中国科学技术大学
University of Science and Technology of China

GAMES 102在线课程

几何建模与处理基础

刘利刚
中国科学技术大学

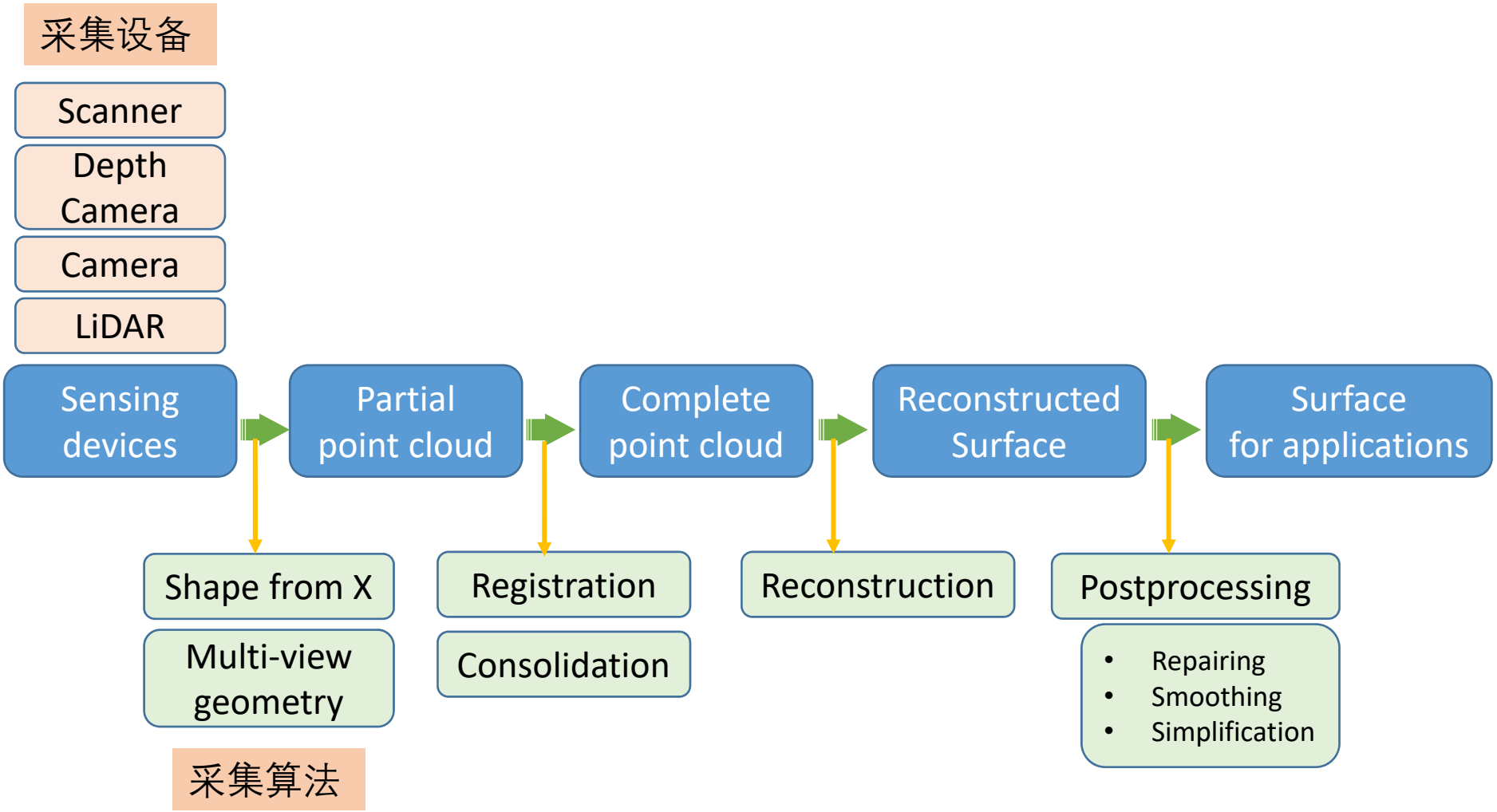
中国科学技术大学

University of Science and Technology of China

GAMES 102在线课程：几何建模与处理基础

几何建模

回顾：曲面重建


采集设备
Scanner
Depth Camera
Camera
LiDAR
Sensing devices
Partial point cloud
Complete point cloud
Reconstructed Surface
Surface for applications
Shape from X
Multi-view geometry
Registration
Consolidation
Reconstruction
Postprocessing
Repairing
Smoothing
Simplification
采集算法

建模(modeling)：设计与重建

- 曲面设计(Design)
  - 不存在的物体：通过人工交互凭空设计出新的物体
    - CAGD (NURBS)、mesh modeling
  - 存在的物体：通过人工交互编辑修改构建出新的物体
    - Editing, deformation
- 曲面重建(Reconstruction)
  - 存在的物体：对其采集并进行数字化构建
  - 也称为：逆向工程、扫描重建
    - Reverse engineering, scanning

Design Modeling

- Design from zero
  - Create a shape by a set of 3D design operators such as extrusion or revolution etc.

- Design from a given shape
  - Select a base shape
  - Select editing elements and editing operators
  - Deform the shape to obtain a new shape

Design Modeling

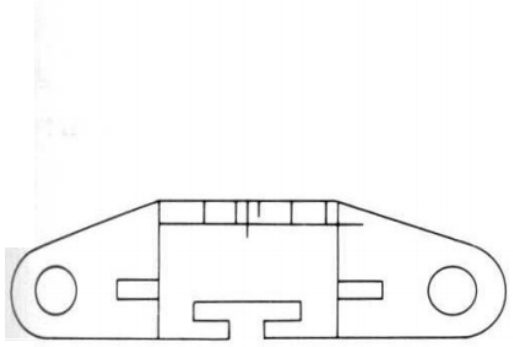
- Design from zero
  - Create a shape by a set of 3D design operators such as extrusion or revolution etc.

- Design from a given shape
  - Select a base shape
  - Select editing elements and editing operators
  - Deform the shape to obtain a new shape

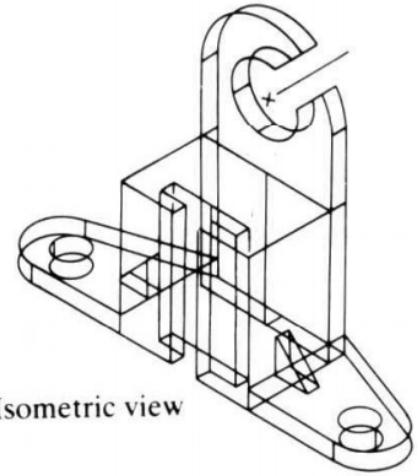
Geometric Modeling Techniques

- Wireframe Modeling
- Surface Modeling
  - Analytical Surface
  - Free-form, Curved, Sculptured Surface
- Solid Modeling Solid Modeling
  - Parametric Modeling
  - Feature Based Modeling
  - Constructive Solid Geometry (CSG)
  - Boundary Representation (B-Rep)

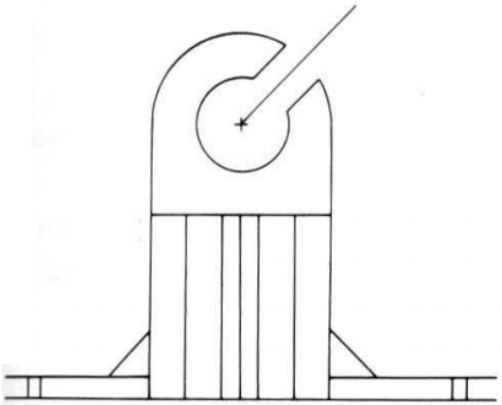
Wireframe Modeling （工程制图）



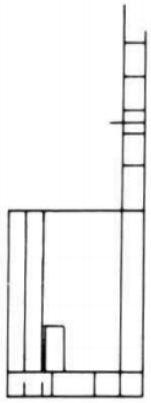
Top view



Isometric view

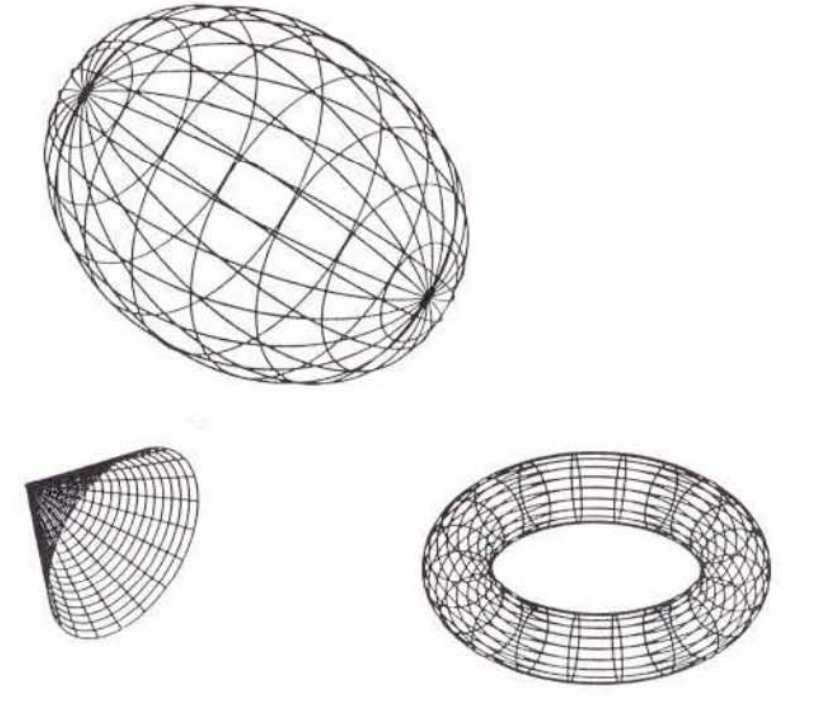


Front view

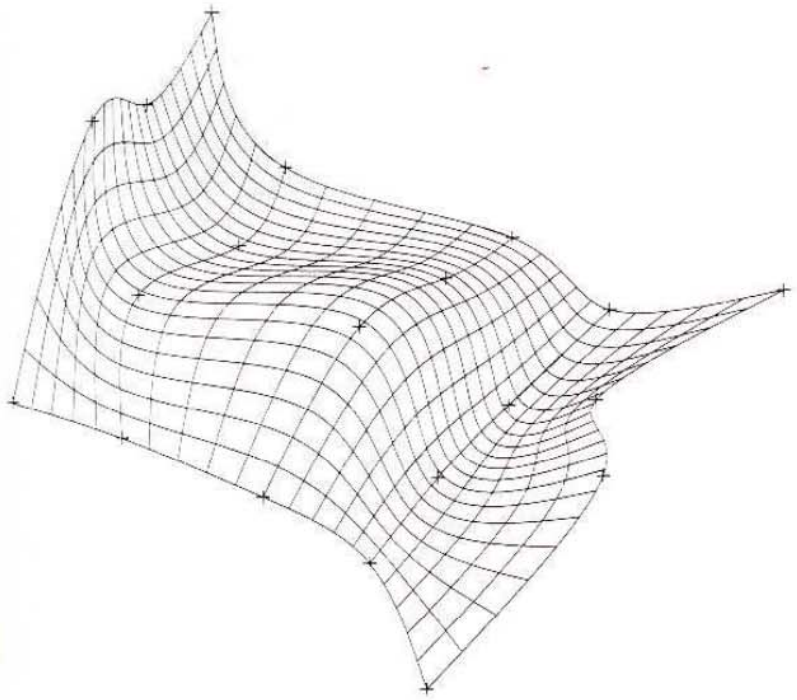


Right view

Surface Modeling (NURBS)



Analytical Surfaces



Free-form, Curved, or
Sculptured Surface

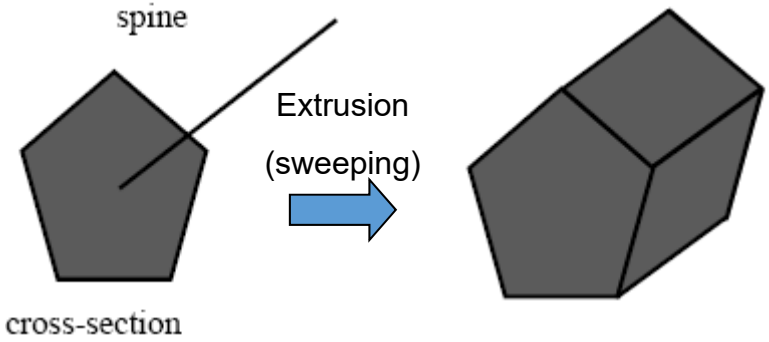
中国科学技术大学

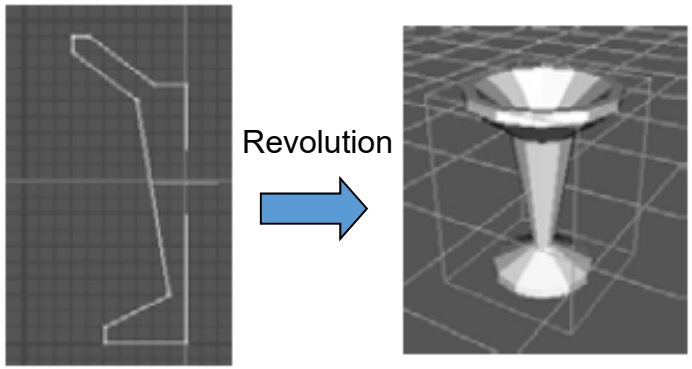
University of Science and Technology of China

Solid Modeling

Solid Modeling

- Modeling by a set of geometric operators
  - Geometric operators: extrusion, revolution, sweeping, lofting, etc.
  - Parametric: various semantic parameters
  - Feature-based: various components such as holes, grooves, fillets, and chamfers.





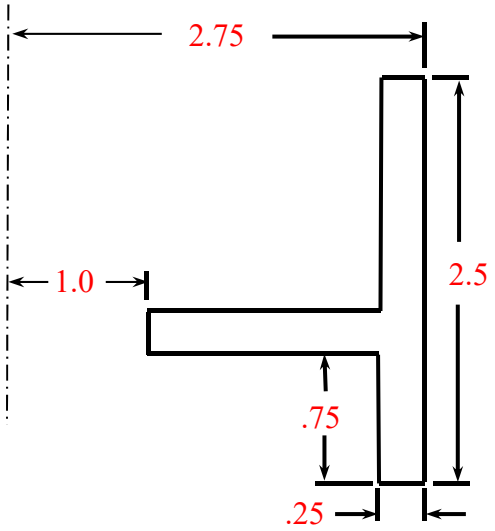
- Commercial software
  - SolidWorks, Creo (PTC), Inventor & Fusion 360 by Autodesk, Unigraphics, Catia, …..

Concepts of Parametric Modeling

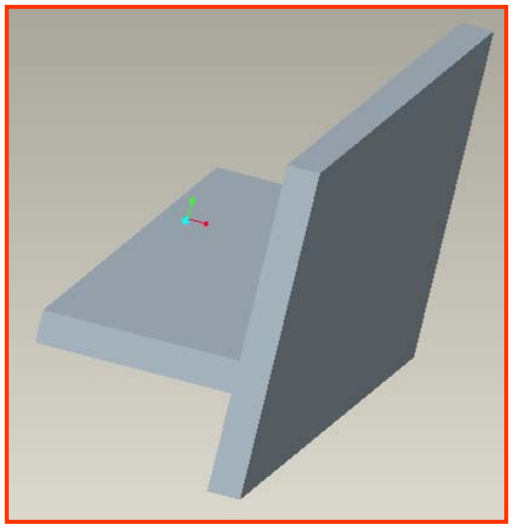
- Sketching
  - Take the word sketch literally. A sketch should be just that, a sketch.
  - When sketching it is not necessary to create geometry with accuracy. Lines, arcs, and additional geometry need not be created with exact dimensions in mind.
- Dimensioning
  - When dimensions are added, the sketch will change size and shape. This is the essence of Parametric Modeling.
- Features
  - Create a 2D sketch and dimension it.
  - Create a feature from the sketch by extruding, revolving, sweeping, and lofting.

Extruding and revolving

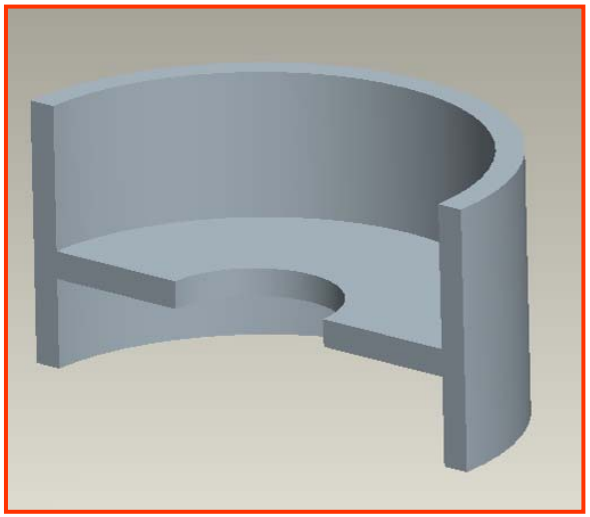
- Extrusion: 平移，沿直线段挤出（扫掠sweeping）
- Revolution: 旋转曲面，沿圆弧挤出



Sketching: profile



Extruding

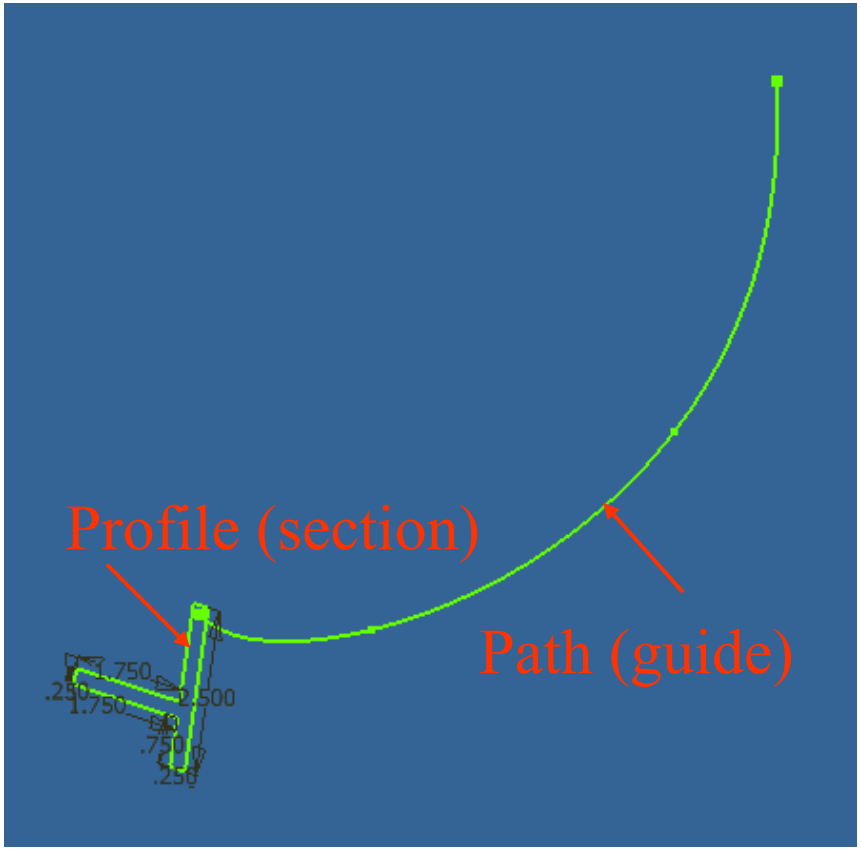


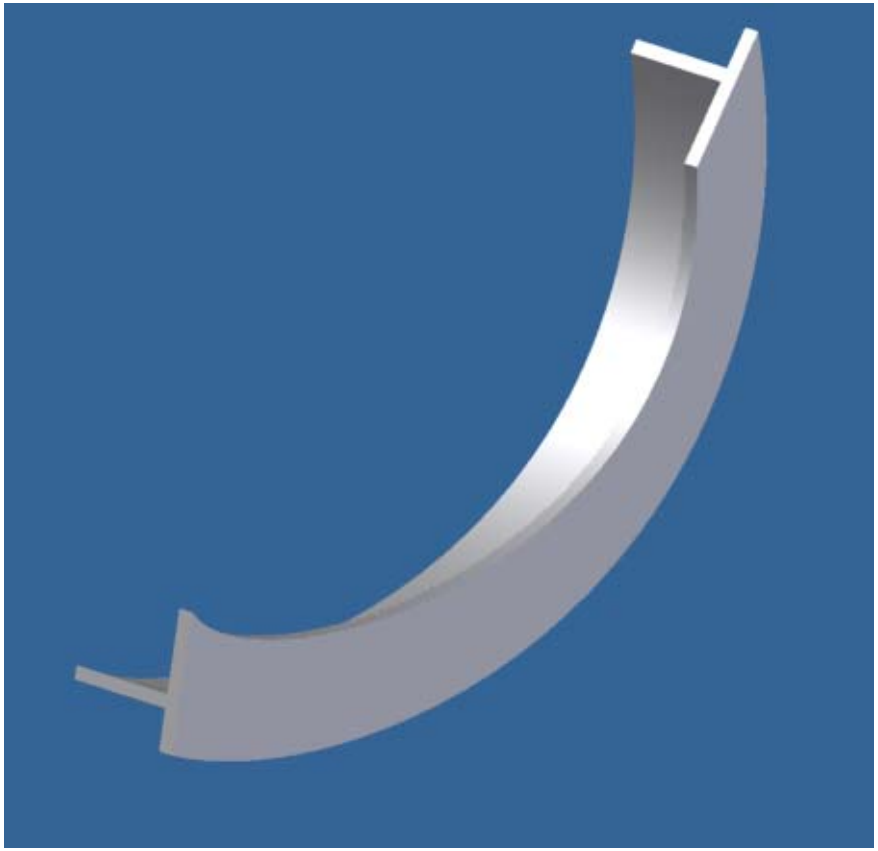
Revolving

Courtesy of Ken Youssefi

Sweeping

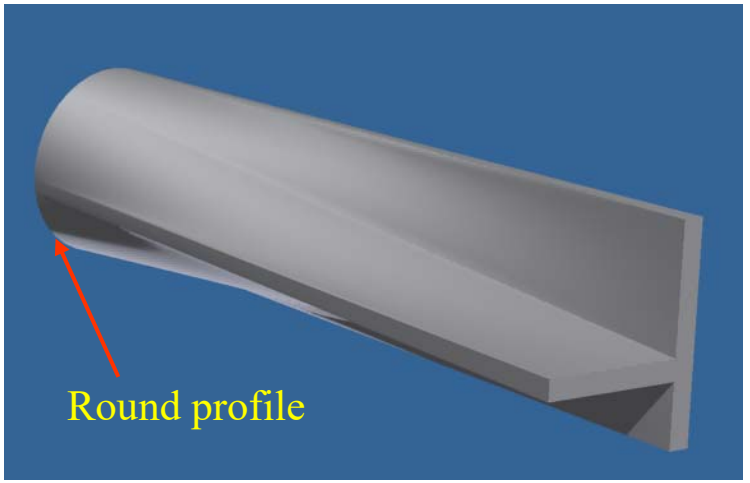
- A sweep feature requires a profile and a path. The profile will follow the path to create the solid.

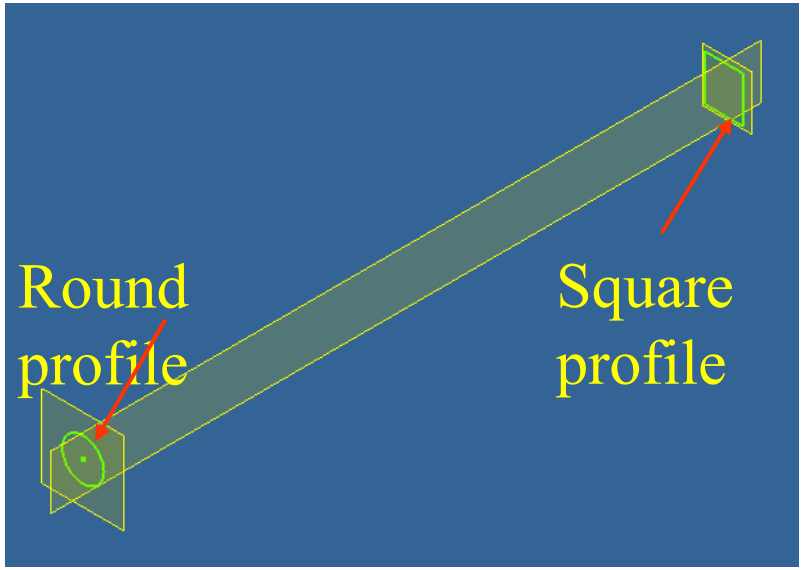


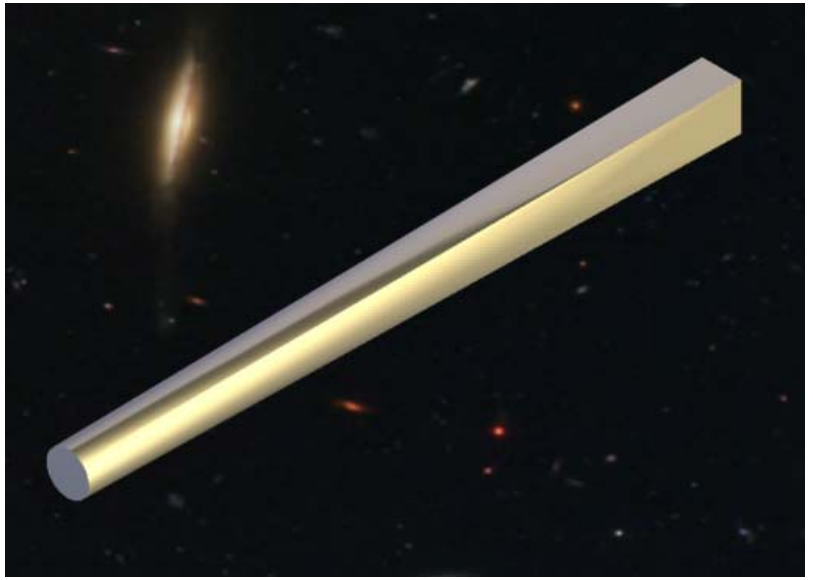


Lofting

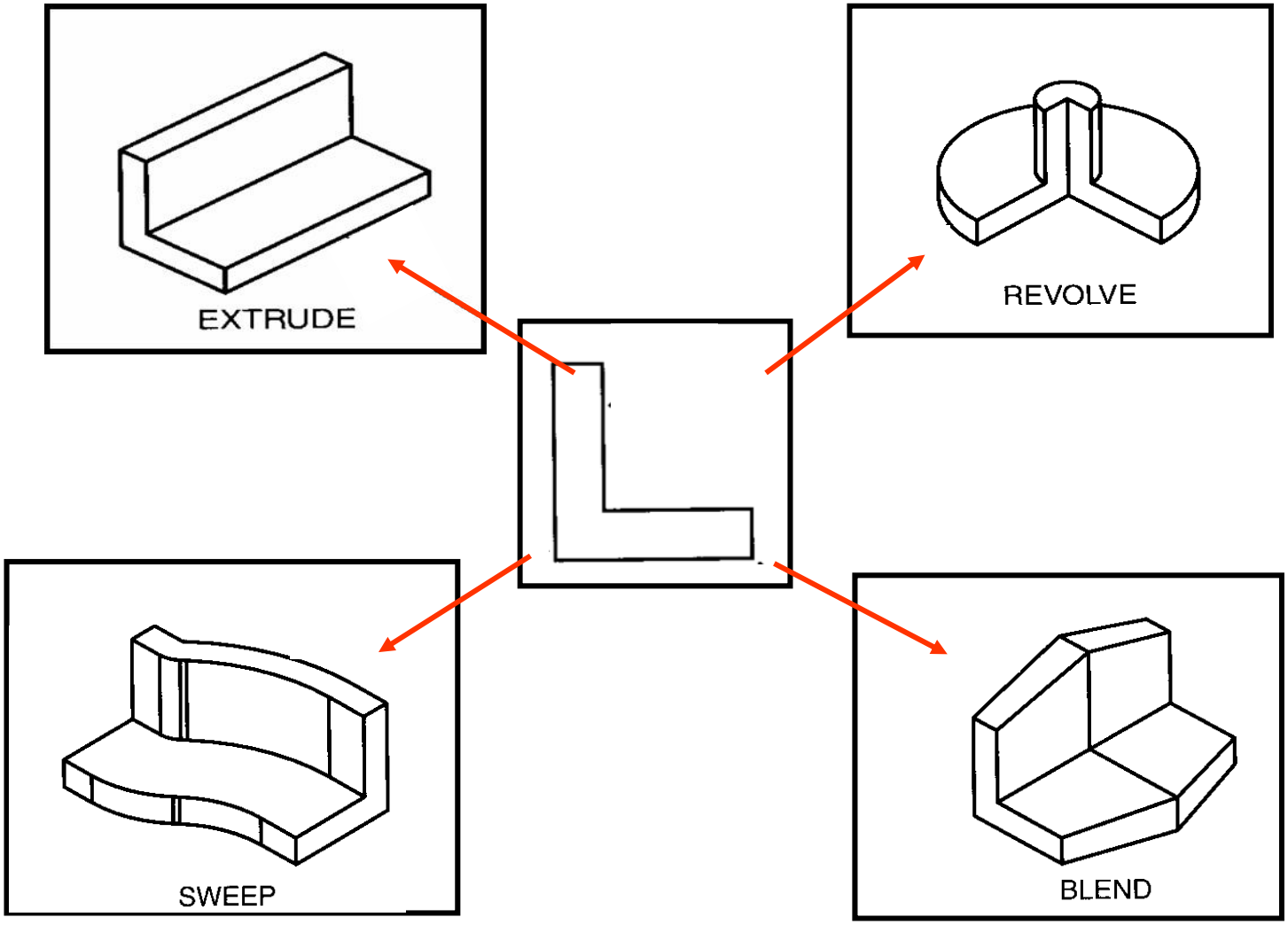
- Sweeping with different profiles
  - Sections (profiles) do not have to be sketched on parallel planes
  - All sections must be either closed or open







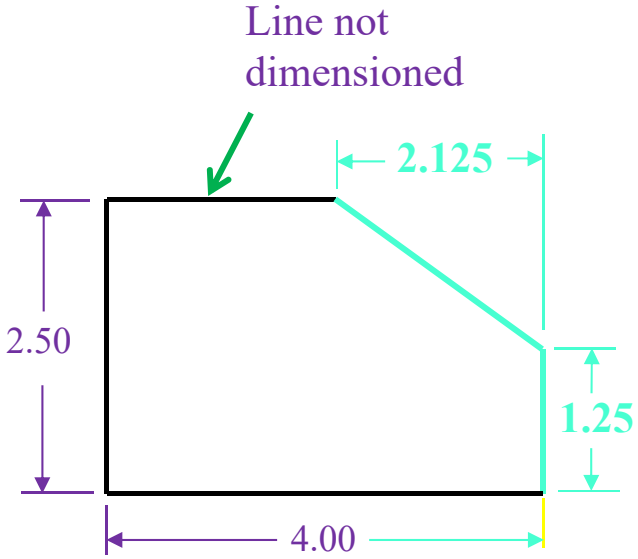
Creating Features from Sketches

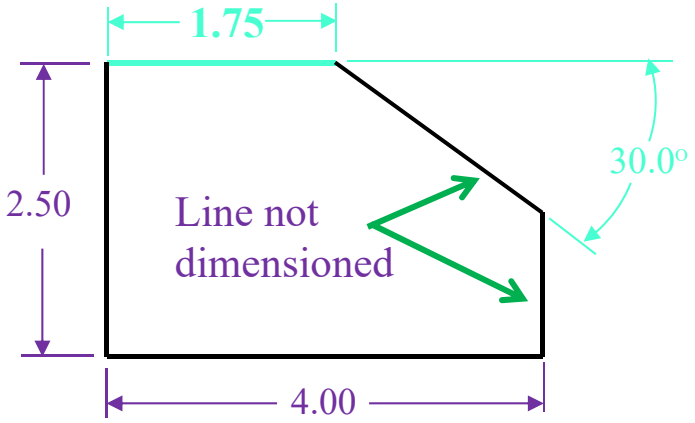


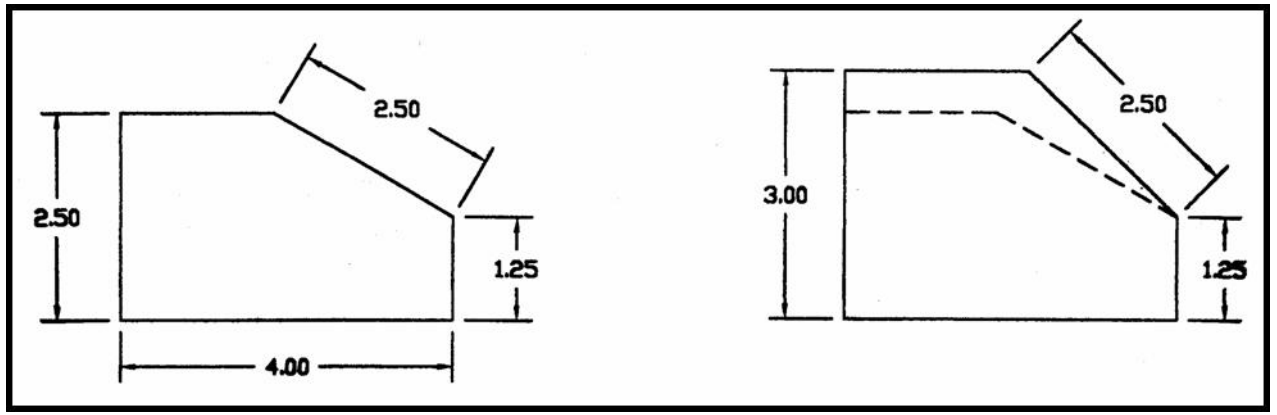
Loft in SolidWorks

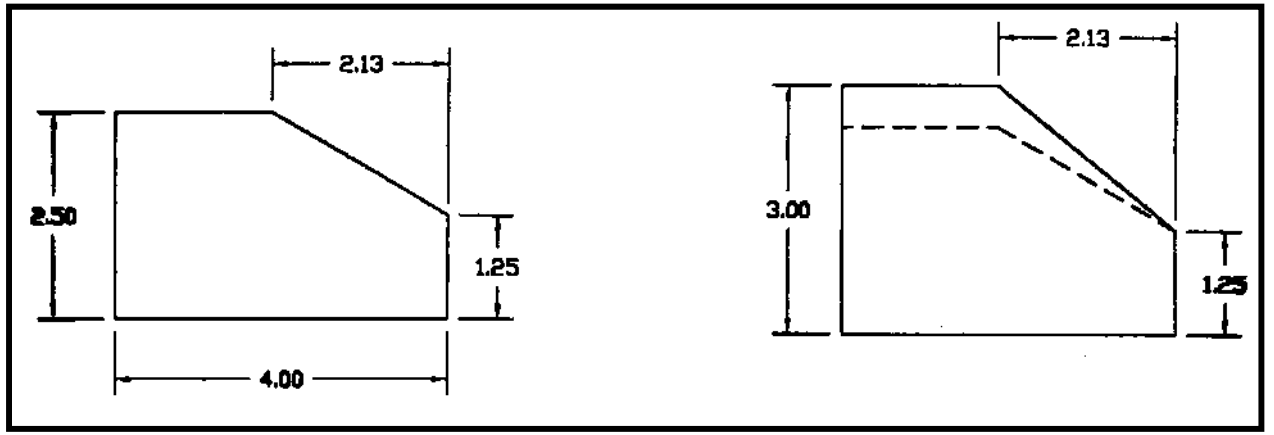
Design Intent (constraints)

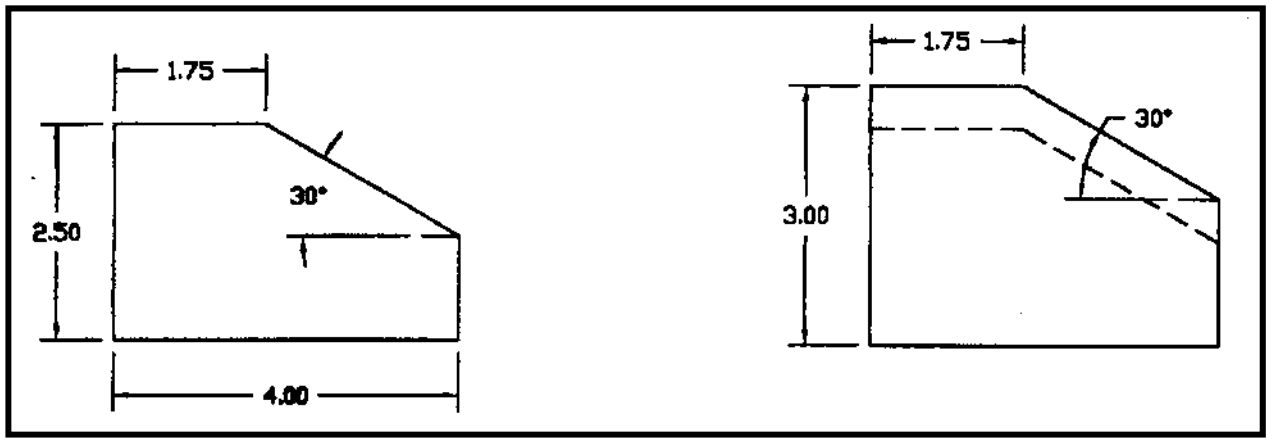
- How your model will react when dimension values are changed?





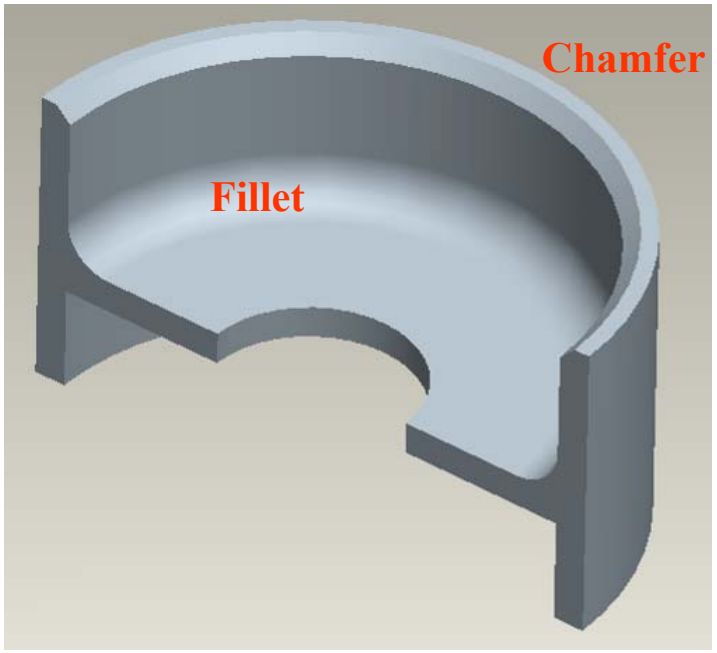

2.50
2.50
1.25
4.00
2.50
3.00
1.25

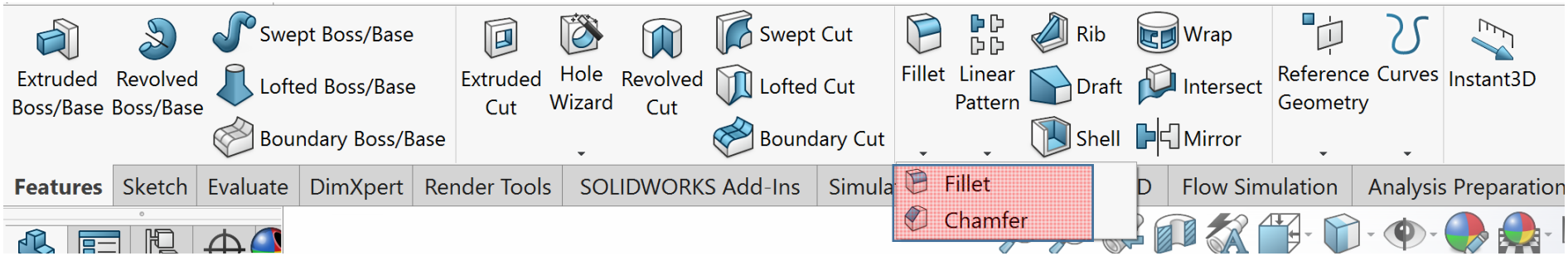

2.13
2.50
1.25
4.00
2.13
3.00
1.25


1.75
30°
2.50
4.00
1.75
30°
3.00

Applied Feature: Chamfer and Fillet

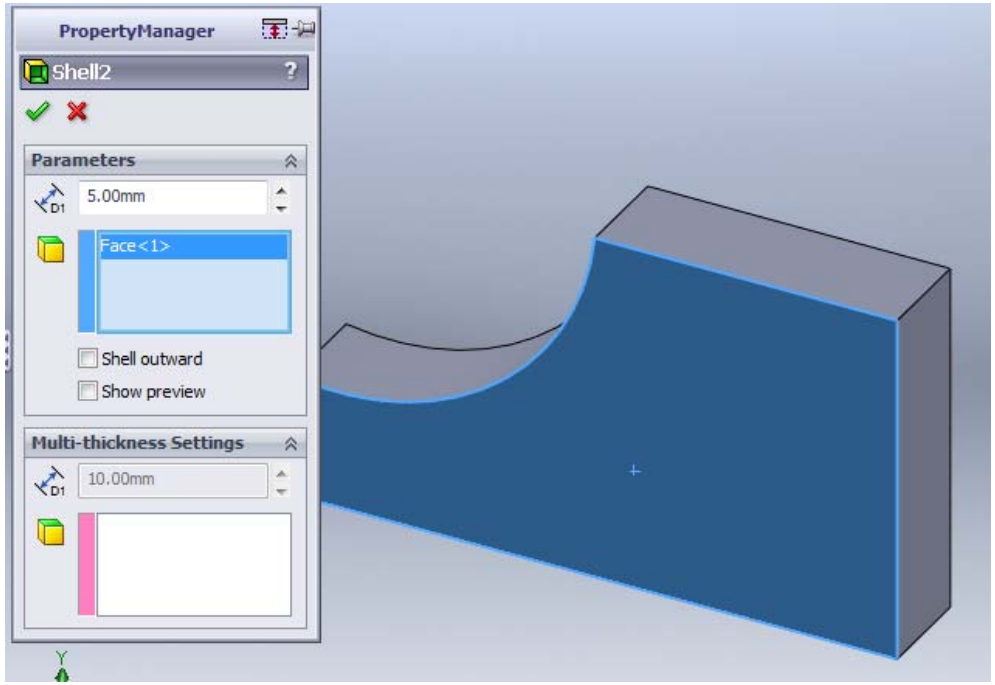
- Applied feature does not require a sketch.
- It is applied directly to the model.

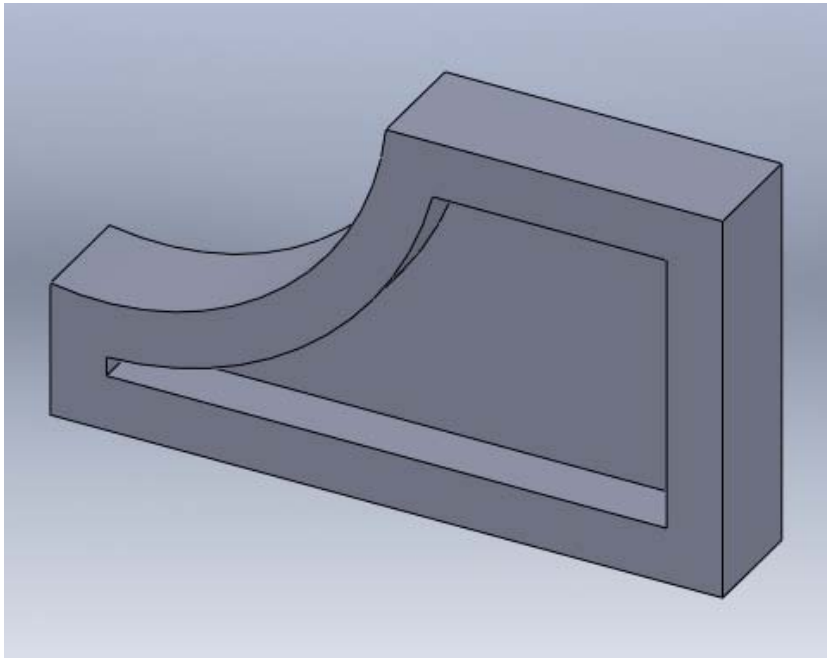


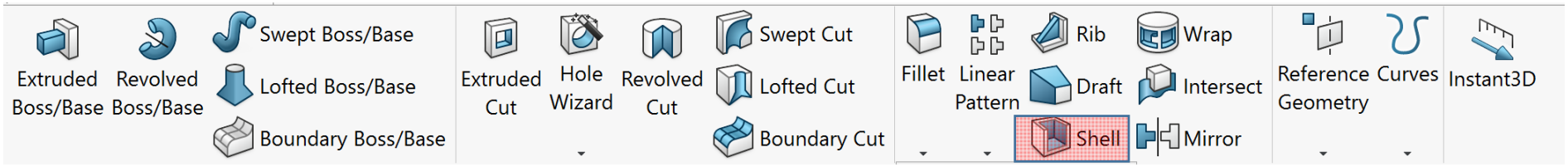


Applied Feature: Shell

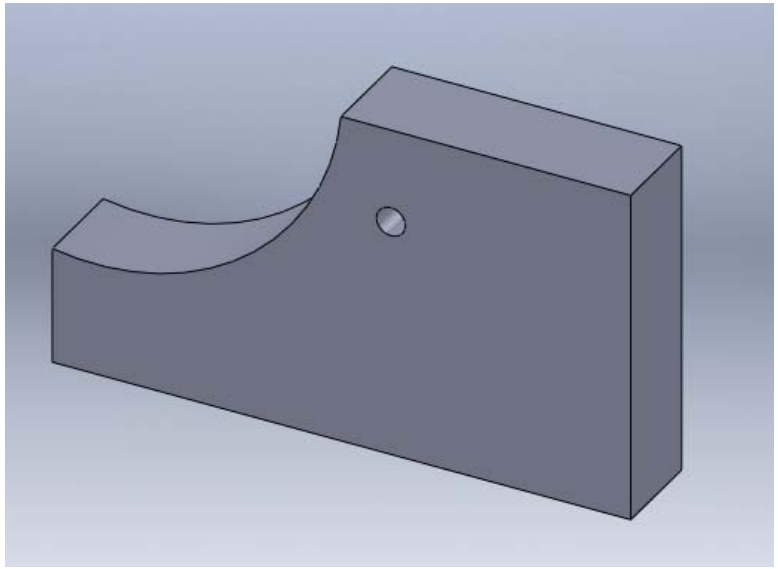
- Shell – hollowing out a solid



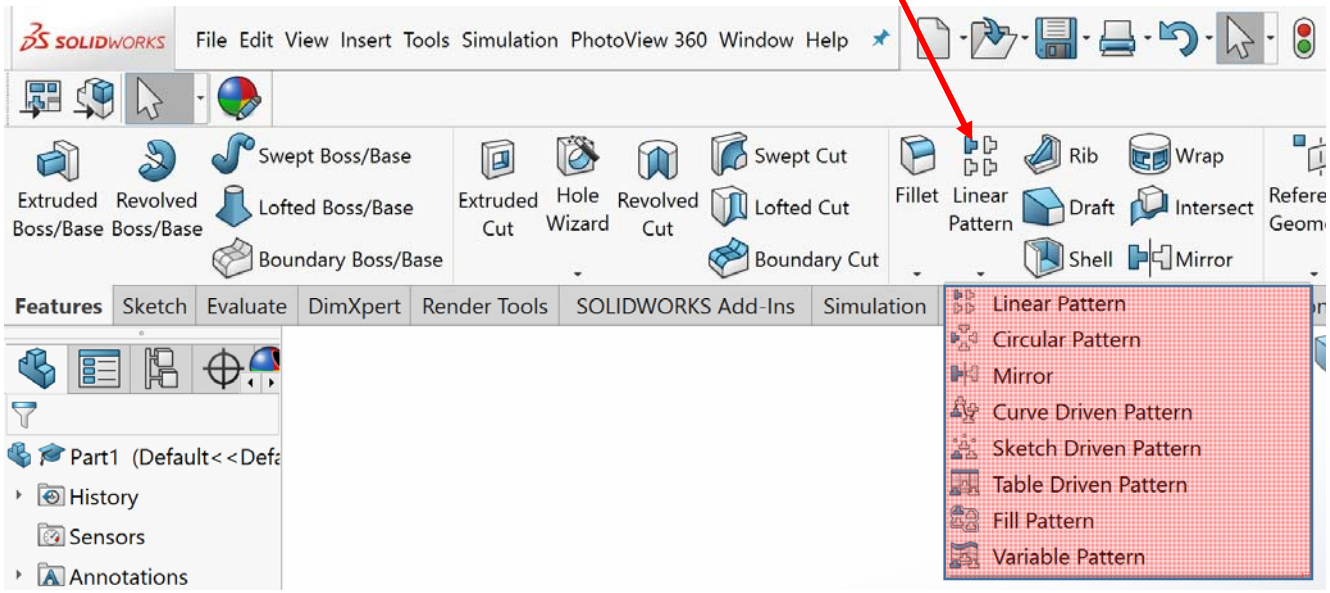




Applied Feature: Patterns



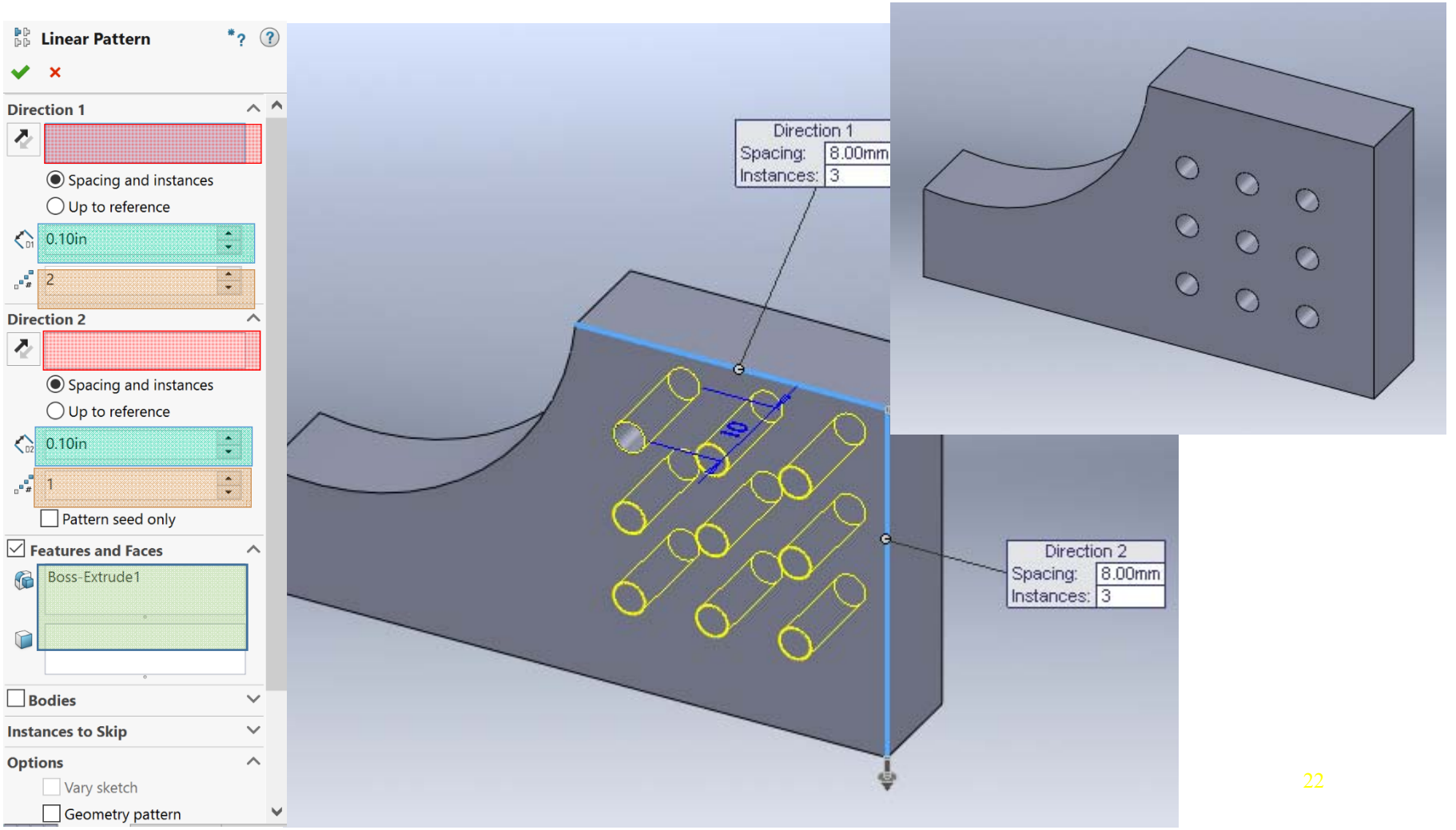
Linear (rectangular) pattern



Applied Features - Patterns

Linear (rectangular) pattern

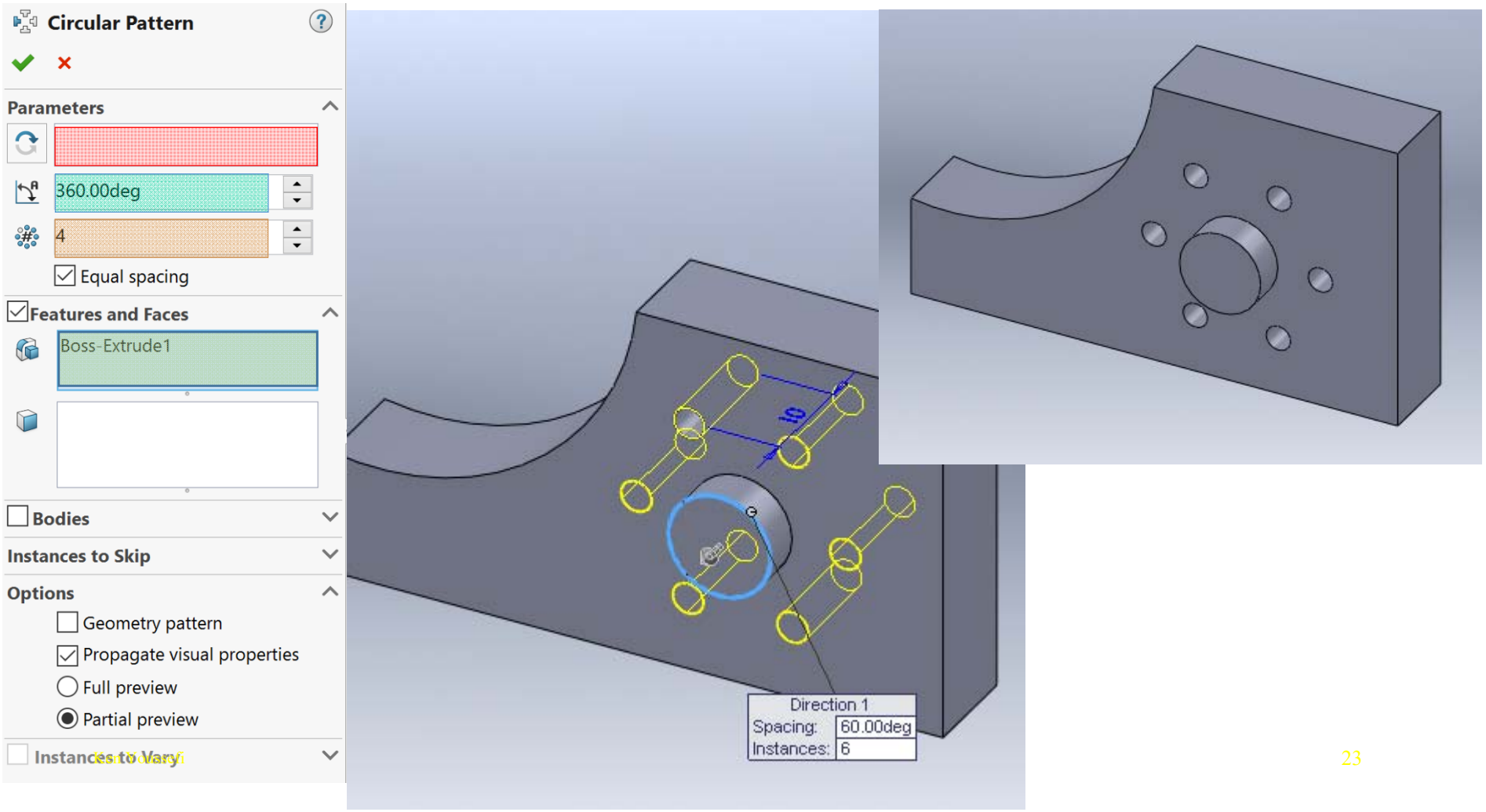
1. Select direction 1 and 2
2. Select spacing in dir. 1 and 2
3. Select # of features in dir. 1 and 2
4. Select feature to pattern



Applied Features - Patterns

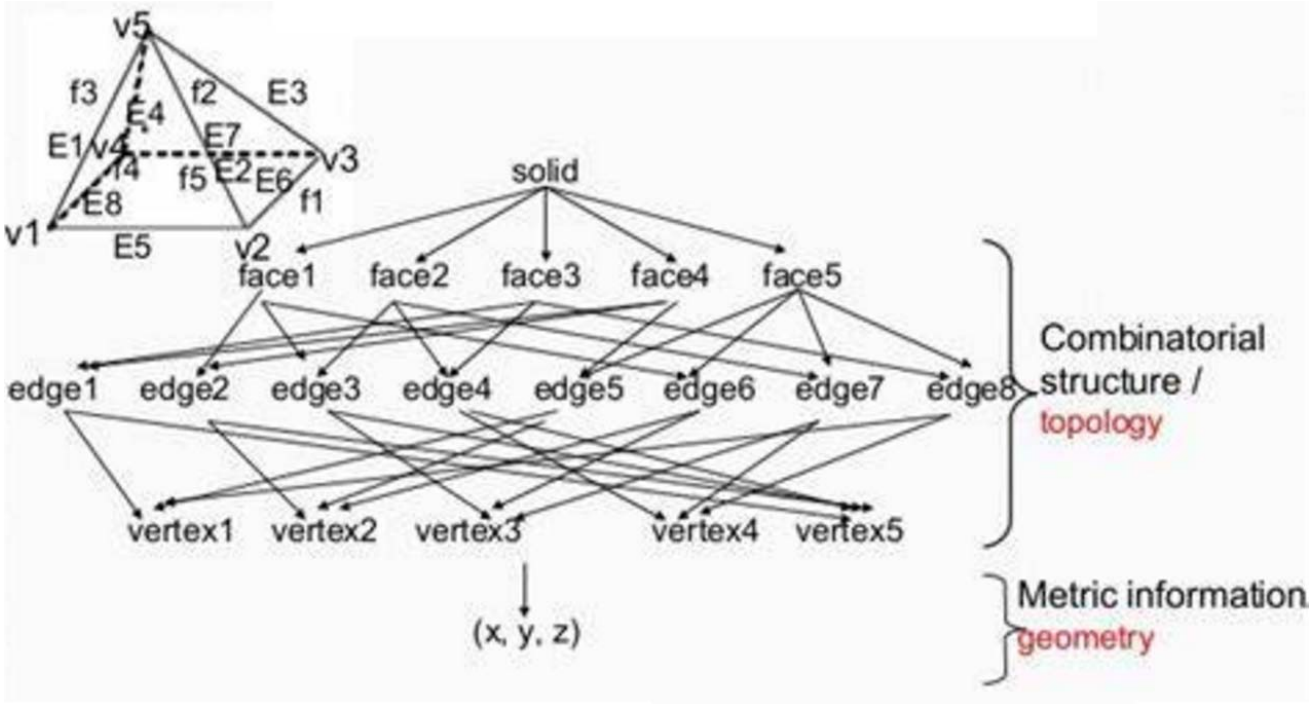
Circular (polar) pattern

1. Select axis of rotation
2. Select spacing between features
3. Select # of features
4. Select feature to pattern



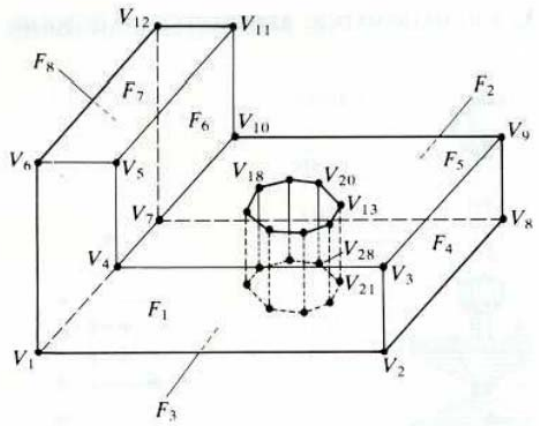
Boundary Representation (B-Rep) Modeling

- BREP defines an object by their spatial boundaries.
  - Topology: faces, edges, and vertices.
  - Geometry: surfaces, curves, and points.
- Data structure: 0bject list, surface list, polygon list, boundary list, line list, point list...

