathbf{v}_x - \mathbf{v}_y^T \mathbf{v}_y = 0$$

Multiple views

$$x_{i,j} = \mathbf{v}_{x,i}^T \mathbf{X}_j$$
$$y_{i,j} = \mathbf{v}_{y,i}^T \mathbf{X}_j$$

Point index is j

View index is i

Multiple views

$$\begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \cdots & & & \\ y_{m,1} & y_{m,2} & \cdots & y_{m,n} \\ y_{1,1} & y_{1,2} & \cdots & y_{1,n} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,n} \\ \cdots & & & \\ y_{m,1} & y_{m,2} & \cdots & y_{m,n} \end{pmatrix} = \begin{pmatrix} \mathbf{v}_{x,1}^T \\ \mathbf{v}_{x,2}^T \\ \cdots \\ \mathbf{v}_{x,m}^T \\ \mathbf{v}_{y,1}^T \\ \mathbf{v}_{y,2}^T \\ \cdots \\ \mathbf{v}_{y,m}^T \end{pmatrix} \begin{pmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \ldots & \mathbf{X}_n \end{pmatrix}$$

$$\mathcal{D} = \mathcal{V}\mathcal{X}$$

Data - observed!

Multiple views

- The data matrix has rank 3
  - so we can factor it into an mx3 factor and a 3xn factor
- But this isn't enough to get camera, points
  - there could be an "internal transformation" as below
  - which creates ambiguities, because it transforms points and cameras

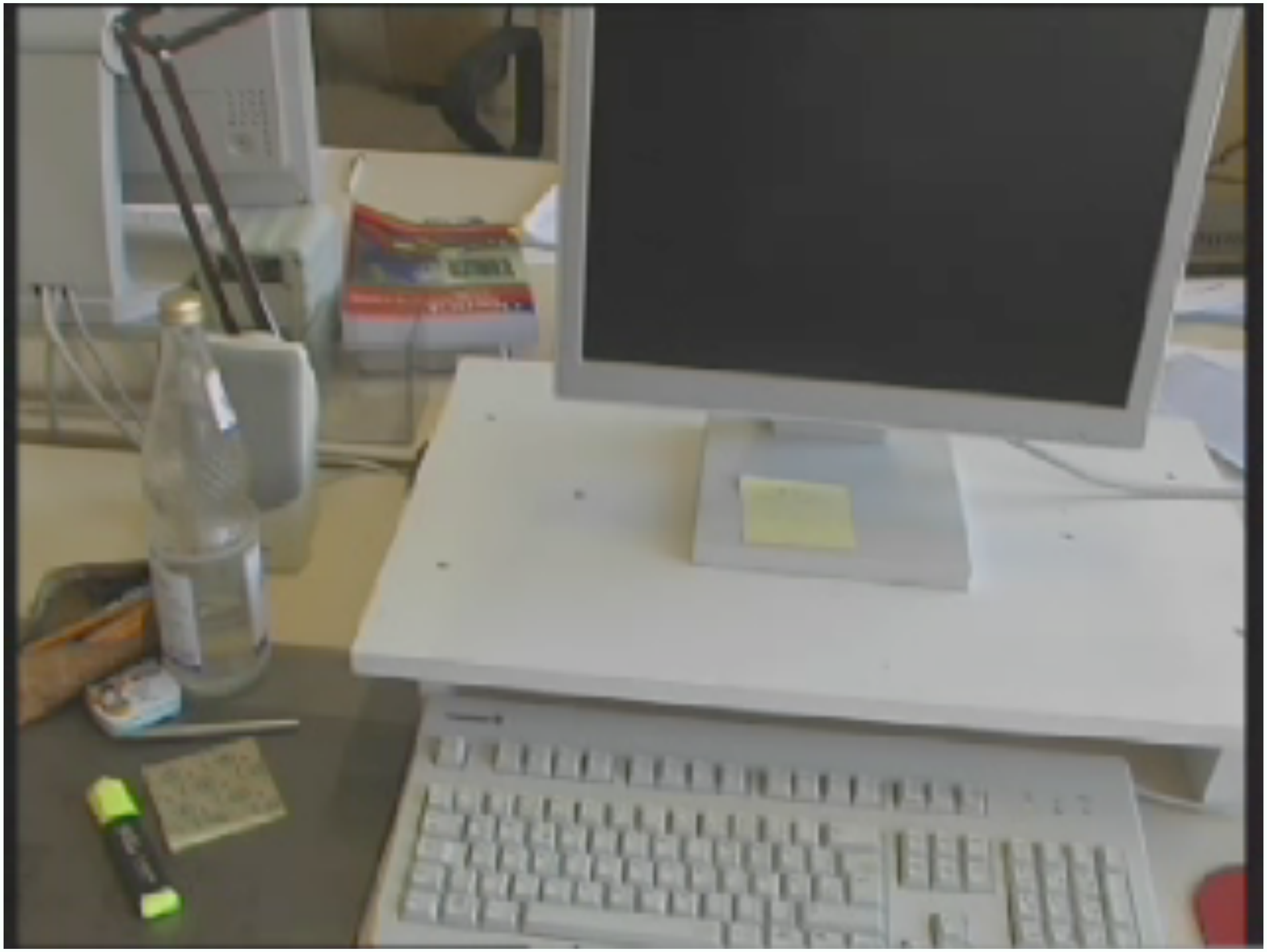
$$\mathcal{D} = \mathcal{V}\mathcal{X} = (\mathcal{V}\mathcal{A})(\mathcal{A}^{-1}\mathcal{X})$$

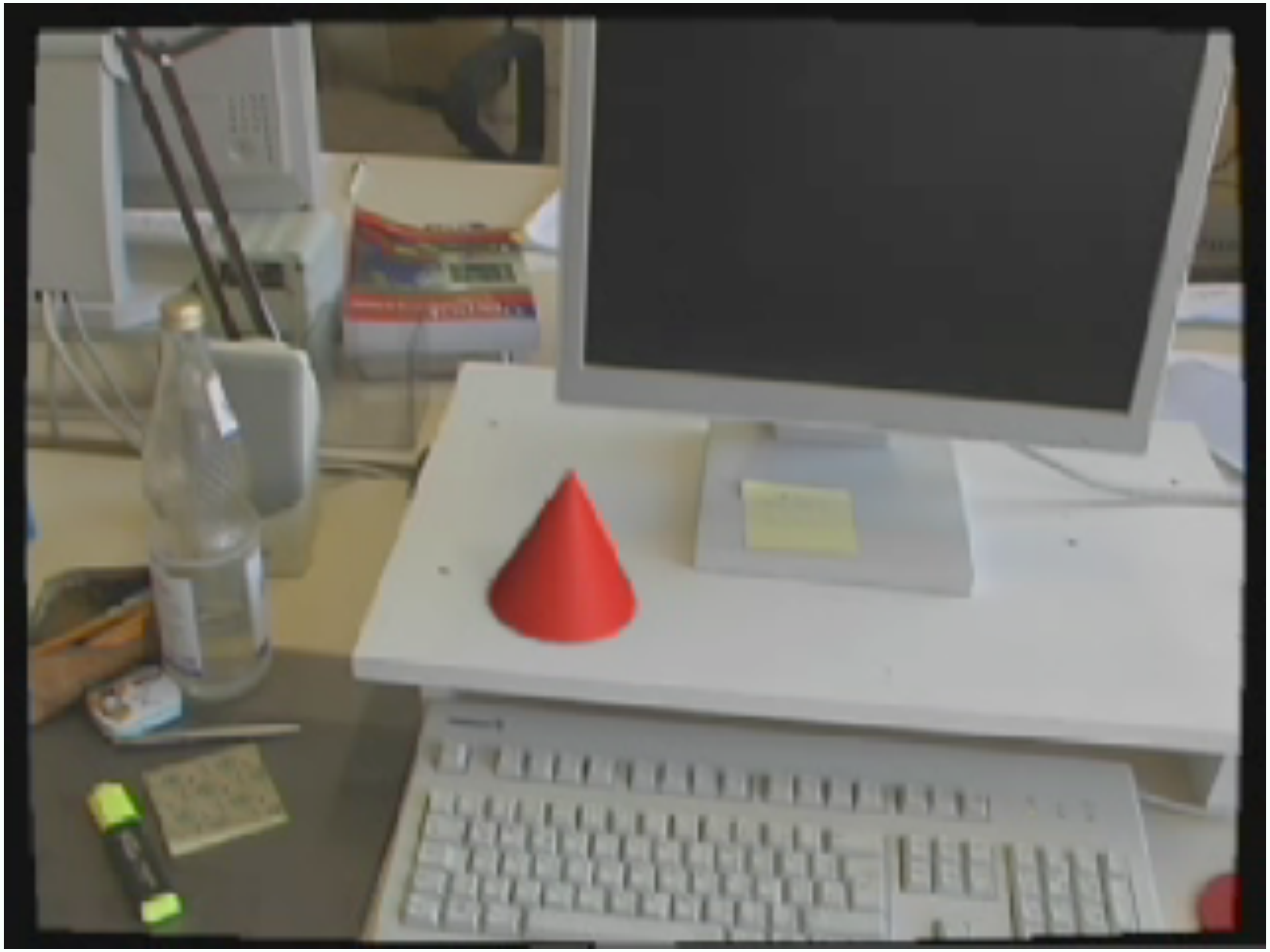
- BUT
  - we know some constraints on the rows of V
    - for each camera, we have constraints from previous slides
    - choose an A that makes these "as close as possible" to true
    - least squares

$$\mathbf{v}_x^T \mathbf{v}_y = 0$$

$$\mathbf{v}_x^T \mathbf{v}_x - \mathbf{v}_y^T \mathbf{v}_y = 0$$

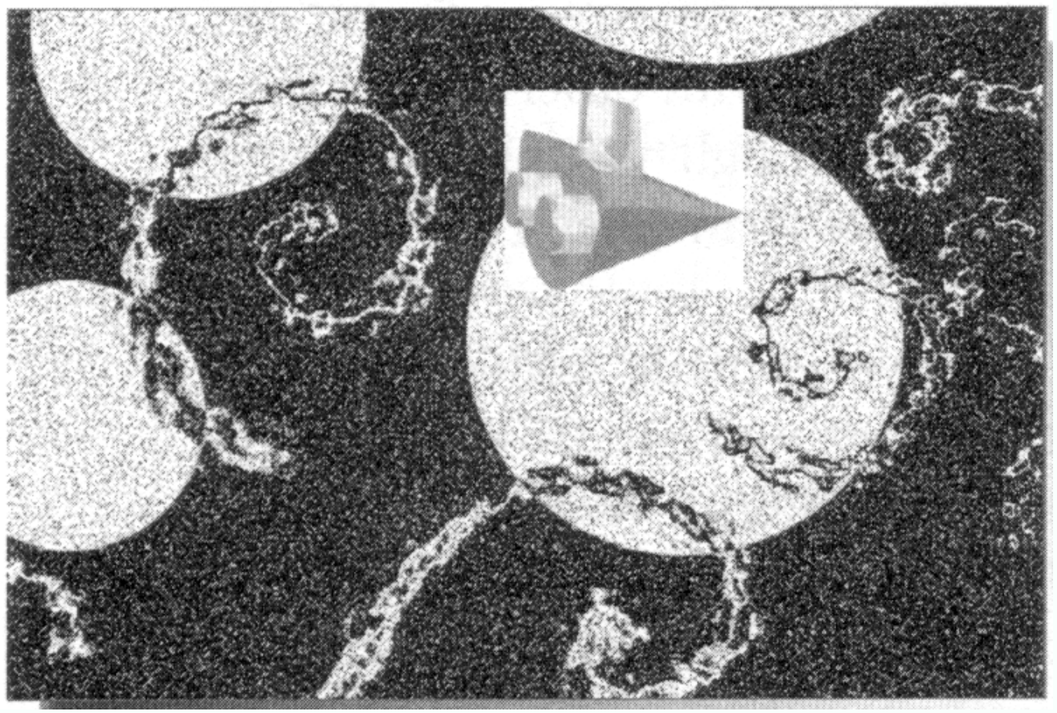
M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, R. Koch, Visual modeling with a hand-held camera, International Journal of Computer Vision 59(3), 207-232, 2004





Compositing

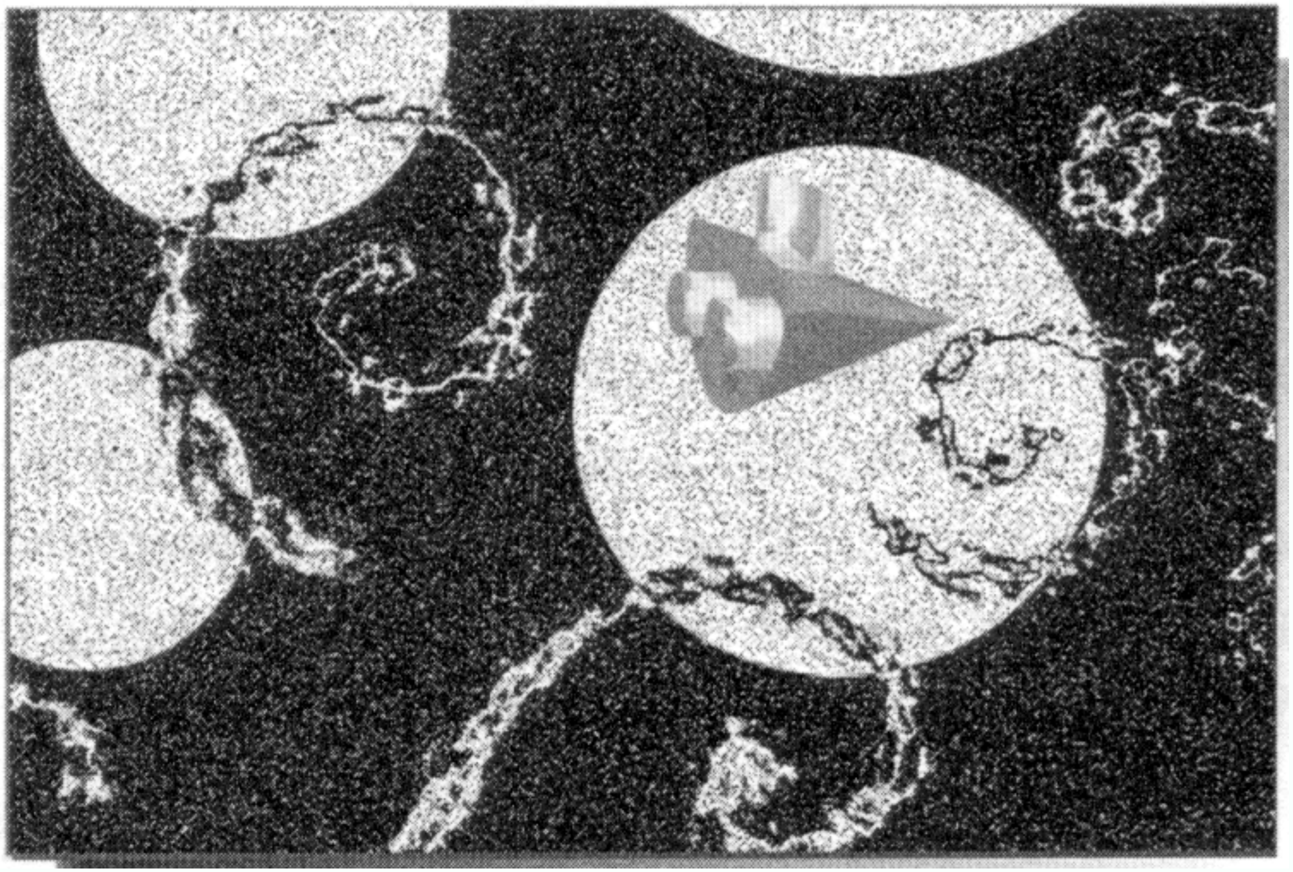
- Overlay one image/film on another
  - variety of types of overlay



Simple overlay - spaceship pixels replace background pixels

From “The computer in the visual arts”, Spalter, 1999

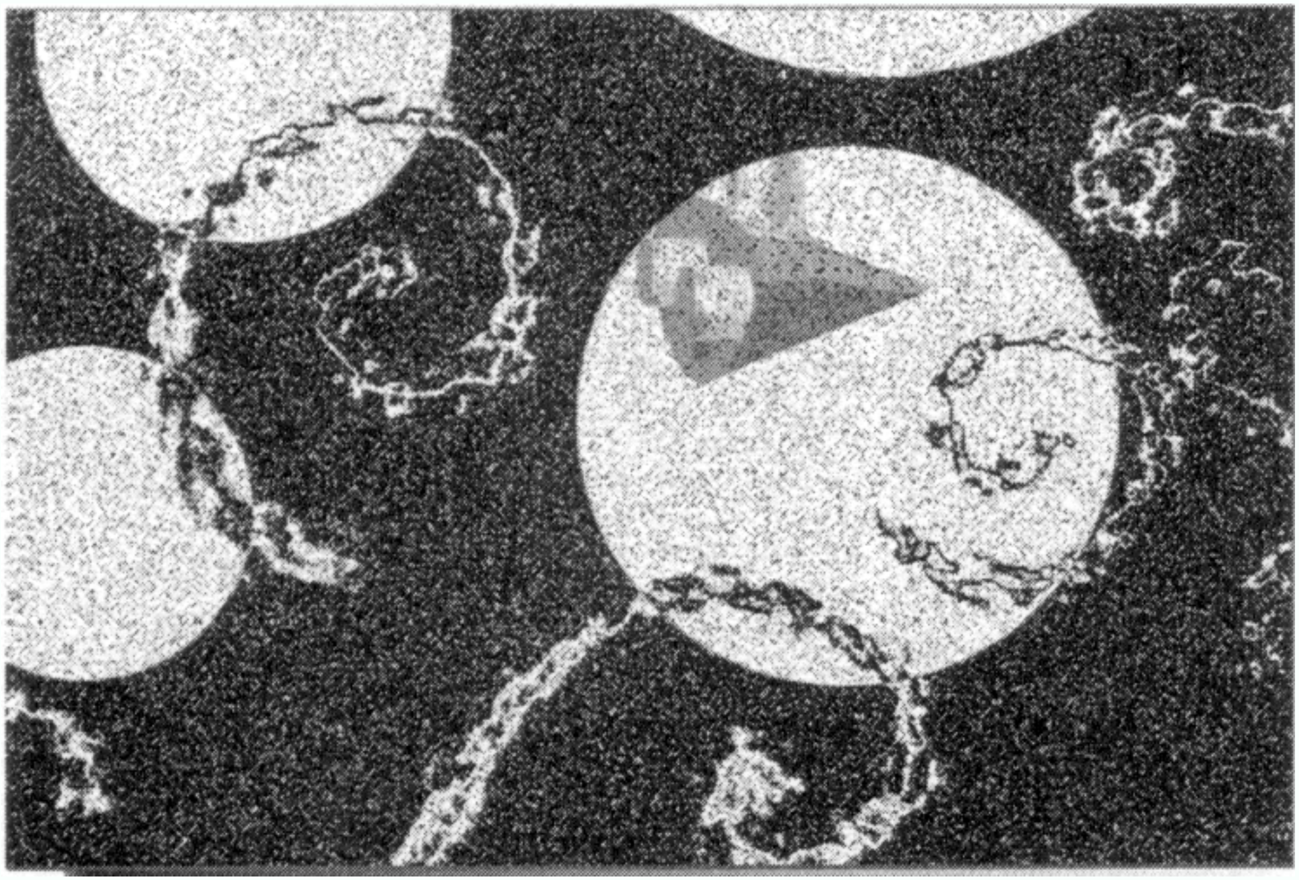
Compositing



Spaceship pixels replace background pixels if they are not white (white is “dropped out”)

