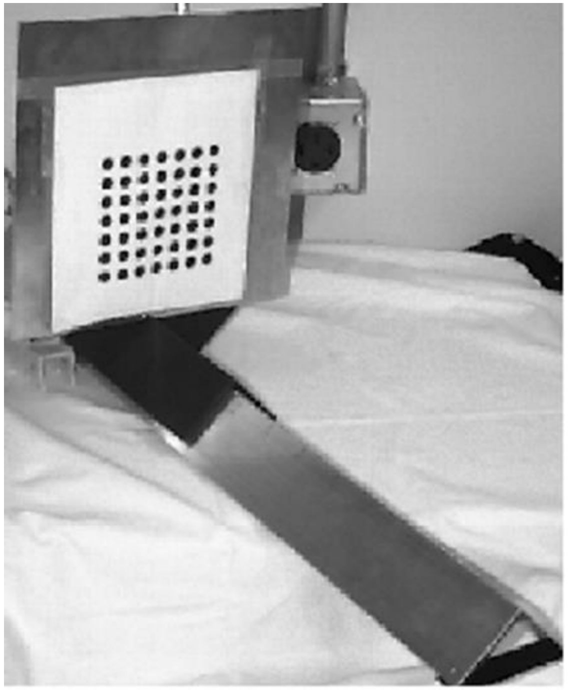
Computer Vision

ECE/CSE 576
Stereo and 3D

Linda Shapiro
Professor of Computer Science & Engineering
Professor of Electrical Engineering

Camera Calibration



The idea is to snap images at different depths and get a lot of 2D-3D point correspondences.

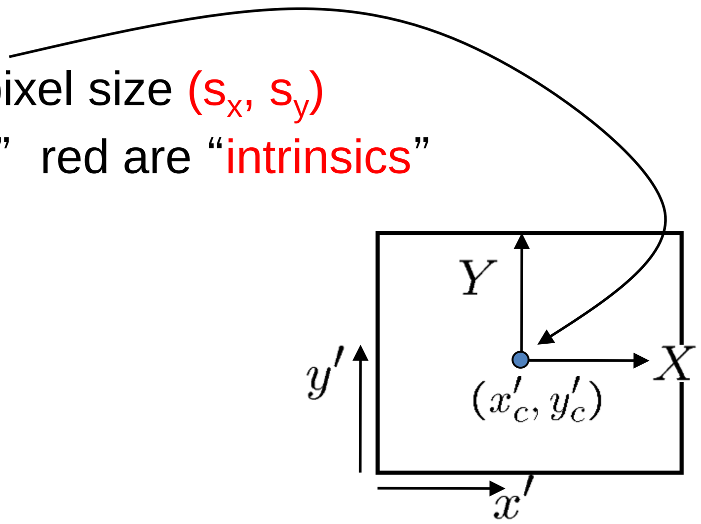
x1, y1, z1, u1, v1
x2, y2, z1, u2, v2
.
.
xn, yn, zn, un, vn

Then solve a system of equations to get camera parameters.

Camera Parameters

A camera is described by several parameters

- Translation T of the optical center from the origin of world coords
- Rotation R of the image plane
- focal length f, principal point (x'_c, y'_c), pixel size (s_x, s_y)
- blue parameters are called "extrinsics," red are "intrinsics"



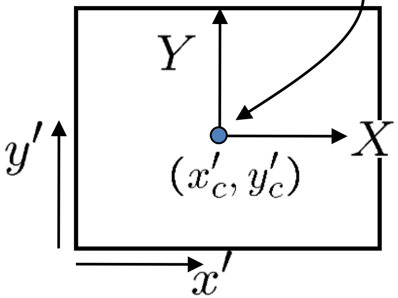
Camera Parameters

A camera is described by several parameters

- Translation T of the optical center from the origin of world coords
- Rotation R of the image plane
- focal length f, principal point (x'_c, y'_c), pixel size (s_x, s_y)
- blue parameters are called "extrinsics," red are "intrinsics"

Projection equation

$$\mathbf{x} = \begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{\Pi X}$$



- The projection matrix models the cumulative effect of all parameters
- Useful to decompose into a series of operations

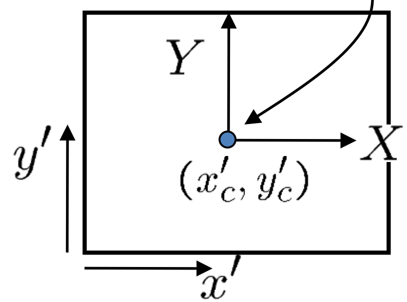
Camera Parameters

A camera is described by several parameters

- Translation T of the optical center from the origin of world coords
- Rotation R of the image plane
- focal length f, principal point (x'_c, y'_c), pixel size (s_x, s_y)
- blue parameters are called "extrinsics," red are "intrinsics"

Projection equation

$$\mathbf{x} = \begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{\Pi X}$$



- Useful to decompose into a series of operations

identity matrix

$$\mathbf{\Pi} = \underbrace{\begin{bmatrix} -fs_x & 0 & x'_c \\ 0 & -fs_y & y'_c \\ 0 & 0 & 1 \end{bmatrix}}_{\text{intrinsics}} \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\text{projection}} \underbrace{\begin{bmatrix} \mathbf{R}_{3x3} & \mathbf{0}_{3x1} \\ \mathbf{0}_{1x3} & 1 \end{bmatrix}}_{\text{rotation}} \underbrace{\begin{bmatrix} \mathbf{I}_{3x3} & \mathbf{T}_{3x1} \\ \mathbf{0}_{1x3} & 1 \end{bmatrix}}_{\text{translation}} \leftarrow [tx, ty, tz]^T$$

- The definitions of these parameters are **not** completely standardized
  - especially intrinsics—varies from one book to another

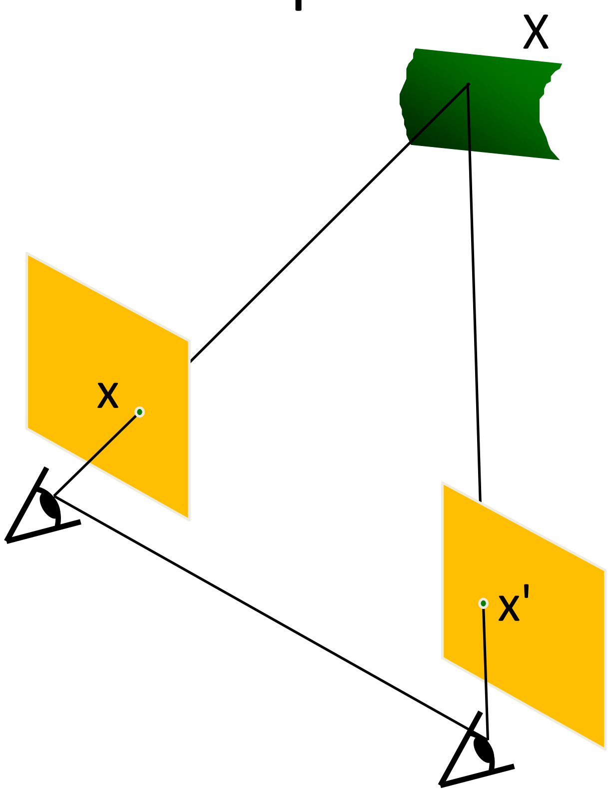
Stereo

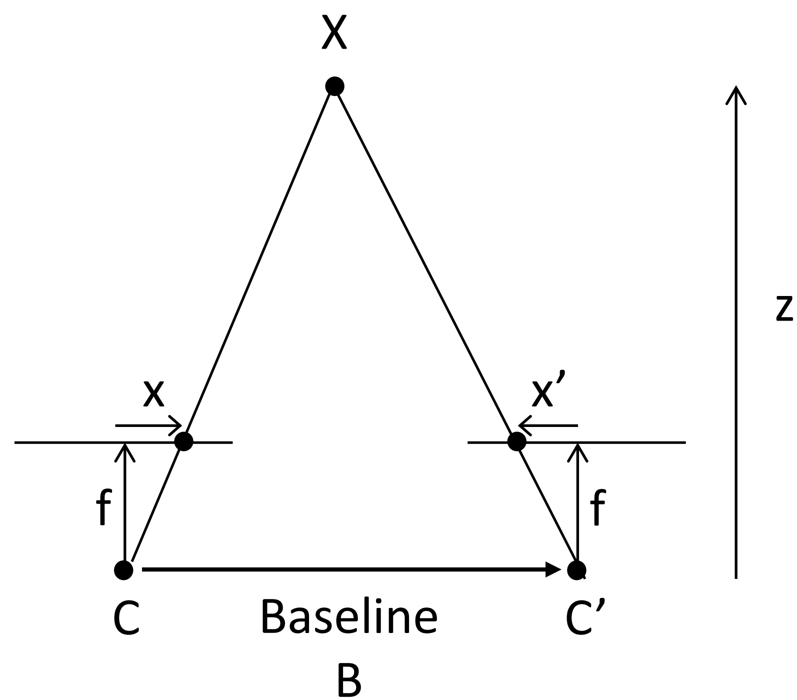
Amount of horizontal movement is …

…inversely proportional to the distance from the camera

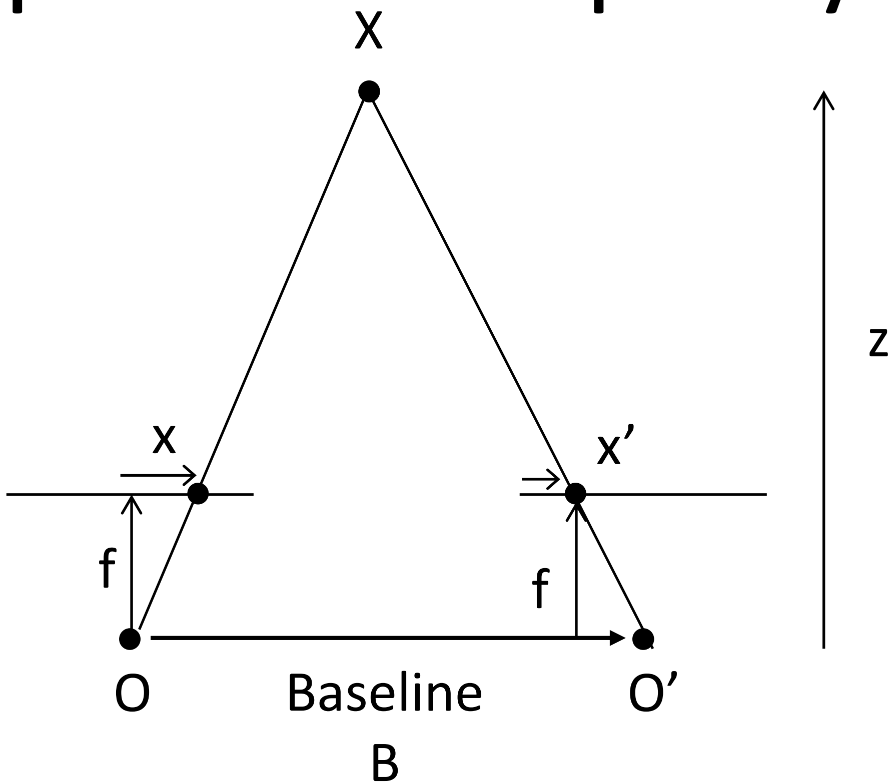
Depth from Stereo

- Goal: recover depth by finding image coordinate x' that corresponds to x





Depth from disparity



$$\boxed{\frac{x - x'}{O - O'} = \frac{f}{z}}$$

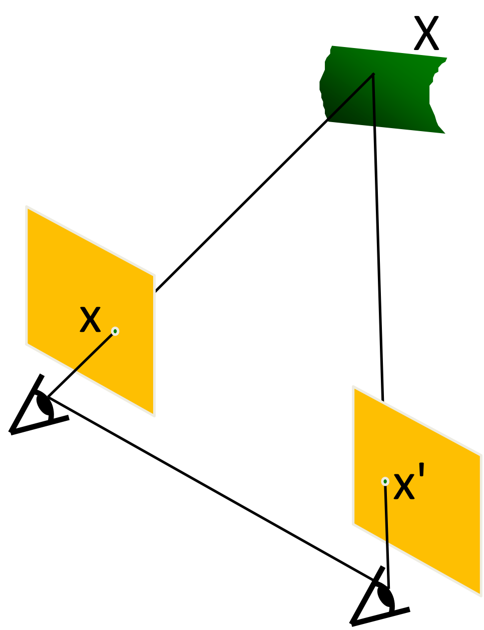
See Chapter 12 of Shapiro and Stockman Text.

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

Disparity is inversely proportional to depth.

Depth from Stereo

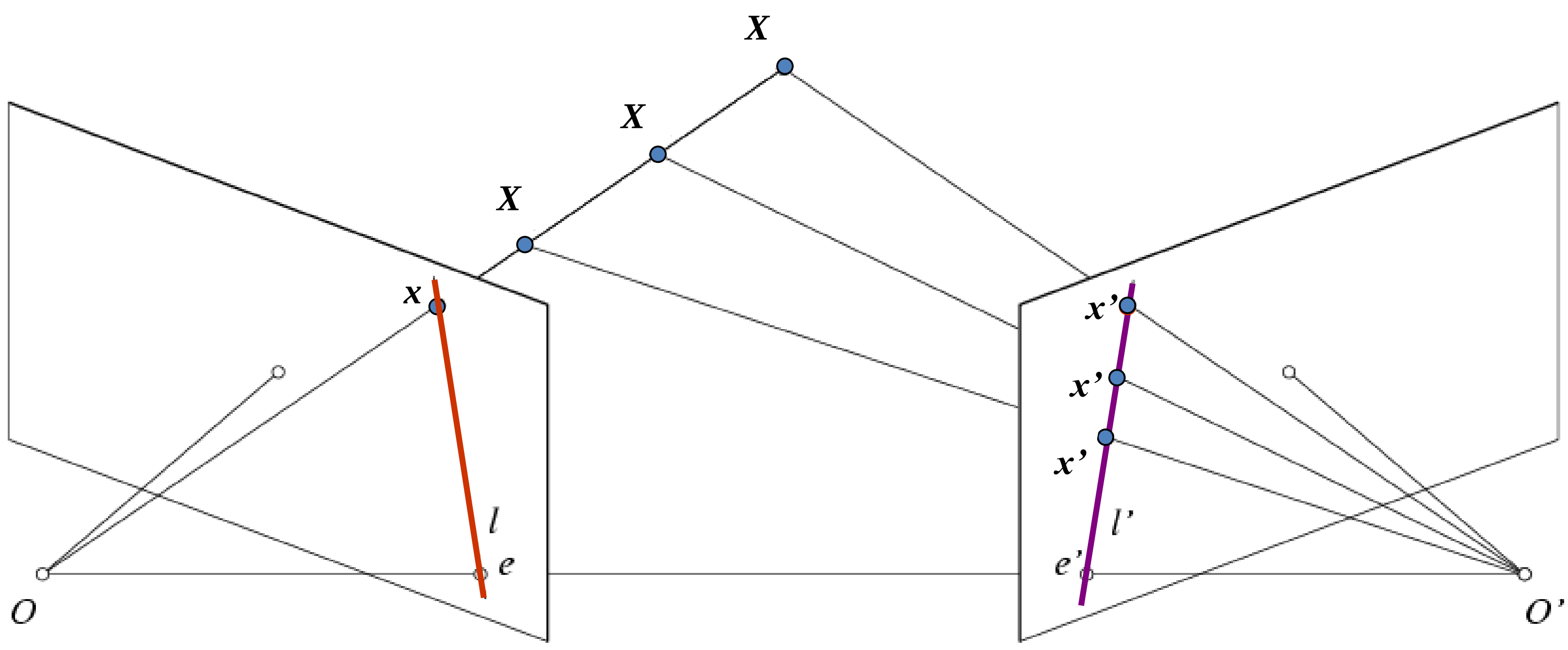
- Goal: recover depth by finding image coordinate x' that corresponds to x
- Sub-Problems
  1. Calibration: How do we recover the relation of the cameras (if not already known)?
  2. Correspondence: How do we search for the matching point x'?



Correspondence Problem

- We have two images taken from cameras with different intrinsic and extrinsic parameters
- How do we match a point in the first image to a point in the second? How can we constrain our search?

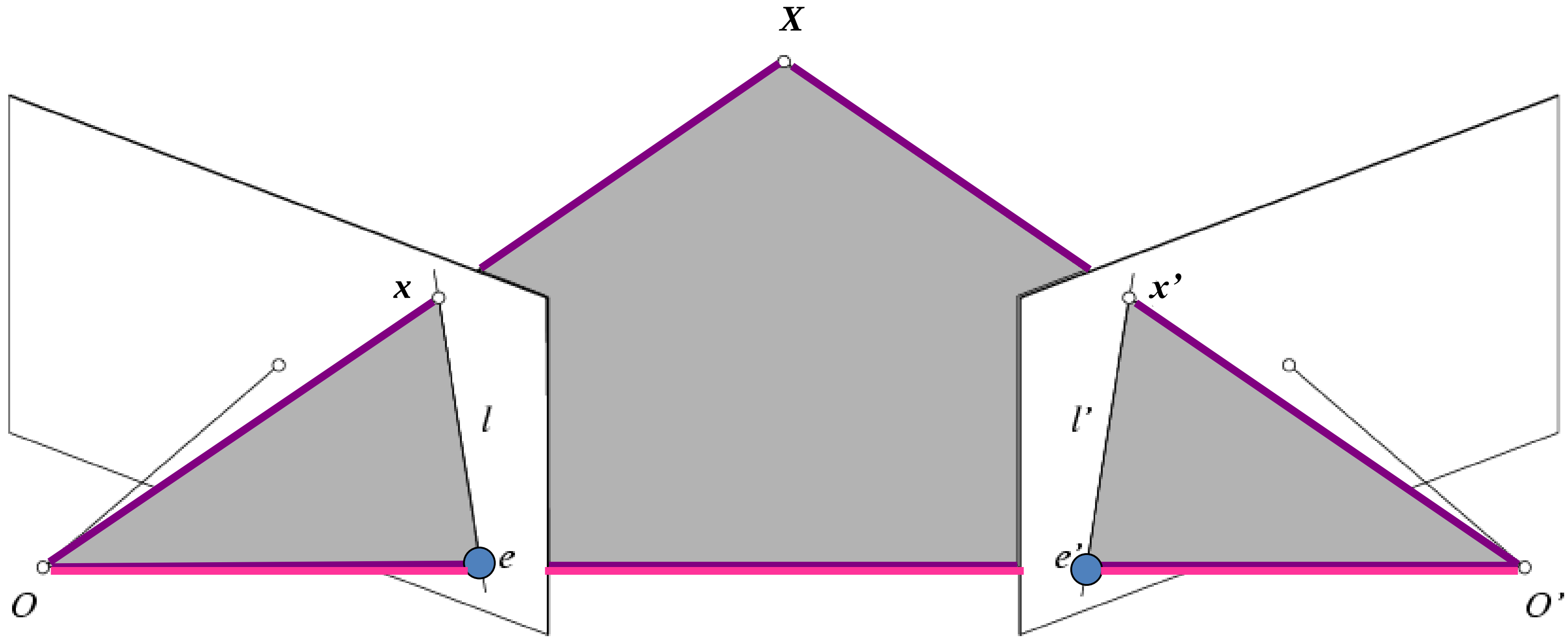
Key idea: Epipolar constraint



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

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

Epipolar geometry: notation

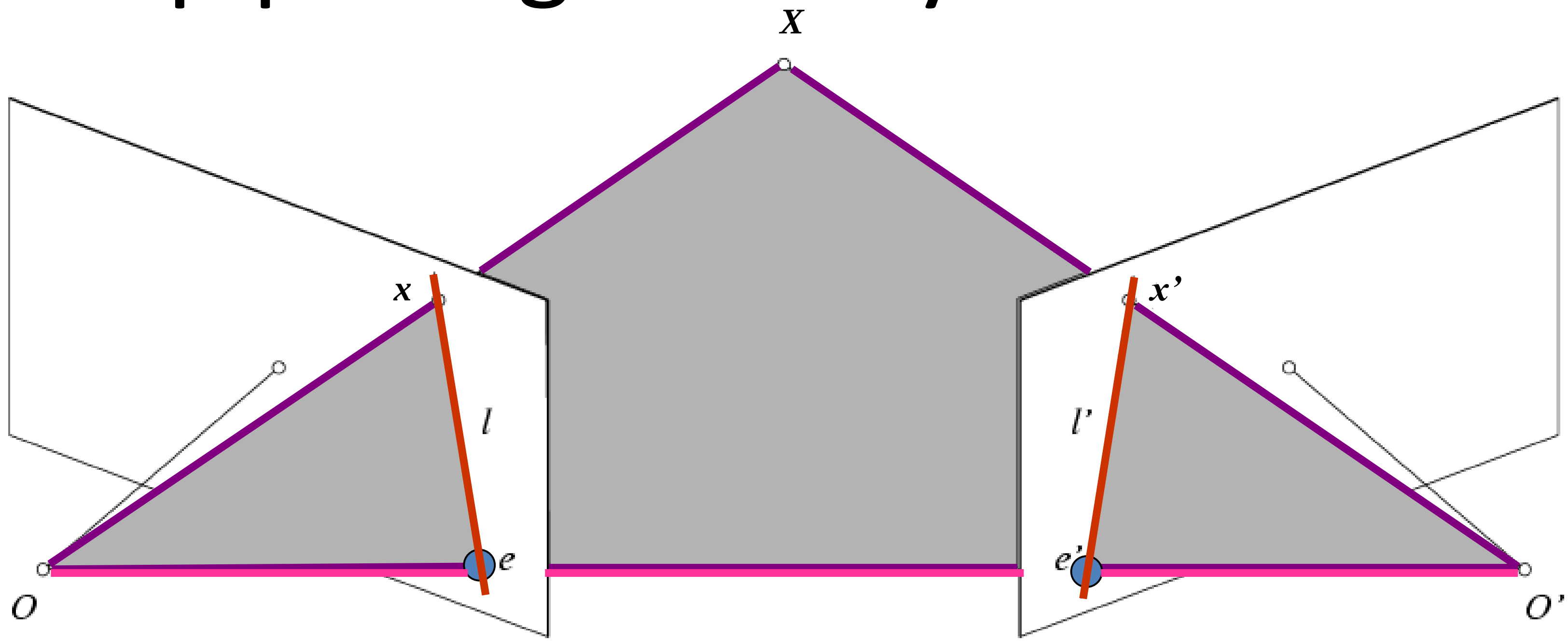


- **Baseline** – line connecting the two camera centers
- **Epipoles**

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

- **Epipolar Plane** – plane containing baseline (1D family)

Epipolar geometry: notation



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

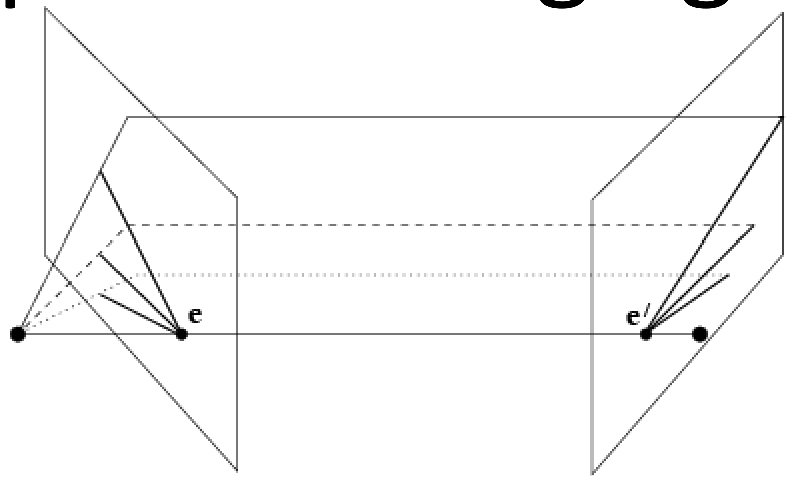
- **Epipoles**

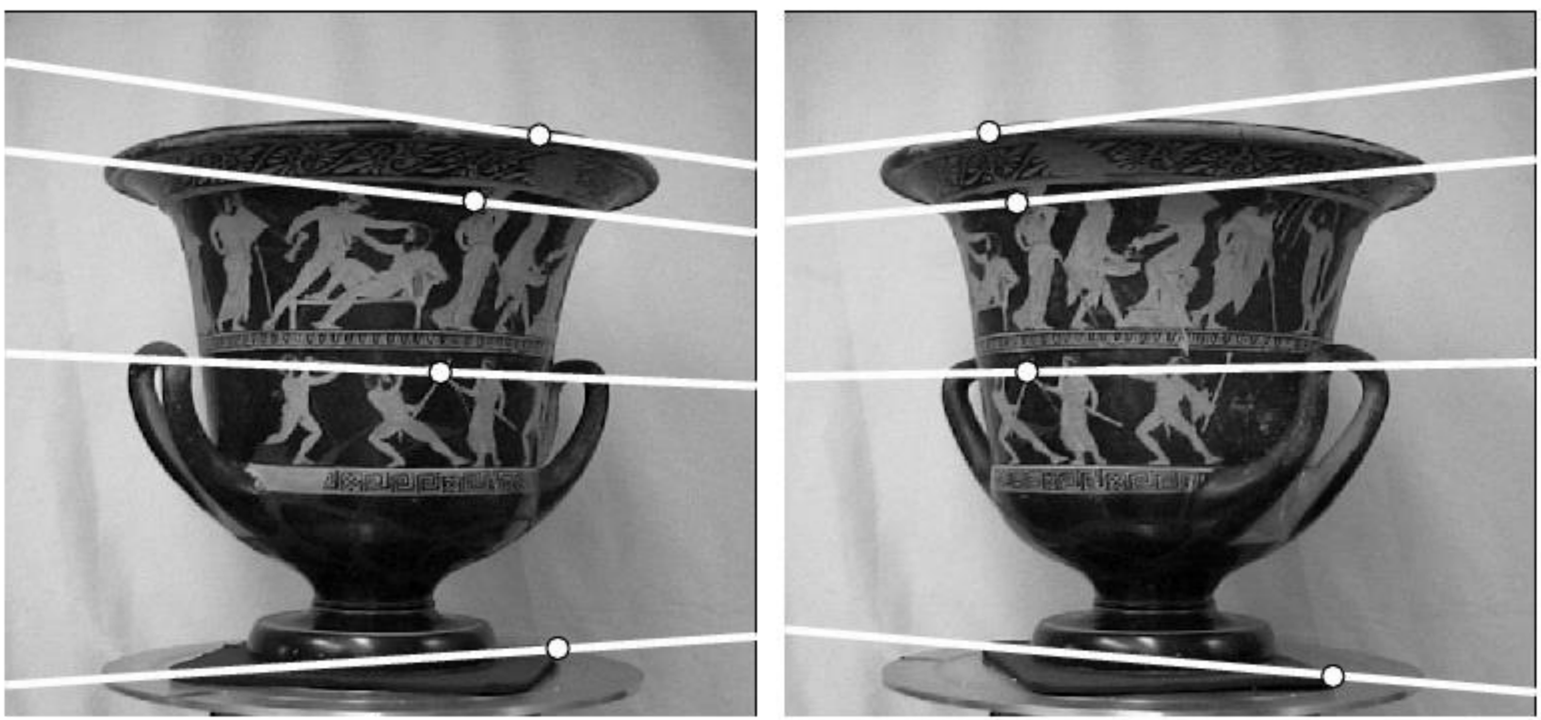
= intersections of baseline with image planes

