

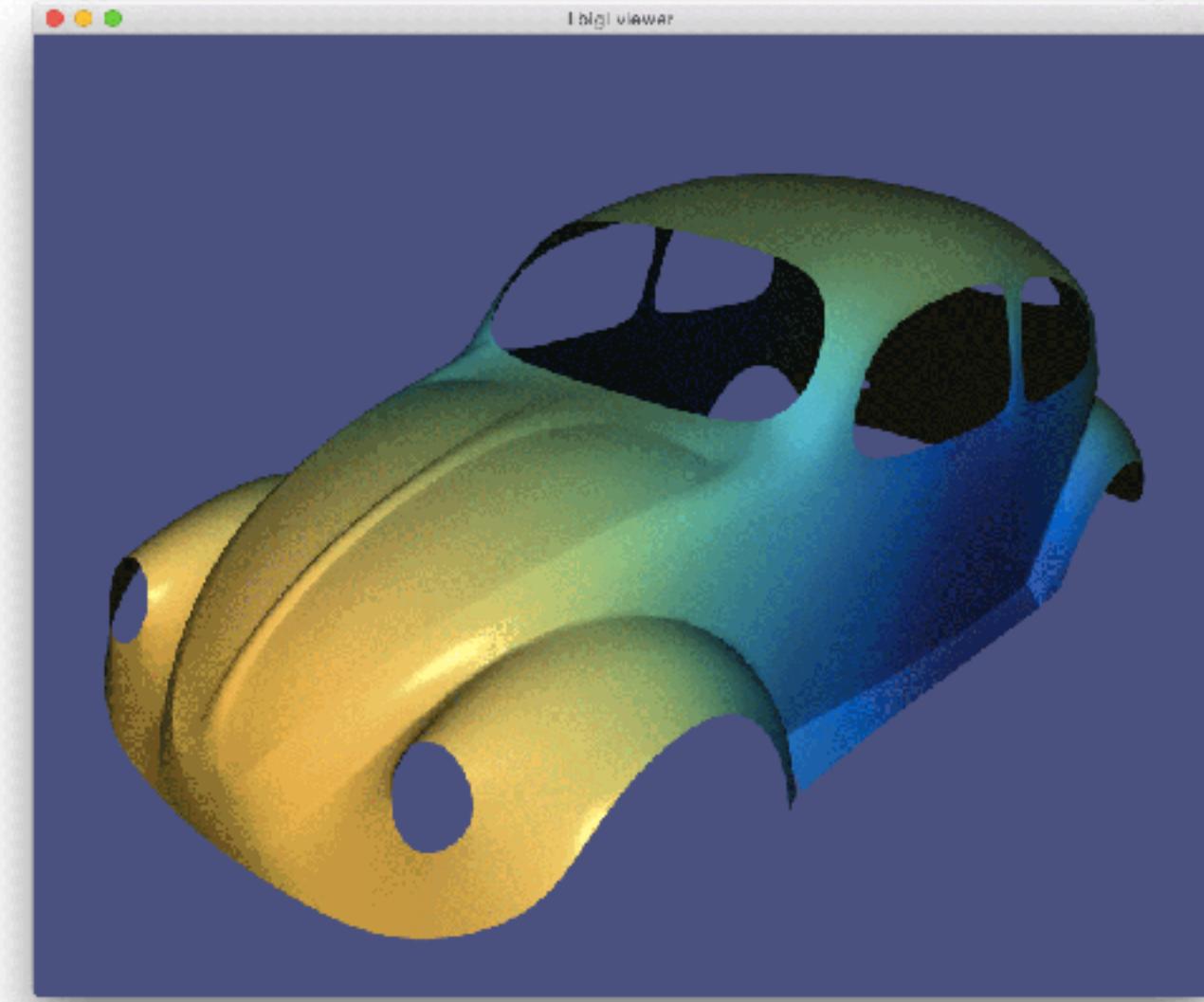
# Introduction To Geometric Modelling

# Course Goals

- Learn how to design, program and analyze algorithms for **interactive 3D shape modeling** and **digital geometry processing**
- Theory and applications of 3D mesh processing
- Hands-on experience with shape modeling and geometry processing algorithms

# Geometric Modeling and Processing

- The shape of an object is an important characteristic (not the only one...)
- Geometry processing:  
computerized modeling of 2D/  
3D geometry