From “The computer in the visual arts”, Spalter, 1999

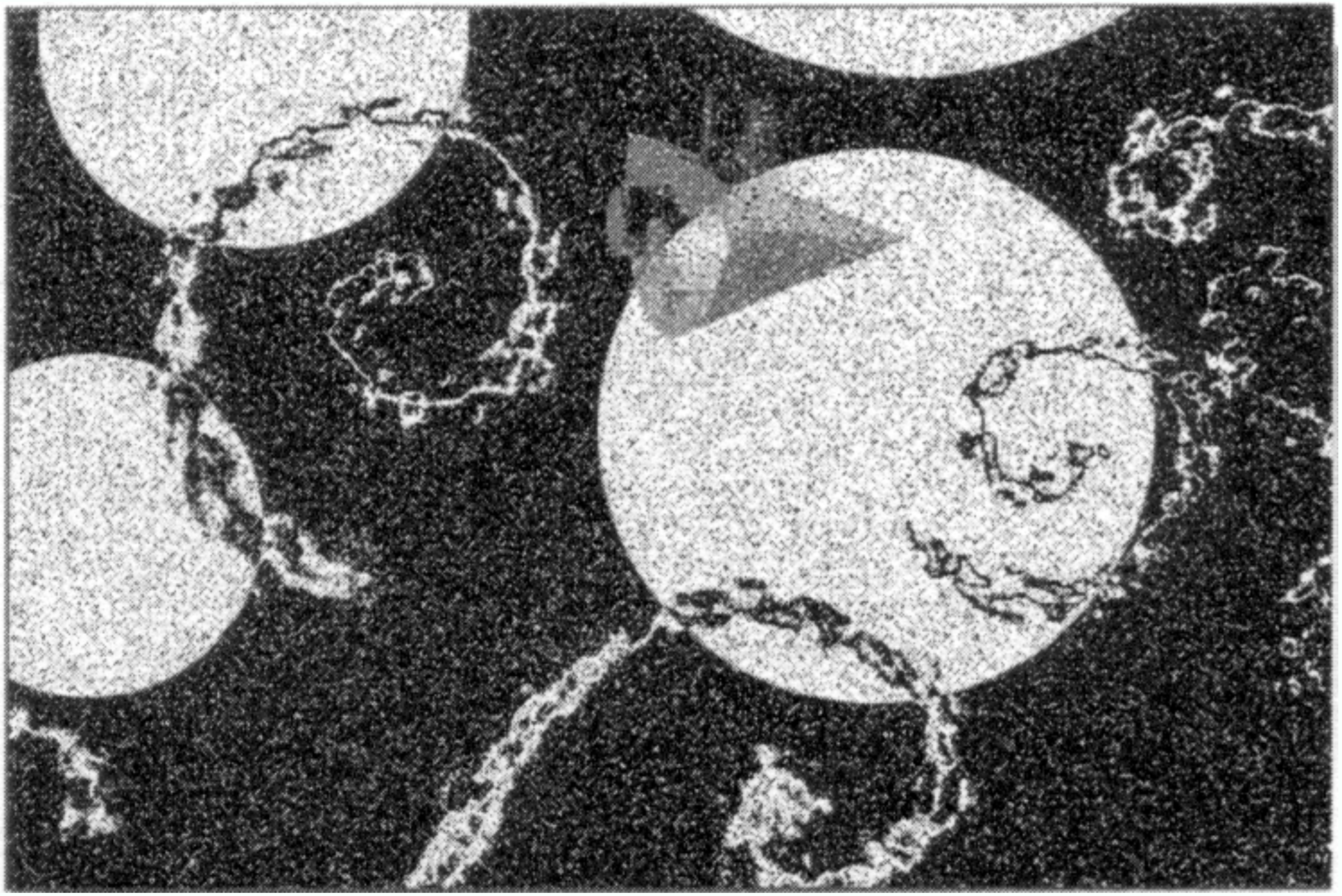
Compositing



Spaceship pixels replace background pixels if they are darker

From “The computer in the visual arts”, Spalter, 1999

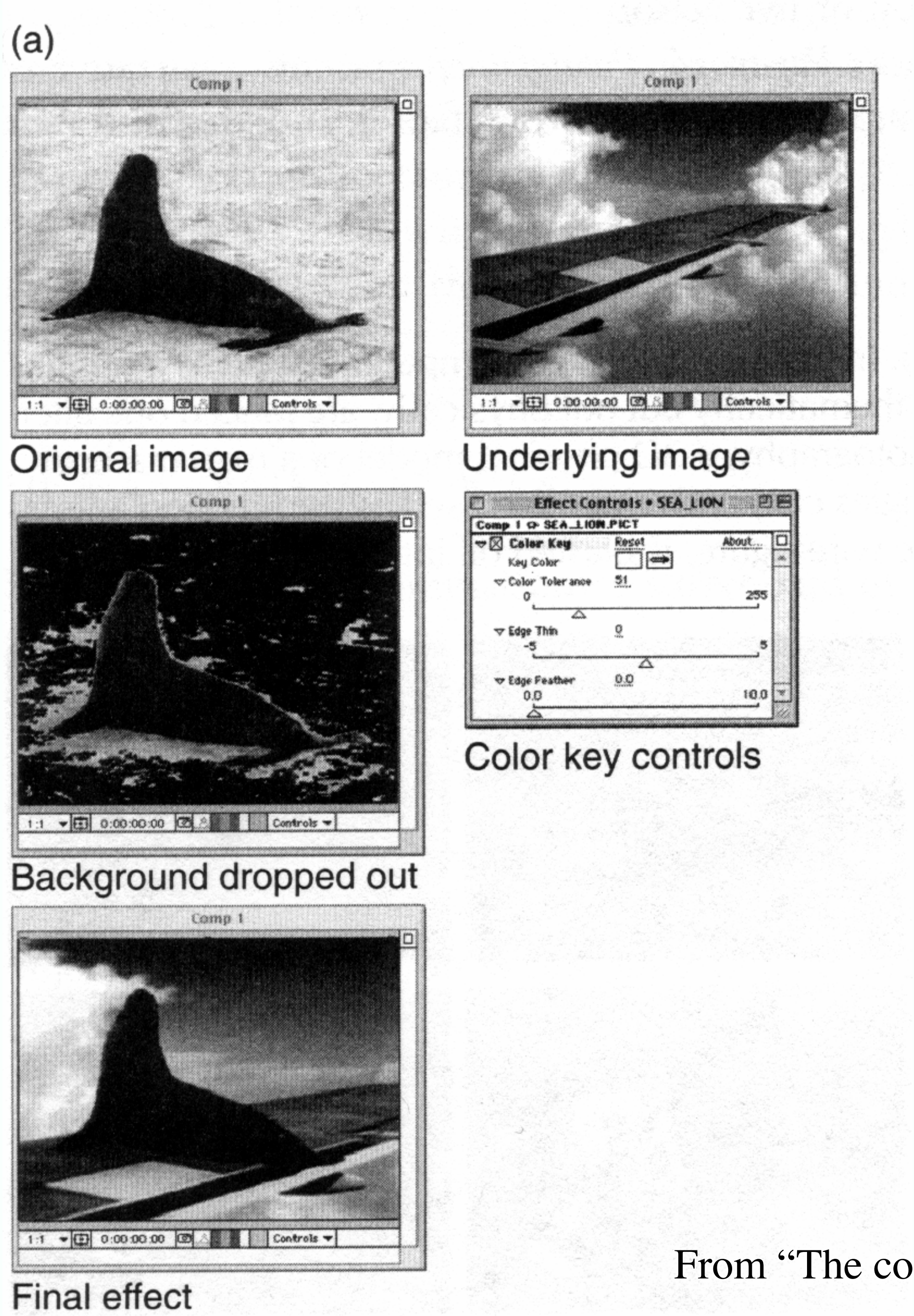
Compositing



Light areas are more transparent - blending

From “The computer in the visual arts”, Spalter, 1999

Compositing



Original image

Underlying image

Background dropped out

Final effect

Color key controls

Effect Controls • SEA_LION

Comp 1 • SEA_LION.PICT

Color Key Reset About...

Key Color

Color Tolerance 51

0 255

Edge Thin 0

-5 5

Edge Feather 0.0

0.0 10.0

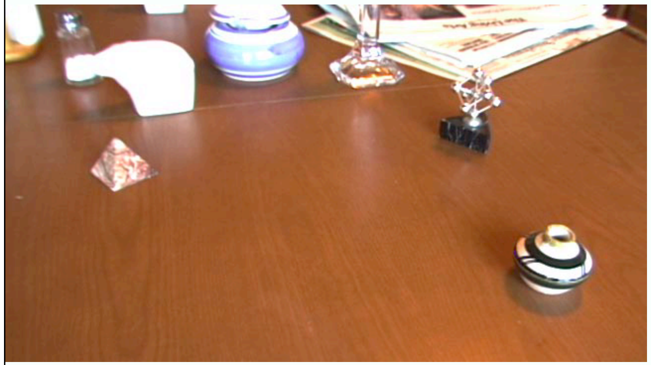
- Note that human intervention might be required to remove odd pixels, if the background doesn't have a distinctive colour
- One can buy sets of images which have been segmented by hand.

From "The computer in the visual arts", Spalter, 1999

Compositing

- Recall image relighting notes
  - we want to insert an object into a scene
  - we have
    - background scene   image is: B
    - model of background scene   image is: Mn
    - model of object in background scene   image is: Mo
- Composite by:
  - at model pixels
    - B+(Mo-Mn)
  - at object pixels
    - Mo
  - at background pixels
    - B

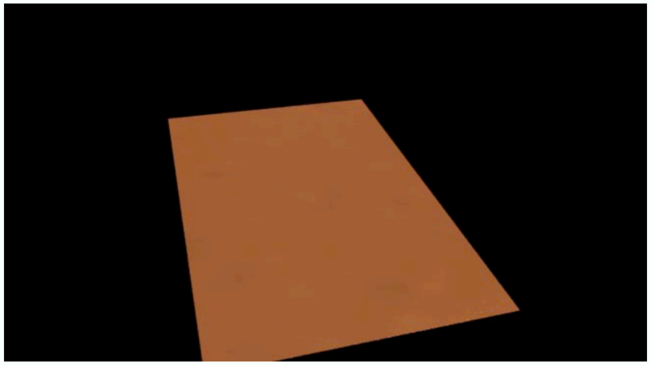
Compositing

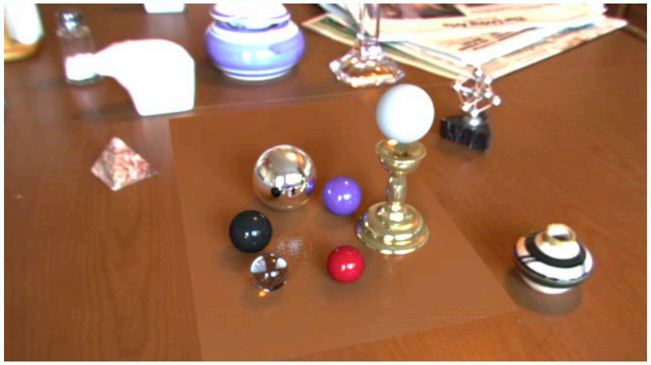


Background image B

Figures from Debevec,
Rendering Synthetic Objects
into Real Scenes:
Bridging Traditional and
Image-based Graphics with
Global Illumination
and High Dynamic Range
Photography1998

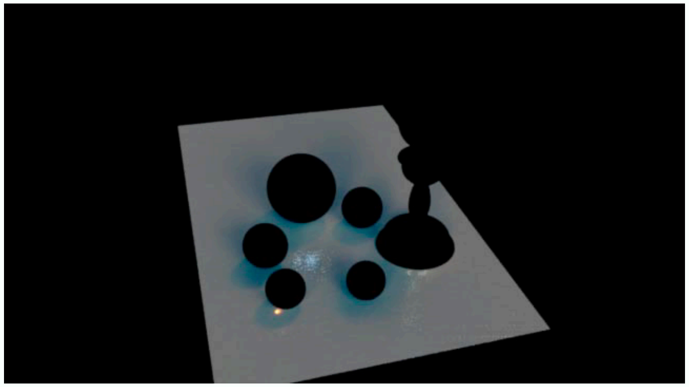
Background model Mn



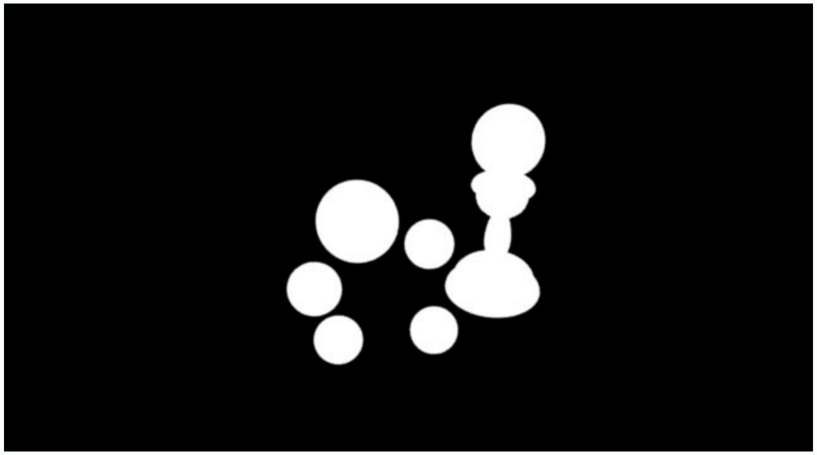


Background model,
rendered with objects Mo,
superimposed on B

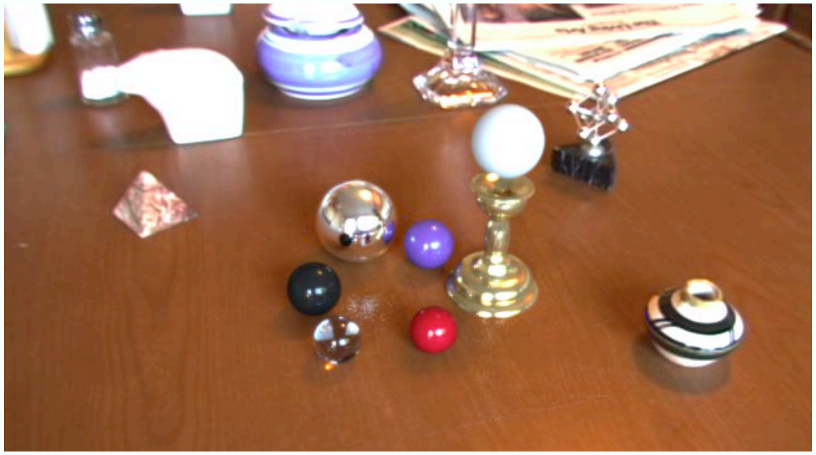
Mo-Mn
in non-object, non-
background pixels



Compositing



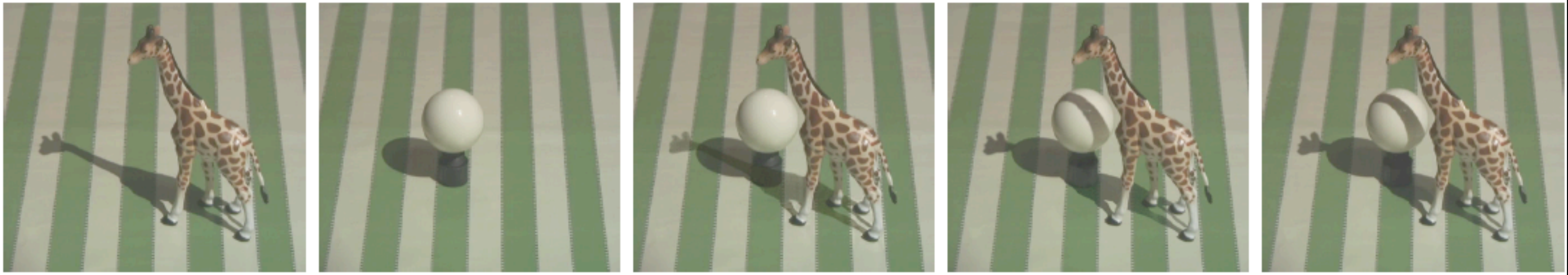
Object mask



Final composite

Figures from Debevec, Rendering Synthetic Objects into Real Scenes:
Bridging Traditional and Image-based Graphics with Global Illumination
and High Dynamic Range Photography1998