= projections of the other camera center

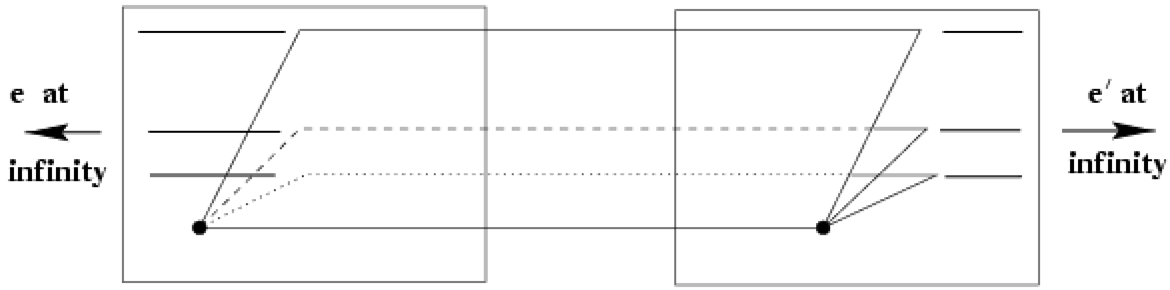
- **Epipolar Plane** – plane containing baseline (1D family)
- **Epipolar Lines** - intersections of epipolar plane with image planes (always come in corresponding pairs)

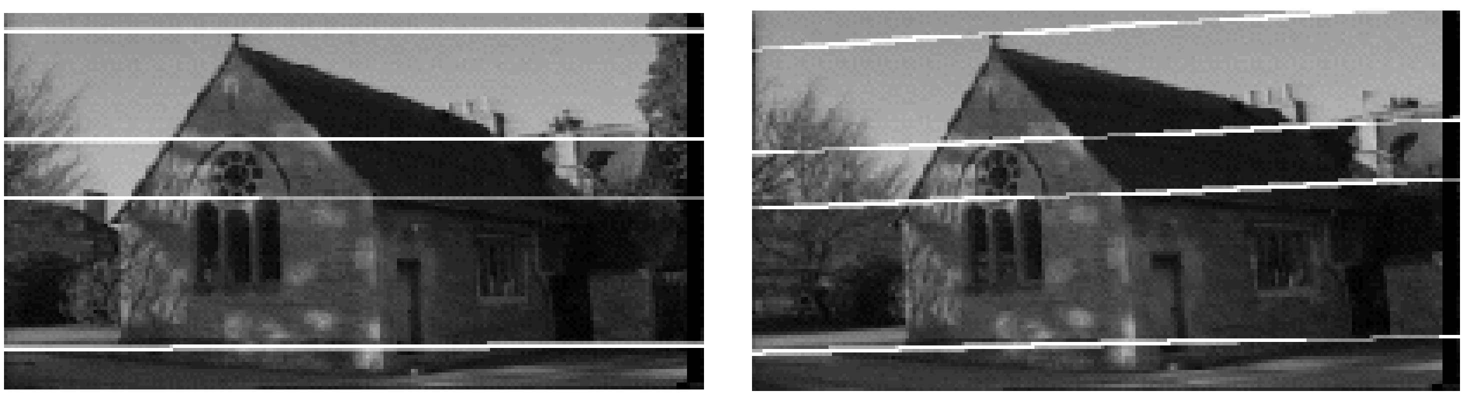
Example: Converging cameras



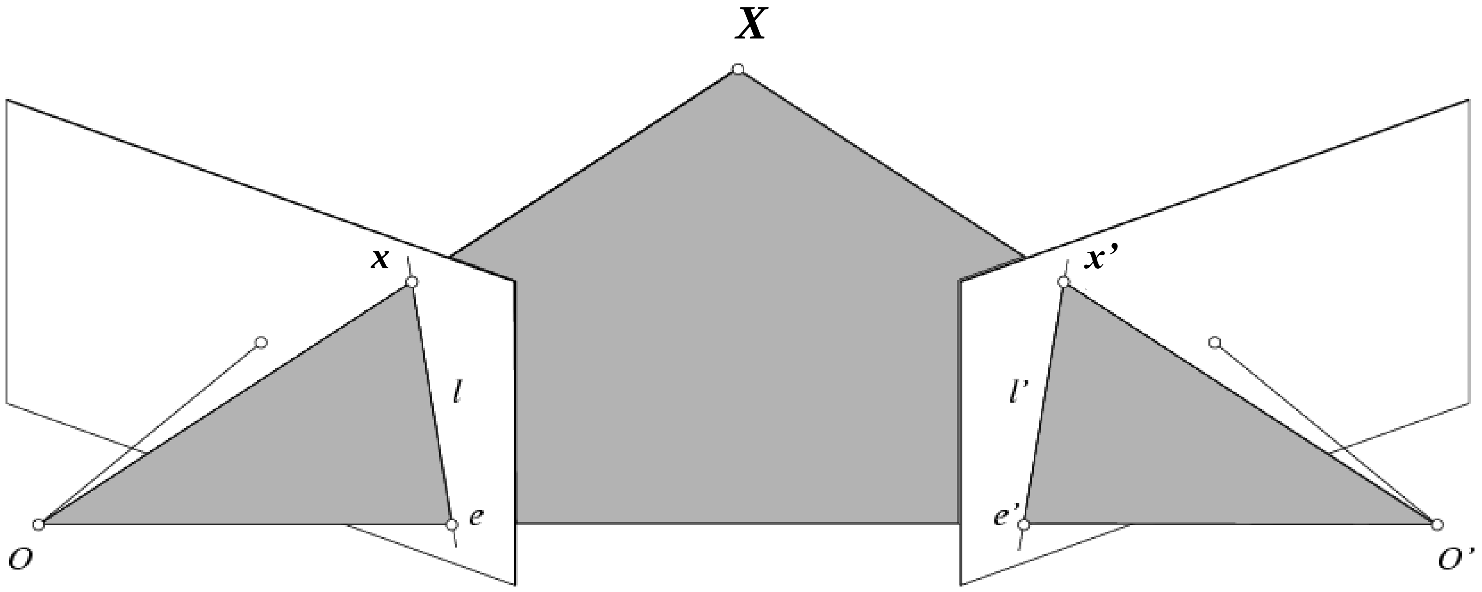


Example: Motion parallel to image plane





Epipolar constraint: Calibrated case



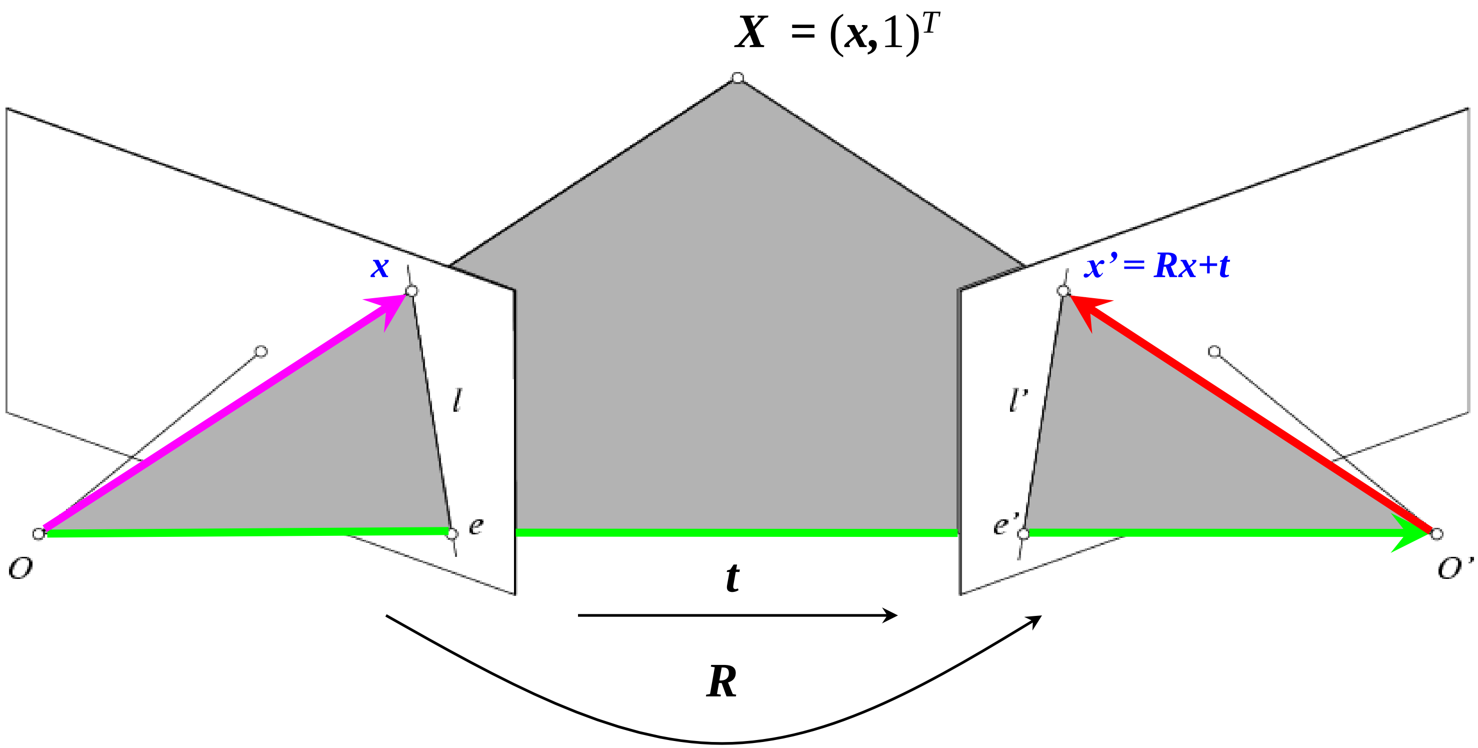
- Assume that the intrinsic and extrinsic parameters of the cameras are known
- We can multiply the projection matrix of each camera (and the image points) by the inverse of the calibration matrix to get *normalized* image coordinates
- We can also set the global coordinate system to the coordinate system of the first camera. Then the projection matrices of the two cameras can be written as **[I | 0]** and **[R | t]**

Simplified Matrices for the 2 Cameras

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} = (\ \mathbf{I} \mid \mathbf{0}\ )$$

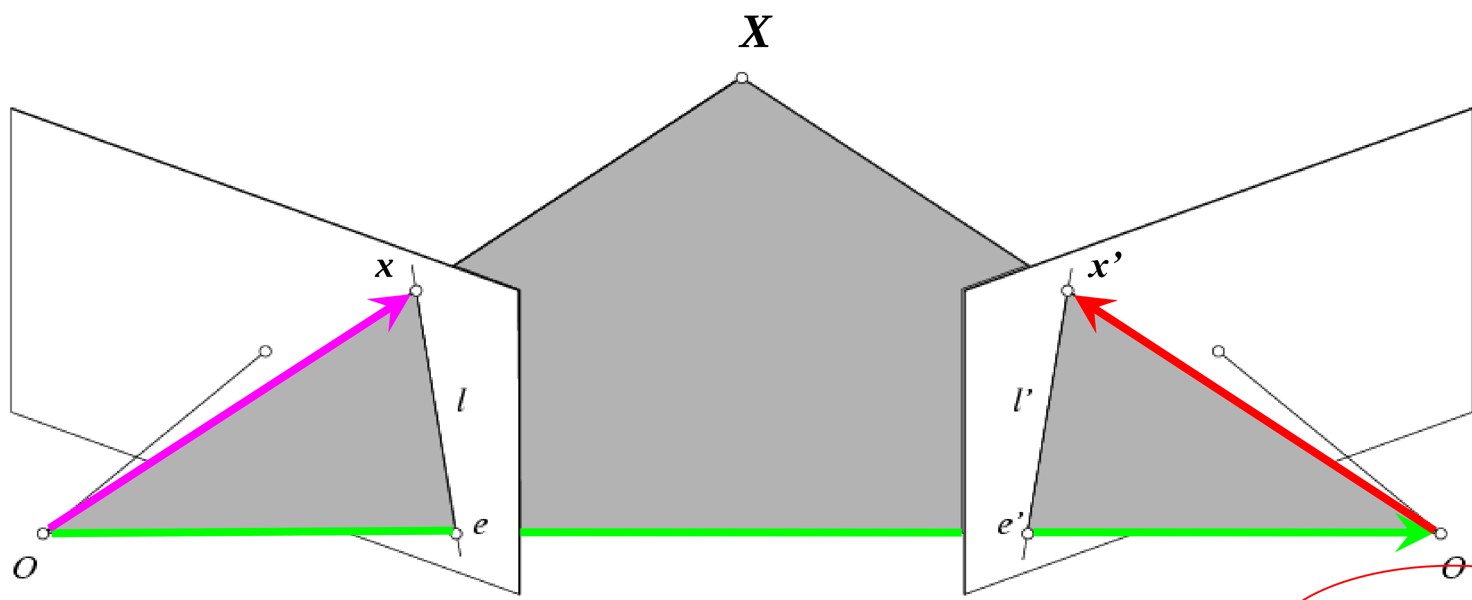
$$\left(\begin{array}{c|c} \mathbf{R} & \mathbf{t} \\ \hline \mathbf{0} & 1 \end{array}\right) = (R \mid T)$$

Epipolar constraint: Calibrated case



The vectors ***Rx***, ***t***, and ***x'*** are coplanar

Epipolar constraint: Calibrated case

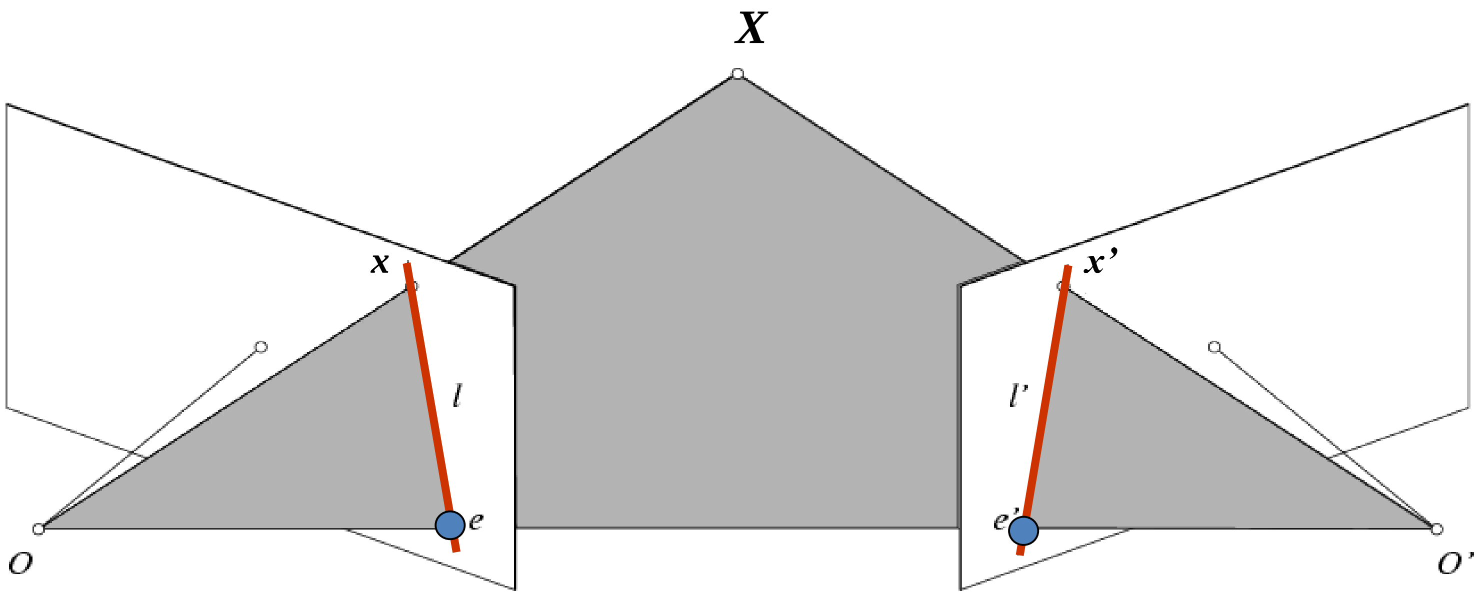


$$\boldsymbol{x}' \cdot [\boldsymbol{t} \times (\boldsymbol{R}\boldsymbol{x})] = 0 \quad \Rightarrow \quad \boldsymbol{x}'^T \boldsymbol{E}\, \boldsymbol{x} = 0 \quad \text{with} \quad \boldsymbol{E} = [\boldsymbol{t}_\times]\boldsymbol{R}$$

Essential Matrix E
(Longuet-Higgins, 1981)

The vectors $\boldsymbol{Rx}$, $\boldsymbol{t}$, and $\boldsymbol{x'}$ are coplanar

Epipolar constraint: Calibrated case



$$\mathbf{x}' \cdot [\boldsymbol{t} \times (\boldsymbol{R}\mathbf{x})] = 0 \quad \Rightarrow \quad \mathbf{x}'^{T} \boldsymbol{E}\, \mathbf{x} = 0 \quad \text{with} \quad \boldsymbol{E} = [\boldsymbol{t}_{\times}]\boldsymbol{R}$$

- $\boldsymbol{E}\,\boldsymbol{x}$ is the epipolar line associated with $\boldsymbol{x}$ ($\boldsymbol{l}' = \boldsymbol{E}\,\boldsymbol{x}$)
- $\boldsymbol{E}^T\boldsymbol{x}'$ is the epipolar line associated with $\boldsymbol{x}'$ ($\boldsymbol{l} = \boldsymbol{E}^T\boldsymbol{x}'$)
- $\boldsymbol{E}\,\boldsymbol{e} = 0$ and $\boldsymbol{E}^T\boldsymbol{e}' = 0$
- $\boldsymbol{E}$ is singular (rank two)
- $\boldsymbol{E}$ has five degrees of freedom

Moving on to stereo...

Fuse a calibrated binocular stereo pair to produce a depth image

image 1

image 2

Dense depth map

Many of these slides adapted from Steve Seitz and Lana Lazebnik

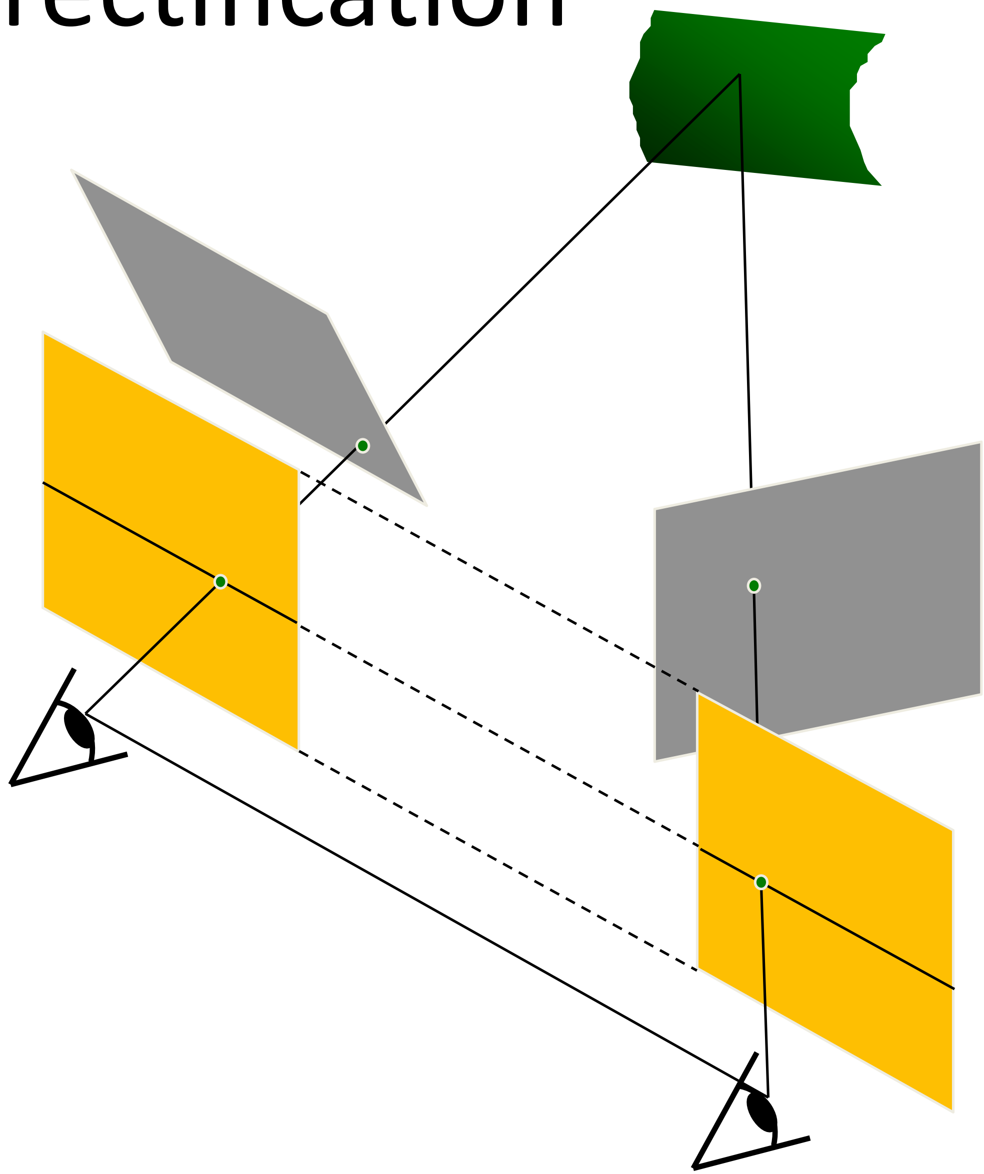
Stereo image rectification

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

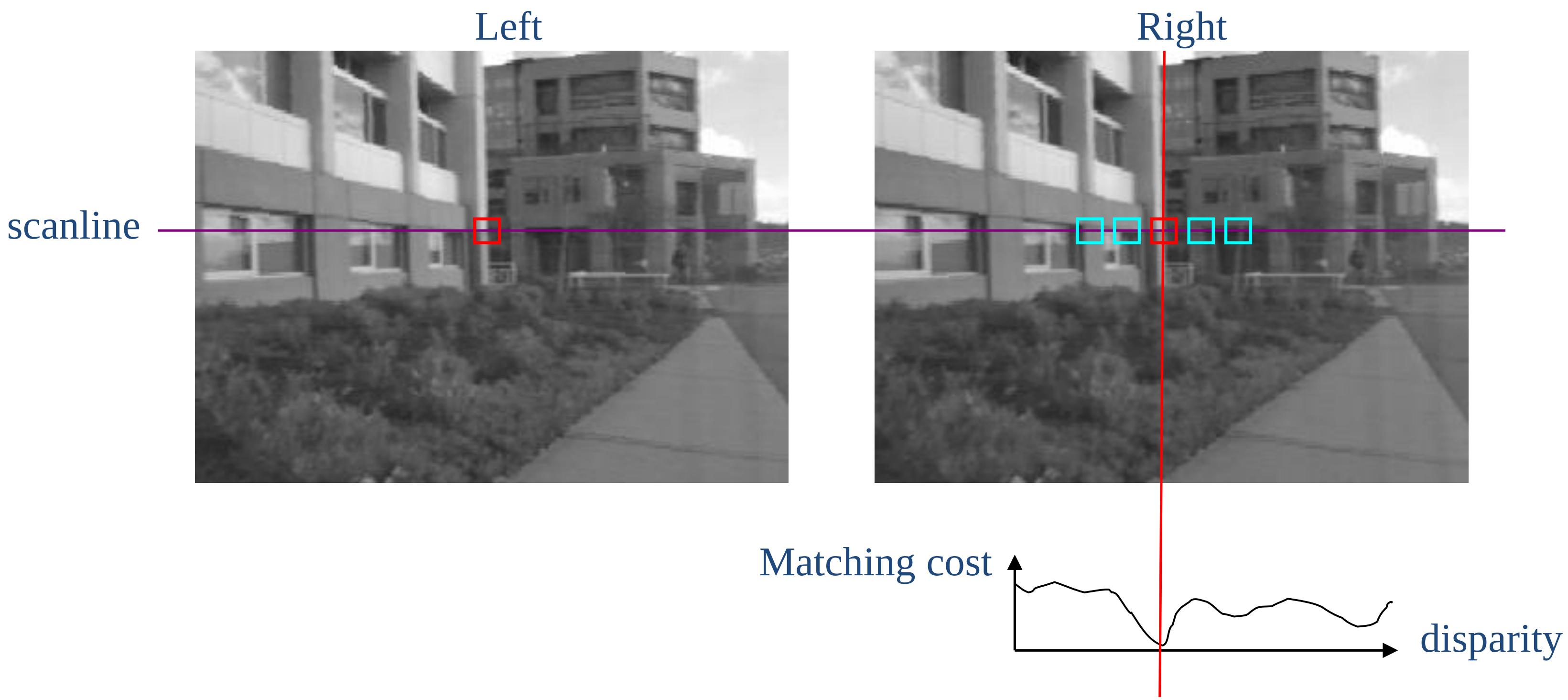
- Pixel motion is horizontal after this transformation

- Two homographies (3x3 transform), one for each input image reprojection

➢ C. Loop and Z. Zhang. [Computing Rectifying Homographies for Stereo Vision](). IEEE Conf. Computer Vision and Pattern Recognition, 1999.