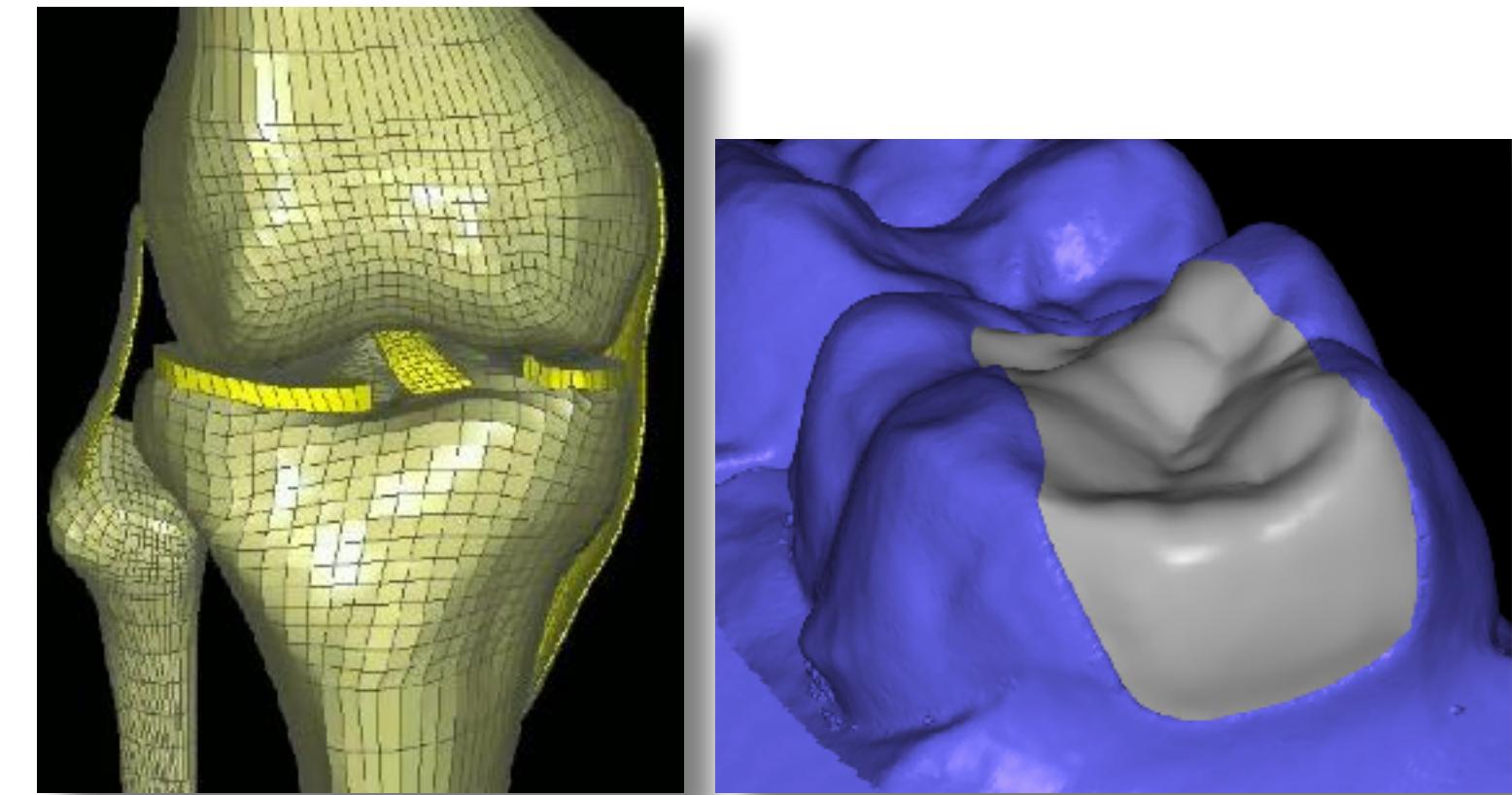


Copyright: Blender

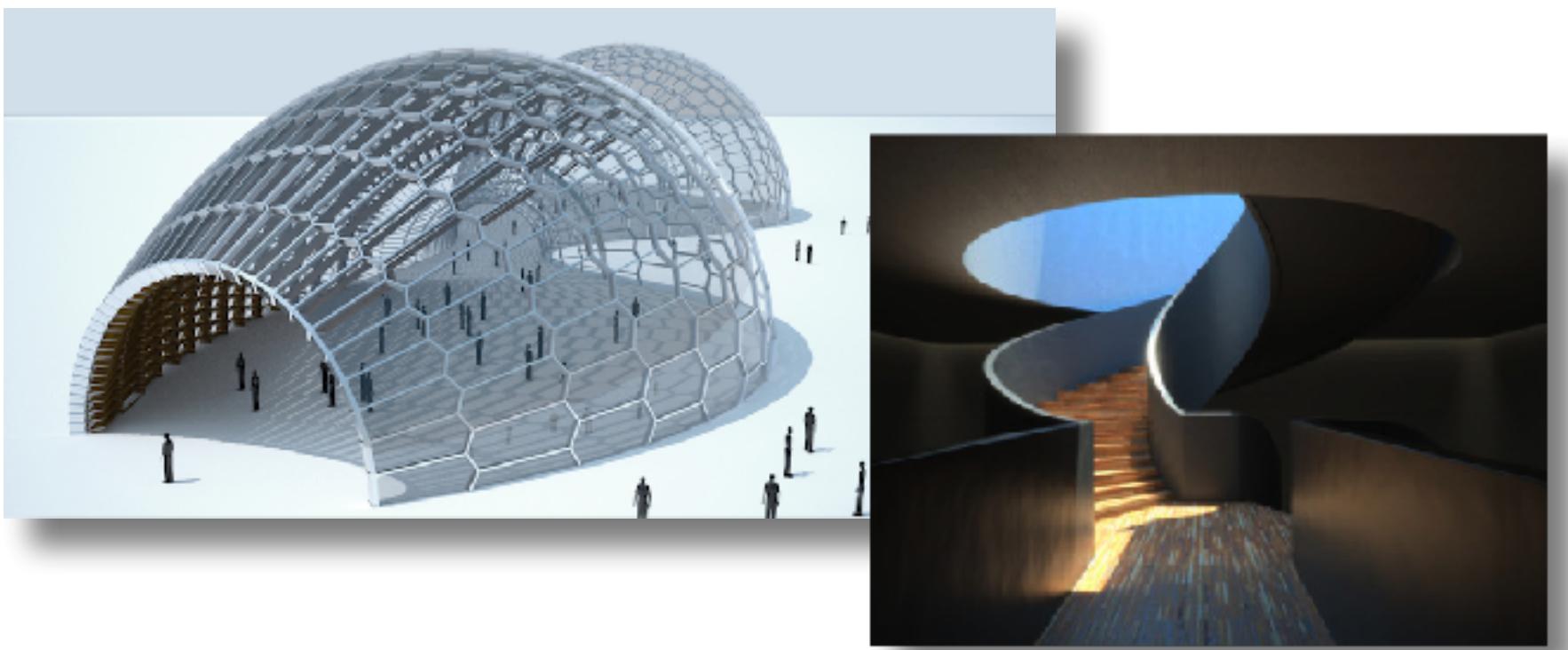
# Applications



Product design and prototyping



Medicine, prosthetics

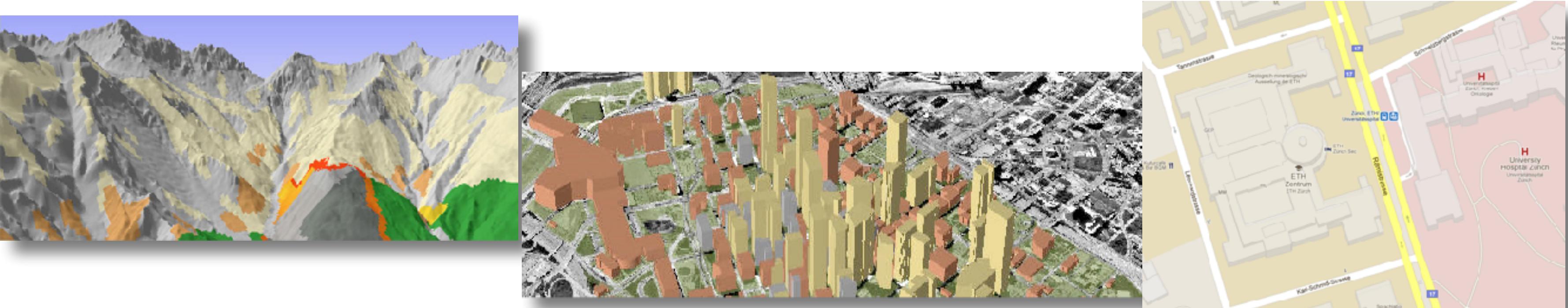


Architecture



Cultural heritage

# Applications



Geographical systems

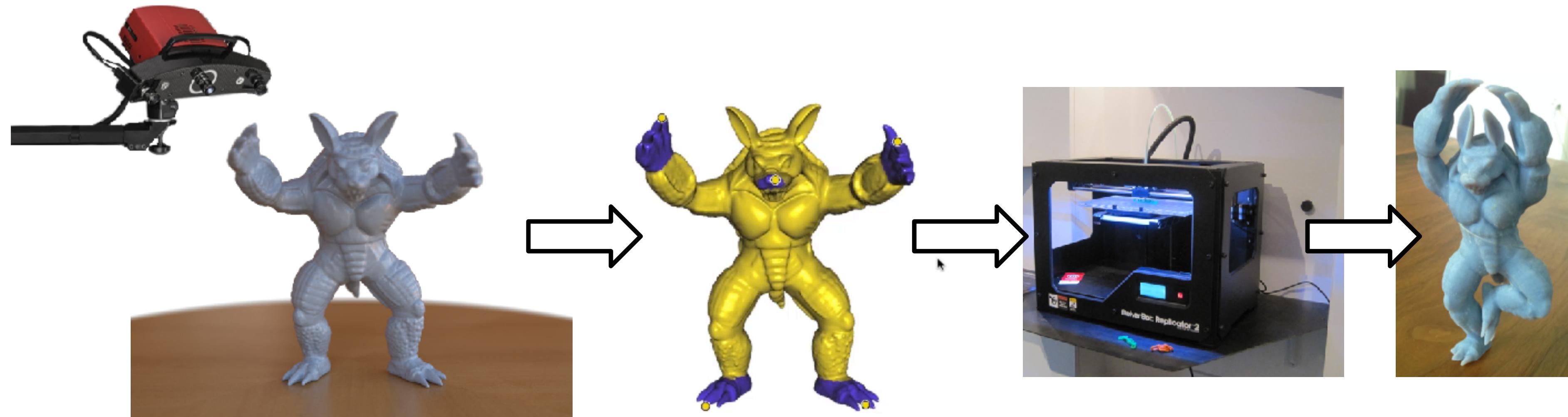


[Bickel et al., ACM SIGGRAPH 2010]

Manufacturing, 3D Printing

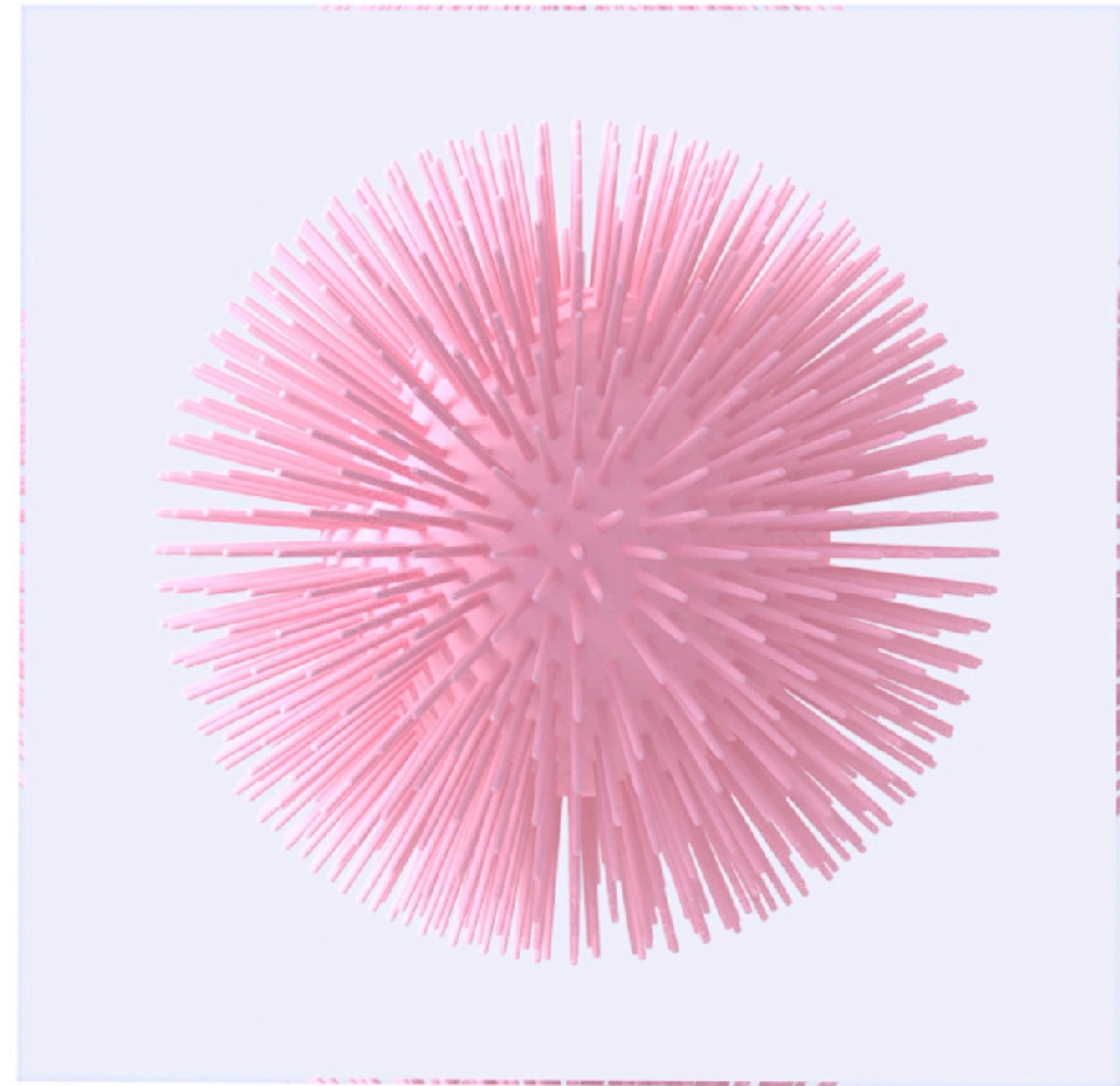
# Fabrication

- Modern scanning and 3D printing technologies allow replication and much more



# Simulation

IPC



**tetrahedra:** 2314K

**contacts per step (max):** 5.6K

**dt:** 0.001

**$\mu$ :** 0

# Digital Geometry Processing (DGP)

- Processing of discrete surfaces/volumes
- Why discrete?
  - Representable by a computer
  - Efficiently rendered by graphics hardware
  - Input to most simulation/analysis tools
  - Easy to acquire(CT, MRI, LIDAR, Kinect...)