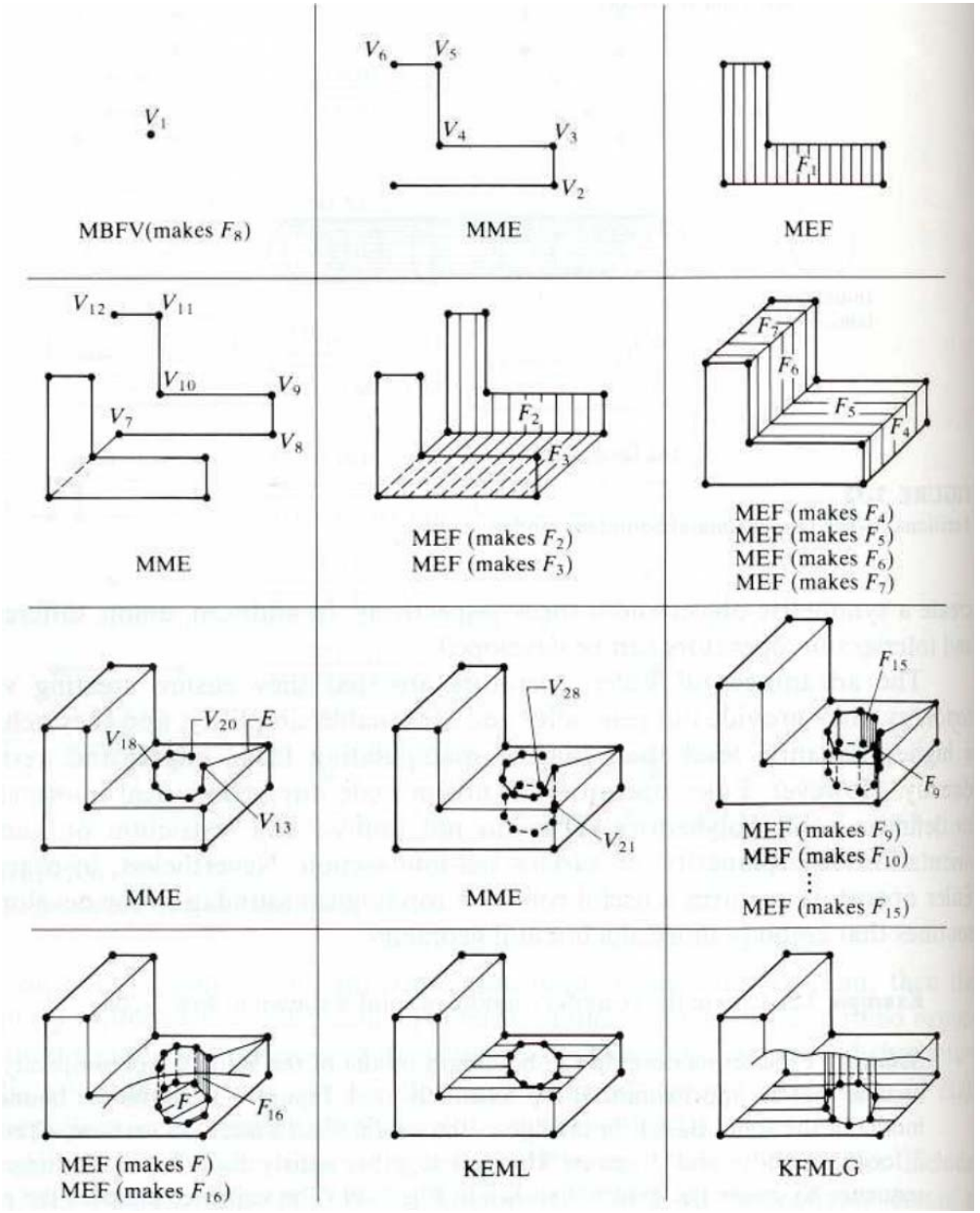


B-Rep

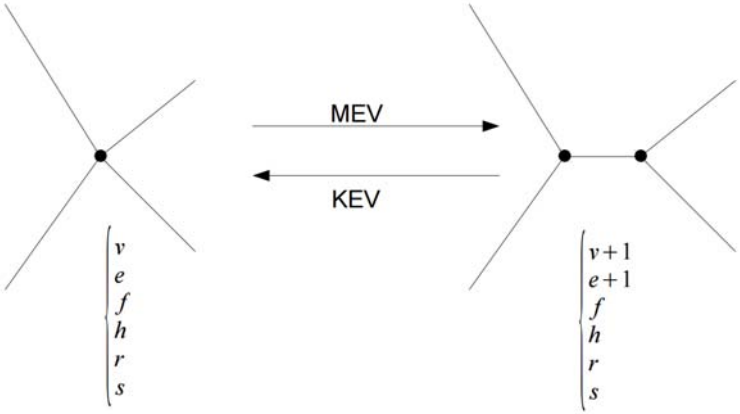
- Euler operators
  - MBFV: Make body, face, vertex
  - MME: Make multi edges
  - MEF: Make edge, face
  - KEML: Kill edge, make loop
  - KFMLG: Kill face, make loop, genus
- Validity Check: V – E + F = 2

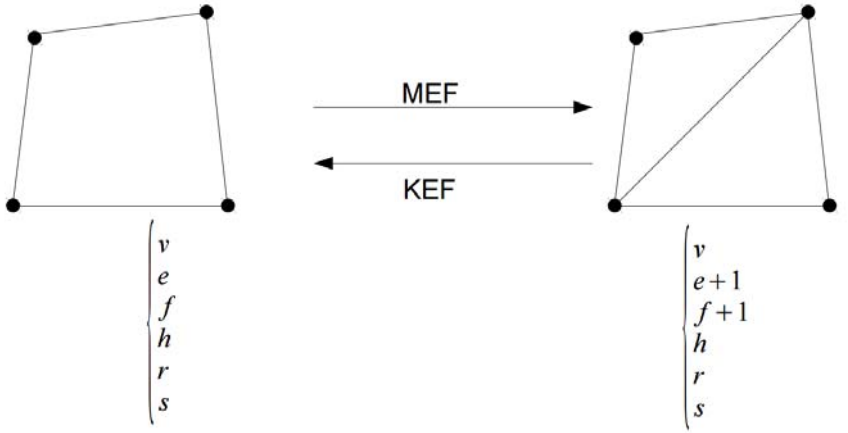


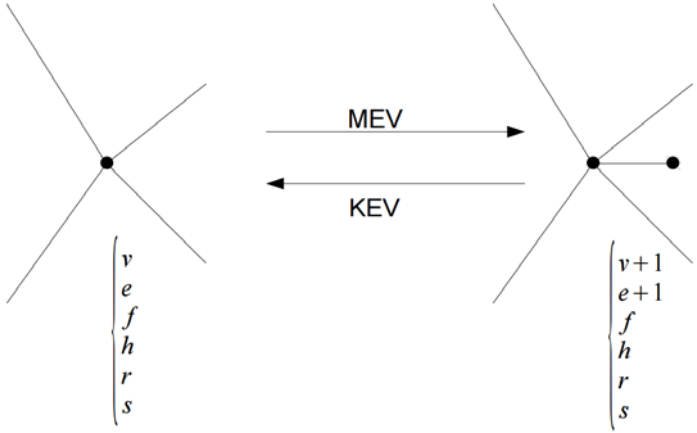
Body, Face, Polygon
(Edge Loop), Edge, Vertex

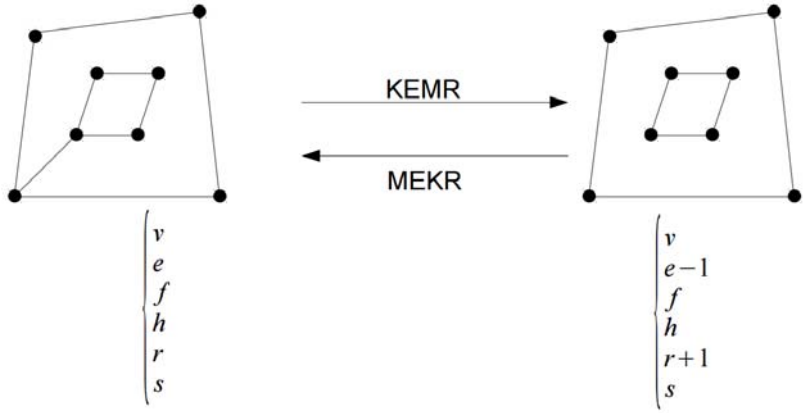


Examples of Euler Operators



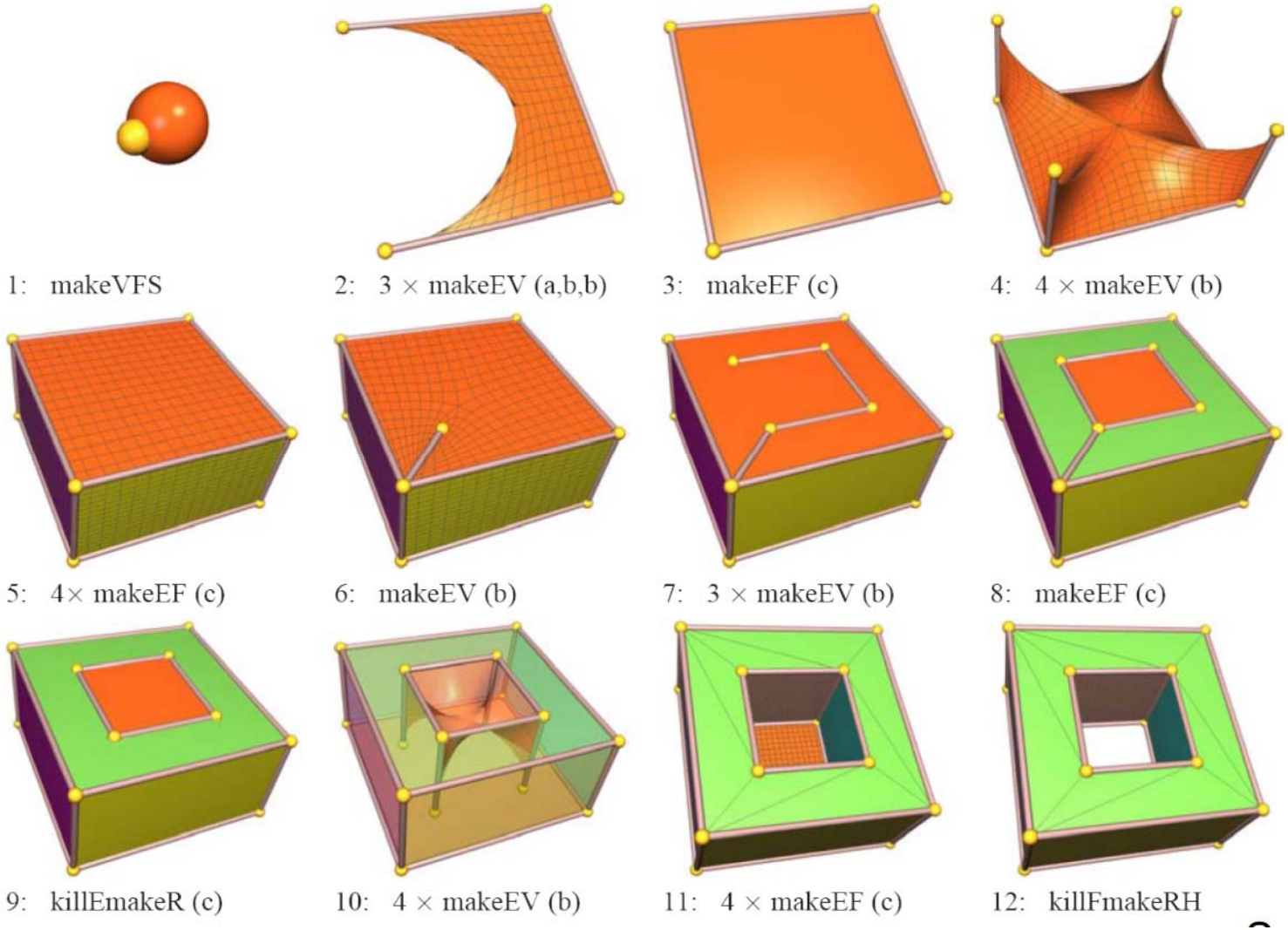






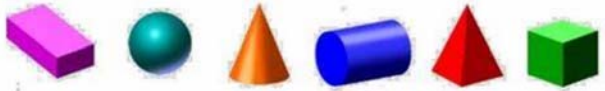
http://www.cgeo.ulg.ac.be/CAO/CAD_07.pdf

Modeling Example of using Euler operators

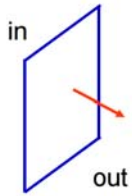


Constructive Solid Geometry (CSG) Modeling

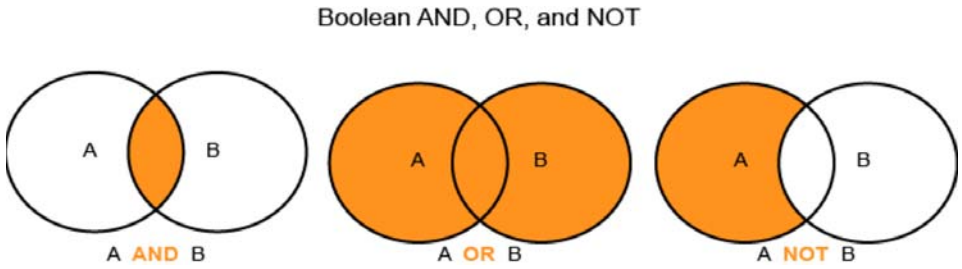
- Predefined primitive objects: cuboids, cylinders, prisms, pyramids, spheres, and cones.

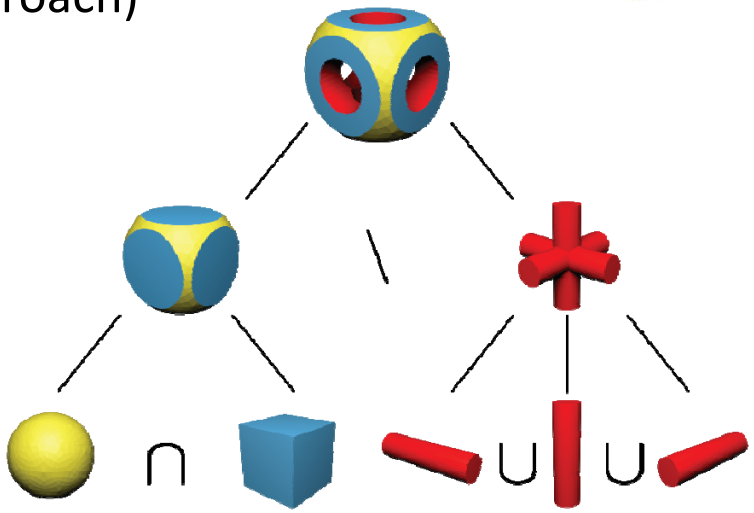


- Boolean operations: union, intersection, difference (subtraction)
- Two types
  - Primitive based CSG: It is a popular CSG scheme which is based on bounded solid primitives, R-sets.
  - Half-space based CSG: This CSG scheme uses unbounded **half-spaces**.

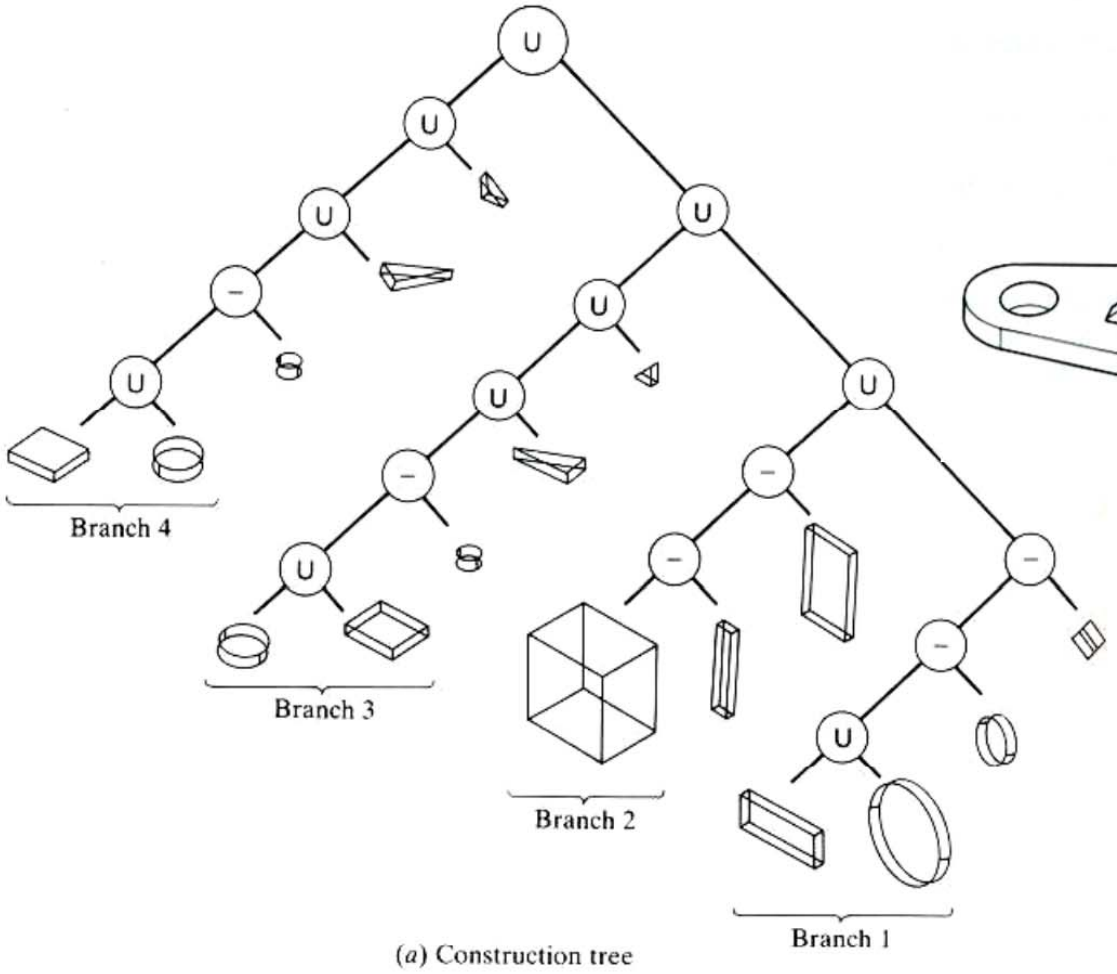


- CSG tree structure (building process/approach)

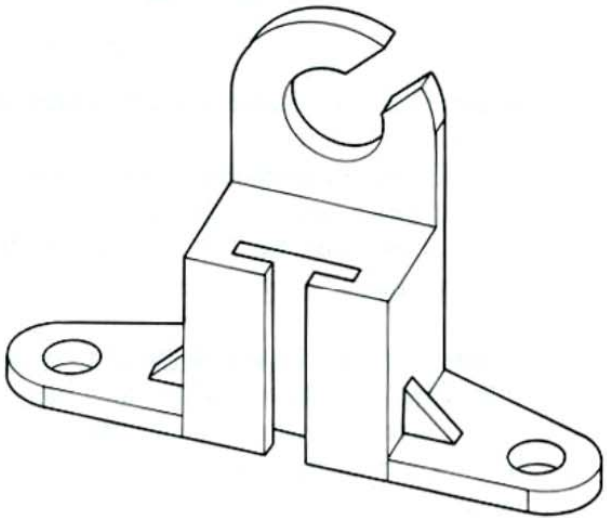




A CSG model



(*a*) Construction tree

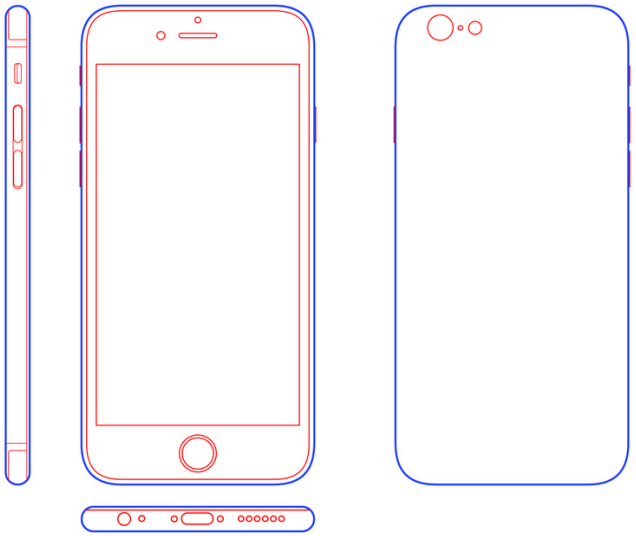


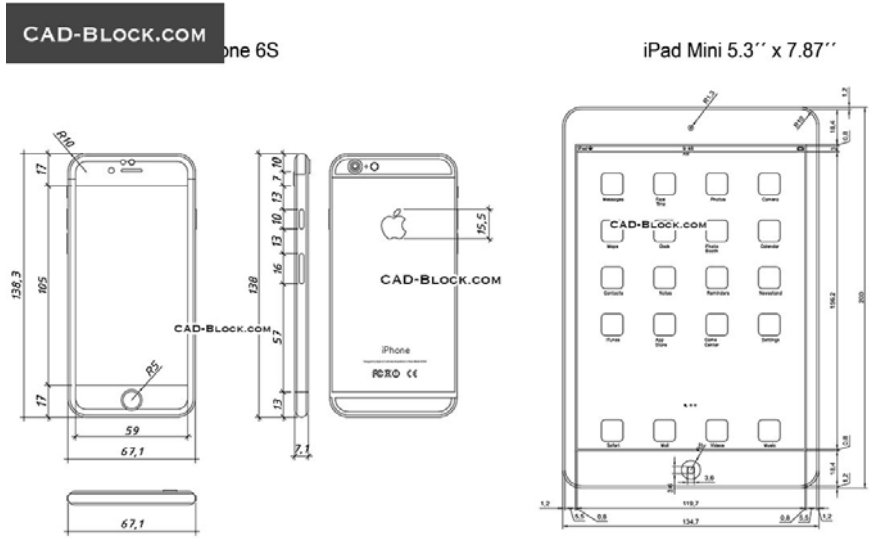
A lengthy, lousy process

B-Rep VS. CSG

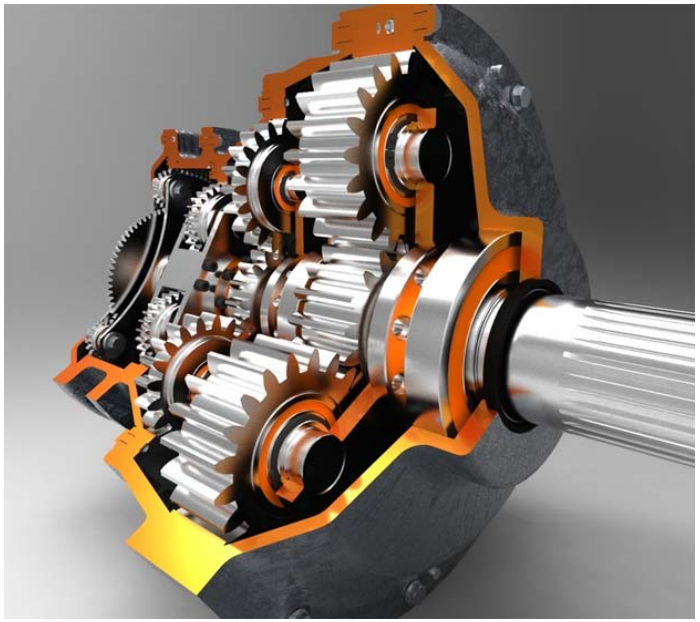
Boundary Representation (BREP)	Constructive Solid Geometry (CSG)
BREP describes only the oriented surface of a solid as a data structure composed of vertices, edges, and faces.	A solid is represented as a set of Boolean expression of primitive solid objects, of a simpler structure.
A BREP object is easily rendered on a graphic display system.	A CSG object is always valid in the sense that its surface is closed and orientable and encloses a volume, provided the primitives are authentic in this sense.
For B-rep, we review the possible surface types, the winged-edge representation schema, and the Euler operators.	For CSG, The basic operations include classifying points, curves, and surfaces concerning a solid; detecting redundancies in the representation; and approximating CSG objects systematically.

CAD Design Example

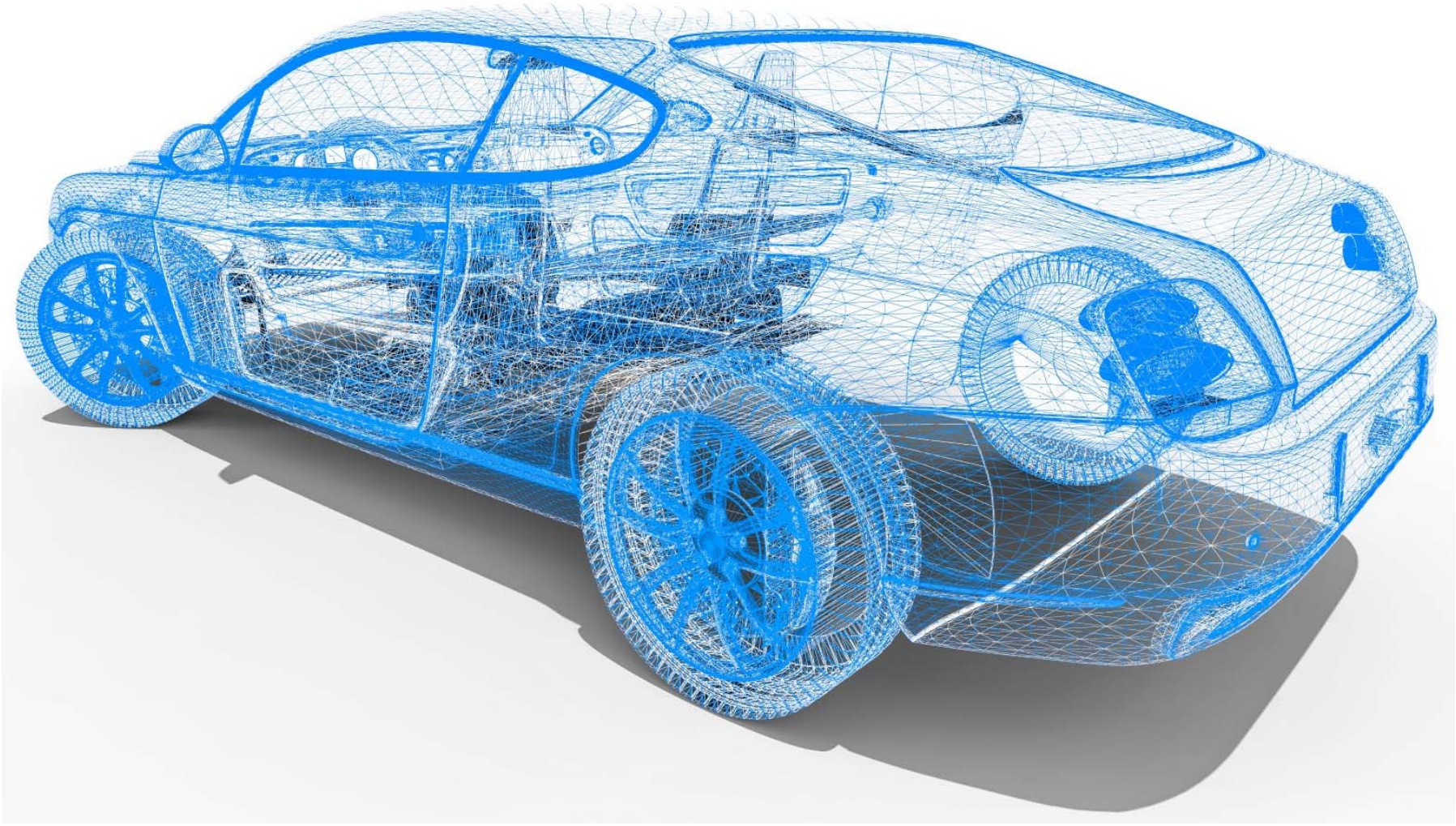




Examples of CAD models



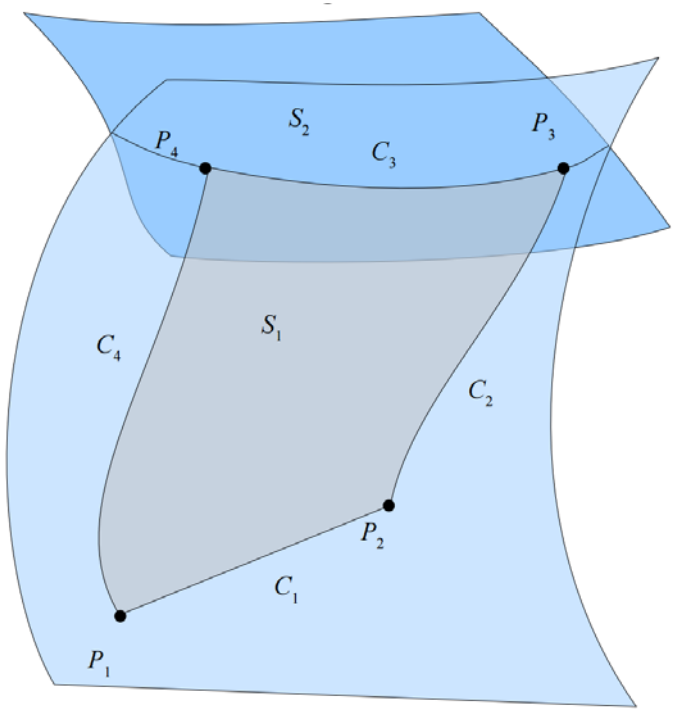
Examples of CAD models

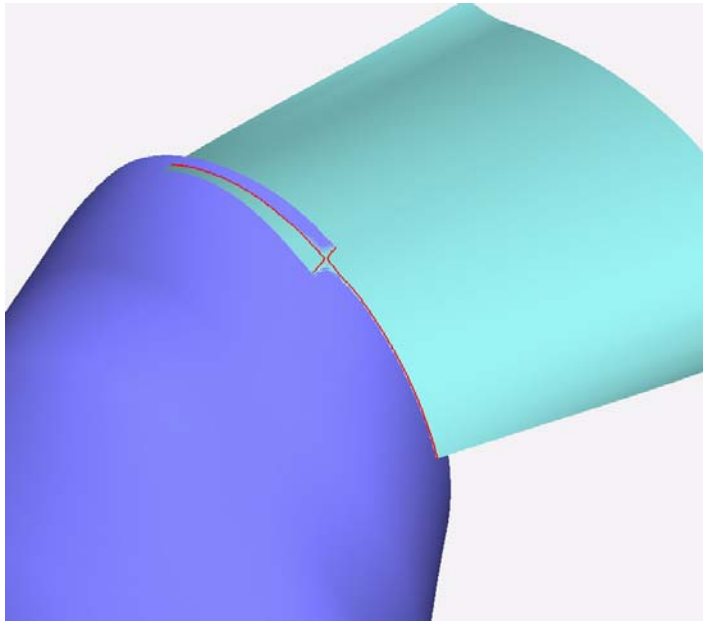


Examples of CAD models

Challenges in CAD systems

- Intersections & Topology robustness/precision
- Parametric design: non-linear optimization





从“中国制造”到“中国智造”

制造业是国民经济的主体，是立国之本、兴国之器、强国之基





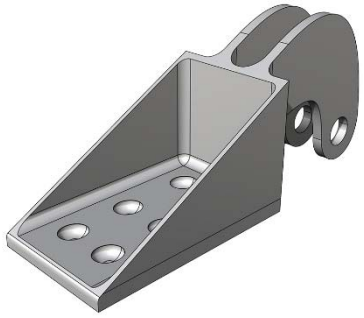
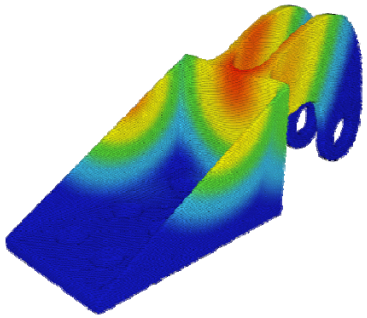
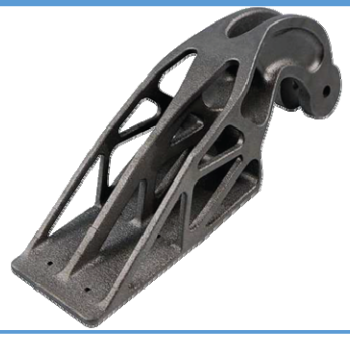
国务院2015：《中国制造2025》

- 设计顶层规划及战略目标
- 制定“三步走”路线图

- ✓ 第一步到2025年，迈入世界制造强国行列
- ✓ 第二步到2035年，达到世界制造强国中等水平
- ✓ 第三步到2049年，进入世界制造强国前列

挑战：国产自主工业设计软件

工业制造的核心要素是工业设计软件

计算机辅助设计(CAD)	计算机辅助分析(CAE)	计算机辅助制造(CAM)
		
CAD 软件 ➢ AutoCAD ➢ SolidWorks	CAE 软件 ➢ Ansys ➢ Abaqus	CAM 软件 ➢ MasterCAM ➢ Pro/NC
几何设计	性能分析	加工制造

- **立志：国产自主工业设计软件**
- **破解：“卡脖子”难题！**

中国科学技术大学

University of Science and Technology of China

GAMES102

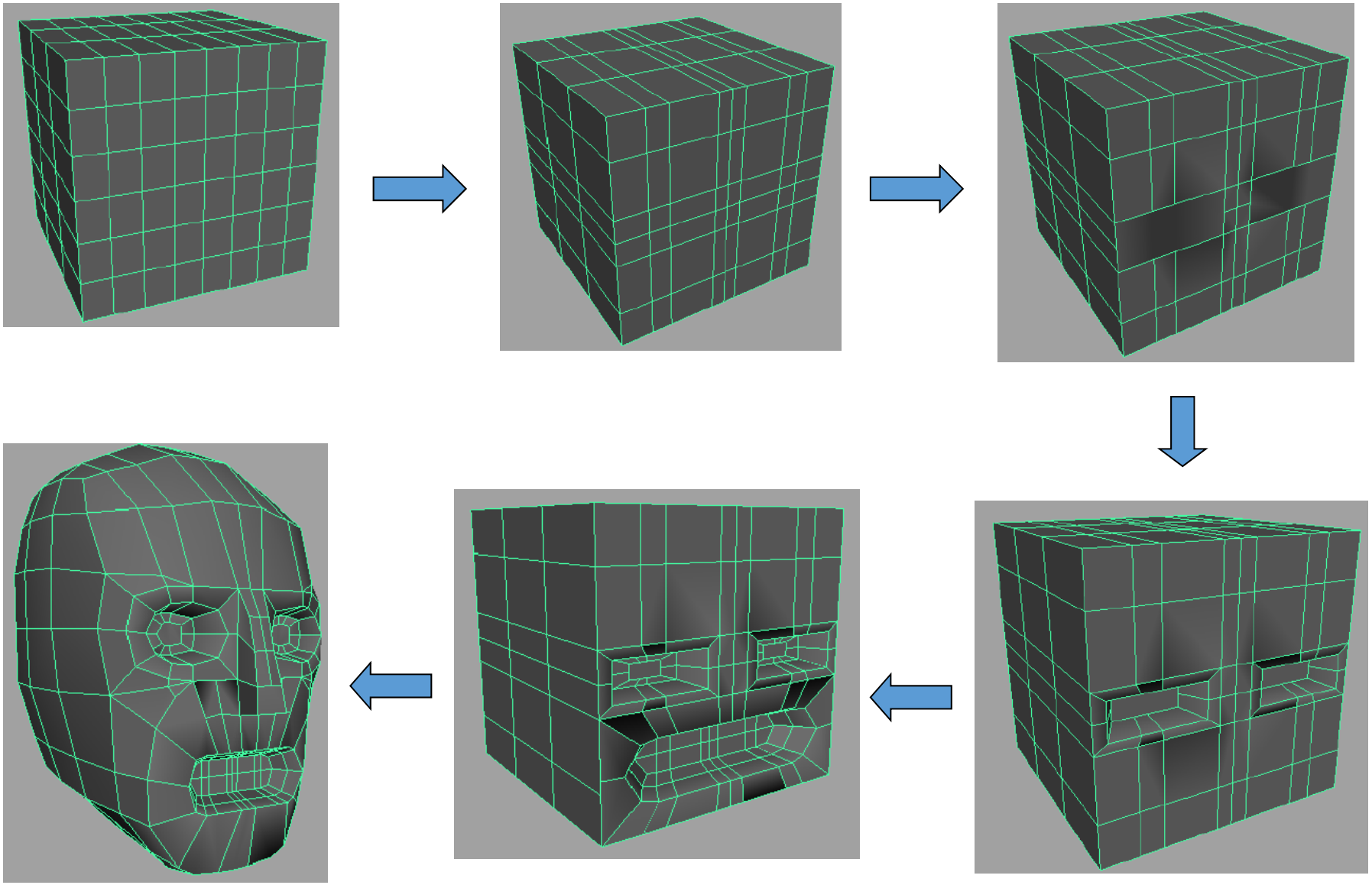
Surface Editing

Design Modeling

- Design from zero
  - Create a shape by a set of 3D design operators such as extrusion or revolution etc.

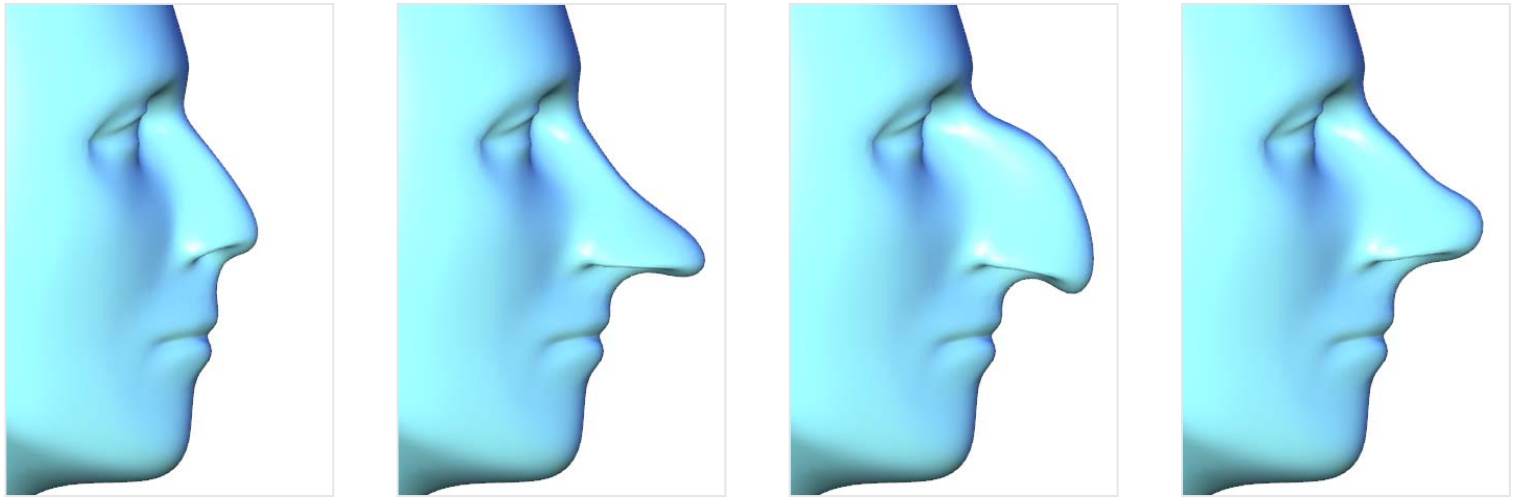
- Design from a given shape (Mesh editing)
  - Select a base shape
  - Select editing elements and editing operators
  - Deform the shape to obtain a new shape

Mesh Surface Editing



Interactive shape modeling

- Modeling is an interactive, iterative process
  - Tools need to be intuitive (interface and outcome)
  - Allow quick experimentation
  - Preserve shape properties

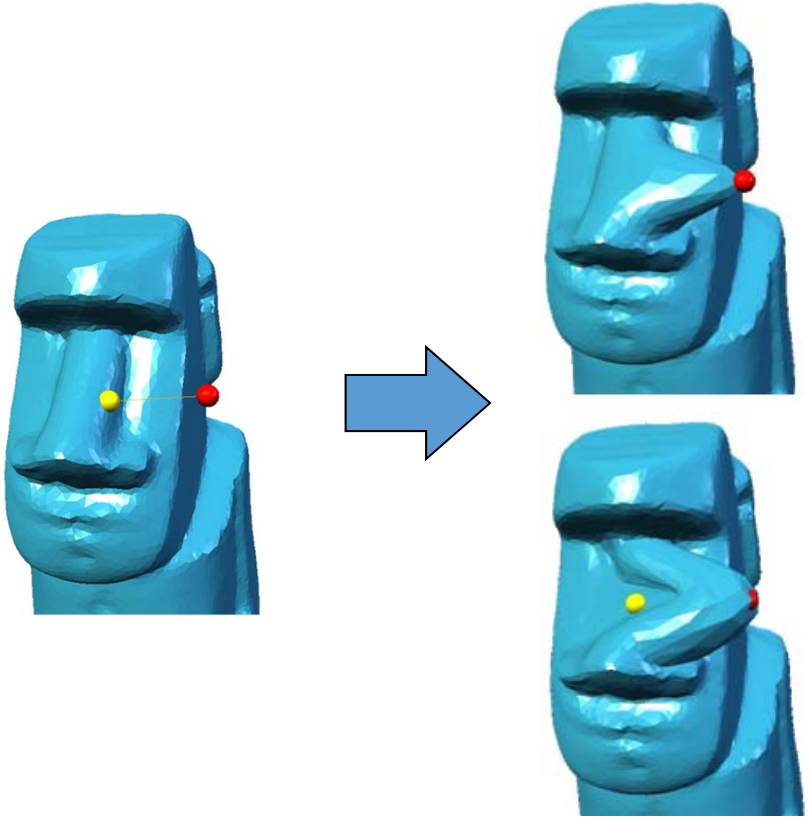


User Interaction (UI)

- User specifications: Intuitive, easy-to-use
  - Dragging handles -- vertices, features, ROI, …
- Deformed shape: adapt to user intents

Challenges:

- Operations
- Preservation of properties
- Semantics



Methodology

- Set a proxy P for the input shape S
  - P is simpler/easier/more semantic than S
  - Find a map between P and S: S=*g*(P)
    - *g* is also called an embedded space
- User interacts and deforms the proxy P to P’
  - Find a deformation map: P’=*h*(P)
- Compute a new shape S’ from P’: S’=*g*(P’)

Key Problem:

- find a good map *g*

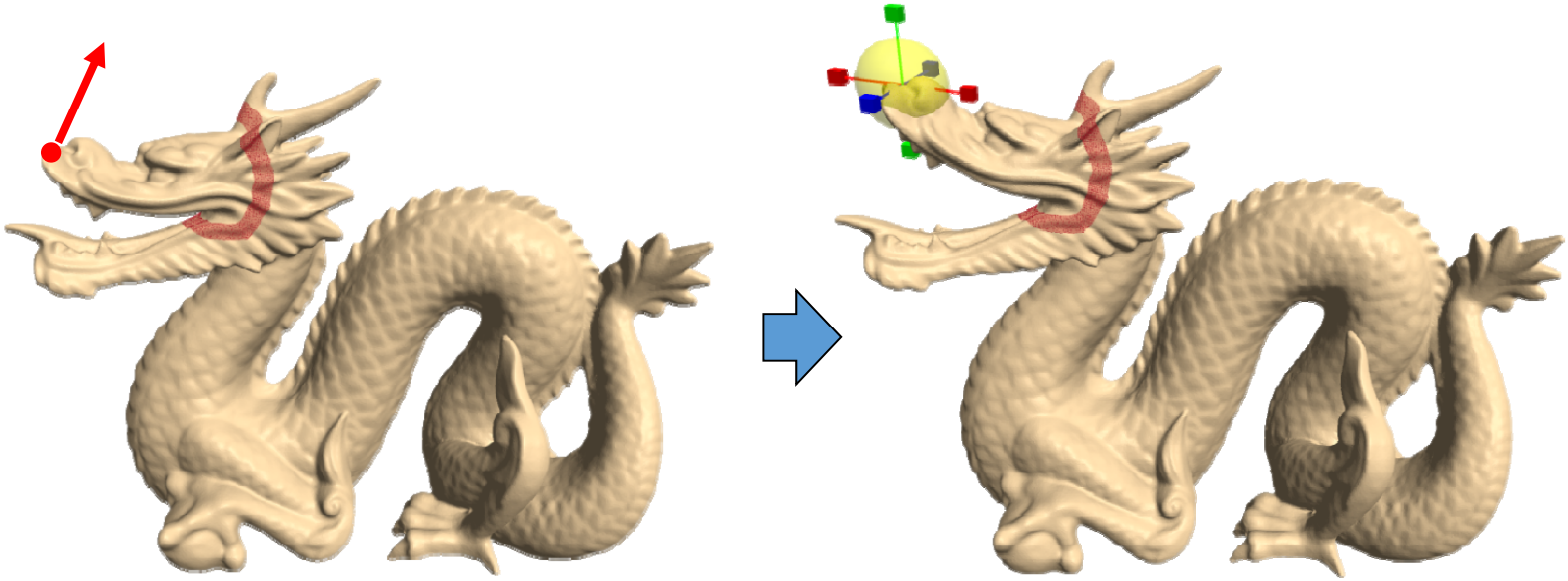
Proxy Shapes (handles)

- Points
  - Points, vertices, …
- Lines/Curves
  - Sketches, skeletons, silhouettes, wires, …
- Meshes
  - Bezier nets, lattices, cages, …
- Other shapes
  - Deformation transfer: learning deformation from other shapes

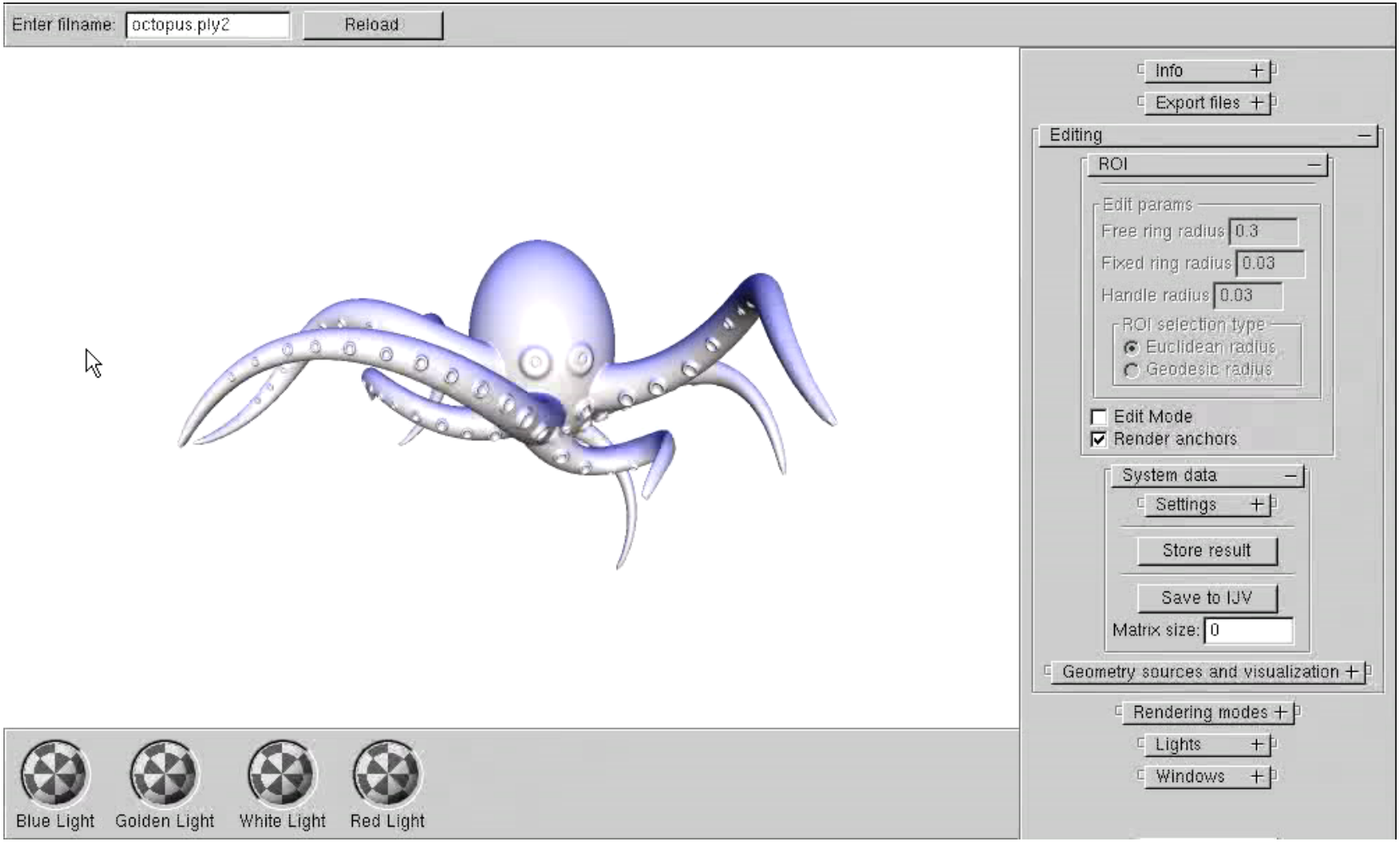
1. Point Proxy

Point based editing

- Fix some vertices
- Drag one or more vertices

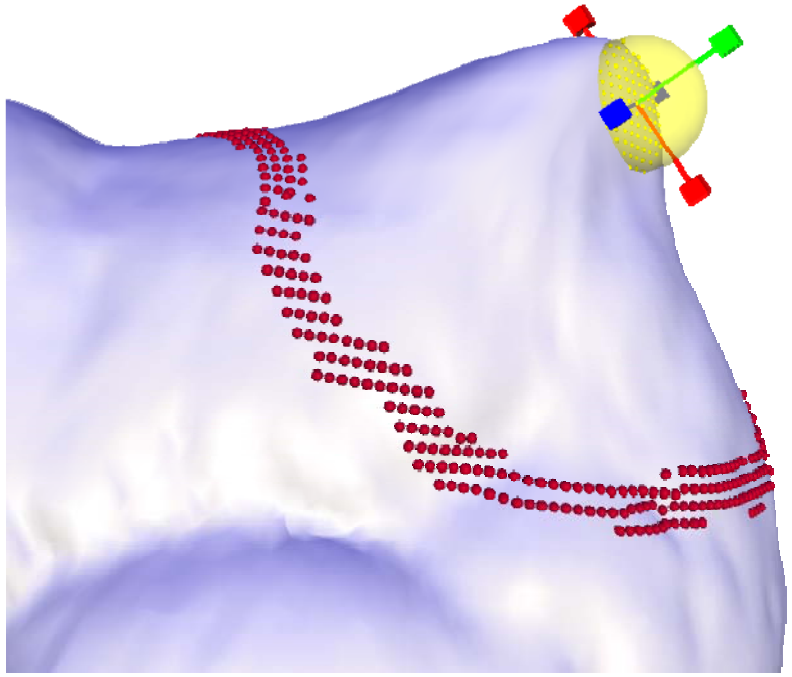


User Interfaces



User Interfaces

- ROI is bounded by a belt (static anchors)
- Manipulation through handle(s)