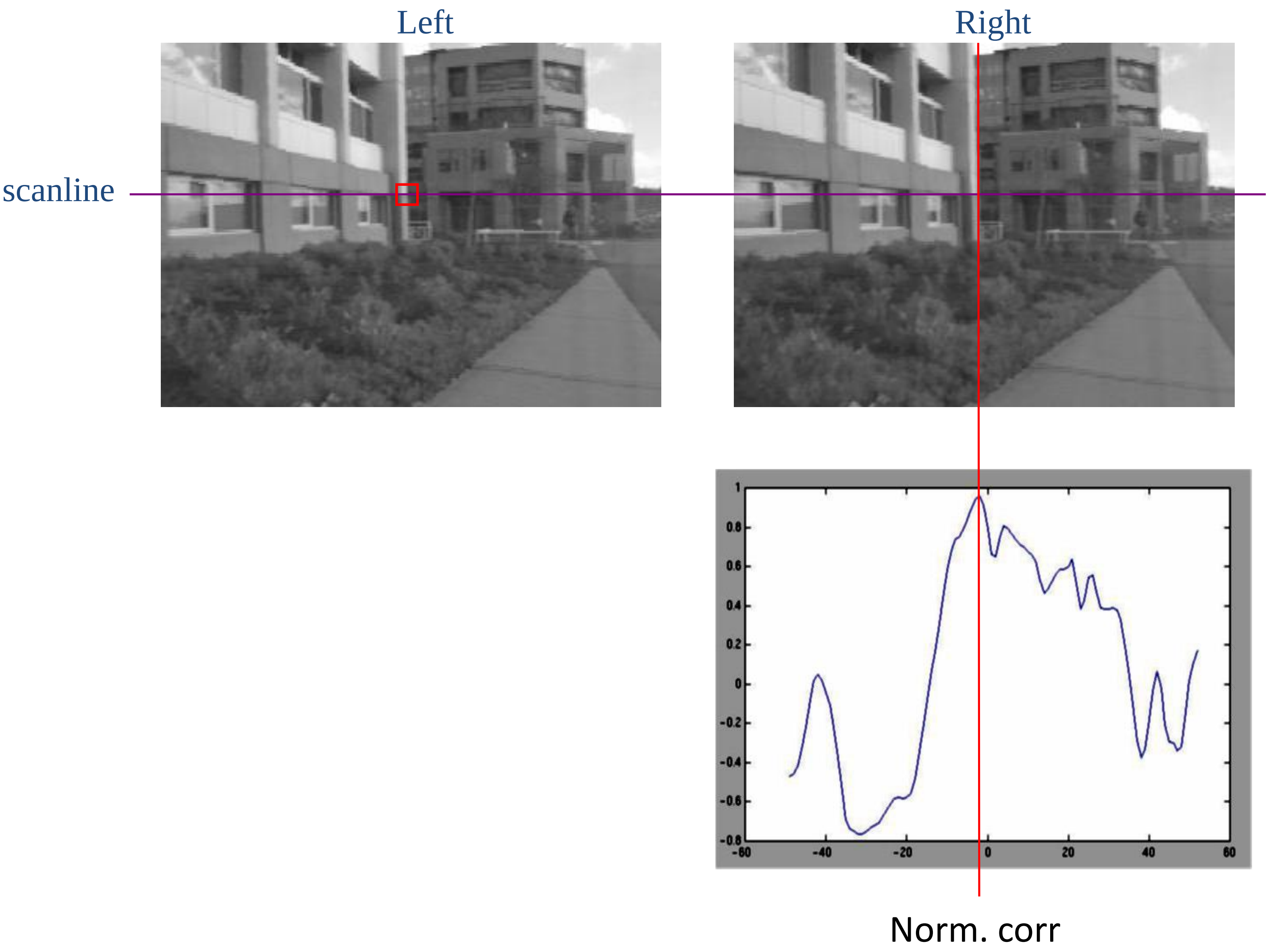


Example



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

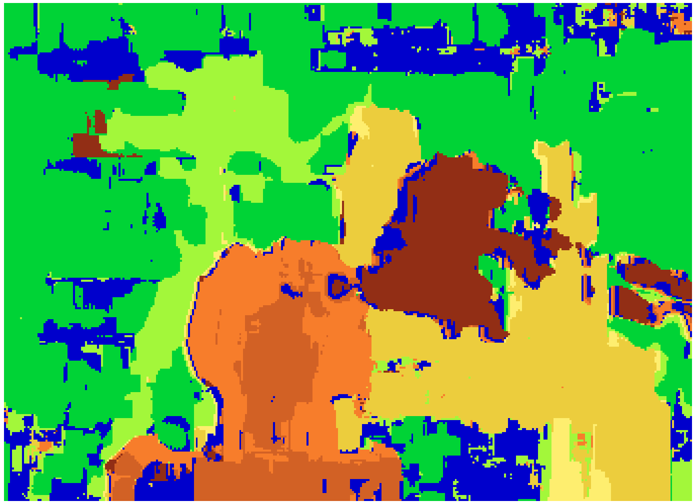
Correspondence search



Results with window search

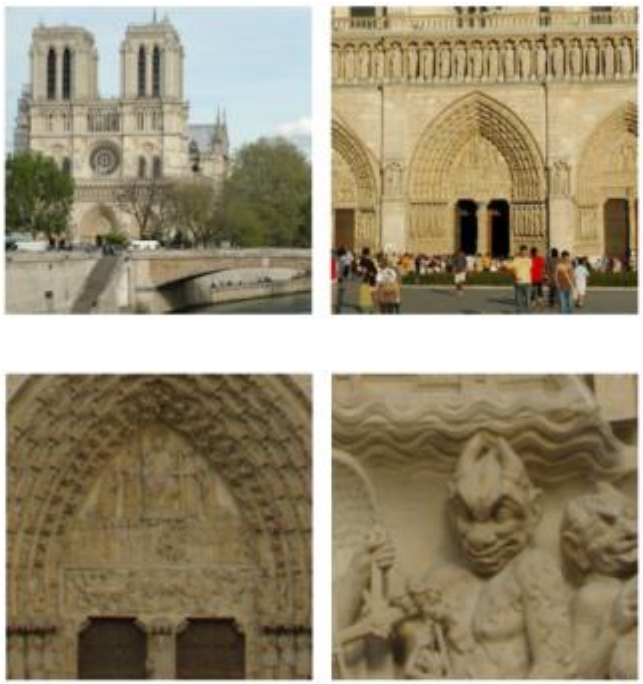
Data

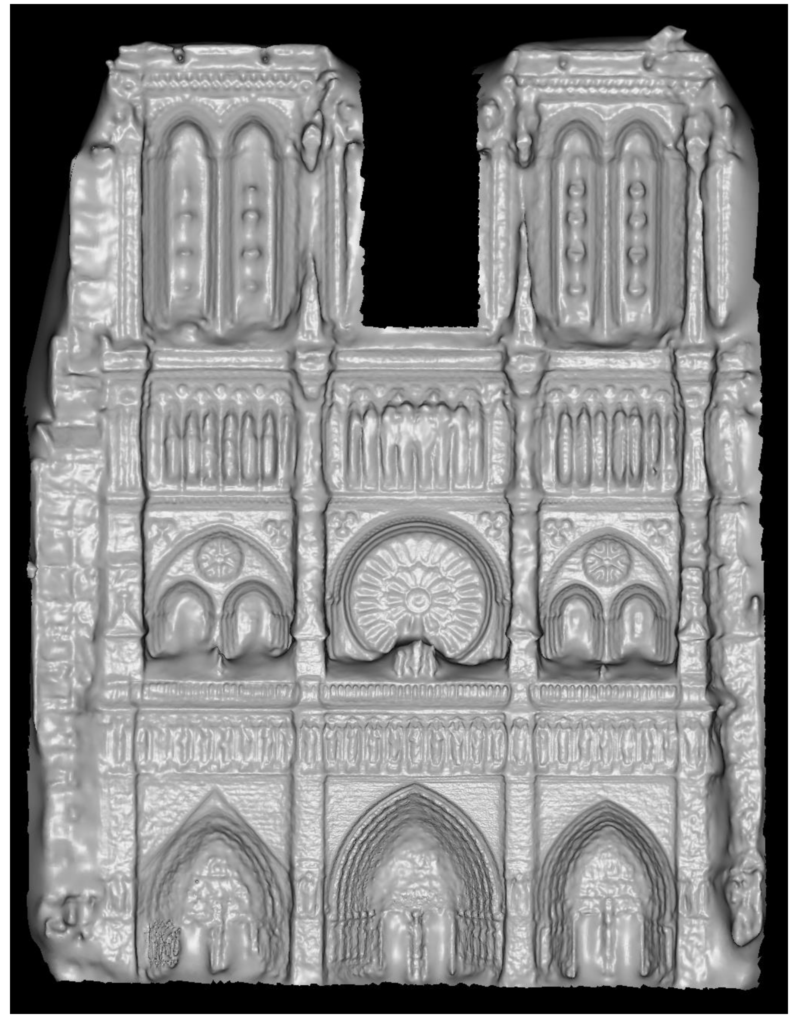
Window-based matching



Ground truth

Using more than two images





[Multi-View Stereo for Community Photo Collections](#)
M. Goesele, N. Snavely, B. Curless, H. Hoppe, S. Seitz
Proceedings of [ICCV 2007](#),

3D model

- "Digital copy" of real object
- Allows us to
  - Inspect details of object
  - Measure properties
  - Reproduce in different material
- Many applications
  - Cultural heritage preservation
  - Computer games and movies
  - City modelling
  - E-commerce

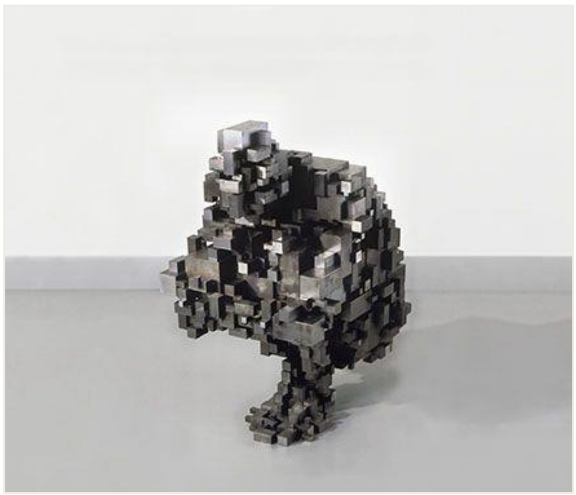




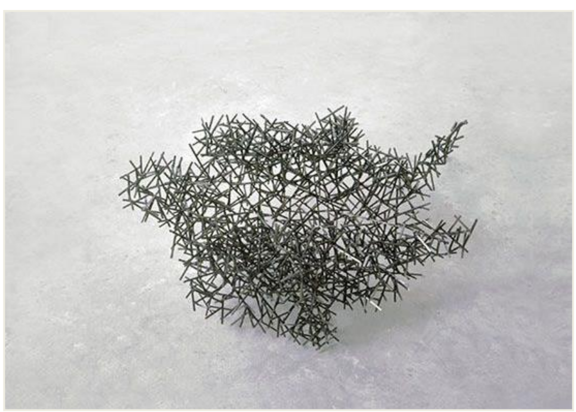
Applications: cultural heritage

SCULPTEUR European project

Applications: art



Block Works Precipitate III 2004
Mild steel blocks 80 x 46 x 66 cm

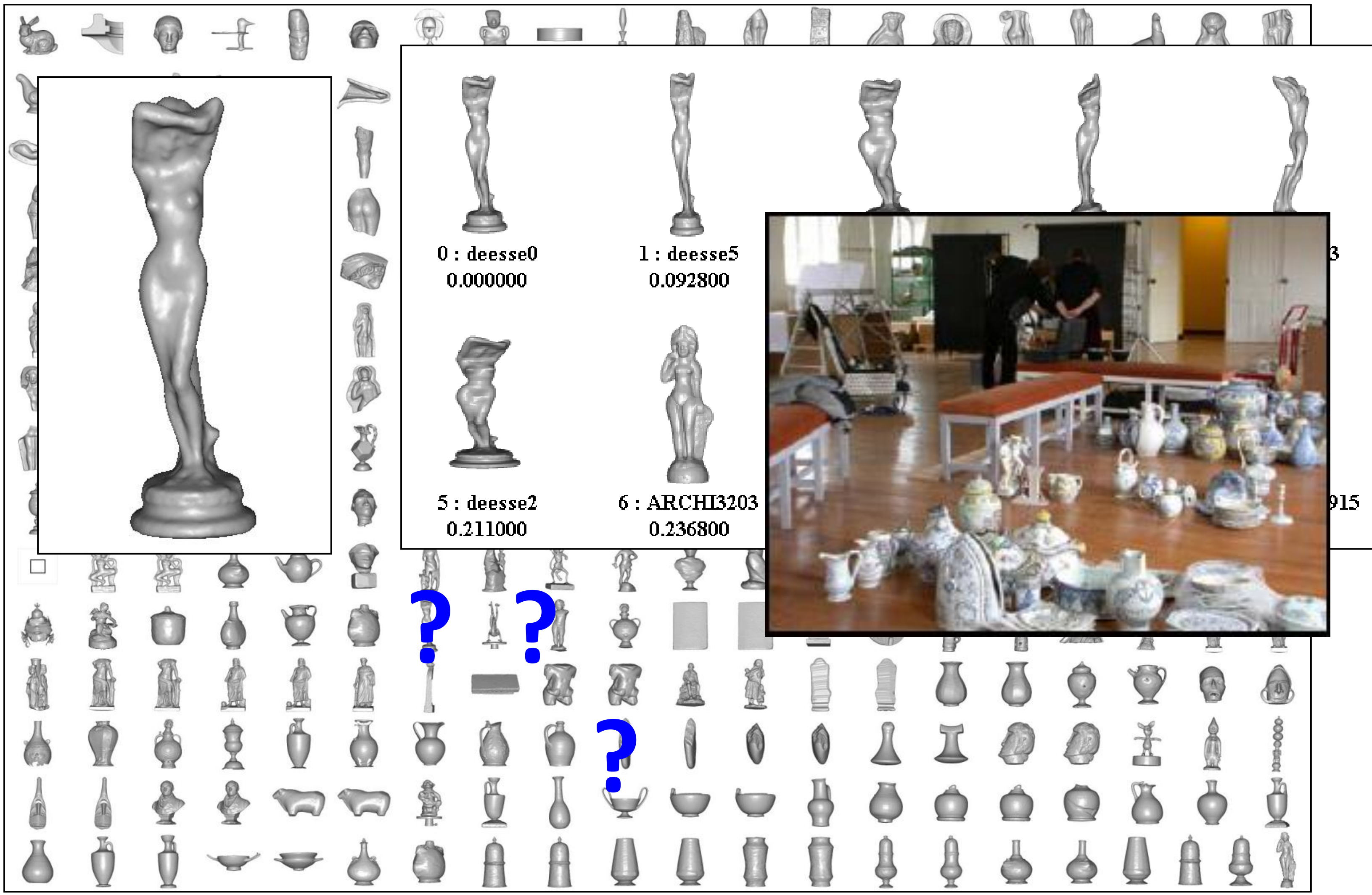


Domain Series Domain VIII Crouching
1999 *Mild steel bar* 81 x 59 x 63 cm

Applications: structure engineering

BODY / SPACE / FRAME, Antony Gormley, Lelystad, Holland

Applications: 3D indexation

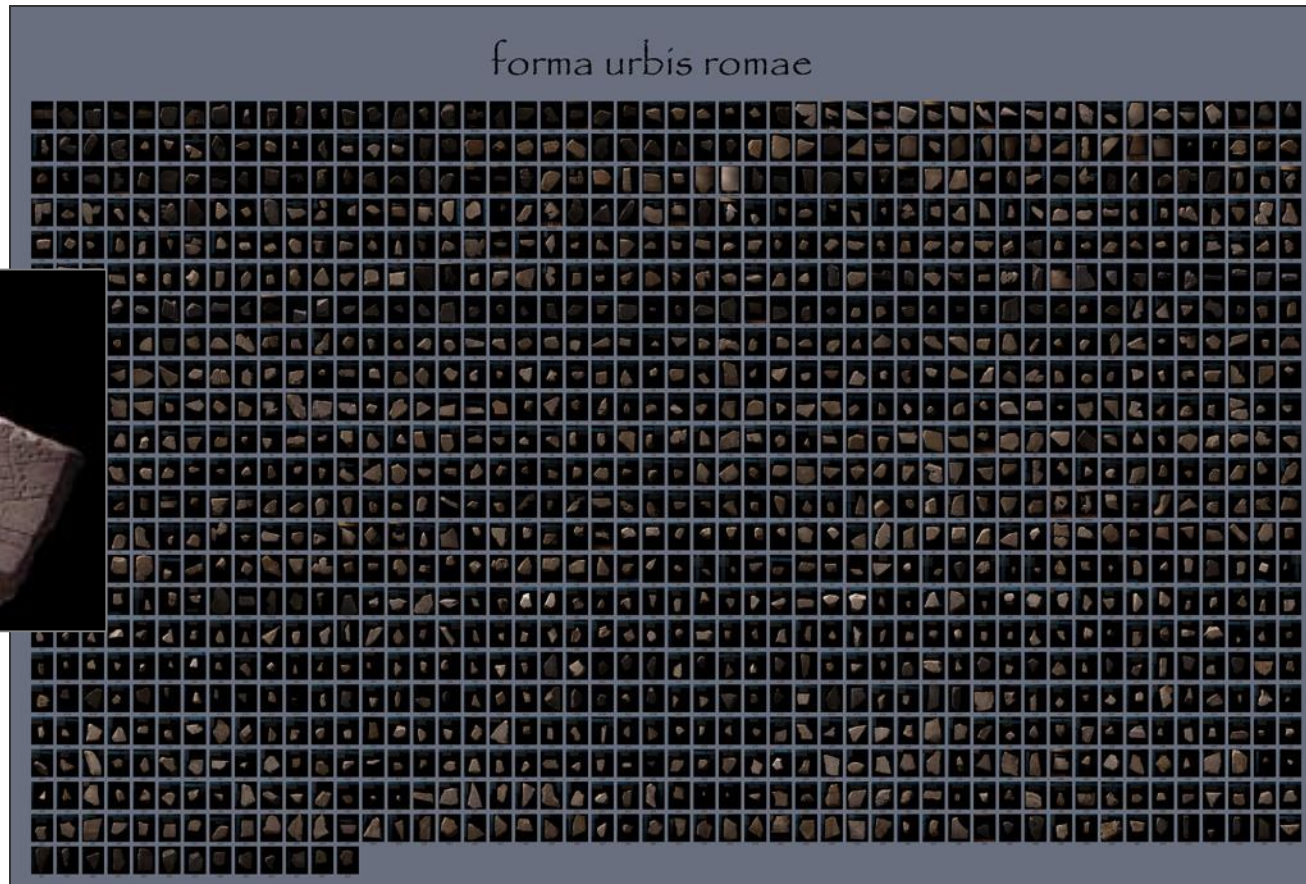


Applications: archaeology

- "forma urbis romae" project

Fragments of the City: Stanford's Digital Forma Urbis Romae Project
David Koller, Jennifer Trimble, Tina Najbjerg, Natasha Gelfand, Marc Levoy
Proc. Third Williams Symposium on Classical Architecture, Journal of Roman Archaeology supplement, 2006.