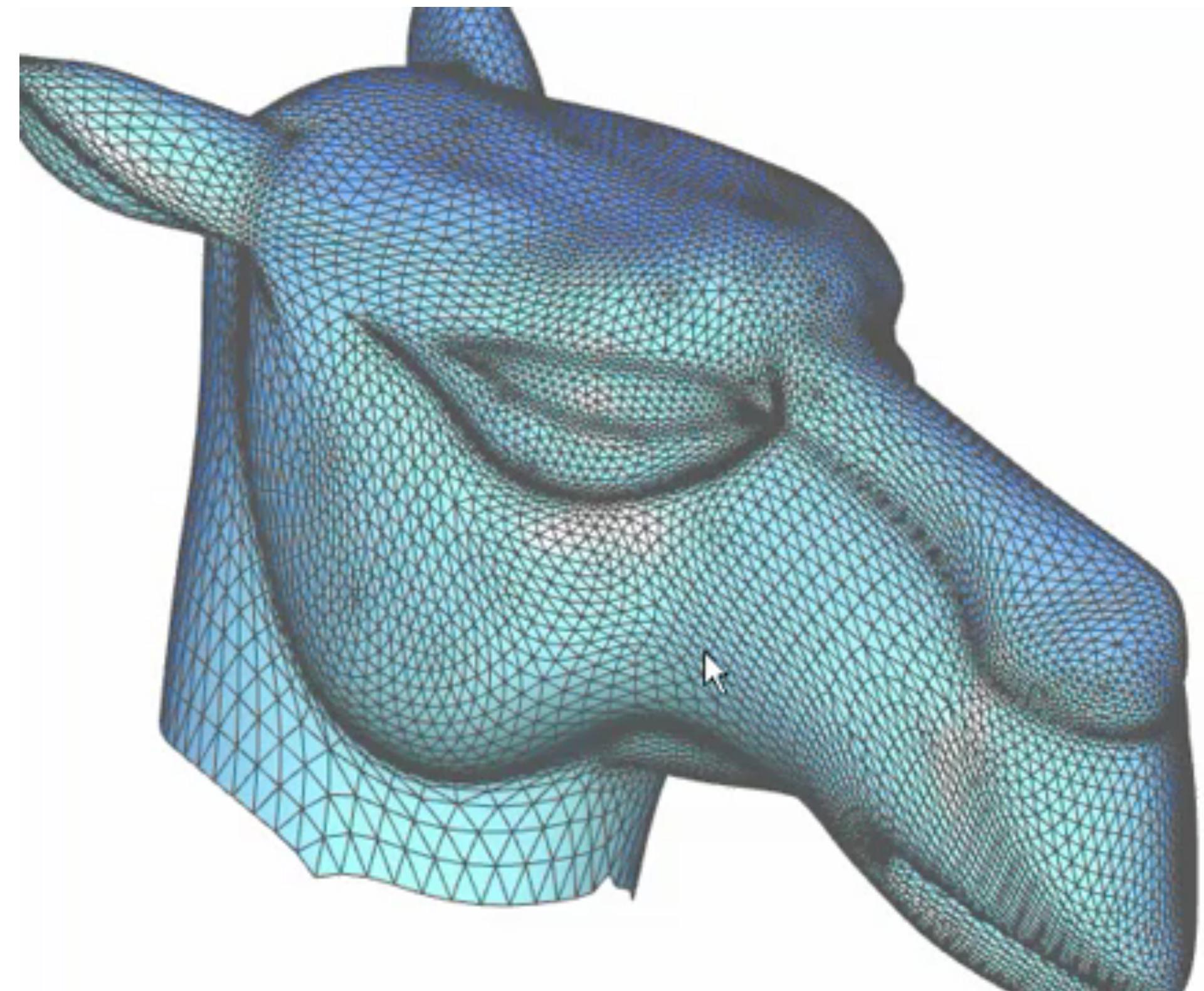
Copyright: The Digital Michelangelo Project



Computer Science

# Interactive Shape Modeling

- Tools for design, editing and animation of digital shapes
  - Interactive means fast algorithms
  - Intuitive – convenient interface and predictable outcome



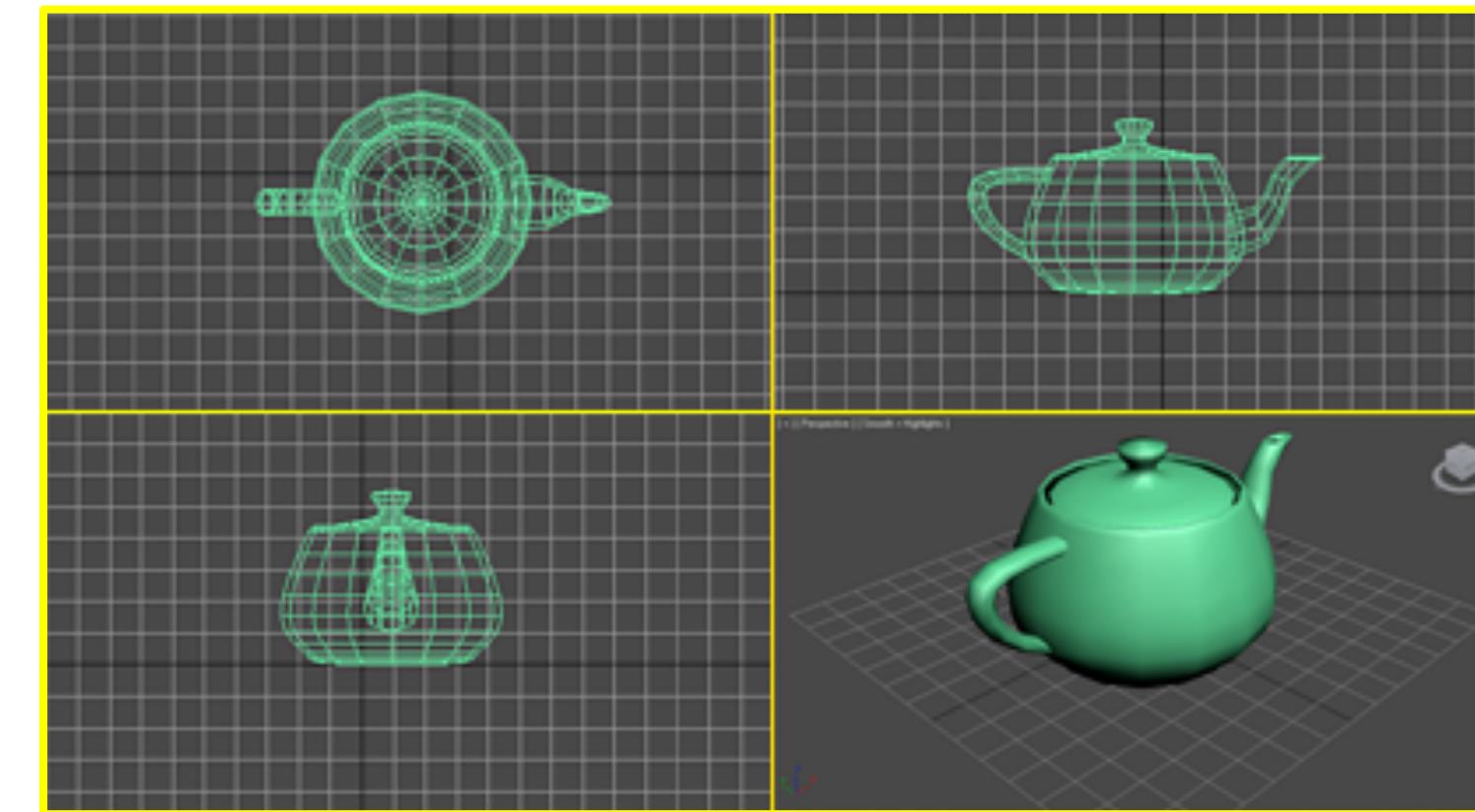
<http://youtu.be/EMx6yNe23ug>

# Digital Shape Modeling

- How do shapes find their way into computers?
  - Geometric modeling is difficult



