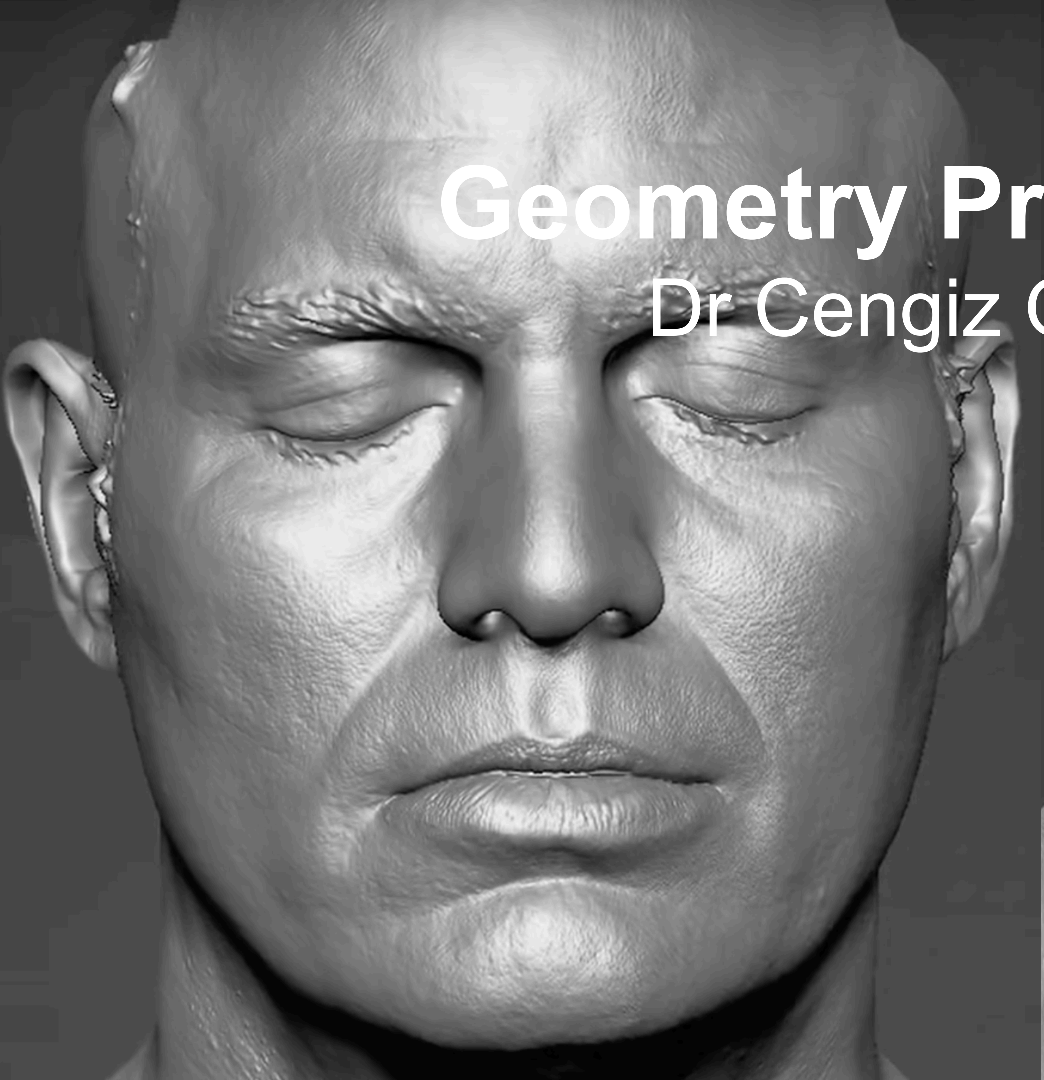
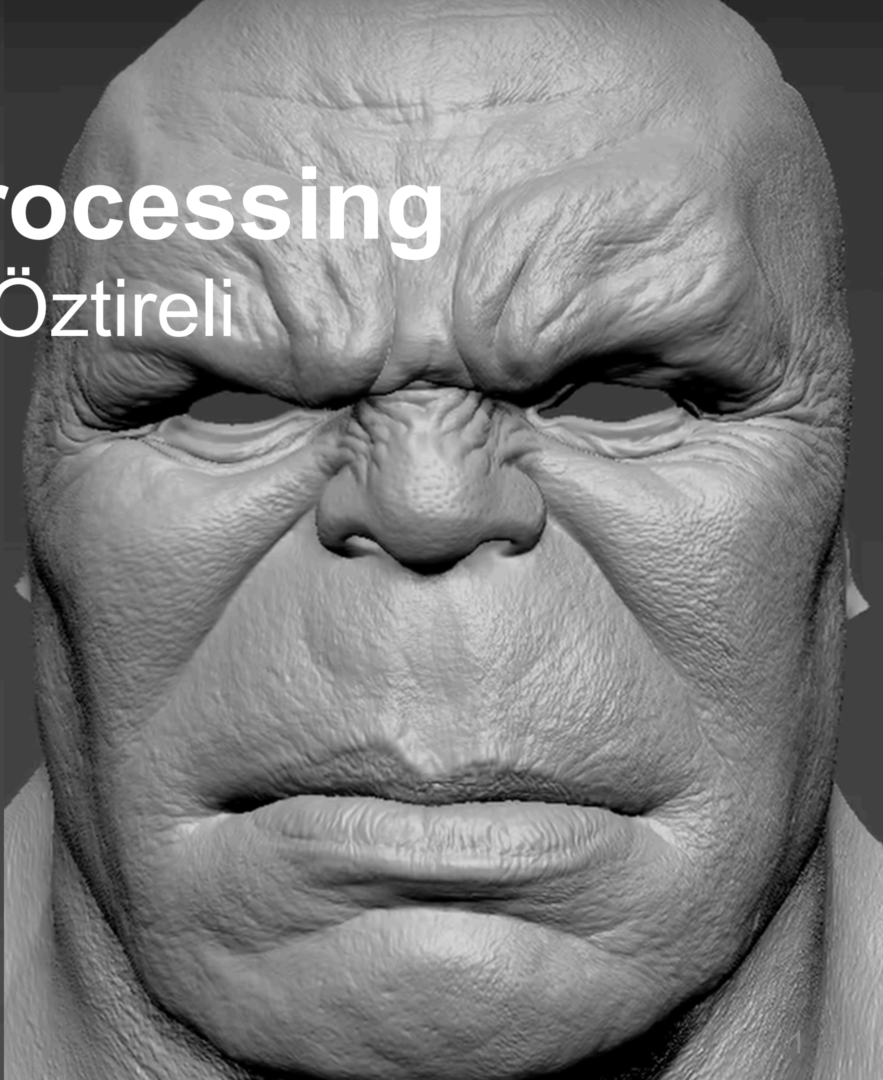
Geometry Processing