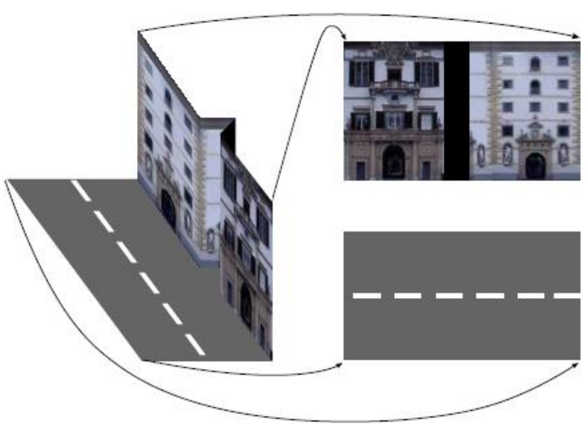
1186 fragments



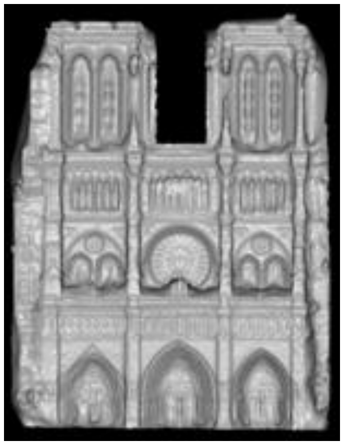
Applications: large scale modelling

[Furukawa10]

[Pollefeys08]

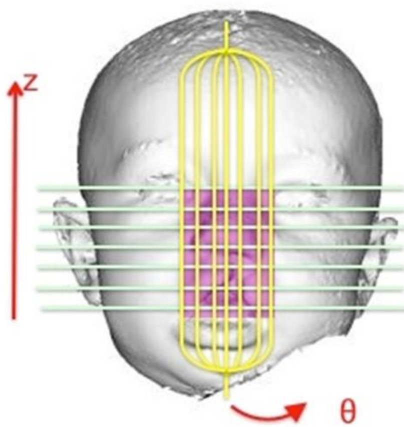


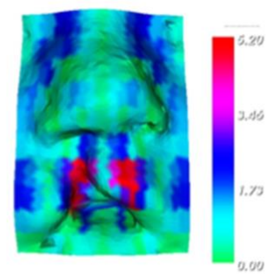
[Cornelis08]



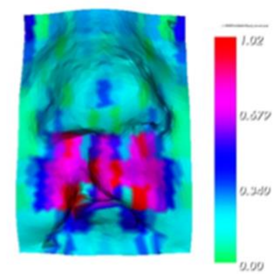
[Goesele07]

Applications: Medicine

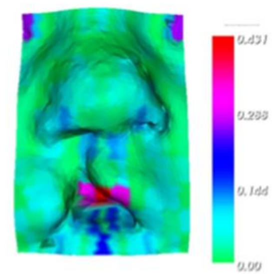




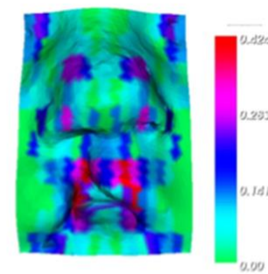
(a) Radius difference



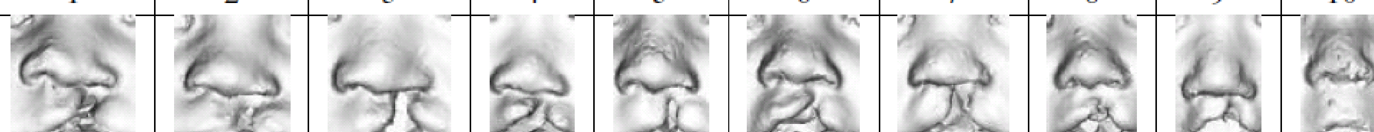
(b) Angle difference



(c) Curvature difference

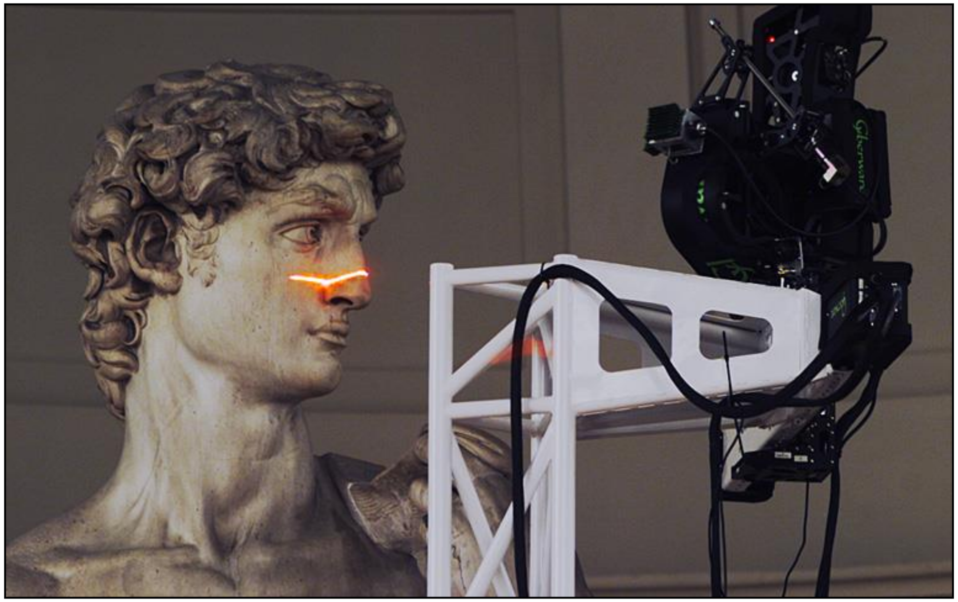


(d) Edge difference

expert's order	1	2	3	4	5	6	7	8	9	10
images										
learning	1	3	2	4	5	6	8	9	7	10
a-lmk	1	2	3	5	6	4	8	7	9	10
mirror	1	2	4	8	5	6	9	3	7	10
m-lmk	1	2	3	4	5	6	9	7	10	8
plane	1	2	3	5	4	6	7	9	10	8

Scanning technologies

- Laser scanner, coordinate measuring machine
  - Very accurate
  - Very Expensive
  - Complicated to use

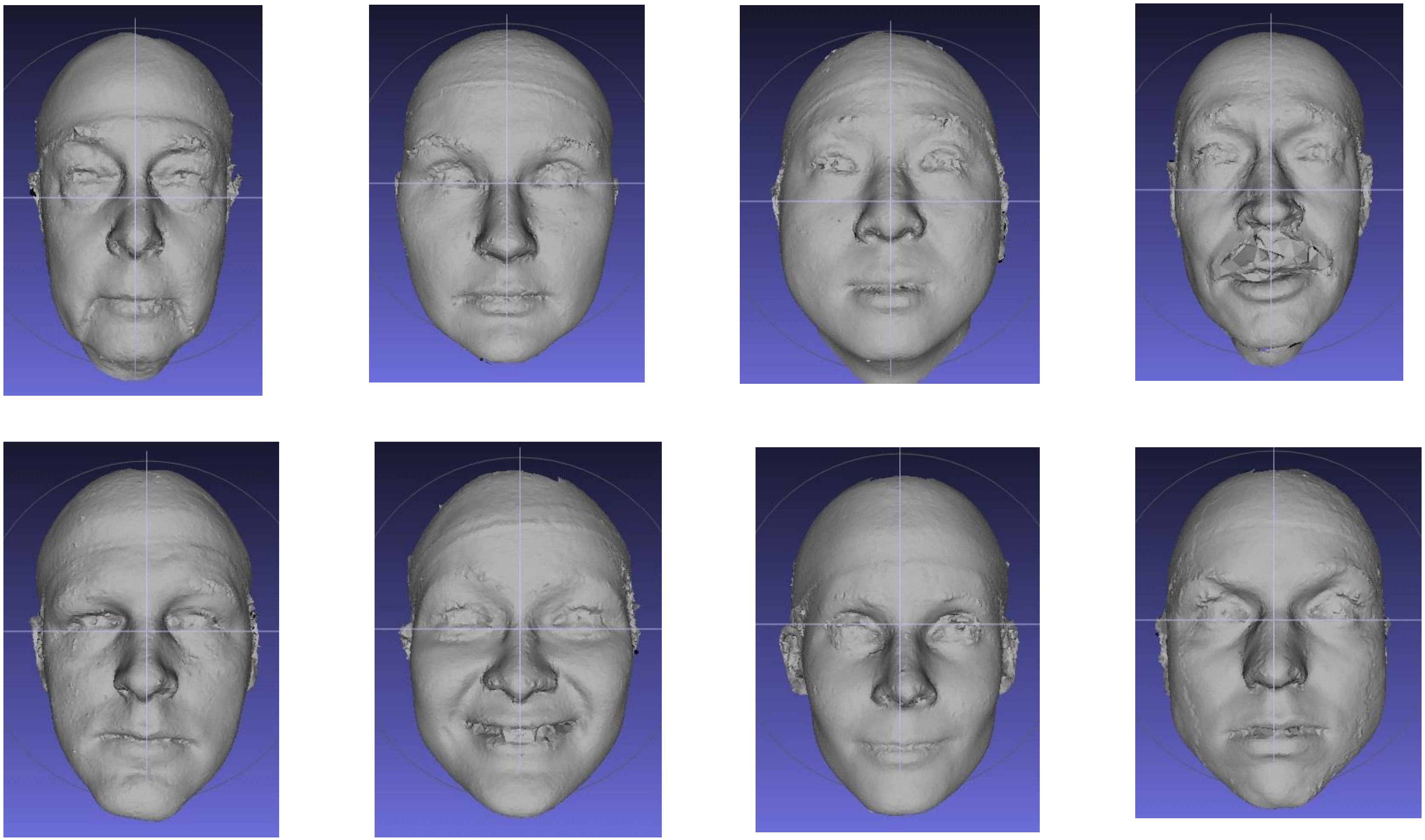


Minolta

Contura CMM

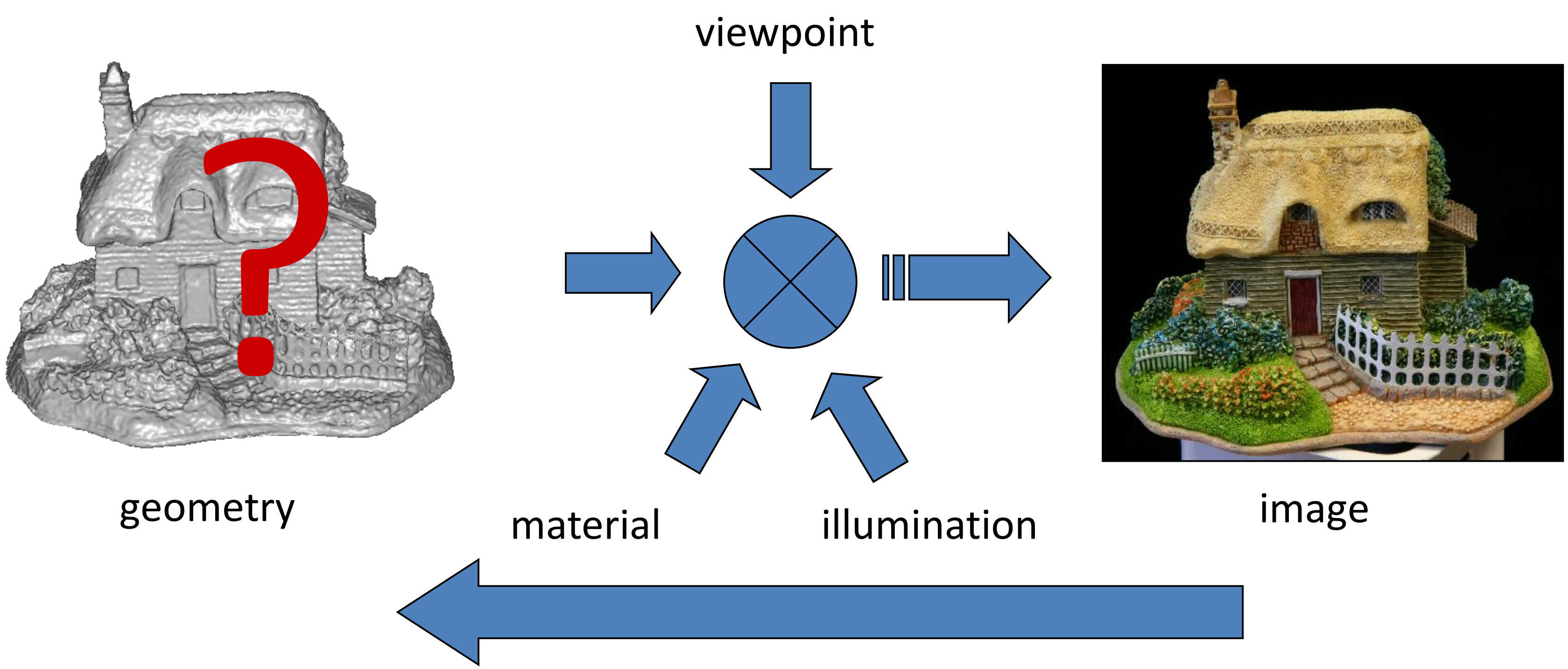
Medical Scanning System

The “Us” Data Set (subset)



3d shape from photographs

“Estimate a 3d shape that would generate the input photographs given the same material, viewpoints and illumination”



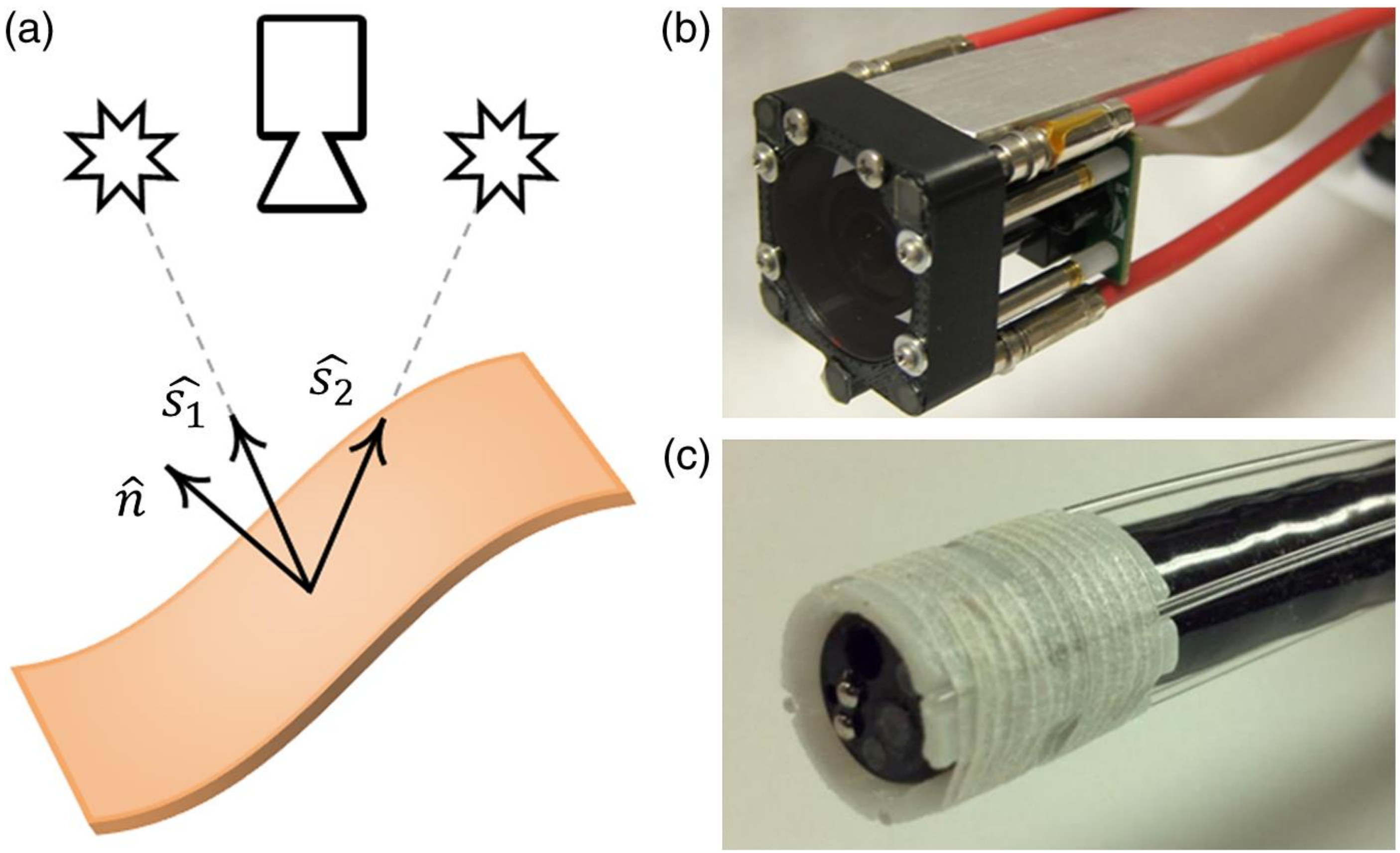
Photometric Stereo

- Estimate the surface normals of a given scene given multiple 2D images taken from the *same* viewpoint, but under *different lighting* conditions.
- Basic photometric stereo required a Lambertian reflectance model:

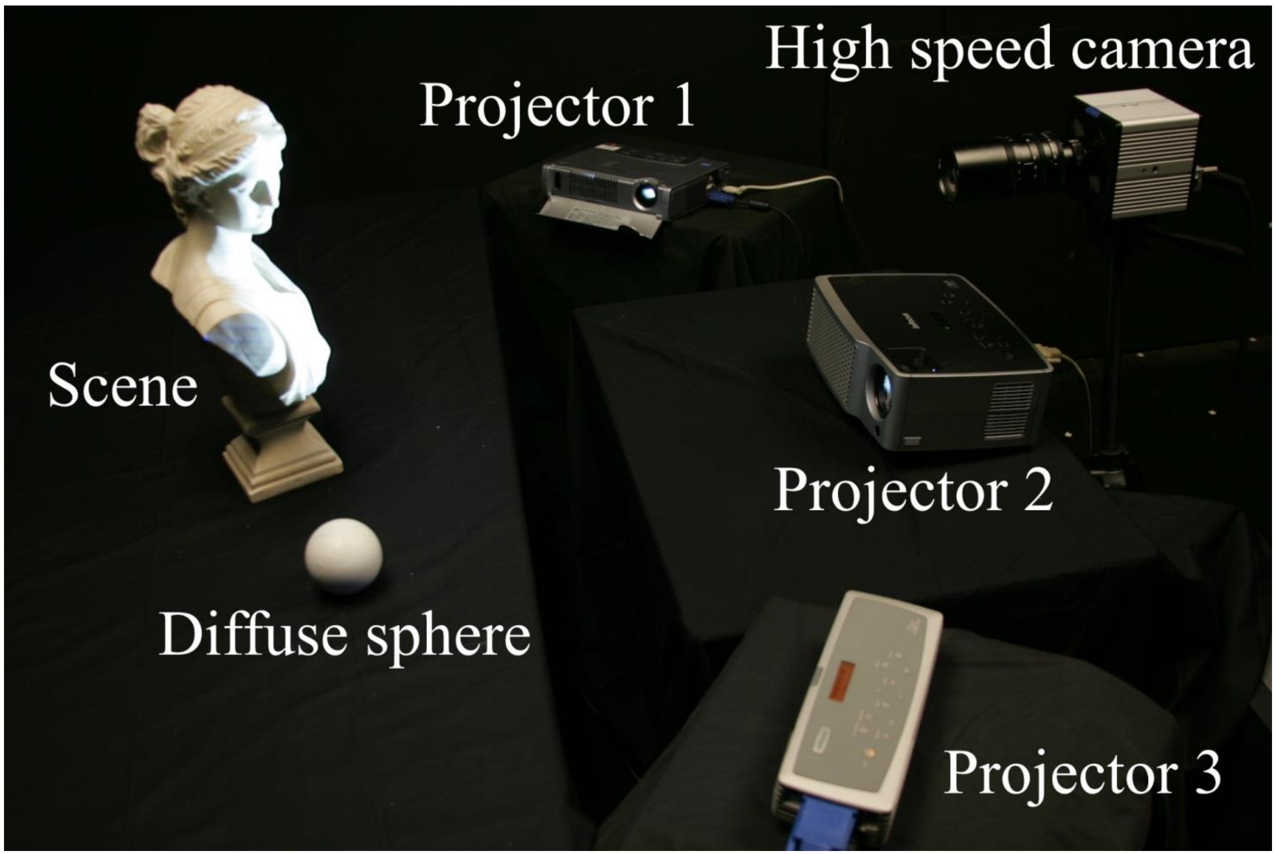
$$I = \rho\, \mathbf{n} \cdot \mathbf{v}$$

where I is pixel intensity, **n** is the normal, **v** is the lighting direction, and ρ is diffuse albedo constant, which is a reflection coefficient.

Basic Photometric Stereo



Basic Photometric Stereo



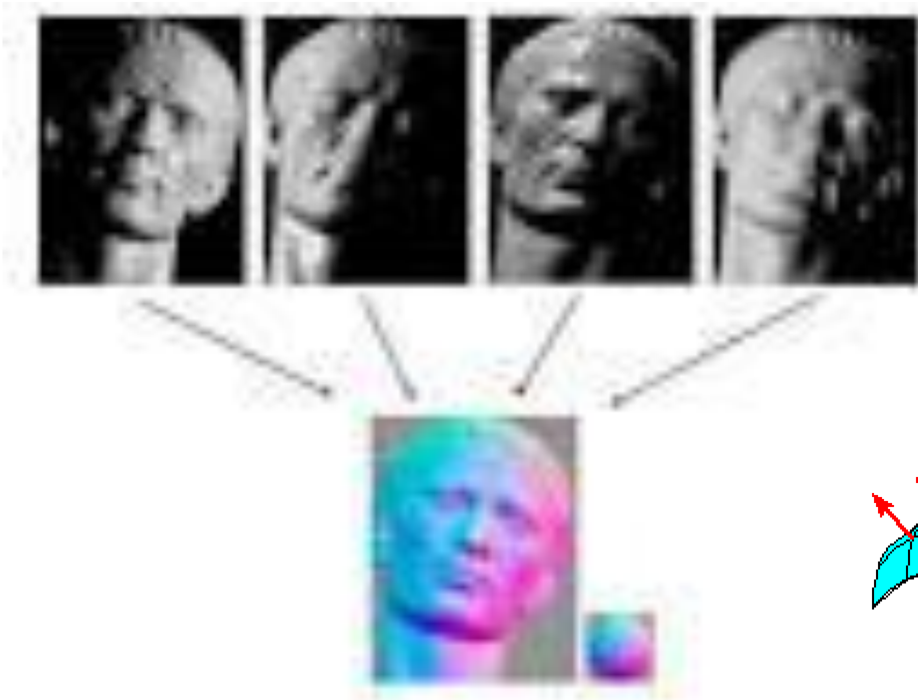
Basic Photometric Stereo

- K light sources
- Lead to K images $R_1(p,q)$, ...,$R_K(p,q)$ each from just one of the light sources being on
- For any (p,q), we get K intensities I_1,...I_K
- Leads to a set of linear equations of the form

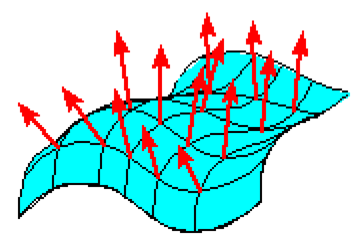
  $I_k = \rho \mathbf{n} \bullet \mathbf{v}_k$
- Solving leads to a surface normal map.

Photometric Stereo

Inputs



3D normals



3d shape from photographs

Photograph based 3d reconstruction is:

- ✓ practical
- ✓ fast
- ✓ non-intrusive
- ✓ low cost
- ✓ Easily deployable outdoors
- ✗ "low" accuracy
- ✗ Results depend on material properties

Reconstruction

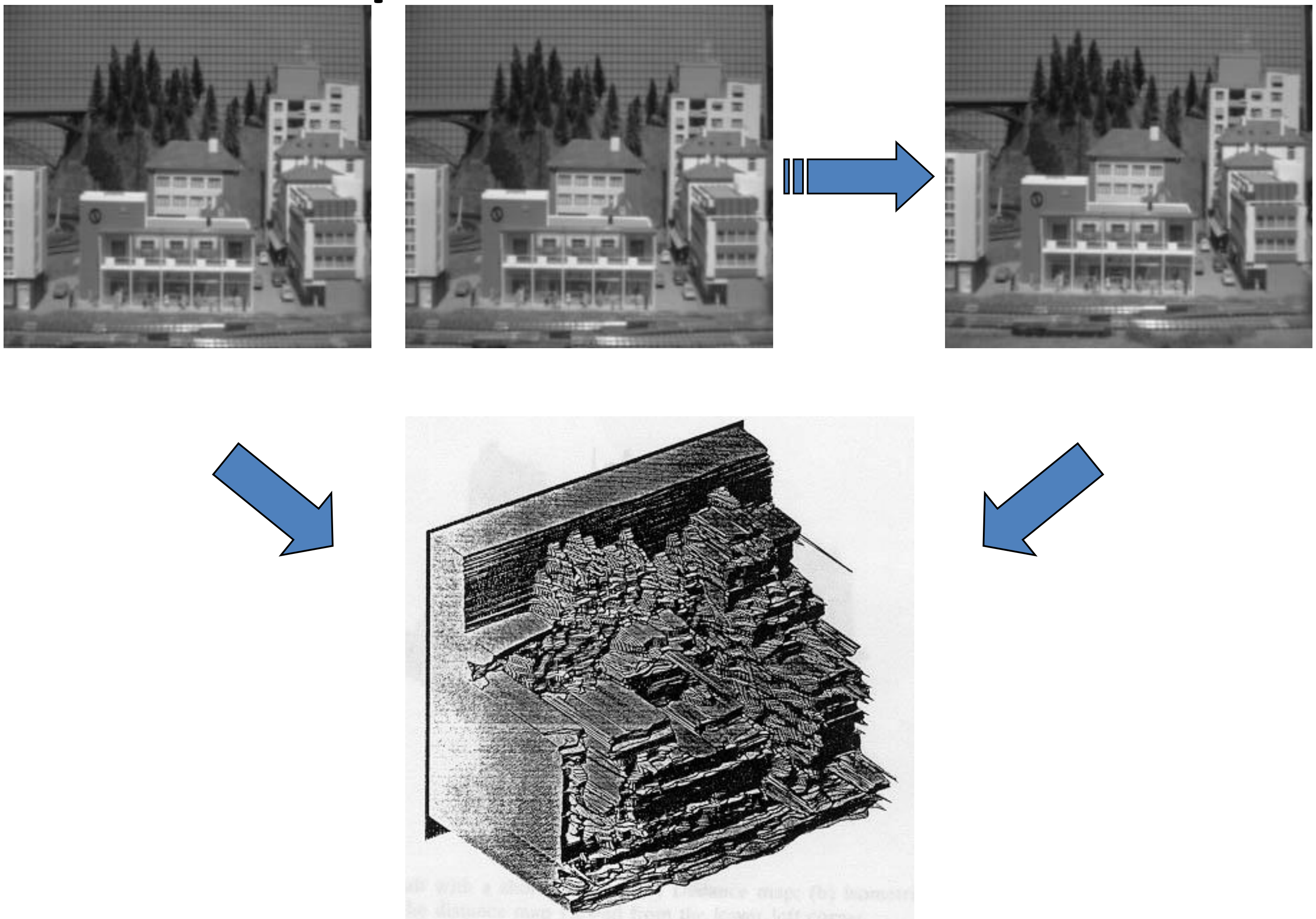
- Generic problem formulation: given several images of the same object or scene, compute a representation of its 3D shape



Reconstruction

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

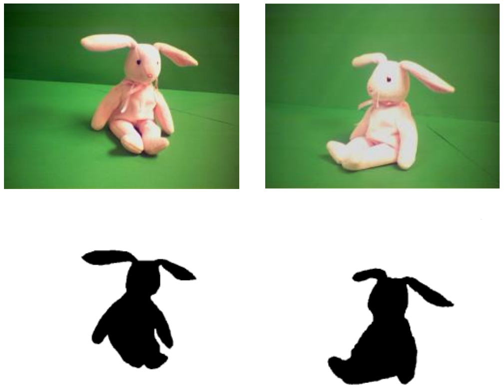
Multiple-baseline stereo

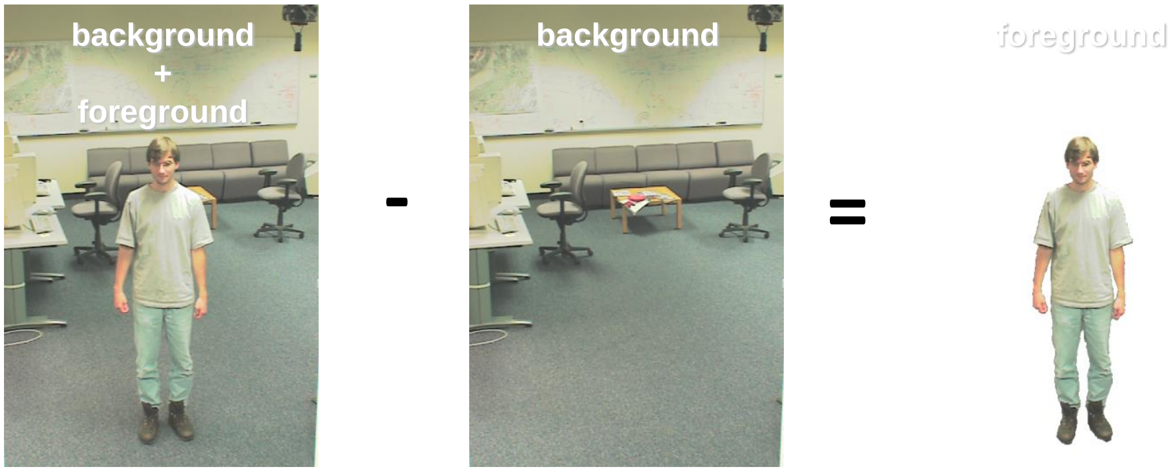


M. Okutomi and T. Kanade, “A Multiple-Baseline Stereo System,” IEEE Trans. on Pattern Analysis and Machine Intelligence, 15(4):353-363 (1993).