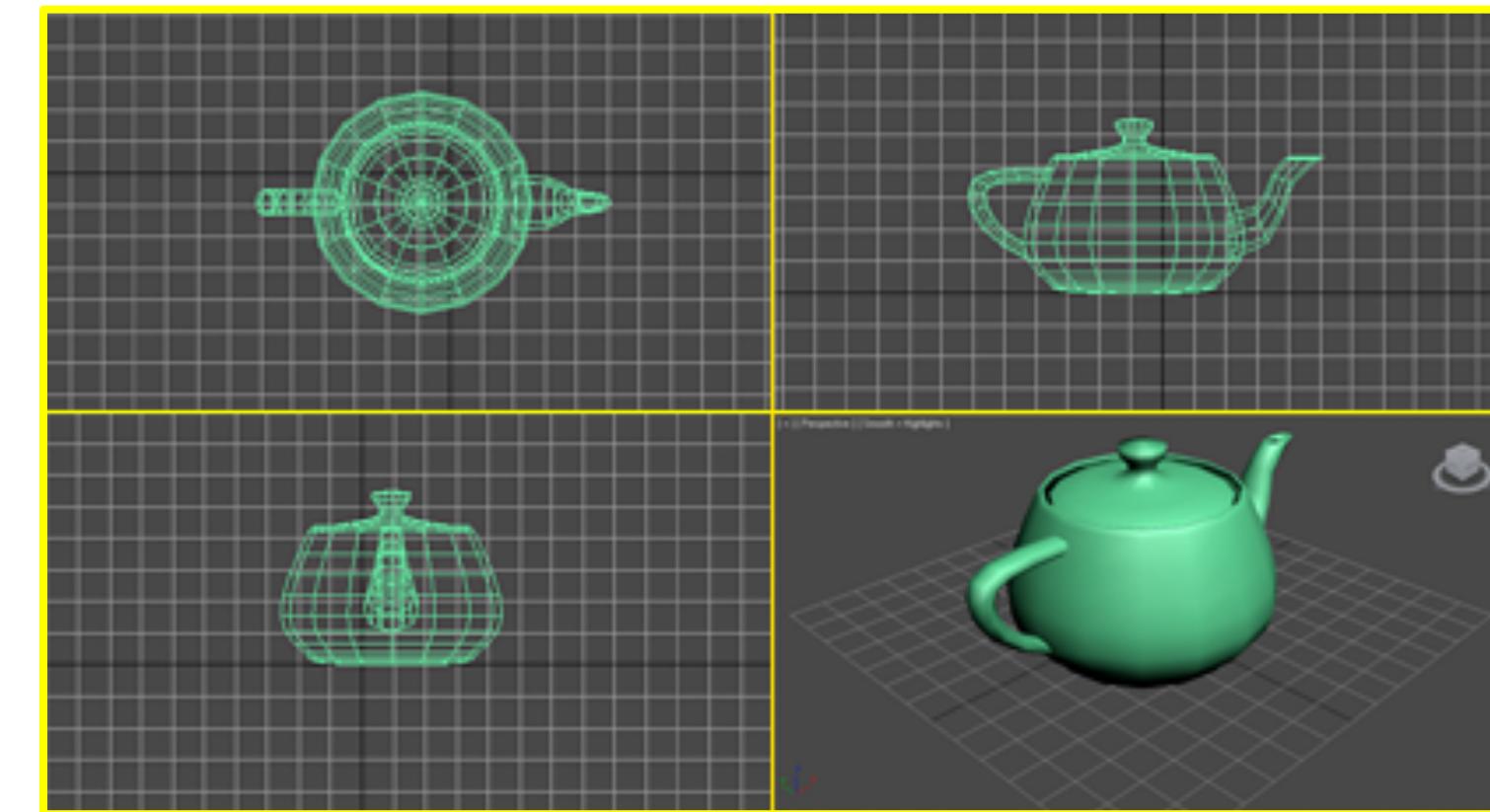
Humans have no  
direct “video out”



“Translation” from 2D  
to 3D is hard

# Digital Shape Modeling

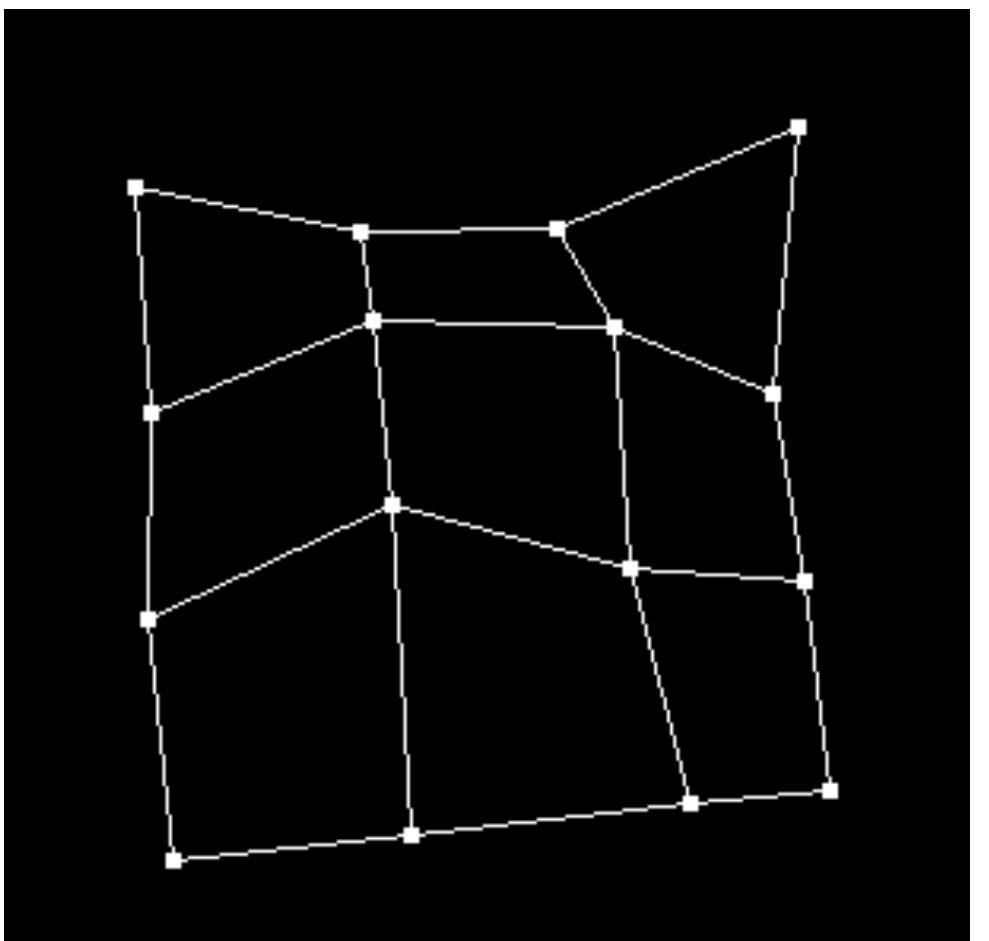
- How do shapes find their way into computers?
  - Geometric modeling is difficult



Use computation to compensate for lack of direct ability to convey visual information