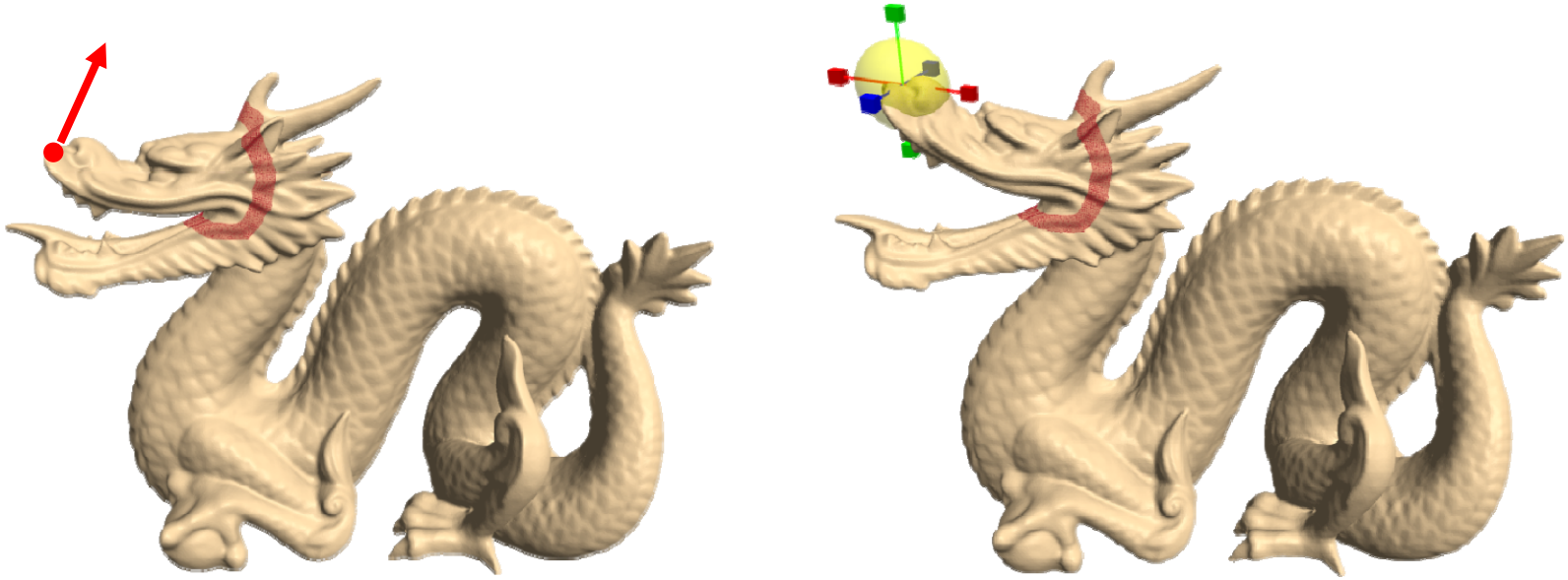


本质数学问题：数据插值问题

Interpolation Problem

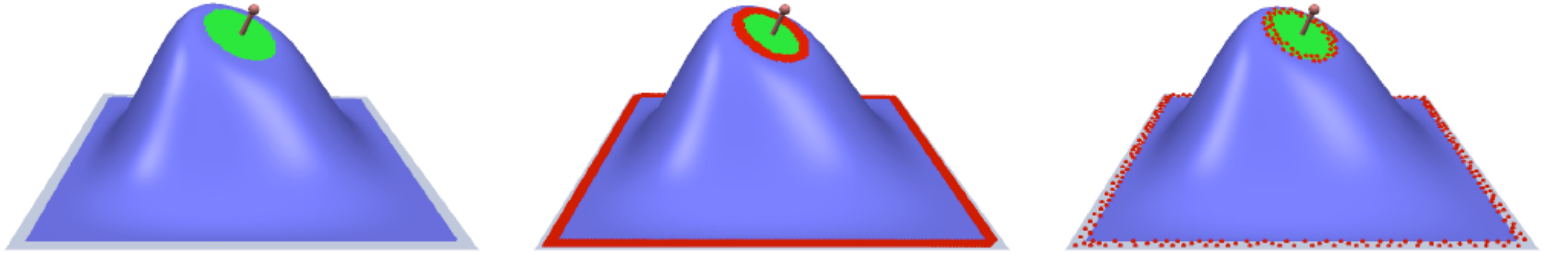
$$minE(f)$$

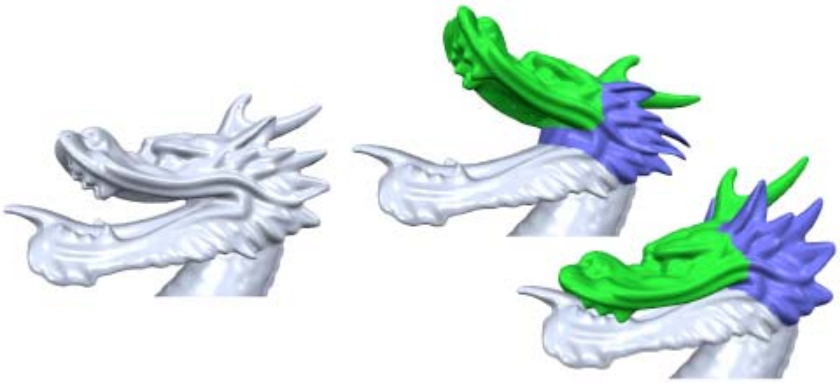
$$\text{s.t.} \quad f(x_i) = y_i, i = 1,2, \dots, n$$

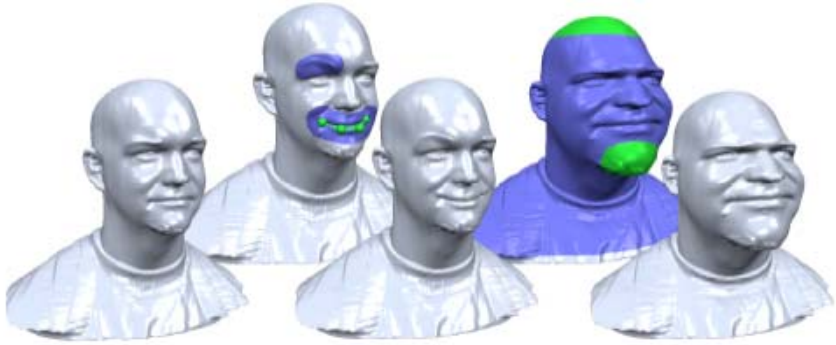


1.1 RBF-based Editing

[2005]

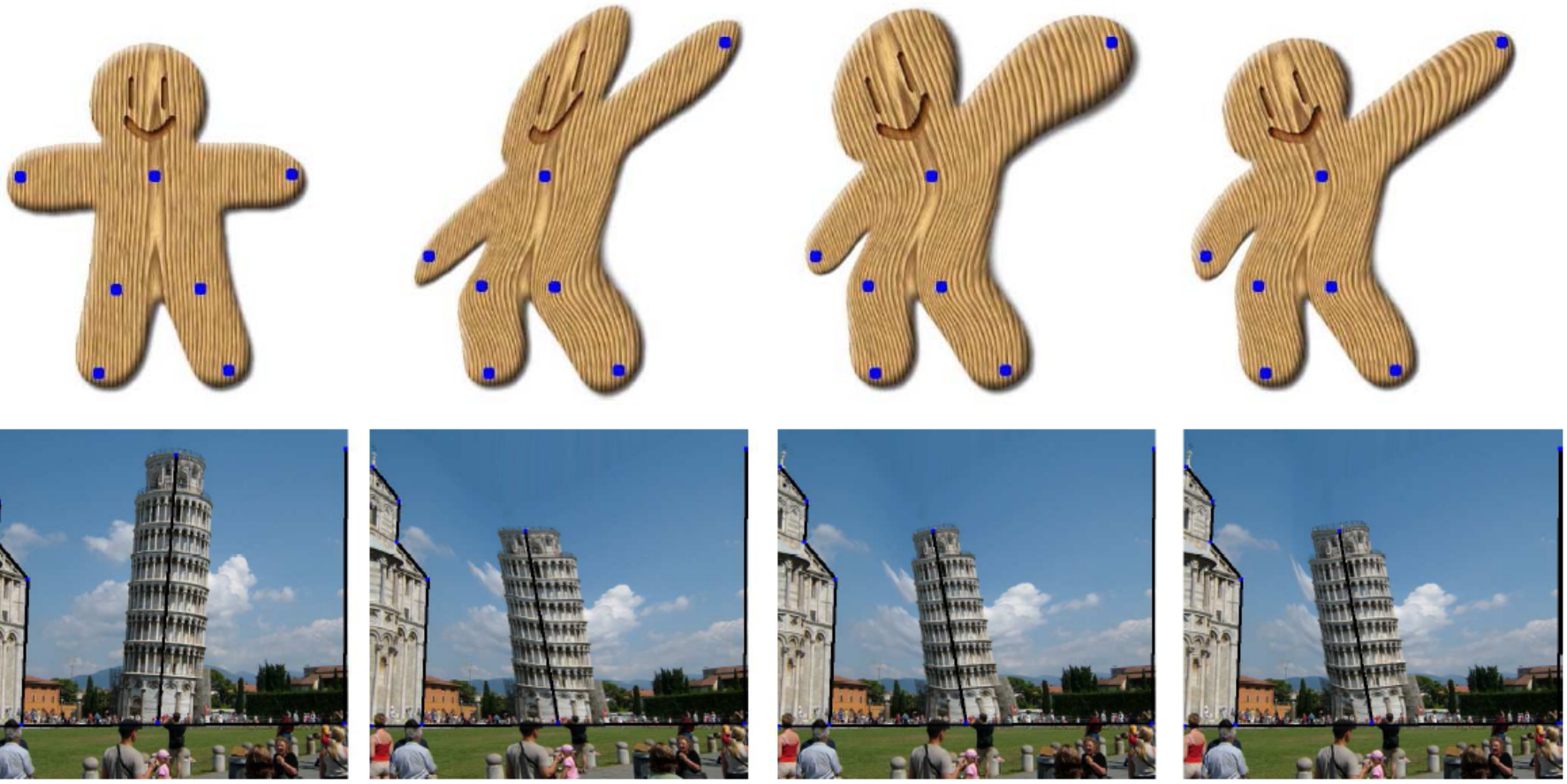






1.2 Moving Least Squares Method

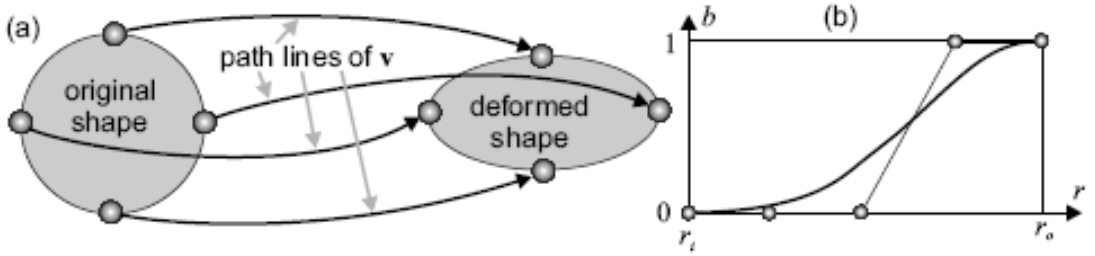
[Siggraph 2006]



1.3 Vector Field Based Deformations

[Siggraph 2006]

- Basic model: Moving vertex along the deformation orbit – defined by the path lines of a vector field v.



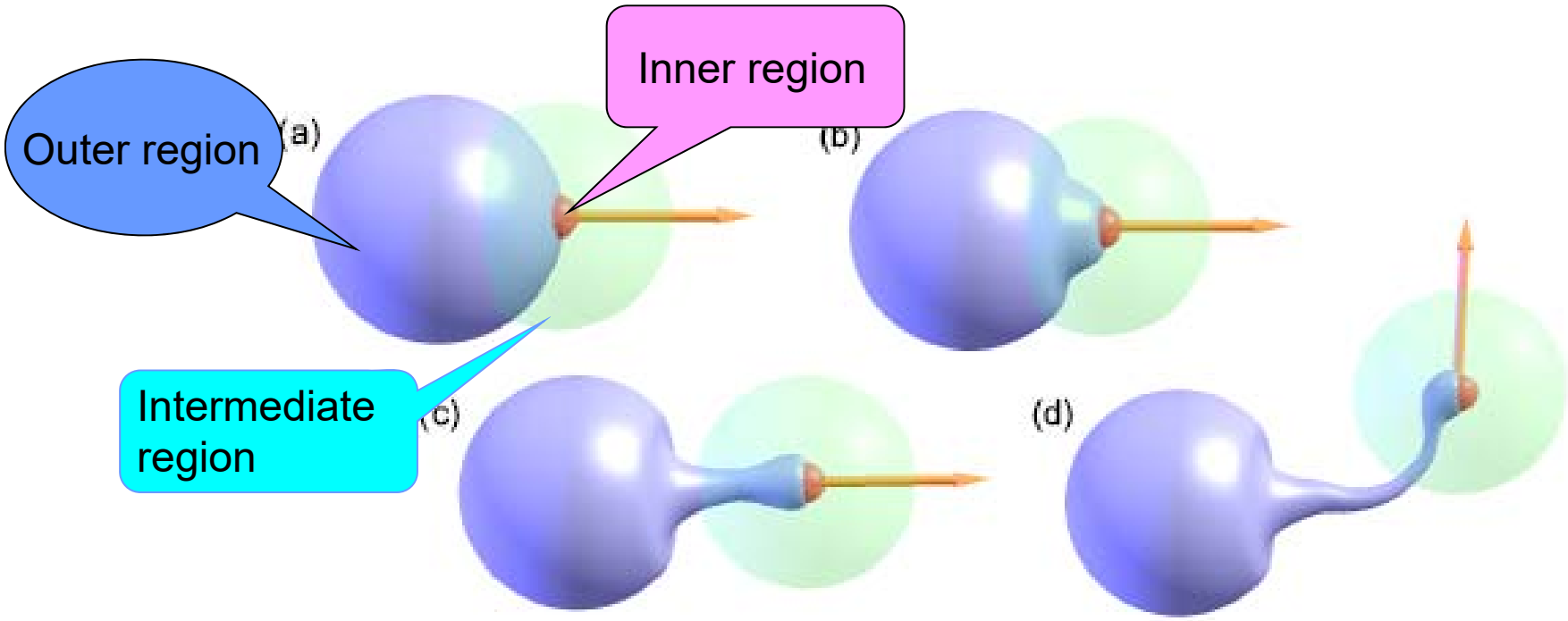
- Given a time-dependent vector field **V**(**X**, t), a *Path Line* **X**(t) in space is an integral.

Vector Field Selection

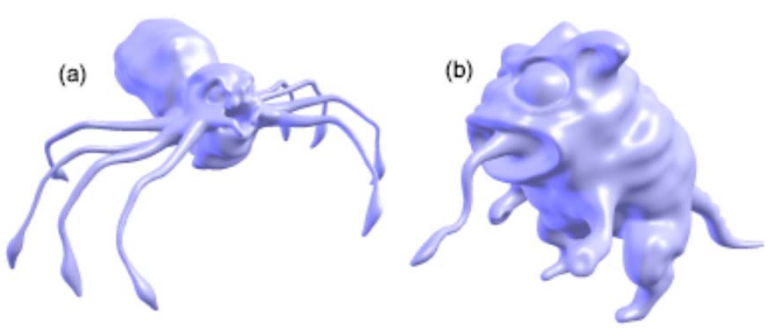
- Deformation Request:
  - No self-intersection
  - Volume-preserving
  - Details-preserving
  - Smoothness of shape in deformation
- Divergence-free Vector Field: $\boldsymbol{V}=(V_1, V_2, V_3)$

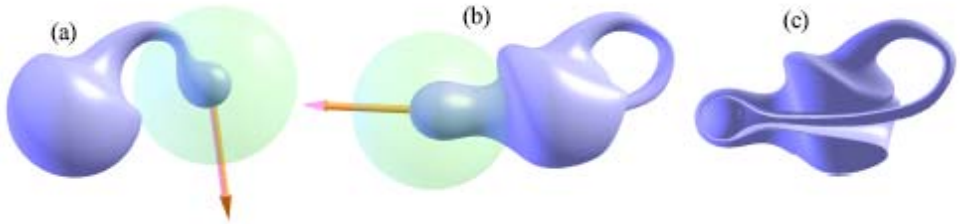
$$div\,\boldsymbol{V} = \frac{\partial V_1}{\partial x} + \frac{\partial V_2}{\partial y} + \frac{\partial V_3}{\partial z} = 0$$

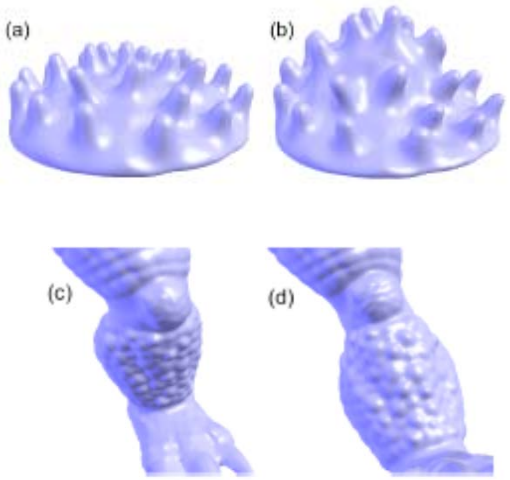
Piecewise Field for Deformation

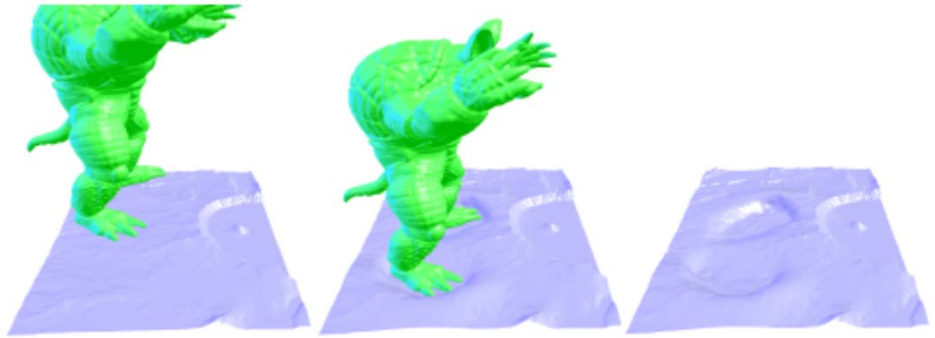


Examples





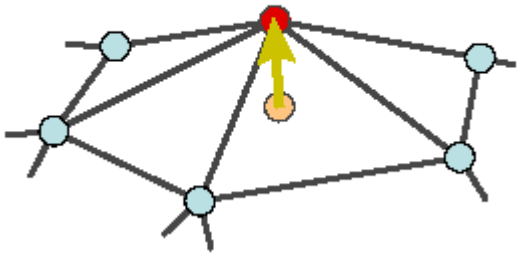


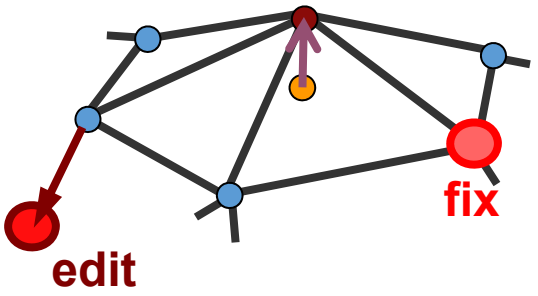


1.4 Laplacian Editing

[Sorkine et al. SGP 2004]

- Preserve local detail – Laplacian coordinates
- Representation with sparse matrices
- Efficient linear surface reconstruction





$$v_i = \sum_{j \in N(i)} w_j v_j + \delta_i$$

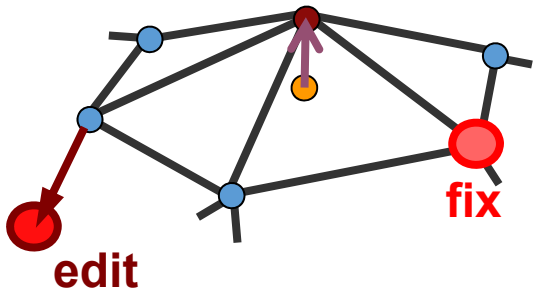
Variational Viewpoint

- Laplacian Approximation

$$\tilde{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \left(\|L\mathbf{x} - \delta^{(x)}\|^2 + \sum_{j \in C} \omega^2 \|x_j - c_j\|^2 \right).$$

- Gradient Approximation

$$\min_{\phi} \int \int_{\Omega} \|\nabla \phi - \mathbf{w}\|^2 dA,$$



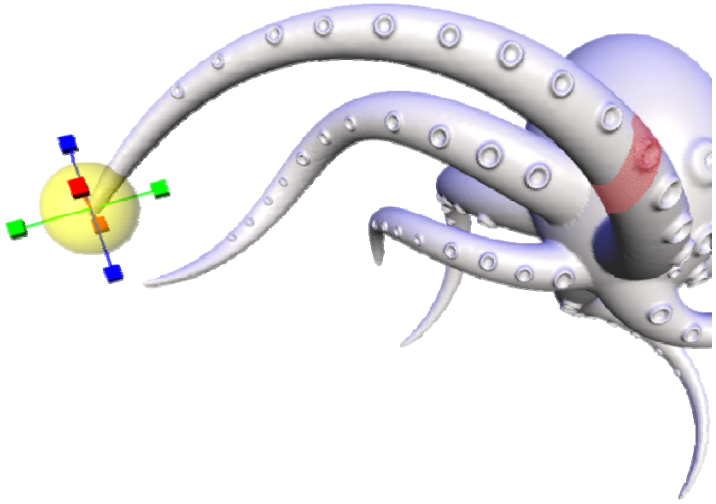
Editing framework

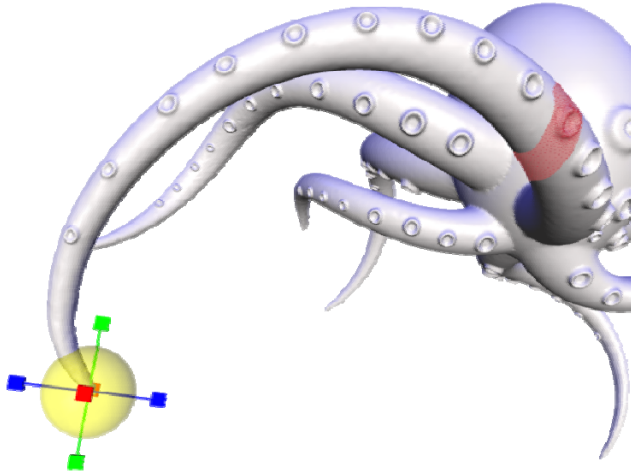
- The spatial constraints will serve as modeling constraints
- Reconstruct the surface every time the modeling constraints are changed

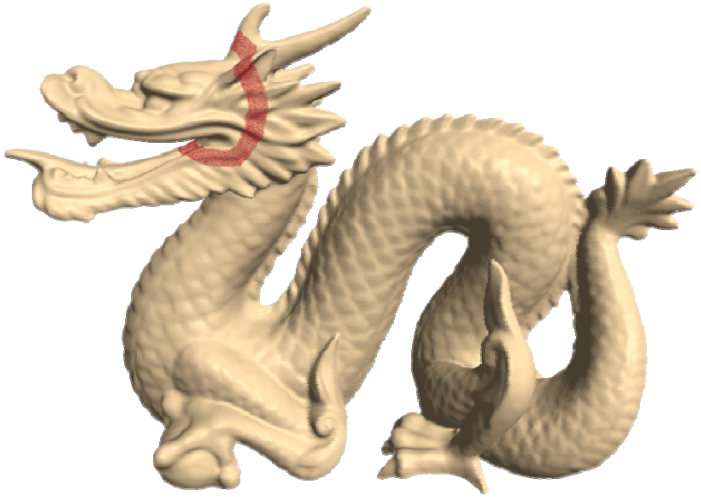
Detail constraints: $L\mathbf{x} = \boldsymbol{\delta}$

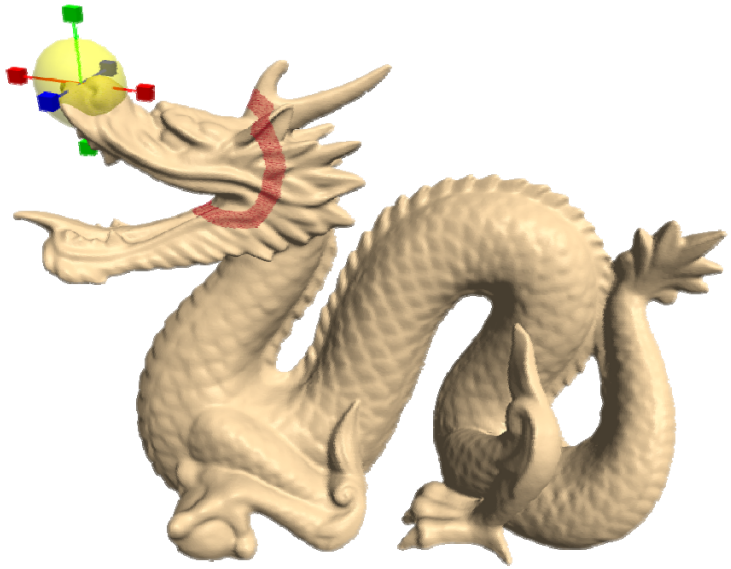
Modeling constraints: $x_j = c_j, \quad j \in \{j_1, j_2, \ldots j_k\}$

Results

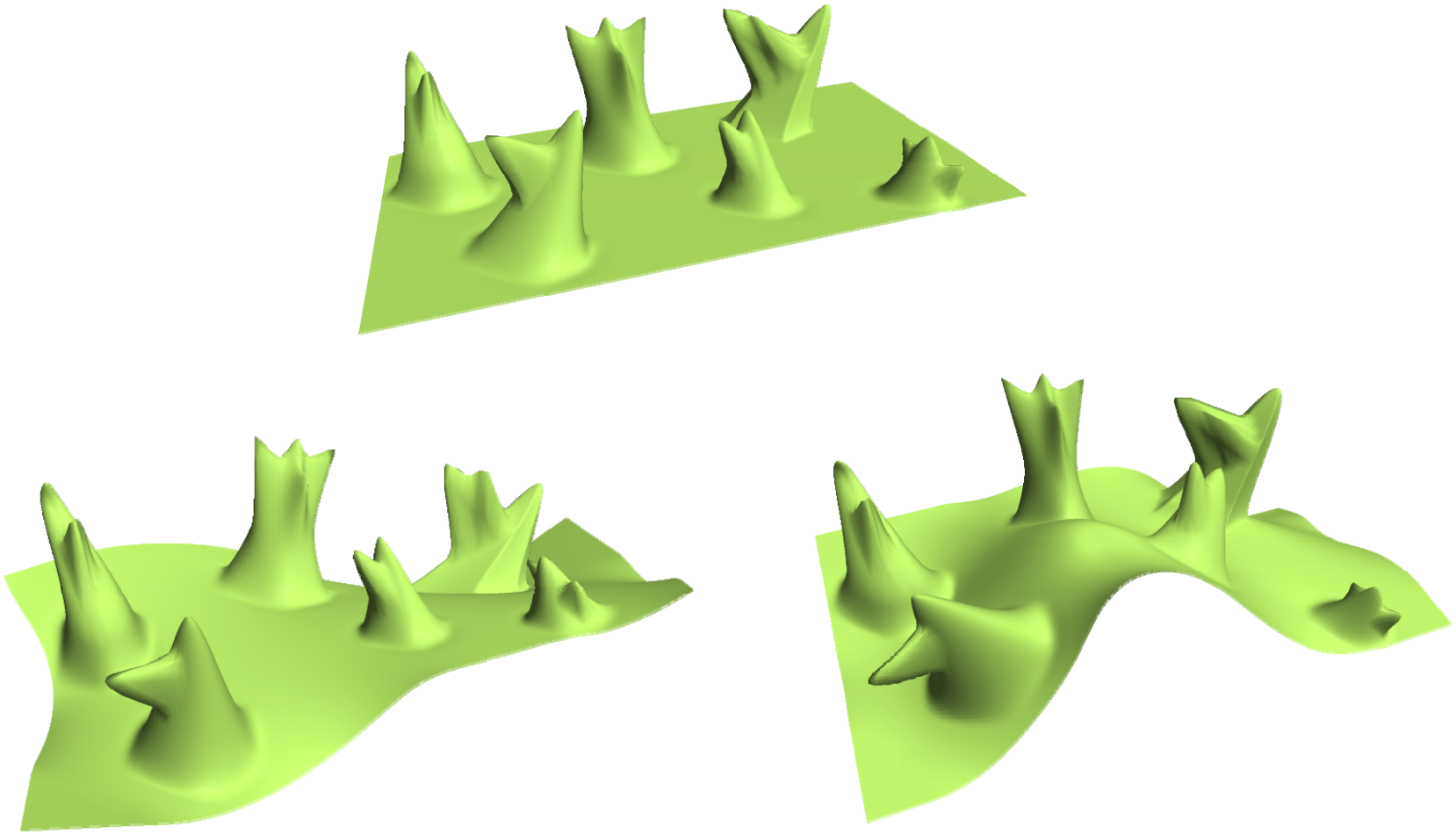






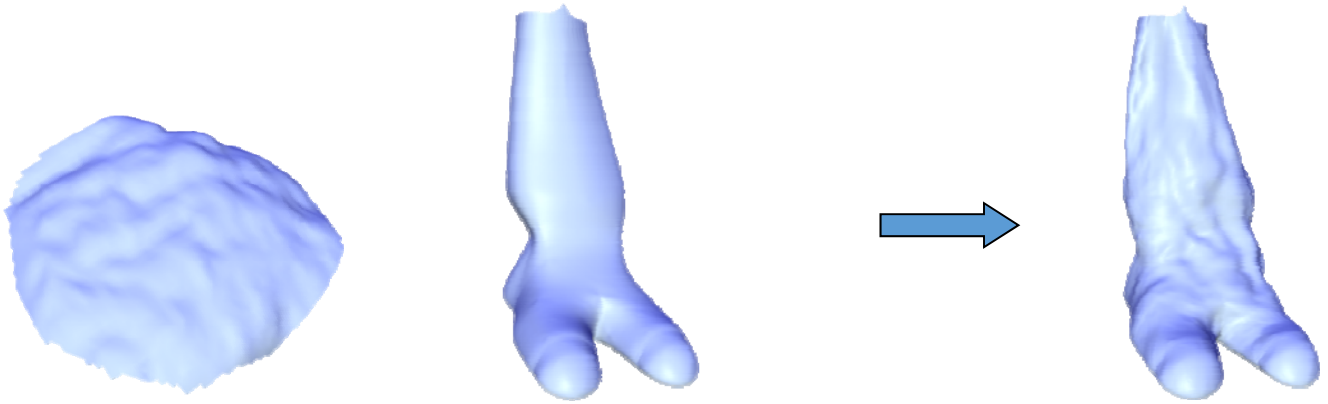


Results

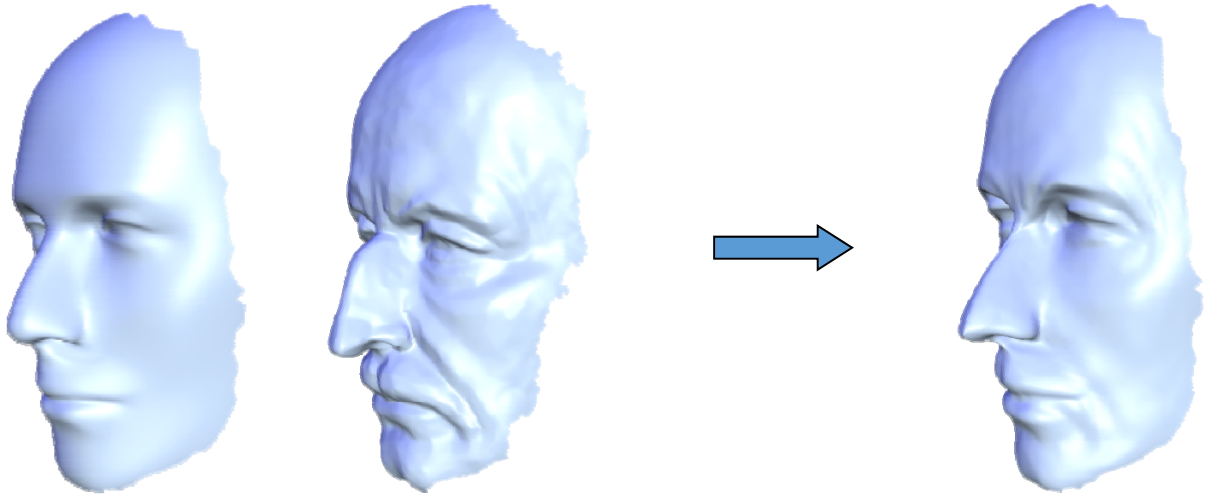


Detail transfer and mixing

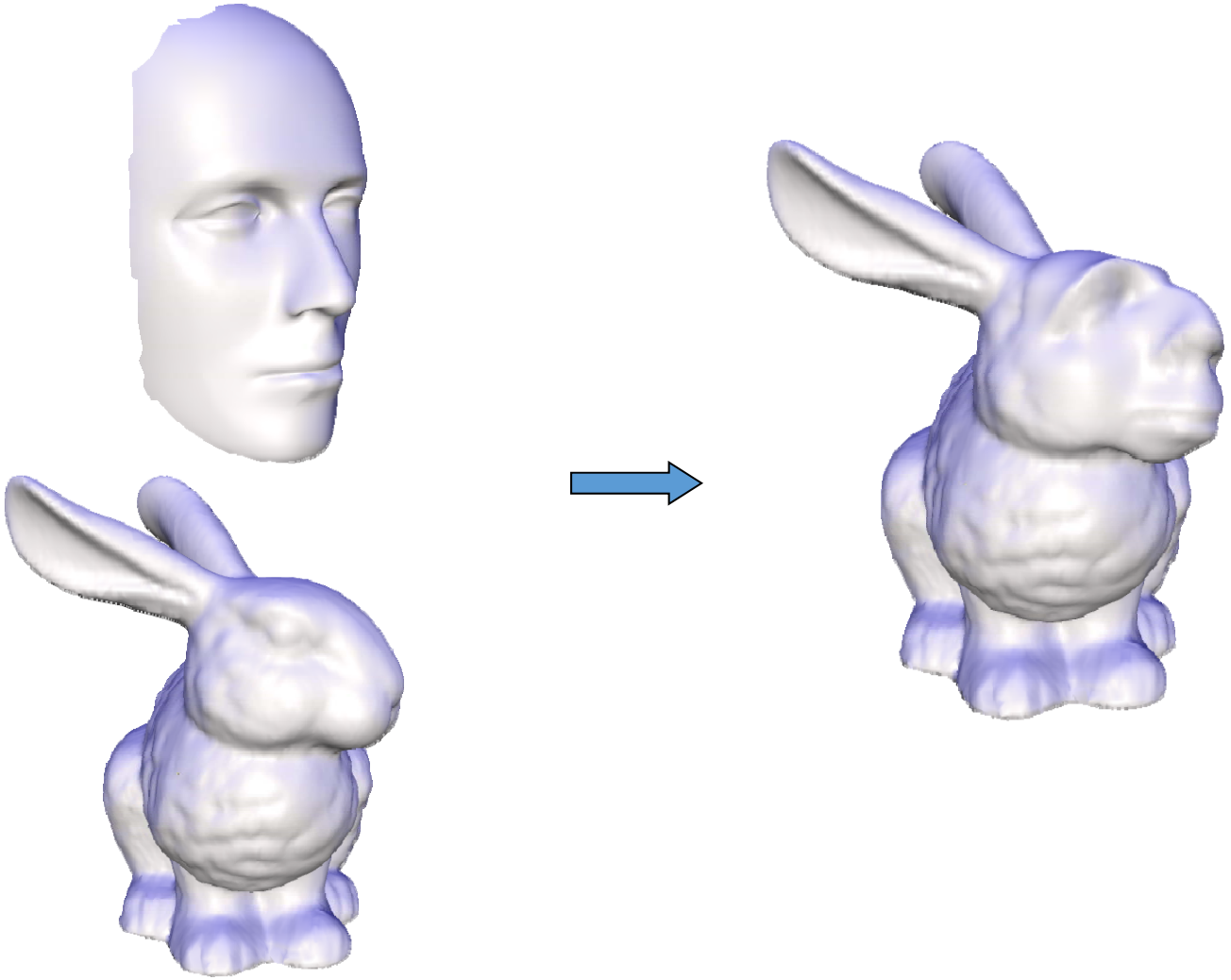
- “Peel“ the coating of one surface and transfer to another



Examples

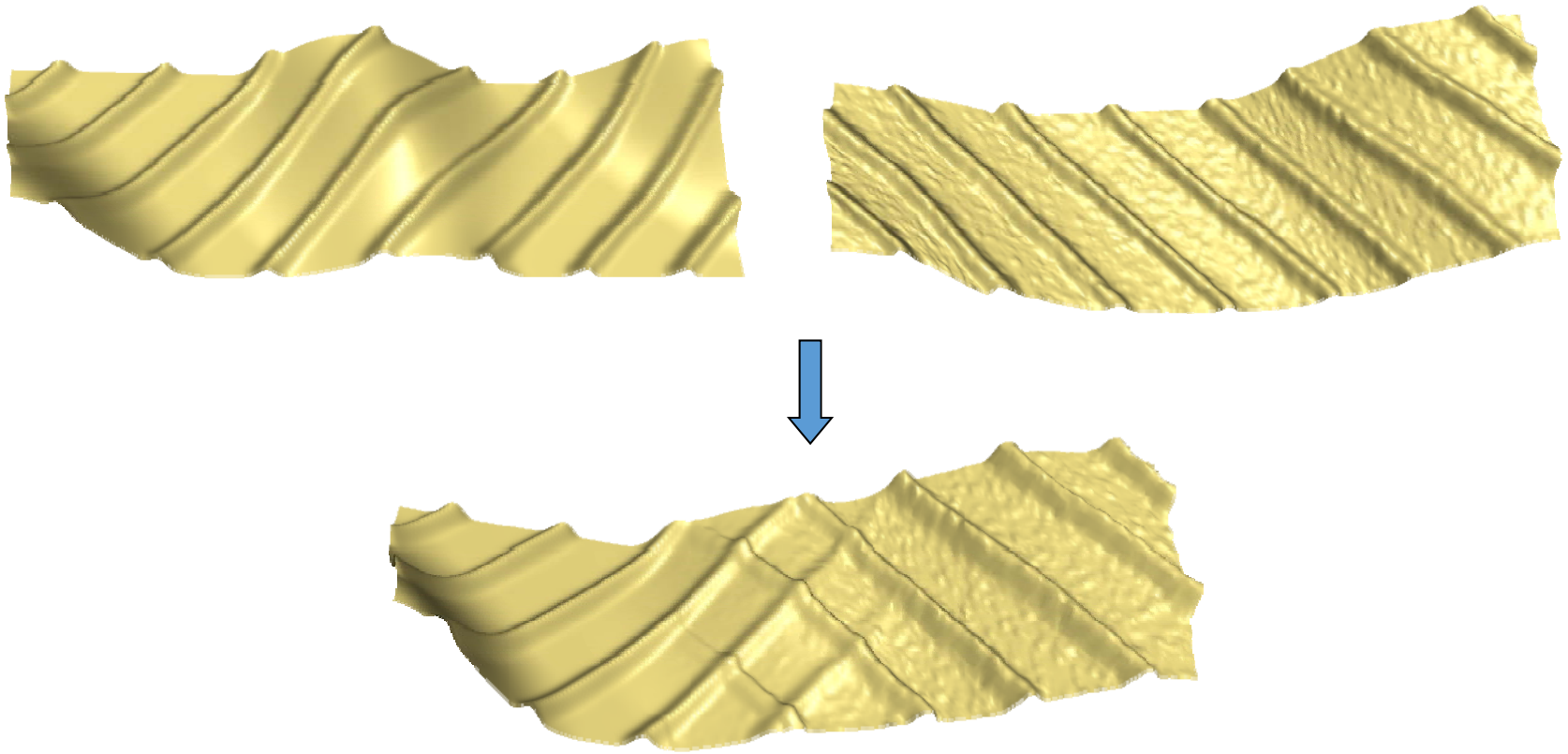


Examples



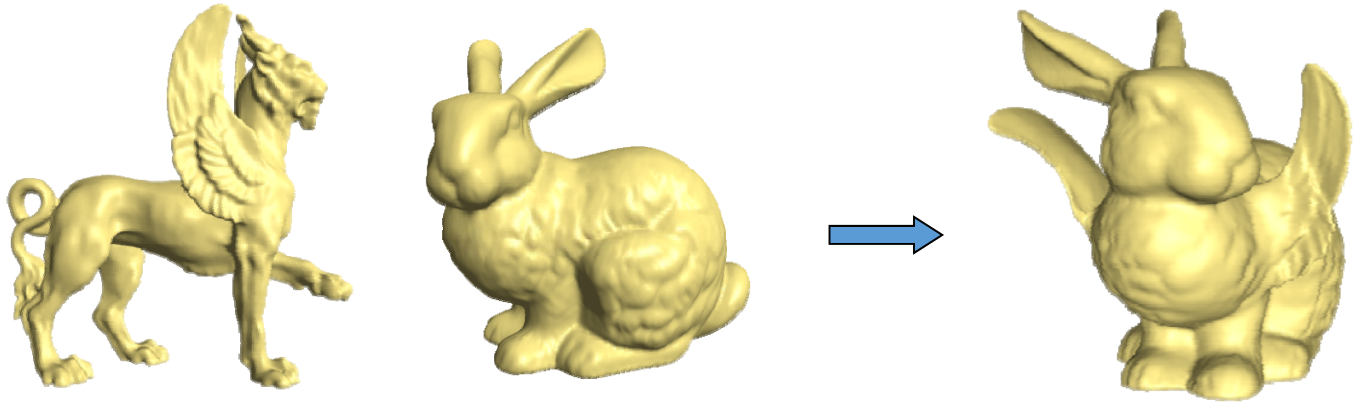
Mixing Laplacians

- Taking weighted average of δ_i and δ'_i



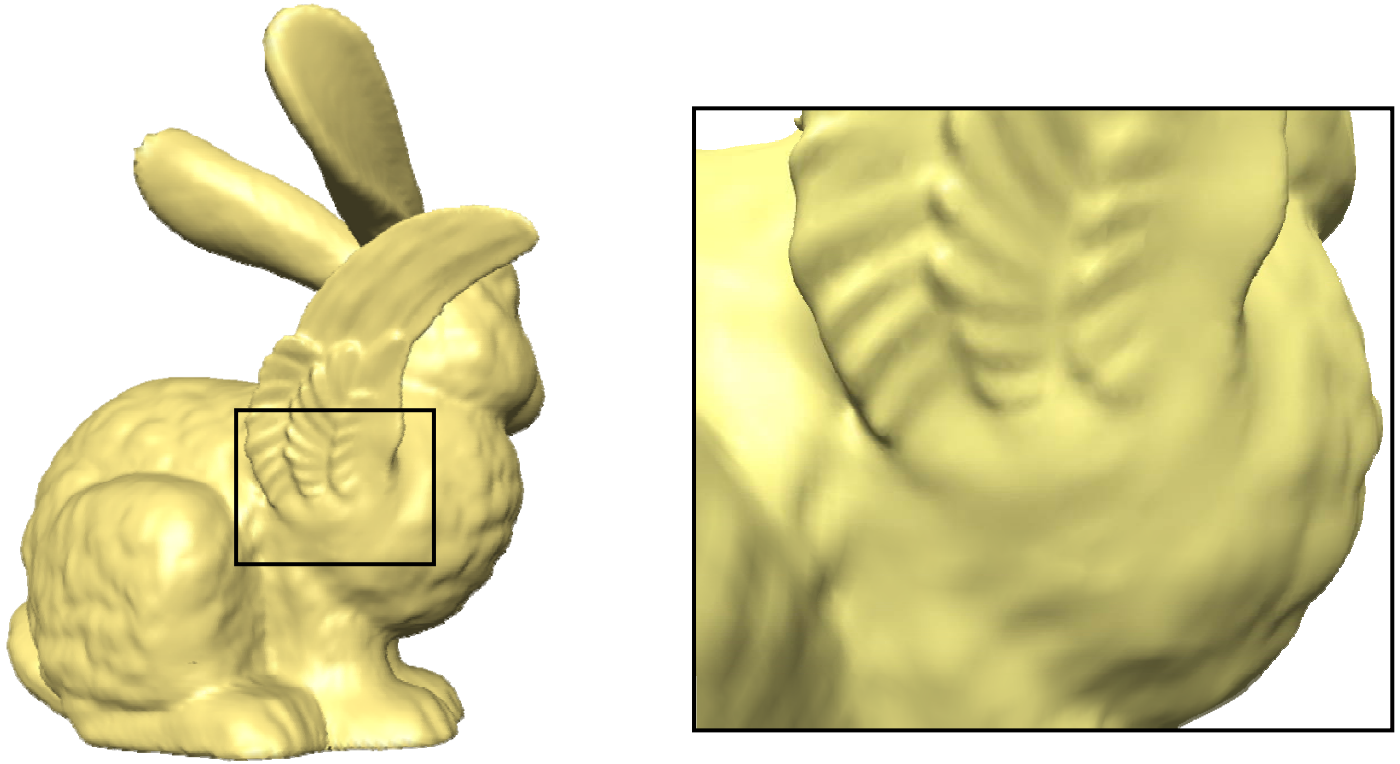
Mesh transplanting

- The user defines
  - Part to transplant
  - Where to transplant
  - Spatial orientation and scale
- Topological stitching
- Geometrical stitching via Laplacian mixing



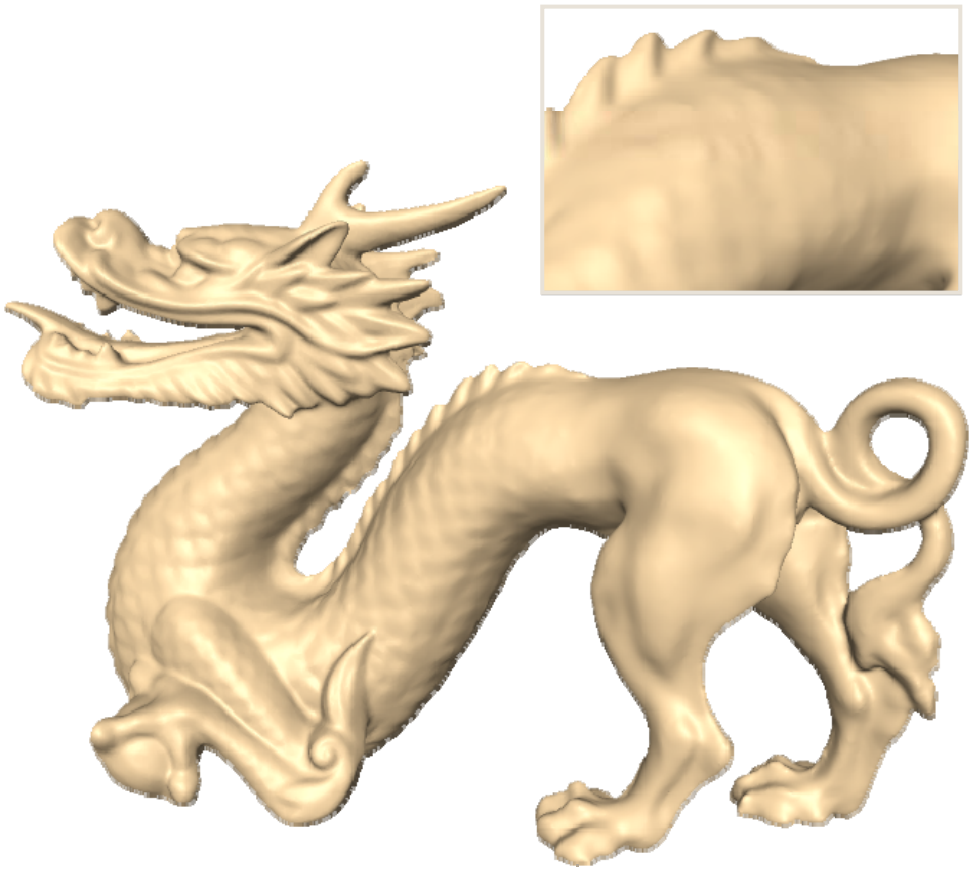
Mesh transplanting

- Details gradually change in the transition area



Mesh transplanting

- Details gradually change in the transition area



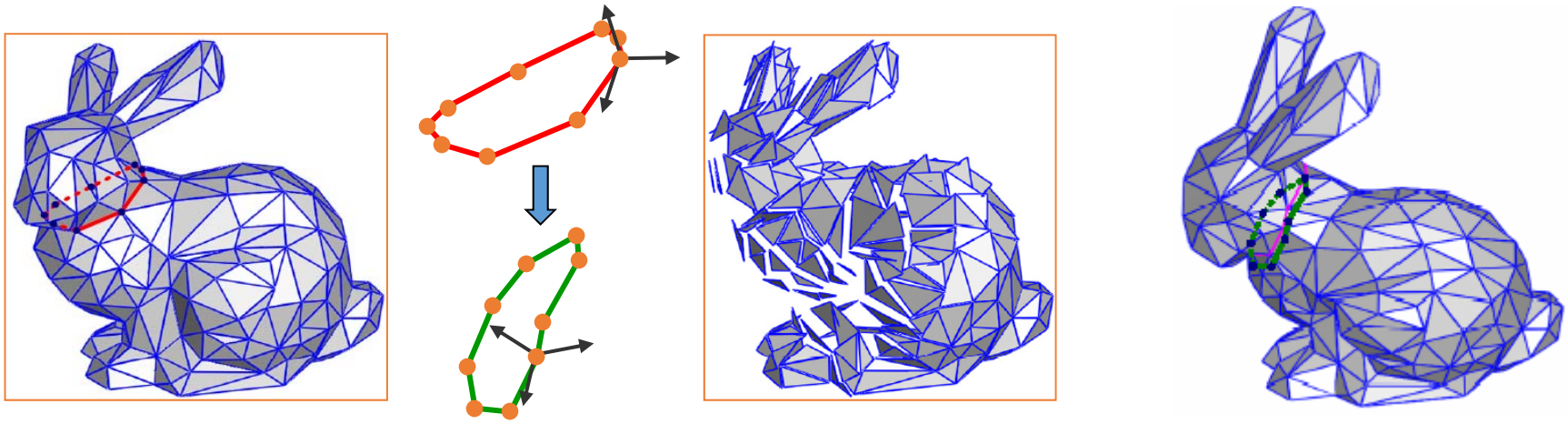
Invariance – solutions

- Explicit transformation of the differential coordinates prior to surface reconstruction
  - Lipman, Sorkine, Cohen-Or, Levin, Rössl and Seidel [SMI 04], “Differential Coordinates for Interactive Mesh Editing“,
    - Estimation of rotations from naive reconstruction
  - Yu, Zhou, Xu, Shi, Bao, Guo and Shum [SIGGRAPH 04], “Mesh Editing With Poisson-Based Gradient Field Manipulation“,
    - Propagation of handle transformation to the rest of the ROI using geodesic distances
  - Zayer, Rössl, Karni and Seidel [EG 05], “Harmonic Guidance for Surface Deformation“,
    - Propagation of handle transformation to the rest of the ROI using harmonic functions

1.5 Poisson Mesh Editing

Yu et al. Mesh Editing With Poisson-Based Gradient Field Manipulation. Siggraph 2004.