Dr Cengiz Öztireli

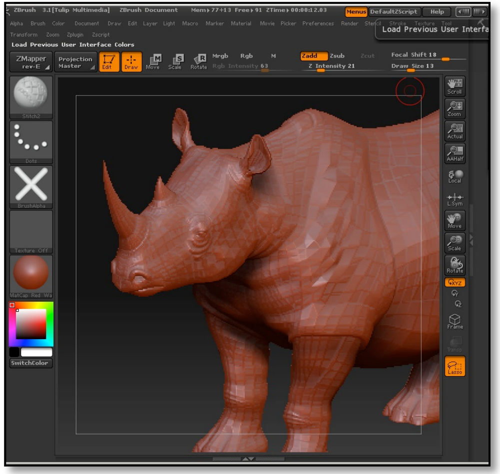


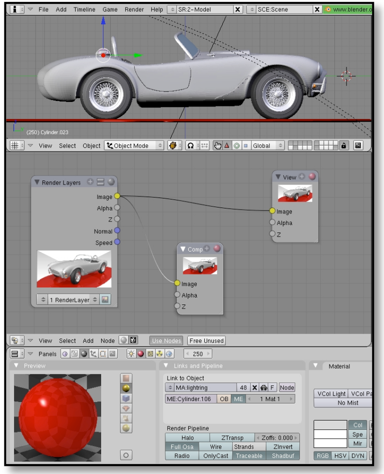


Sources of Geometry

Acquisition from the real world

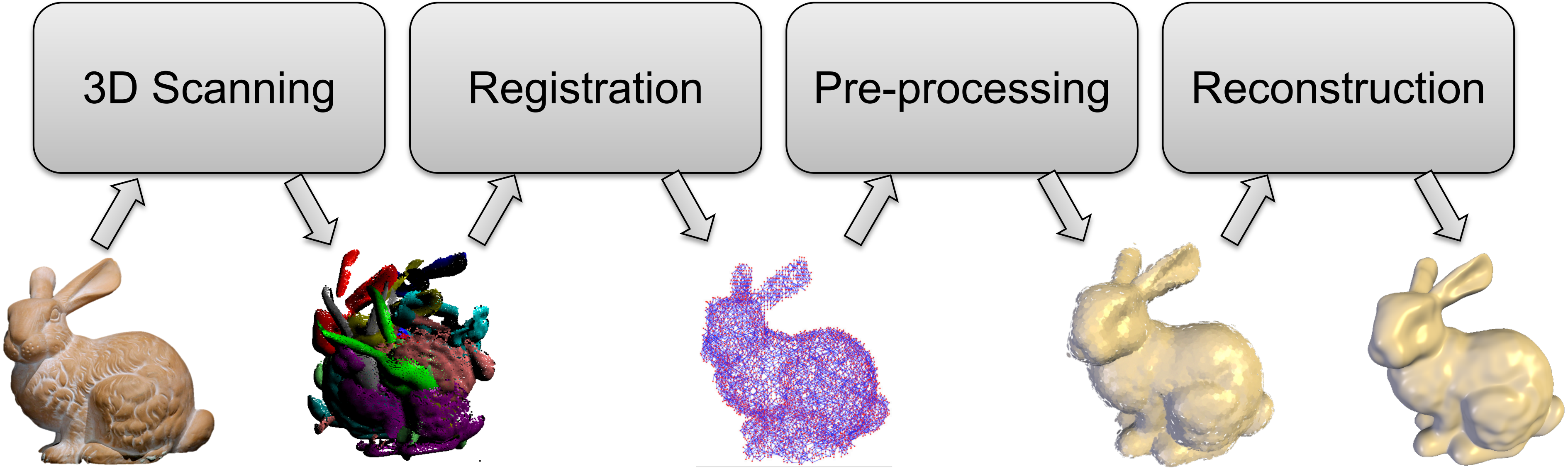
Modeling applications





Shape Acquisition

- Digitizing real world objects



Shape Acquisition

- 3D Scanning

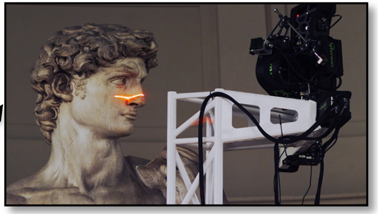
Touch Probes

+ Precise
- Small objects

Optical Scanning

+ Fast
- Glossy objects

Active



Passive

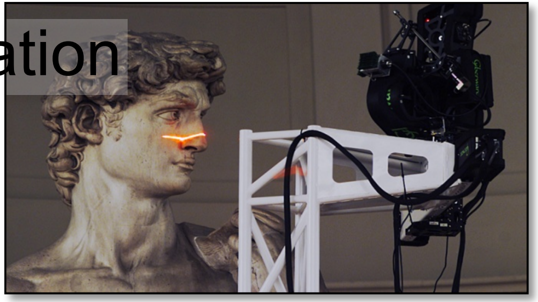
Shape Acquisition

- Optical Scanning – Active Systems

LIDAR

Measures the time it takes the laser beam to hit the object and come back

Triangulation Laser

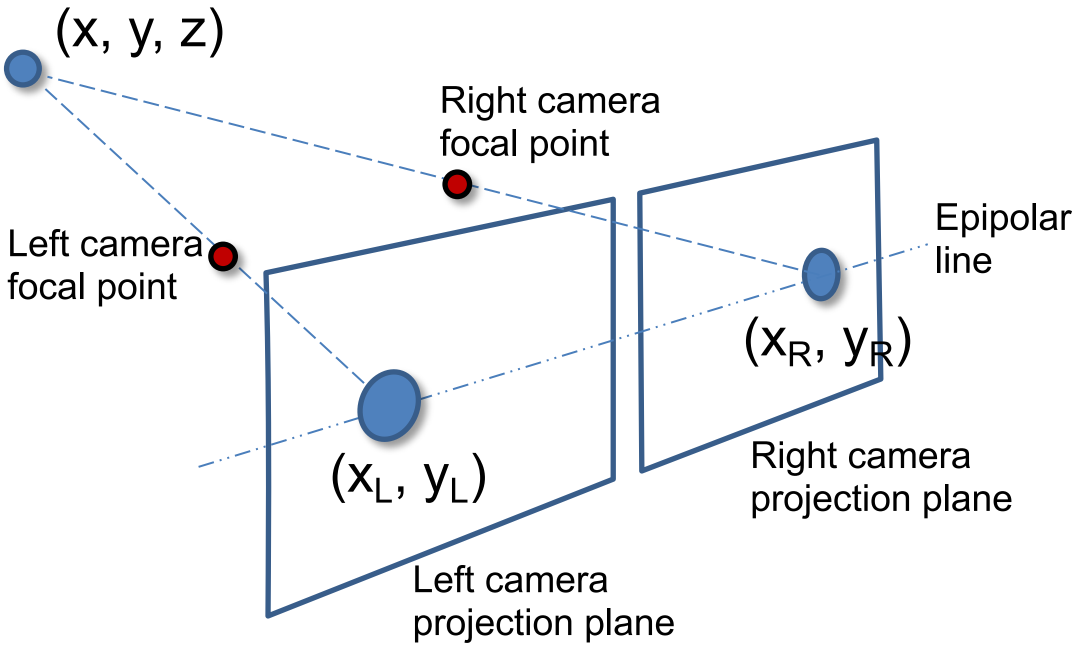


Projected laser beam is photographed, giving the distance of the pattern

Shape Acquisition

- Optical Scanning – Passive Systems

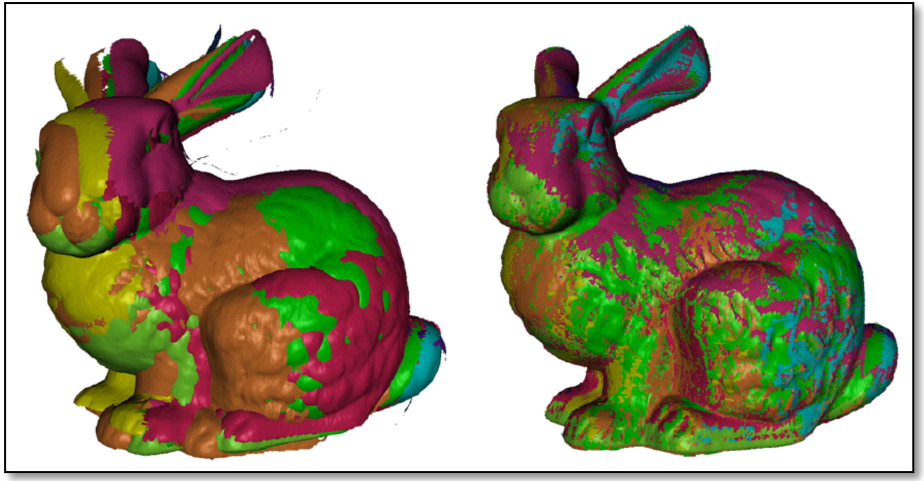
Multi-view Stereo



UNIVERSITY OF CAMBRIDGE

Shape Acquisition

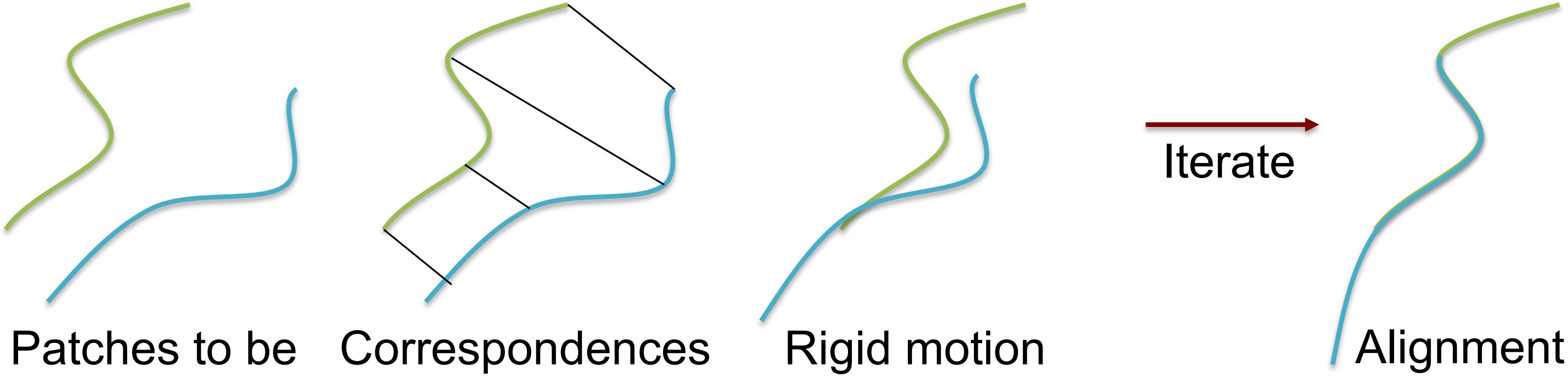
- Registration
  - Bringing scans into a common coordinate frame