# Computer-Aided Geometric Design

- Traditional pipeline for modeling shapes from scratch

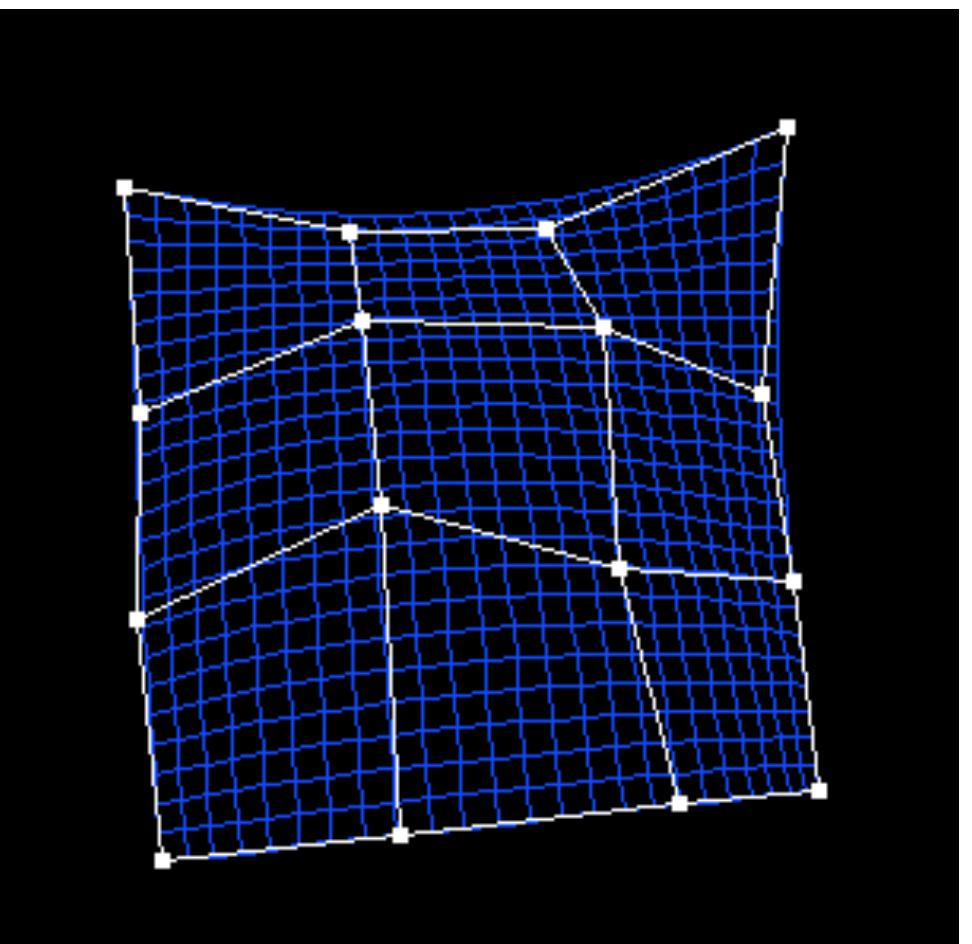


User defines a layout  
of surface patches and  
control points



# Computer-Aided Geometric Design

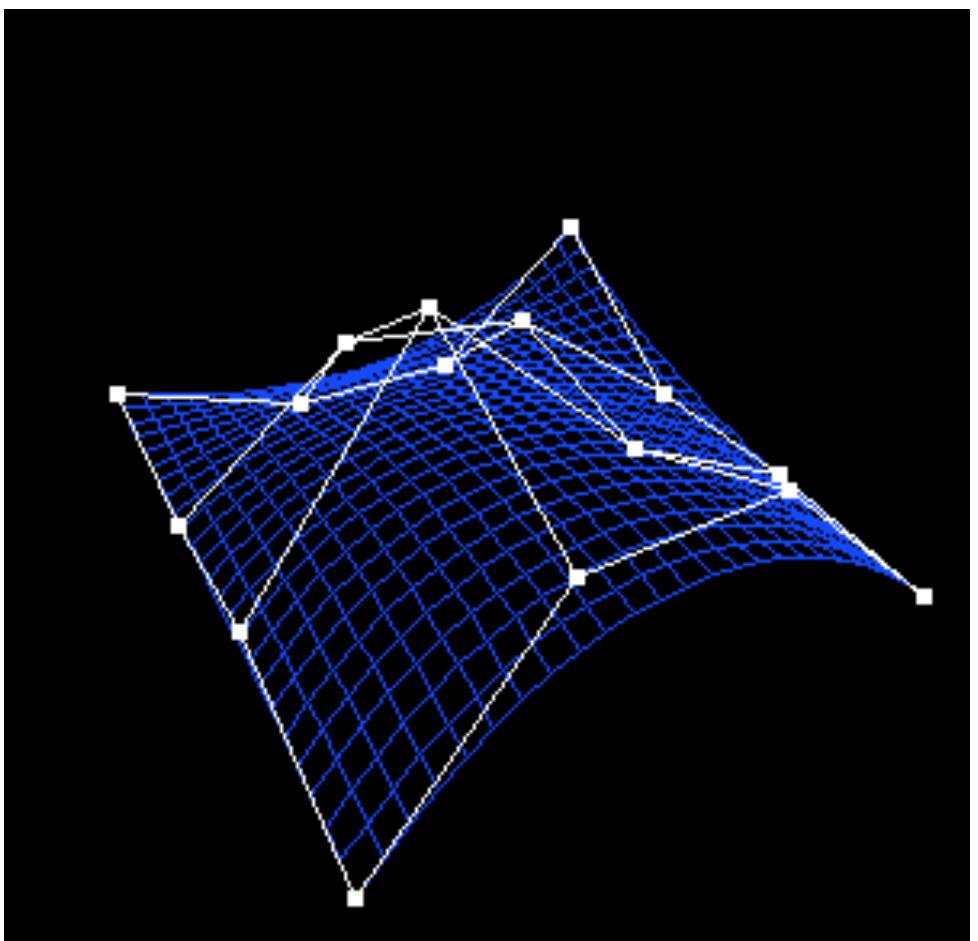
- Traditional pipeline for modeling shapes from scratch