- The representation: the gradients of the functions X, Y, Z on each triangle of the mesh
- Deformation: propagate the transformation of the handle onto the ROI using geodesic distances



Poisson editing – images

- Inspiration: Poisson Image Editing [Pérez et al. 03]

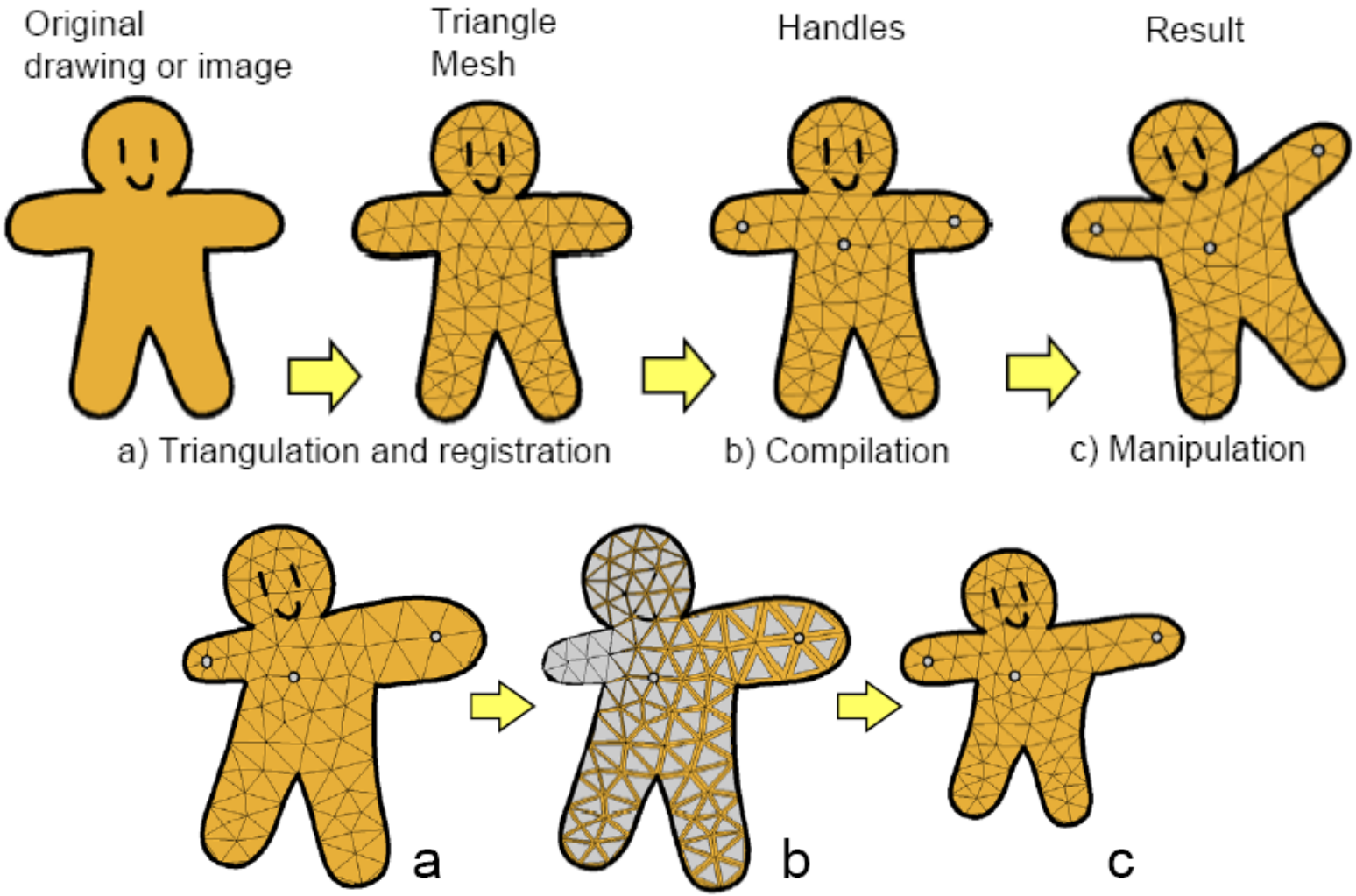
- Reconstruct a function from its gradients via the Poisson equation:

$$\arg\min_{f} \int_{\Omega} \|\nabla f - \mathbf{w}\|^2, \quad \text{s.t.} \quad f\big|_{\partial\Omega} = f^*\big|_{\partial\Omega}$$

$$\Delta f = \operatorname{div} \mathbf{w} \quad \text{with} \quad f\big|_{\partial\Omega} = f^*\big|_{\partial\Omega}$$

1.6 As-rigid-as-possible Deformation

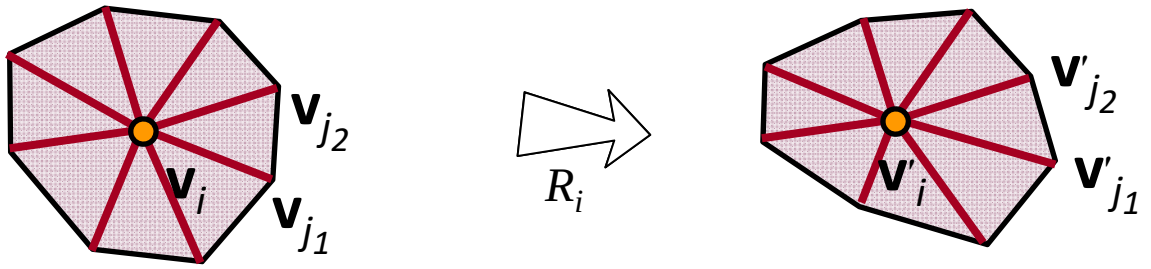
[Siggraph 2003]



1.7 ARAP Modeling

[Sorkine and Alexa, As-Rigid-As-Possible Surface Modeling. SGP 2007]

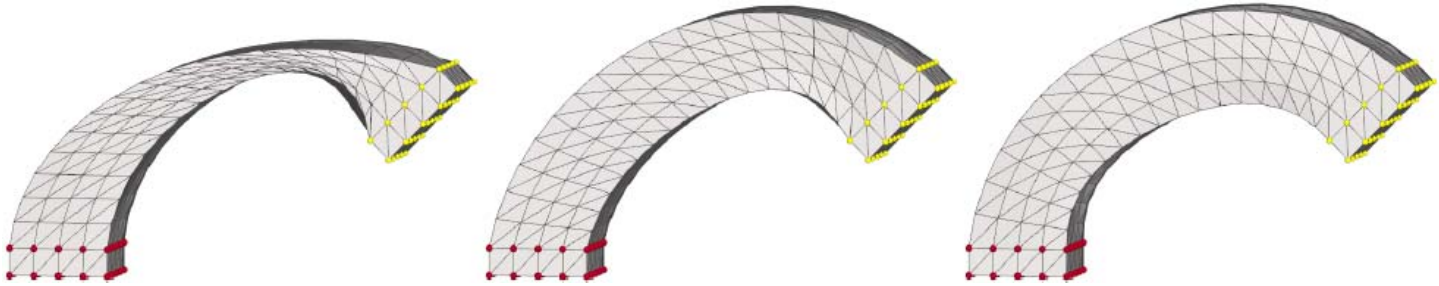
- Ask all star edges to transform rigidly by some rotation R, then the shape of the cell is preserved



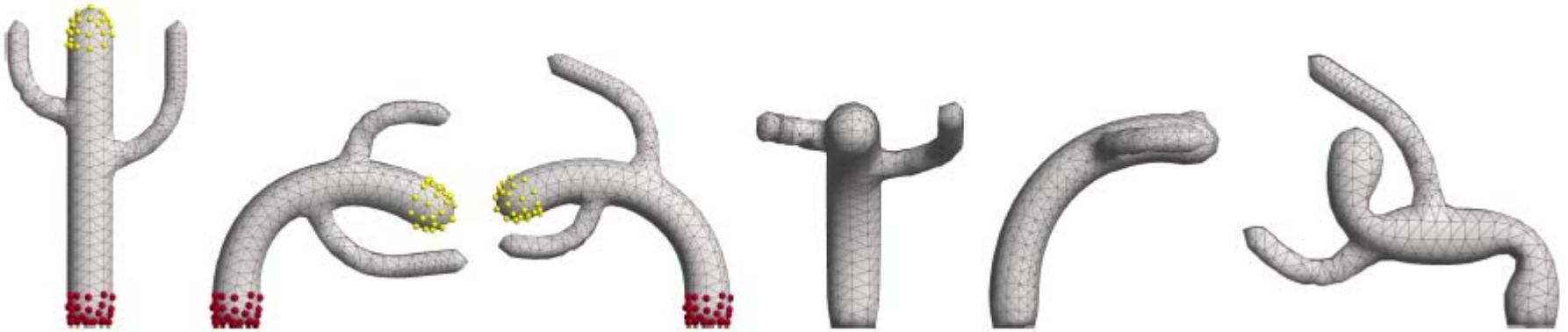
$$\min_{\mathbf{v}'} \sum_{i=1}^{n} \sum_{j \in N(i)} \left\| (\mathbf{v}'_i - \mathbf{v}'_j) - R_i(\mathbf{v}_i - \mathbf{v}_j) \right\|^2$$

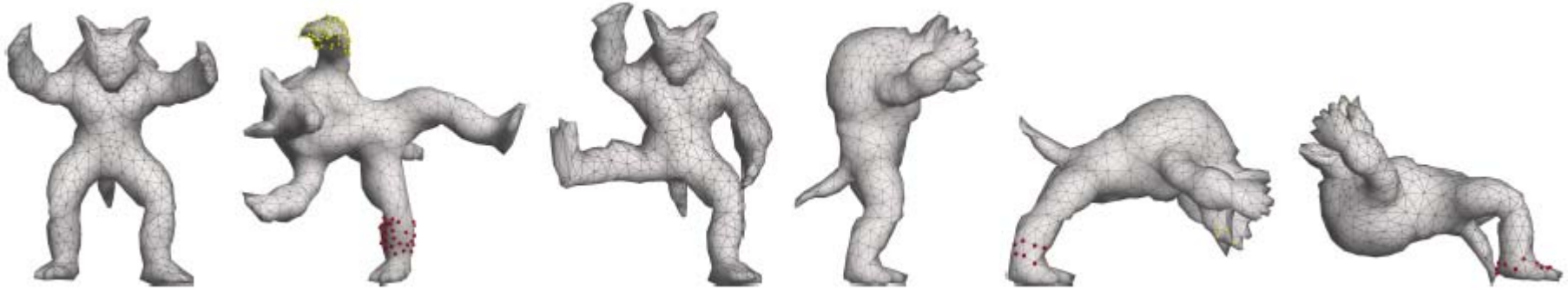
$$s.t.\ \mathbf{v}'_j = \mathbf{c}_j,\ j \in C$$

ARAP Modeling



initial guess 1 iterations 4 iterations

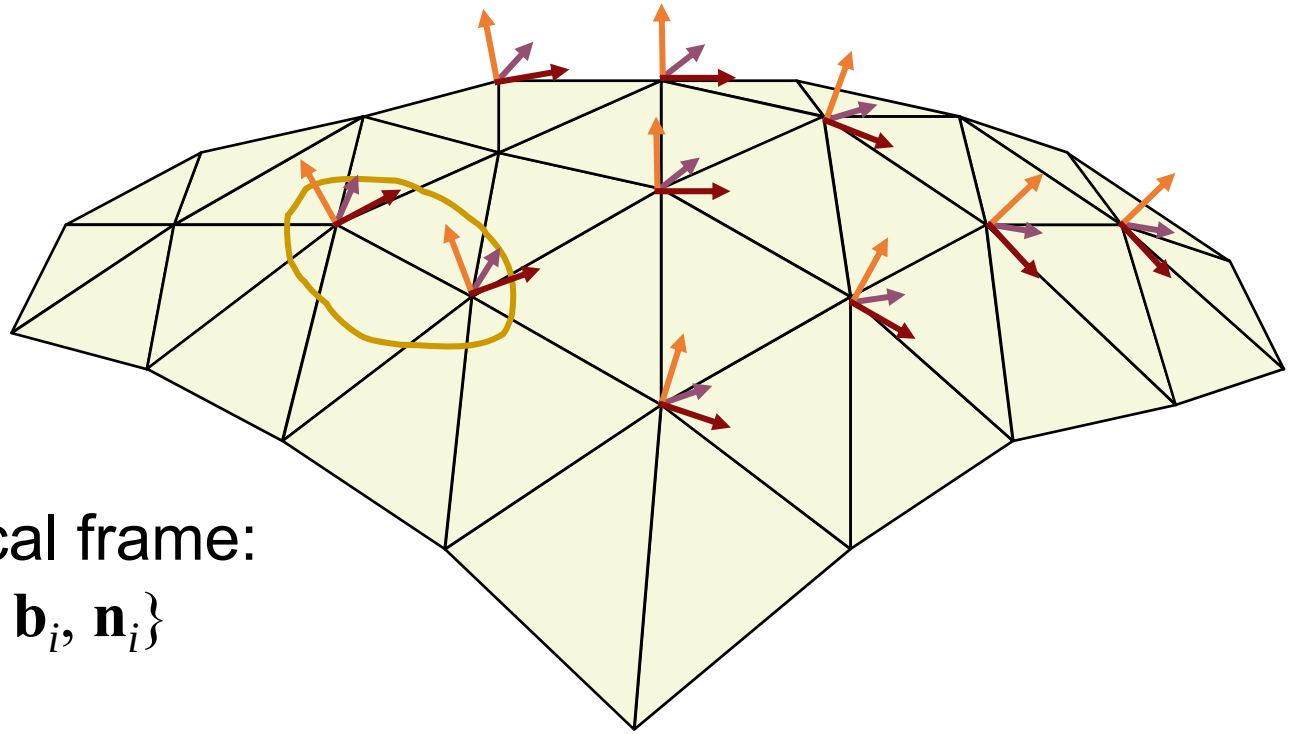




1.8 Linear Rotation-invariant Coordinates

[Lipman et al. Siggraph 05]

- Keep a local frame at each vertex
- Prescribe changes to some selected frames



Local frame:

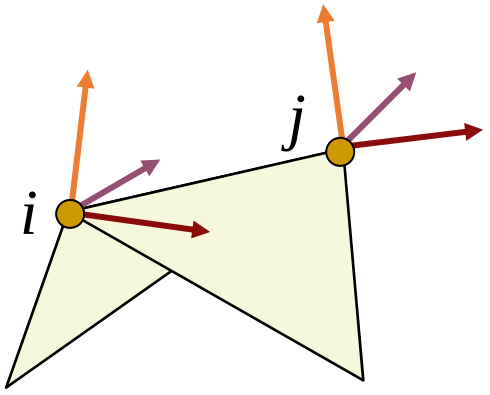
$\{\mathbf{a}_i, \mathbf{b}_i, \mathbf{n}_i\}$

Frame-based deformations

- Encode the differences between adjacent frames
- Solve for the new frames in least-squares sense

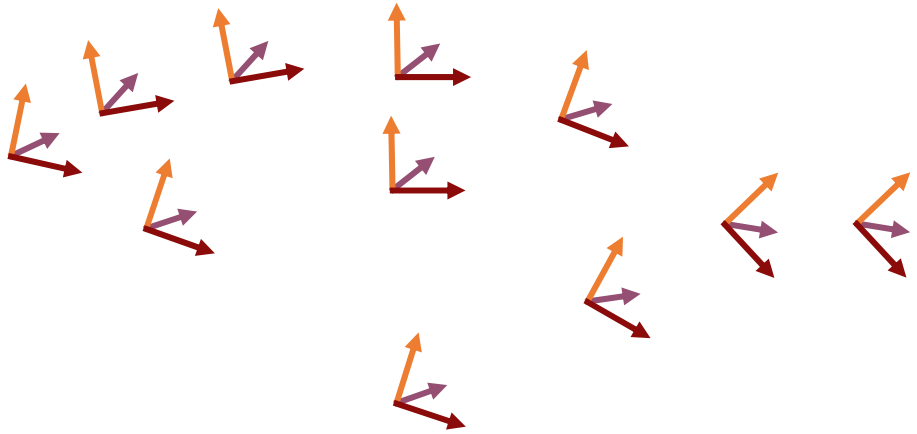
$$\mathbf{a}_i - \mathbf{a}_j = \alpha_1 \mathbf{a}_i + \alpha_2 \mathbf{b}_i + \alpha_3 \mathbf{n}_i$$
$$\mathbf{b}_i - \mathbf{b}_j = \beta_1 \mathbf{a}_i + \beta_2 \mathbf{b}_i + \beta_3 \mathbf{n}_i$$
$$\mathbf{n}_i - \mathbf{n}_j = \gamma_1 \mathbf{a}_i + \gamma_2 \mathbf{b}_i + \gamma_3 \mathbf{n}_i$$
$$\ldots\ldots$$

constraints



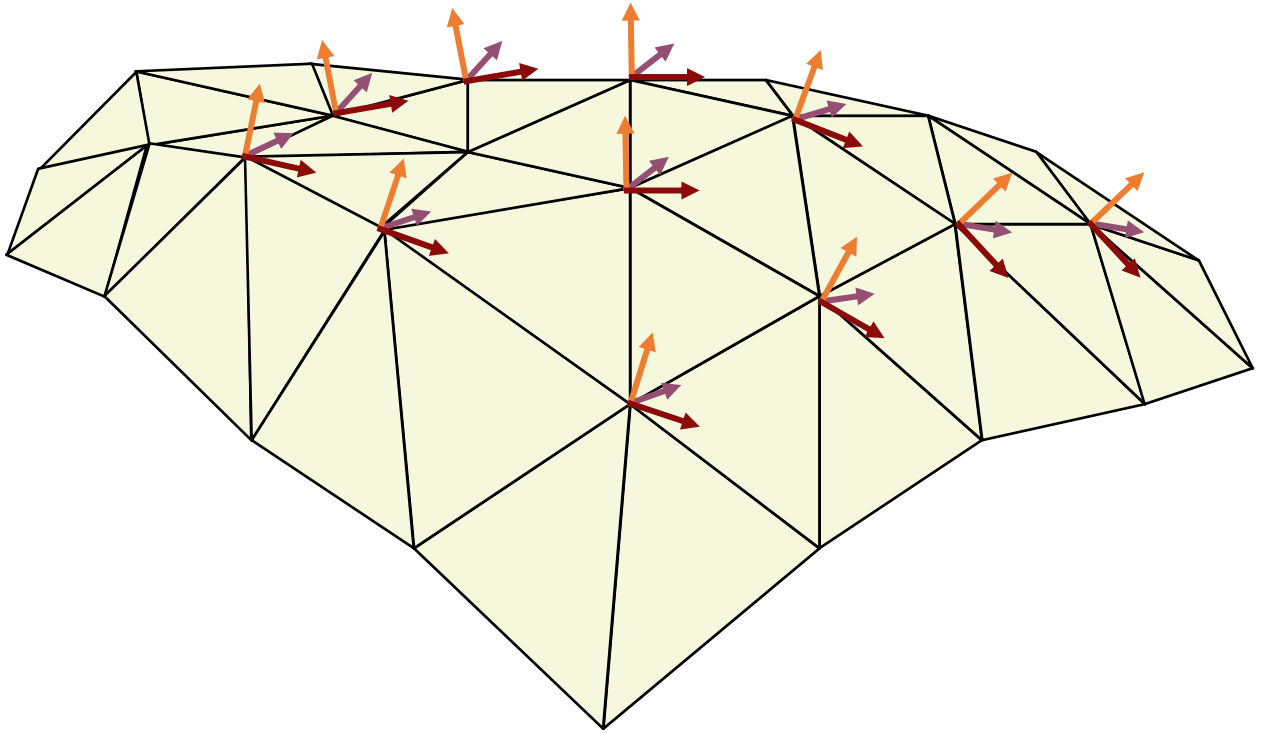
Frame-based deformations

- Reconstruction:
  - After having the frames, solve for positions

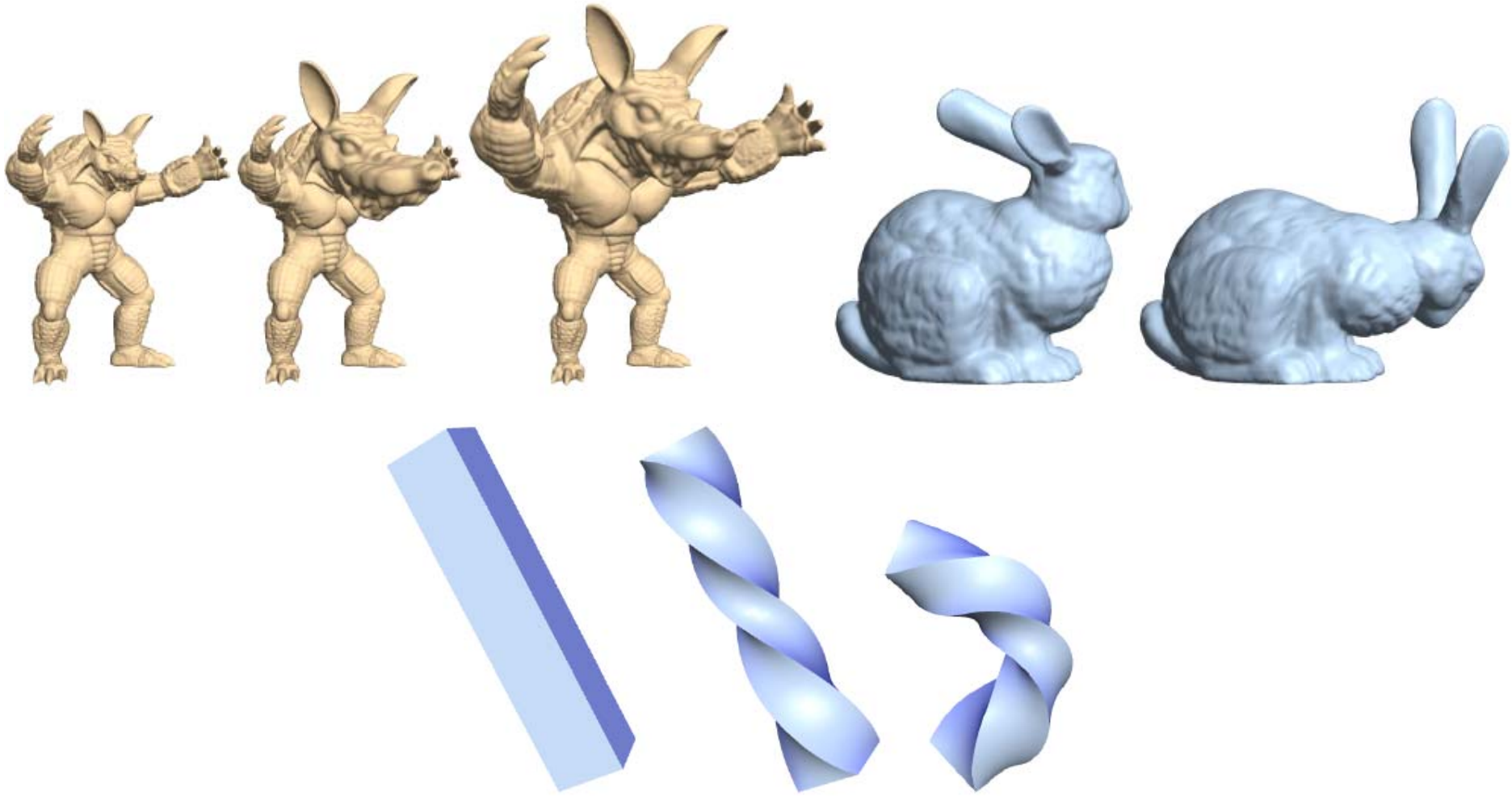


Frame-based deformations

- Reconstruction:
  - After having the frames, solve for positions



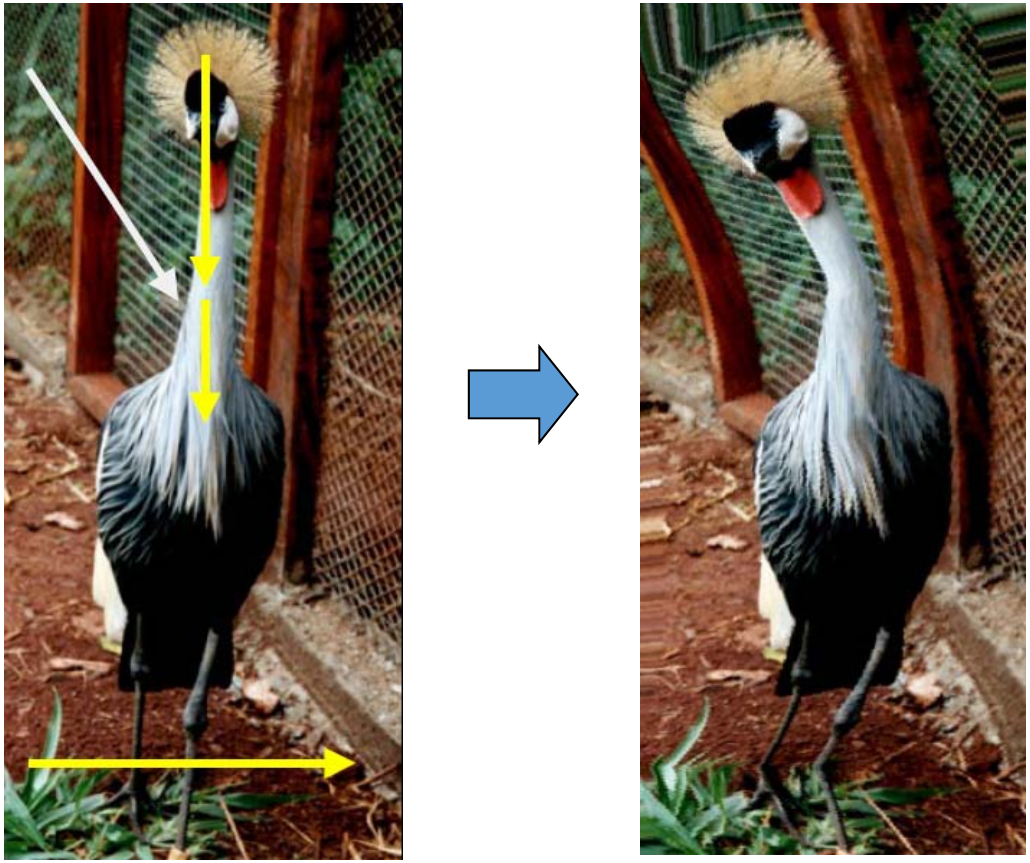
Results



2. Curve Proxy

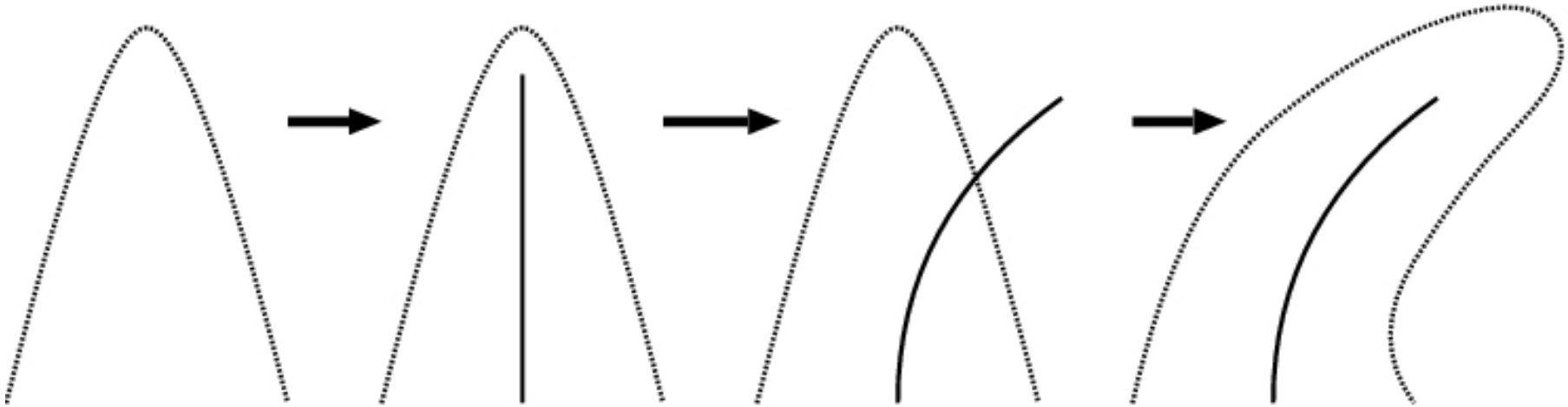
User Specifications

- Line or curve features

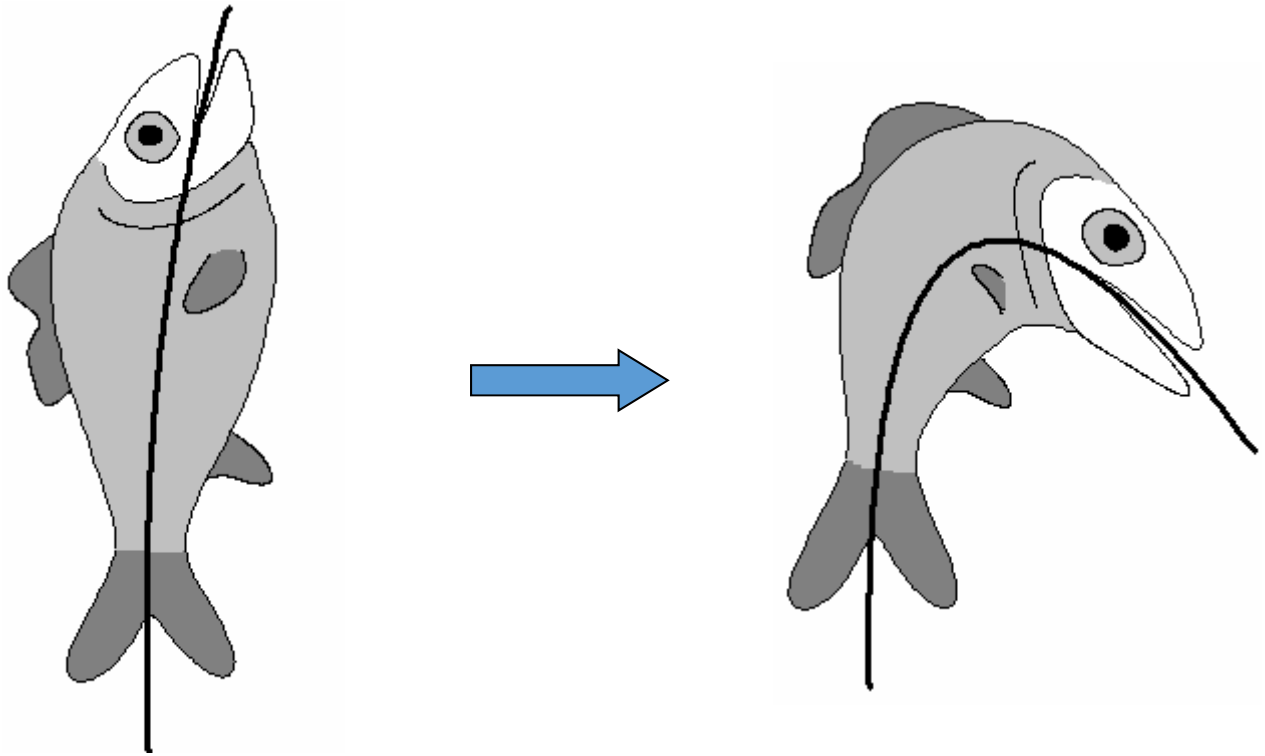


2.1 Skeleton based

- Axial Deformation [1994]

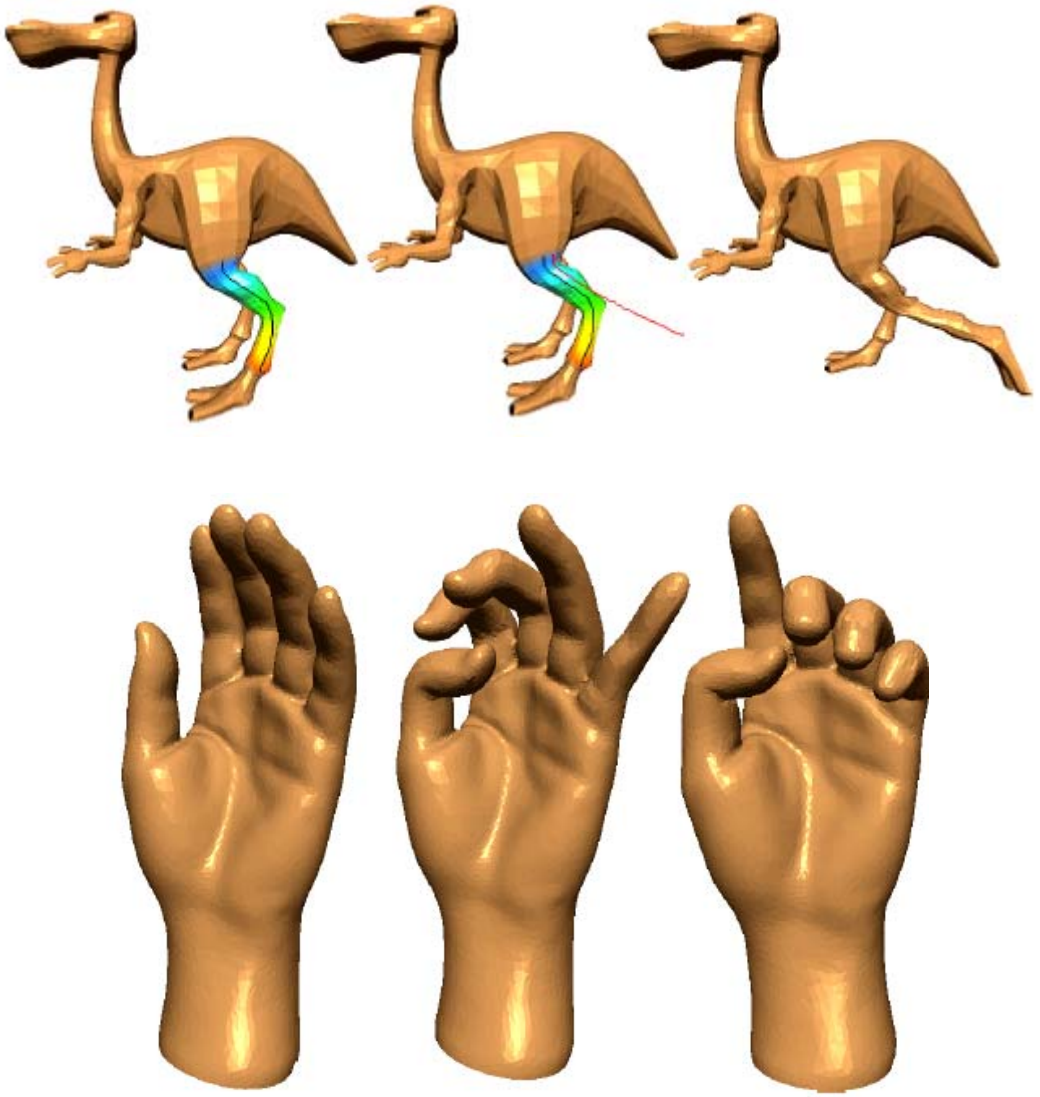


Axial Deformation

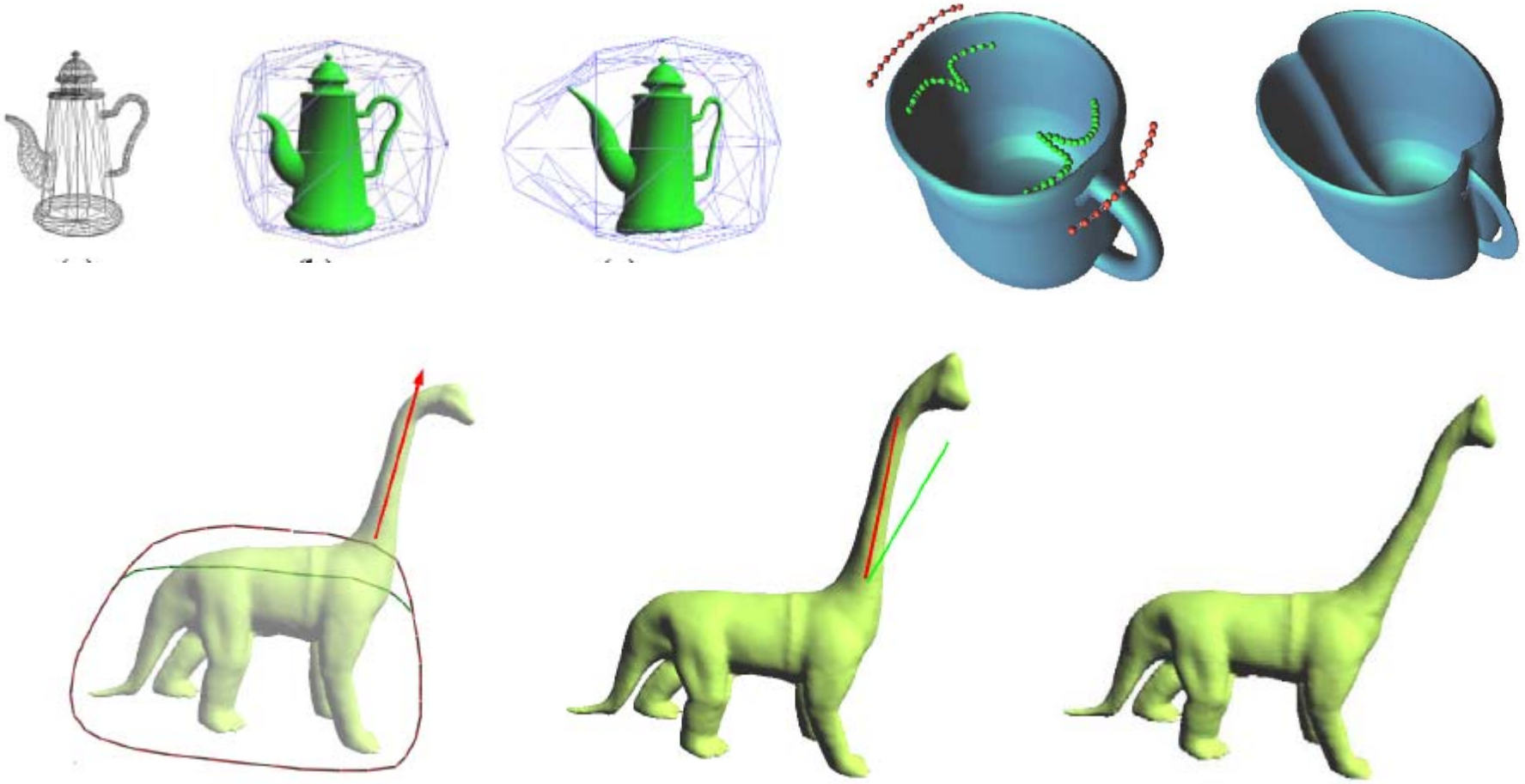


2.2 Sketching Deformations

[2005]

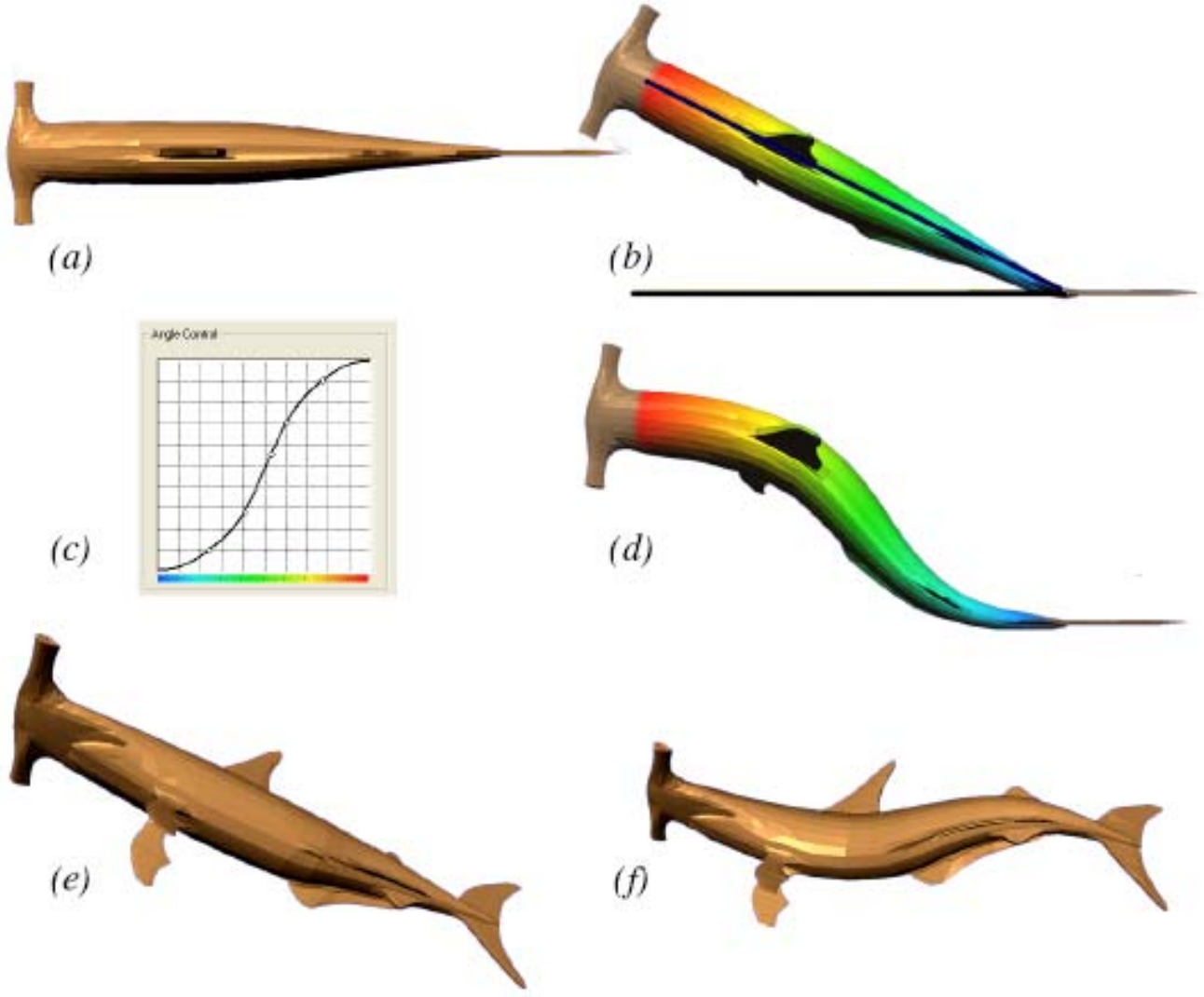


Sketching Deformation



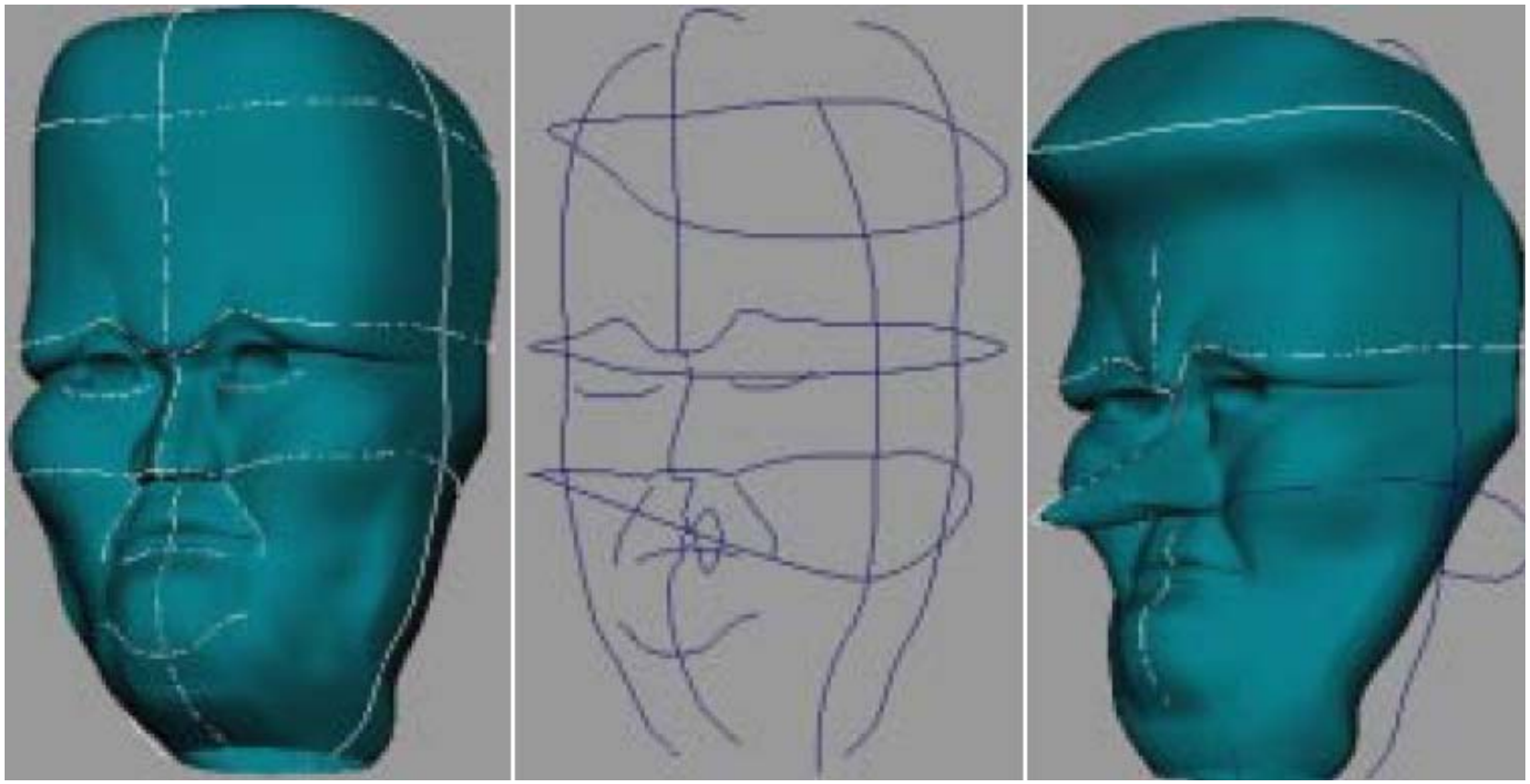
Skeleton-based Deformation

[2002]


(a)
(b)
Angle Control
(c)
(d)
(e)
(f)

2.3 Wires based

[Siggraph 1999]

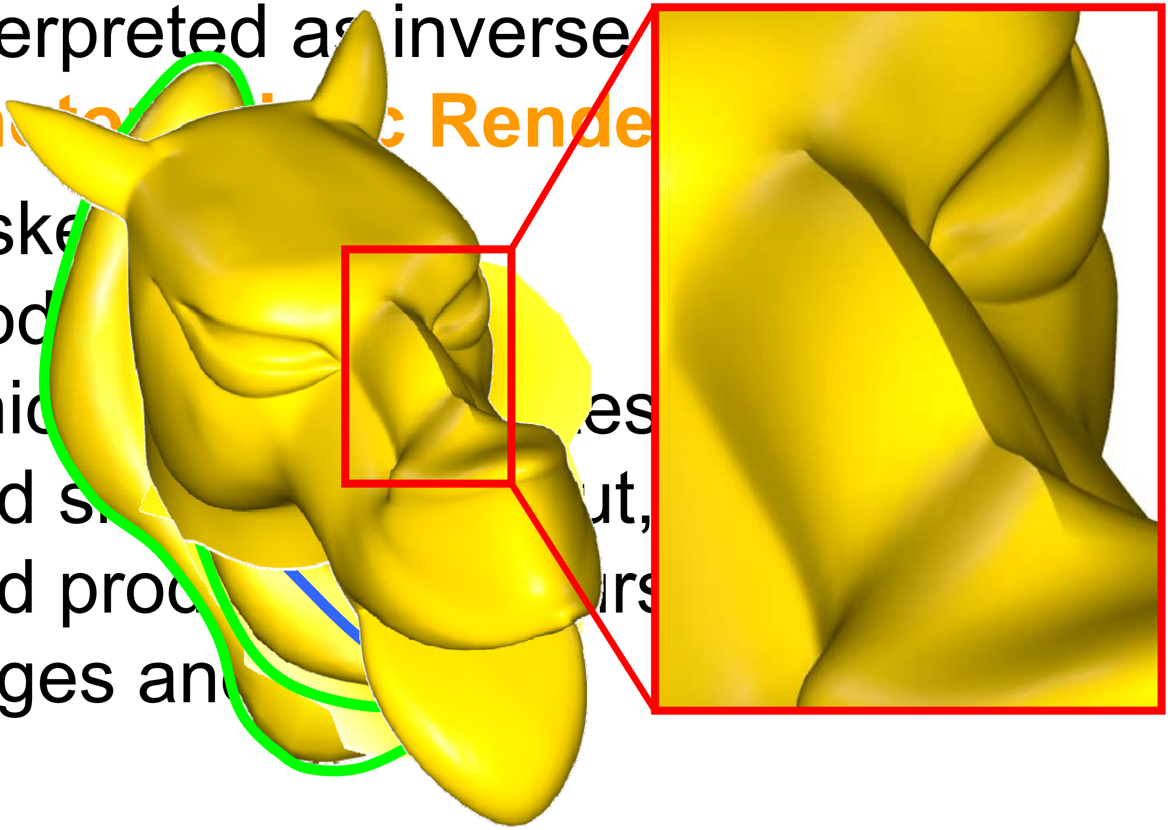


2.4 Silhouette sketching based

[Nealen et al. Siggraph 05]

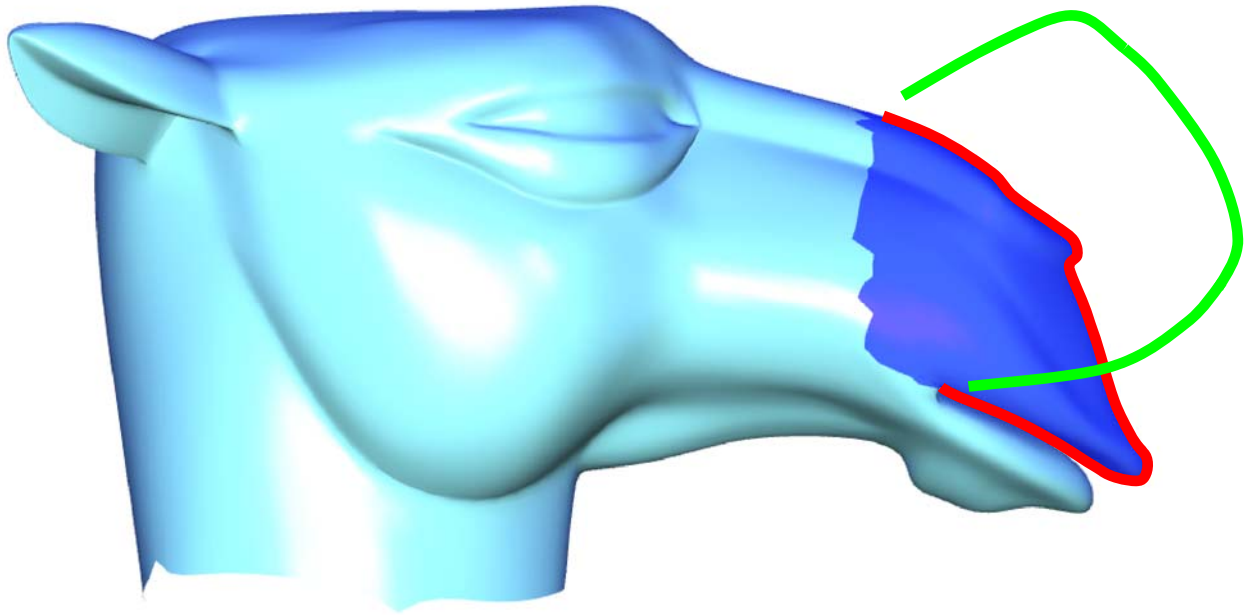
• Silhouette sketching

- ... can be interpreted as inverse ... **Ph... c Rende...**
- A ske... mod... whic... es and s... ut, and proc... rs ridges an...