Shape Acquisition

- Registration

Iterative Closest Point Algorithms



Patches to be aligned

Correspondences

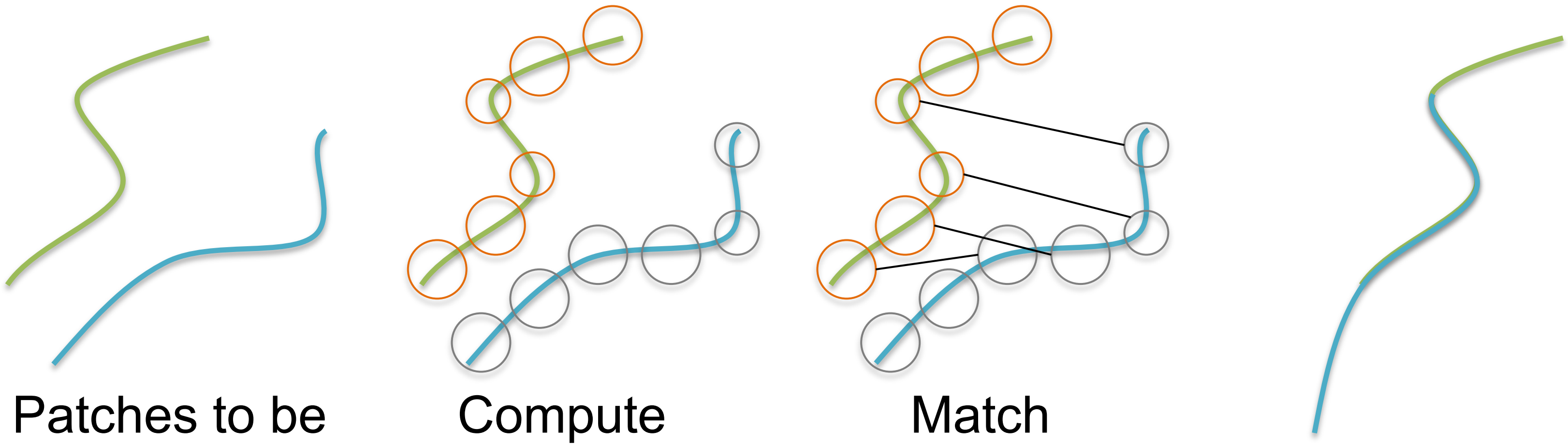
Rigid motion

Alignment

Shape Acquisition

- Registration

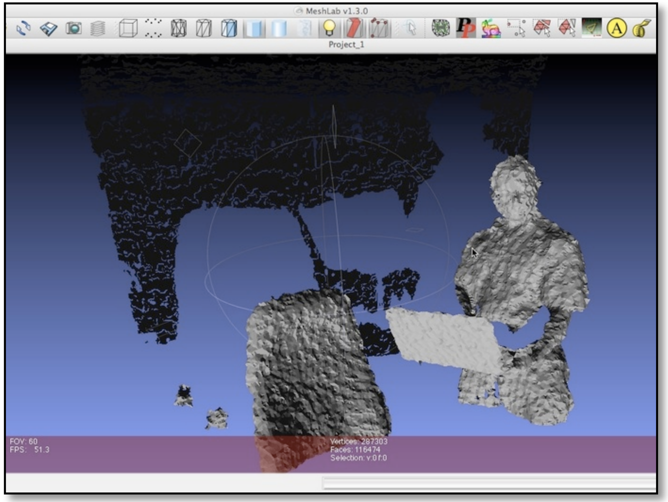
Feature-based Methods



Patches to be aligned | Compute descriptors | Match descriptors | Alignment

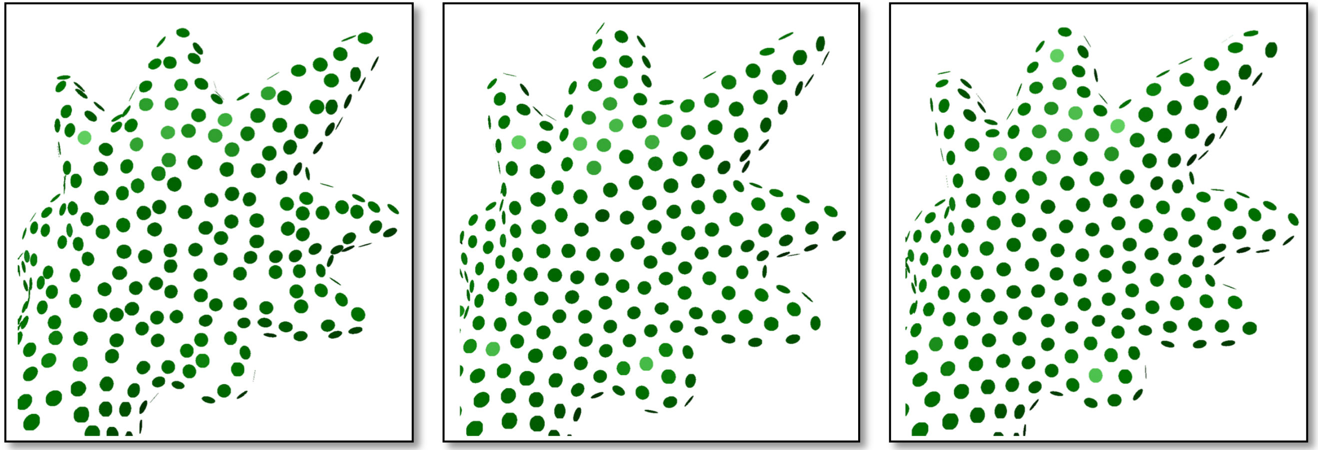
Shape Acquisition

- Pre-processing
  - Cleaning, repairing, resampling



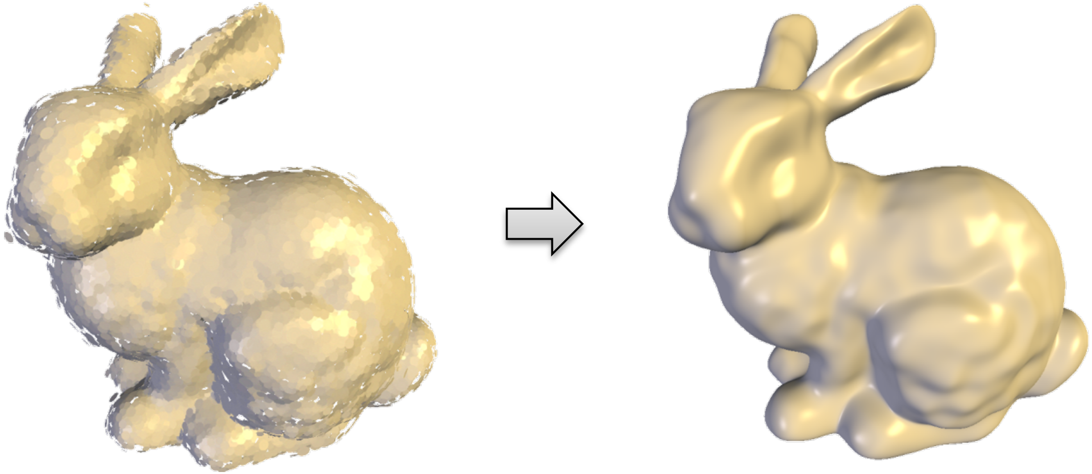
Shape Acquisition

- Pre-processing
  - Sampling for accurate reconstructions



Shape Acquisition

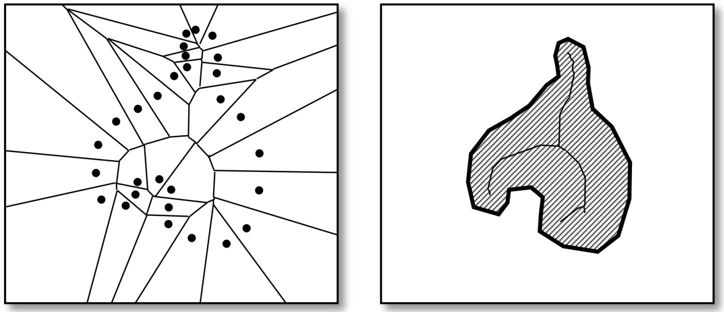
- Reconstruction
  - Mathematical representation for a shape