User defines a layout  
of surface patches and  
control points

# Computer-Aided Geometric Design

- Traditional pipeline for modeling shapes from scratch

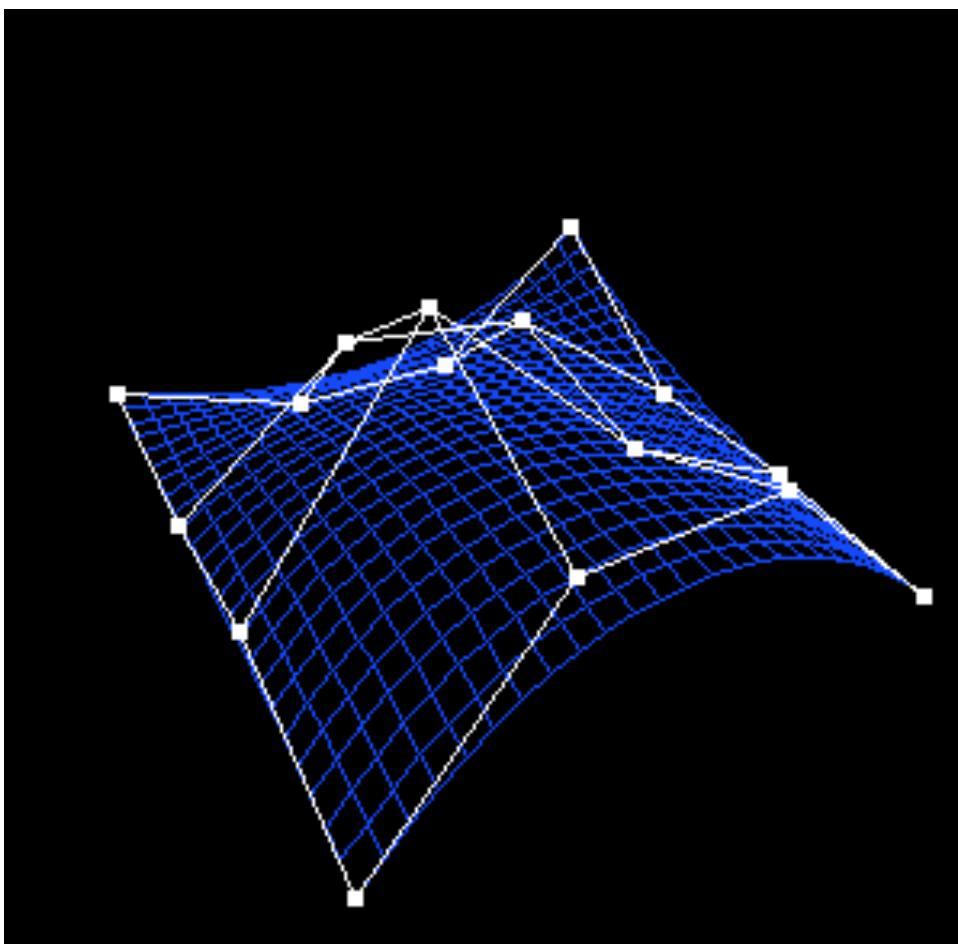


User defines a layout  
of surface patches and  
control points

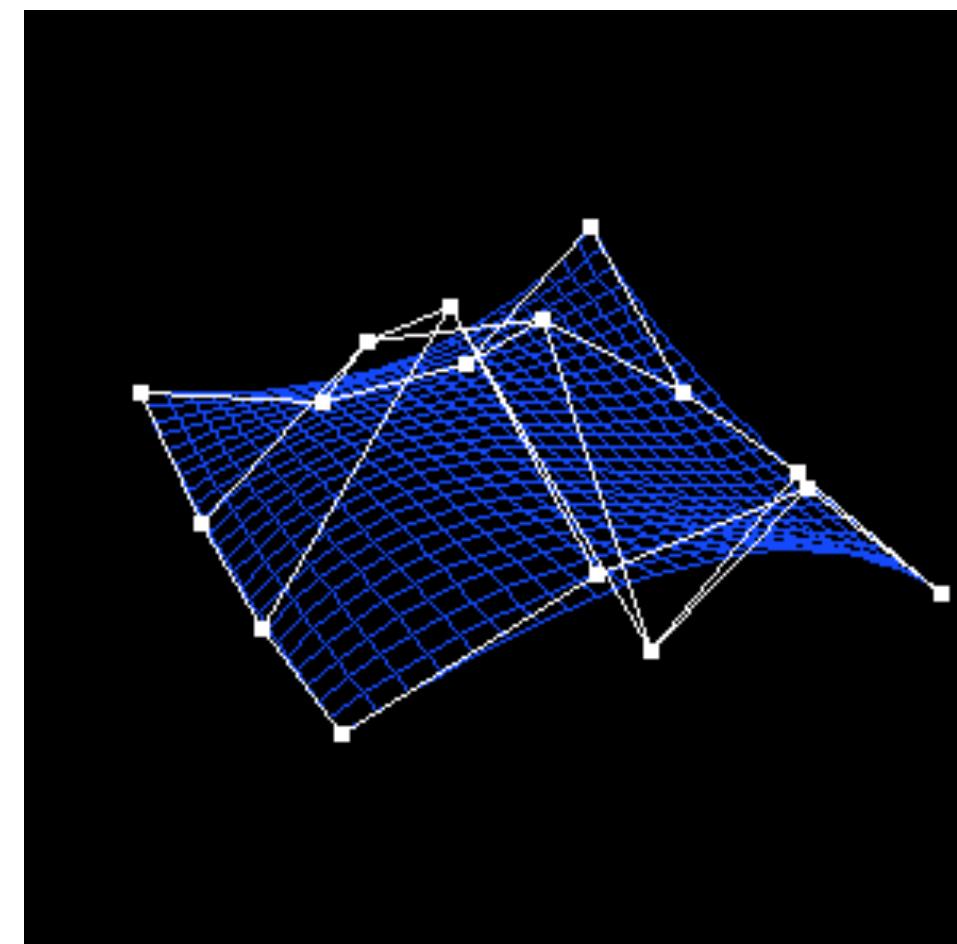
Editing is performed  
by moving control  
points and/or  
prescribing tangents