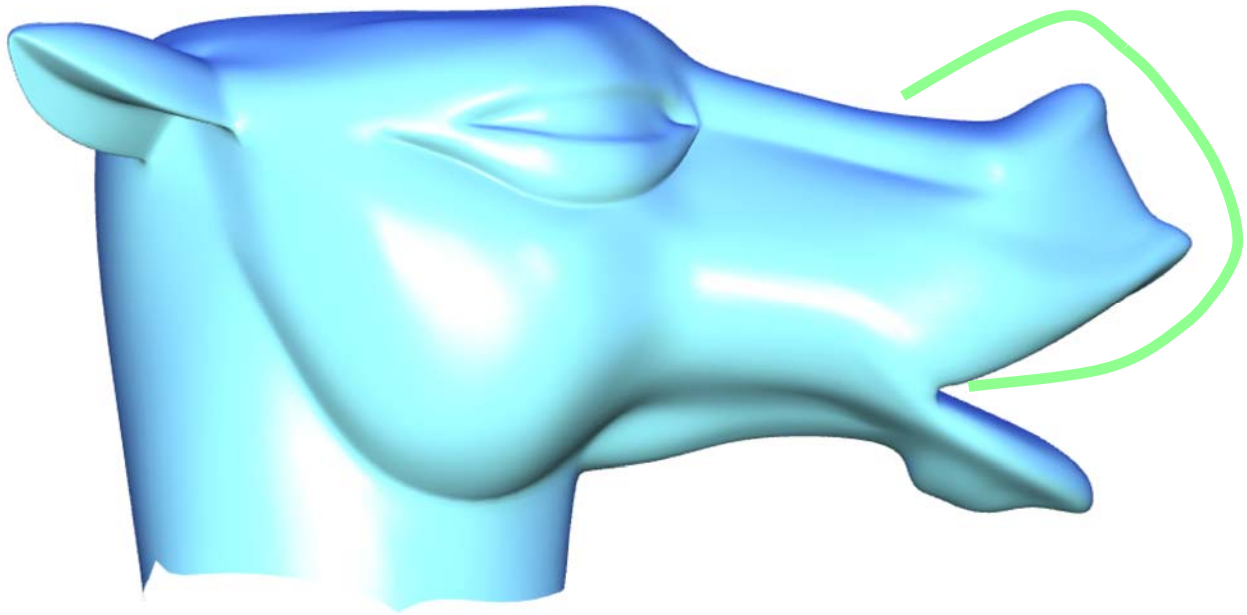
Silhouette Sketching

- Approximate sketching
  - Balance weighting between detail and positional constraints



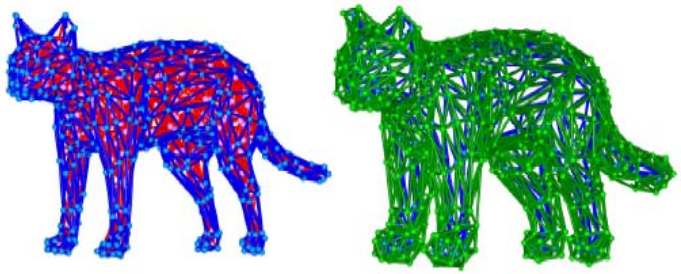
Silhouette Sketching

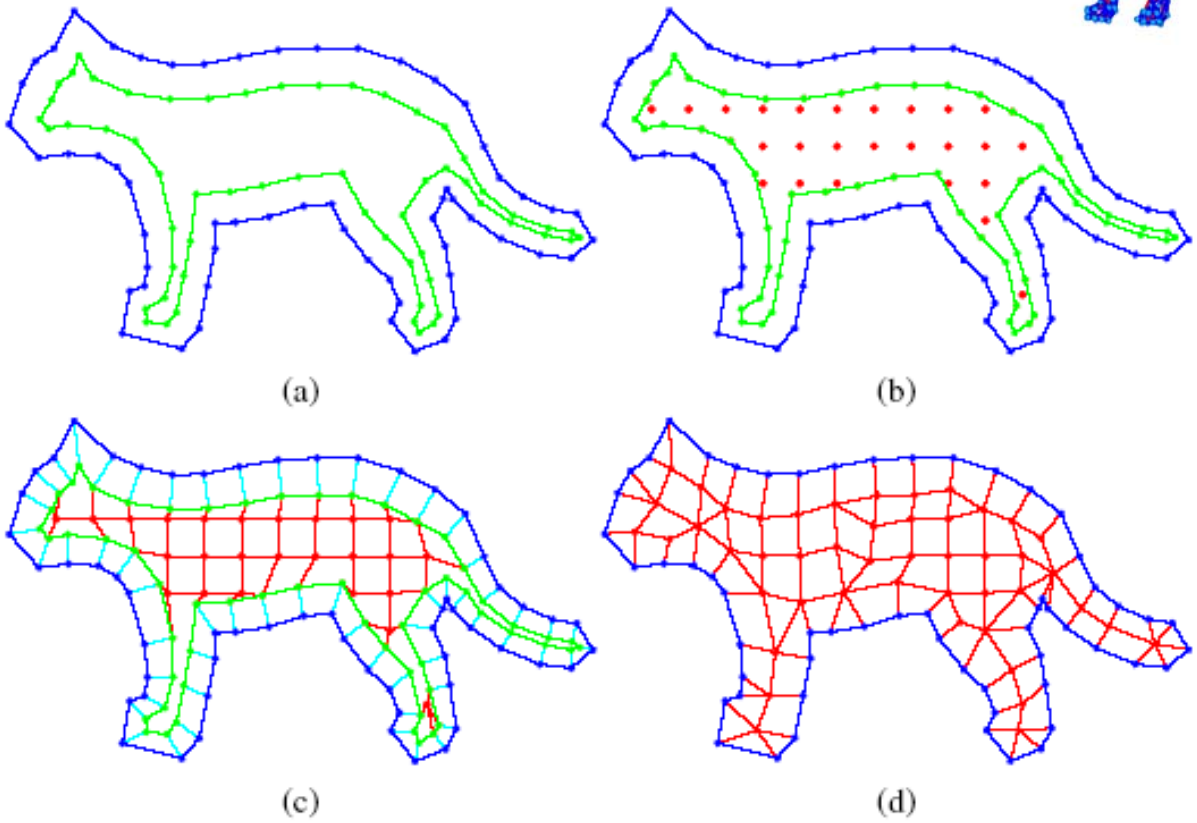
- Approximate sketching
  - Balance weighting between detail and positional constraints

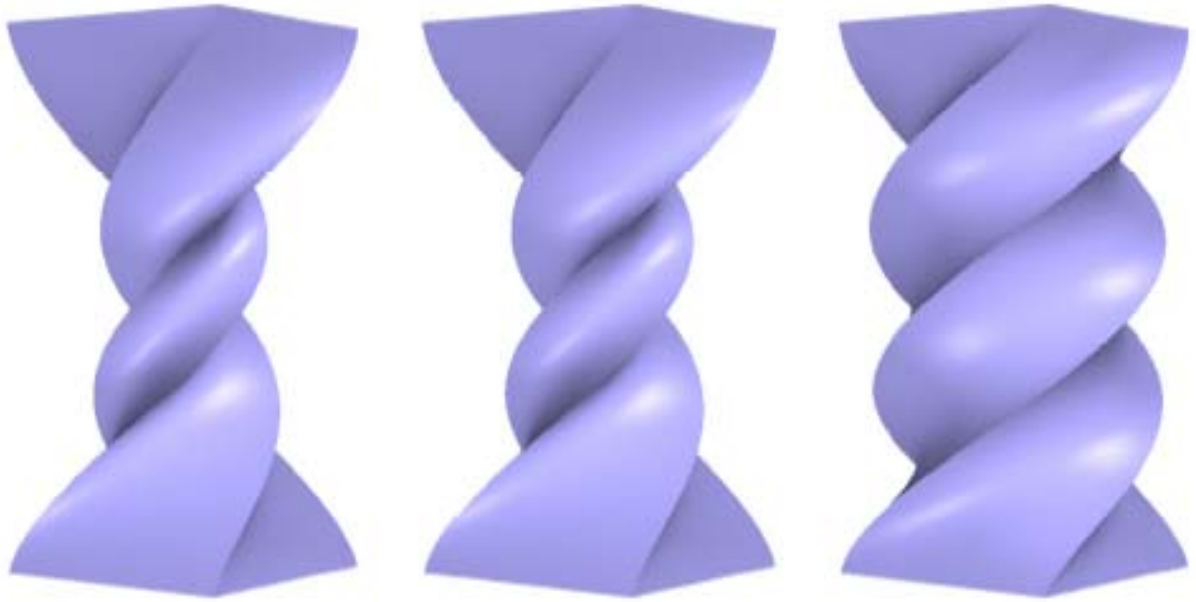


2.5 Volumetric Graph Laplacian

[Zhou et al. Siggraph 2005]



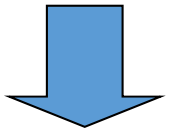


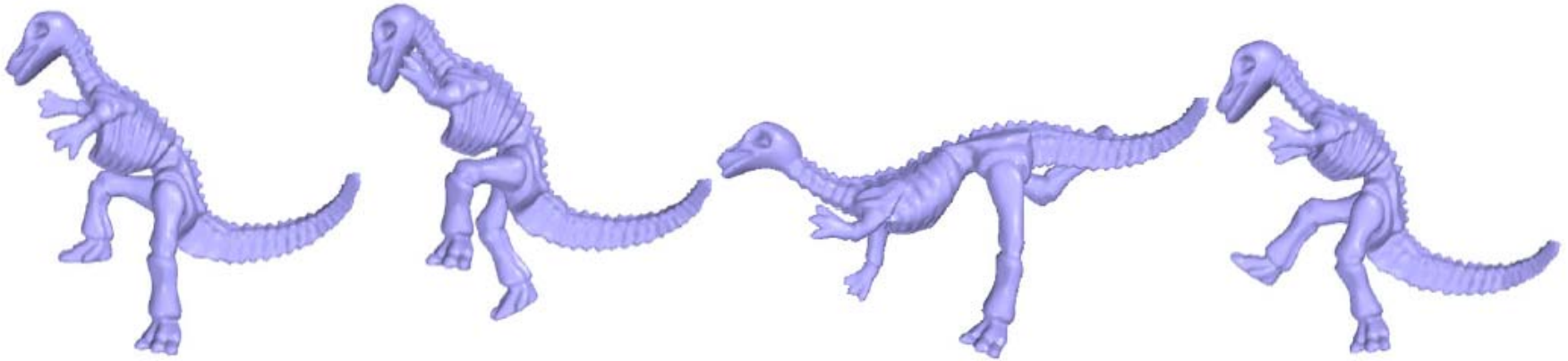


(a) Laplacian surface (b) Poisson mesh (c) VGL

(a) Laplacian surface (b) Poisson mesh (c) VGL

Animation Modeling from 2D Cartoons





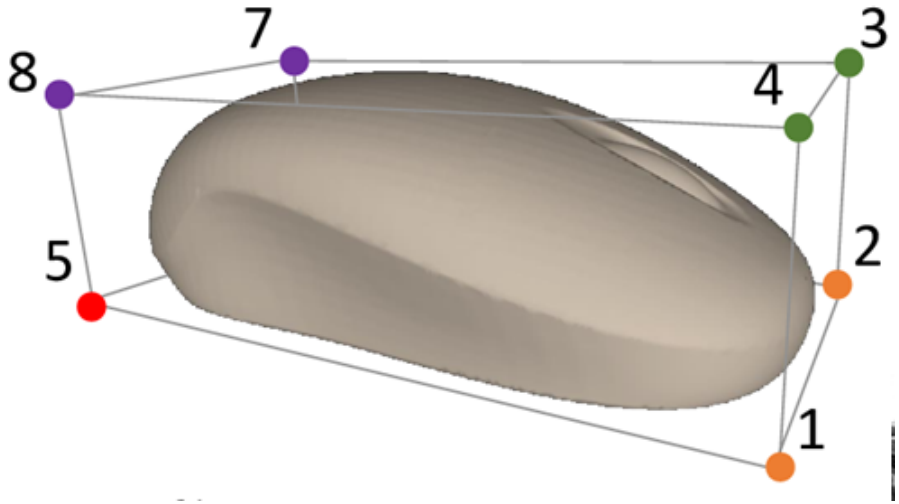
Short Summary: curve proxy

- Curve features
- Skeletons
- Silhouette
- Free-form sketches
- Wires
- 2D image contours
- …

3. Mesh Proxy

User Specifications

- Define a mesh proxy for the shape: generally a simpler, coarse mesh which encloses the shape
  - The shape is embedded into the mesh proxy
- Deform the mesh proxy using point/line base methods
- Advantages:
  - You can deform arbitrary objects
  - Independent of object representation



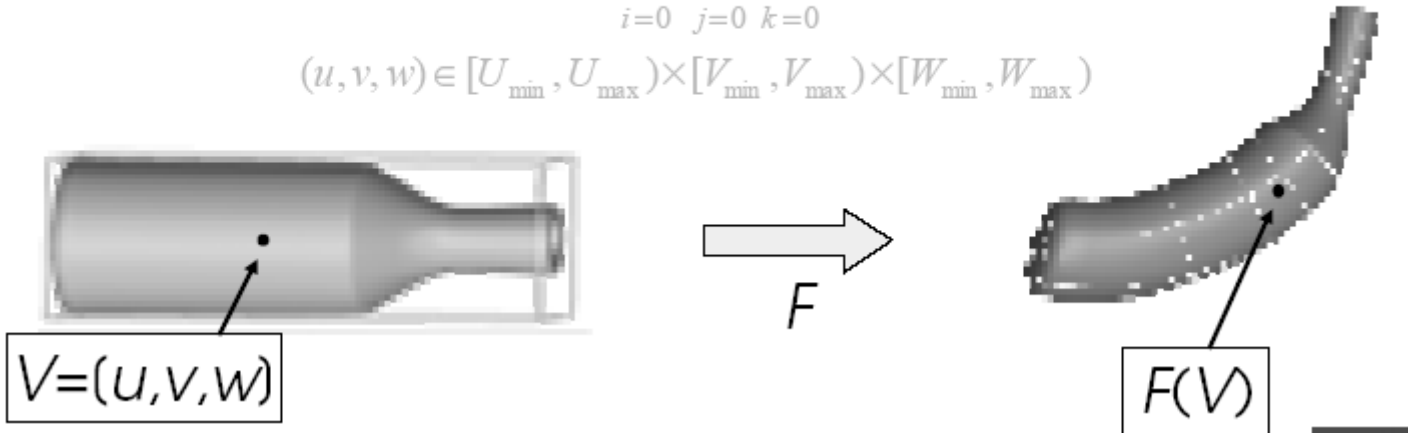
3.1 Free-form Deformation (FFD)

[Sederberg et al. 86]

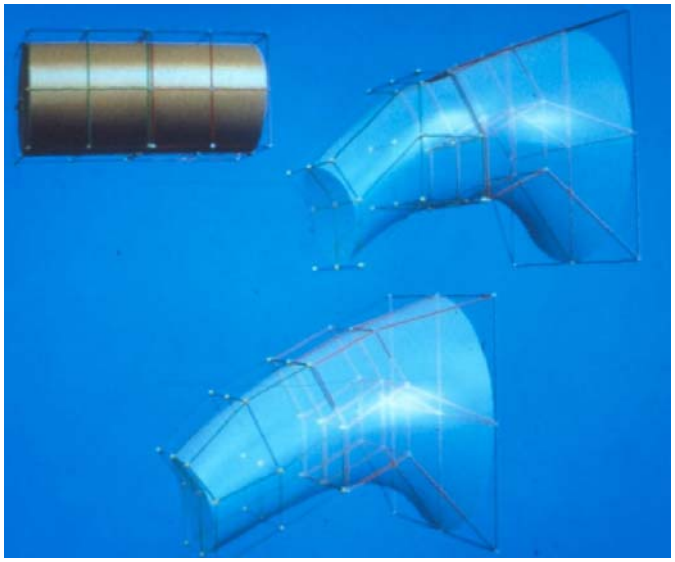
- Proxy: a **lattice** mesh
  - more easily parametrized than the object
  - Deformation defined by the tensor Bezier solid

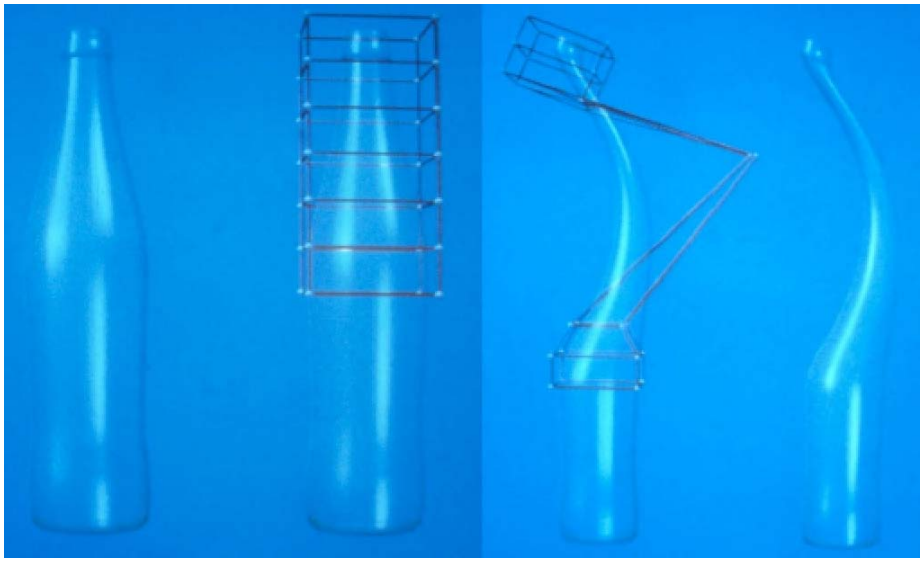
$$X_{new} = F(u,v,w) = \sum_{i=0}^{l}\sum_{j=0}^{m}\sum_{k=0}^{n} P_{ijk} B_i(u) B_j(v) B_k(w)$$

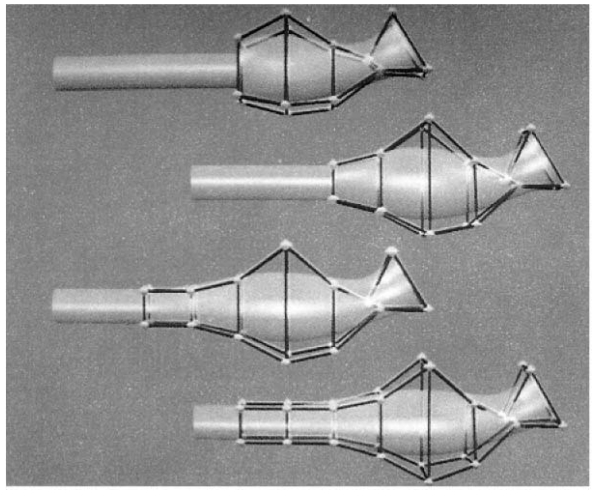
$$(u,v,w) \in [U_{\min}, U_{\max}) \times [V_{\min}, V_{\max}) \times [W_{\min}, W_{\max})$$

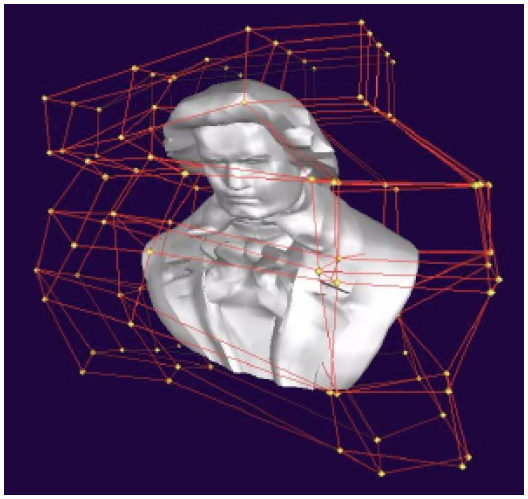


FFD Examples



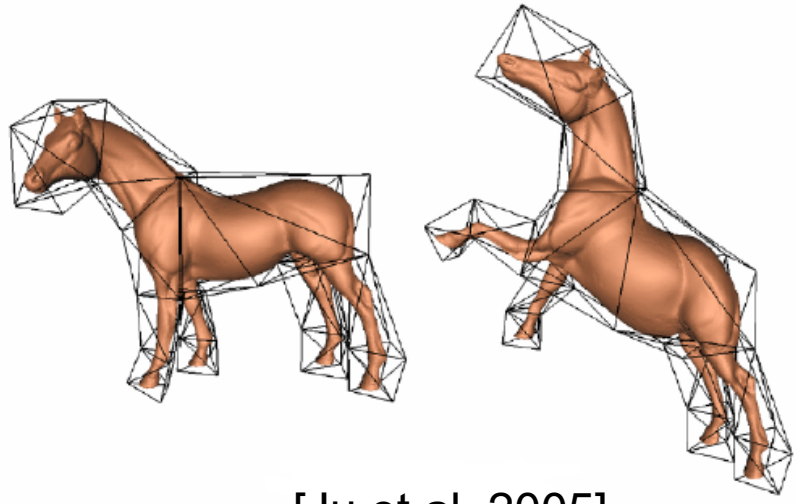




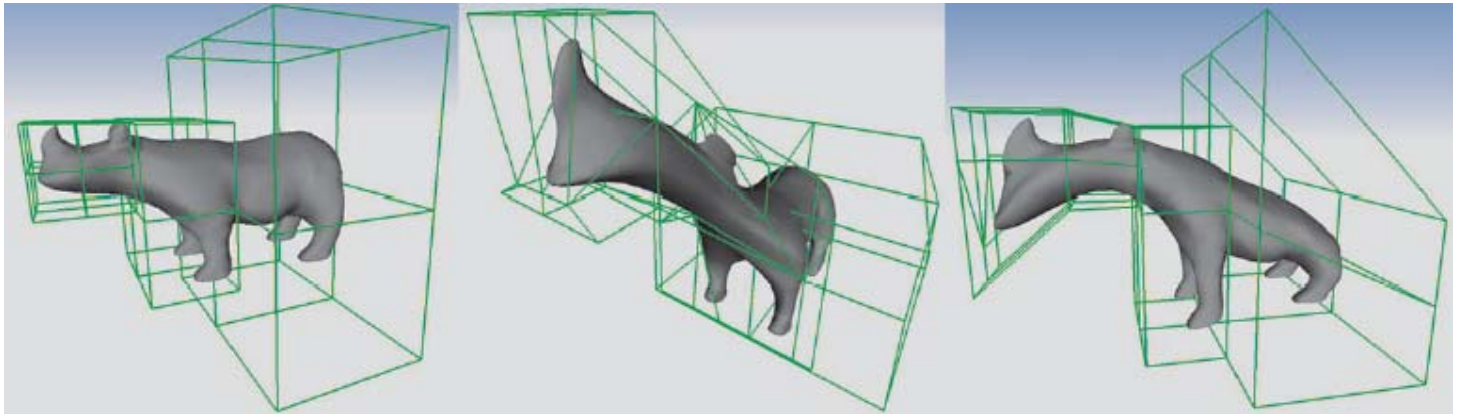


3.2 Cage-based deformation

- Proxy: cage
  - Barycentric coordinates
  - Many works…



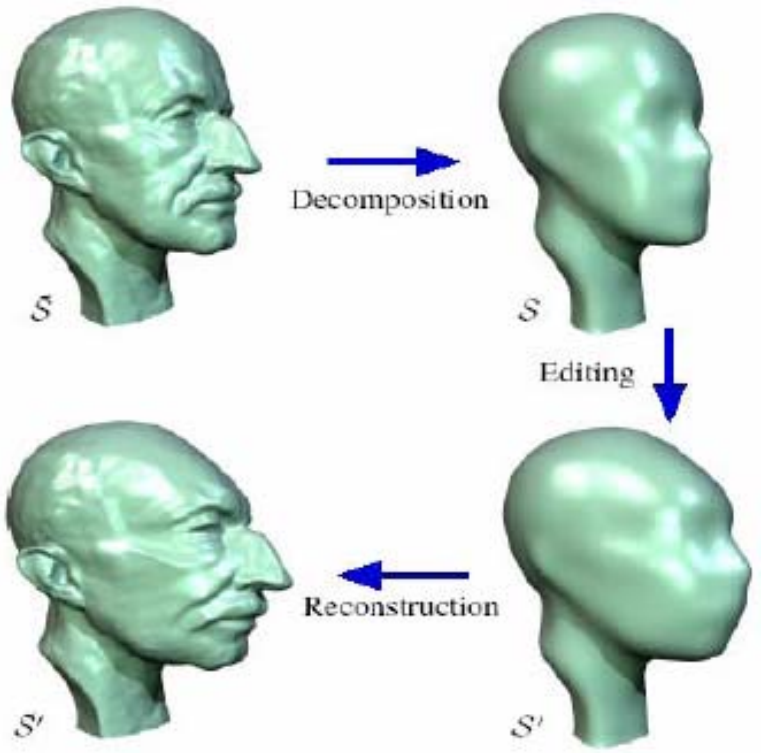
[Ju et al. 2005]



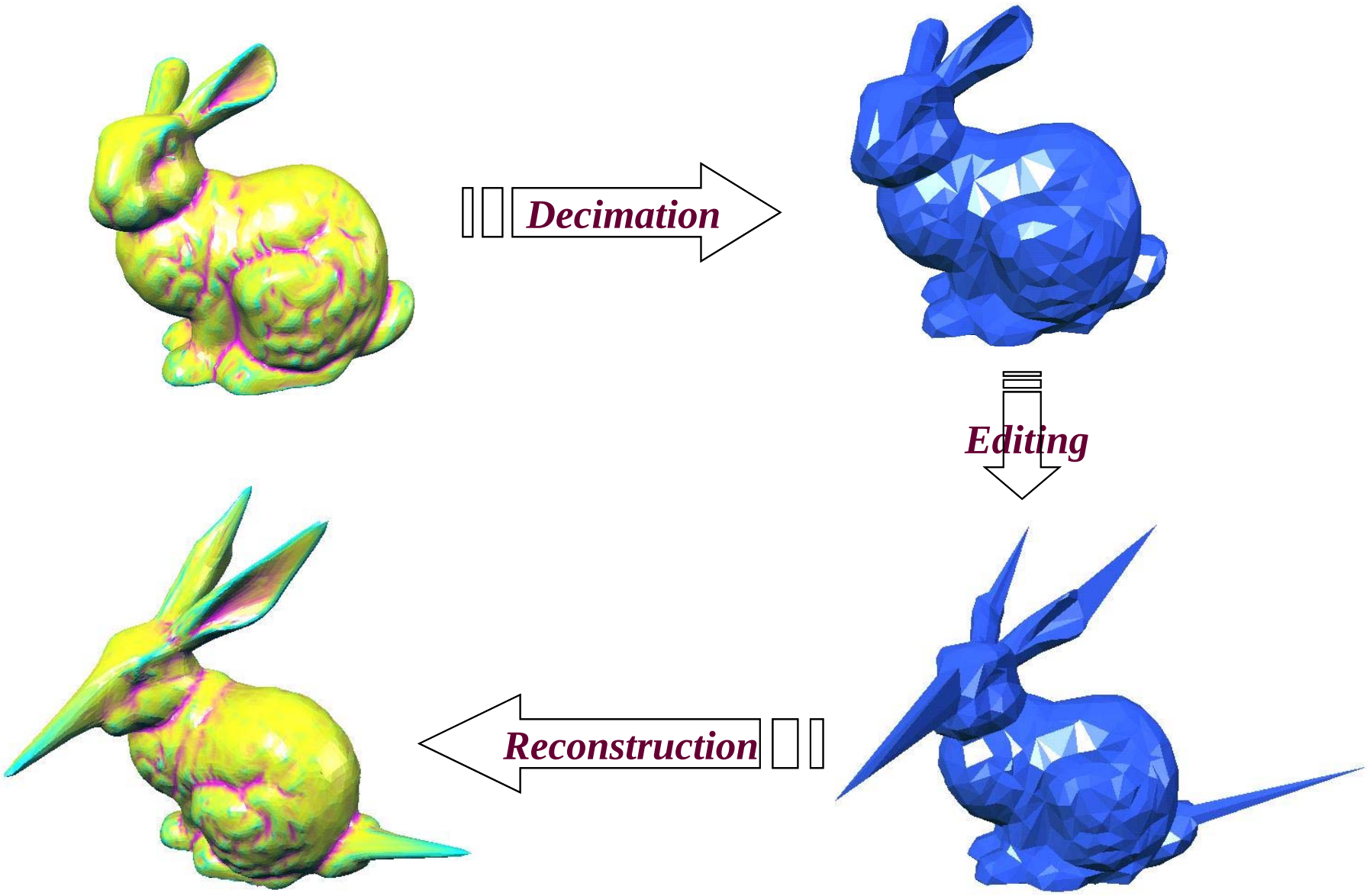
3.3 Multiresolution Editing

- Proxy: simplified shape

- Pros
  - Preserving details, scalable
- Cons
  - Instable reconstruction for large deformation
  - Resampling problem
- Invariant variables
  - Detail information



[Kobbelt et al. 2003]


Decimation
Editing
Reconstruction

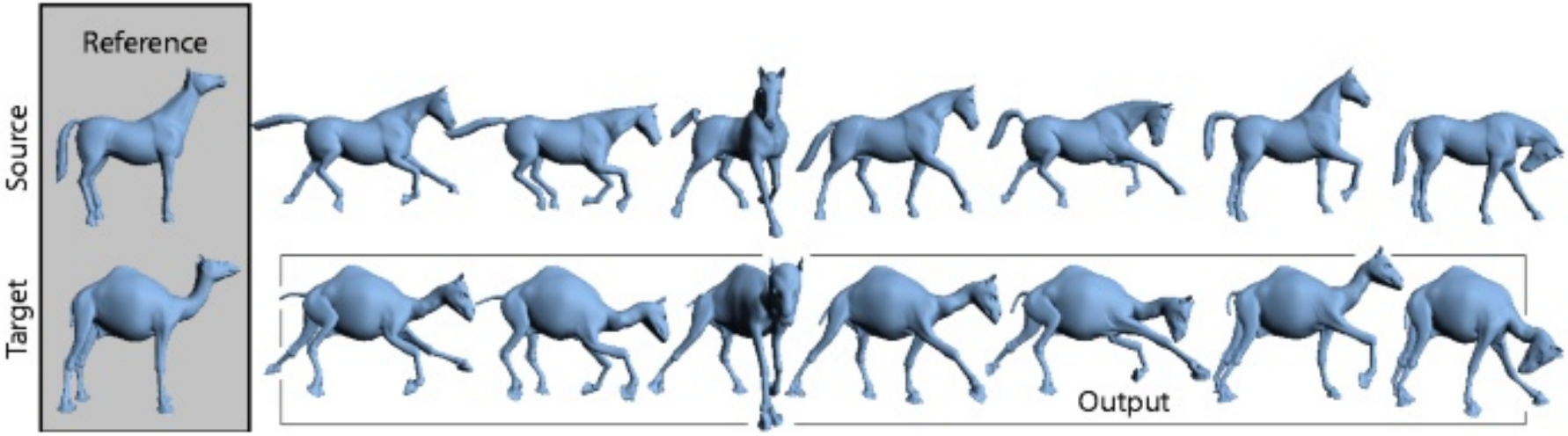
Short Summary: mesh proxy

- Lattice
- Cage
- Simplified shape
- …

4. Deformation Transfer

Deformation Transfer

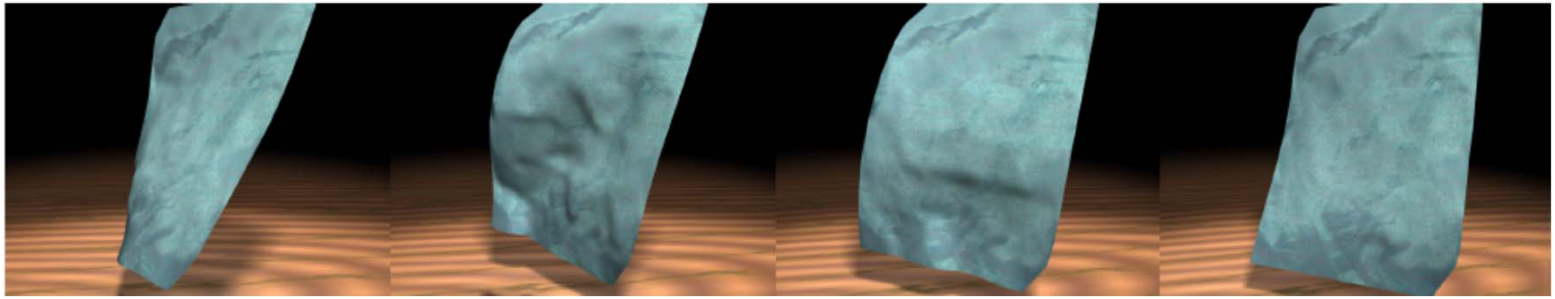
Siggraph 2004



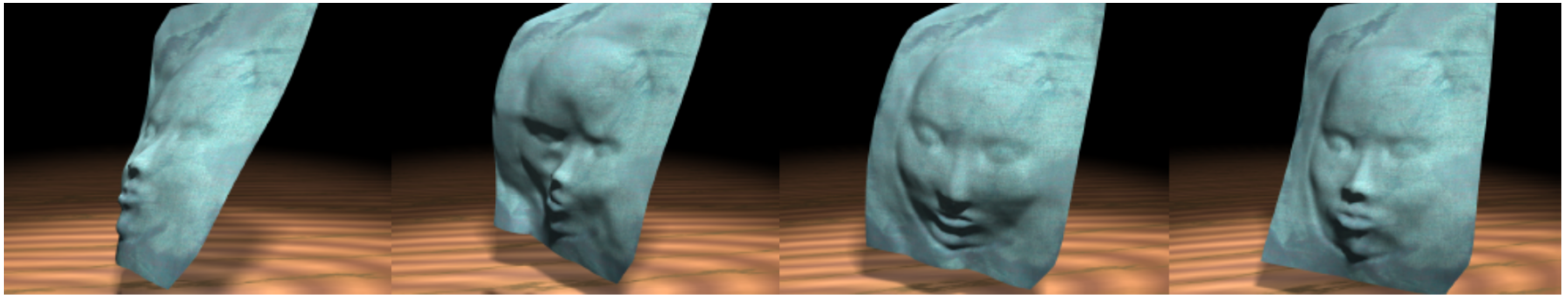
Editing Arbitrary Deforming Surface Animations

Siggraph 2006

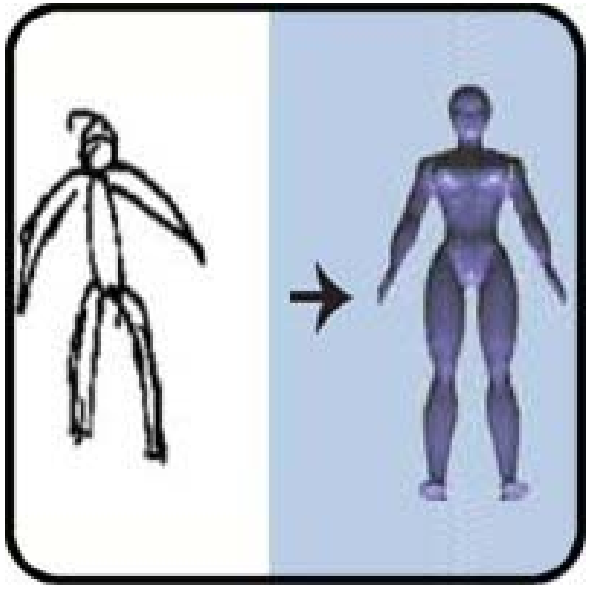
Deforming Surface



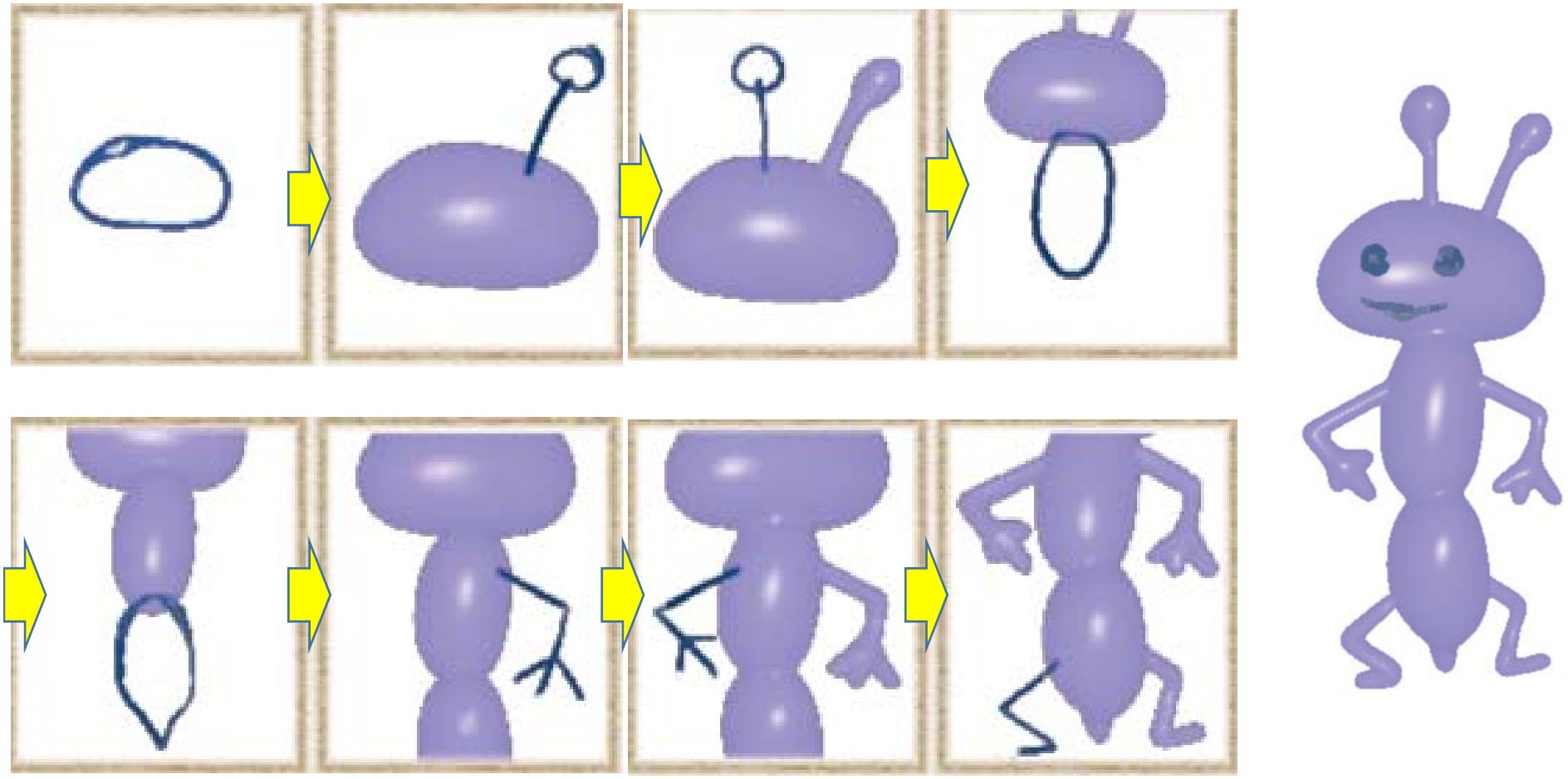
Editing Surface



5. Sketch-based Modeling
(基于草图的建模)

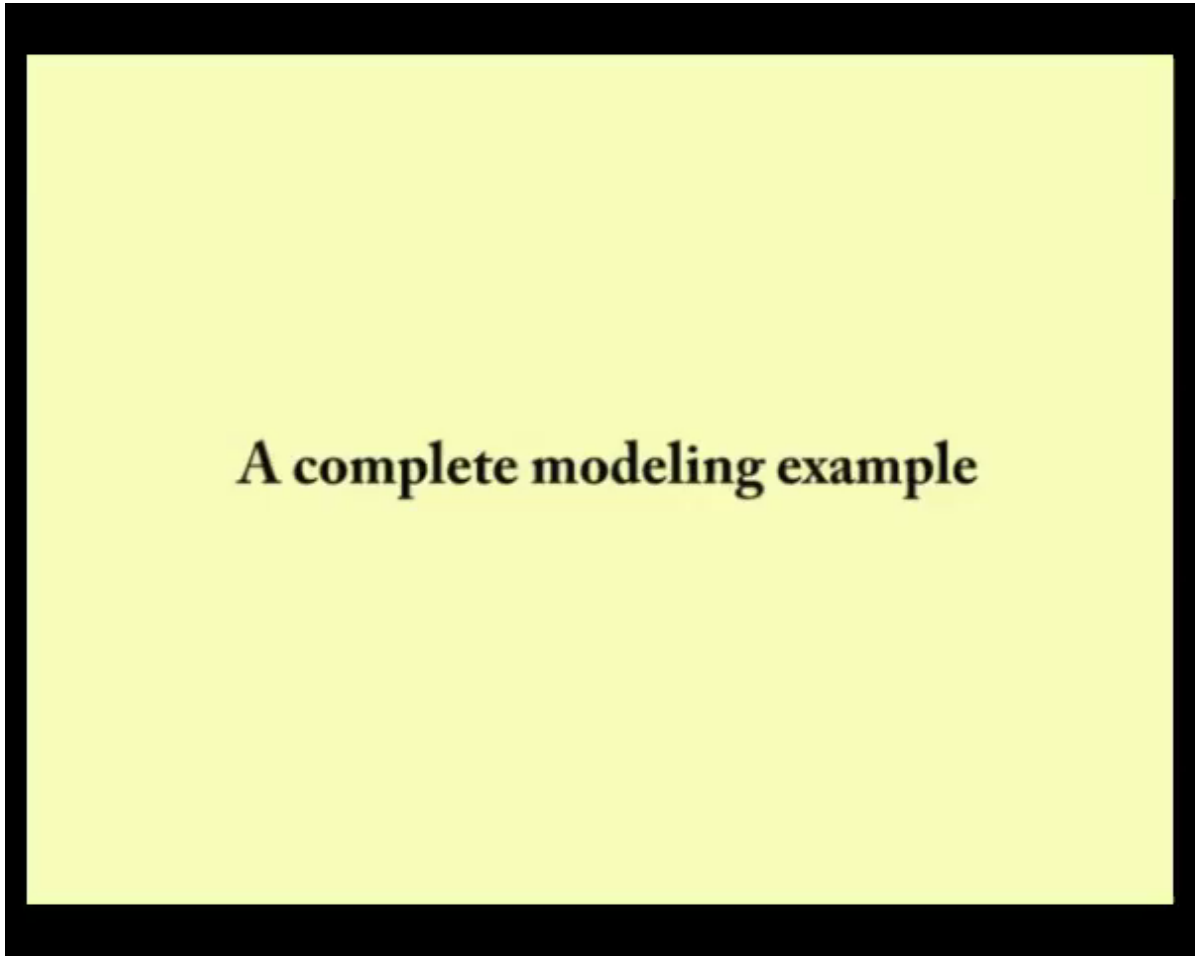


An Example



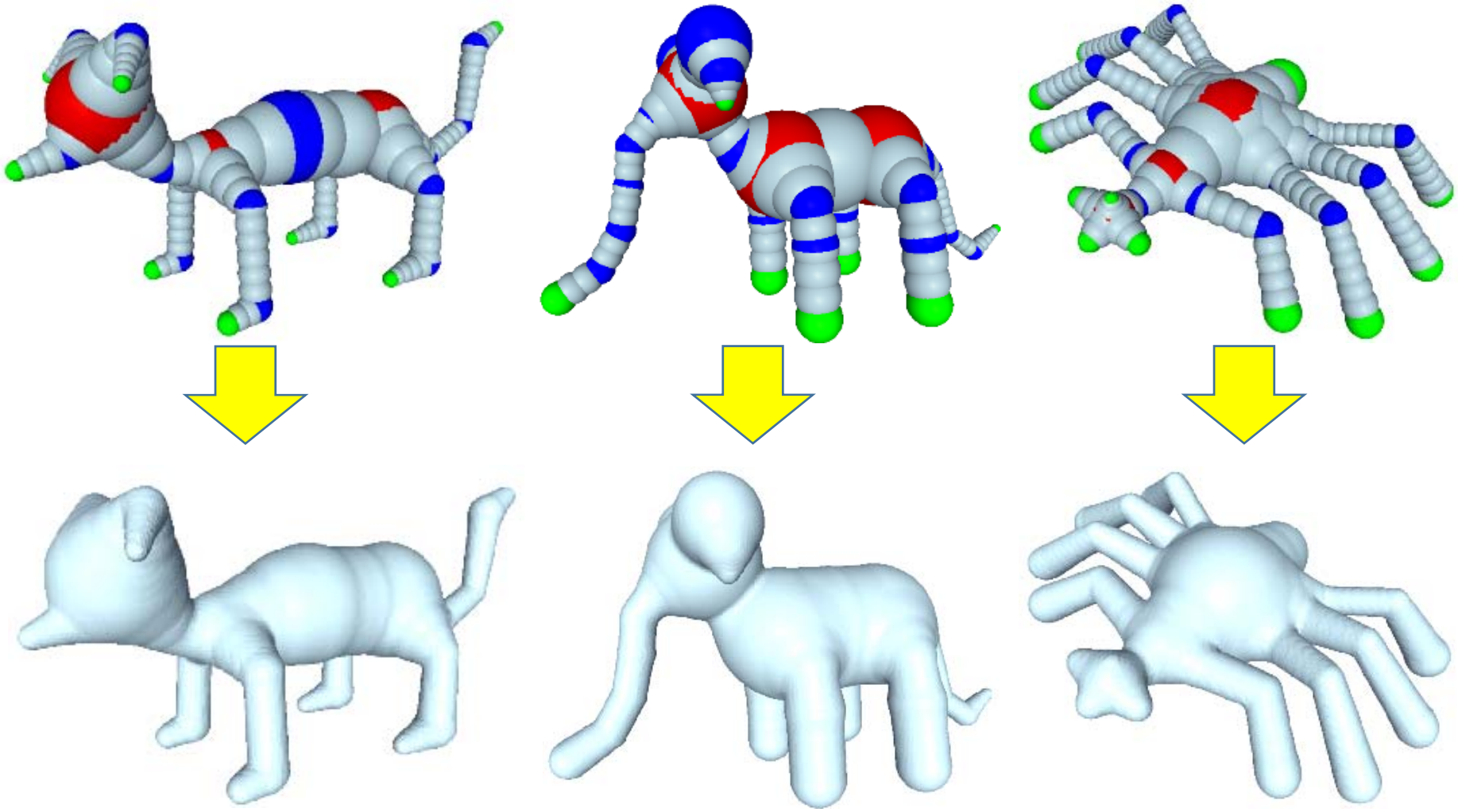
B-Mesh

[Ji et al. 2010]

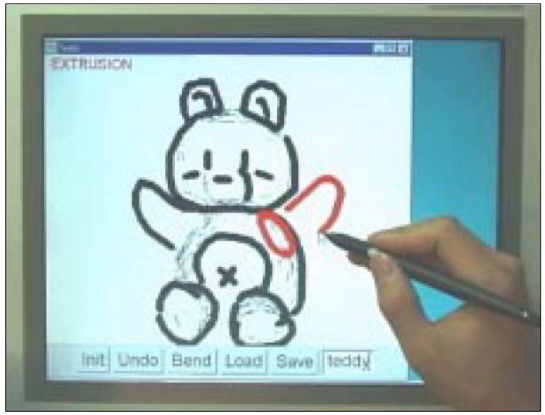


B-Mesh

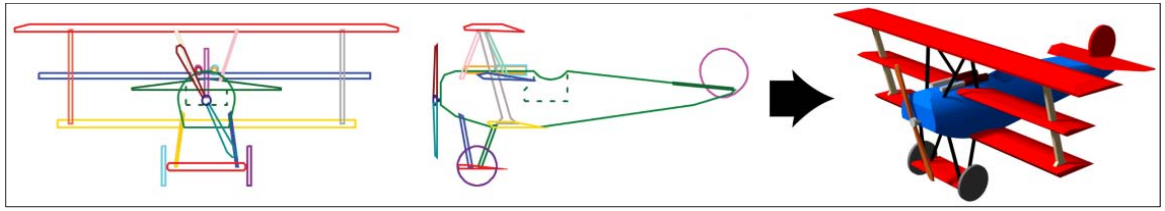
[Ji et al. 2010]



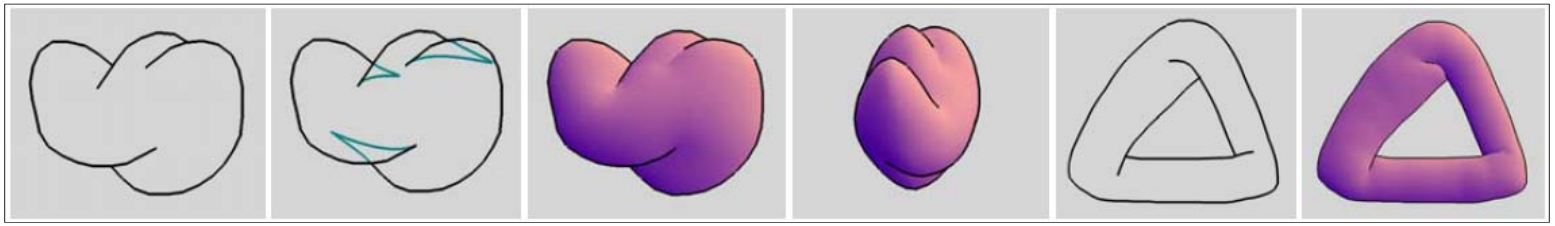
Sketch-based Modeling Systems



[Igarashi et al. 1999]



[Rivers et al. 2010]



[Karpenko et al. 2006]

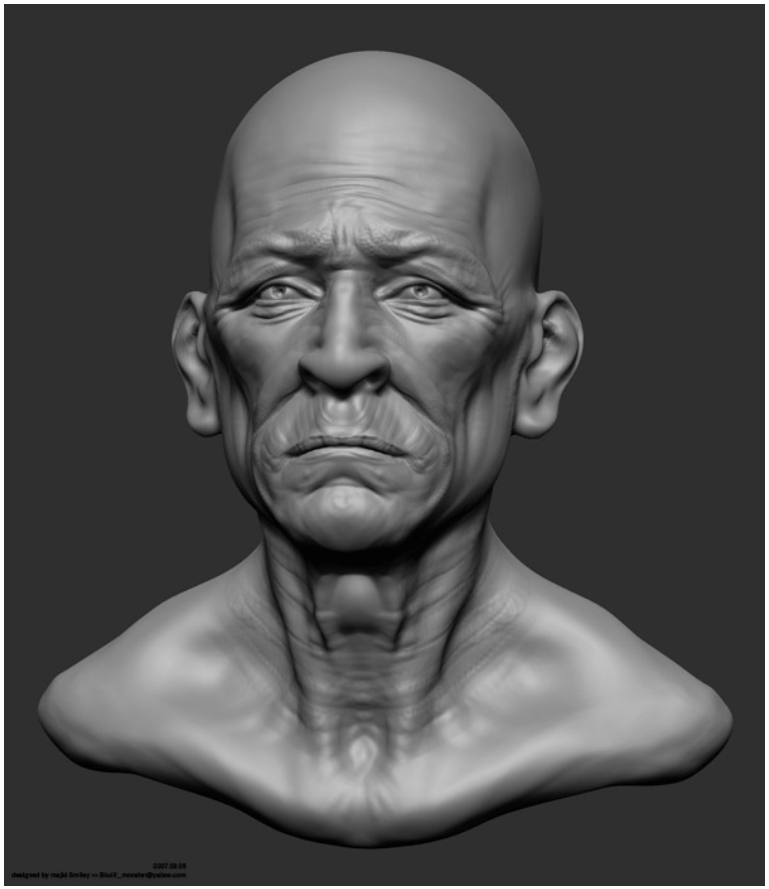
More…

Sketch2Scene [Xu et al. 2013]

Sketch-to-Design [Xie et al. 2013]

6. Sculpturing Modeling

Highly Detailed Meshes



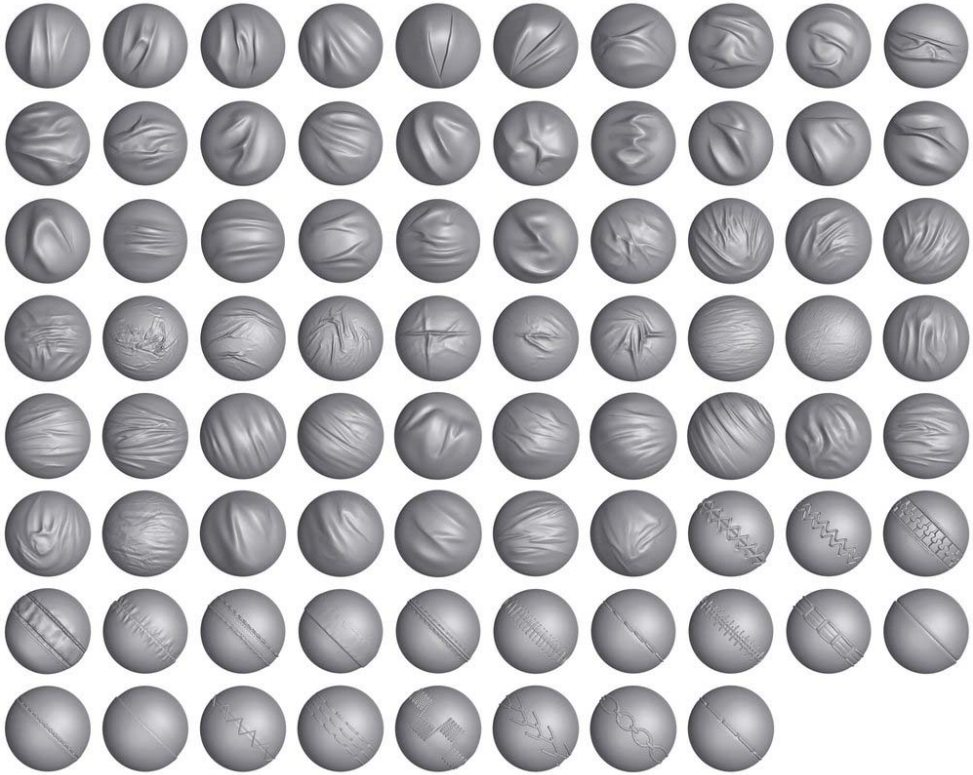
Motivation: Sculpturing

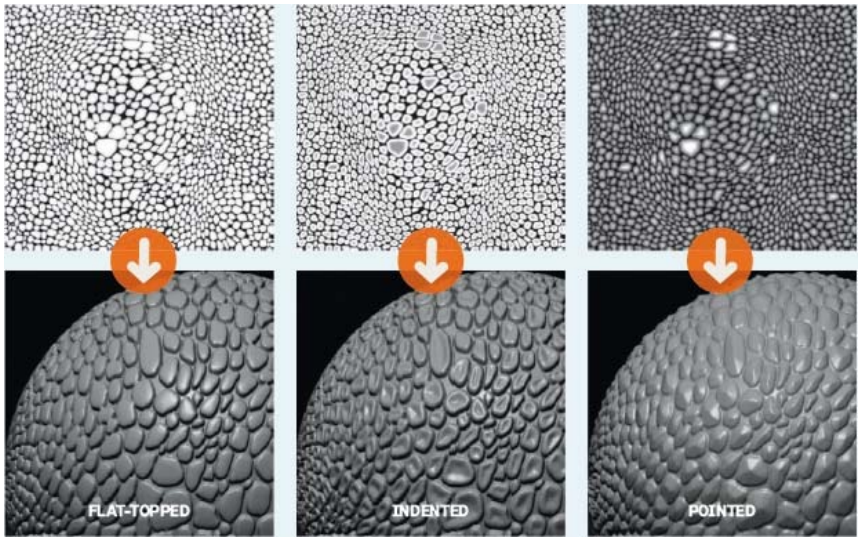
Sculpturing Modeling

- A series of sculpting/brush operators

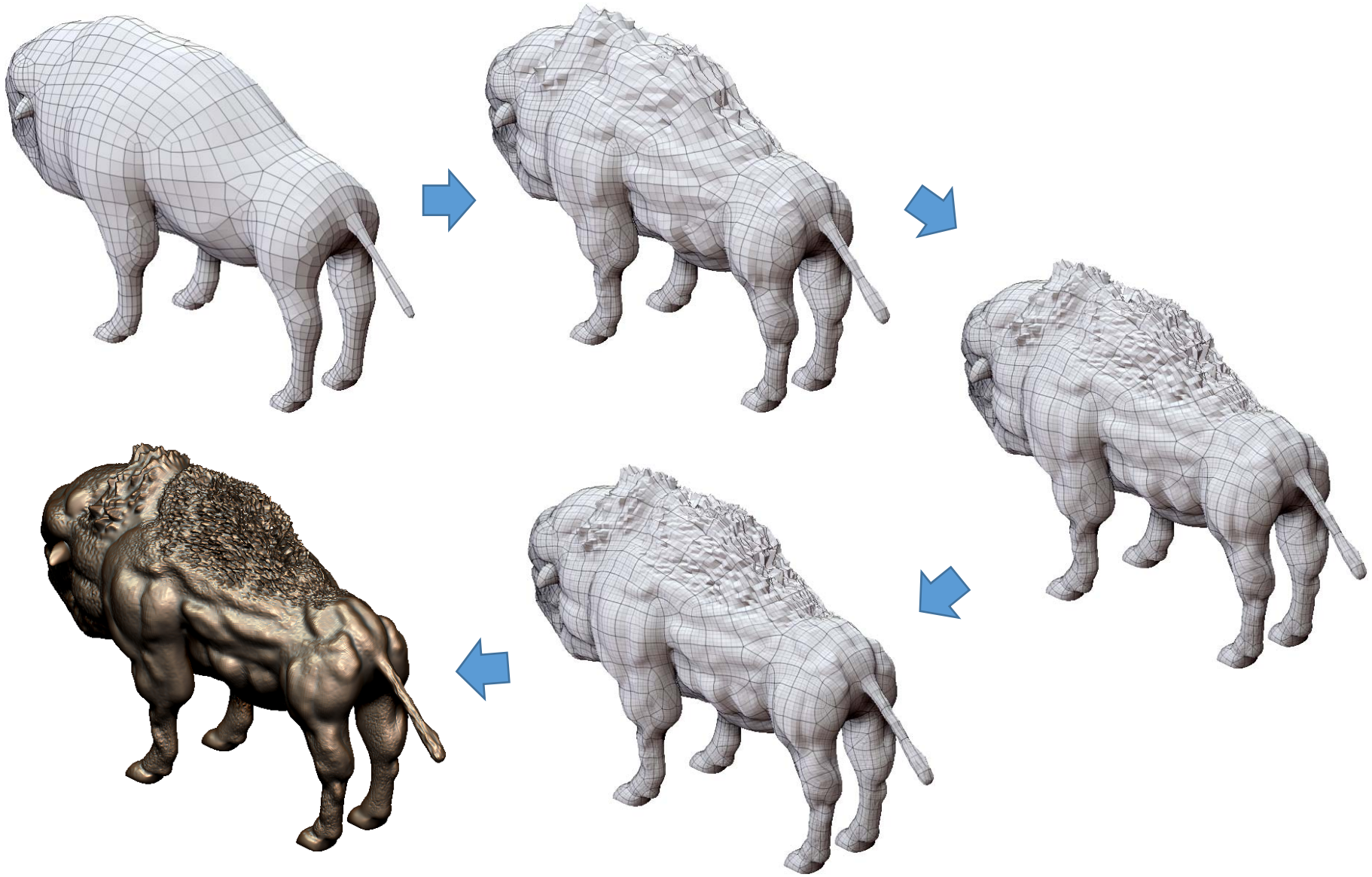
Sculpting Modeling Tools

3D Brushes





Sculpting Modeling



Summary: Shape Editing

- Points
  - Points, vertices, …
- Lines/Curves
  - Sketches, skeletons, silhouettes, wires, …
- Meshes
  - Bezier nets, lattices, cages, …
- Other shapes
  - Deformation transfer: learning deformation from other shapes
- Physically-based shape deformation (*)
  - Material-based, FEM, simulation