Reconstruction from silhouettes

- Can be computed robustly
- Can be computed efficiently

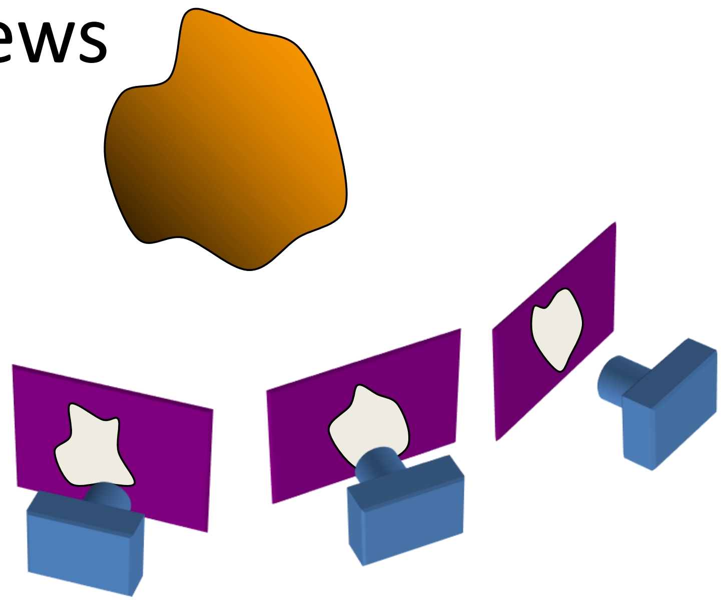




Reconstruction from Silhouettes

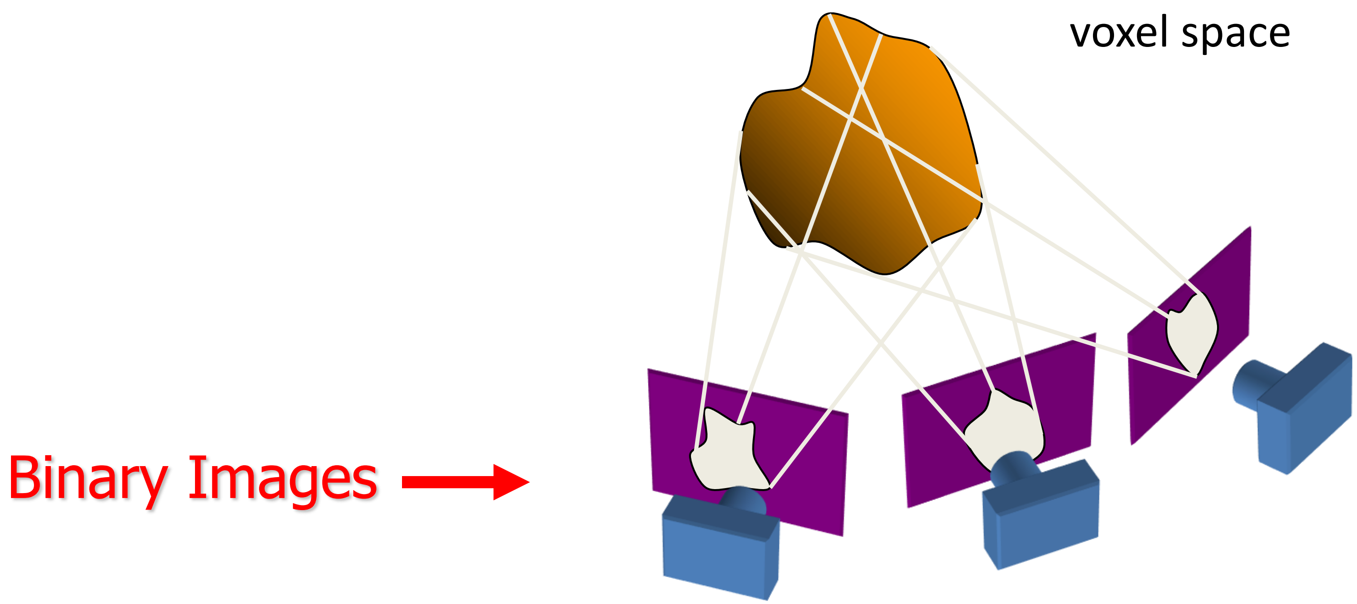
- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in **all** views

Binary Images



Reconstruction from Silhouettes

- The case of binary images: a voxel is photo-consistent if it lies inside the object's silhouette in all views



Finding the silhouette-consistent shape (*visual hull*):

- *Backproject* each silhouette
- Intersect backprojected volumes

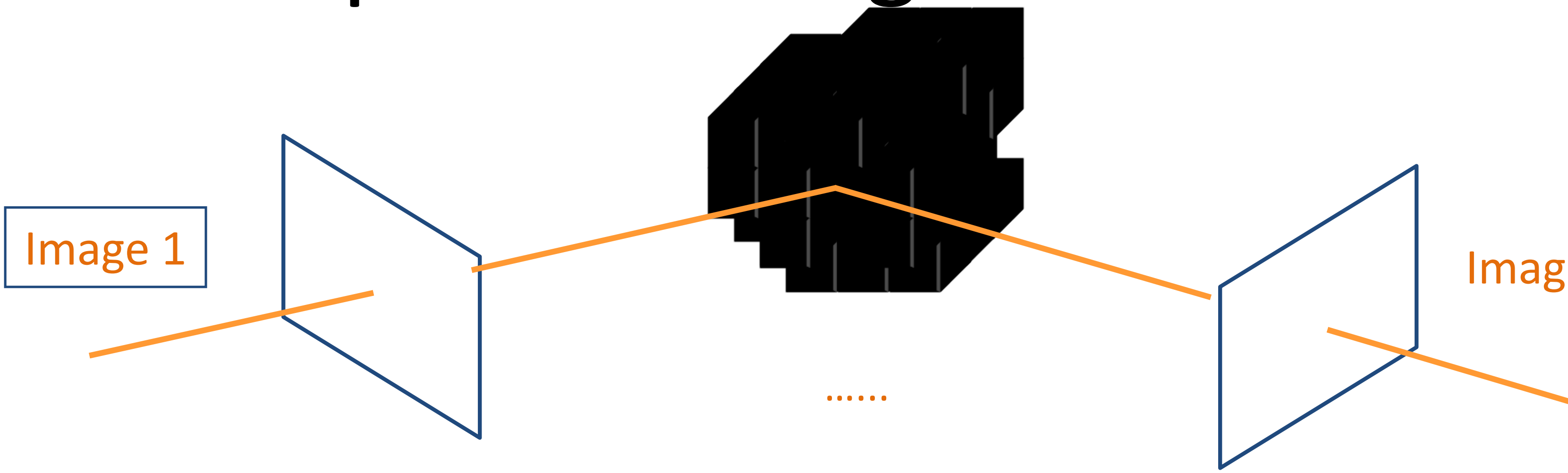
Calibrated Image Acquisition

Calibrated Turntable

Selected Dinosaur Images

Selected Flower Images

Space Carving in General

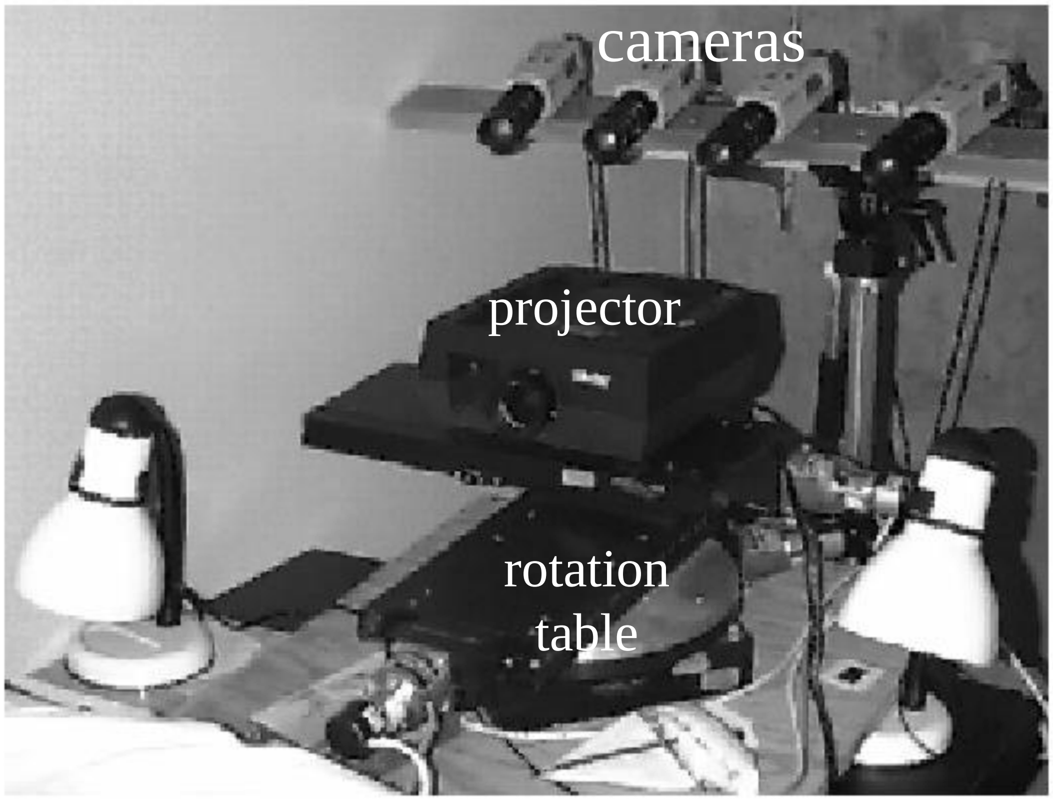


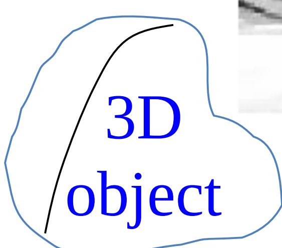
- Space Carving Algorithm
  - Initialize to a volume V containing the true scene
  - Choose a voxel on the outside of the volume
  - Project to visible input images
  - Carve if not photo-consistent (inside object’s silhouette)

K. N. Kutulakos and S. M. Seitz, **[A Theory of Shape by Space Carving](#)**, *ICCV* 1999

Our 4-camera light-striping stereo system

(now deceased)





Calibration Object

The idea is to snap images at different depths and get a lot of 2D-3D point correspondences.

Surface Modeling and Display from Range and Color Data

Kari	Pulli	UW
Michael	Cohen	MSR
Tom	Duchamp	UW
Hugues	Hoppe	MSR
John	McDonald	UW
Linda	Shapiro	UW
Werner	Stuetzle	UW

UW = University of Washington
Seattle, WA USA

MSR = Microsoft Research
Redmond, WA USA

Introduction

Goal

- develop robust algorithms for constructing 3D models from range & color data

- use those models to produce realistic renderings of the scanned objects

Surface Reconstuction

Step 1: Data acquisition

Obtain range data that covers the object. Filter, remove background.

Step 2: Registration

Register the range maps into a common coordinate system.

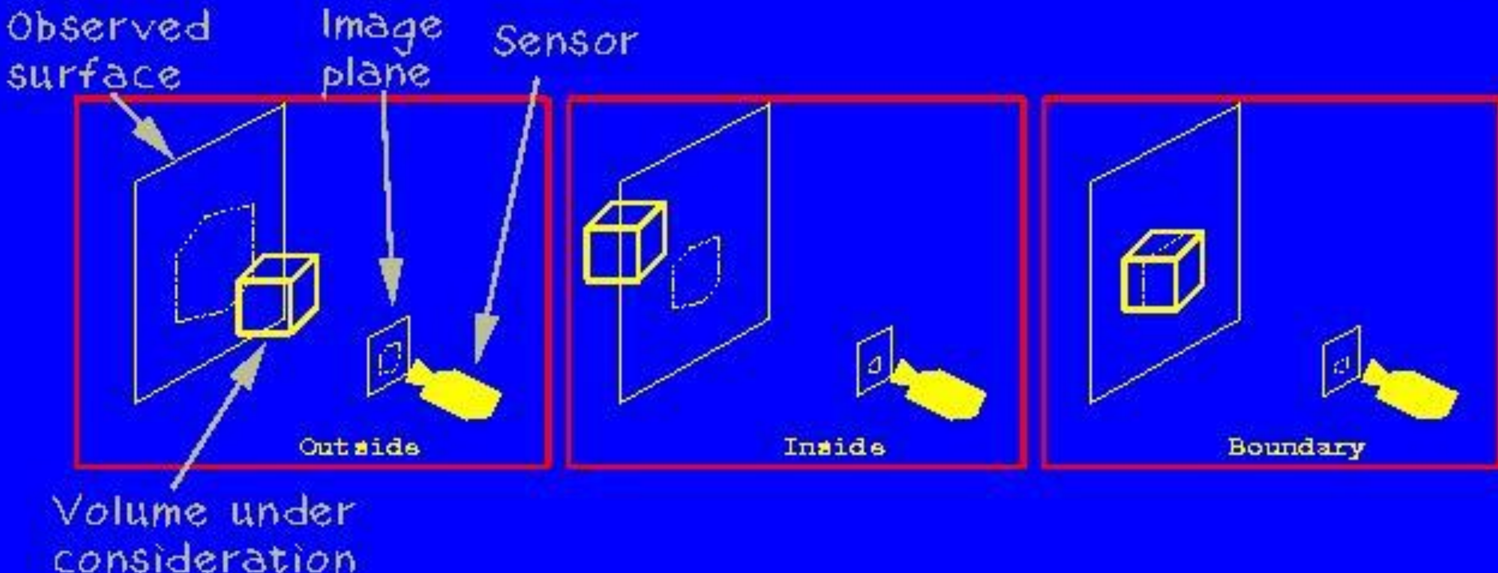
Step 3: Integration

Integrate the registered range data into a single surface representation.

Step 4: Optimization

Fit the surface more accurately to the data, simplify the representation.

Carve space in cubes



Label cubes

- Project cube to image plane (hexagon)
- Test against data in the hexagon

3D space is made up of many cubes.

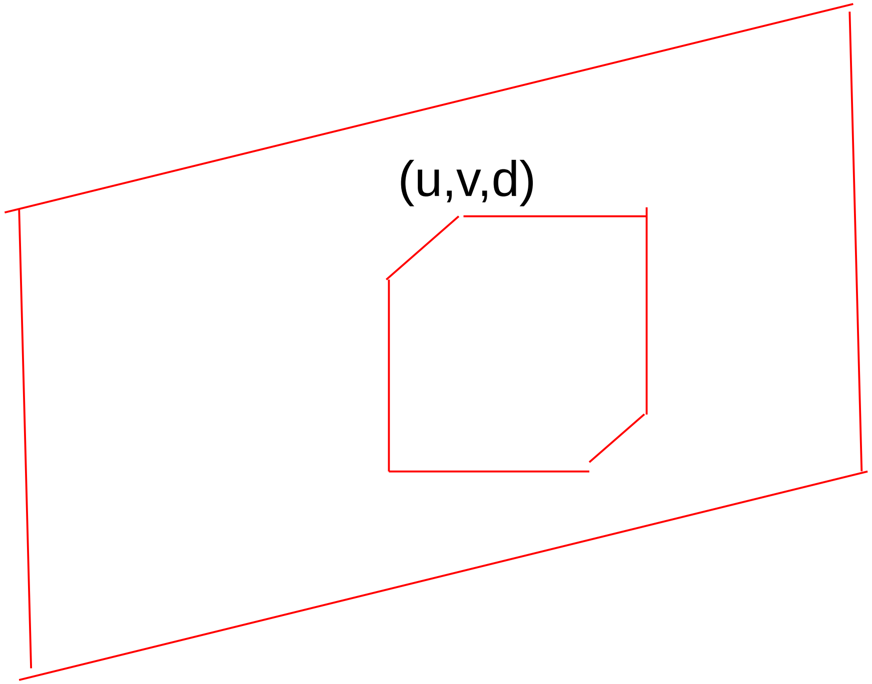
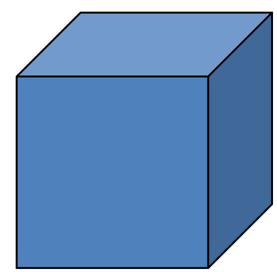


image plane
depth map

(

(x,y,z)



one of many cubes
in virtual 3D cube space

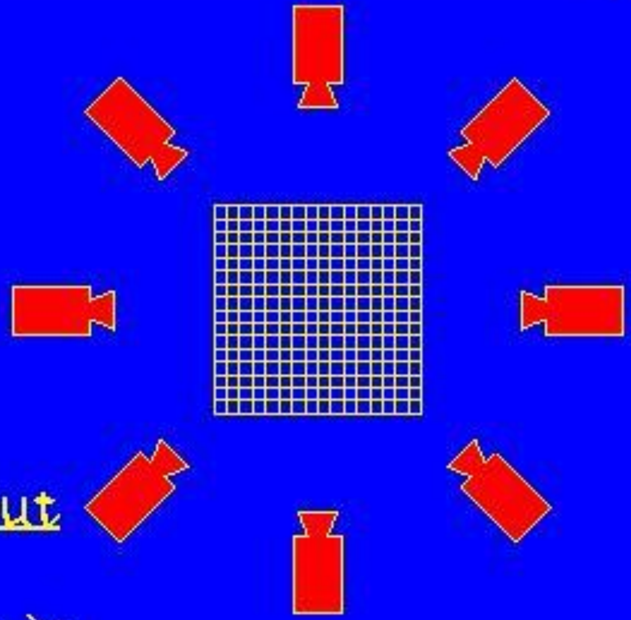
OUTSIDE

Several views

Processing order:

FOR EACH cube

FOR EACH view



Rules:

<u>any</u> view thinks cube's <u>out</u>

=> it's <u>out</u>

<u>every</u> view thinks cube's <u>in</u>

=> it's <u>in</u>

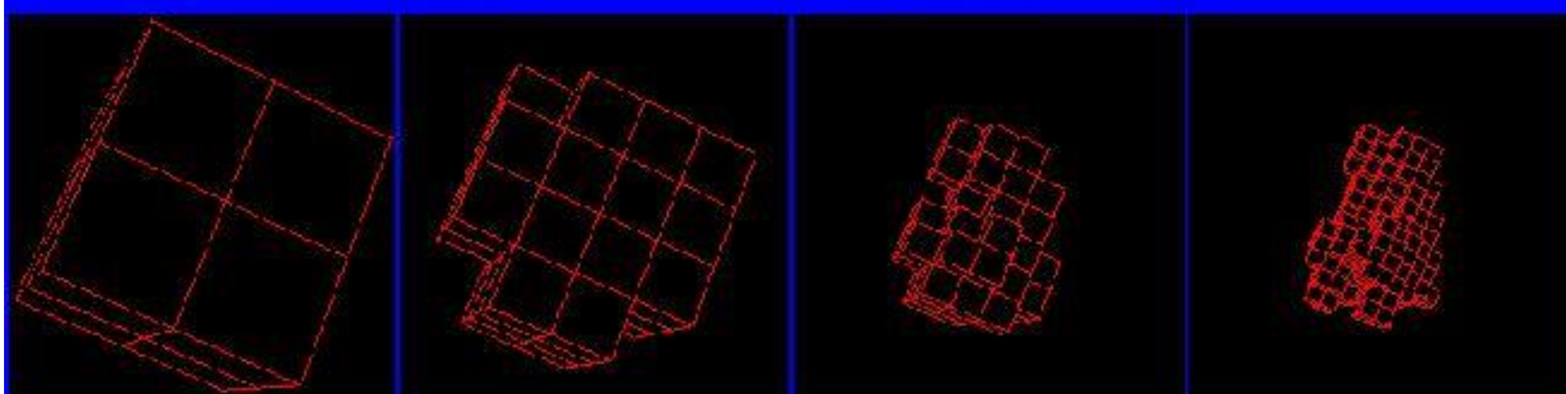
else

=> it's at boundary

Hierarchical space carving

- Big cubes => fast, poor results
- Small cubes => slow, more accurate results
- Combination = octrees

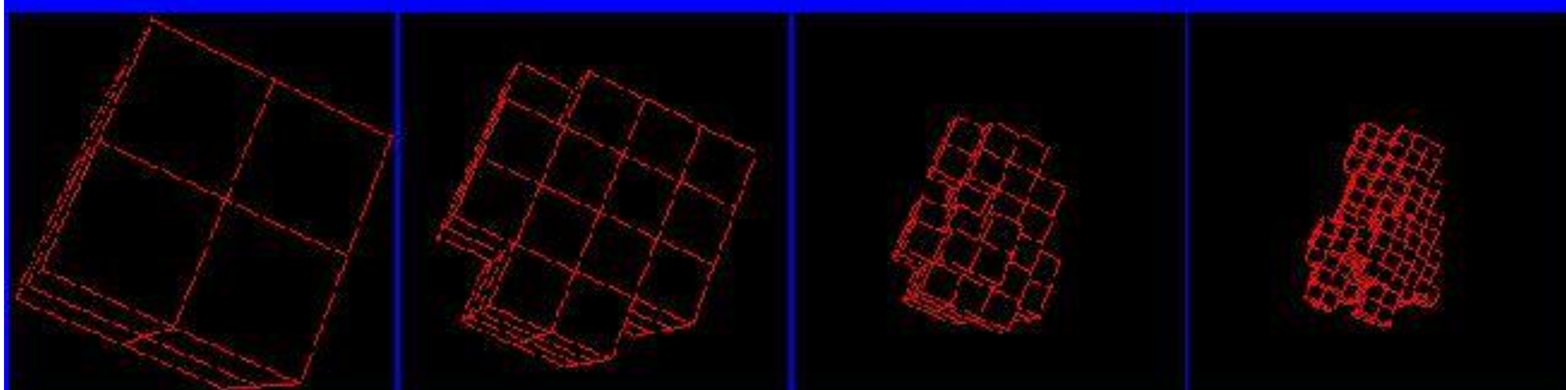
RULES:
- cube's out => done
- cube's in => done
- else => recurse