More interesting compositing problems



(a) Foreground scene

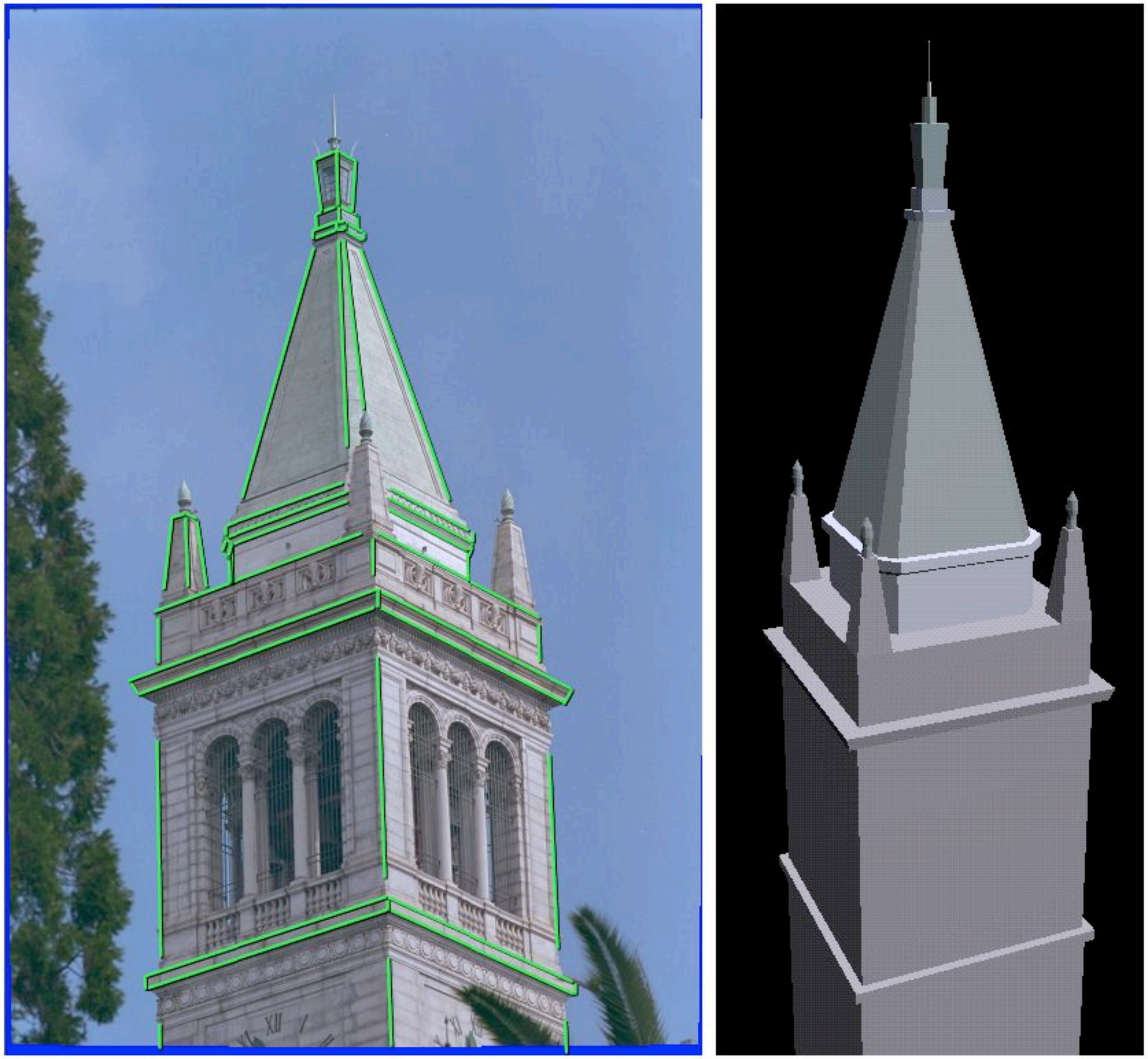
(b) Background scene

(c) Blue screen composite

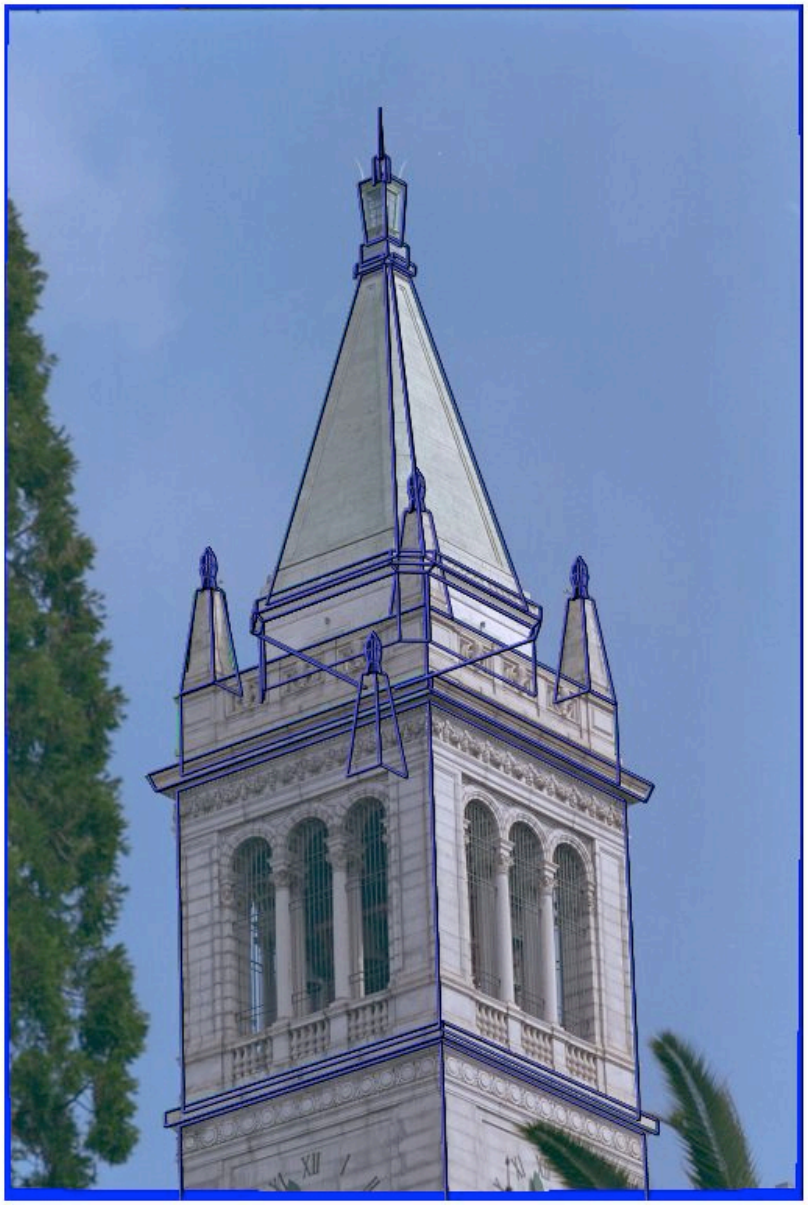
(d) Our method

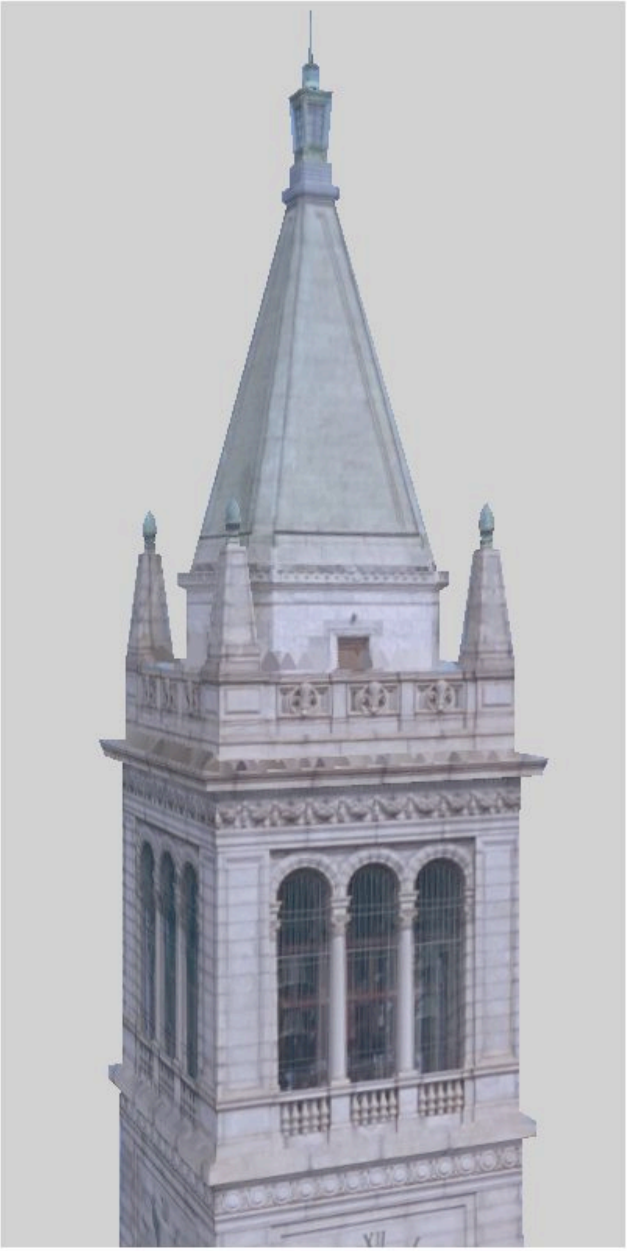
(e) Reference photograph

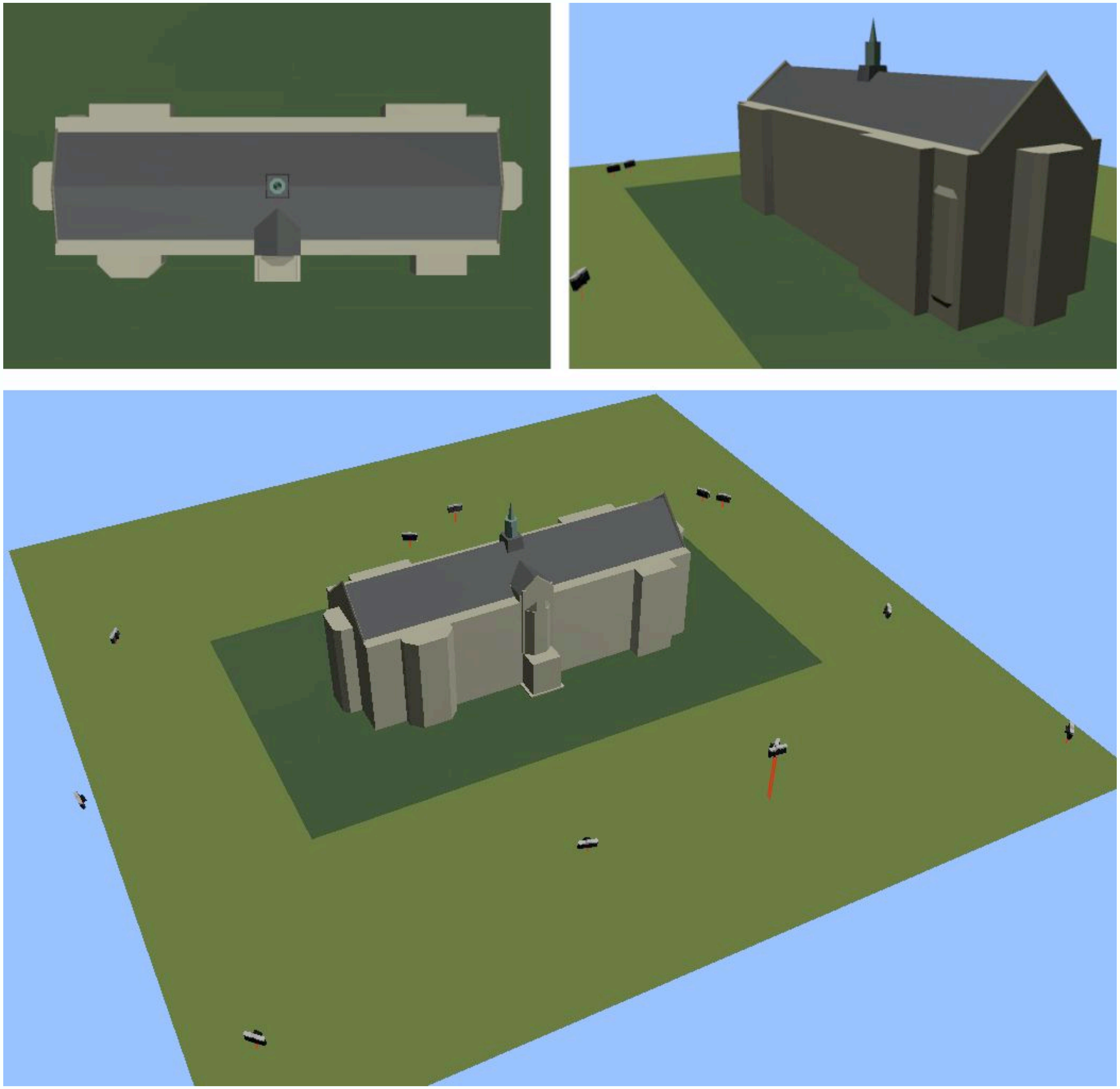
Figure from Shadow matting and compositing, Chuang et al 2002



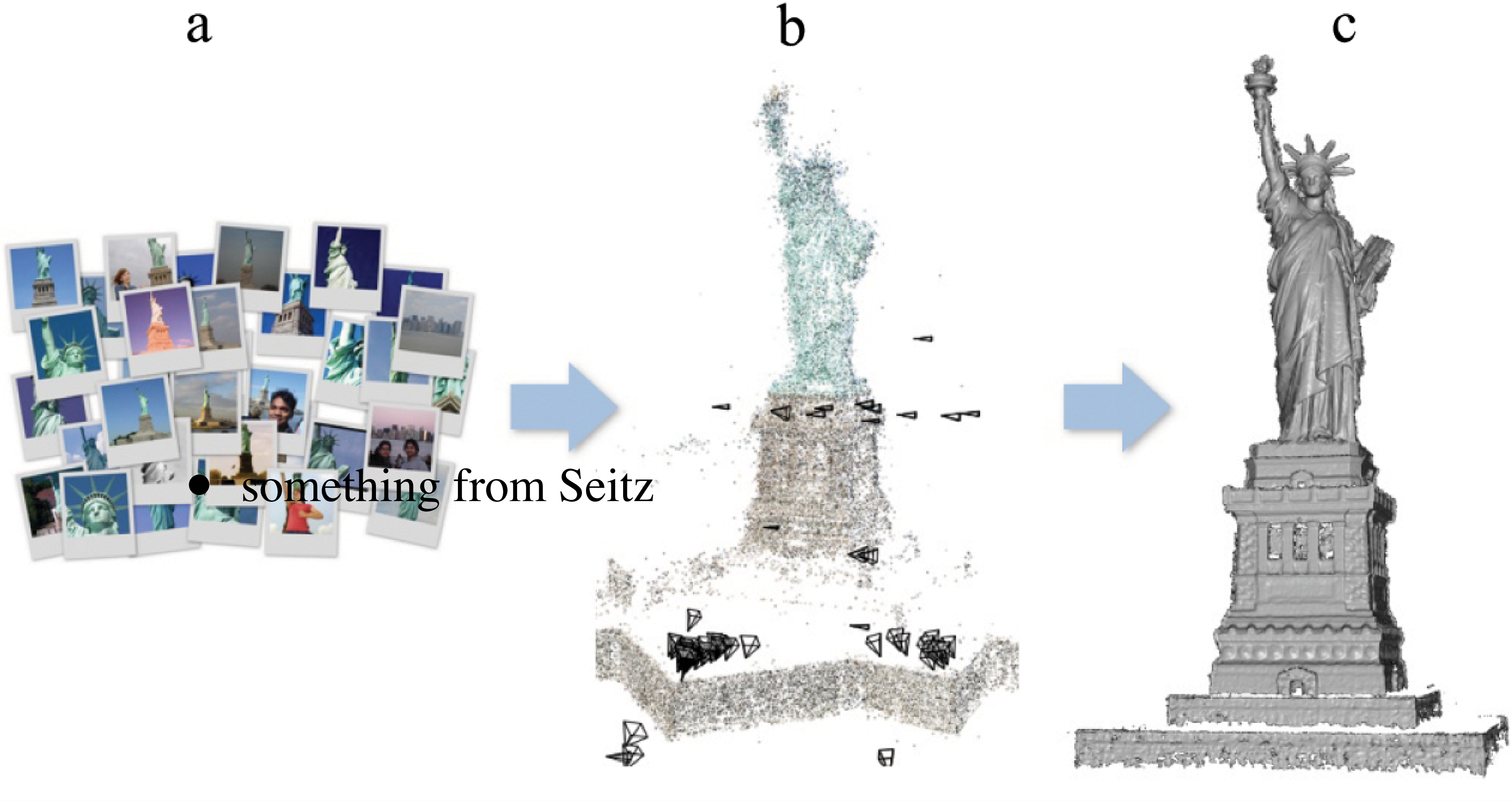
Work by Paul Debevec and Jitendra Malik







Capturing and animating occluded cloth - R White, K Crane, DA Forsyth SIGGRAPH 2007