中国科学技术大学
University of Science and Technology of China

Data-Driven Modeling

三维几何模型数据库

1. Modeling by Examples

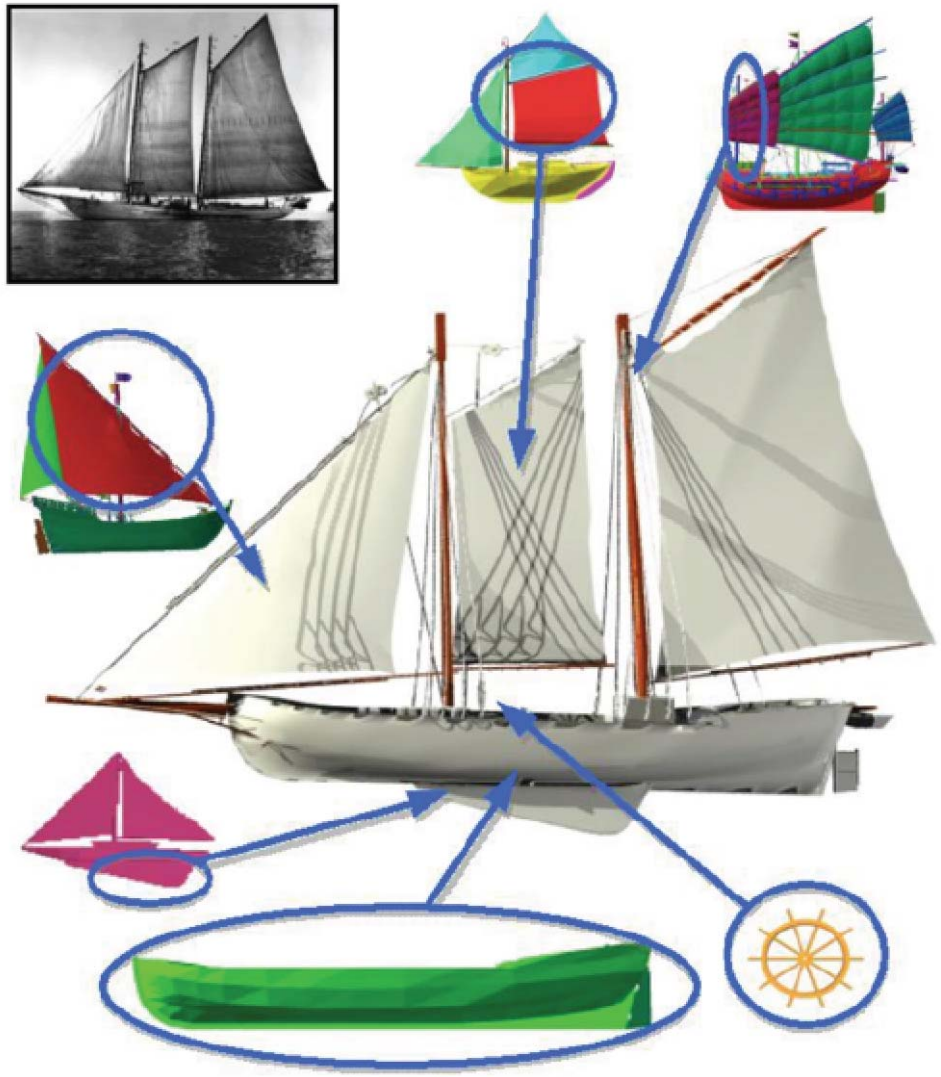
[Funkhouser et al. Siggraph 2004]

1. Segment into parts
2. Search 3D database for parts
3. Compose parts into a model

Modeling by Examples

[Funkhouser et al. Siggraph 2004]

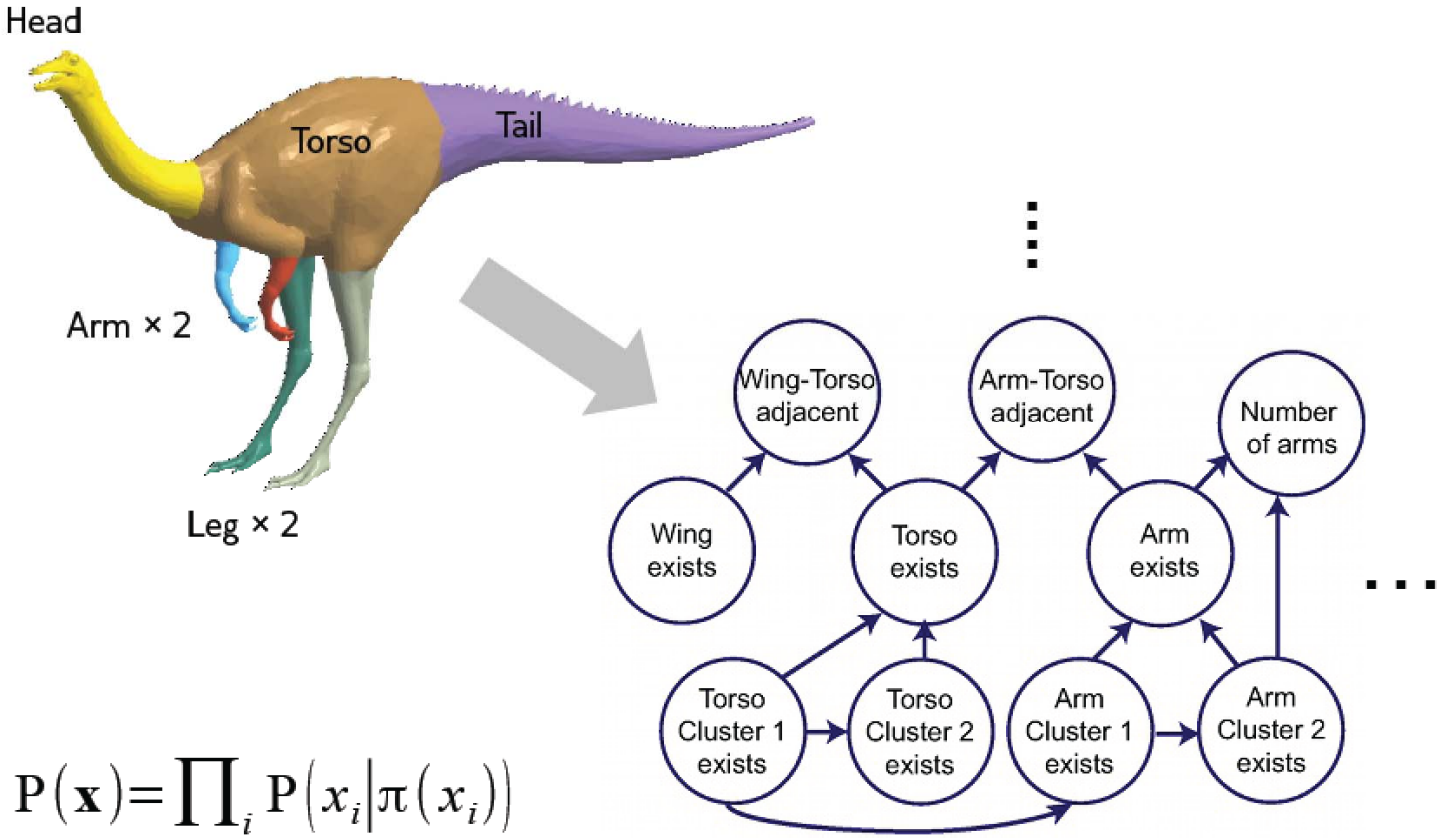
1. Segment into parts
2. Search 3D database for parts
3. Compose parts into a model



2. Probabilistic Reasoning

[Chaudhuri et al. Siggraph 2011]

- Design space

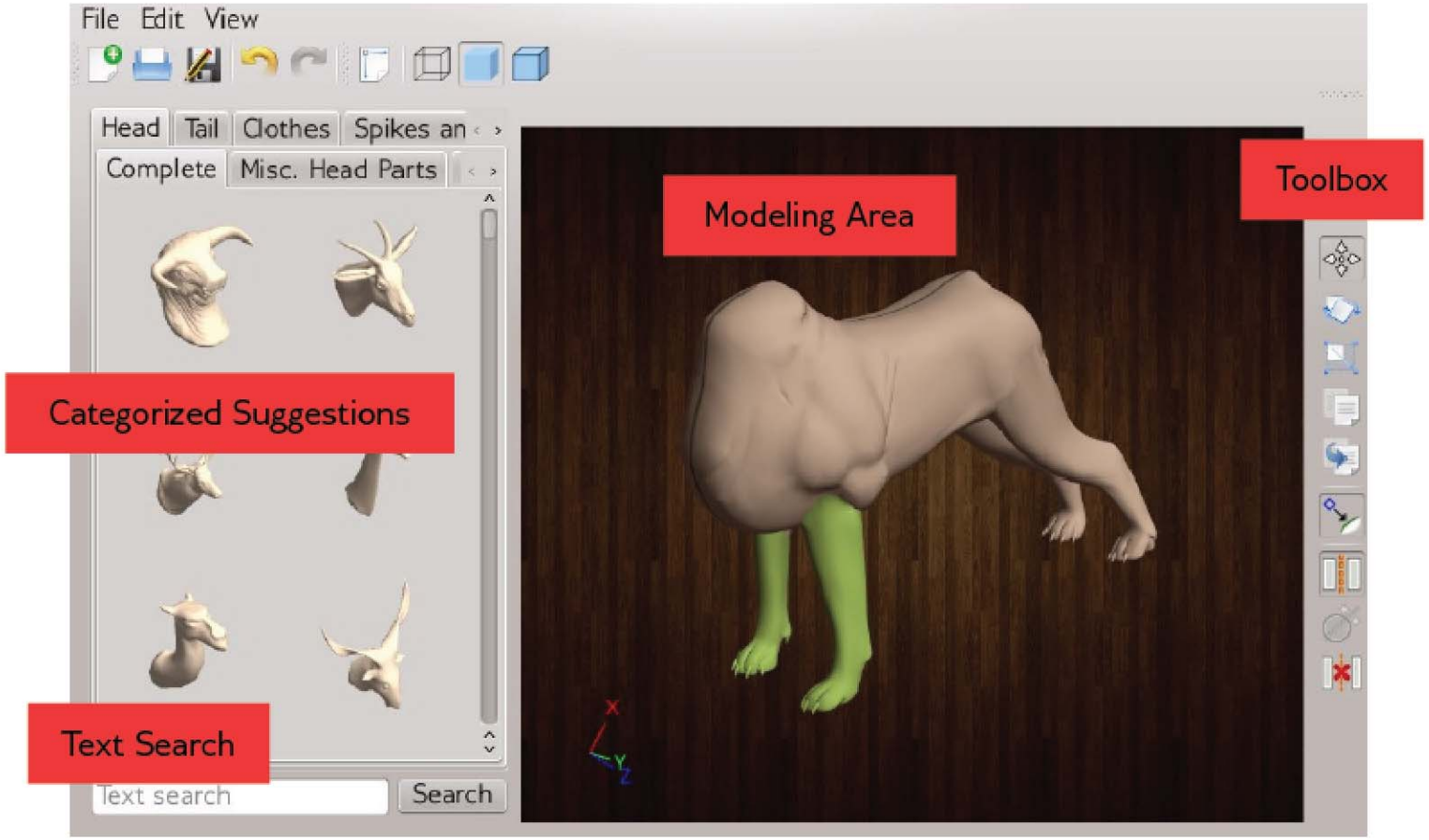


$$\mathrm{P}(\mathbf{x}) = \prod_i \mathrm{P}\left(x_i \middle| \pi(x_i)\right)$$

2. Probabilistic Reasoning

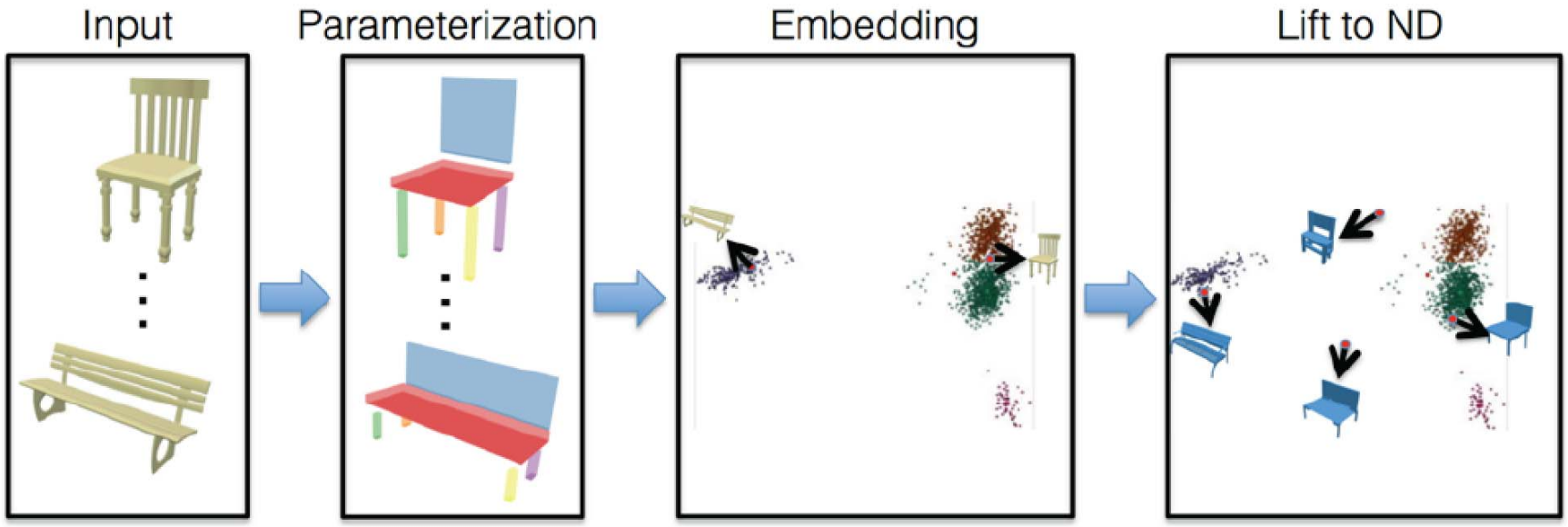
[Chaudhuri et al. Siggraph 2011]

- Exploring the design space



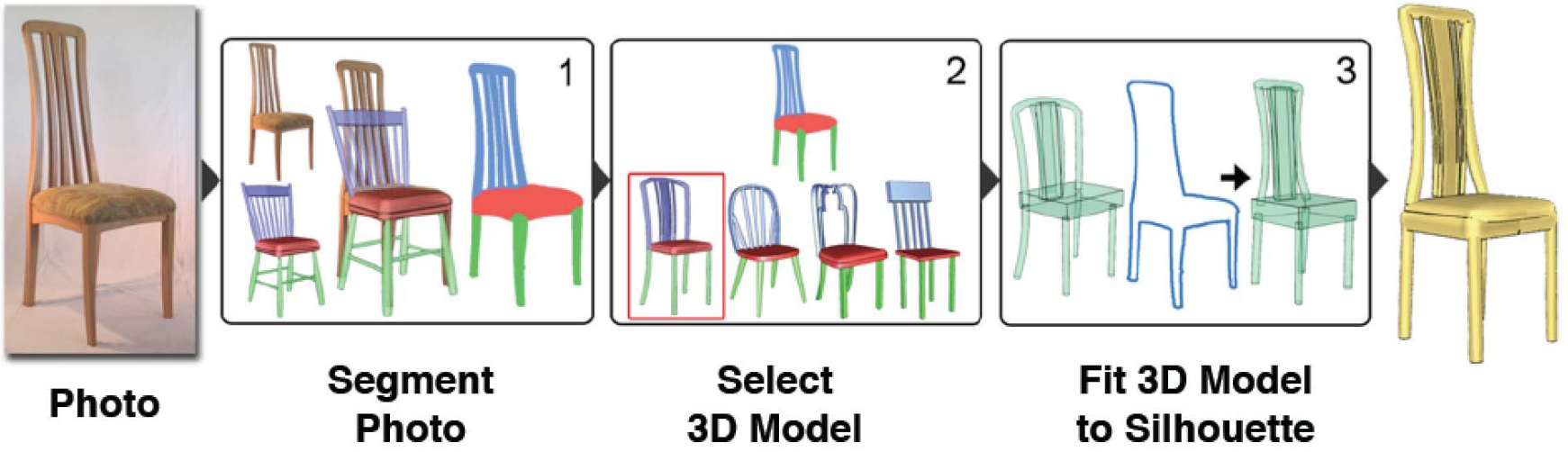
3. ShapeSynth: Exploration as Design Tool

[Averkiou et al. Siggraph 2014]



4. Photo-inspired Modeling

[Xu et al. Siggraph 2012]

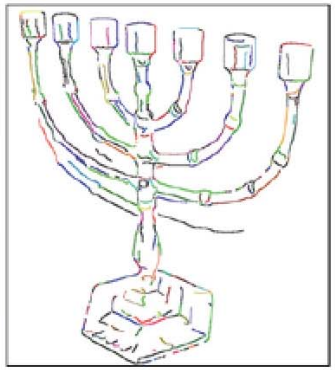


Examples

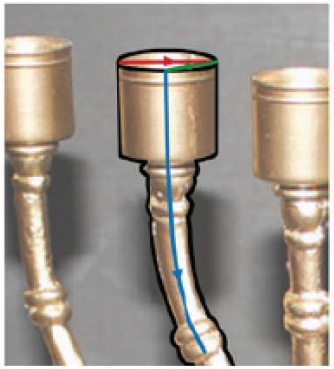
5. Primitive-based Modeling

Chen et al. 2013

Input



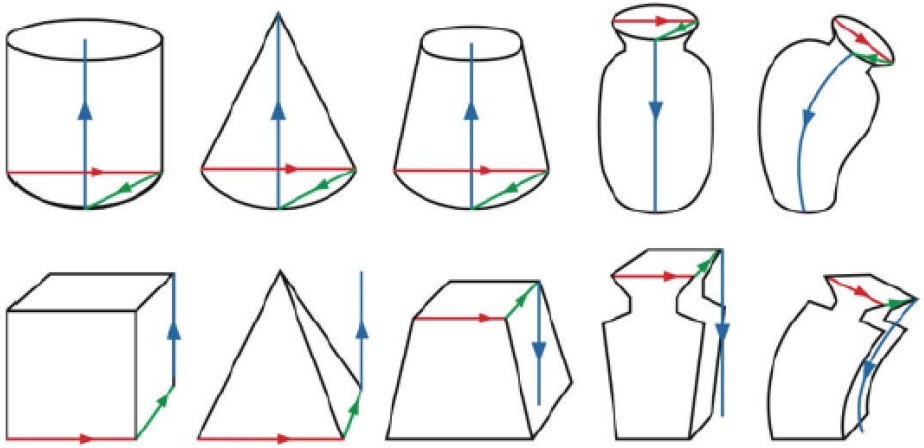
Edges



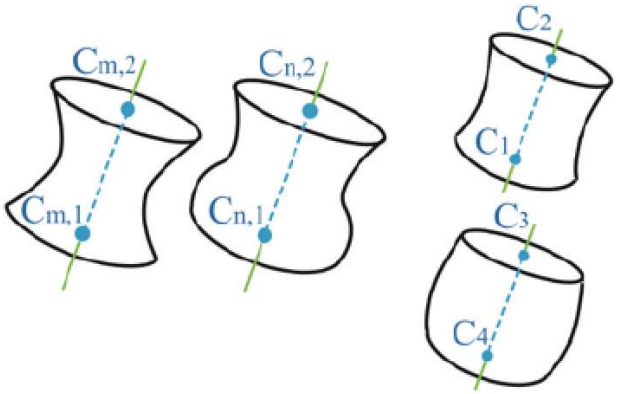
User Input

Reconstruction

Edited Picture



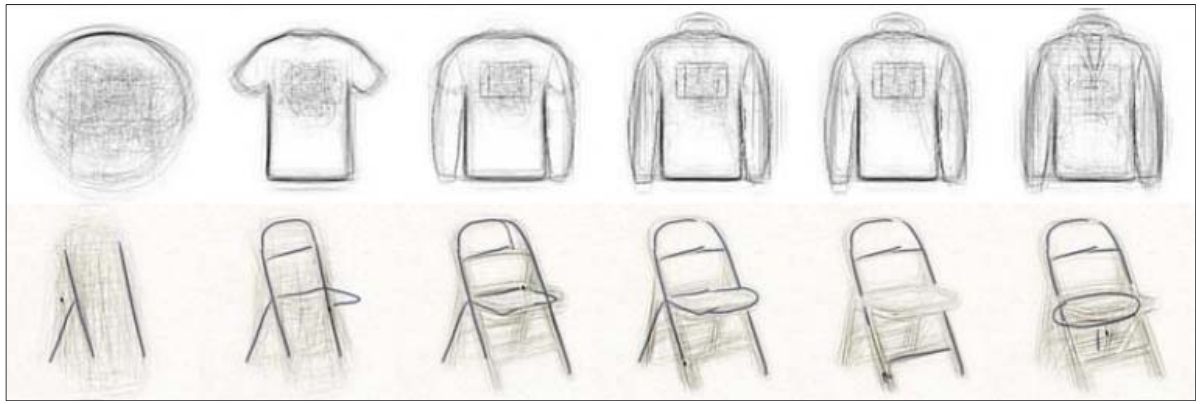
Generalized Cylinders and Cuboids



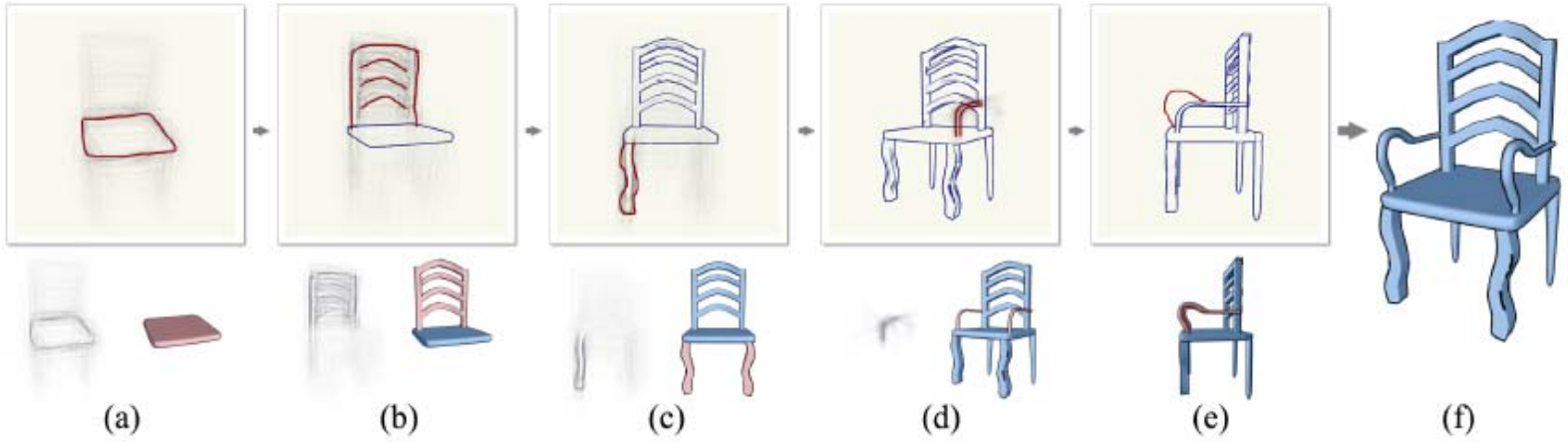
Parallelism Co-planarity

6. Shadow Guidance

Fan et al. 2013

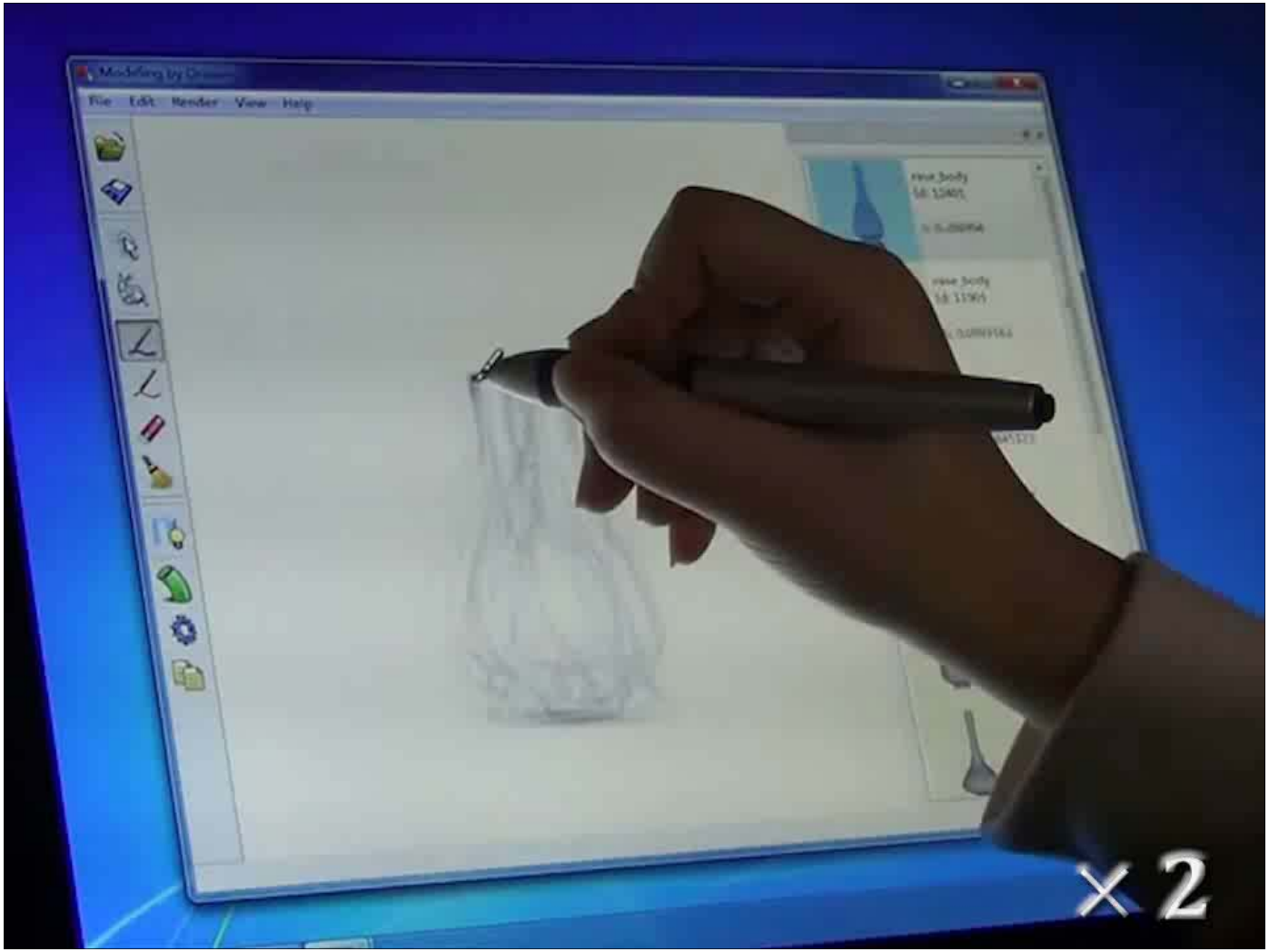


ShadowDraw [Lee et al. 2011]



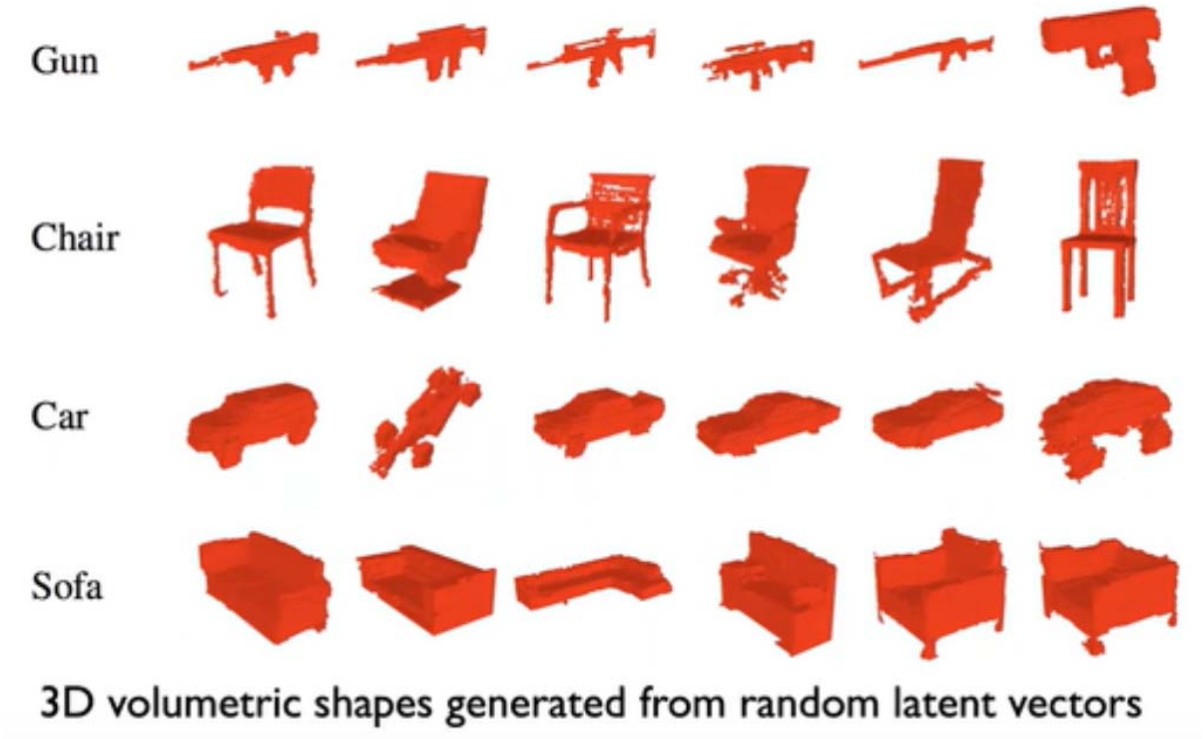
6. Shadow Guidance

Fan et al. 2013



7. Learning Generative Models of 3D Shapes (3D-GAN)

- Emergence of various 3D-GANs for shape generation



3D volumetric shapes generated from random latent vectors

Chinagraph 2020 Key note talk by Hao Zhang:
https://www.bilibili.com/video/BV1Pa4y1x7au?p=3

Chaudhuri, et al. Learning Generative Models of 3D Structures. Computer Graphics Forum (Eurographics STAR), 2020.

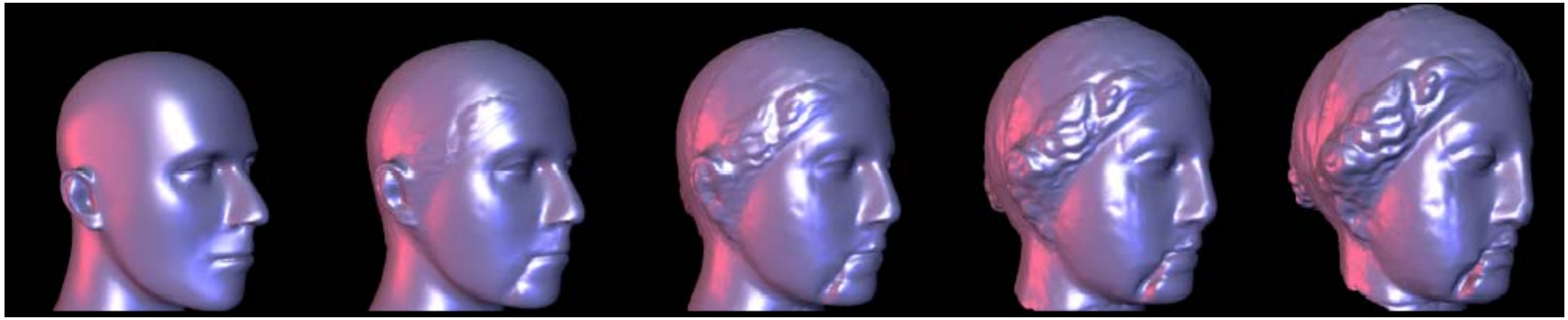
中国科学技术大学

University of Science and Technology of China

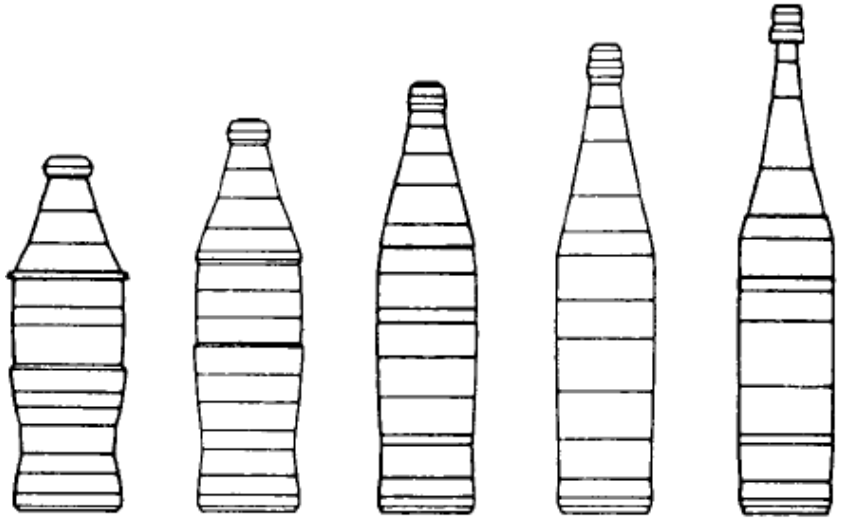
Shape Morphing

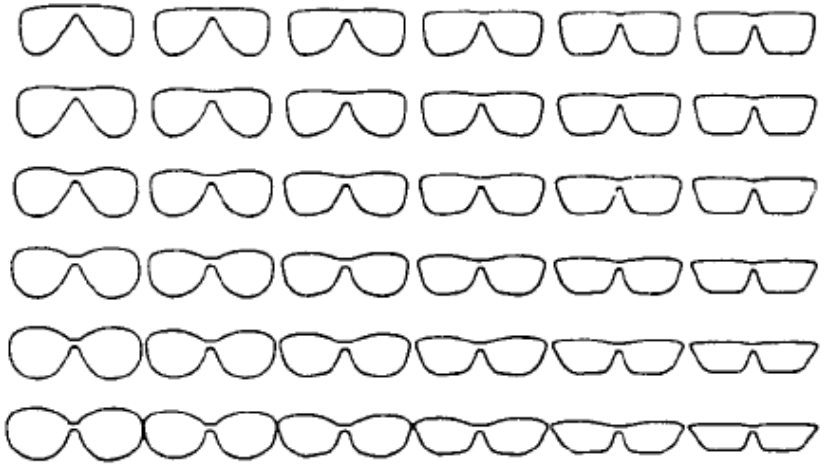
Shape Morphing

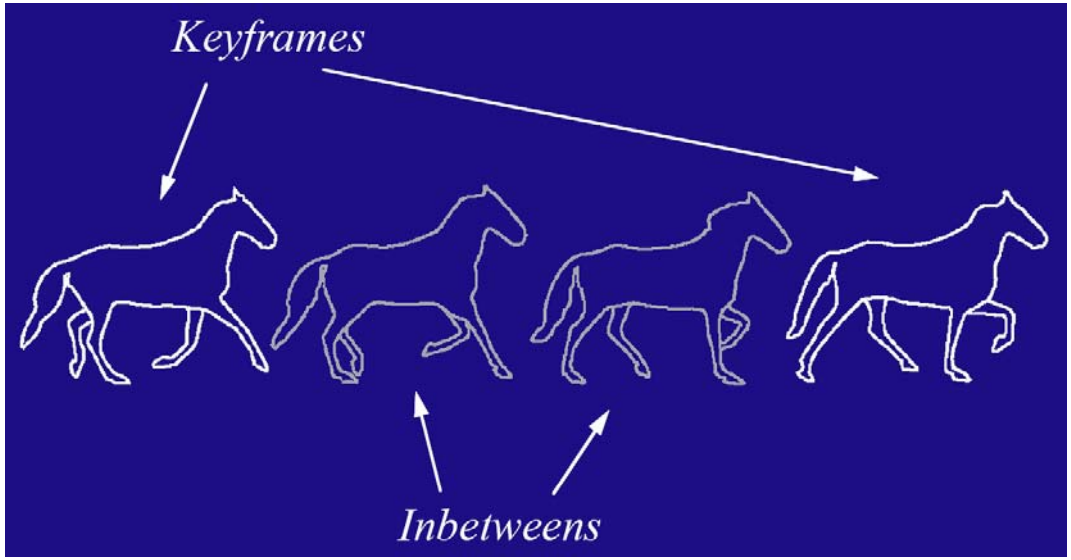
- Given two objects produce sequence of intermediate objects that gradually evolve from one object to the other
  - Interpolate object shapes
  - Interpolate object attributes
    - Color, texture, normal, etc.



Applications

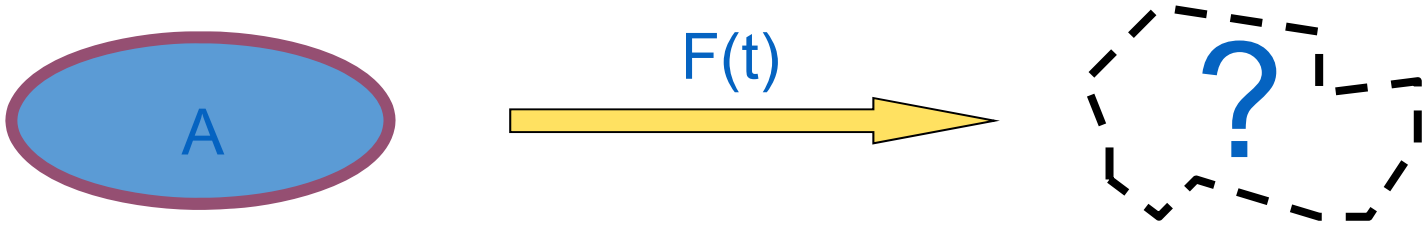




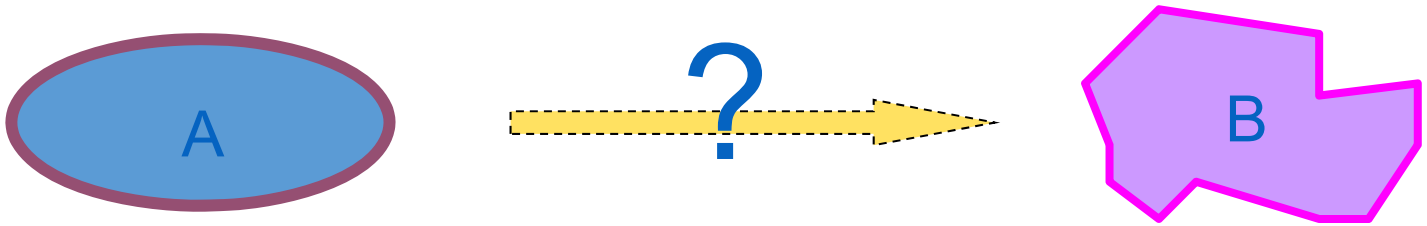


Shape Transformations

- Warping: Unary Op
  - Given Object A and F(t), find Object B



- Morphing: Binary Op
  - Given Object A and Object B, find F(t)



Terminologies

- Morphing
- Metamorphosis
- Shape blending
- Shape averaging
- Shape interpolation
- Shape transition

Rules for Good Morphing

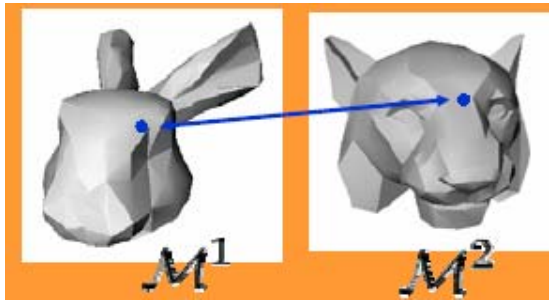
- Natural
  - Keep as much as possible of the two shapes during the transformation
    - Volume, curvature, area, etc...
  - Subjective aesthetic criteria
- User control
  - intuitive
  - not too heavy
  - can be adapted to user's knowledge

Morphing

- Input: two meshes source & target
  - Frames at t_0 and t_n
- Output: sequence of intermediate meshes
  - Frames t_1 to t_{n-1}
- Intermediate mesh:
  - For each point on source/target model specify location at time t_i consistent with source & target

Morphing: Two Sub-Problems

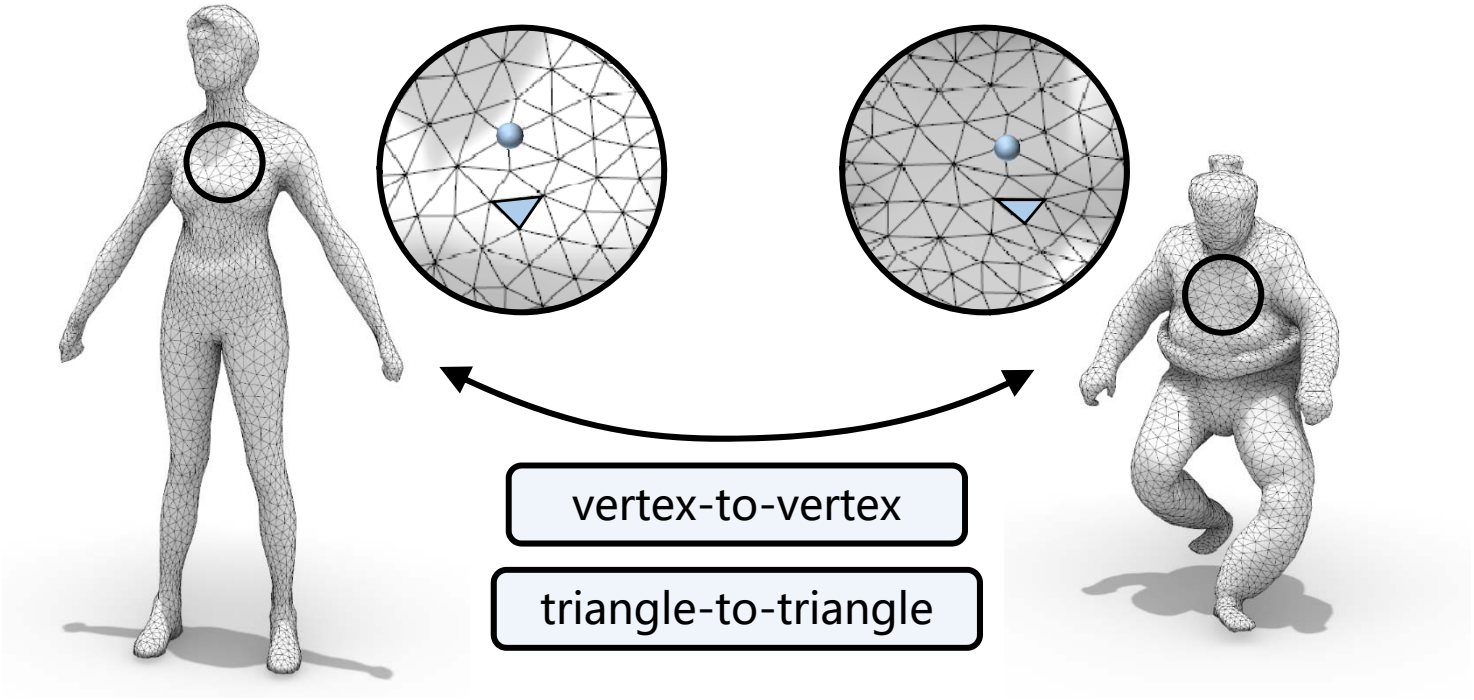
- Correspondence problem
  - Compatible meshes
  - For each point on source/target meshes find corresponding point on second mesh = Parameterization



- Path problem
  - Inbetween shapes
  - Specify trajectory in time for each point
    - For mesh – specify vertex trajectory

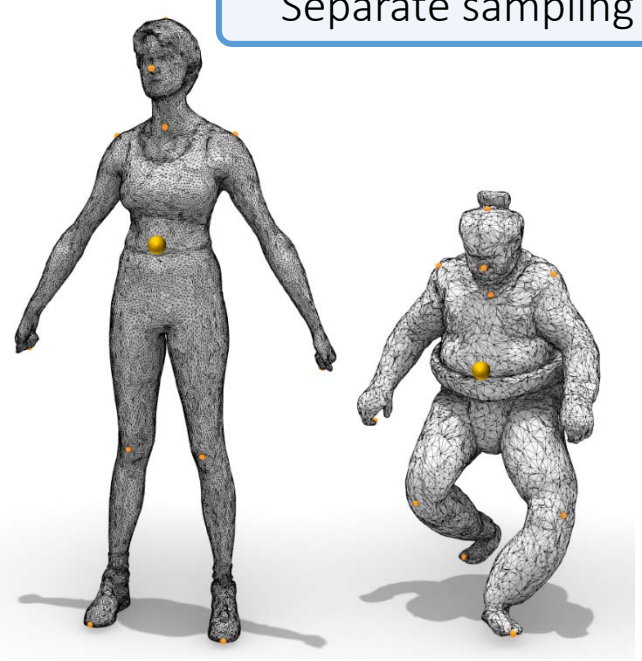
1. Vertex Correspondence

- Compatible meshes (consistent meshes, inter-map)
  - Each vertex on source mesh mapped to vertex on target (and vice versa)
  - Have common connectivity



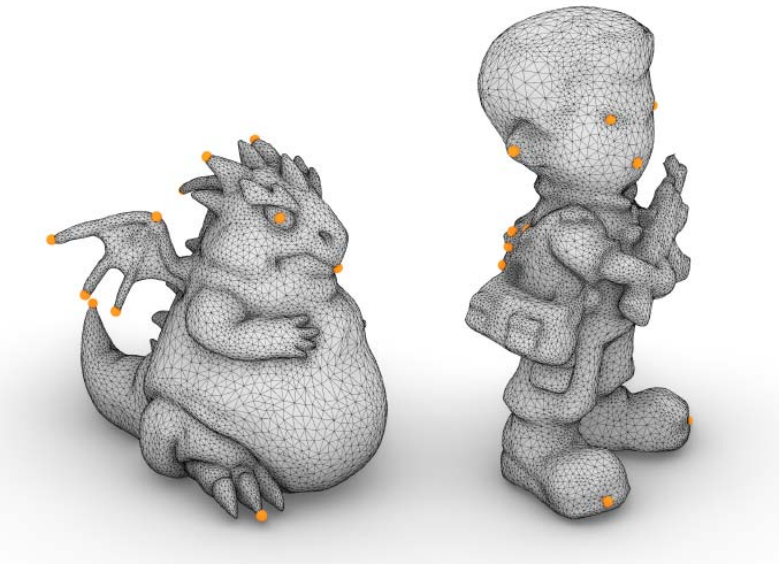
Challenges

Separate sampling



Homogeneous models

Different geometries



Heterogeneous models

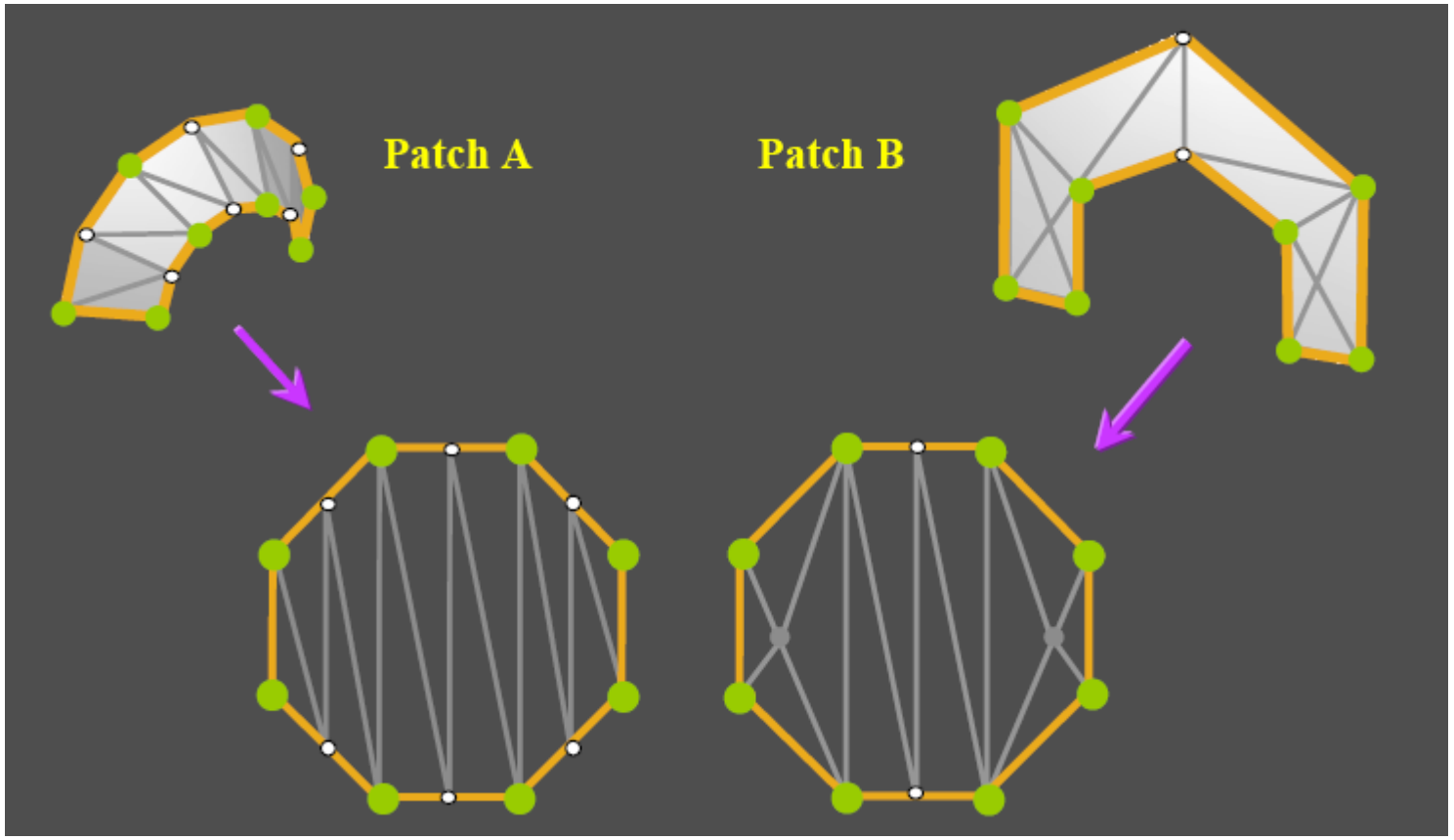
Correspondence: Parameterization

- To compute map between source mesh S and target mesh T parameterize both on common domain D:

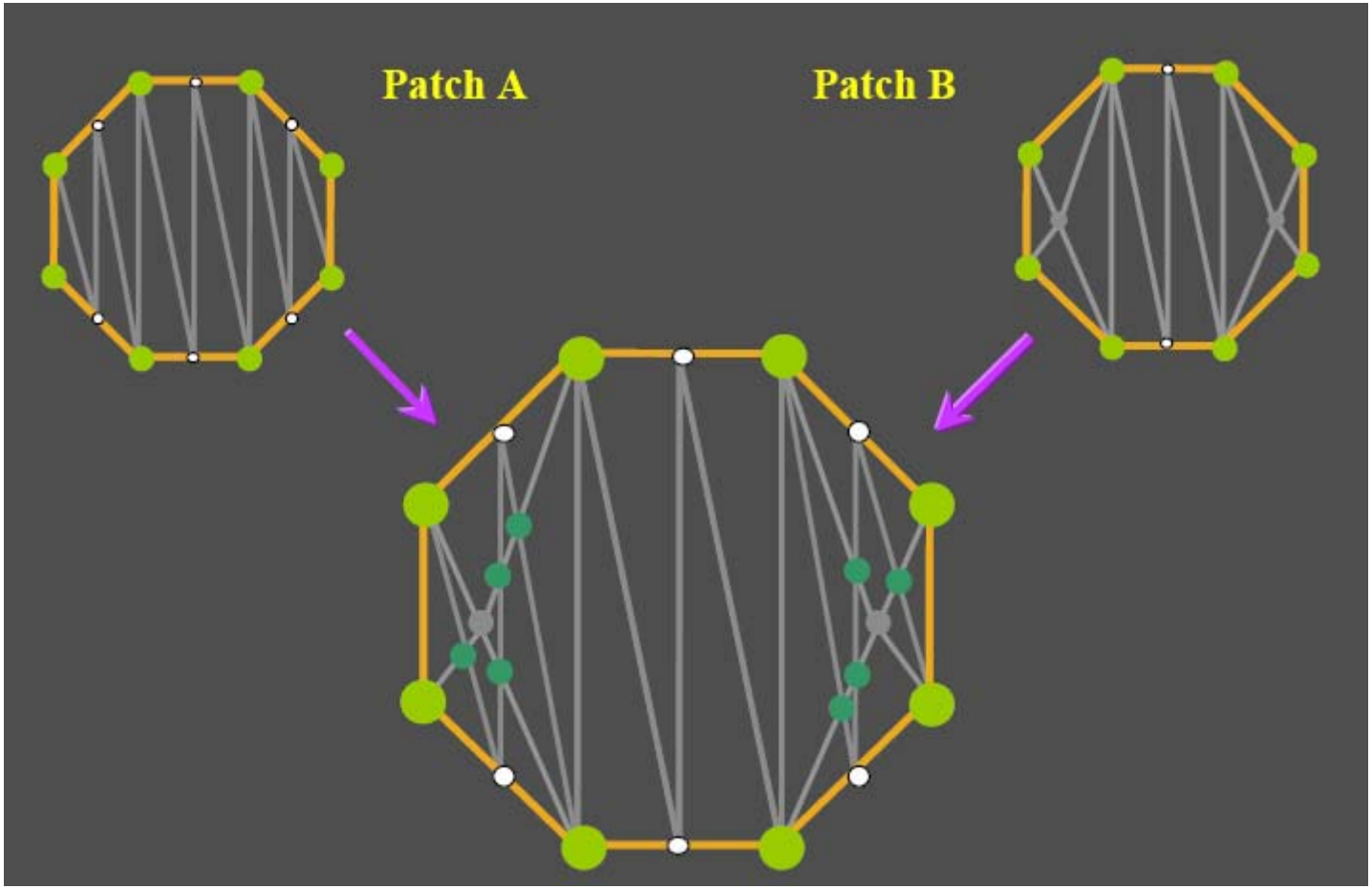
$$F_s : S \rightarrow D$$
$$F_t : S \rightarrow D$$
$$F_{st} = F_t^{-1} F_s$$

- Common domain options
  - 2D patch(es) – works for genus 0 + boundary
    - Use convex boundary (why?)
  - Sphere
  - Base mesh