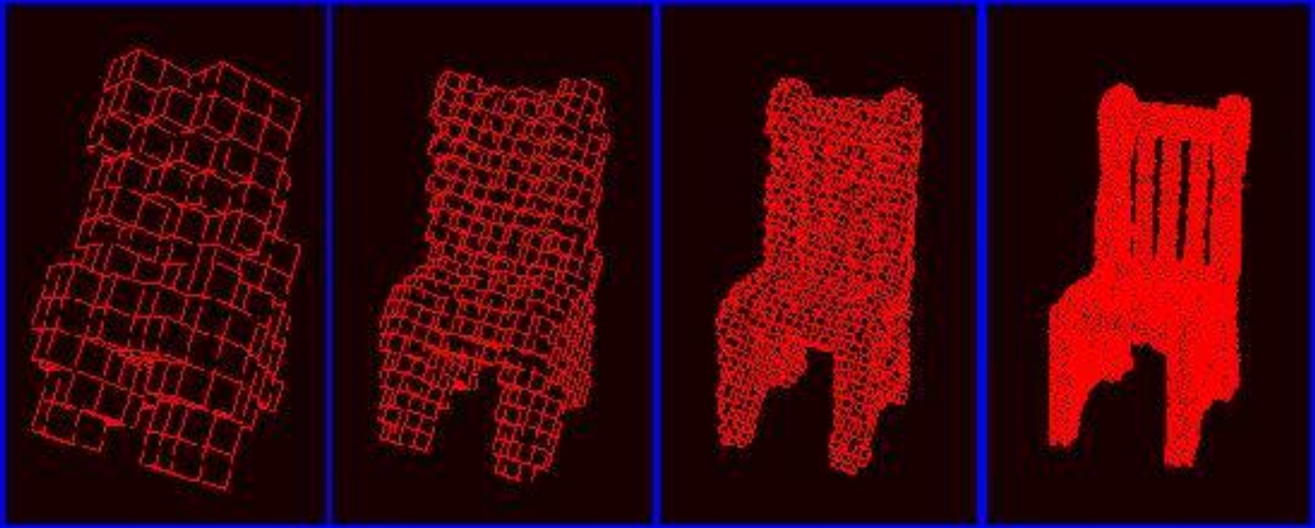
Hierarchical space carving

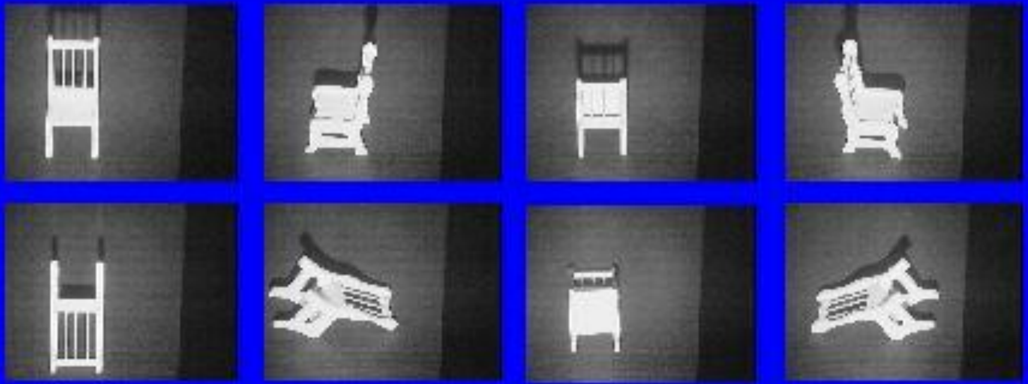
- Big cubes => fast, poor results
- Small cubes => slow, more accurate results
- Combination = octrees

RULES:
- cube's out => done
- cube's in => done
- else => recurse

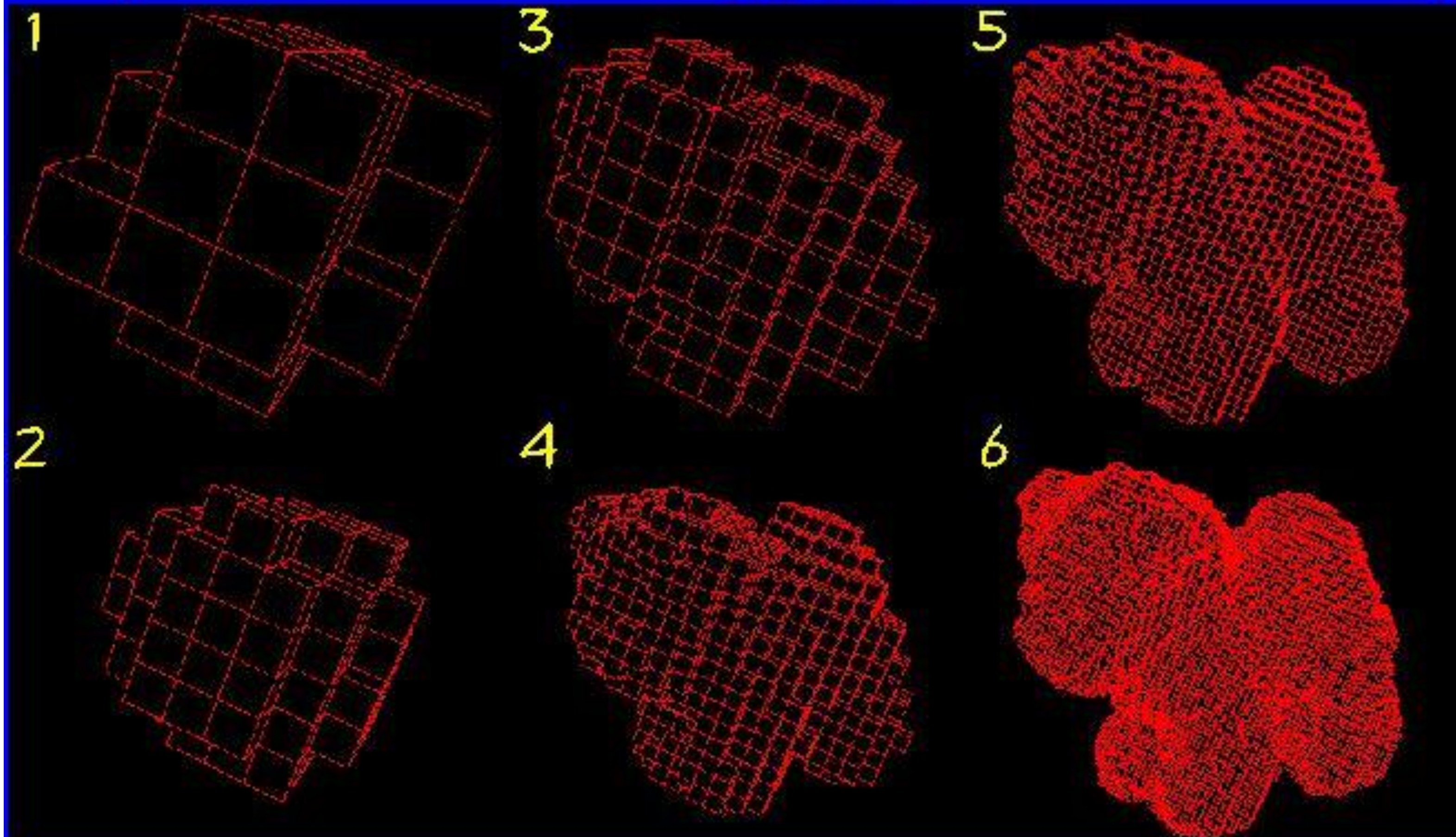


The rest of the chair

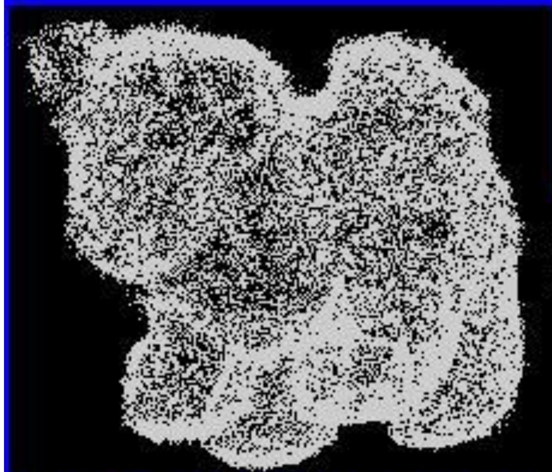




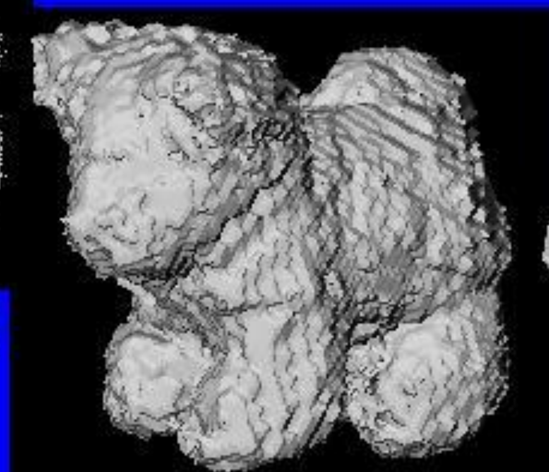
Same for a husky pup



Optimizing the dog mesh



Registered points

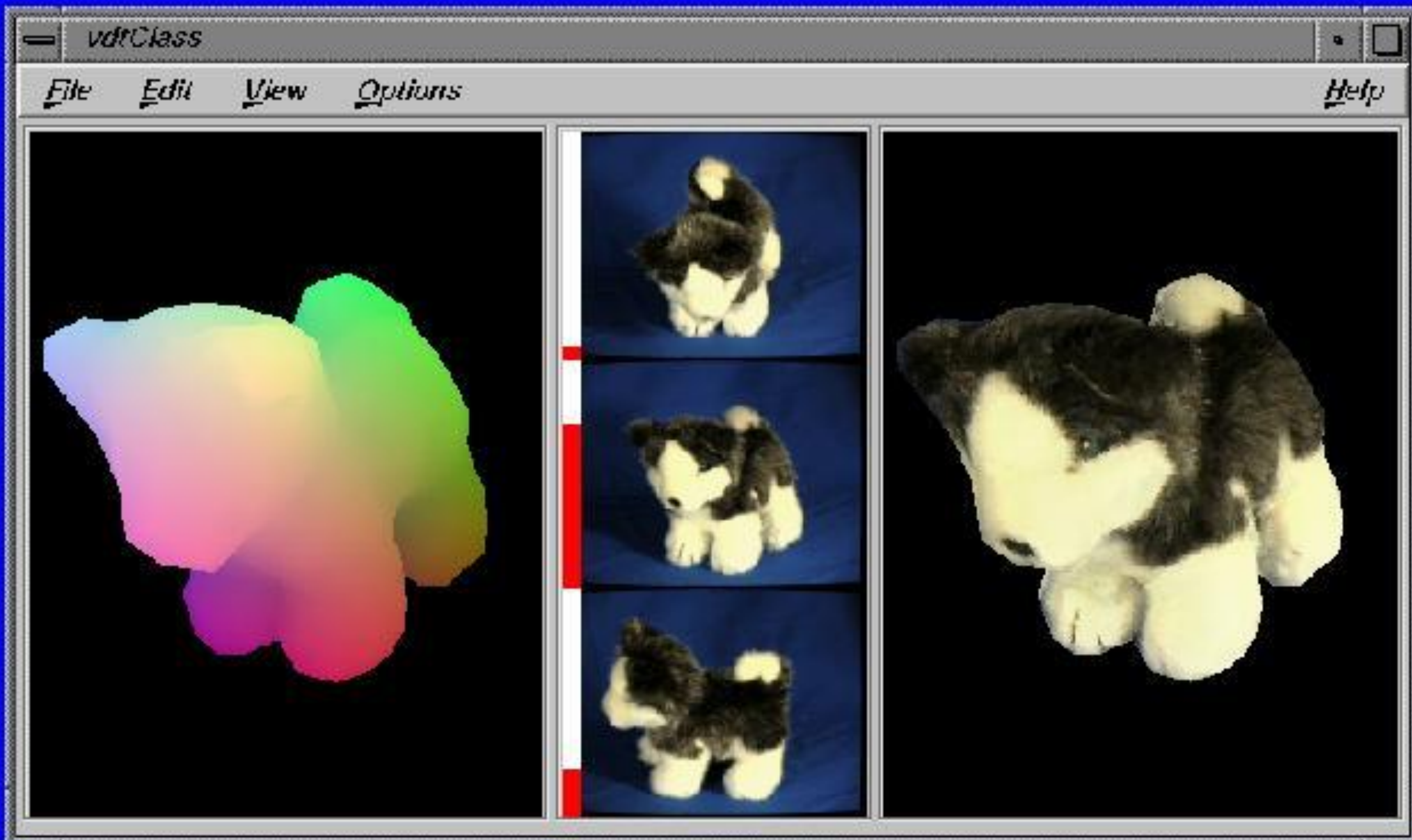


Initial mesh

Optimized mesh

View dependent texturing

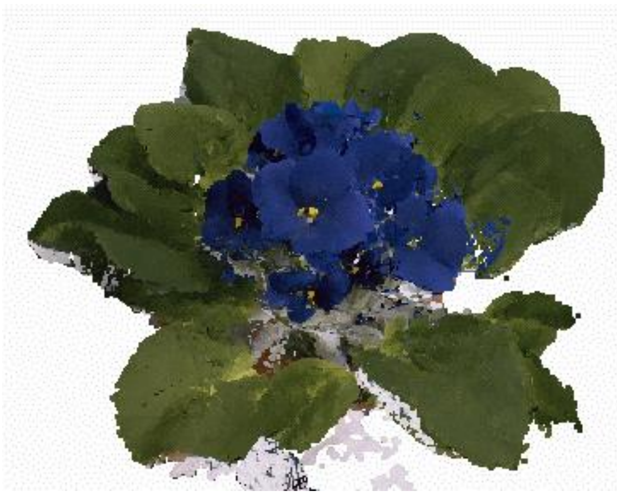
Our viewer



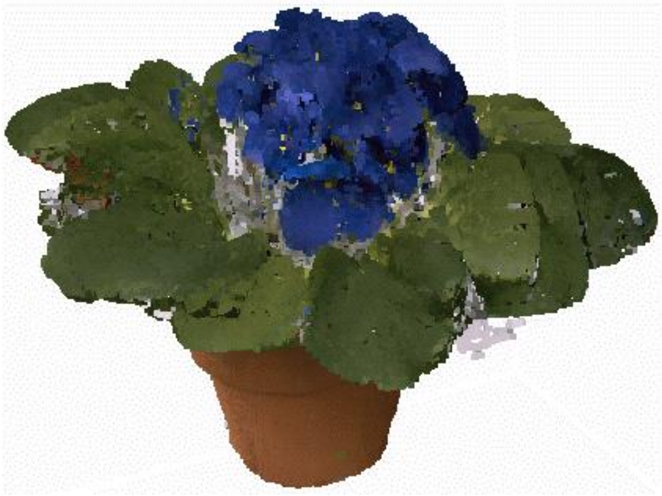
More: Space Carving Results: African Violet

Input Image (1 of 45)

Reconstruction



Reconstruction



Reconstruction

Source: S. Seitz

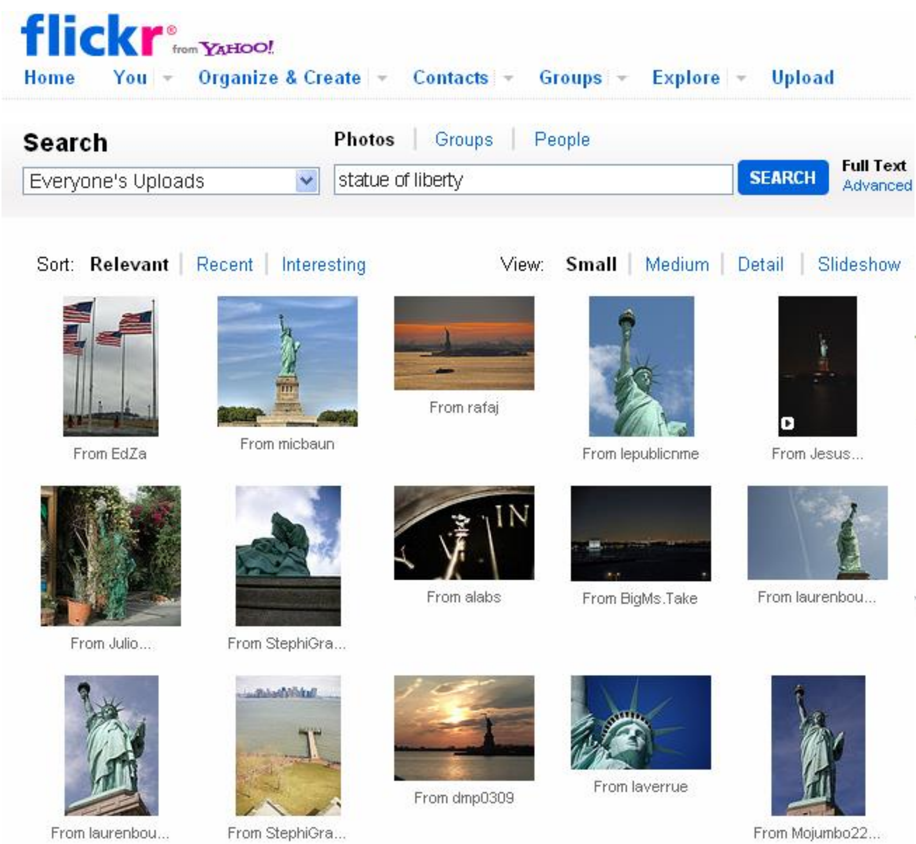
More: Space Carving Results: Hand

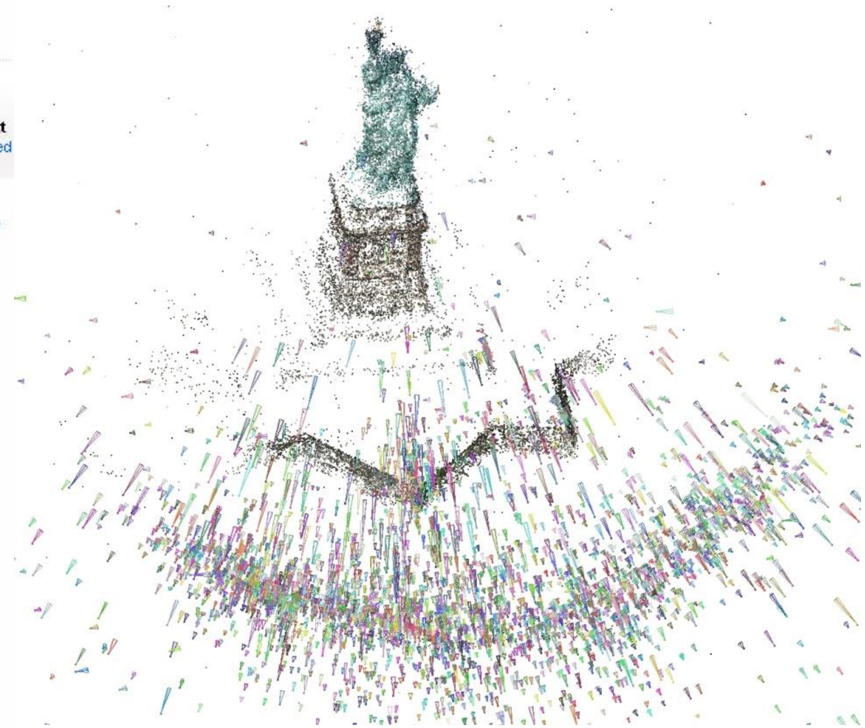
Input Image
(1 of 100)

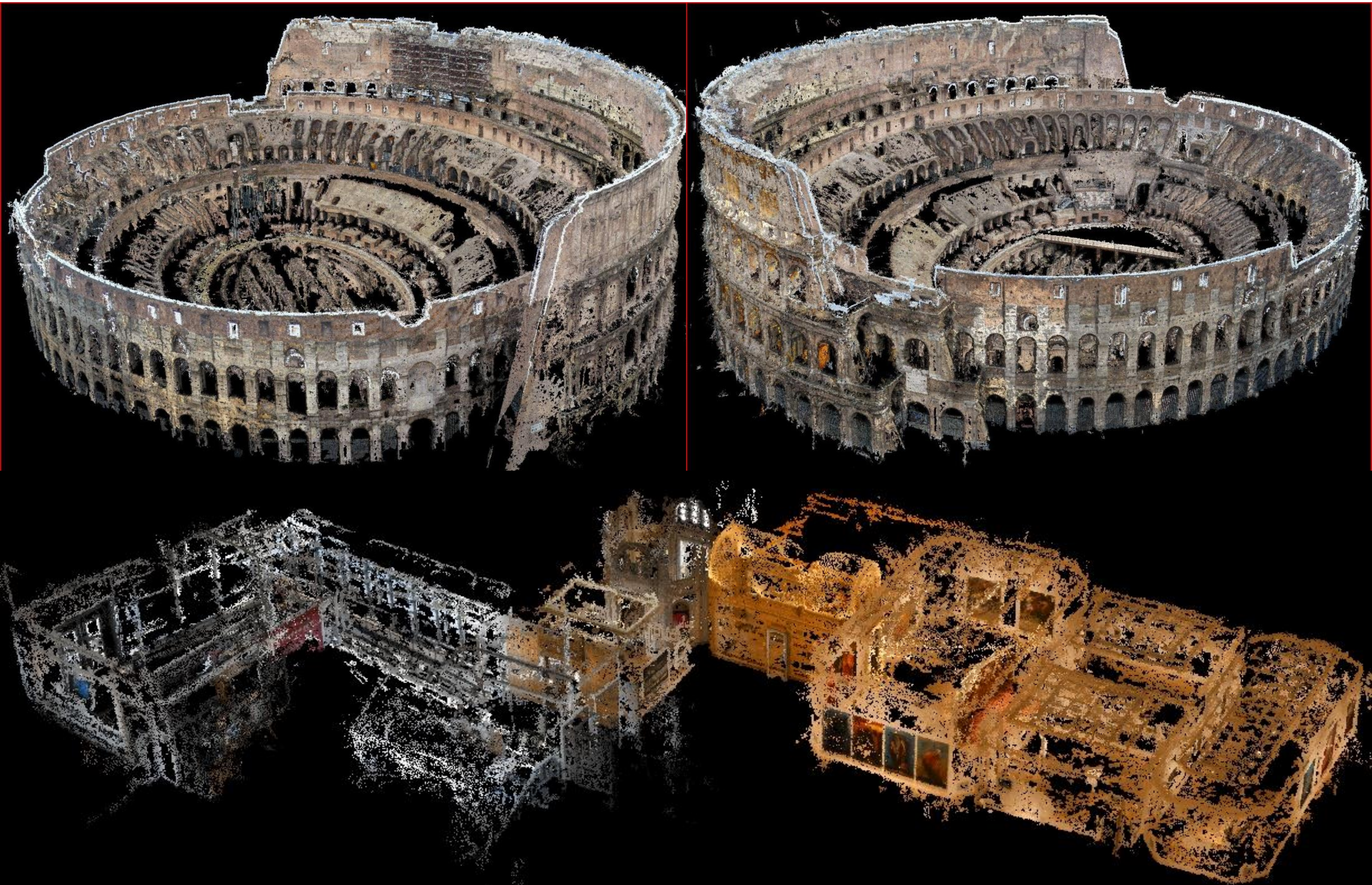
Views of Reconstruction

Stereo from community photo collections

- Up to now, we've always assumed that camera calibration is known
- For photos taken from the Internet, we need *structure from motion* techniques to reconstruct both camera positions and 3D points.