- something from Seitz

M Goesele, N Snavely, B Curless, H Hoppe, “Multi-view stereo for community photo collections”, ICCV 2007

Photo Tourism

Exploring photo collections in 3D

Noah Snavely Steven M. Seitz Richard Szeliski

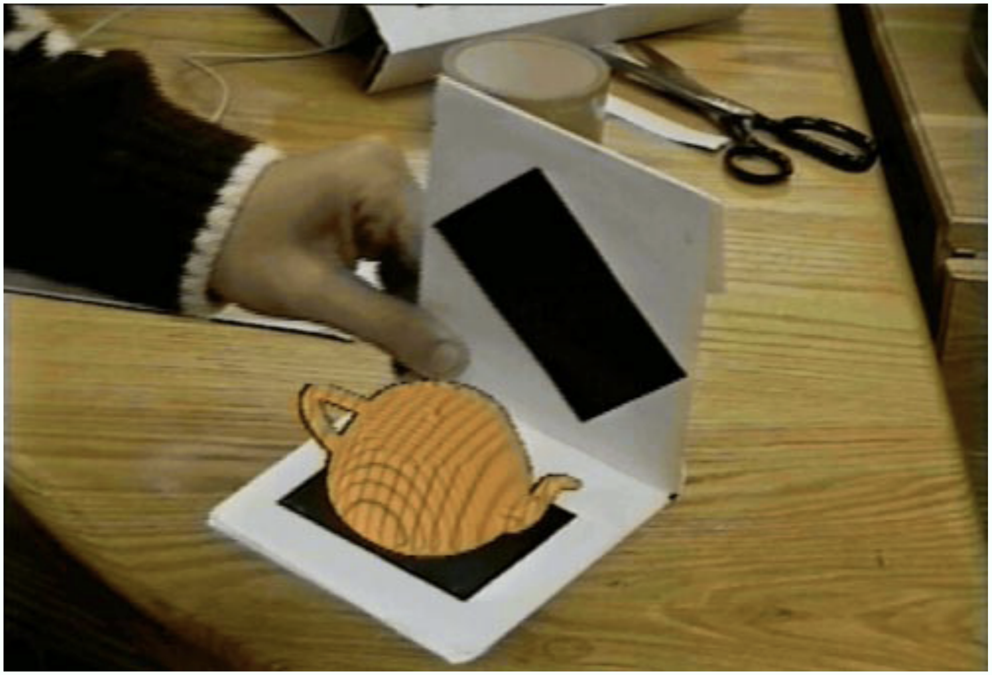
University of Washington *Microsoft Research*

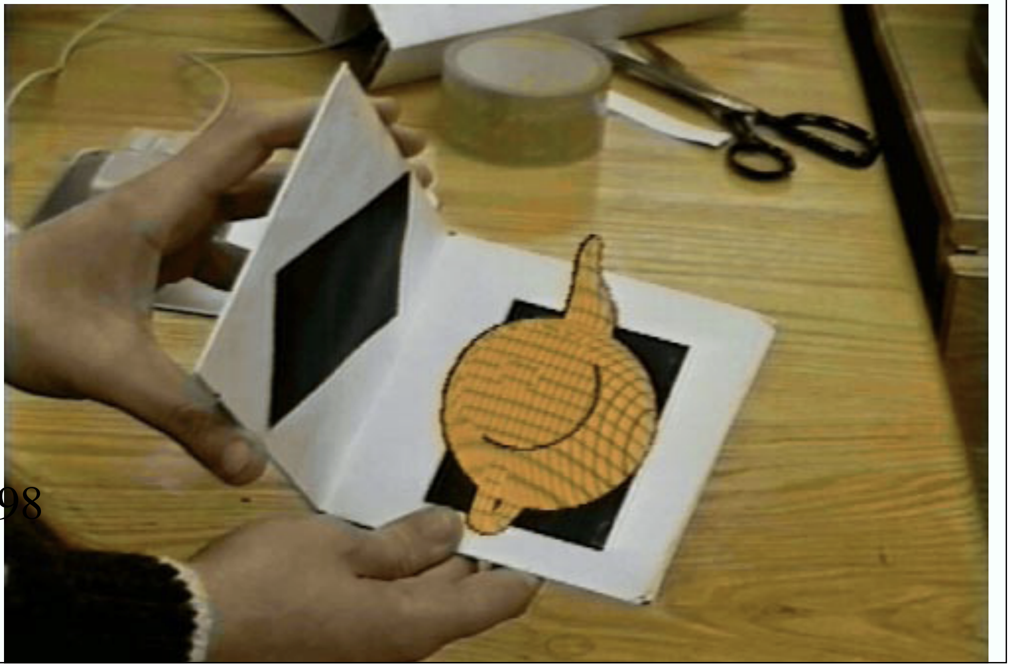
SIGGRAPH 2006

Noah Snavely, Steven M. Seitz, Richard Szeliski, "Photo tourism: Exploring photo collections in 3D," ACM Transactions on Graphics (SIGGRAPH Proceedings), 25(3), 2006, 835-846.

Camera calibration

- Two strategies:
  - Perspective cameras
    - calibration object has known points in 3D
    - find projections
    - compute camera using least squares
  - Scaled Orthography
    - projection is linear (no division, no H.C.’s)
    - world points as unique linear combination of calibration points
    - image projection is same linear combination of projected calibration points

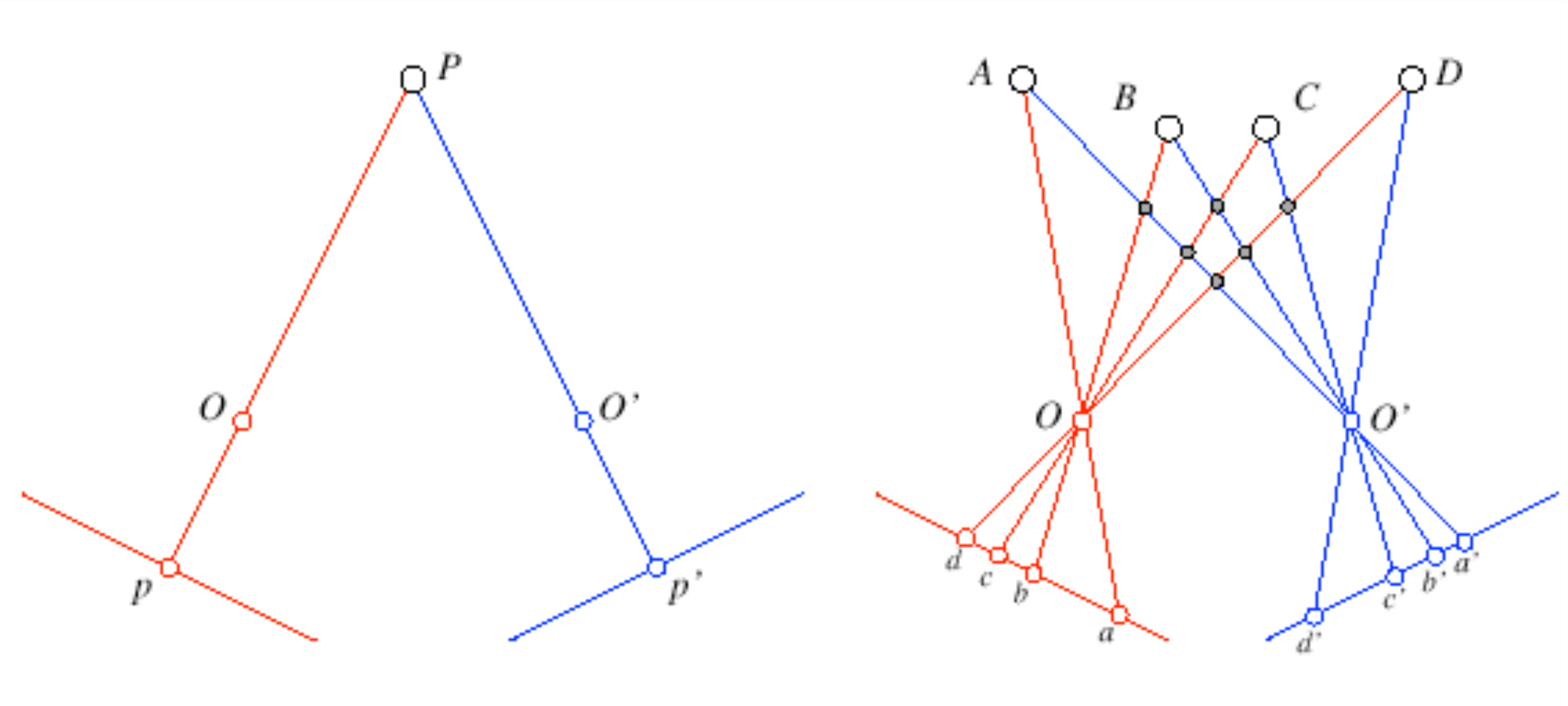




Calibration-Free Augmented Reality
Kiriakos N. Kutulakos and James R. Vallino, 1998

Two views

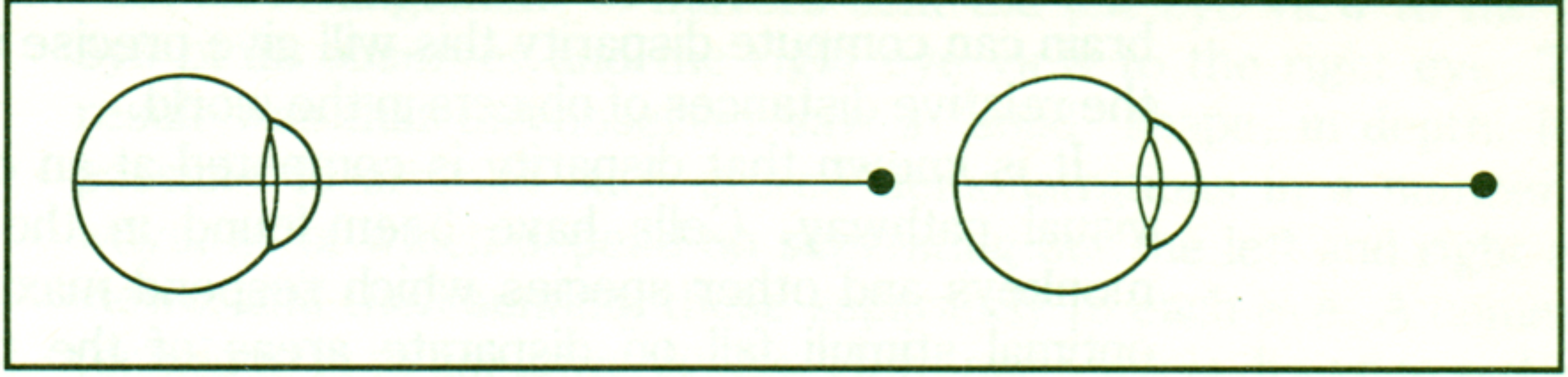
- Depth cues include
  - vergence
  - accomodation
  - stereopsis
  - motion
- Issues
  - what geometric information is available?
  - what matches are available? are correct?