Shape Acquisition

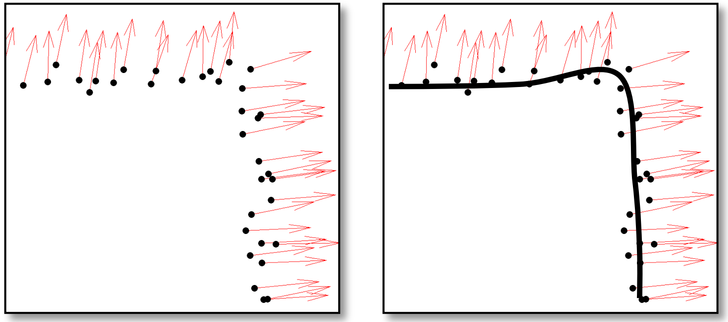
- Reconstruction

Connect-the-points Methods



+ Theoretical error bounds
– Expensive
– Not robust to noise

Approximation-based Methods



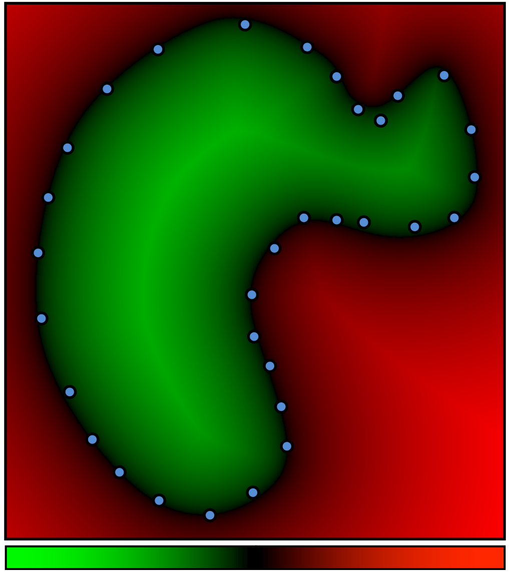
+ Efficient to compute
+ Robust to noise
– No theoretical error bounds

Shape Acquisition

- Approximating an implicit function

$$f : \mathbb{R}^3 \rightarrow \mathbb{R}$$

with value > 0 outside the shape and < 0 inside



Shape Acquisition

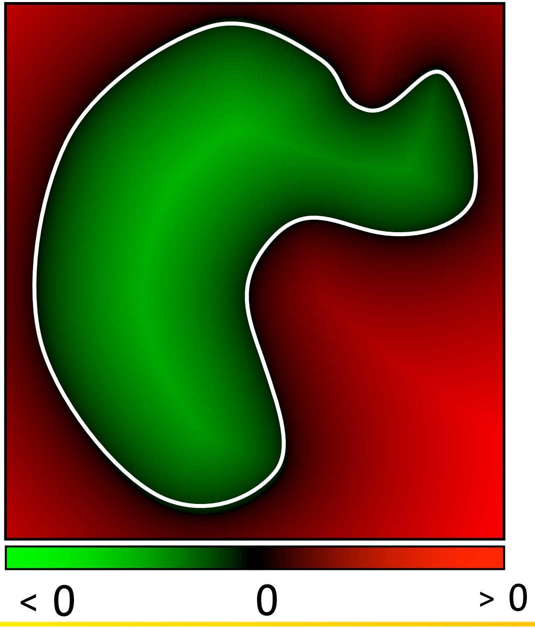
- Approximating an implicit function

$$f : \mathbb{R}^3 \to \mathbb{R}$$

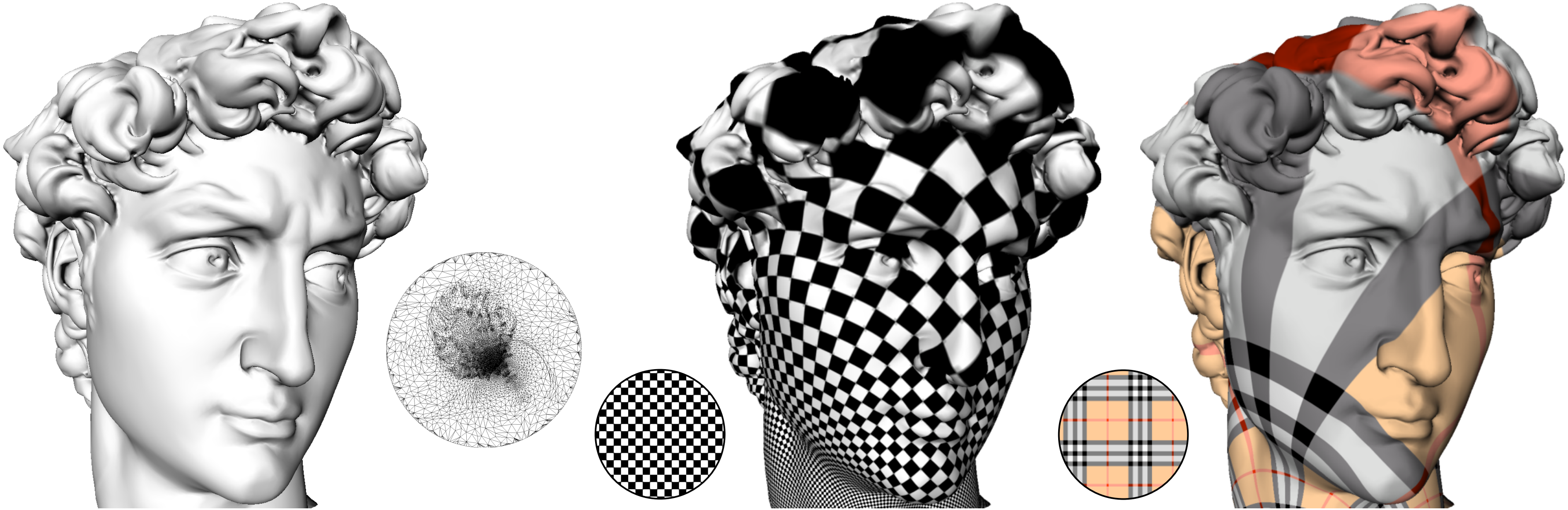
with value > 0 outside the shape and < 0 inside