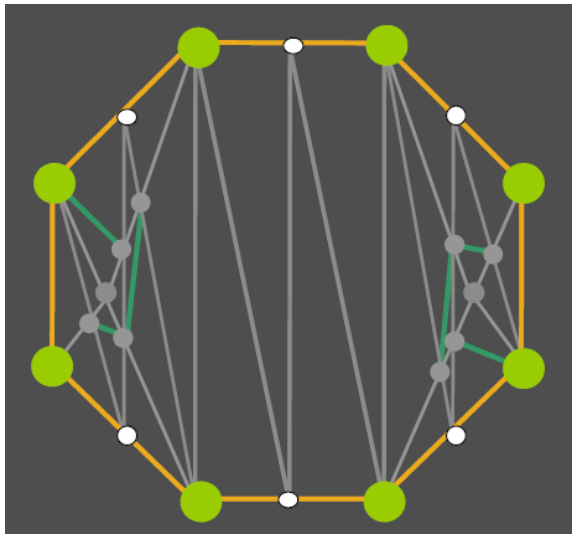
1.1 Parameterization: Planar domain

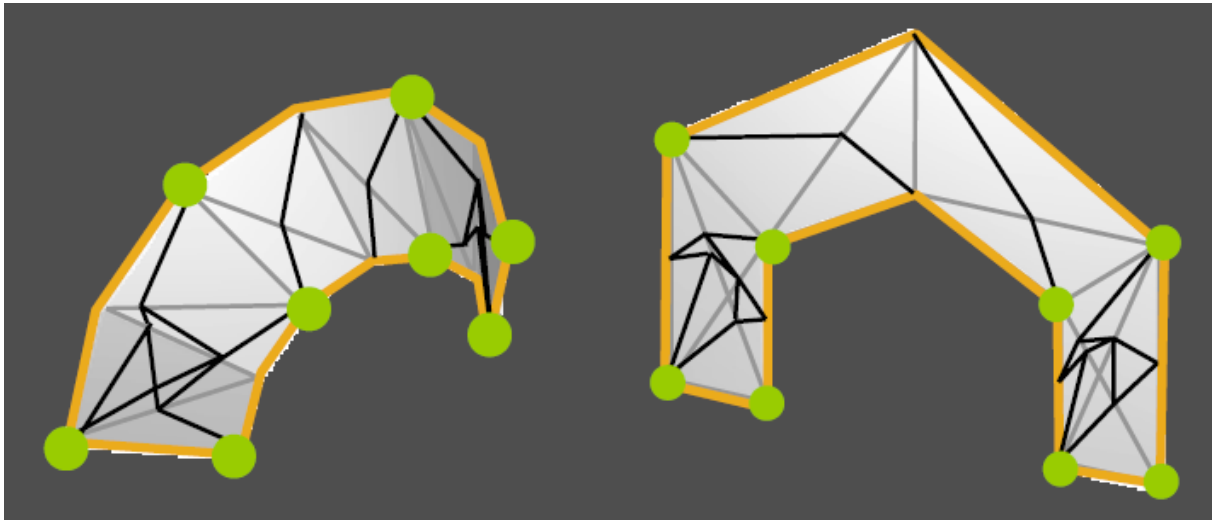


1.1 Parameterization: Planar domain

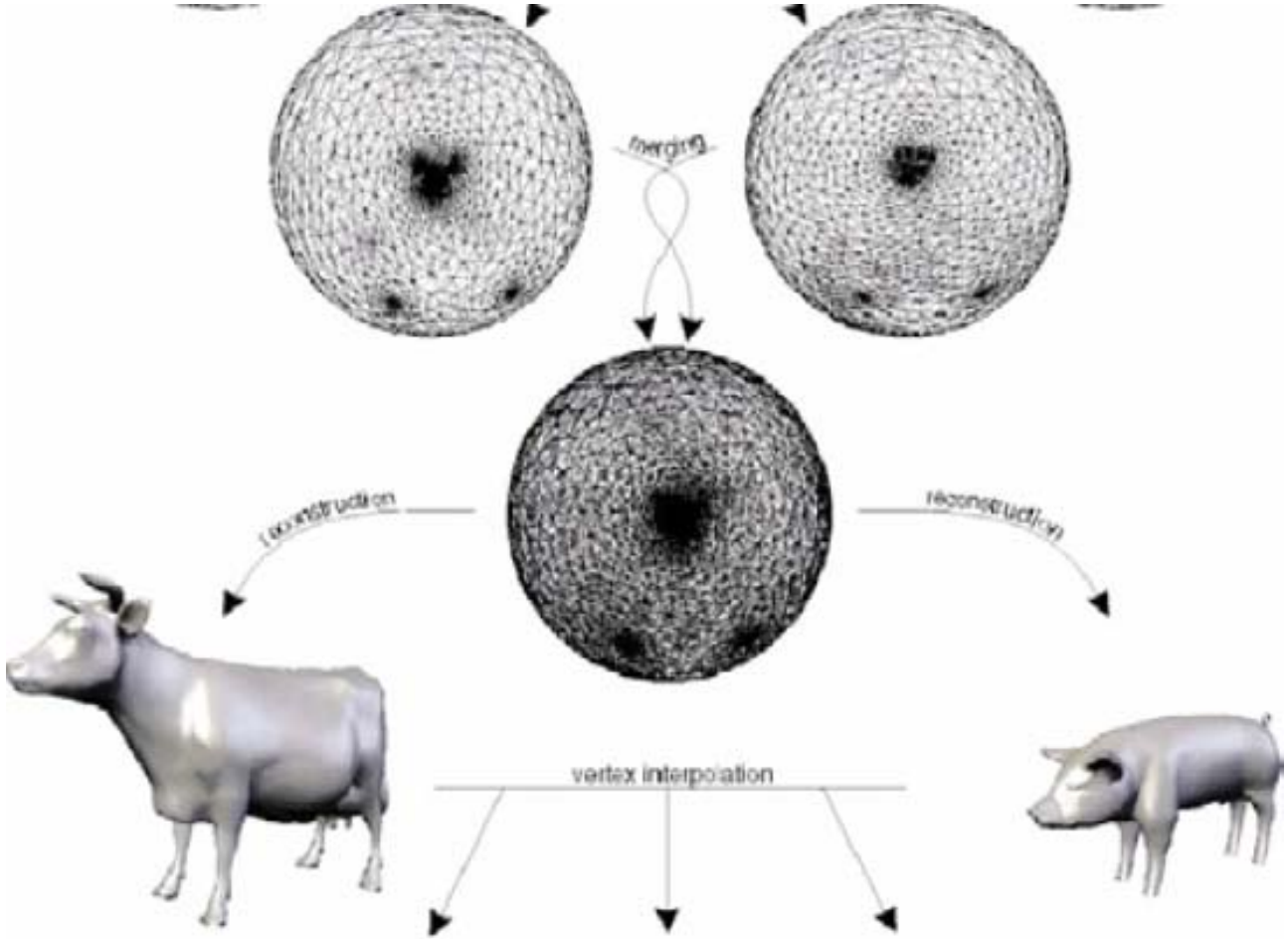


1.1 Parameterization: Planar domain





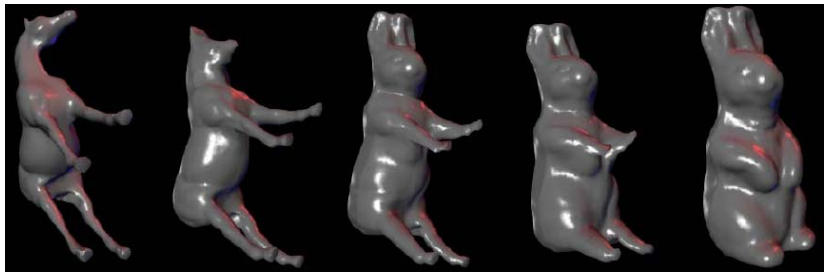
1.2 Parameterization: Sphere domain

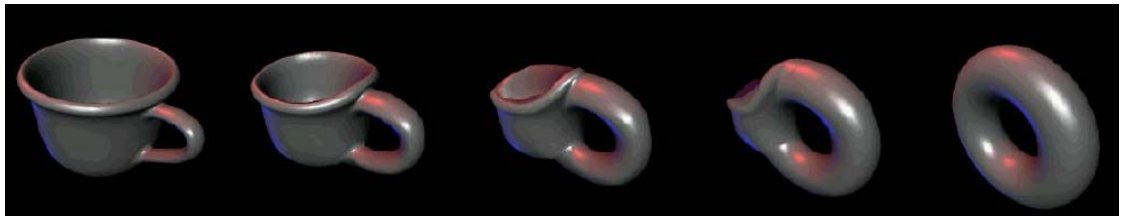


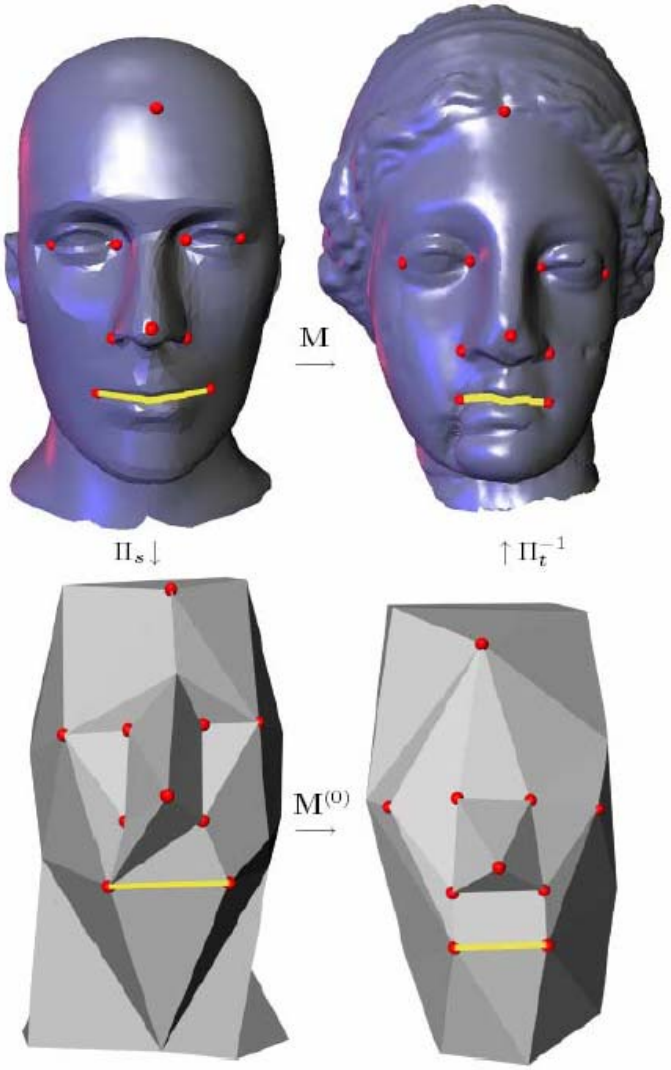
1.3 Parameterization: Base mesh

Lee et al. 1999

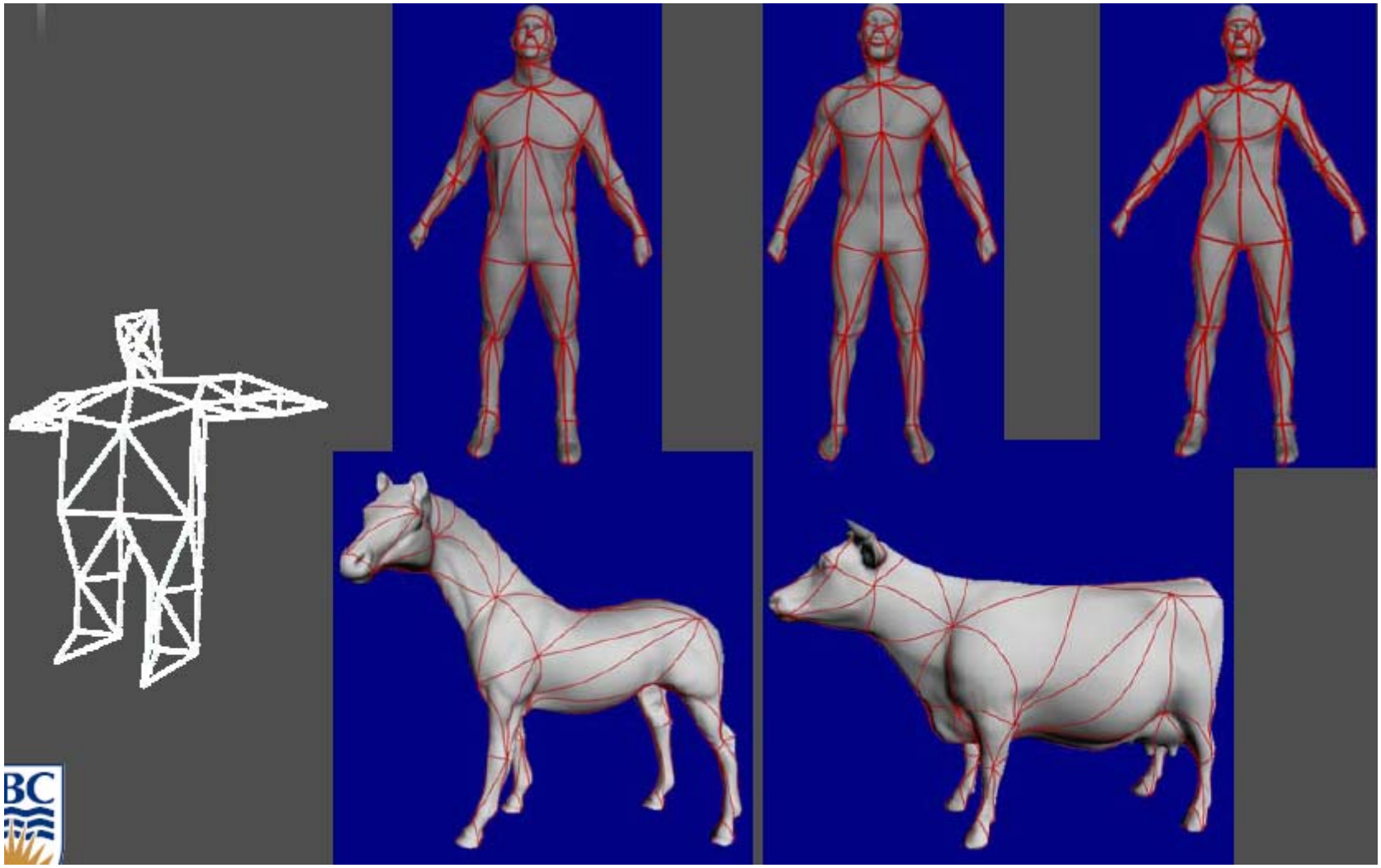
- MAPS



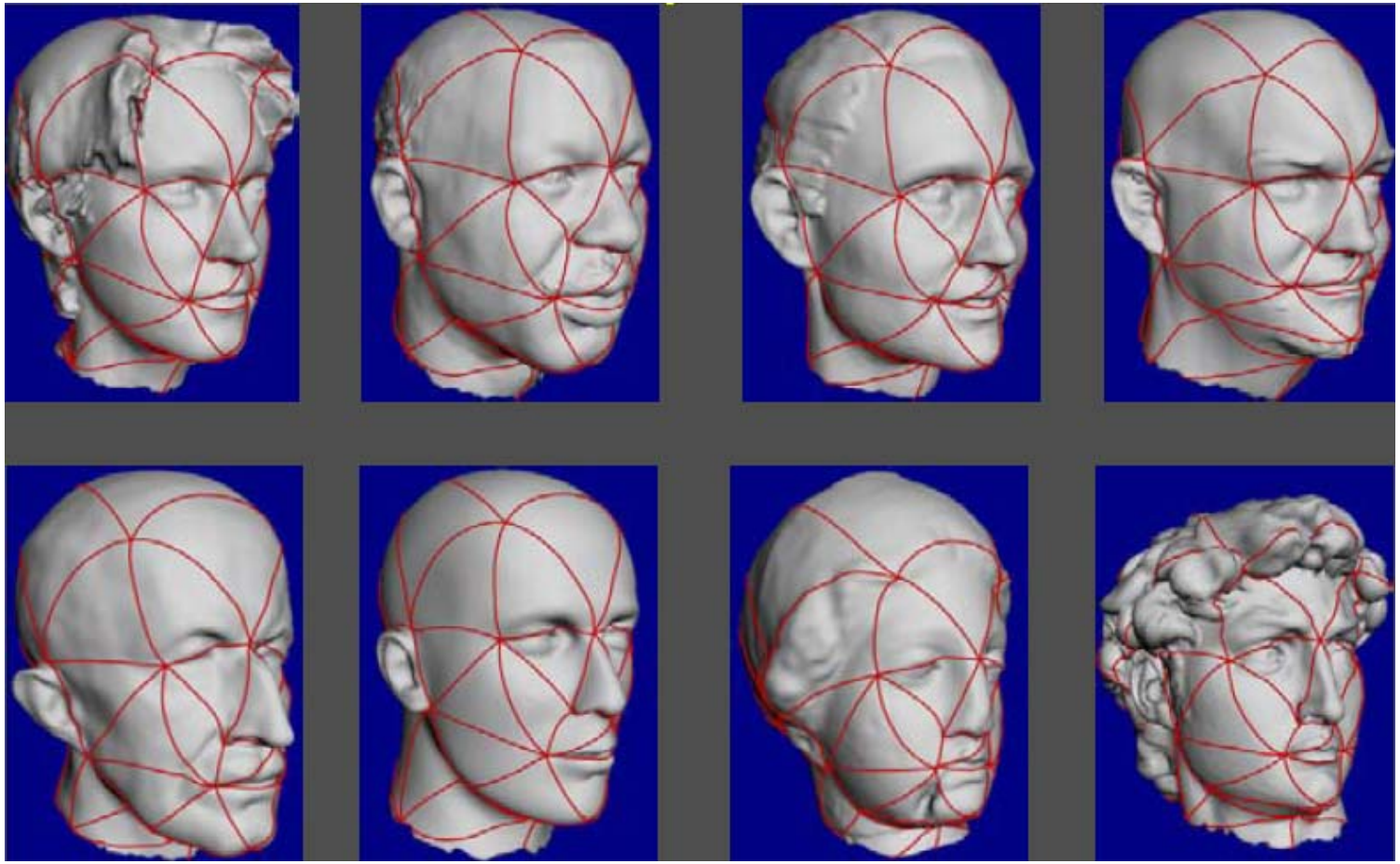




1.3 Parameterization: Base mesh

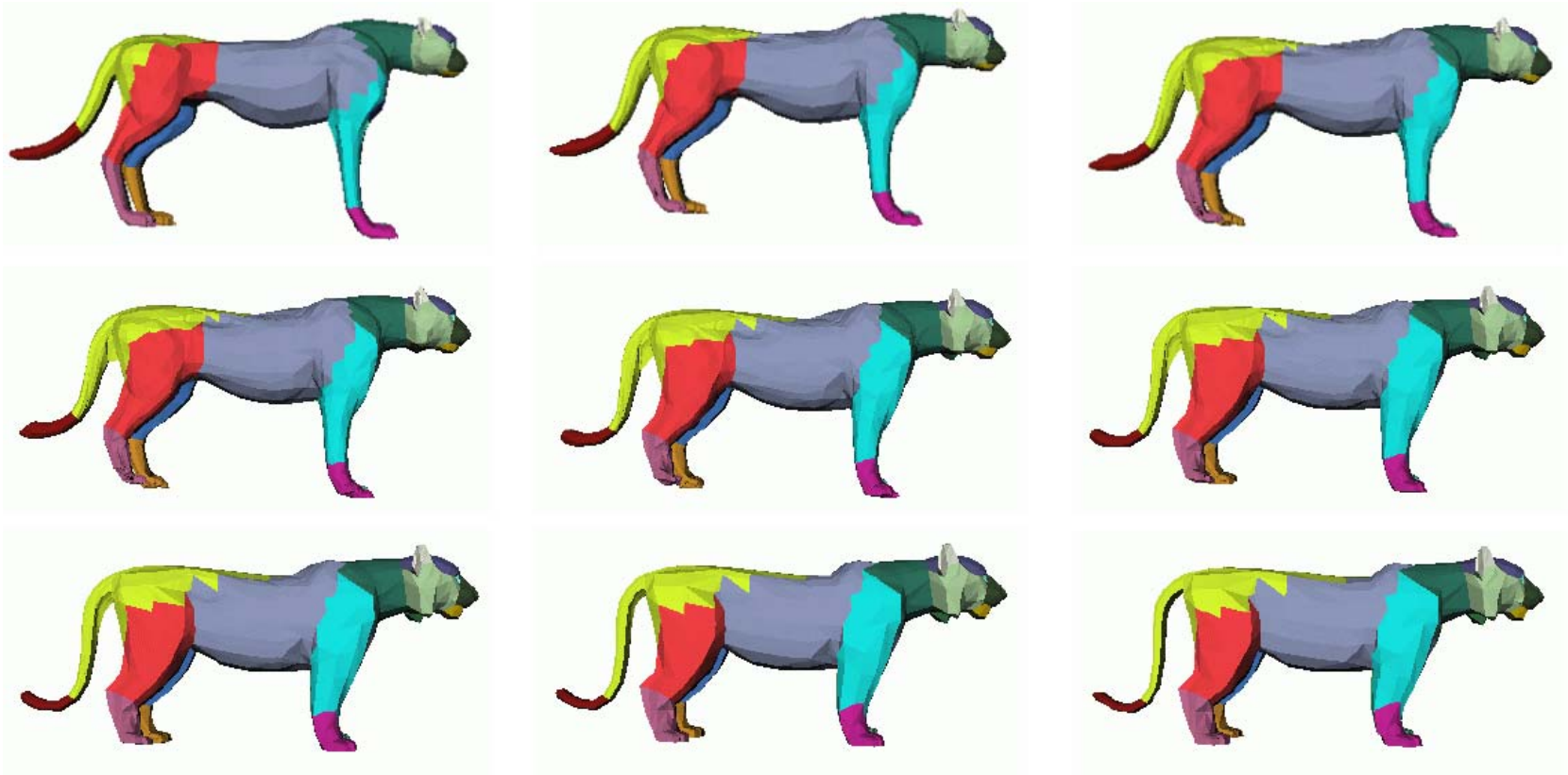


1.3 Parameterization: Base mesh



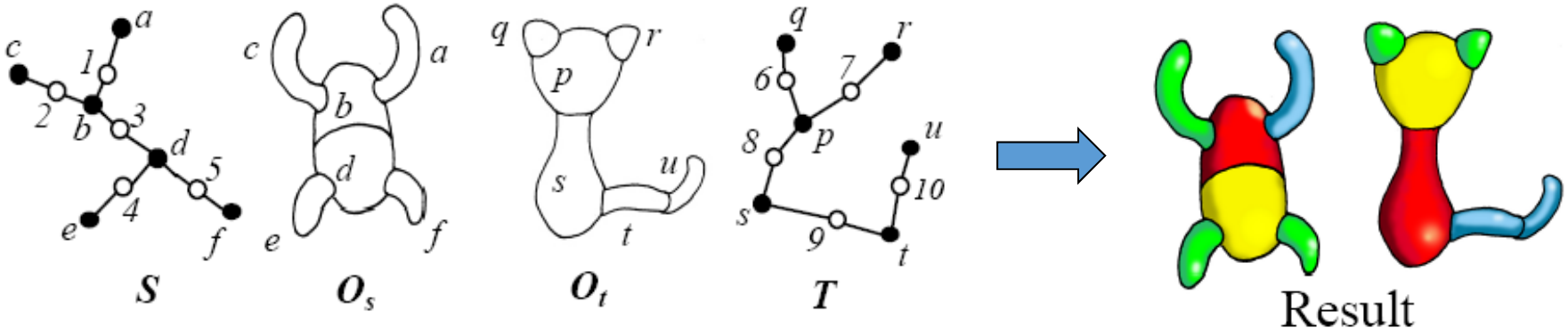
1.4 Decomposition Based

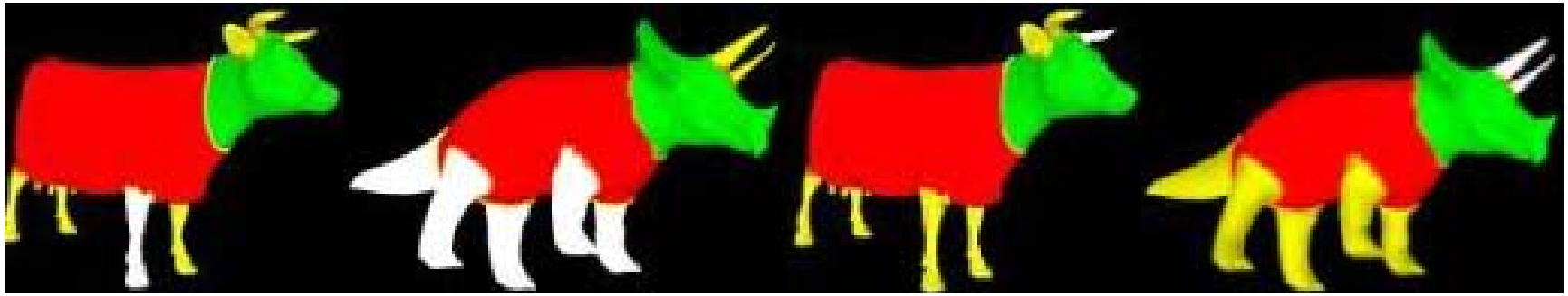
[Shlafman et al. 2002]



1.5 Component Based

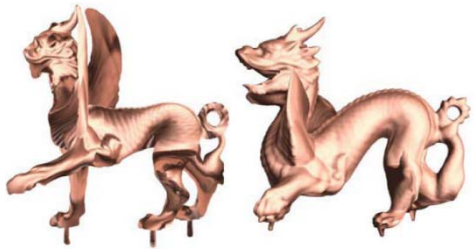
[Zhao et al. 2003]





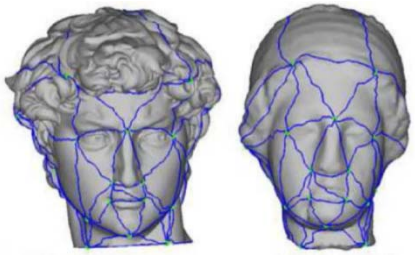
1.6 Many Recent Works

Overlaying-based

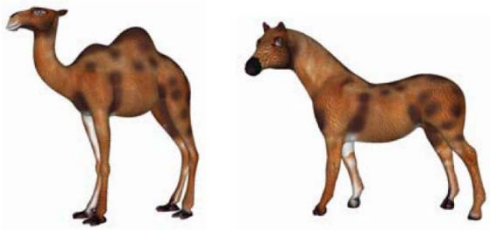


[Schreiner et al. 2004]

Subdivision-based

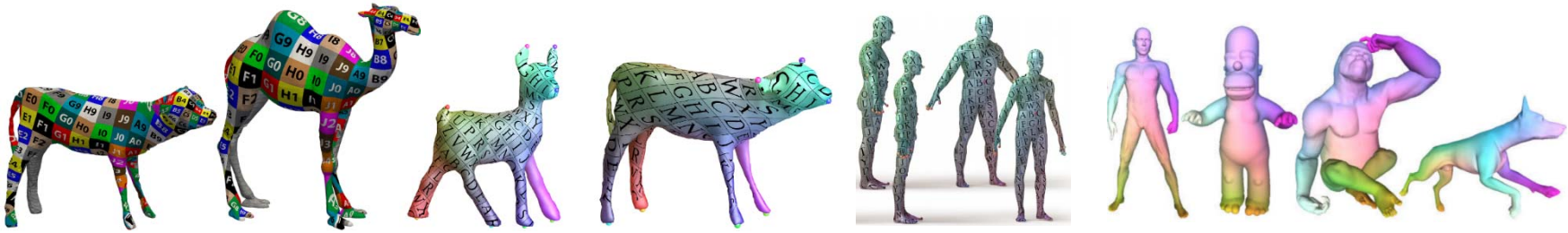


[Kwok et al. 2012]



[Kraevoy et al. 2004]

Common domain-based

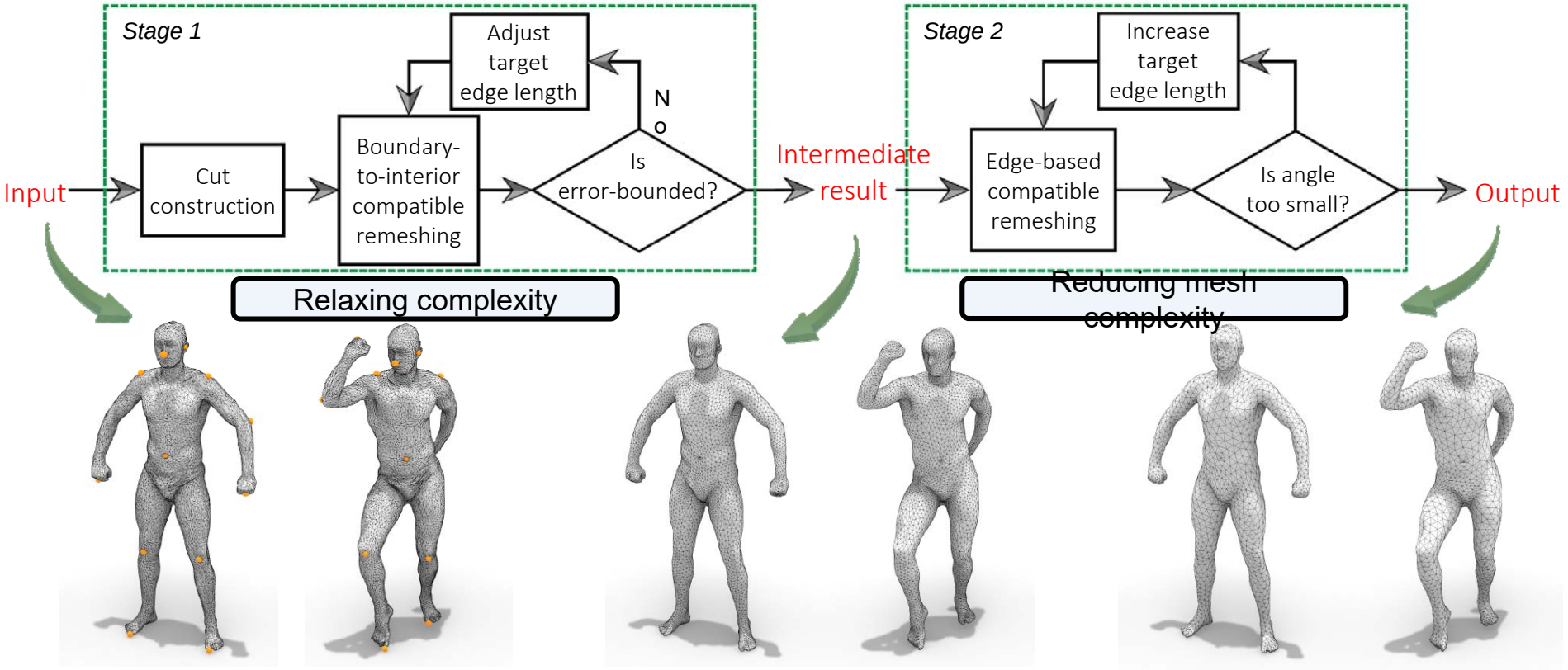


[Aigerman et al. 2014] [Aigerman et al. 2015] [Aigerman et al. 2016] [Mandad et al. 2016]

1.7 Error-Bounded Compatible Remeshing

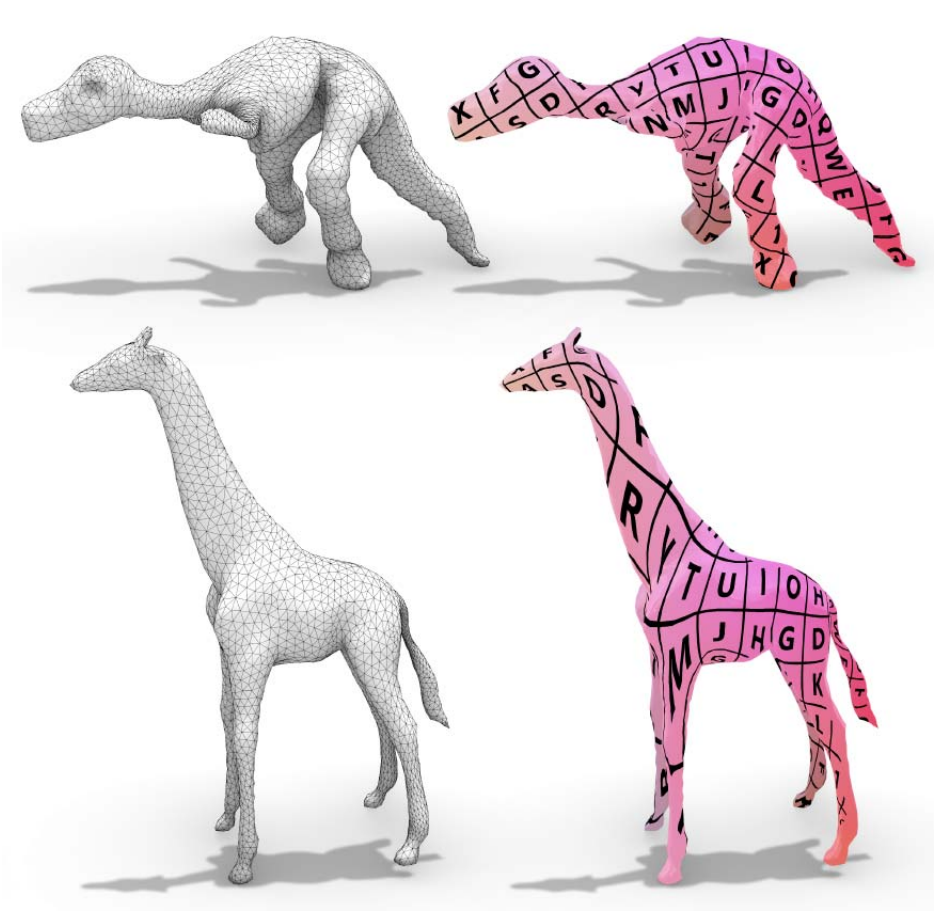
Yang et al. Error-Bounded Compatible Remeshing. Siggraph 2020.

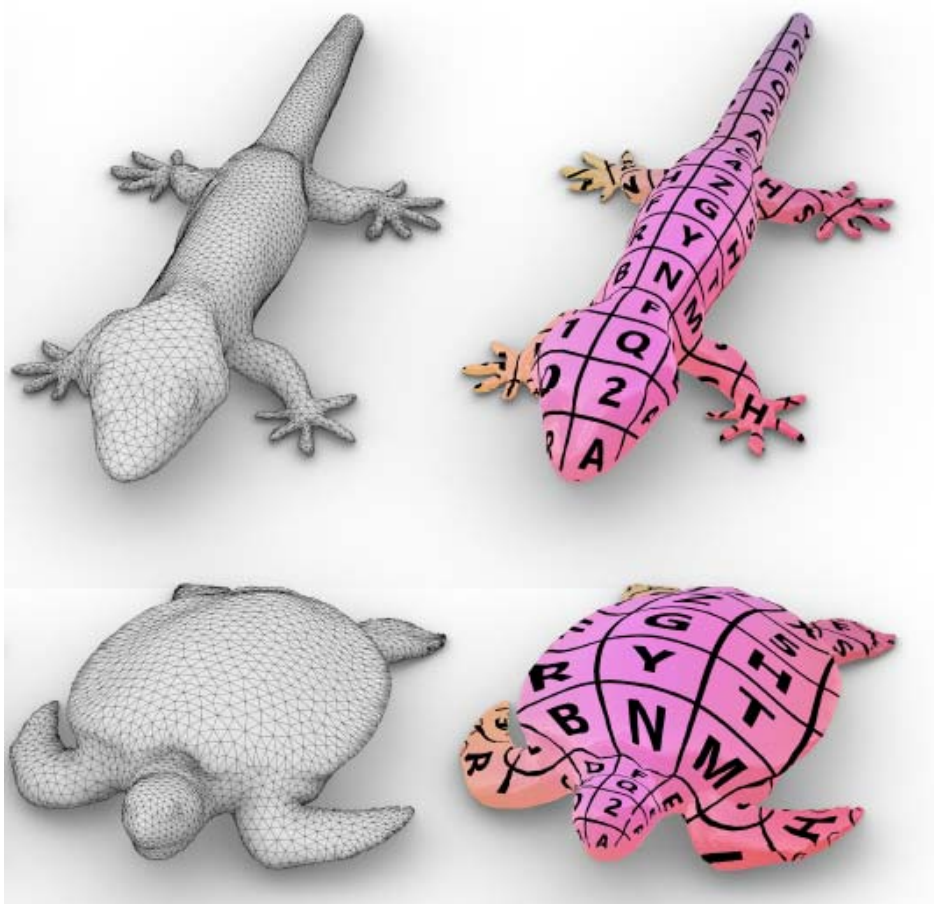
- Optimization based method



1.7 Error-Bounded Compatible Remeshing

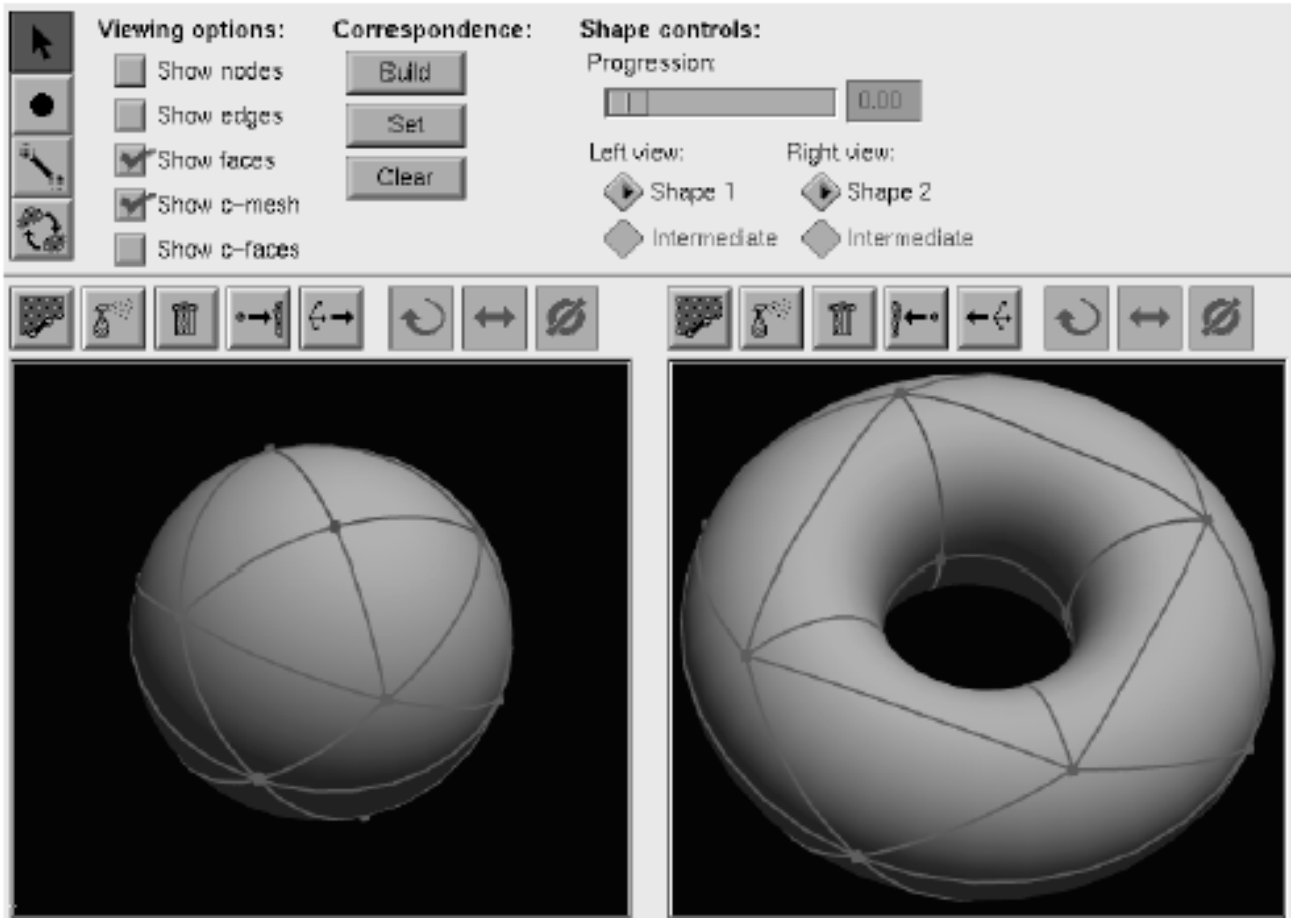
Yang et al. Error-Bounded Compatible Remeshing. Siggraph 2020.





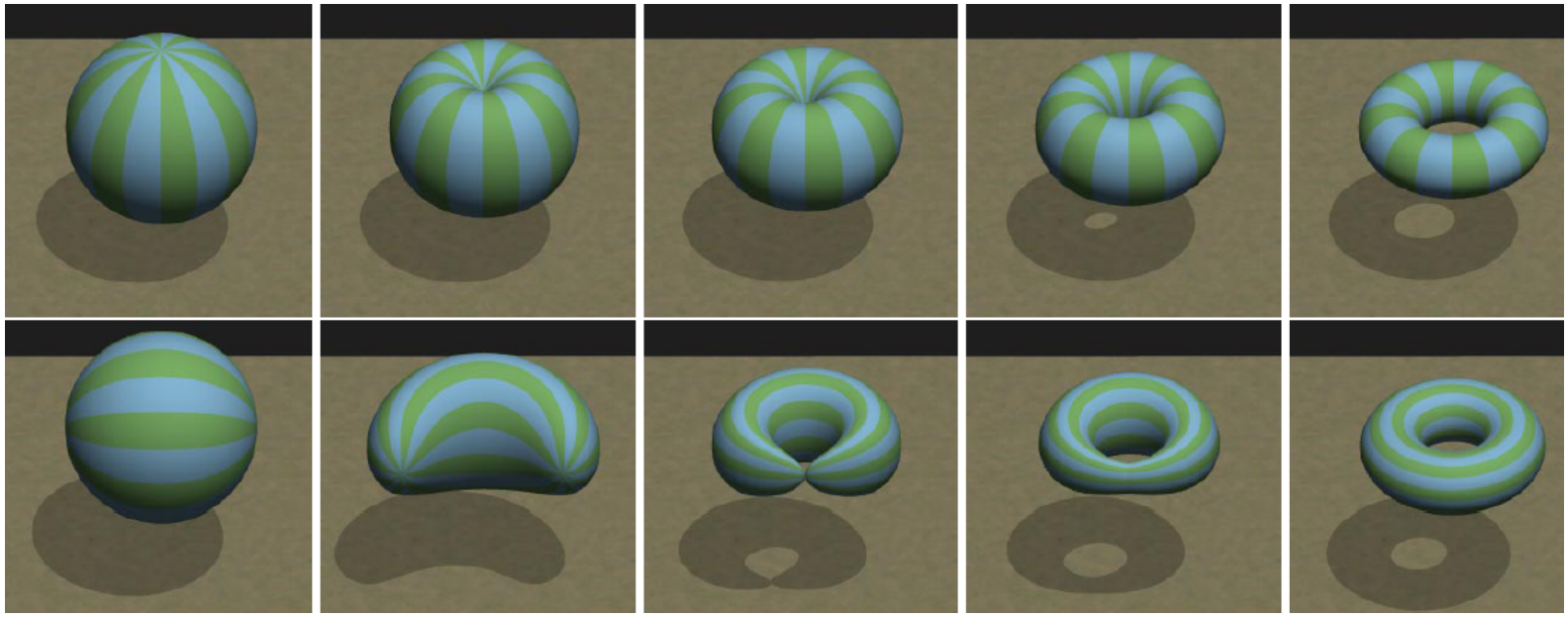
1.8 Different Topologies

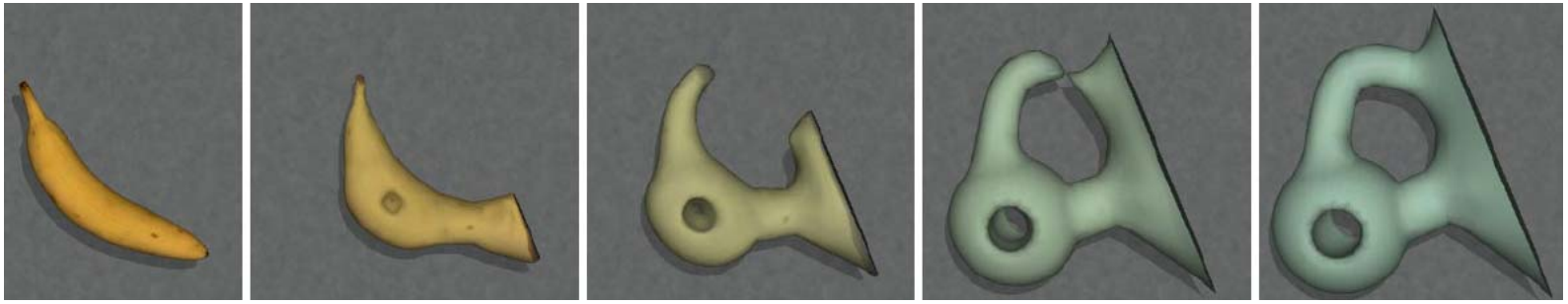
[DeCarlo et al. 1996]



1.8 Different Topologies

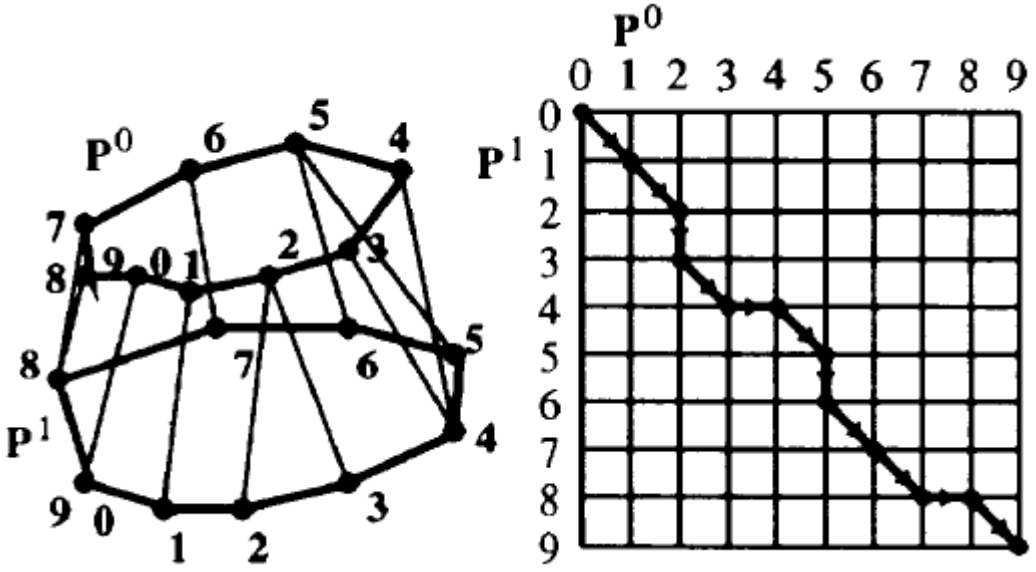
[DeCarlo et al. 1996]

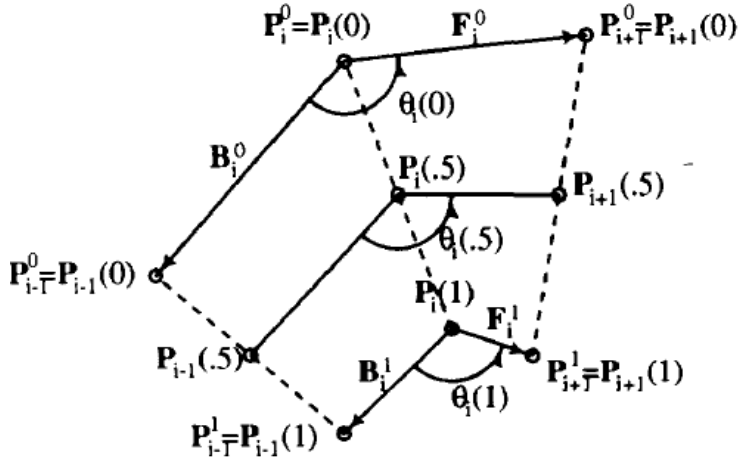




More: Correspondences between planar shapes – Matching

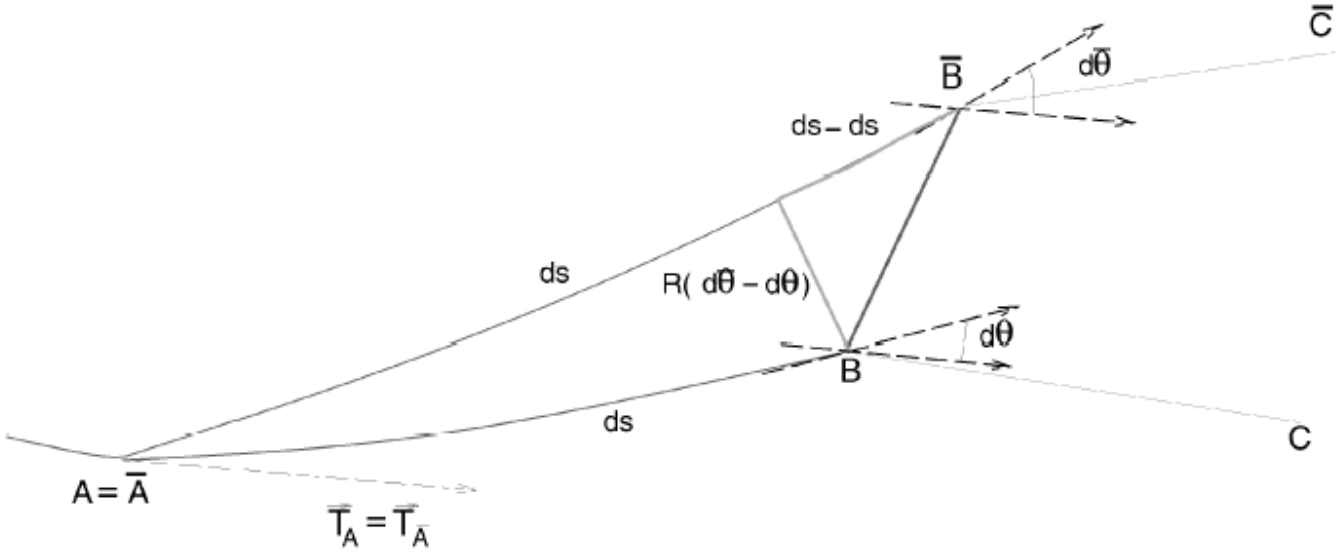
- Physically Based Method [Sederberg et al. 1992]

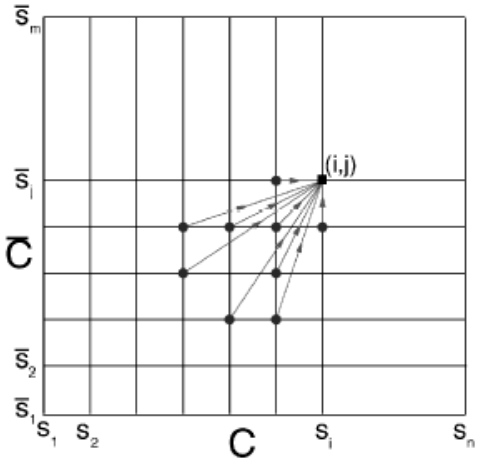




More: Correspondences between planar shapes – Matching

- Curve Aligning [Sebastian et al. 2003]

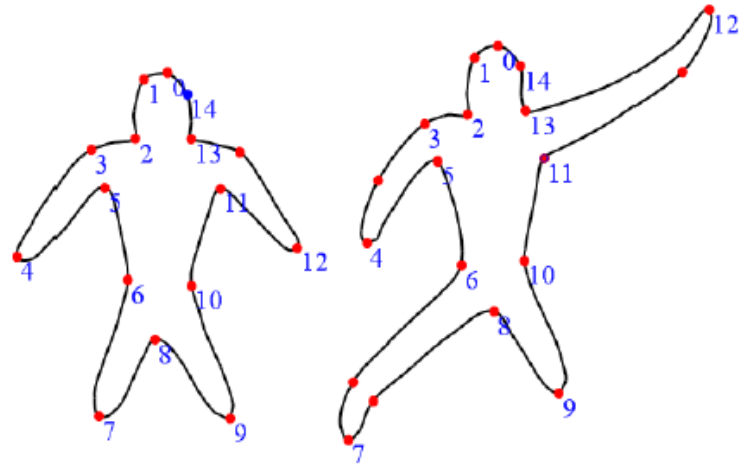


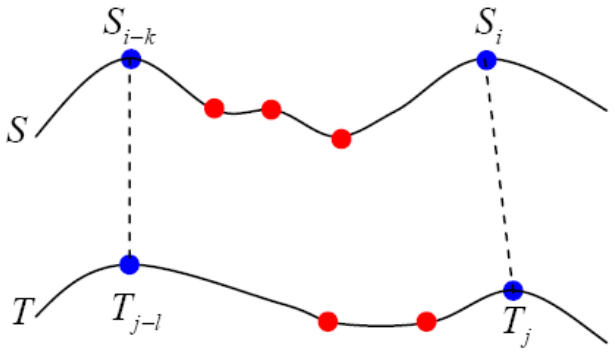


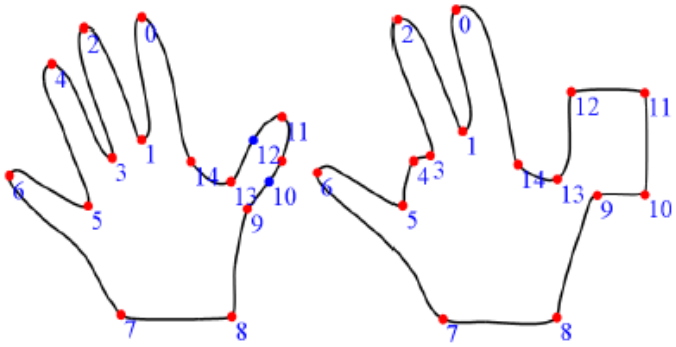
$$\mu[g] = \int_C \left| \frac{\partial}{\partial s} \left(\overline{C}(\overline{s}) - C(s) \right) \right|^2 ds + R \int_C \left(\kappa(s) - \overline{\kappa}(\overline{s}) \right)^2 ds,$$

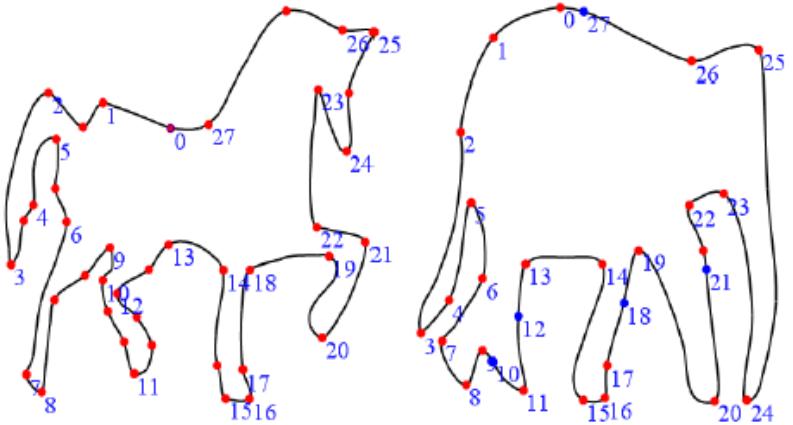
More: Correspondences between planar shapes – Matching

- Perceptually Based Method [Liu et al. 2004]