Correspondence errors = depth errors

Accomodation and focus

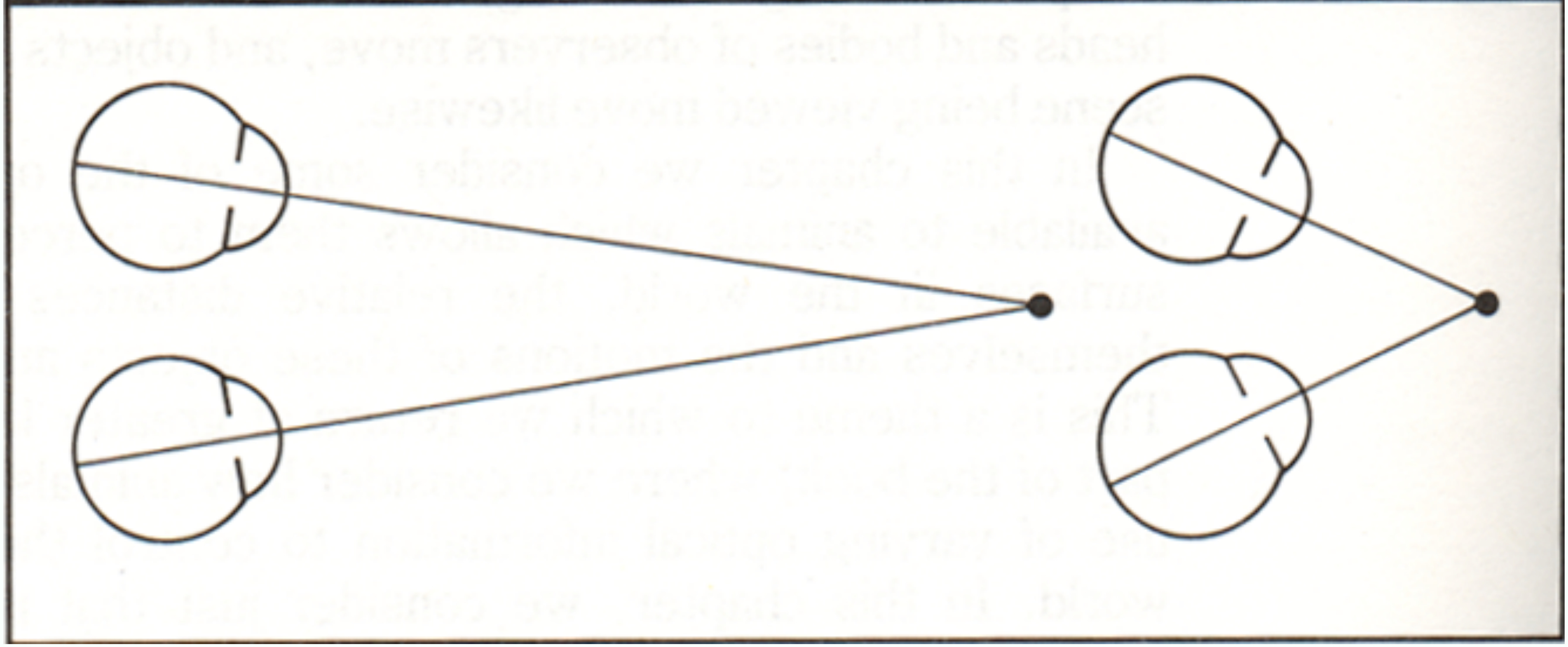
FIGURE 7.2



From Bruce and Green, Visual Perception,
Physiology, Psychology and Ecology

Convergence

FIGURE 7.1



From Bruce and Green, Visual Perception,
Physiology, Psychology and Ecology

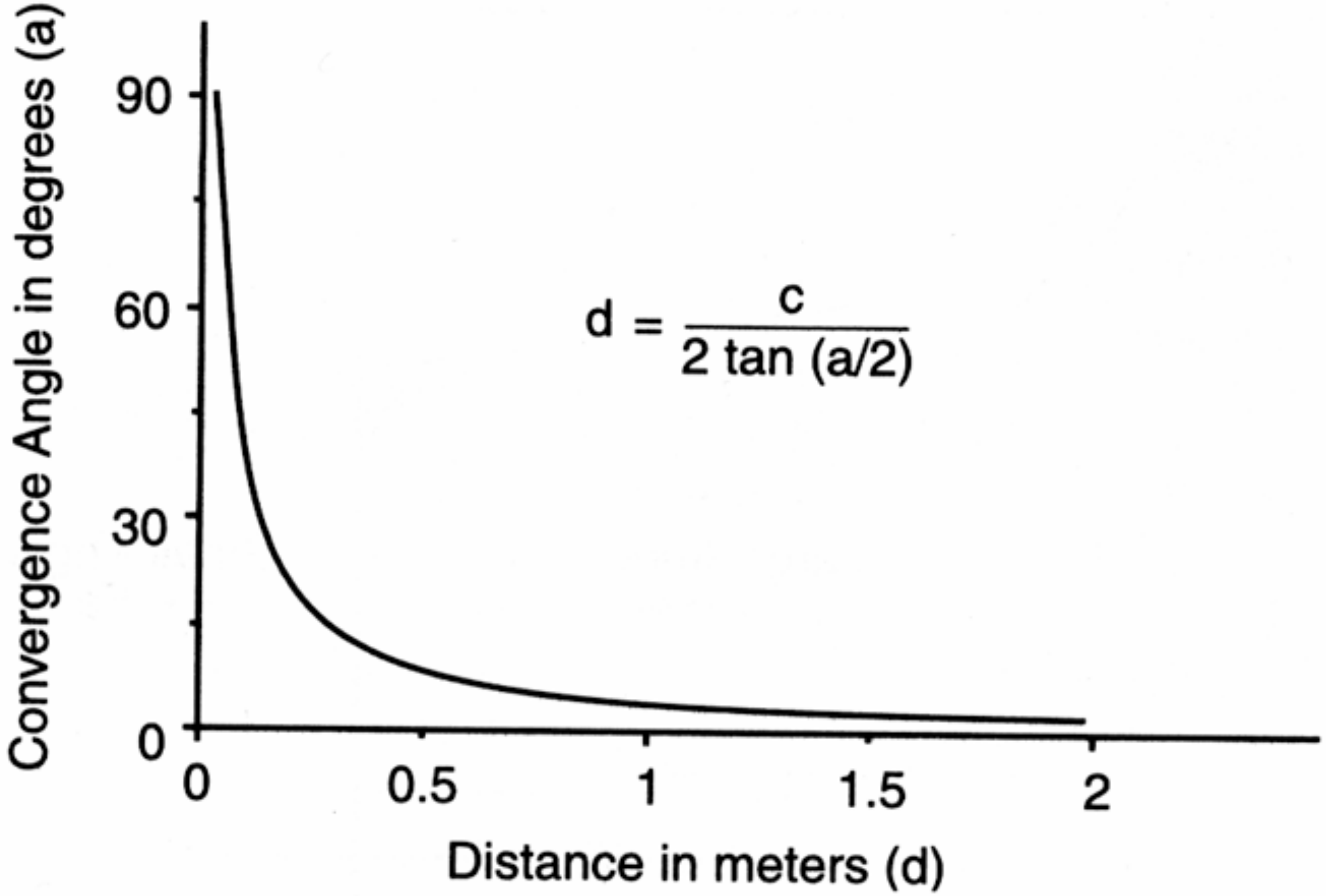
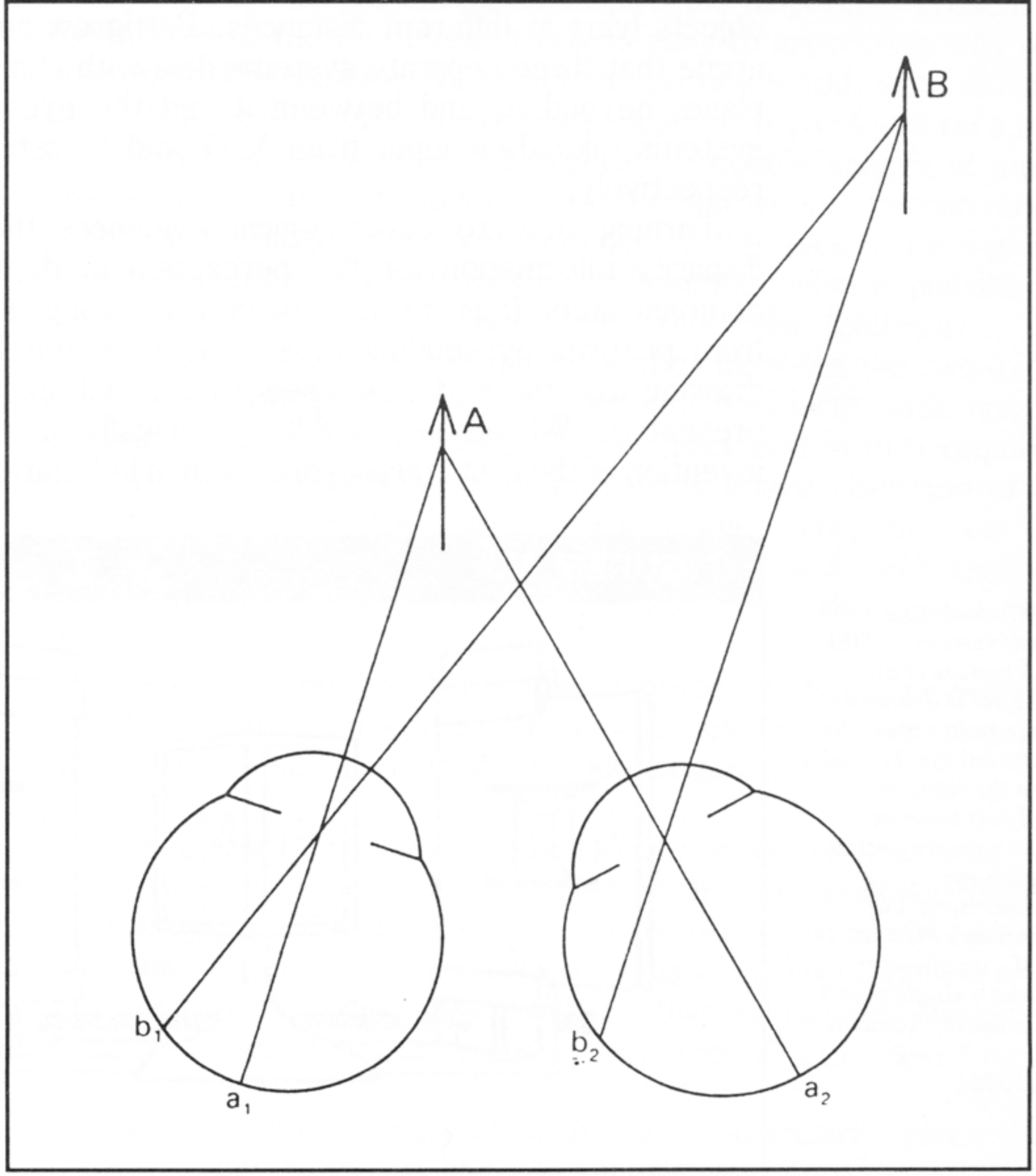


Figure 5.2.3 Convergence as a function of distance. The angle of convergence changes rapidly with distances up to a meter or two but very little after that.

From Palmer, “Vision Science”, MIT Press

FIGURE 7.3



From Bruce and Green, Visual Perception,
Physiology, Psychology and Ecology

Disparity occurs when
Eyes verge on one object;
Others appear at different
Visual angles

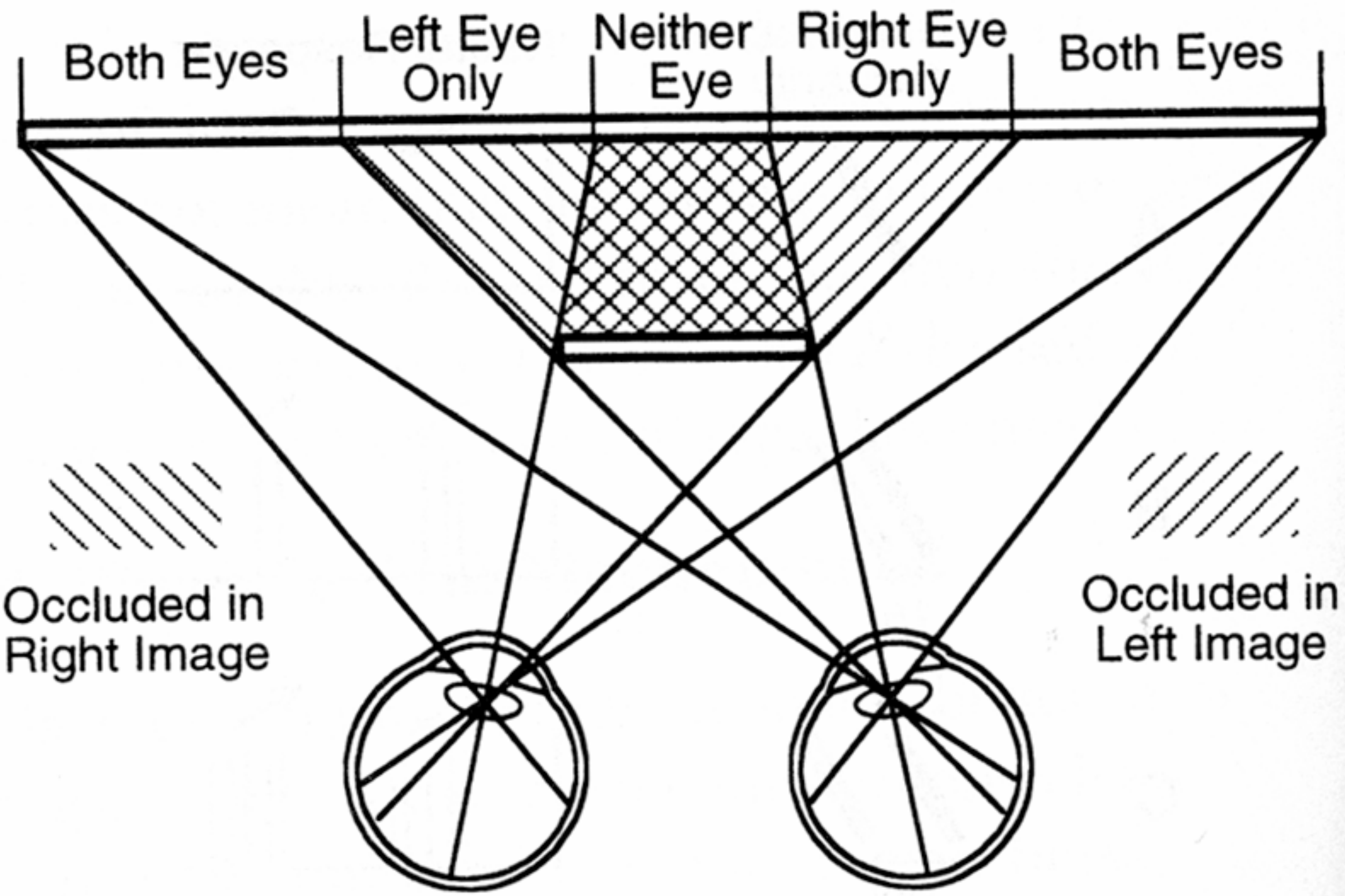
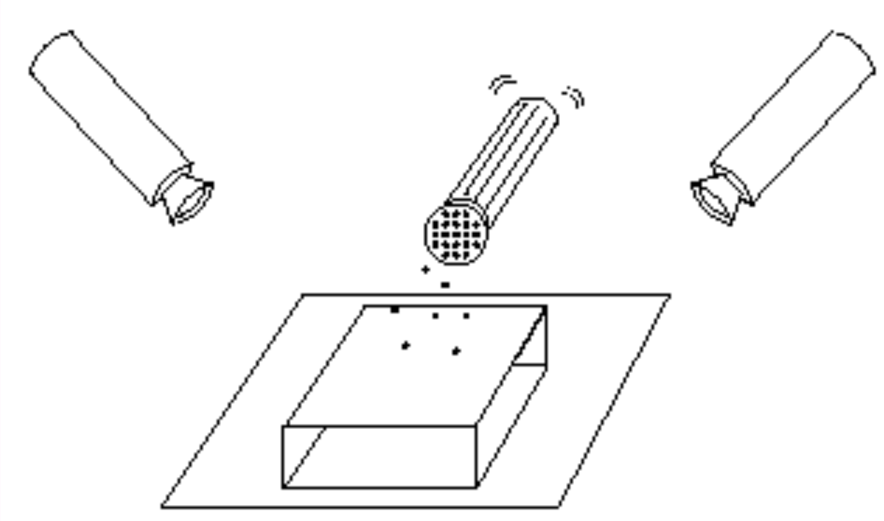


Figure 5.3.23 Da Vinci stereopsis. Depth information also arises from the fact that certain parts of one retinal image have no corresponding parts in the other image. (See text for details.)

From Palmer, "Vision Science", MIT Press

Random Dot Stereograms



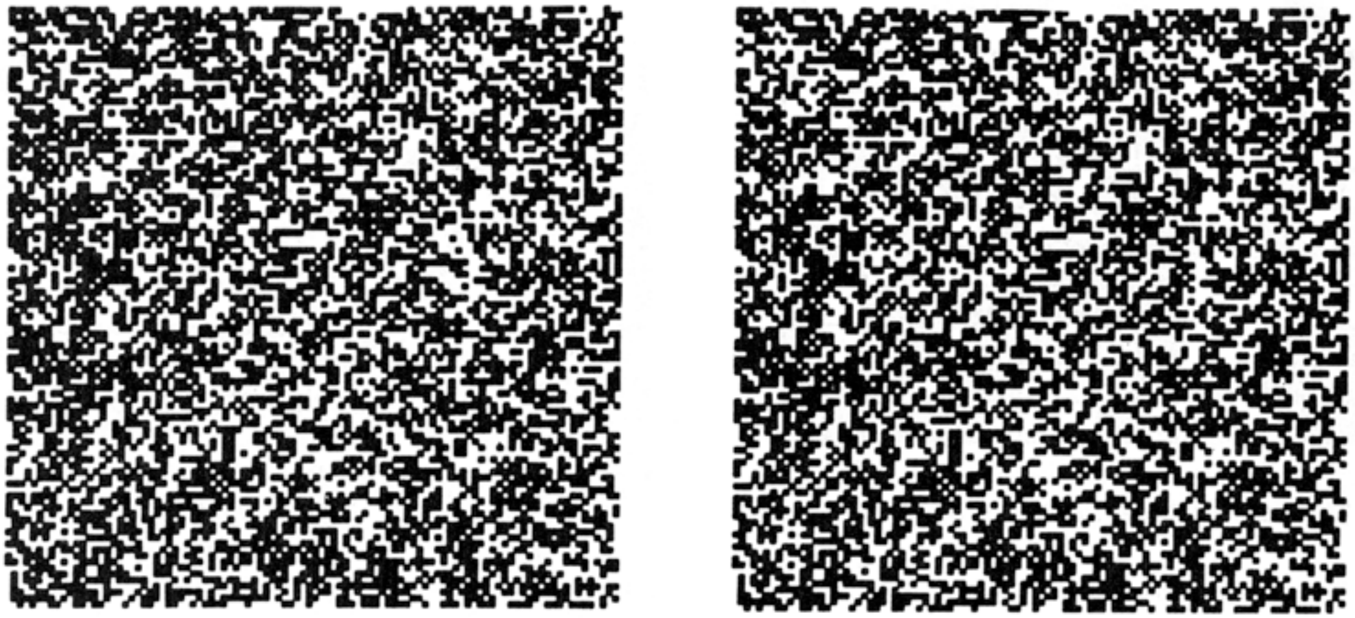


Figure 5.3.8 A random dot stereogram. These two images are derived from a single array of randomly placed squares by laterally displacing a region of them as described in the text. When they are viewed with crossed disparity (by crossing the eyes) so that the right eye's view of the left image is combined with the left eye's view of the right image, a square will be perceived to float above the page. (See pages 210–211 for instructions on fusing stereograms.)

From Palmer, “Vision Science”, MIT Press

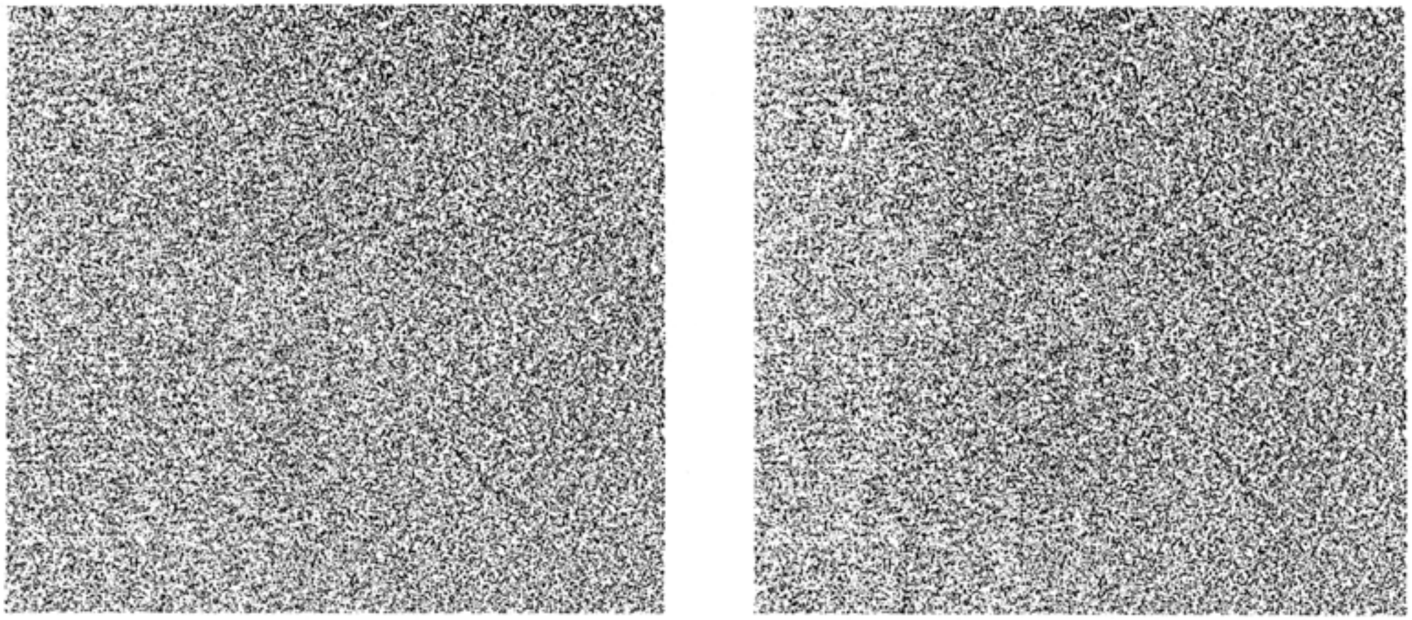


Figure 5.3.9 A random dot stereogram of a spiral surface. If these two images are fused with crossed convergence (see text on pages 210–211 for instructions), they can be perceived as a spiral ramp coming out of the page toward your face. This perception arises from the small lateral displacements of thousands of tiny dots. (From Julesz, 1971.)