Head Reconstruction from Uncalibrated Internet Photos

- Input: Internet photos in different poses and expressions

- Output: 3D model of the head

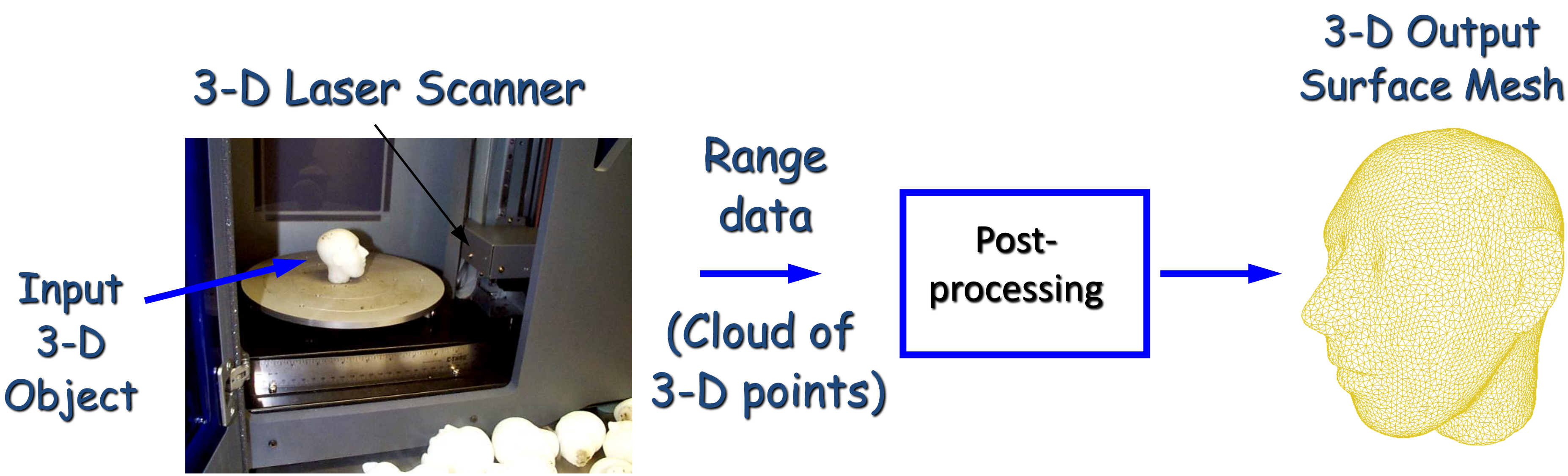
work of
Shu Liang

Recognizing Deformable Shapes

Salvador Ruiz Correa
(CSE/EE576 Computer Vision I)

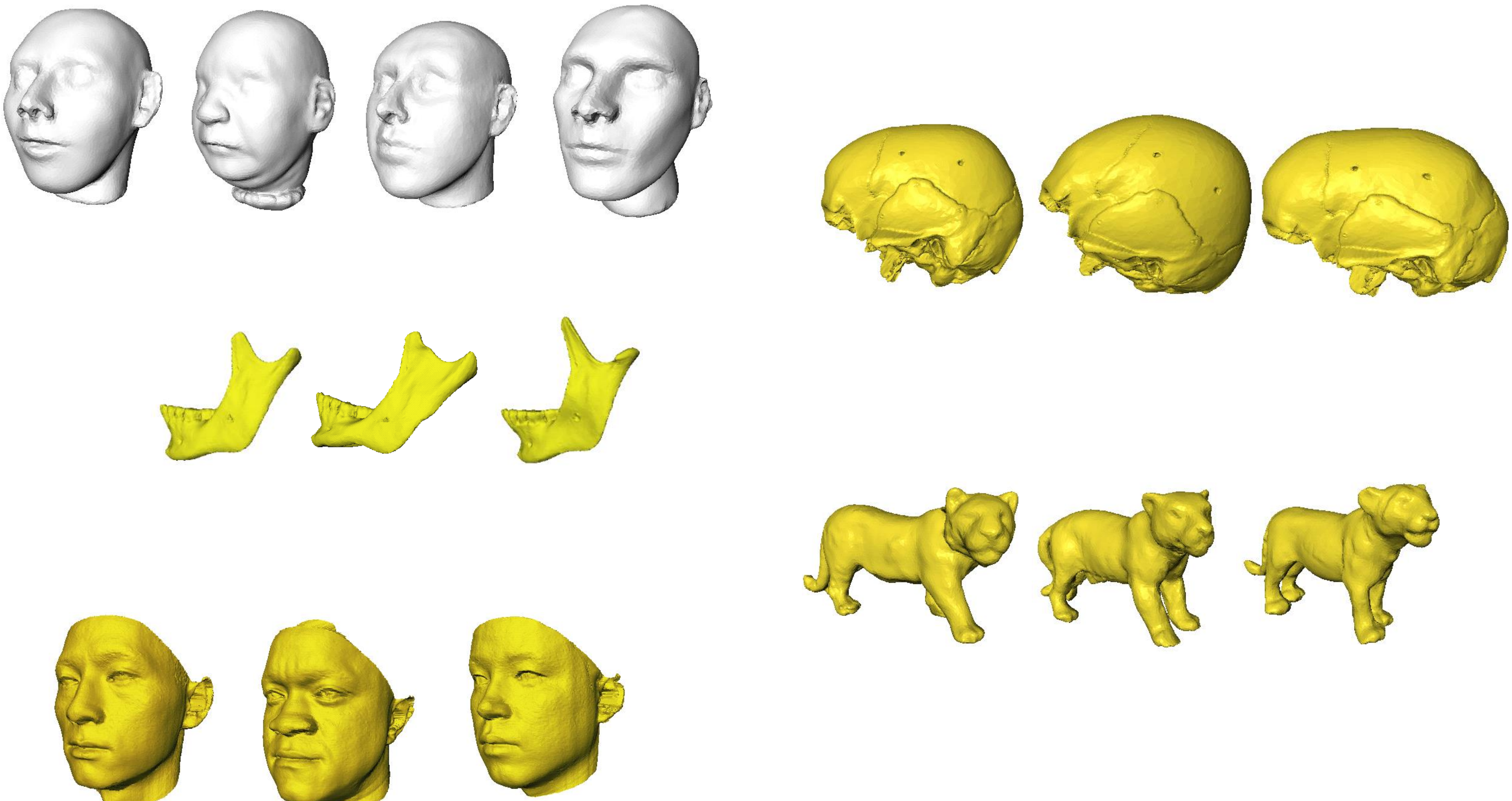
Goal

- We are interested in developing algorithms for recognizing and classifying deformable object shapes from range data.

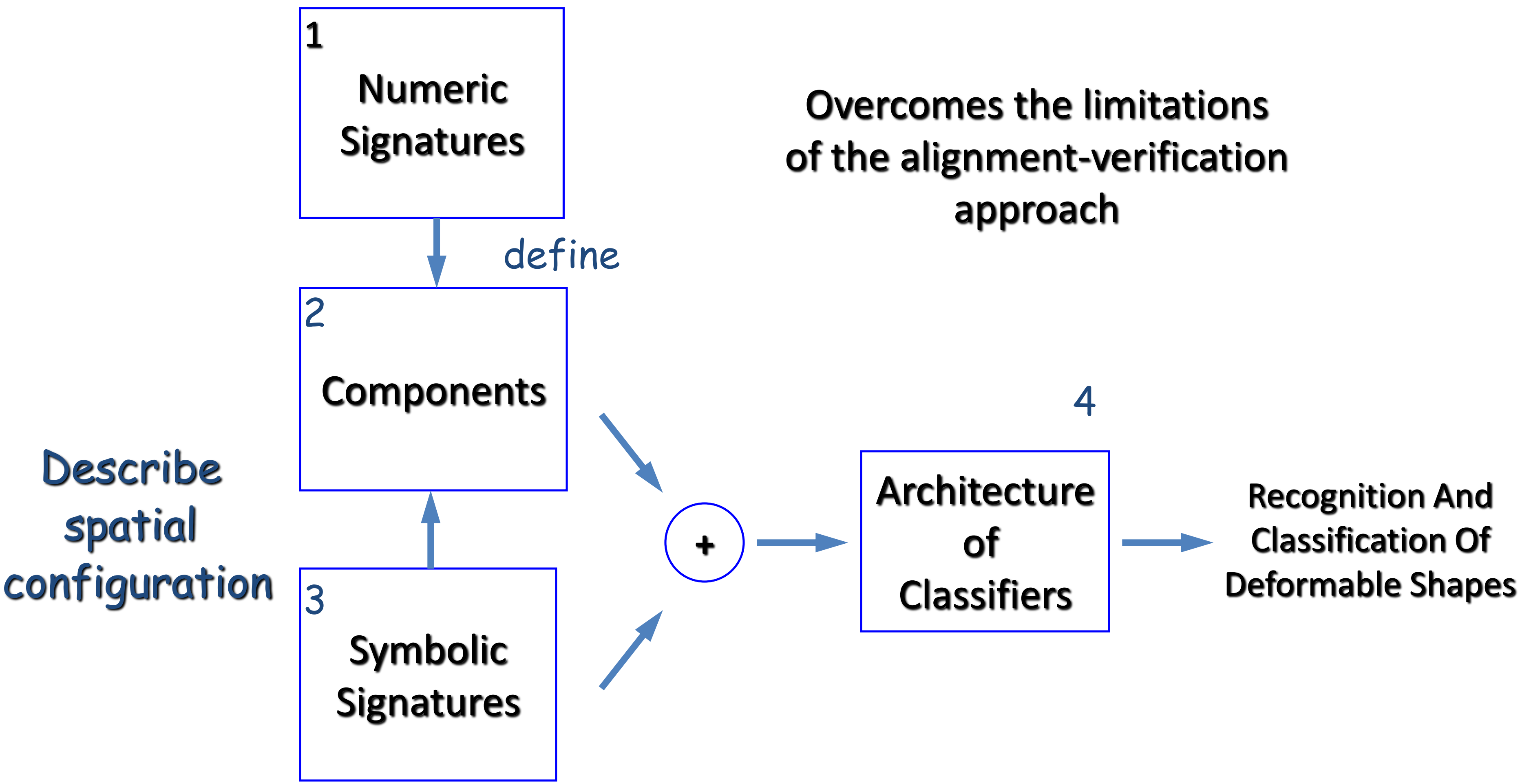


- This is a difficult problem that is relevant in several application fields.

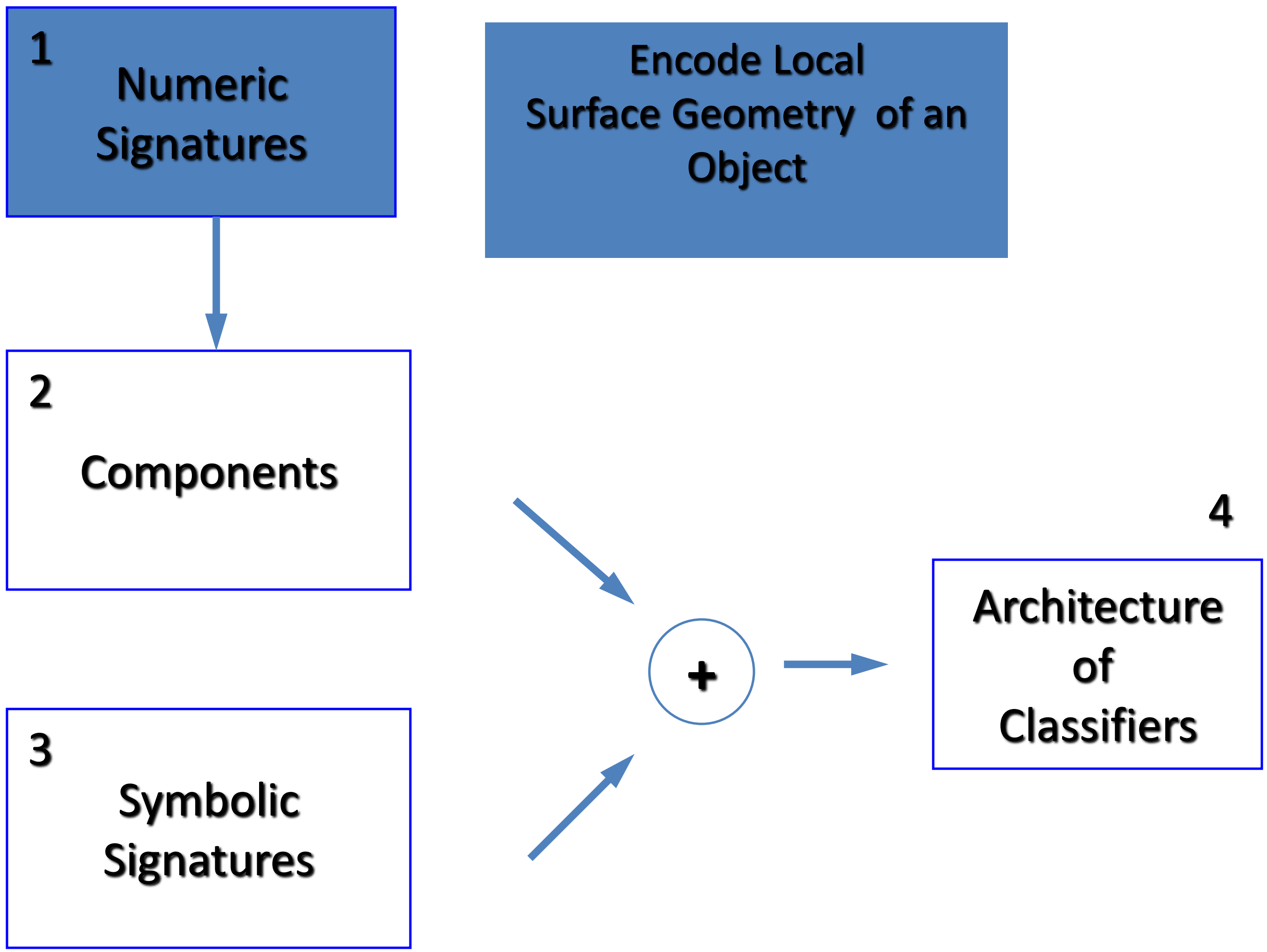
What Kind Of Deformations?



Component-Based Methodology



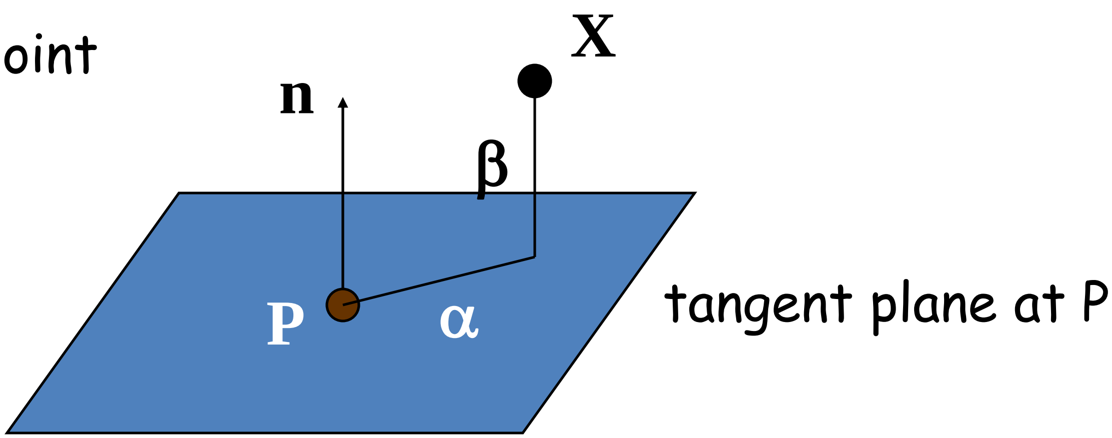
Numeric Signatures



The Spin Image Signature

P is the selected vertex.

X is a contributing point
of the mesh.



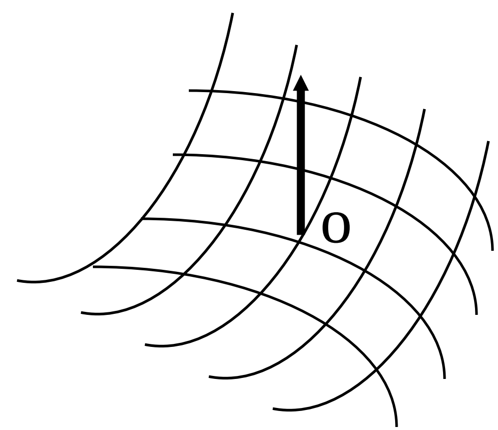
α is the perpendicular distance from X to P's surface normal.

β is the signed perpendicular distance from X to P's tangent plane.

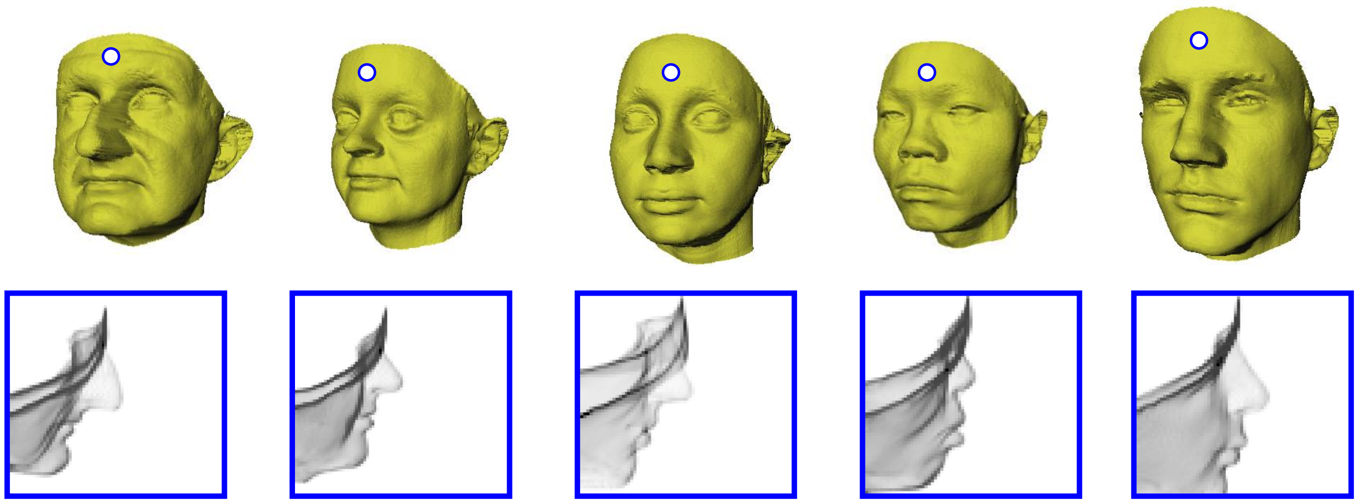
Spin Image Construction

- A spin image is constructed
  - about a specified oriented point o of the object surface
  - with respect to a set of contributing points C, which is controlled by maximum distance and angle from o.
- It is stored as an array of accumulators $S(\alpha,\beta)$ computed via:
- For each point c in C(o)

  1. compute α and β for c.
  2. increment S (α,β)

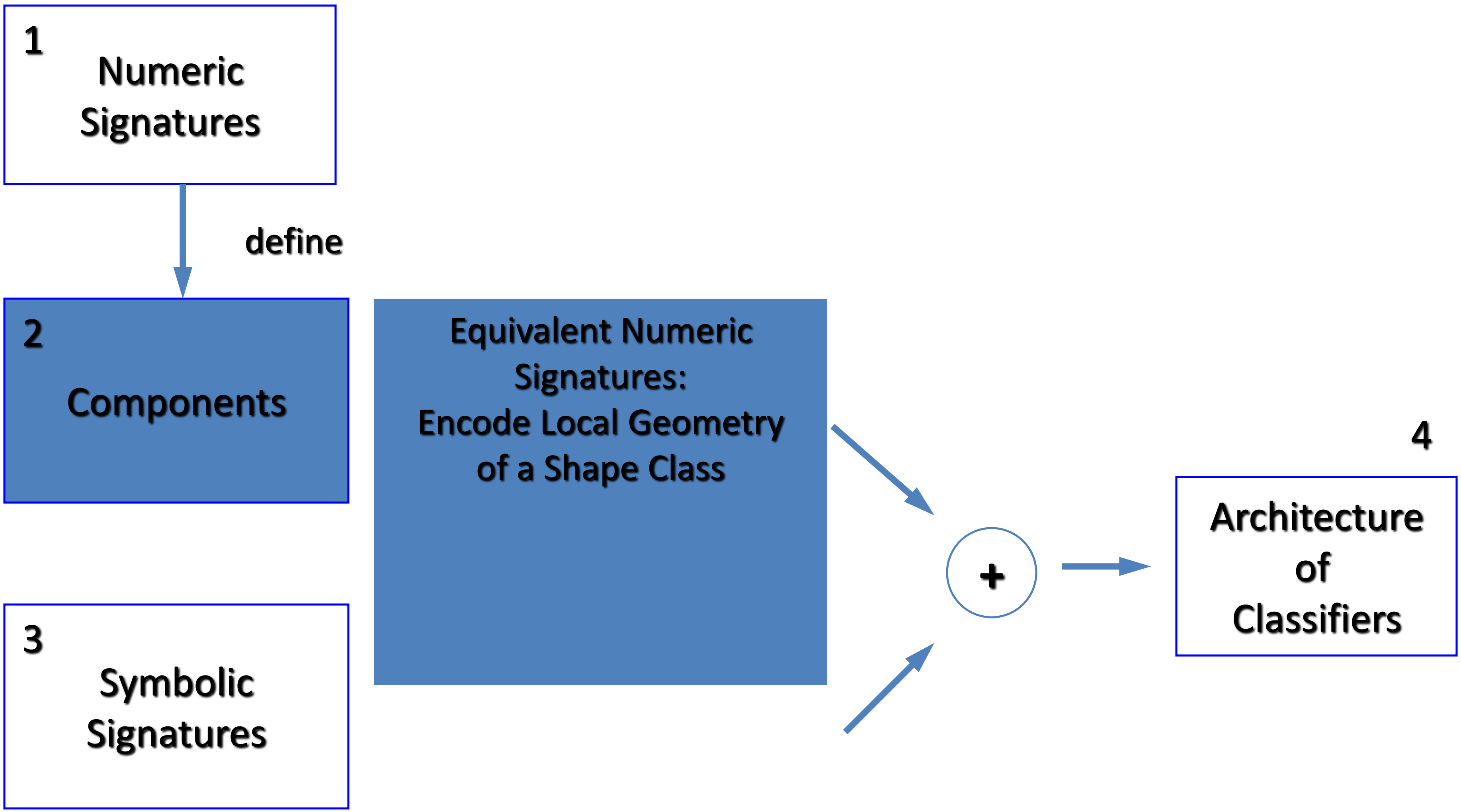


Numeric Signatures: Spin Images

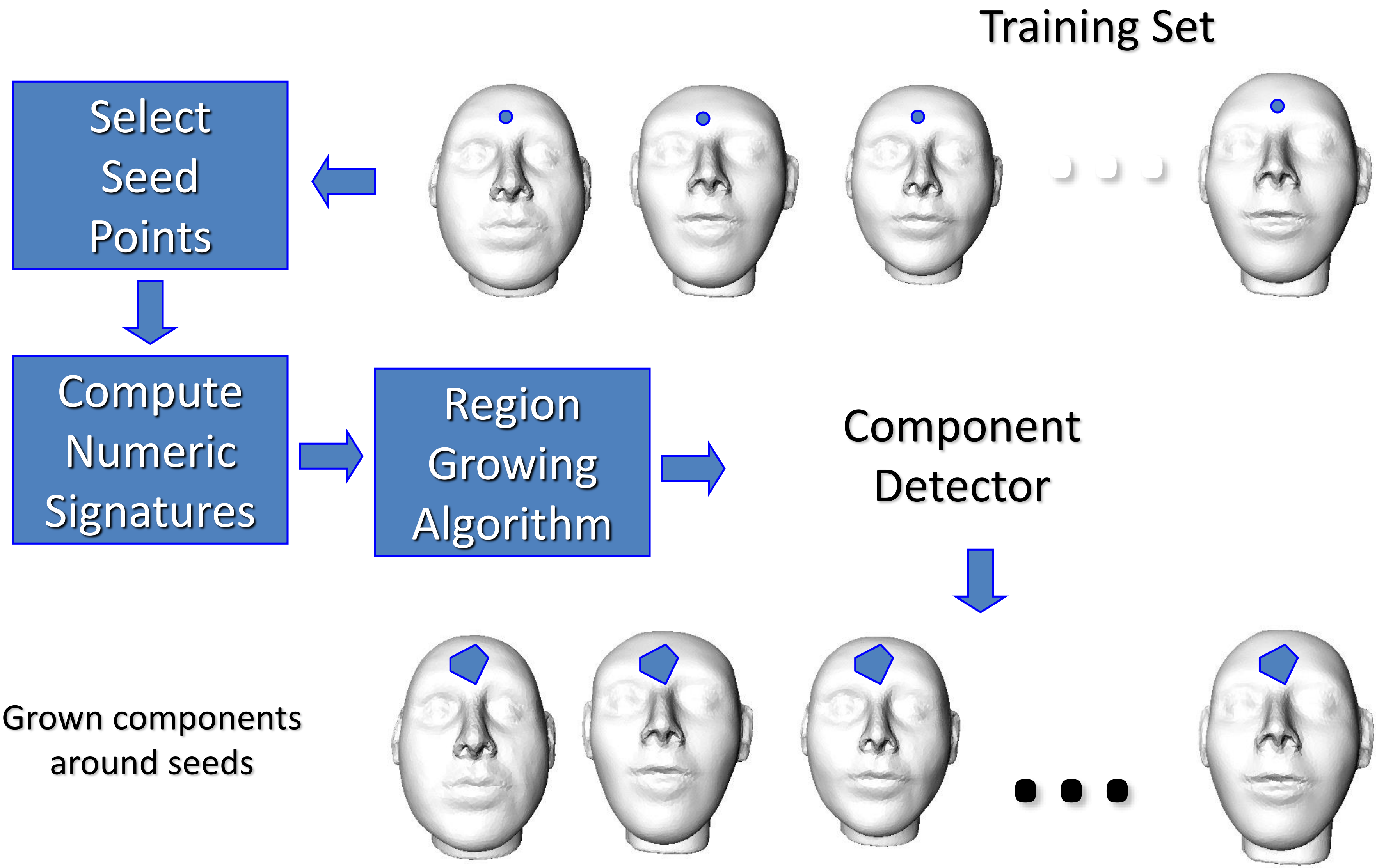


- Rich set of surface shape descriptors.
- Their spatial scale can be modified to include local and non-local surface features.
- Representation is robust to scene clutter and occlusions.

Components

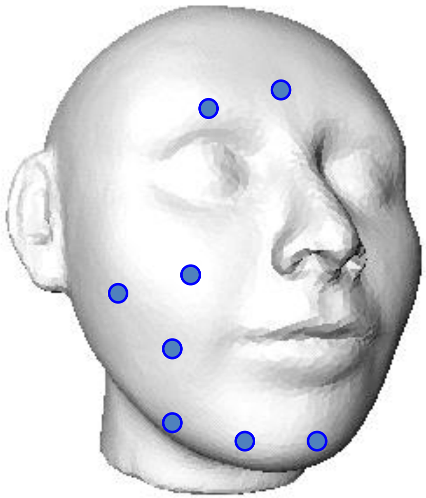

1
Numeric Signatures
define
2
Components
Equivalent Numeric Signatures:
Encode Local Geometry of a Shape Class
3
Symbolic Signatures
+
4
Architecture of Classifiers

How To Extract Shape Class Components?