$$\{\mathbf{x} : f(\mathbf{x}) = 0\}$$

extract zero set

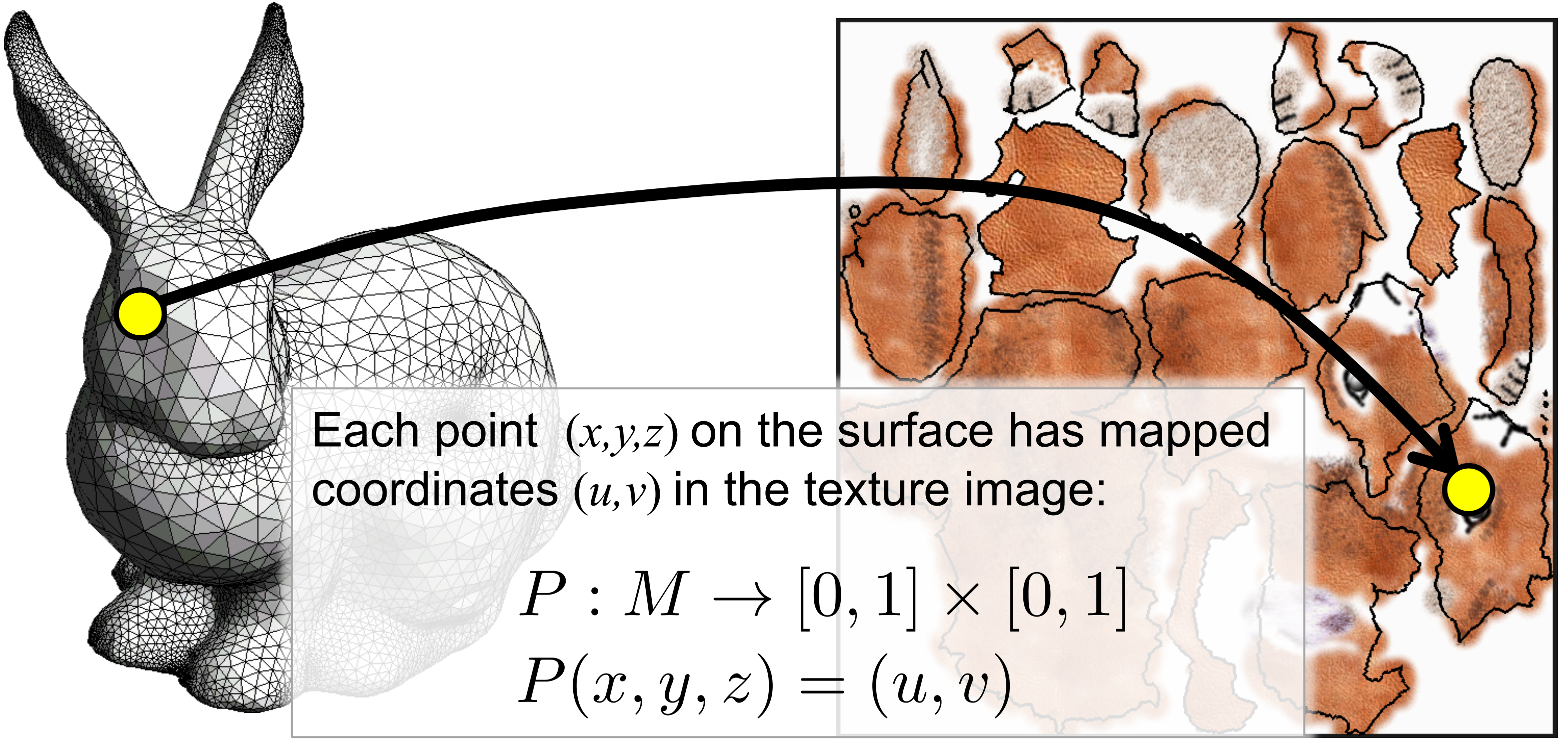


Texture Mapping



Parametrization

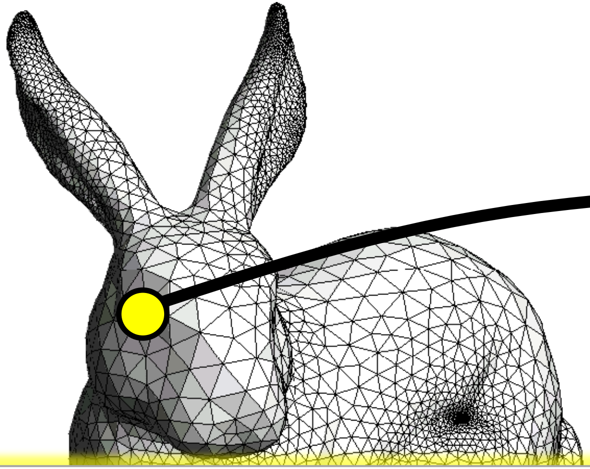
Texture Mapping

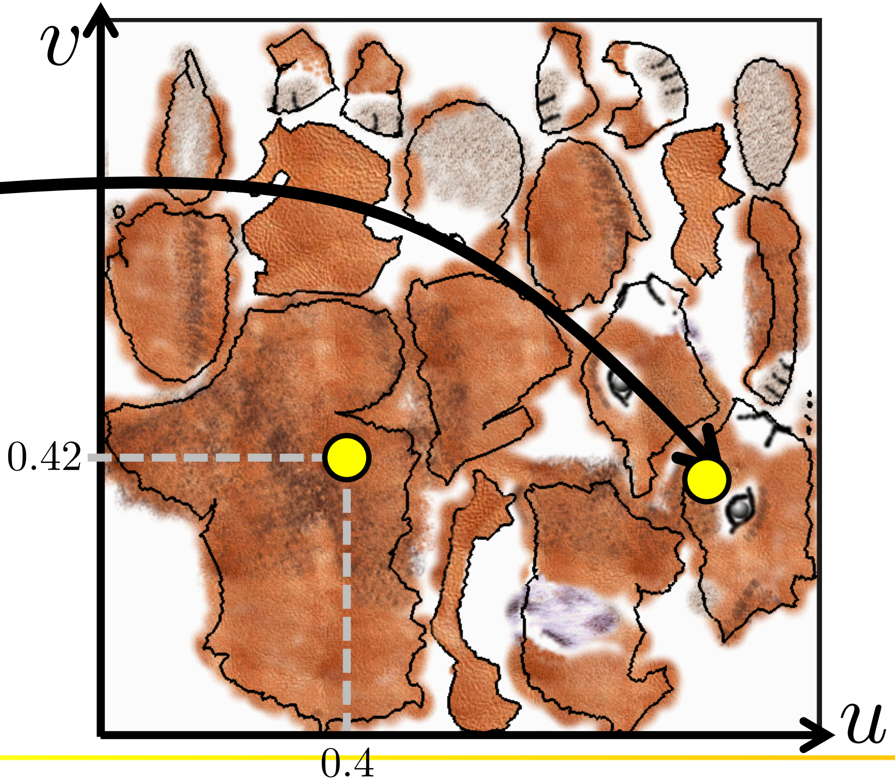


Each point (x,y,z) on the surface has mapped coordinates (u,v) in the texture image:

$$P : M \to [0,1] \times [0,1]$$
$$P(x,y,z) = (u,v)$$

Texture Mapping

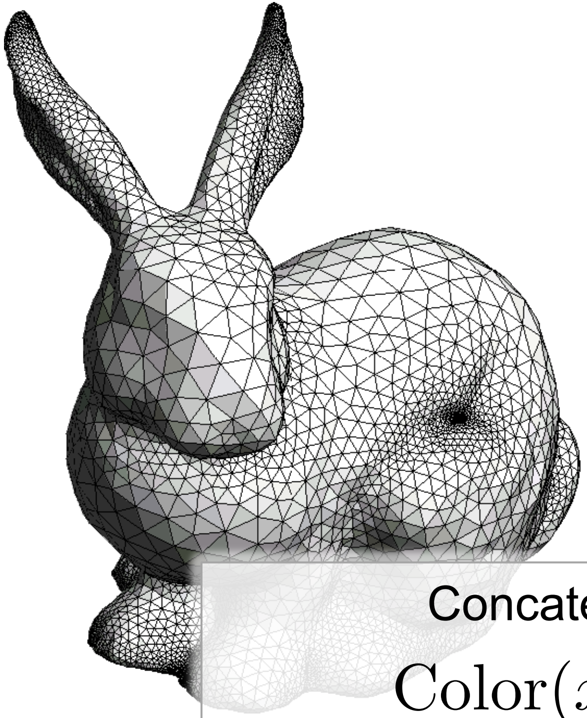


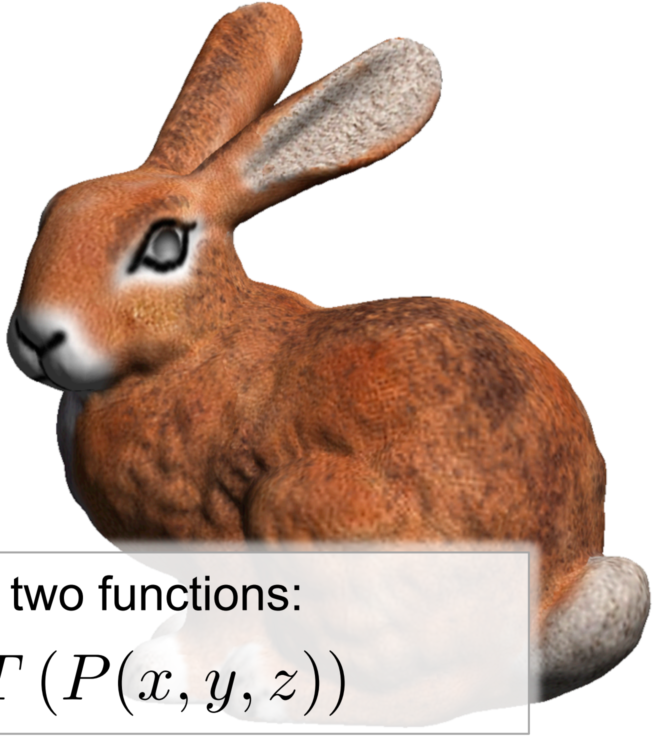


Texture itself is a function:

$$T : [0,1] \times [0,1] \to \text{RGB}$$
$$T(u,v) = (r,g,b)$$

Texture Mapping

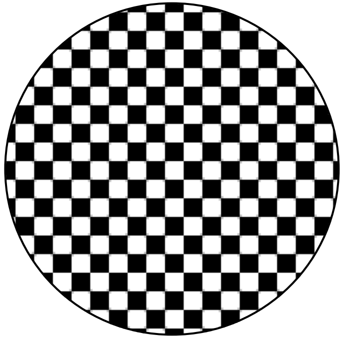




Concatenation of the two functions:

$$\mathrm{Color}(x, y, z) = T\left(P(x, y, z)\right)$$

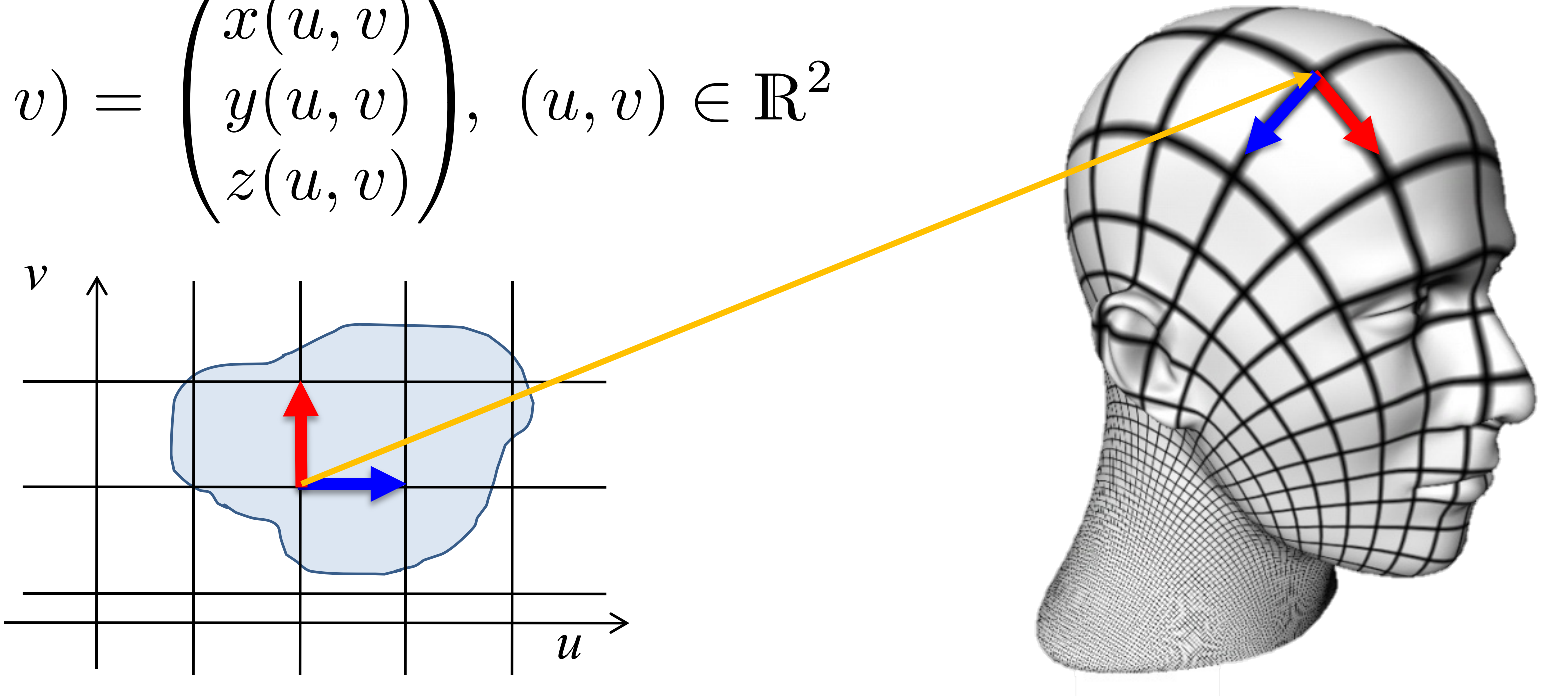
Parametrization



Texture image

Parametrization

$$\mathbf{p}(u,v) = \begin{pmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{pmatrix}, \quad (u,v) \in \mathbb{R}^2$$