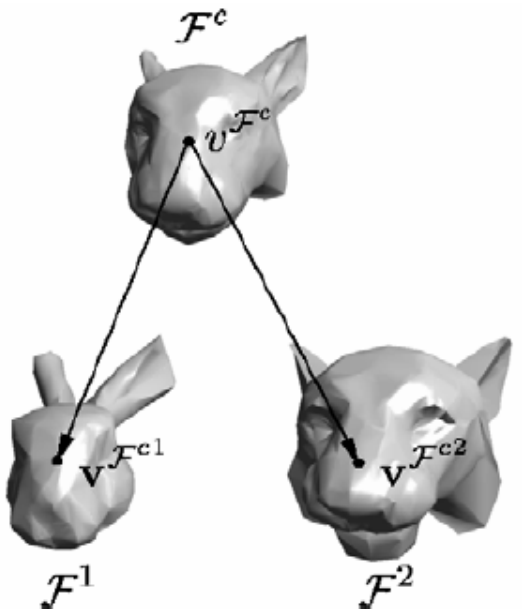






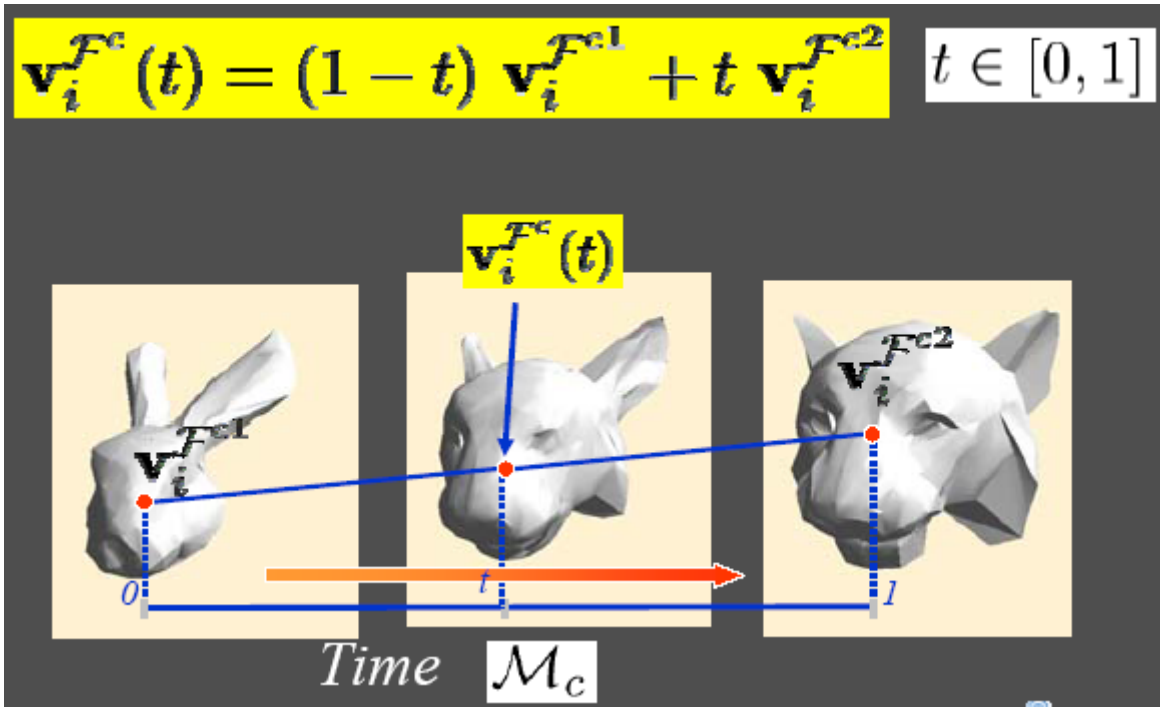
2. Vertex Path (Trajectory)

- Input:
  - All vertices on source & target have one-to-one correspondence with each other
  - Each vertex has two 3D coords v^{Fc1} (source) and v^{Fc2} (target)
- Output: generate the intermediate shapes from two shapes



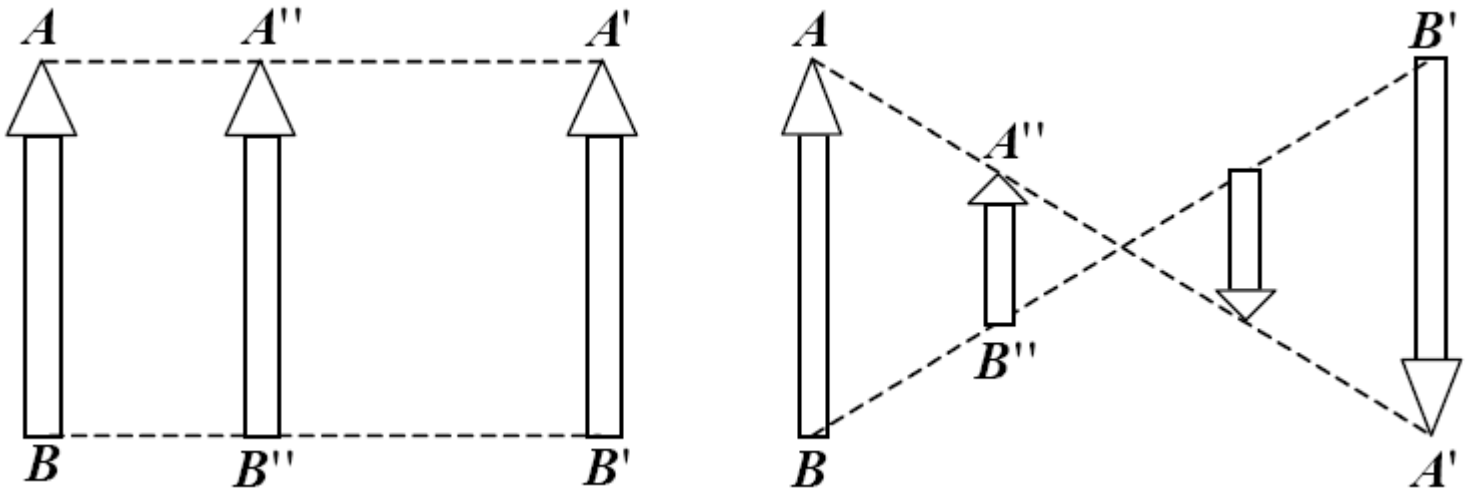
Simplest Method: Linear Interpolation

- Linear interpolation between corresponding points
- Work well for many cases
  - many drawbacks

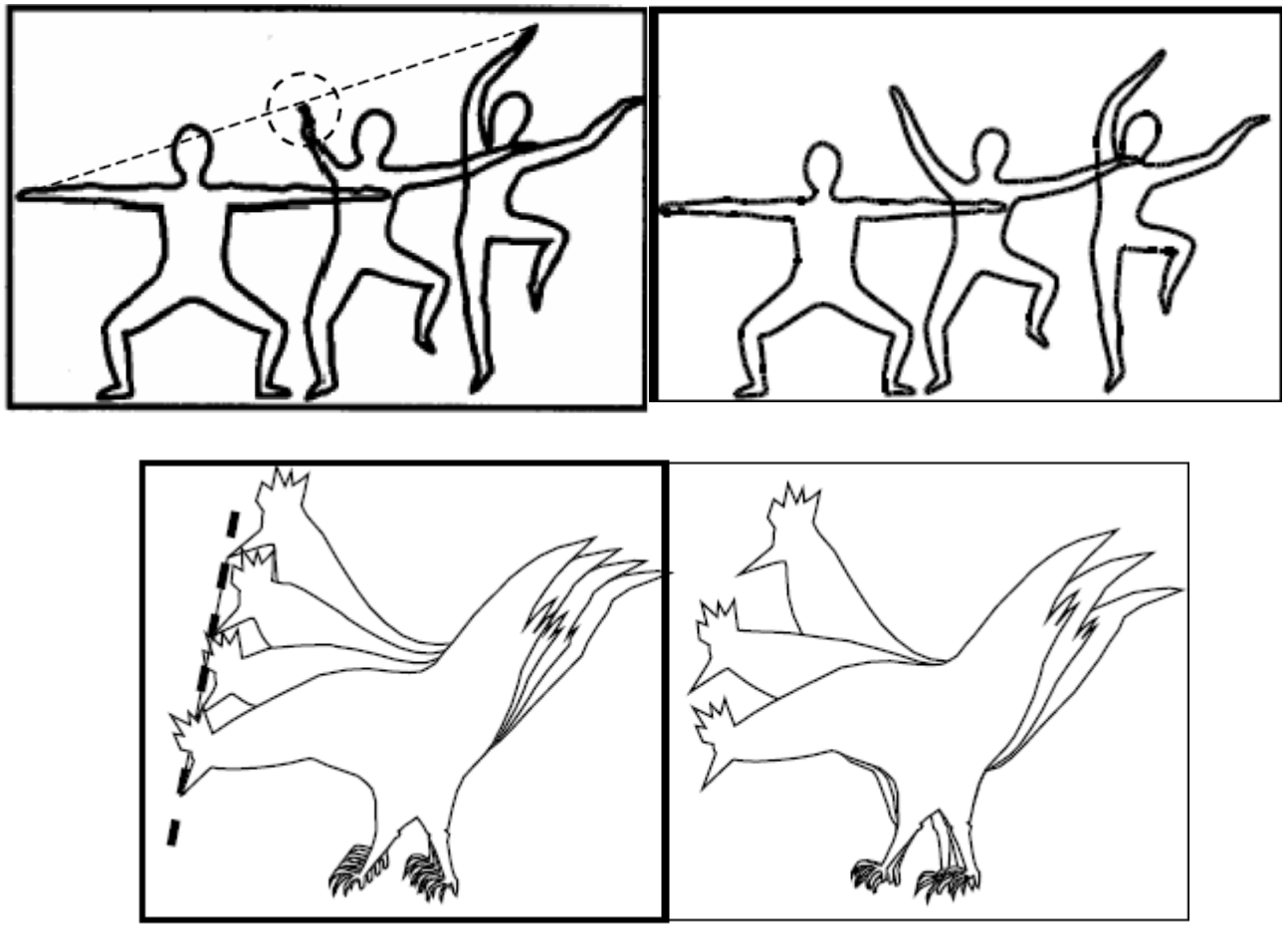


Linear Interpolation

- Simple and easy
- Drawbacks
  - Shrinkage



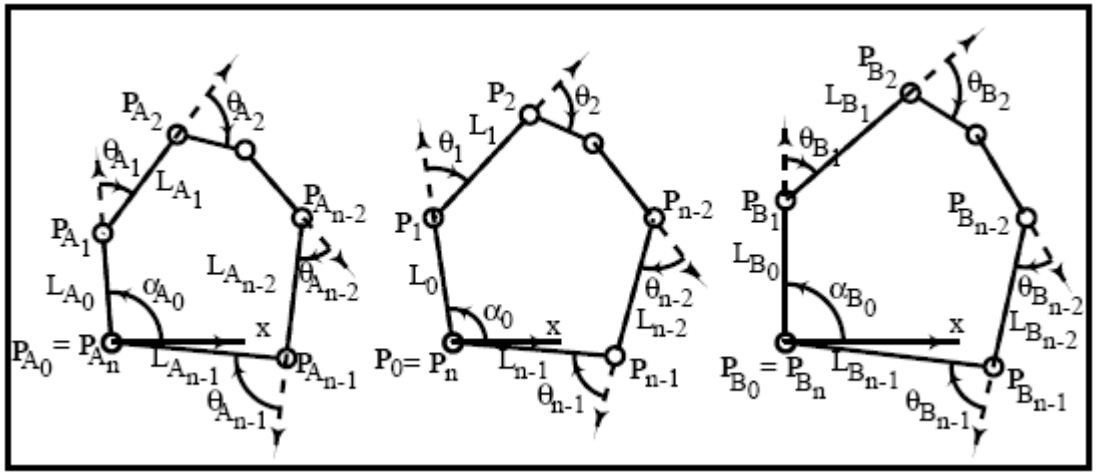
Unnatural Results in Linear Interpolation



Intrinsic Approach

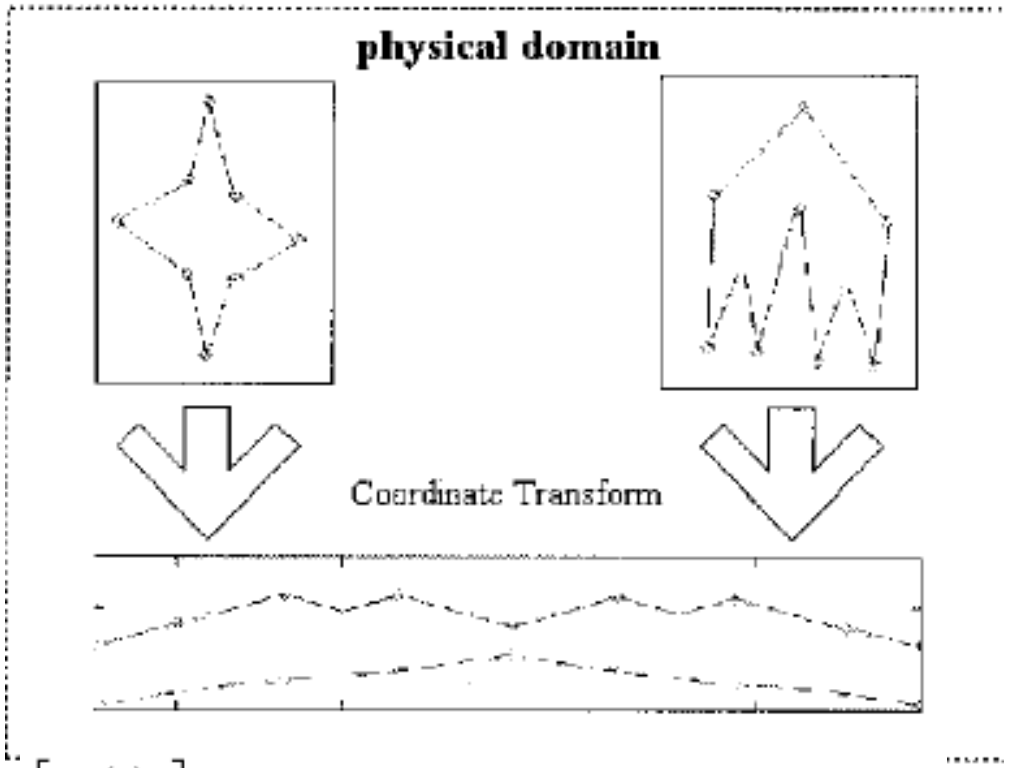
Sederberg et al. 1993

- Intrinsic variables
  - edge lengths
  - turning angles



Fourier Approach

Chen et al. 2001



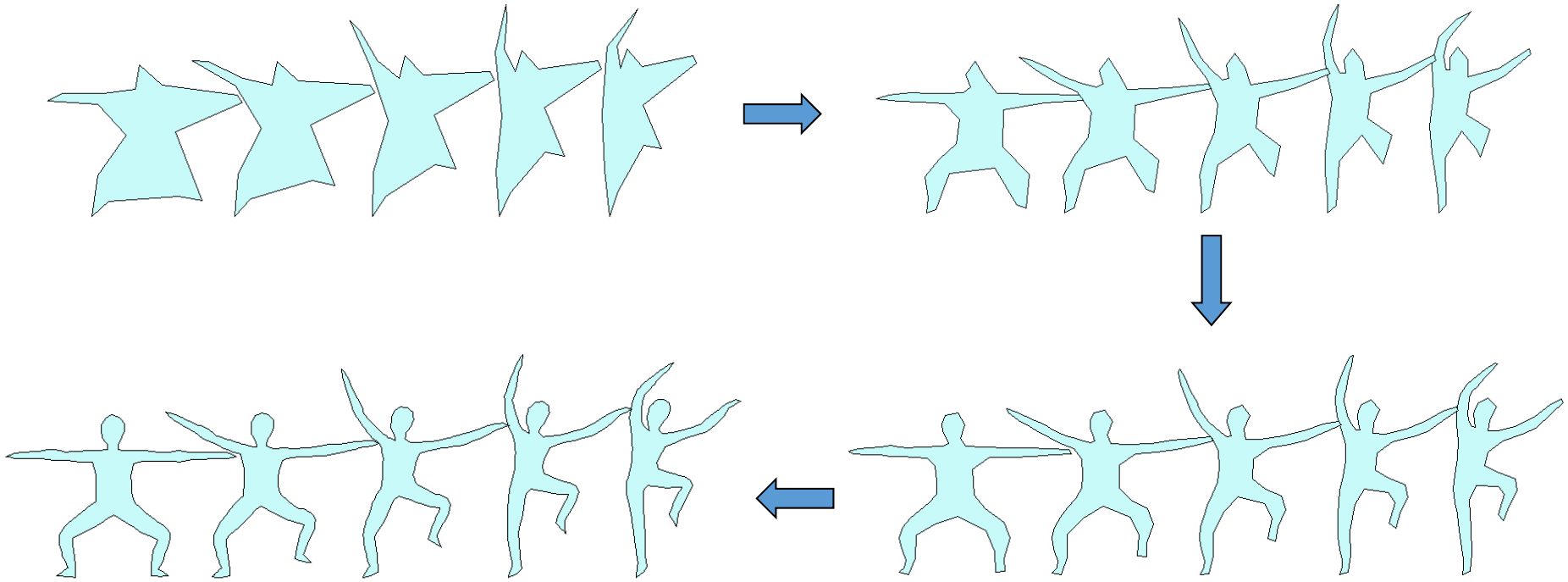
$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$$

$$= \begin{bmatrix} a_0 \\ c_0 \end{bmatrix} + \sum_{k=1}^{\infty} \begin{bmatrix} a_k & b_k \\ c_k & d_k \end{bmatrix} \begin{bmatrix} \cos(2\pi kt) \\ \sin(2\pi kt) \end{bmatrix}$$

Wavelet Approach

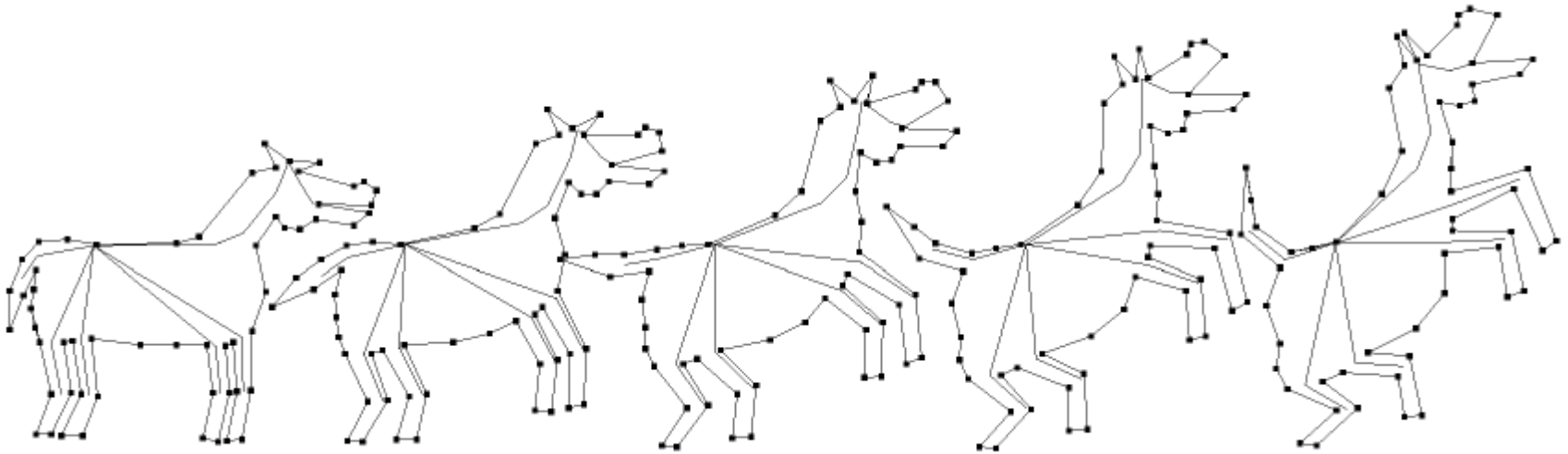
- Wavelet decomposition

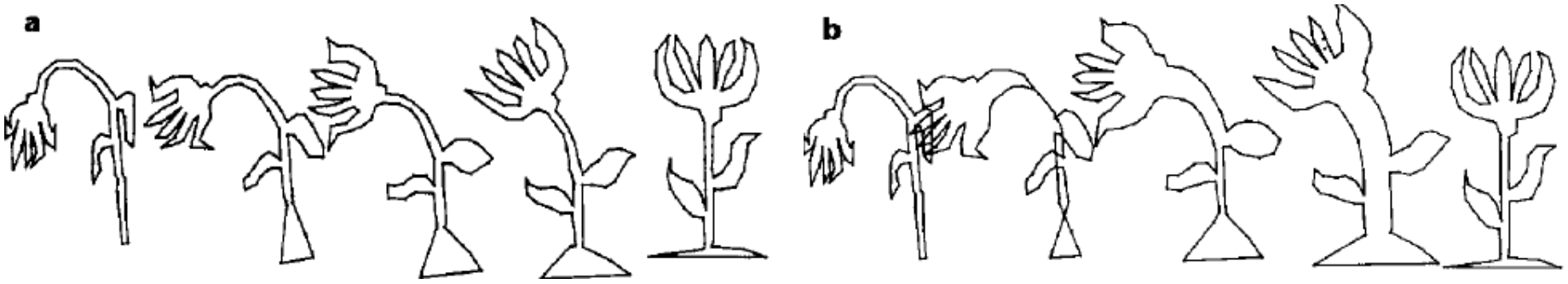
Zhang et al. 2000



Star Skeleton Representation

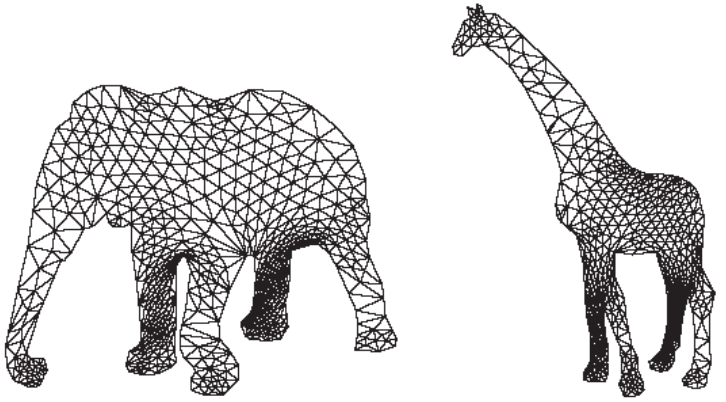
[Shapira et al. 1995]





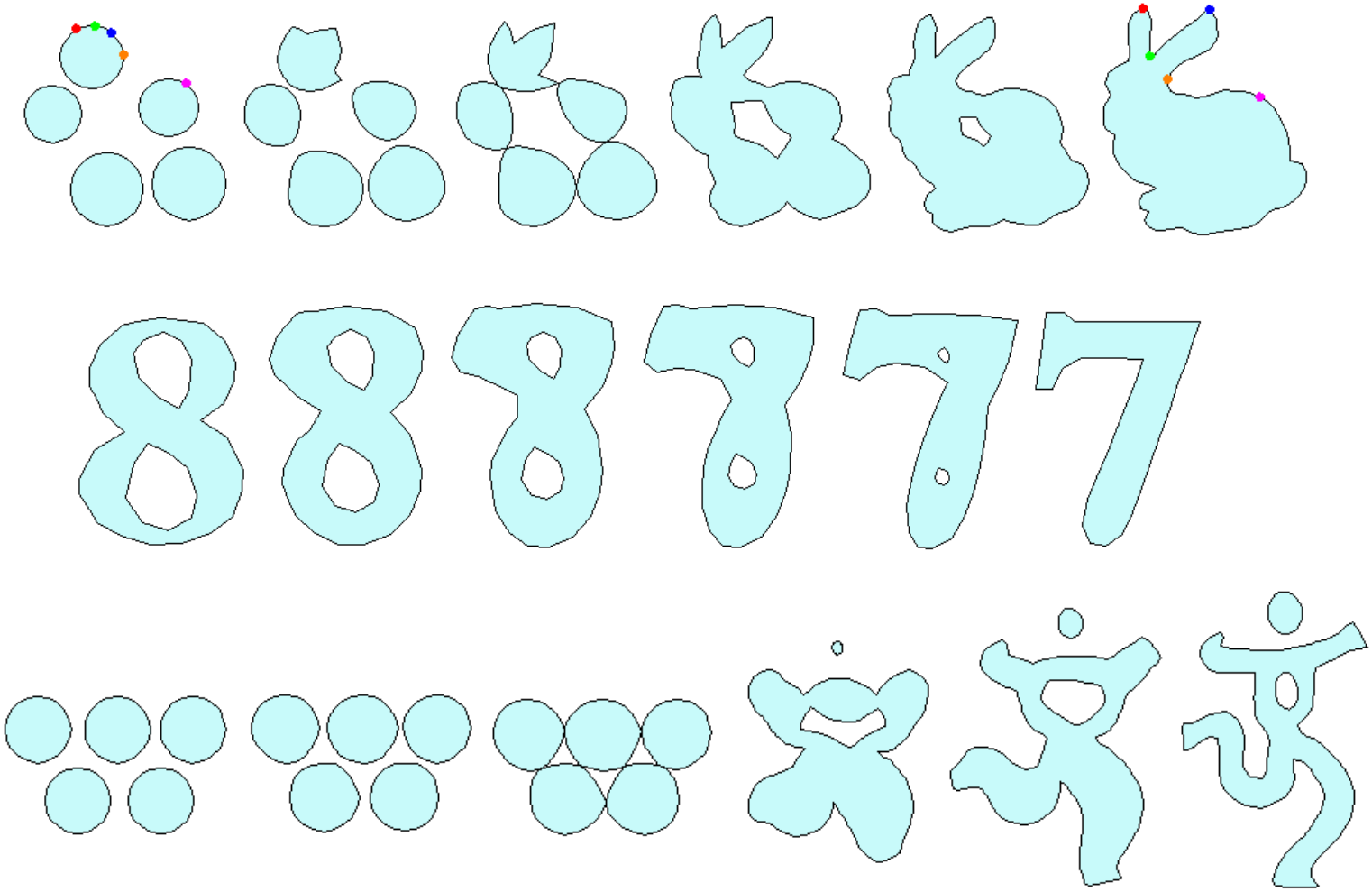
Interior Based Approach

- Based on compatible triangulation
  - [Gotsman and Surazhsky, 1999-2001]
  - As-rigid-as-possible [Alexa et al. 2000]



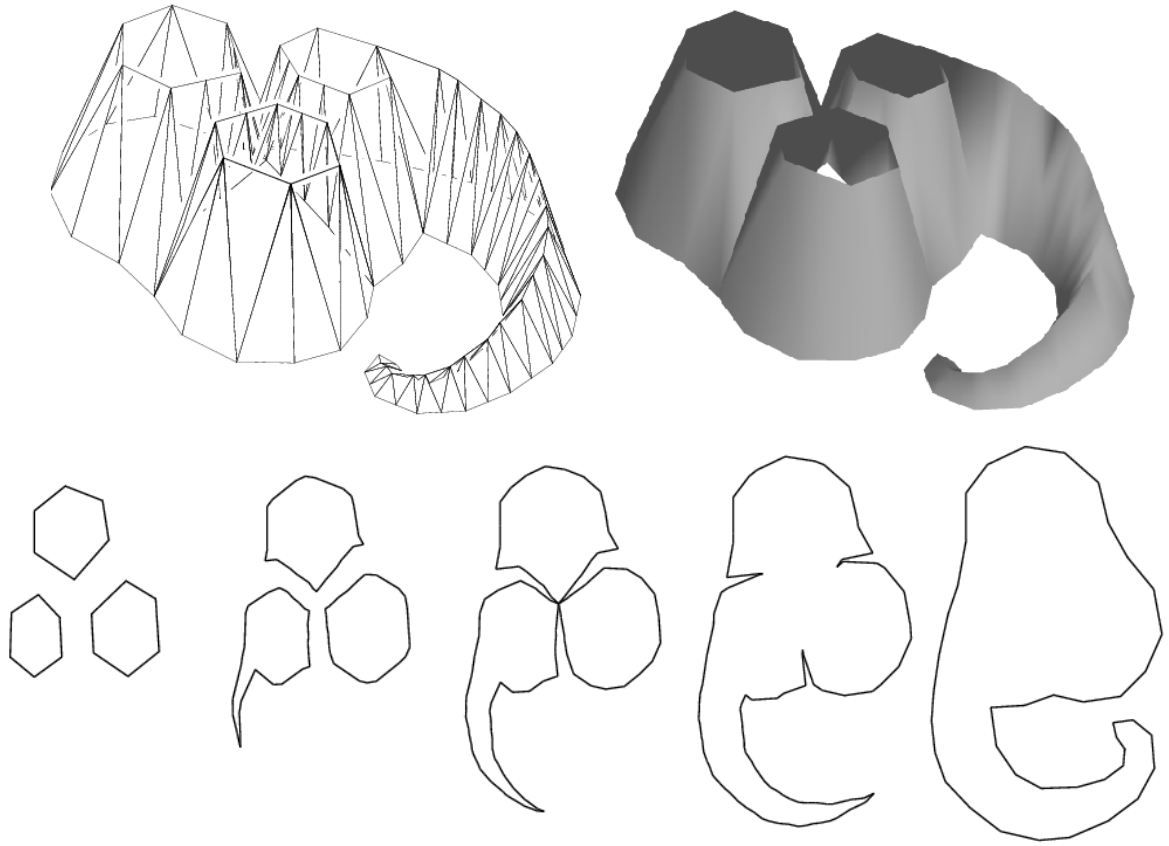
Morphing between Different Topologies

Liu et al. 2005



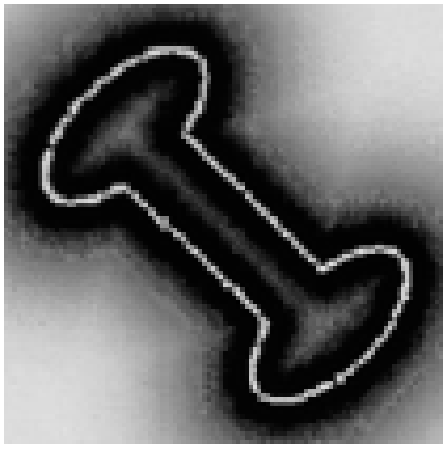
Implicit Approaches

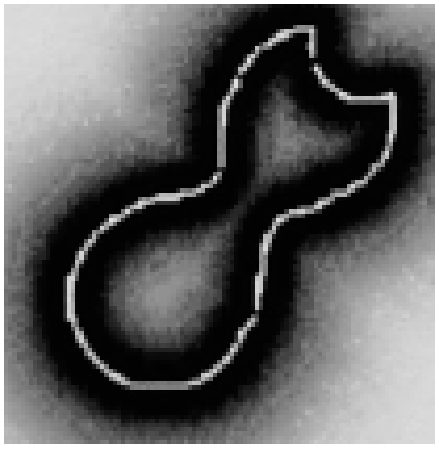
- Construct a 4D function which interpolates two shapes (with iso-value 0 and 1 respectively)



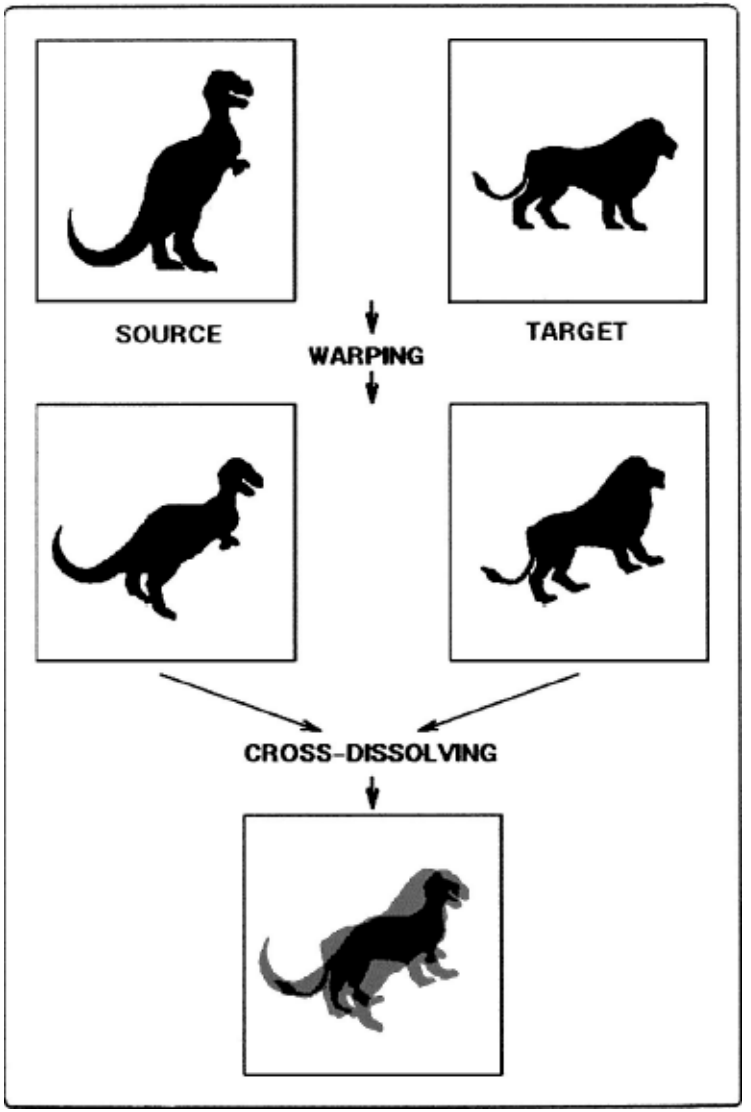
Distance Field

- Distance field of a shape

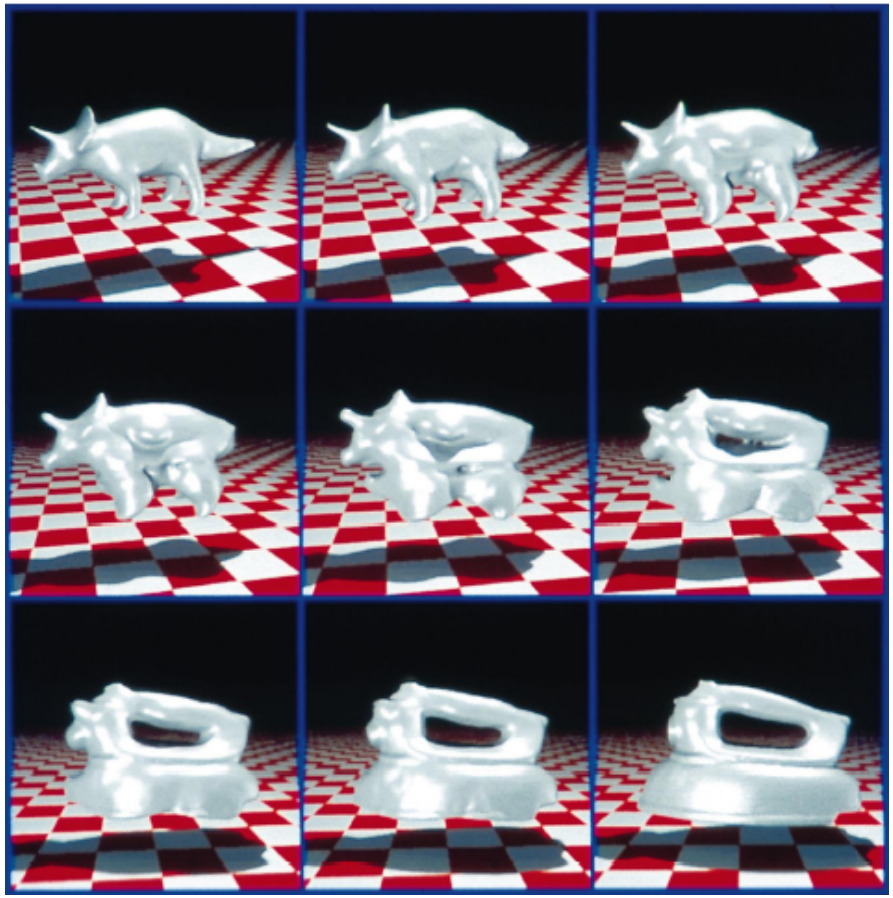


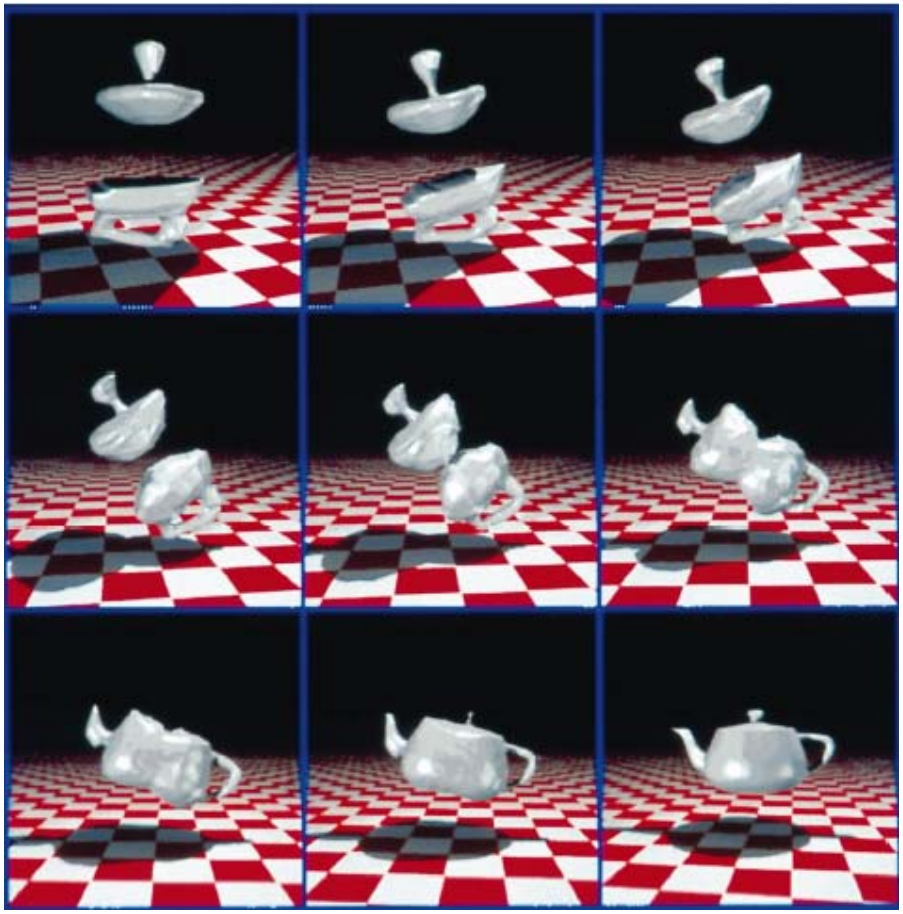


[Cohen-Or et al. 1998]



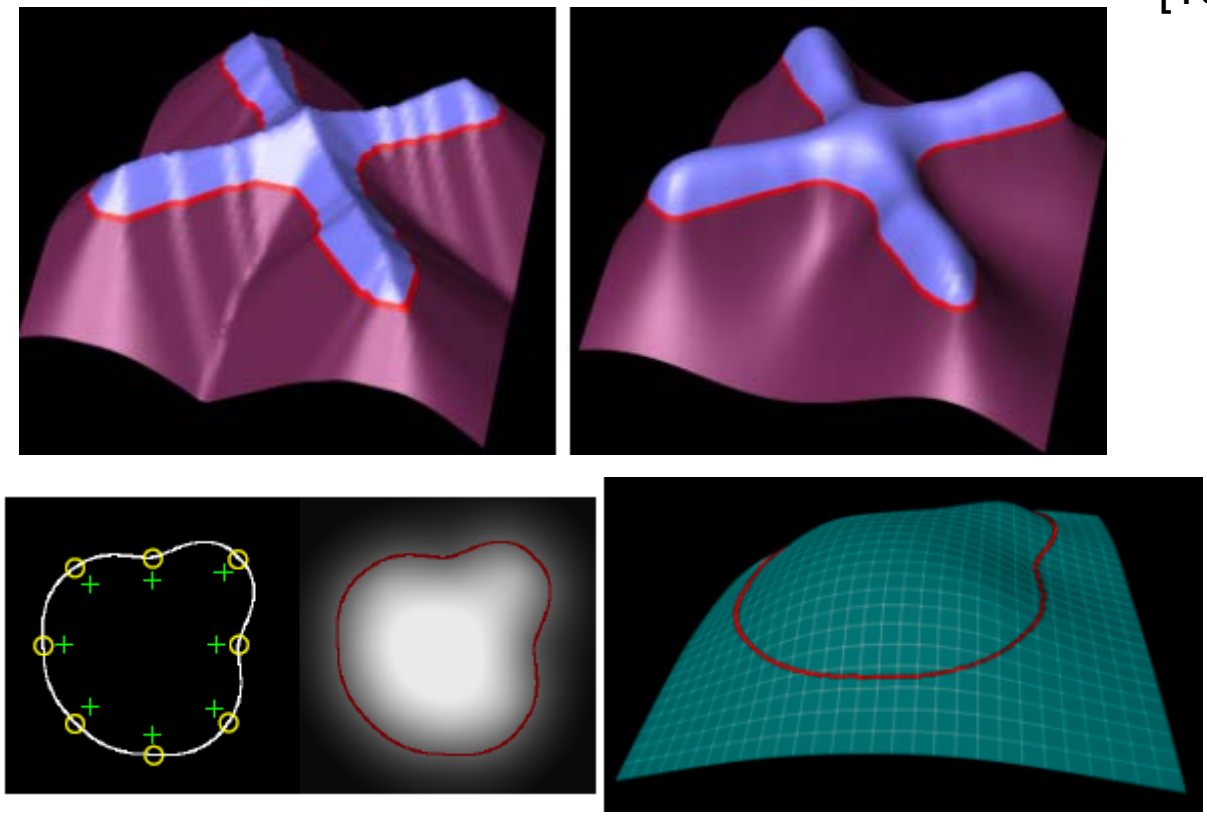
Distance Field





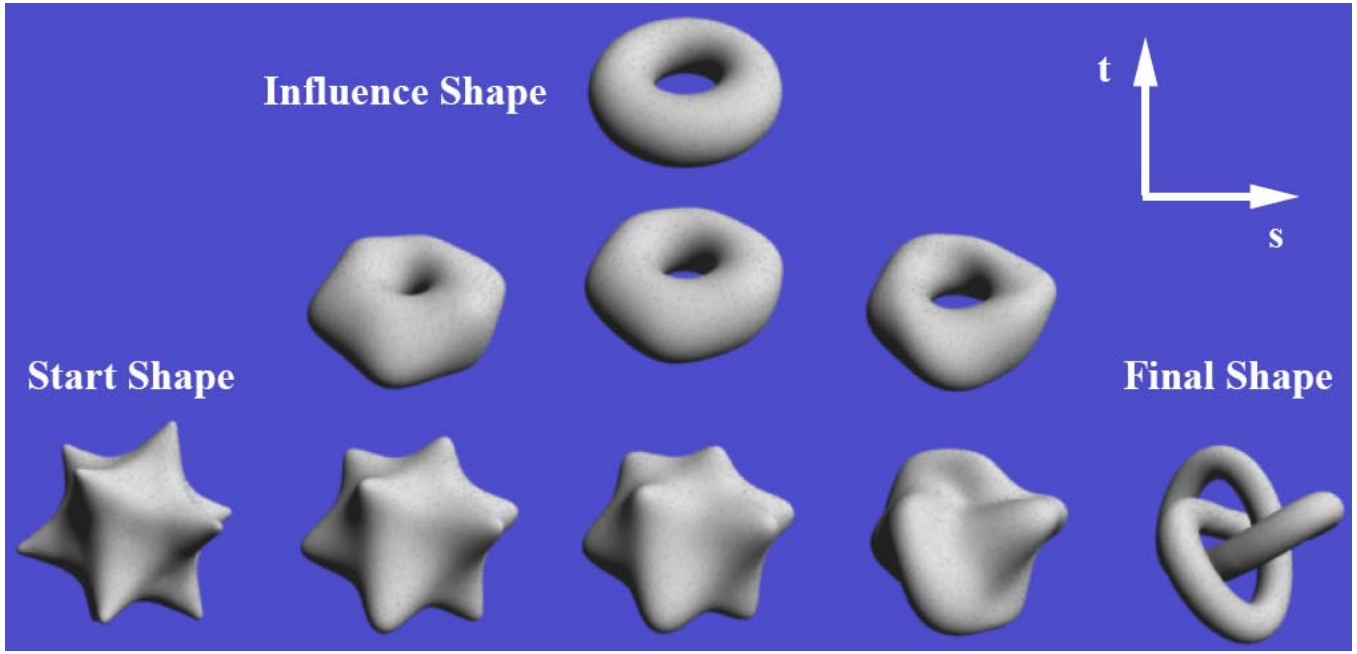
Variational Implicit Function

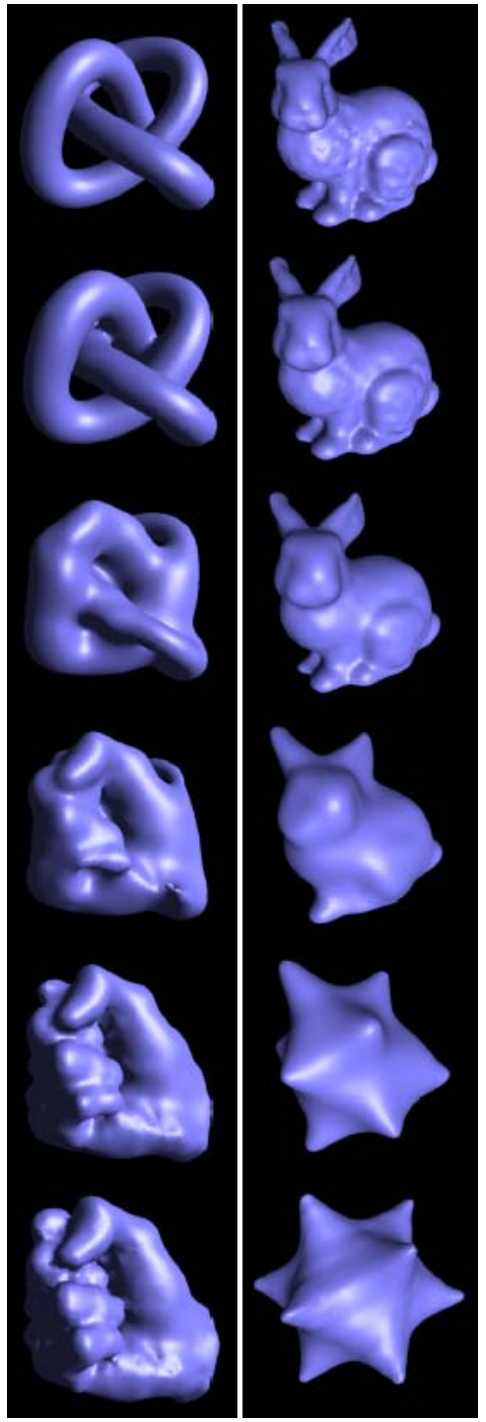
[Turk et al. 1999]



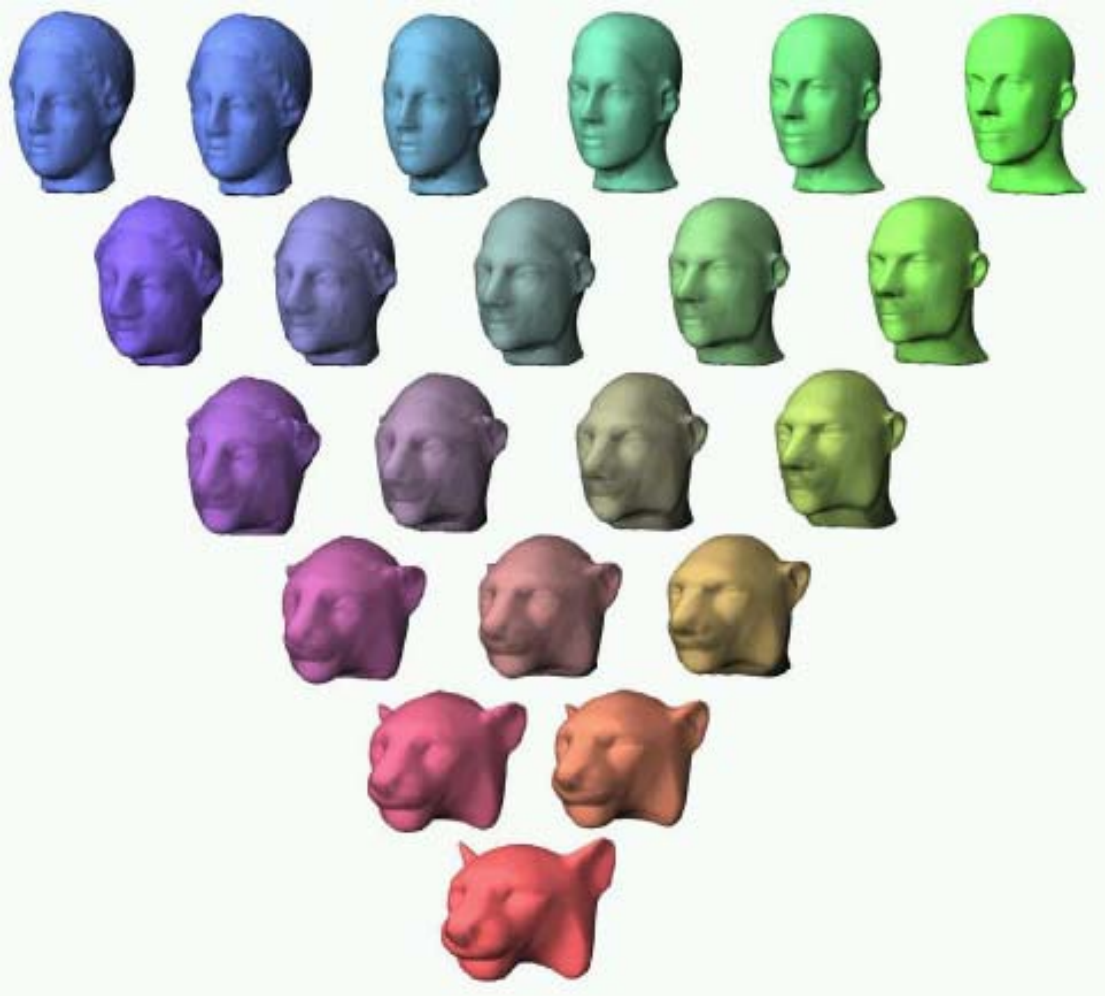
$$f(\mathbf{x}) = \sum_{j=1}^{n} d_j \phi(\mathbf{x} - \mathbf{c}_j) + P(\mathbf{x})$$

Examples



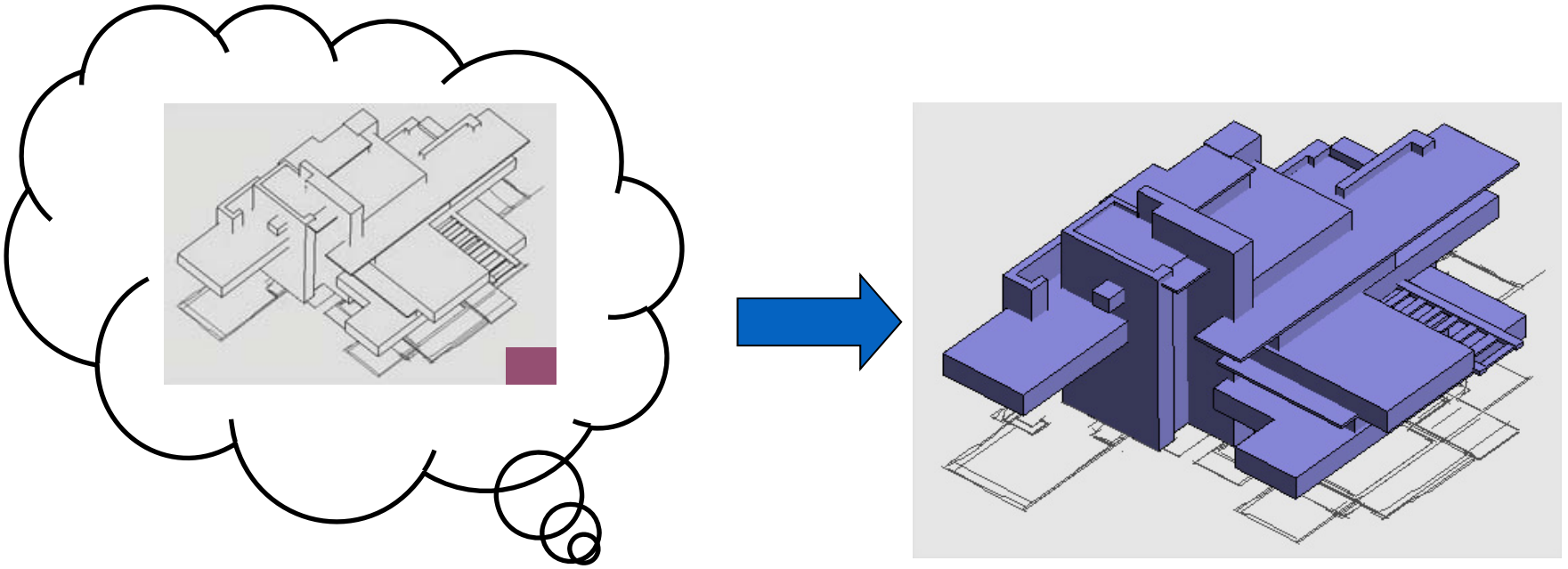


Polymorph:
Morphing between multiple shapes



Summary

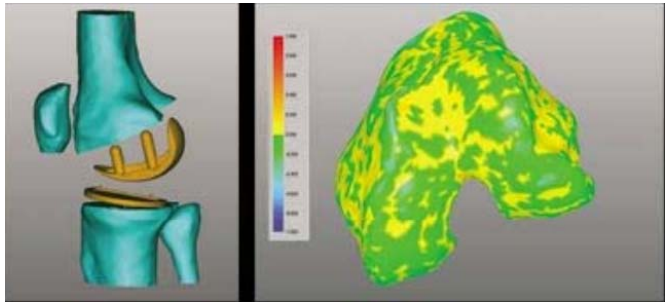
3D content creation is still hard

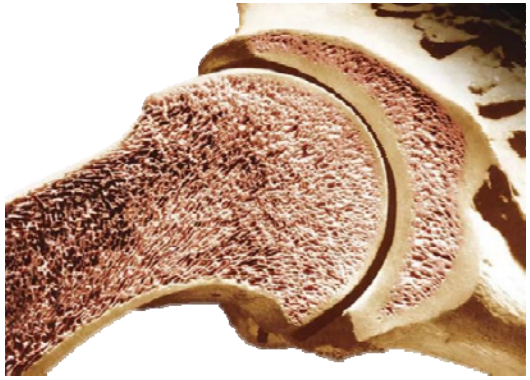


- One of the most central and challenging problems in graphics
  - (Comparing to simulation and rendering)

Challenges to Geometric Modeling

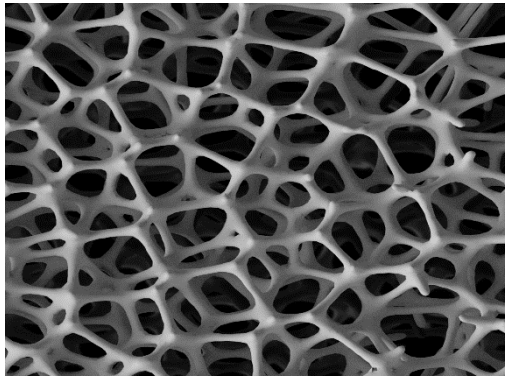
- Modeling porous structure
- Modeling non-homogeneous materials
  - varying density
  - changing composition
  - multiple phases (solid, liquid)
  - …





生物多孔组织

多孔泡沫



微胞腔结构

中国科学技术大学

University of Science and Technology of China

谢　谢　！