From Palmer, “Vision Science”, MIT Press

Homogenous coordinates refresher

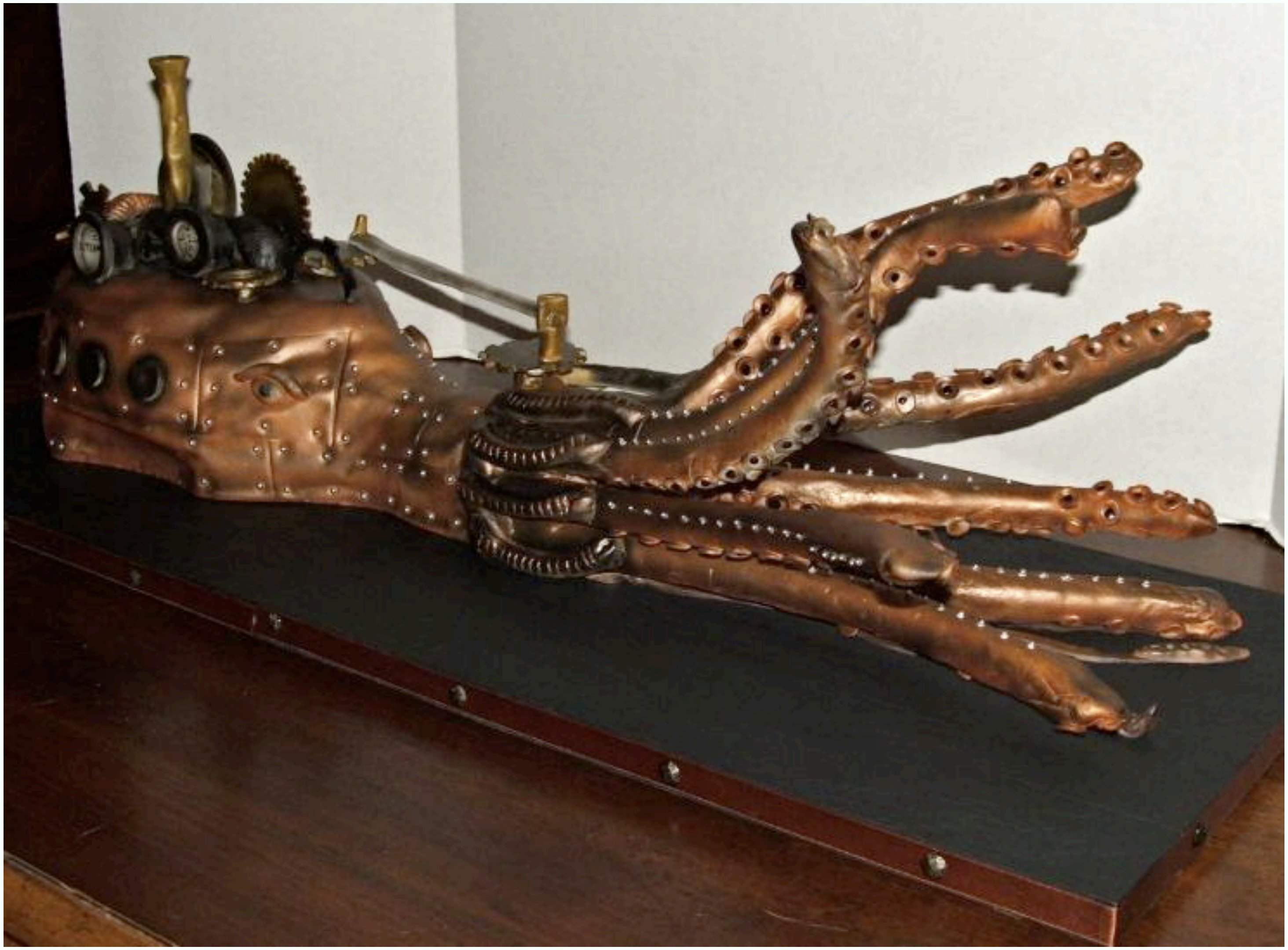
- Remember:
  - 3 coordinates in plane
  - 4 in 3D
  - equivalence relation --- two points are the same if one is parallel to other
- Lines on the plane
  - can be described using homogenous coords
- Planes in 3D
  - can be described using homogenous coords

Useful geometric construction

- Equation of line through p_1, p_2
    - det(p_1, p_2, x)=0
- Equation of plane through P_1, P_2, P_3
    - det(P_1, P_2, P_3, x)=0

The fundamental matrix

- A point in view one can lie on a line in view two
  - not anywhere IMPORTANT
  - only on epipolar line
- Each point corresponds to a line
  - the coefficients of the line depend linearly on the point’s coefficients
- The family of lines passes through a point
  - the epipole



What do we know about matches?

- Geometry:
  - We work with points and lines in HC’s
  - A point in left image corresponds to a line in right image
    - the coefficients of the line depend linearly on the point’s coefficients
  - A 2D family of points in left gives a 1D family of lines in right
    - also, right->left
- All this means
  - there is a Fundamental matrix
    - which has determinant zero

$$\mathbf{x}_{\mathrm{left}}^{T}\mathcal{F}\mathbf{x}_{\mathrm{right}} = 0$$

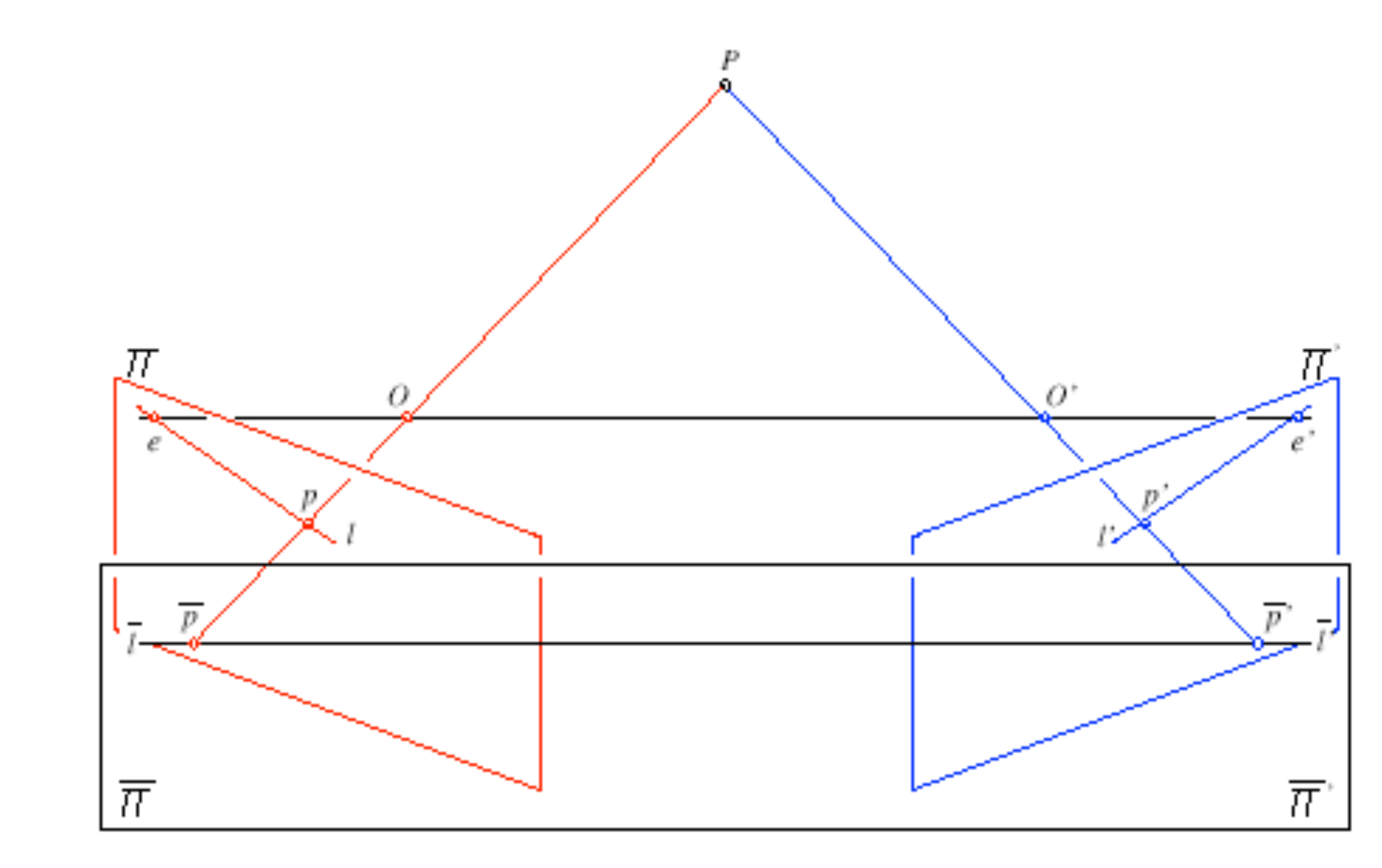
Estimating the fundamental matrix

- We need to estimate 7 degrees of freedom
- Algorithm 1:
  - Take 8 point correspondences
  - Estimate linearly
- Algorithm 2 (better):
  - Take 7 point correspondences
  - Estimate linear family
  - Solve cubic
    - Check roots with 8’th point if three real

Pragmatics

- Simplify activities by assuming
  - That camera image planes are coplanar
  - That focal lengths are the same
  - That the separation is parallel to the scanlines
  - (all this used to be called the epipolar configuration)

Rectification



Triangulation

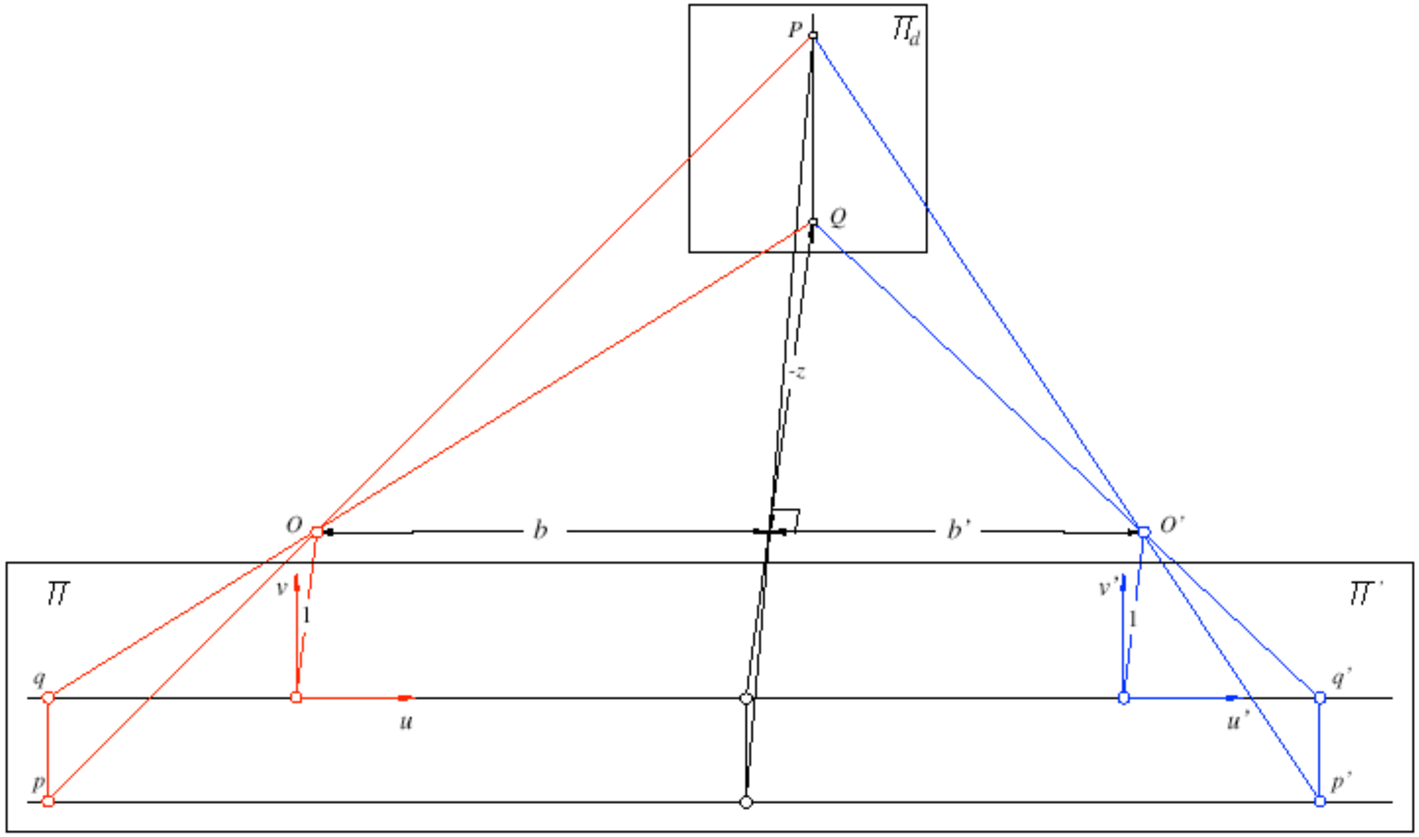
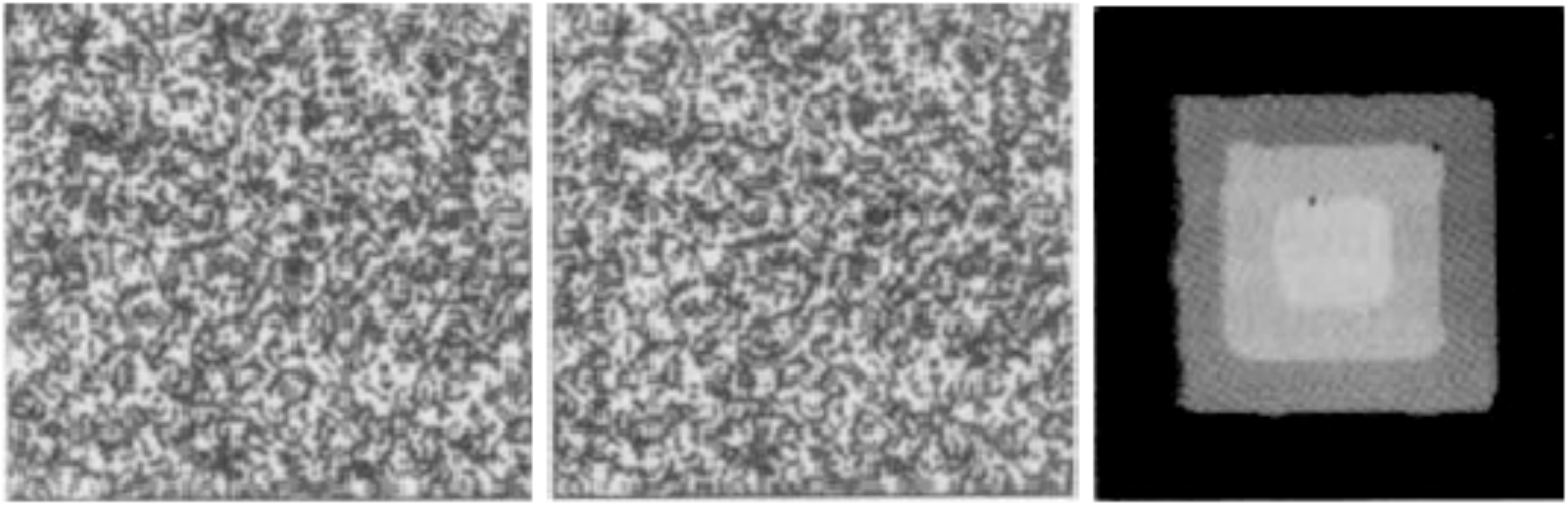
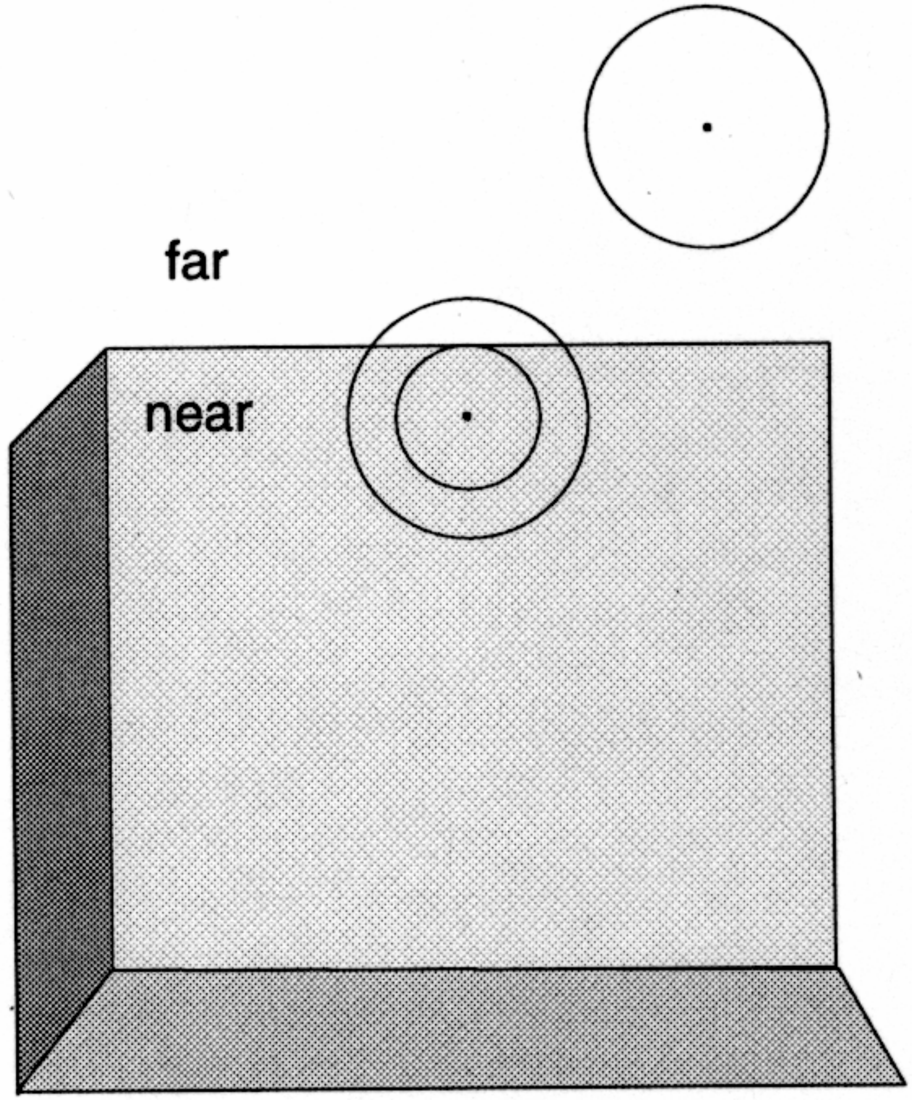


Figure 13.6. Triangulation for rectified images: the rays associated with two points p and p' on the same scanline are by construction guaranteed to intersect in some point P. As shown in the text, the depth of P relative to the coordinate system attached to the left camera is inversely proportional to the disparity $d = u' - u$. In particular, the preimage of all pairs of image points with constant disparity d is a *frontoparallel* plane Π_d (i.e., a plane parallel to the camera retinas).