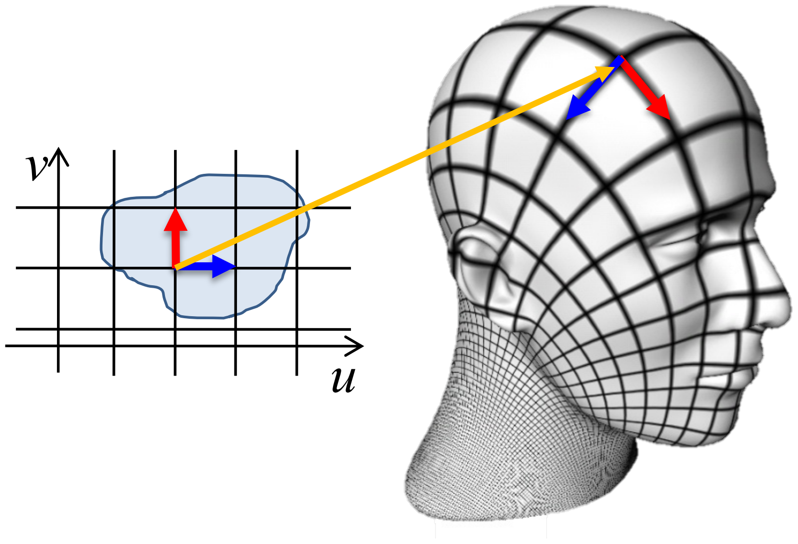


Parametrization

$$\mathbf{p}(u,v) = \begin{pmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{pmatrix}, \quad (u,v) \in \mathbb{R}^2$$

$$\mathbf{p}_u = \frac{\partial \mathbf{p}(u,v)}{\partial u}, \quad \mathbf{p}_v = \frac{\partial \mathbf{p}(u,v)}{\partial v}$$

$$\mathbf{I} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix}$$

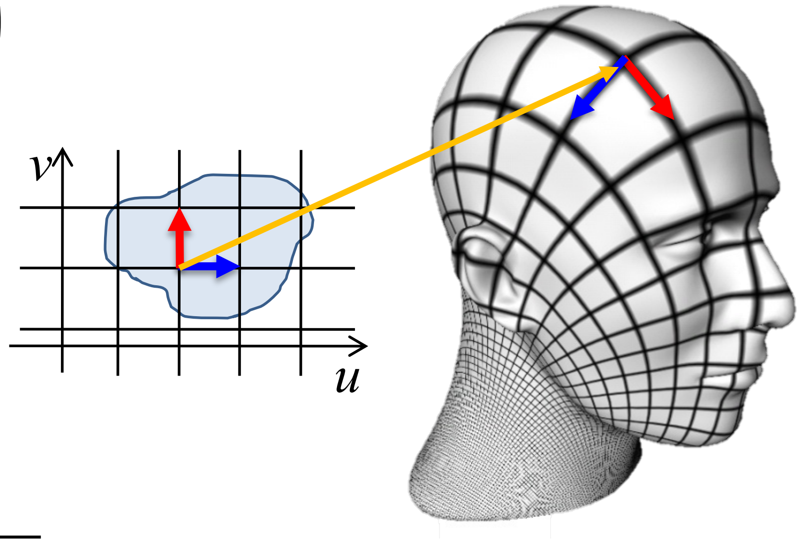


Parametrization

$$\mathbf{I} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix}$$

Angle change

Length change

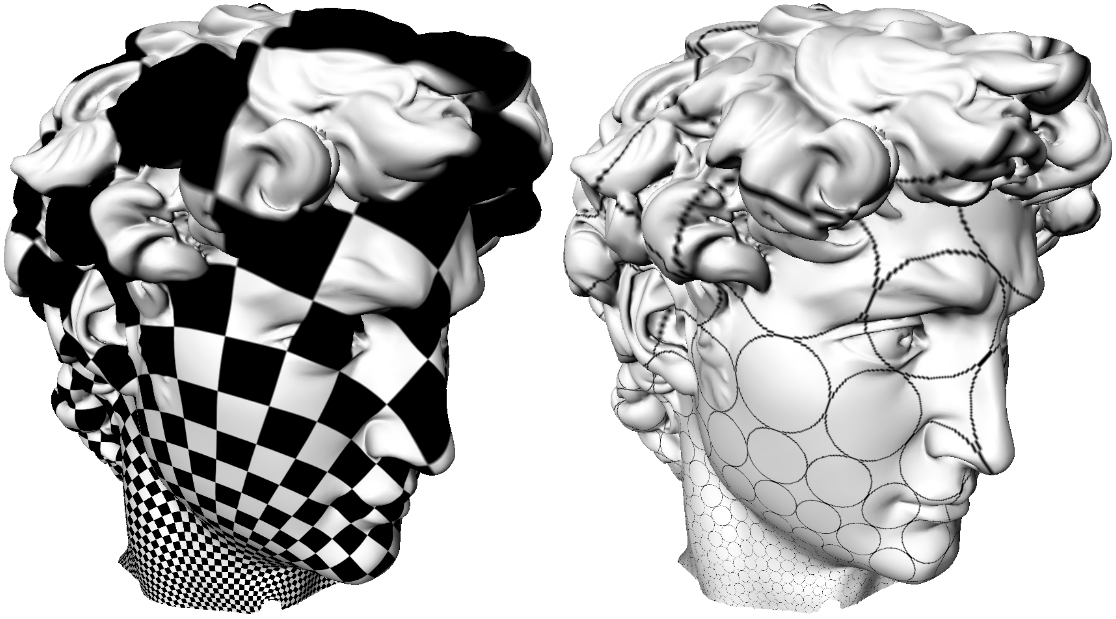


Area distortion: $dA = \sqrt{EG - F^2}\, du dv$

Parametrization

Conformal parametrization
(angle preservation)

$$\mathbf{I} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix} = \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix}$$