Component Extraction Example

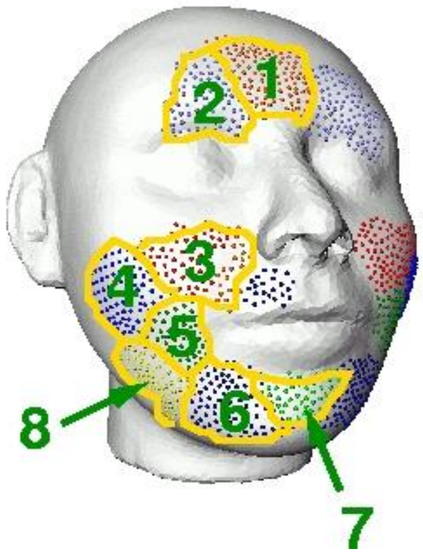
Selected 8 seed points by hand



Grow one region at the time
(get one detector
per component)

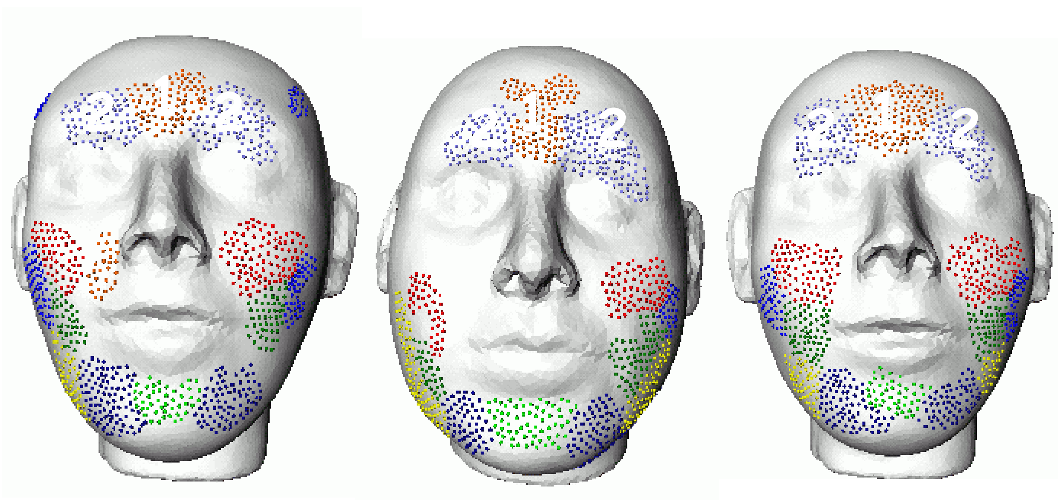
Region
Growing

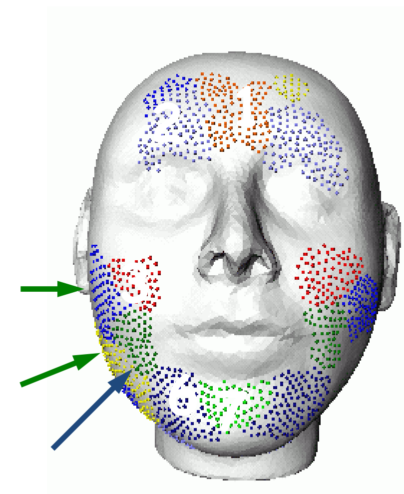
Labeled
Surface Mesh



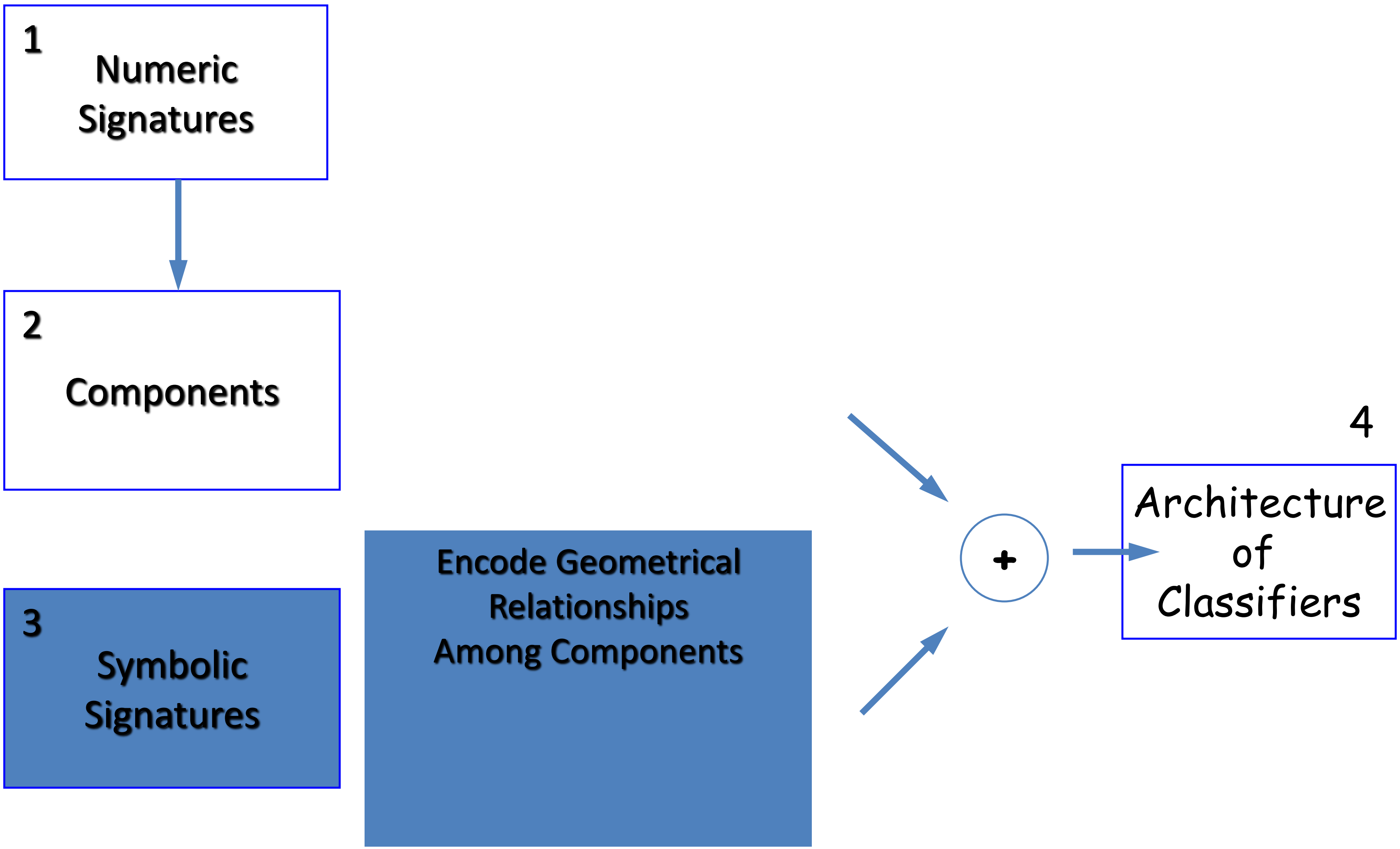
Detected
components on a
training sample

How To Combine Component Information?





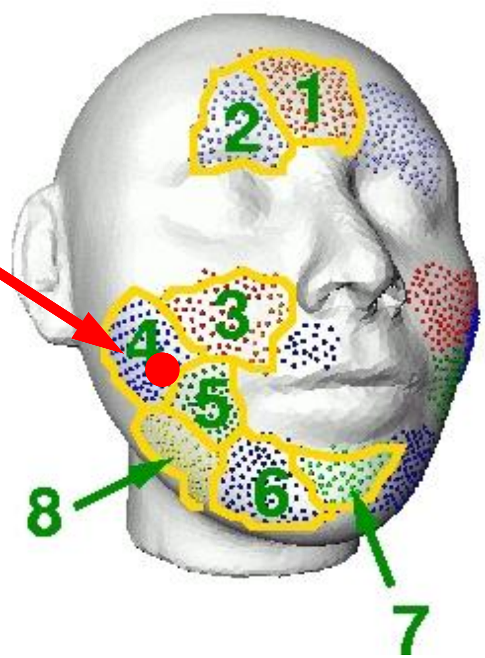
Symbolic Signatures



Symbolic Signature

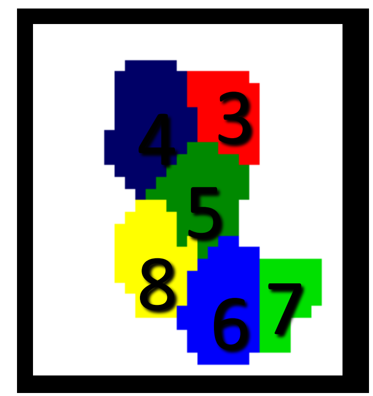
Labeled
Surface Mesh

Critical
Point P



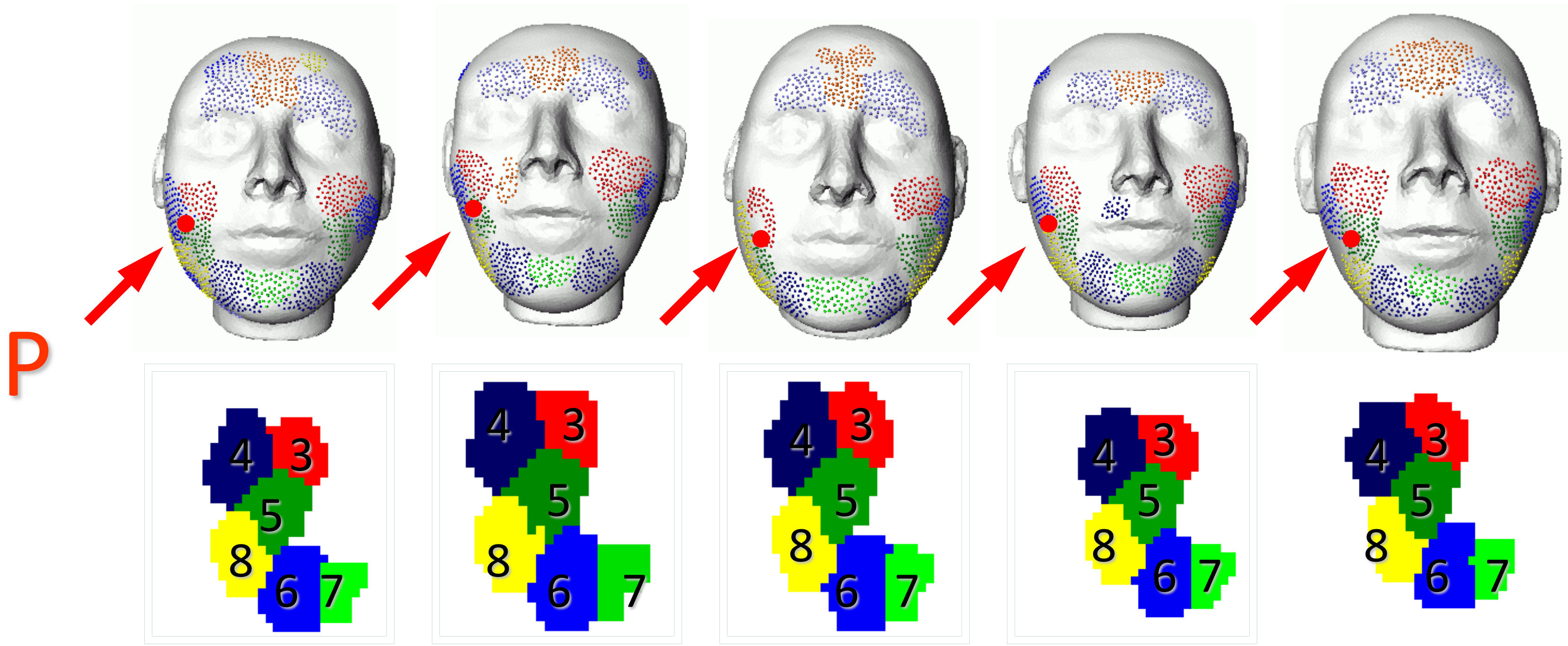
Encode
Geometric
Configuration

Symbolic
Signature at P



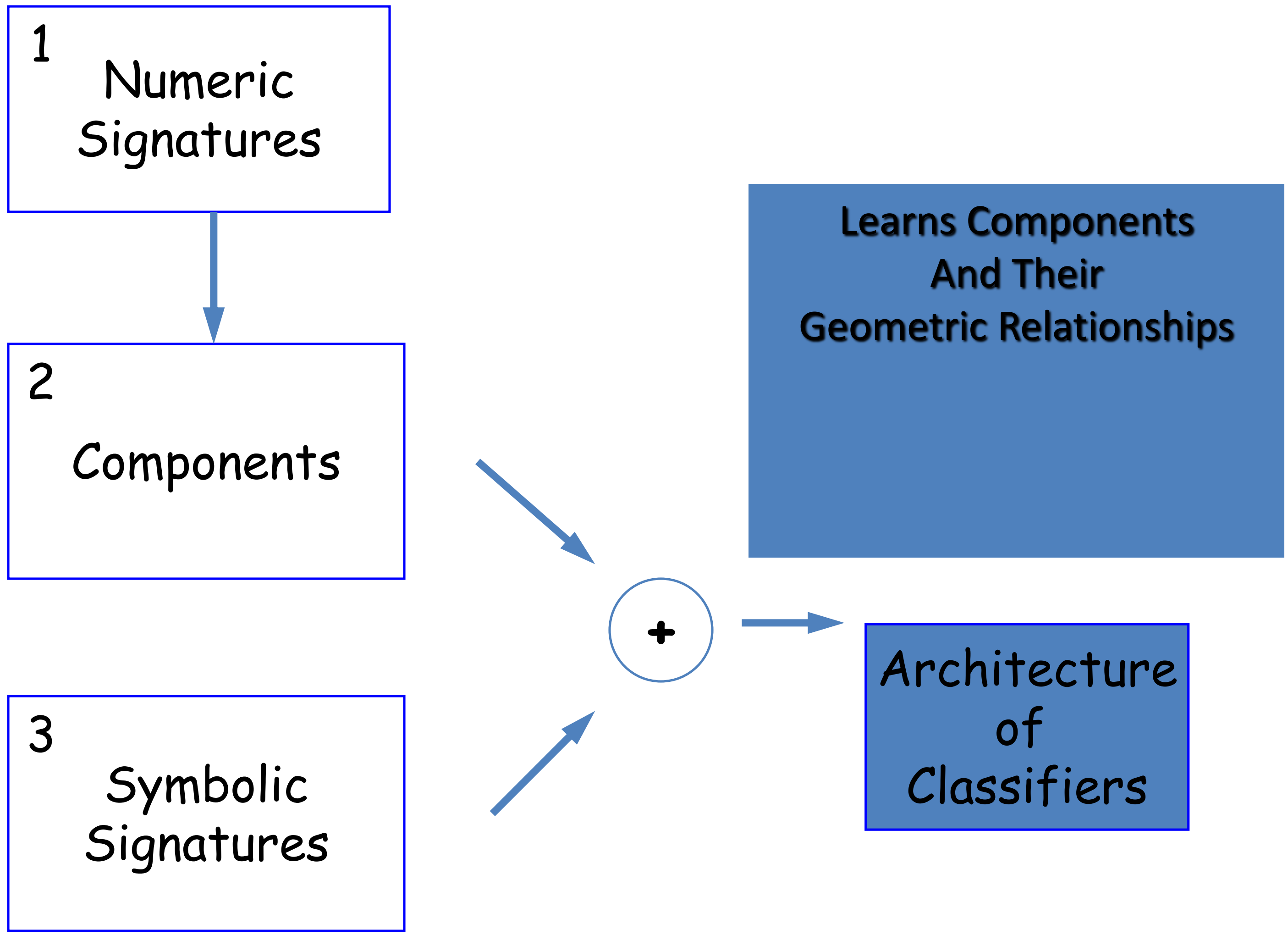
Matrix storing
component
labels

Symbolic Signatures Are Robust To Deformations



Relative position of components is stable across deformations: experimental evidence

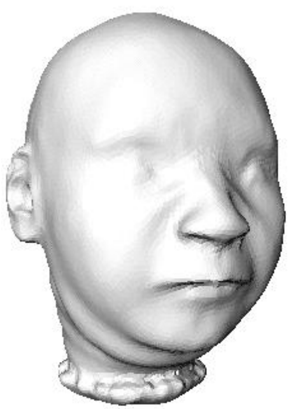
Architecture of Classifiers



Proposed Architecture

Verify spatial configuration of the components

Input → Identify Components → Labeled Mesh → Identify Symbolic Signatures → Class Label



Surface Mesh

-1

(Abnormal)

Two classification stages

Experimental Validation

Recognition Tasks: 4 (T1 - T4)

Classification Tasks: 3 (T5 – T7)

No. Experiments: 5470

Rotary Table

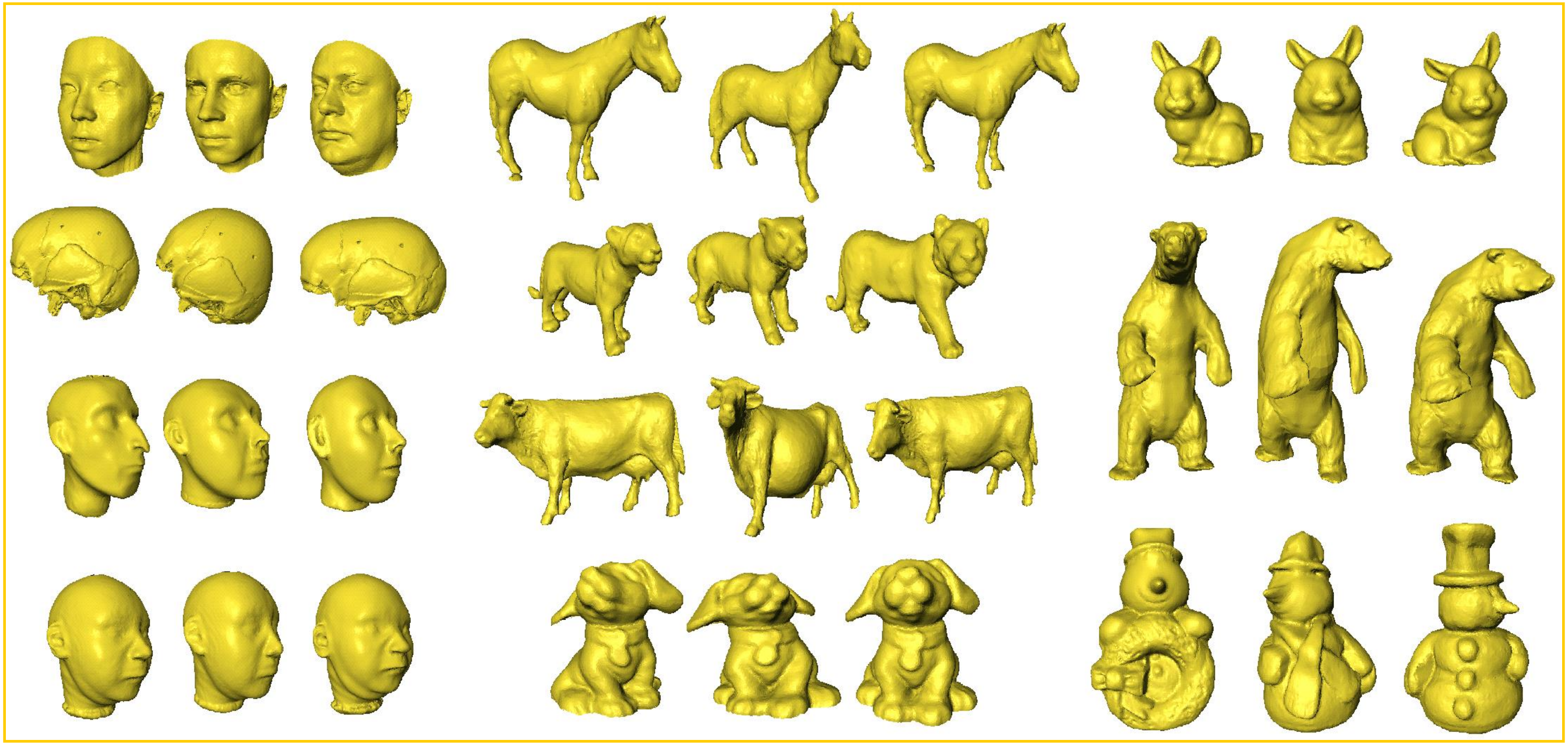
Setup

Laser

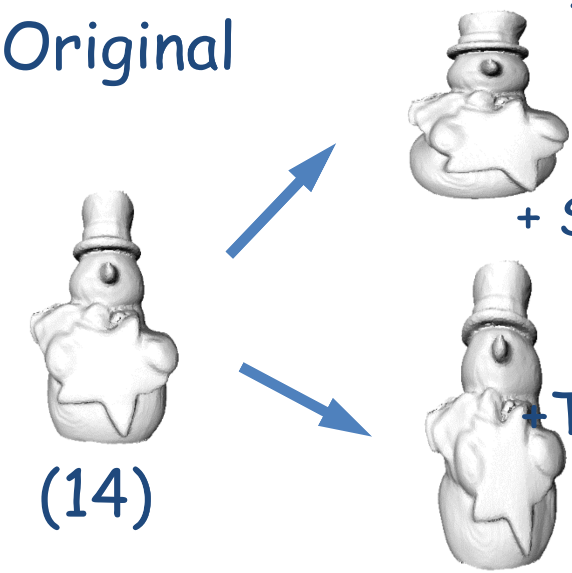
Recognition

Classification

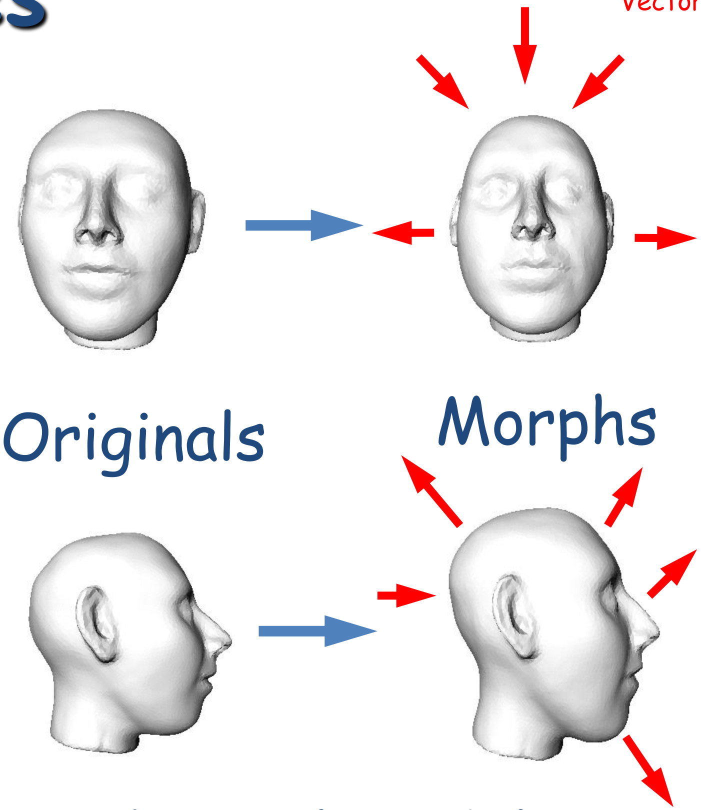
Shape Classes



Enlarging Training Sets Using Virtual Samples



Global Morphing Operators



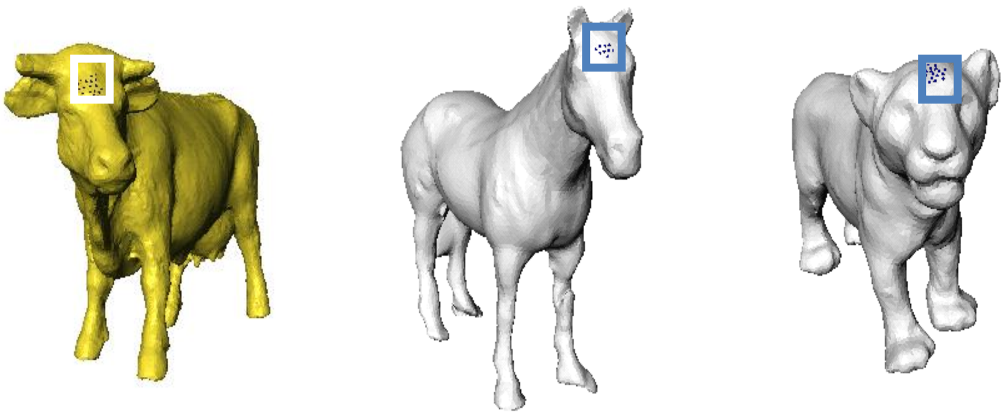
Physical Modeling

Task 1: Recognizing Single Objects (1)

- No. Shape classes: 9.
- Training set size: 400 meshes.
- Testing set size: 200 meshes.
- No. Experiments: 1960.
- No. Component detectors:3.
- No. Symbolic signature detectors: 1.
- Numeric signature size: 40x40.
- Symbolic signature size: 20x20.
- No clutter and occlusion.

Task 1: Recognizing Single Objects (2)

- Snowman: 93%.
- Rabbit: 92%.
- Dog: 89%.
- Cat: 85.5%.
- Cow: 92%.
- Bear: 94%.
- Horse: 92.7%.
- Human head: 97.7%.
- Human face: 76%.



Recognition rates (true positives)

(No clutter, no occlusion, complete models)

Main Contributions (2)

- A region growing algorithm for learning shape class components.
- A novel architecture of classifiers for abstracting the geometry of a shape class.
- A validation of our methodology in a set of large scale recognition and classification experiments aimed at applications in scene analysis and medical diagnosis.