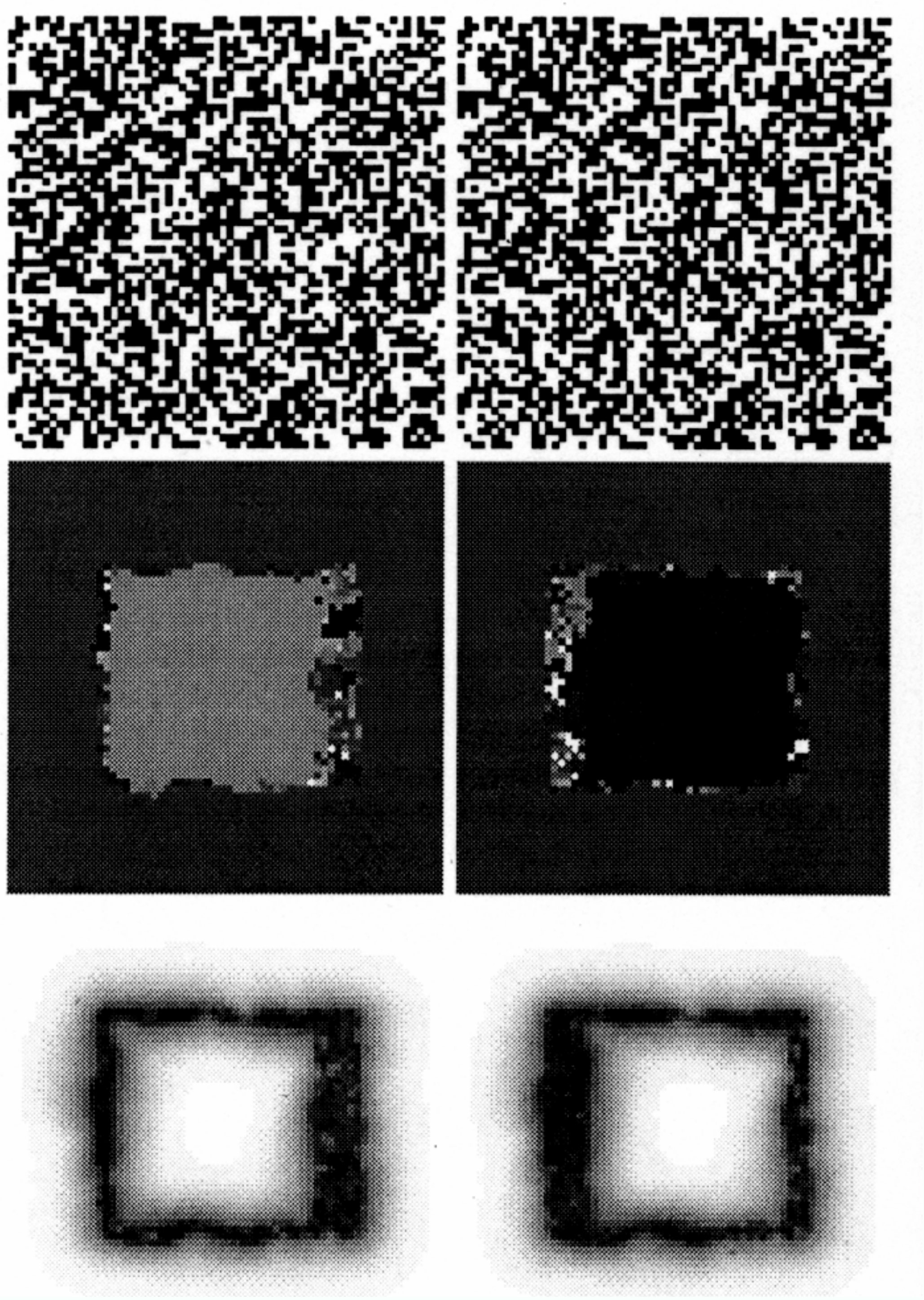
Pragmatics

- Issue
  - Match points
- Strategy
  - correspondences occur only along scanlines
  - represent points from coarse to fine
    - scale problems - some scales are misleading
- Issue
  - some points don’t have correspondences (occlusion)
- Match left to right, then right to left
  - if they don’t agree, break match

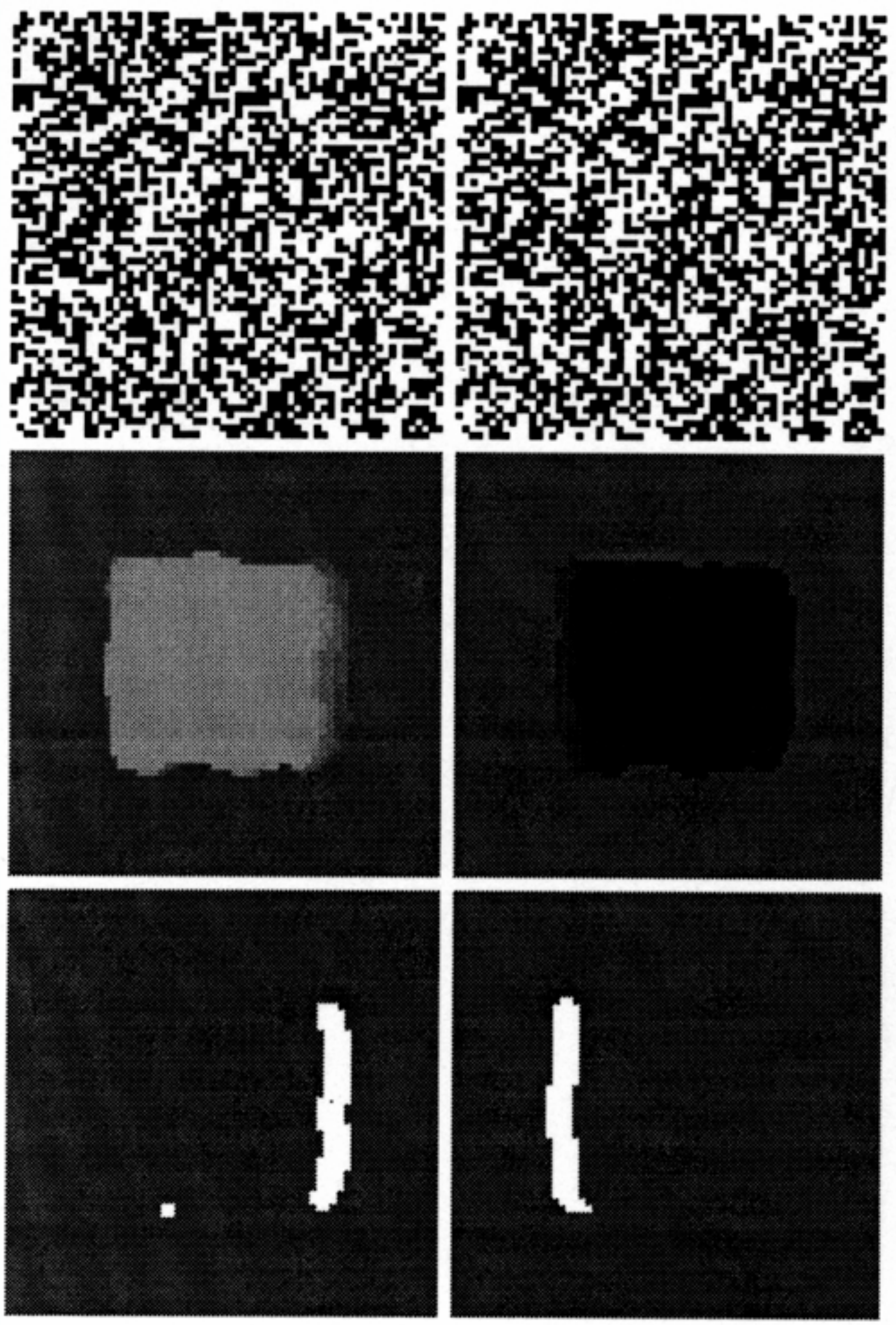




From Jones and Malik, “A computational framework for determining Stereo correspondences from a set of linear spatial filters



From Jones and Malik, “A computational framework for determining
Stereo correspondences from a set of linear spatial filters



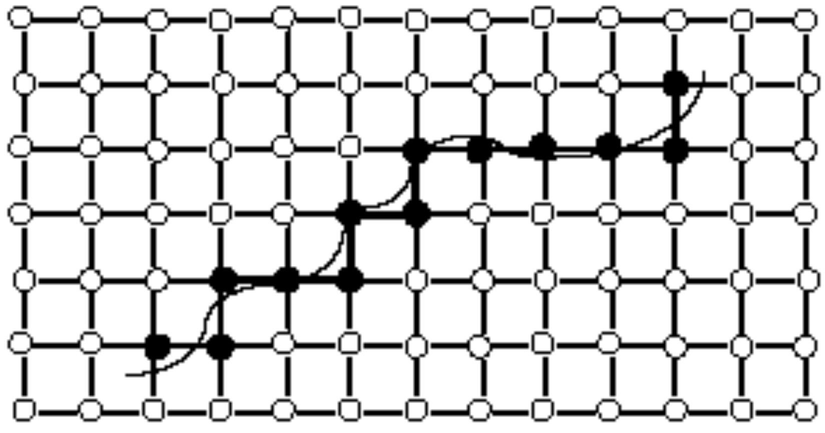
From Jones and Malik, “A computational framework for determining
Stereo correspondences from a set of linear spatial filters

Stereopsis - SOA

- We can estimate fundamental matrices accurately
- Highly accurate two view stereo is available
    - reconstructions are compared to ground truth
    - code available
    - check http://vision.middlebury.edu/stereo/
        -

Moving your head with IBR

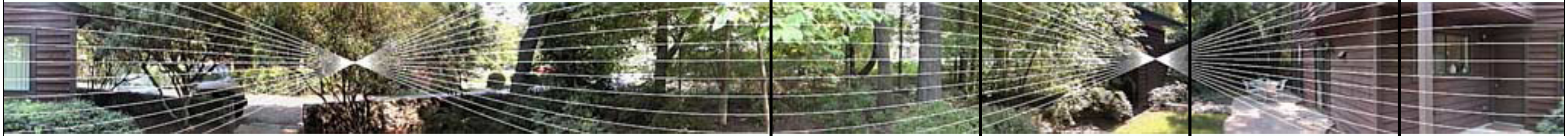
- Recall
  - we have cylindrical panoramas, so it’s easy to rotate
- but what if we translate?
  - assume we have many samples
  - camera is
    - translated back from forward sample
    - translated forward from back sample
  - compute epipolar structure for each pair
  - match to get depths
  - now predict pixels
    - one from forward, one from back
    - decide which to show, or interpolate



Moving your head with IBR-2

Cylinder 1

Cylinder 2

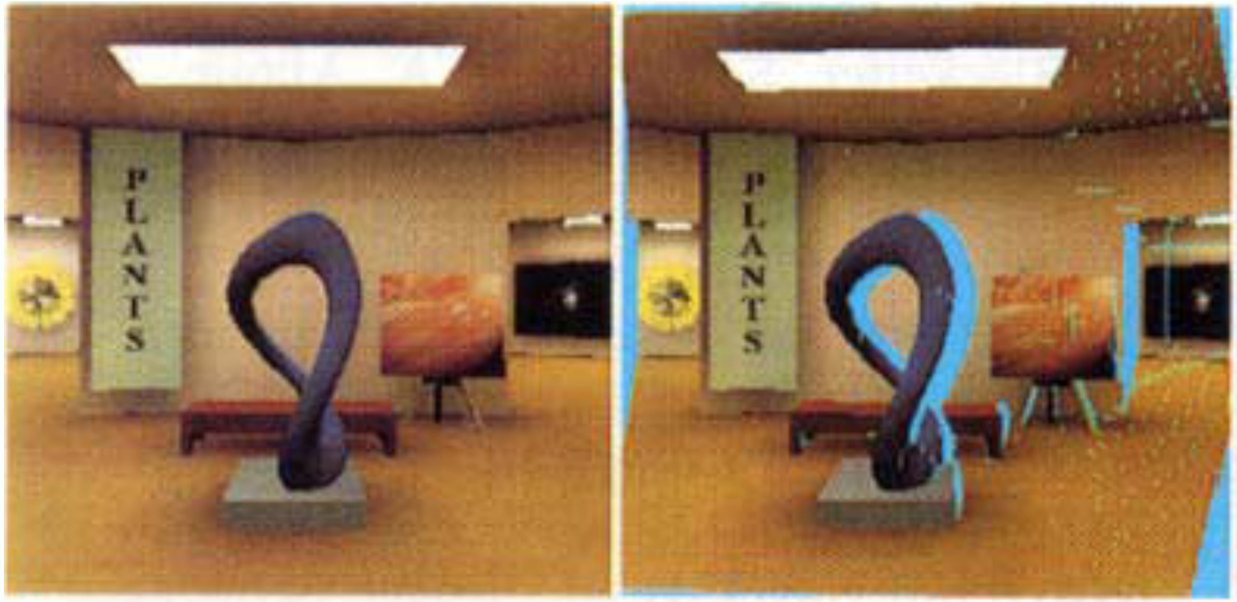


Epipoles

McMillan and Bishop, 95

Moving your head with IBR - 3

- There might be holes in the interpolate forward (back)
- Fill these in using
  - other interpolate
  - texture synthesis