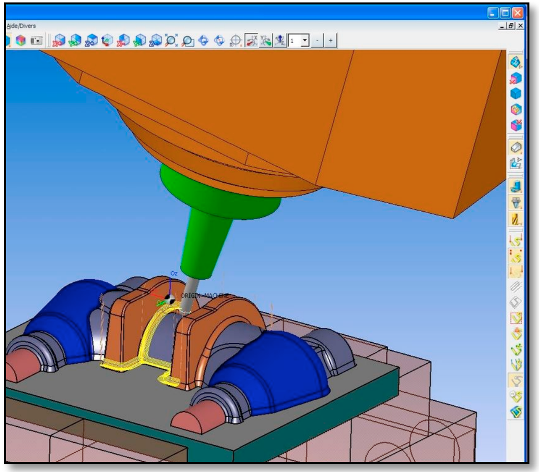


Editing Geometry

- Modeling tools

Sculpting

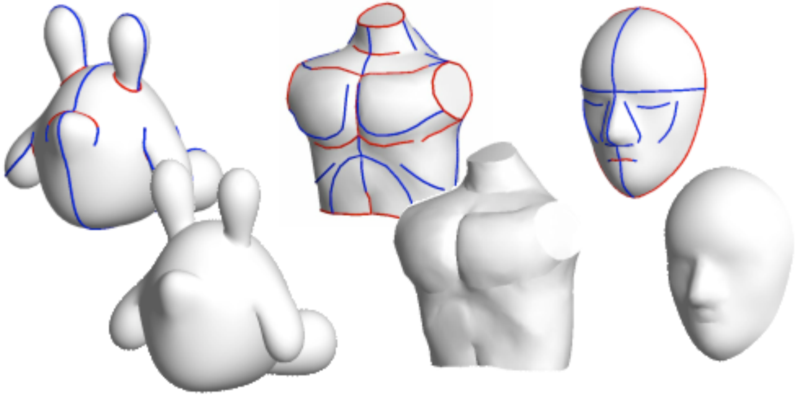
CAD/CAM

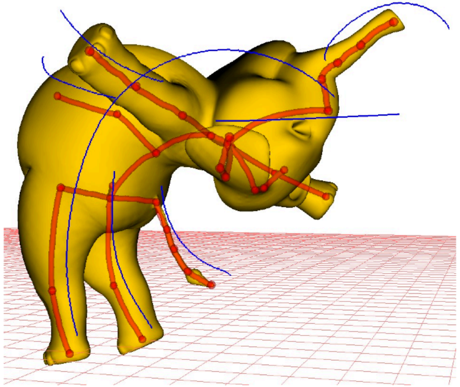


Procedural

Editing Geometry

- Interactive & sketch-based interfaces

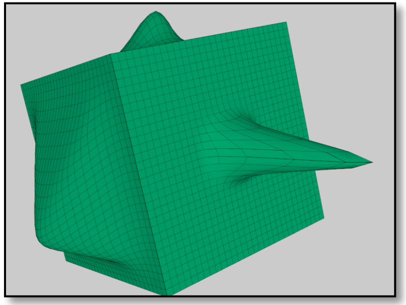




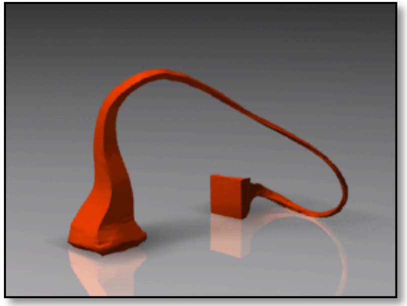
Editing Geometry

- Deformations

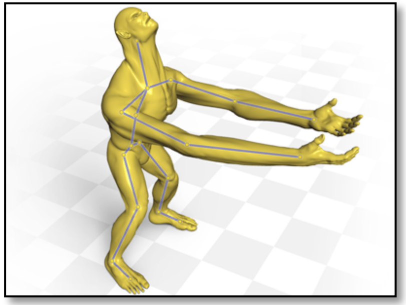
Free-form



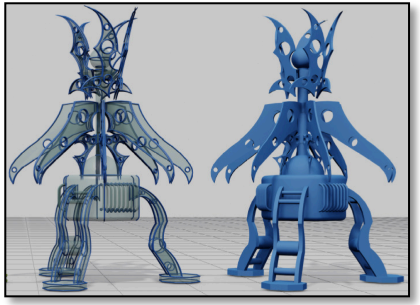
Elastic



Skeletal



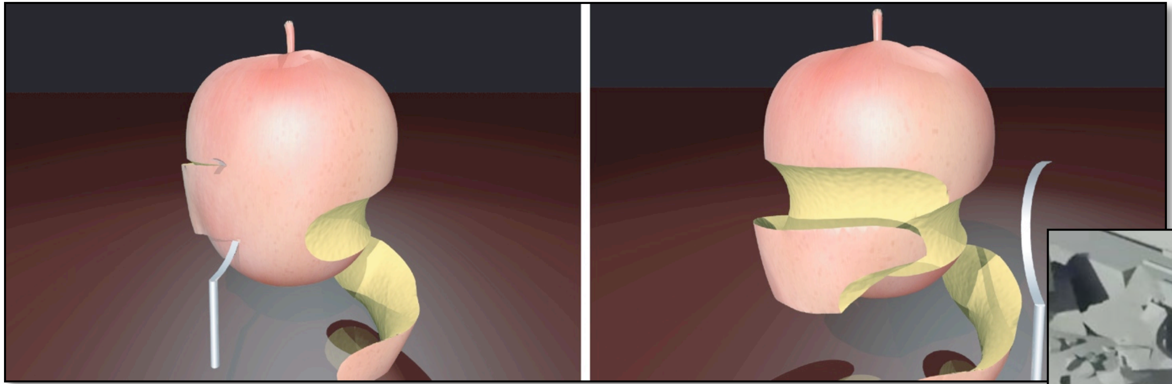
Structure-aware

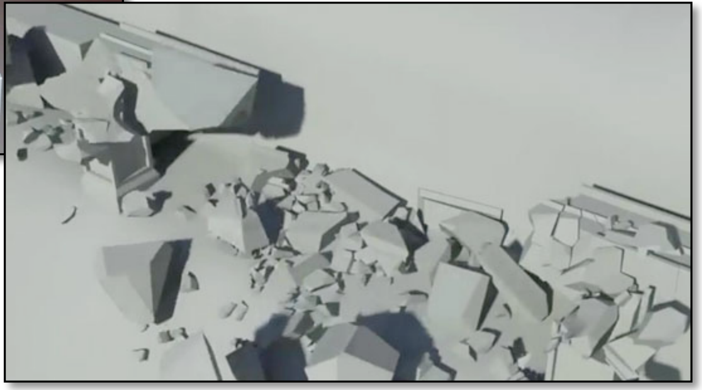


More structure

Editing Geometry

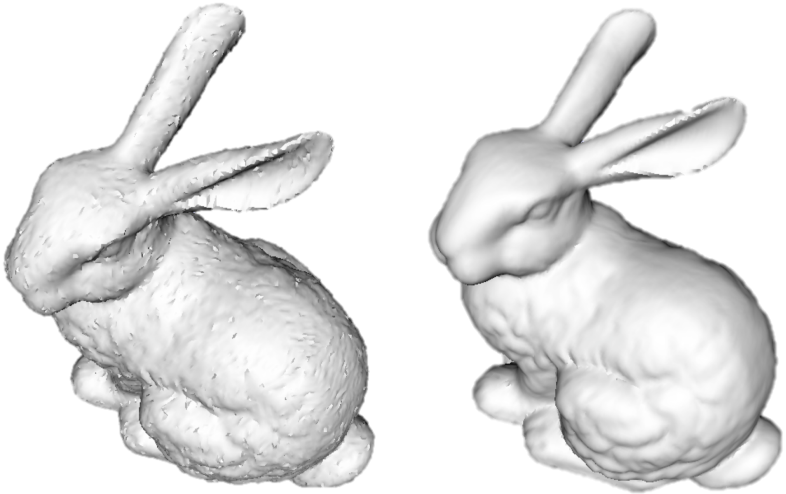
- Cutting & fracturing

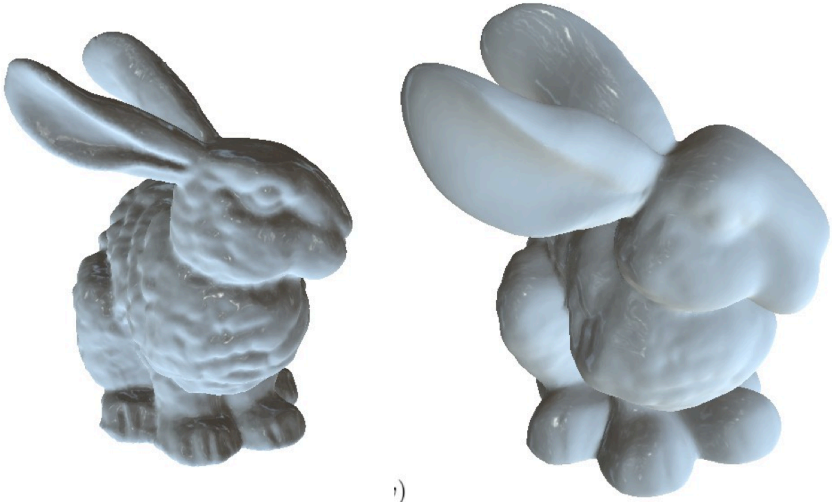




Editing Geometry

- Smoothing & filtering





Editing Geometry

- Compression & Simplification