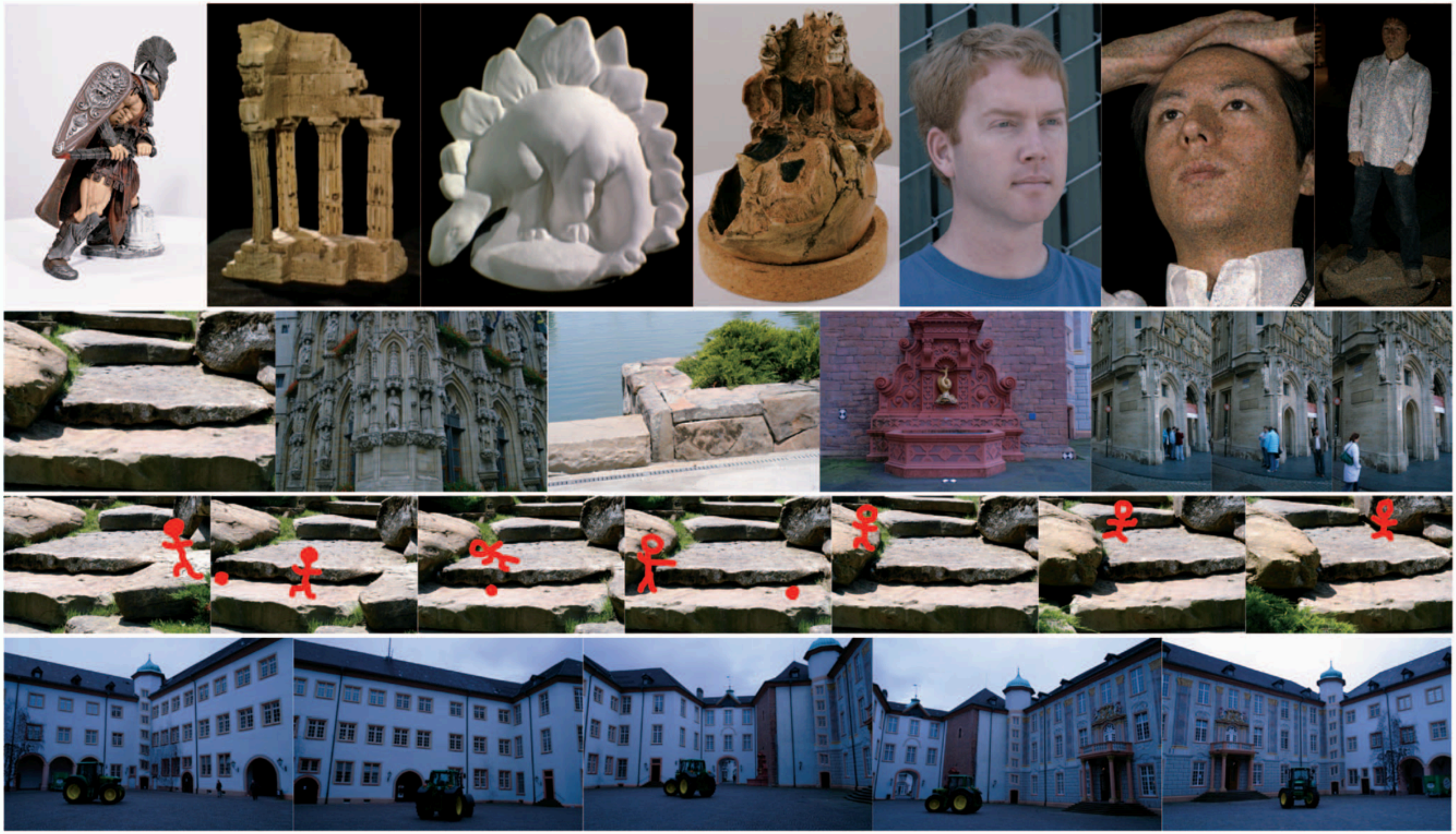


Chen+Williams 93

Multi view stereopsis

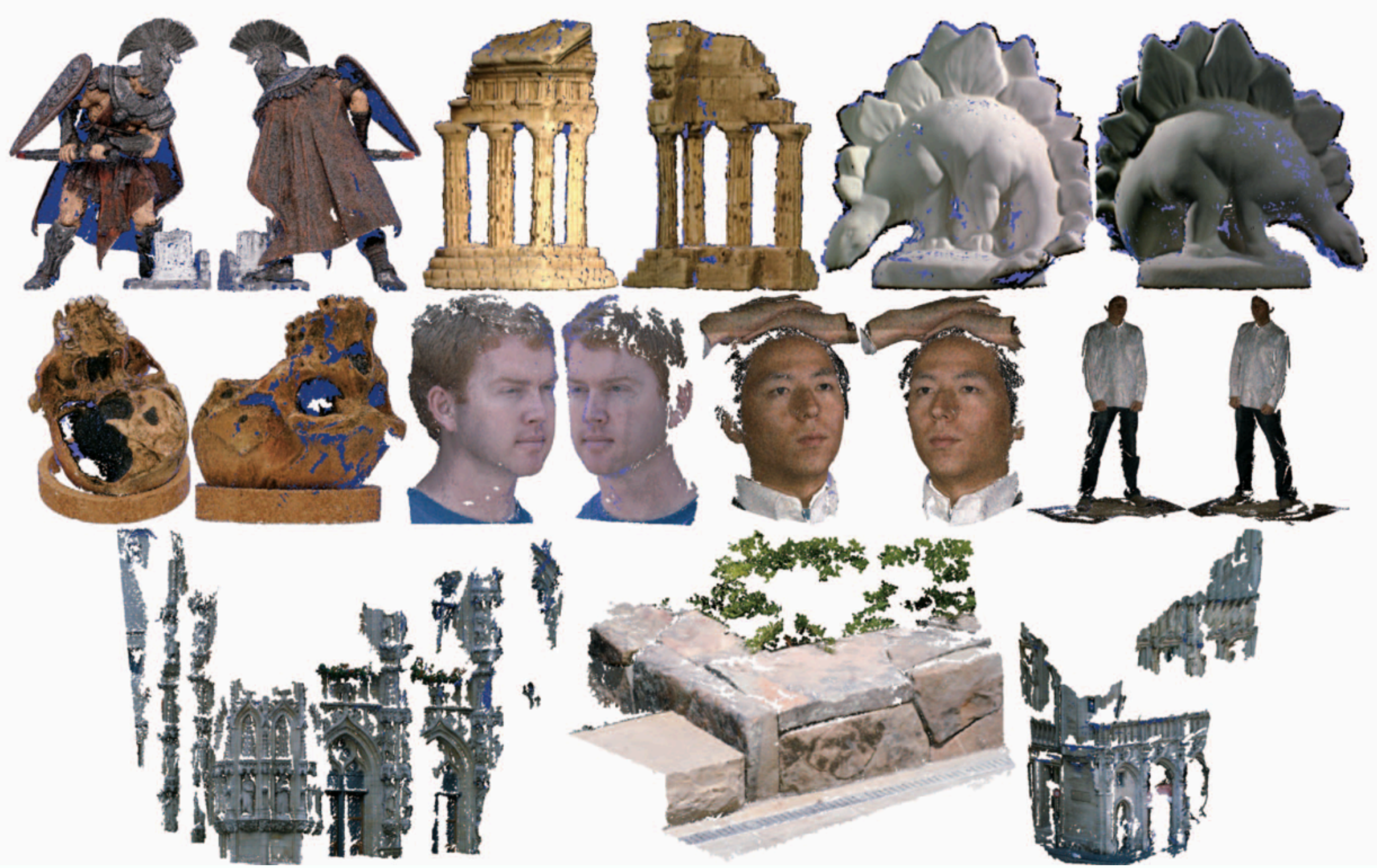
- More than two views make things better
  - but now we have to know where cameras are
    - can infer (next slides), or know
    - point constraints are richer

Multiview stereopsis - SOA



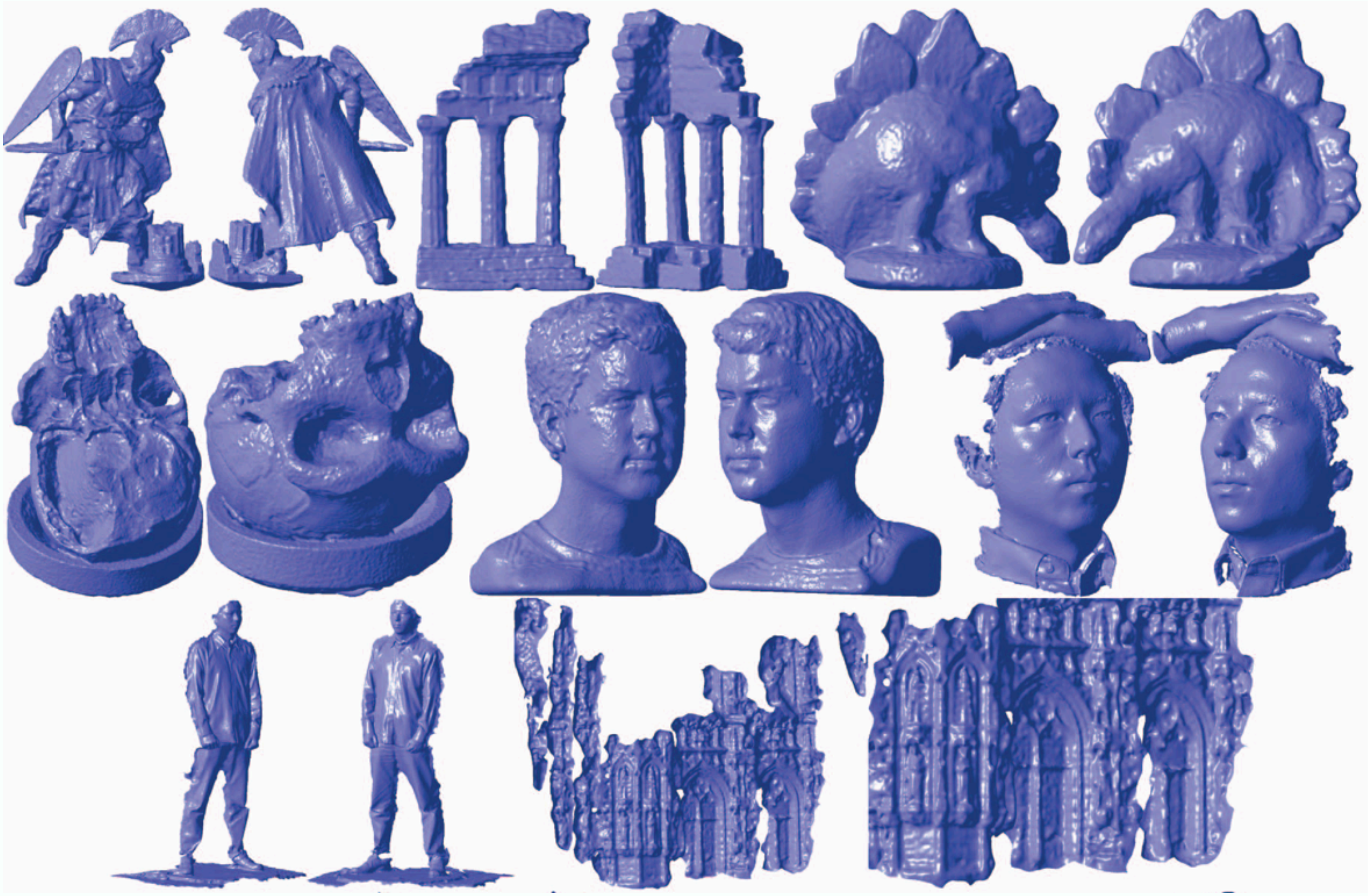
Datasets - from Furukawa + Ponce, 2010

Multiview stereopsis - SOA



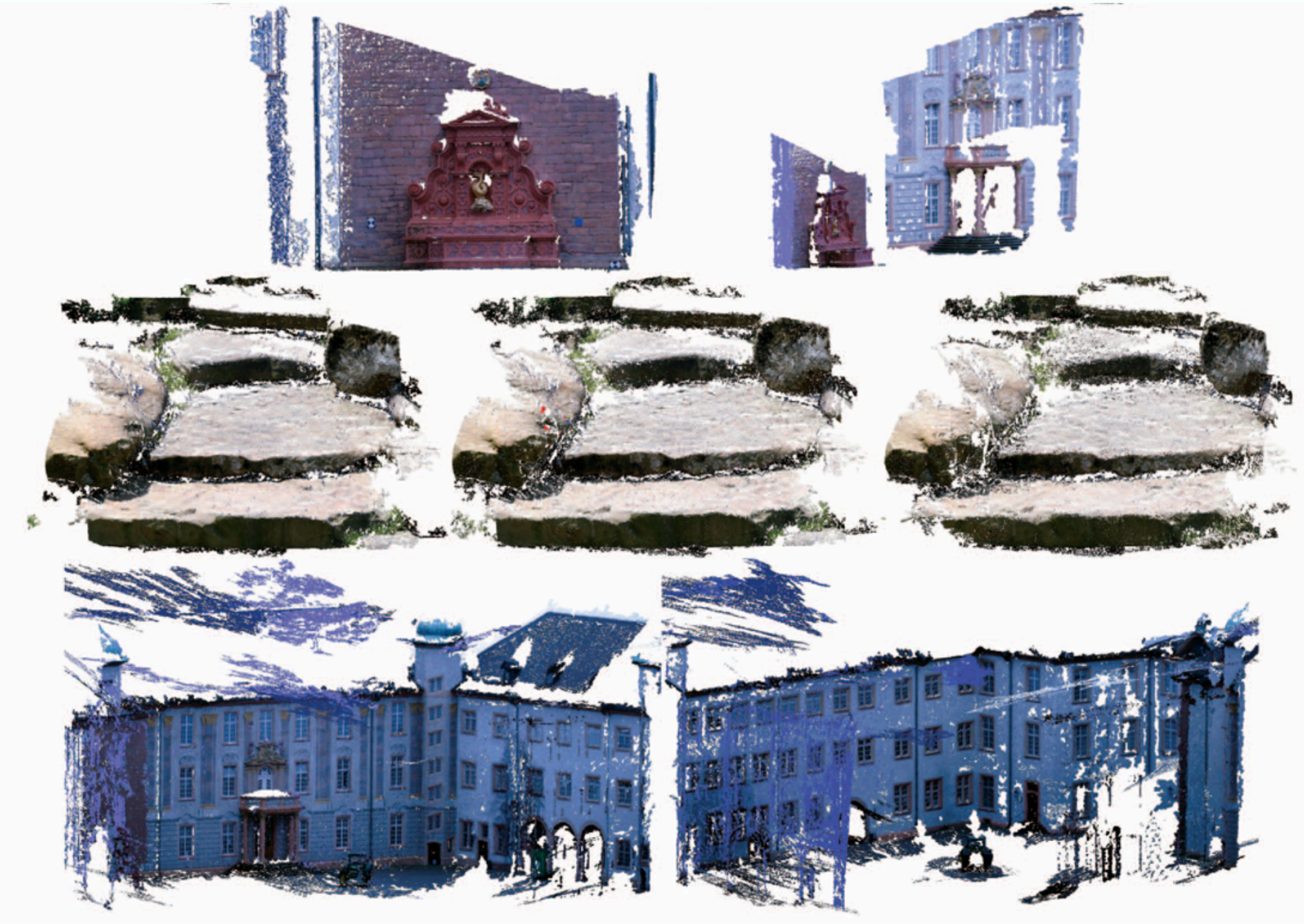
Patches - from Furukawa + Ponce, 2010

Multiview stereopsis - SOA



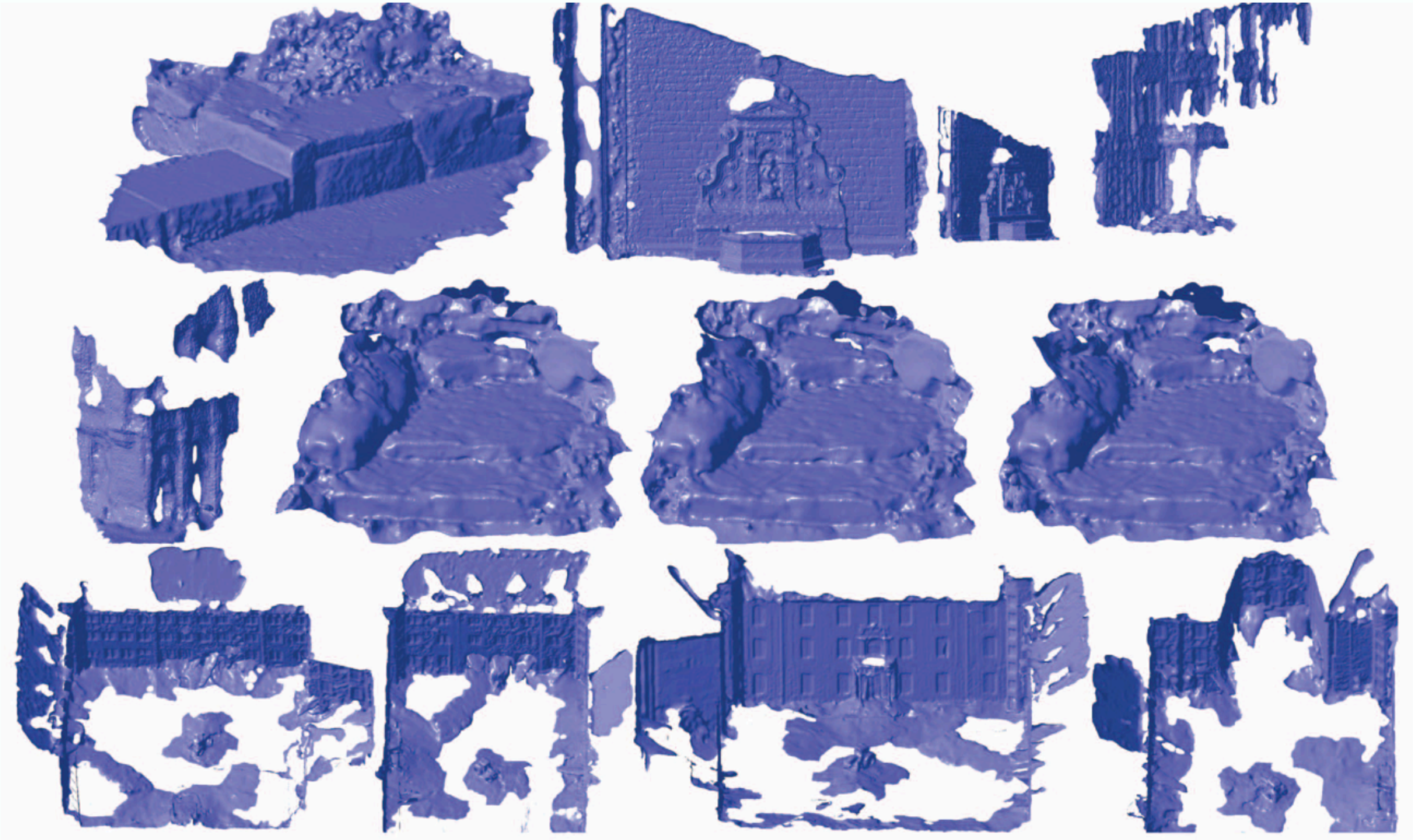
Mesh models - from Furukawa + Ponce, 2010

Multiview stereopsis - SOA



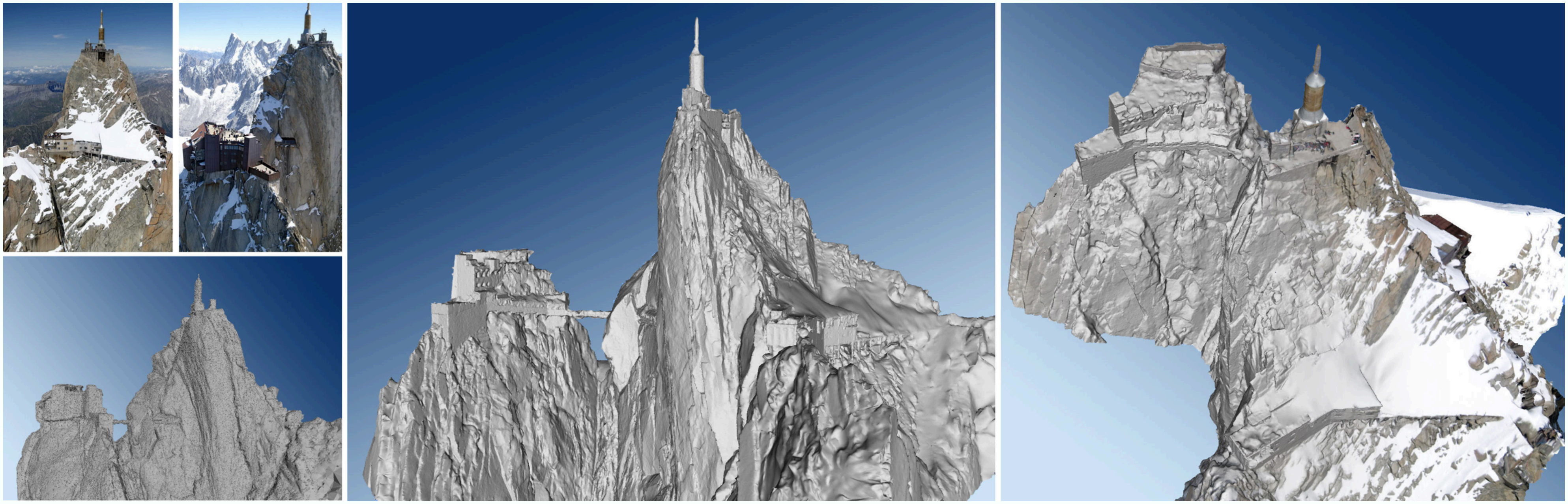
Patches - from Furukawa + Ponce, 2010

Multiview stereopsis - SOA



Patches - from Furukawa + Ponce, 2010

Multiview Stereo - SOA



Hiep et al 09, mountain reconstruction from helicopter views

Multiview Stereo - SOA

Hiep et al 09,