# Computer-Aided Geometric Design

- Traditional pipeline for modeling shapes from scratch



User defines a layout  
of surface patches and  
control points



Editing is performed  
by moving control  
points and/or  
prescribing tangents

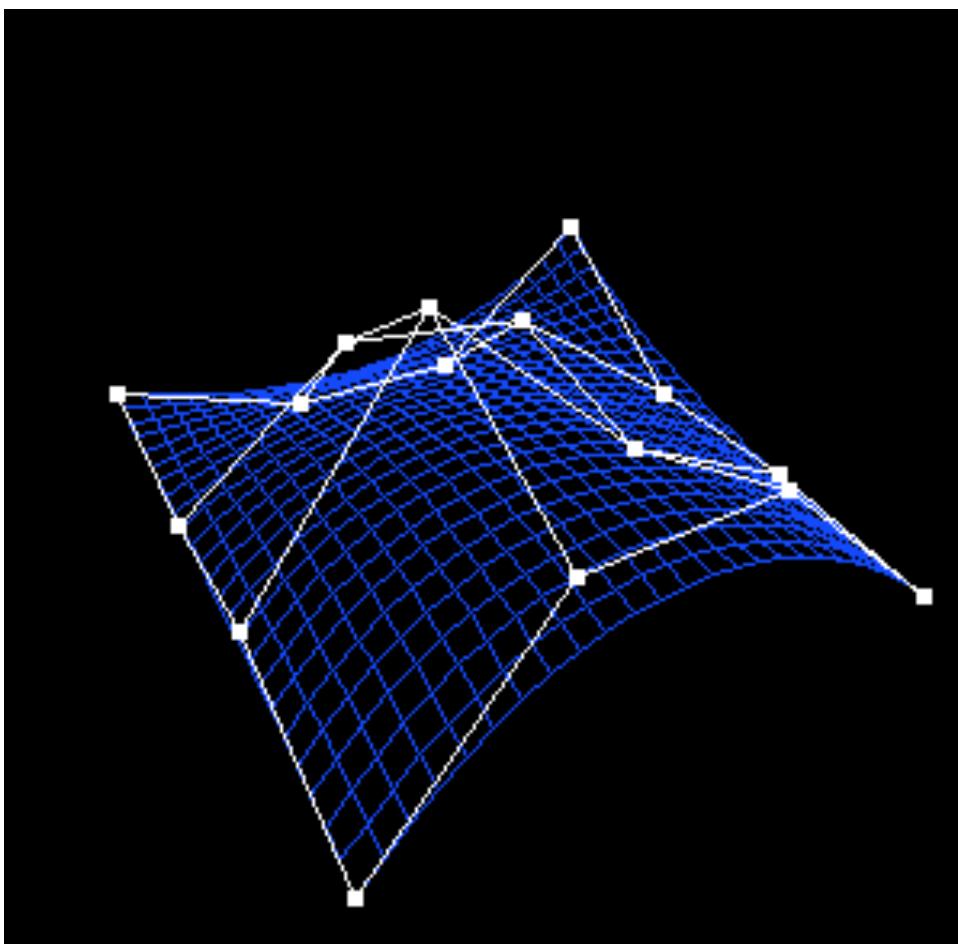


University  
of Victoria

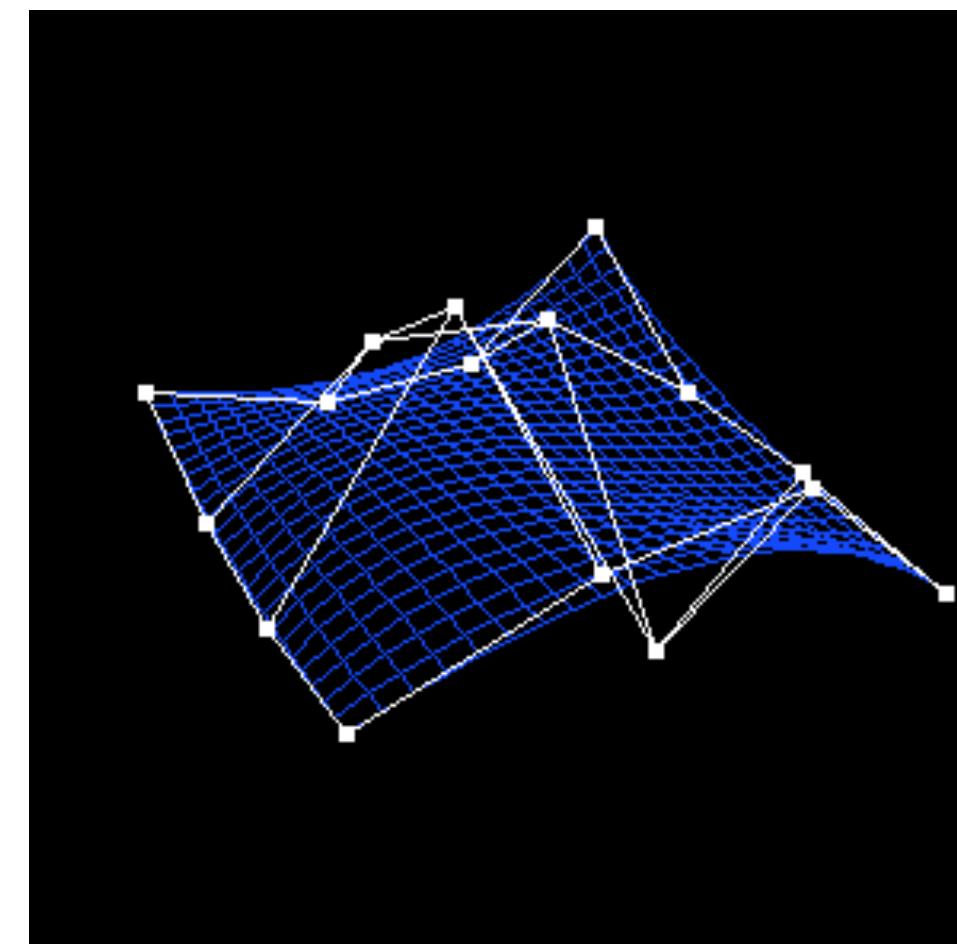
Computer Science

# Computer-Aided Geometric Design

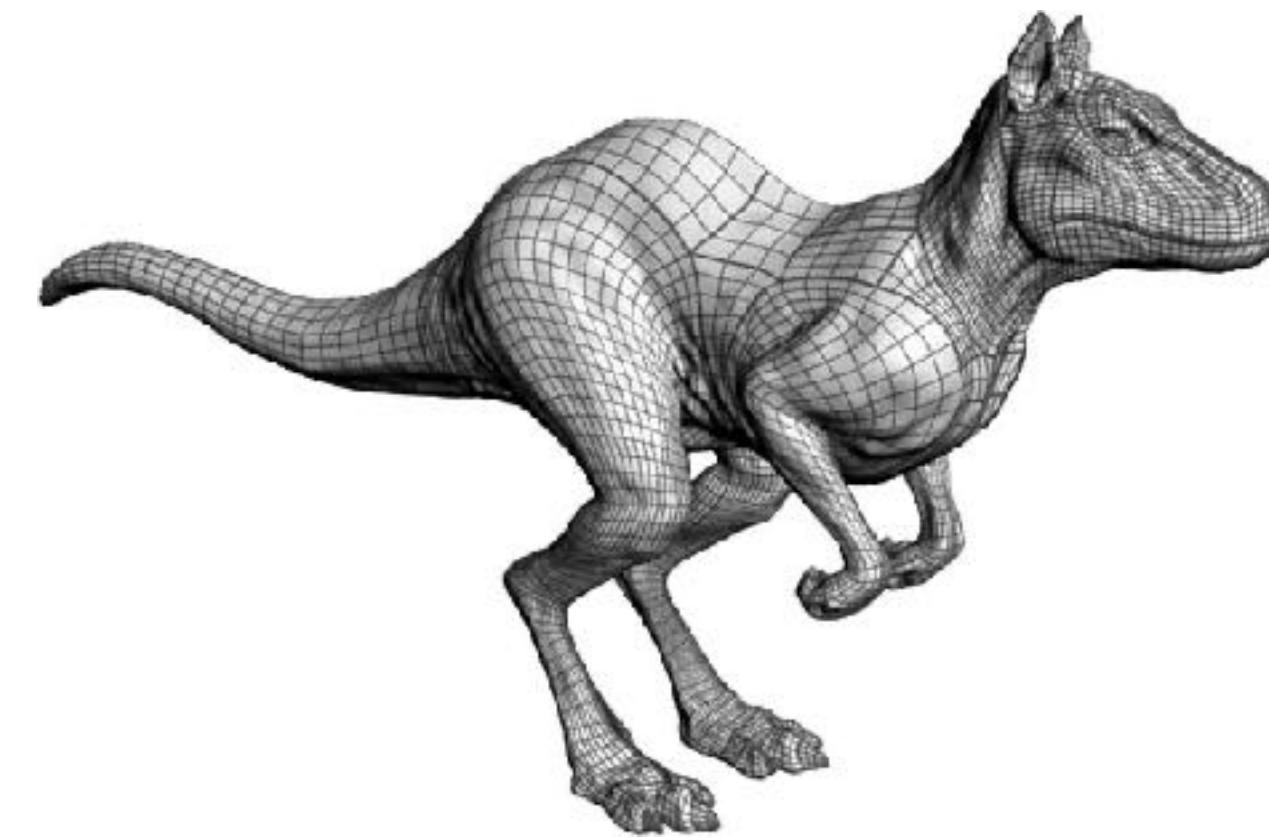
- Traditional pipeline for modeling shapes from scratch



User defines a layout of surface patches and control points

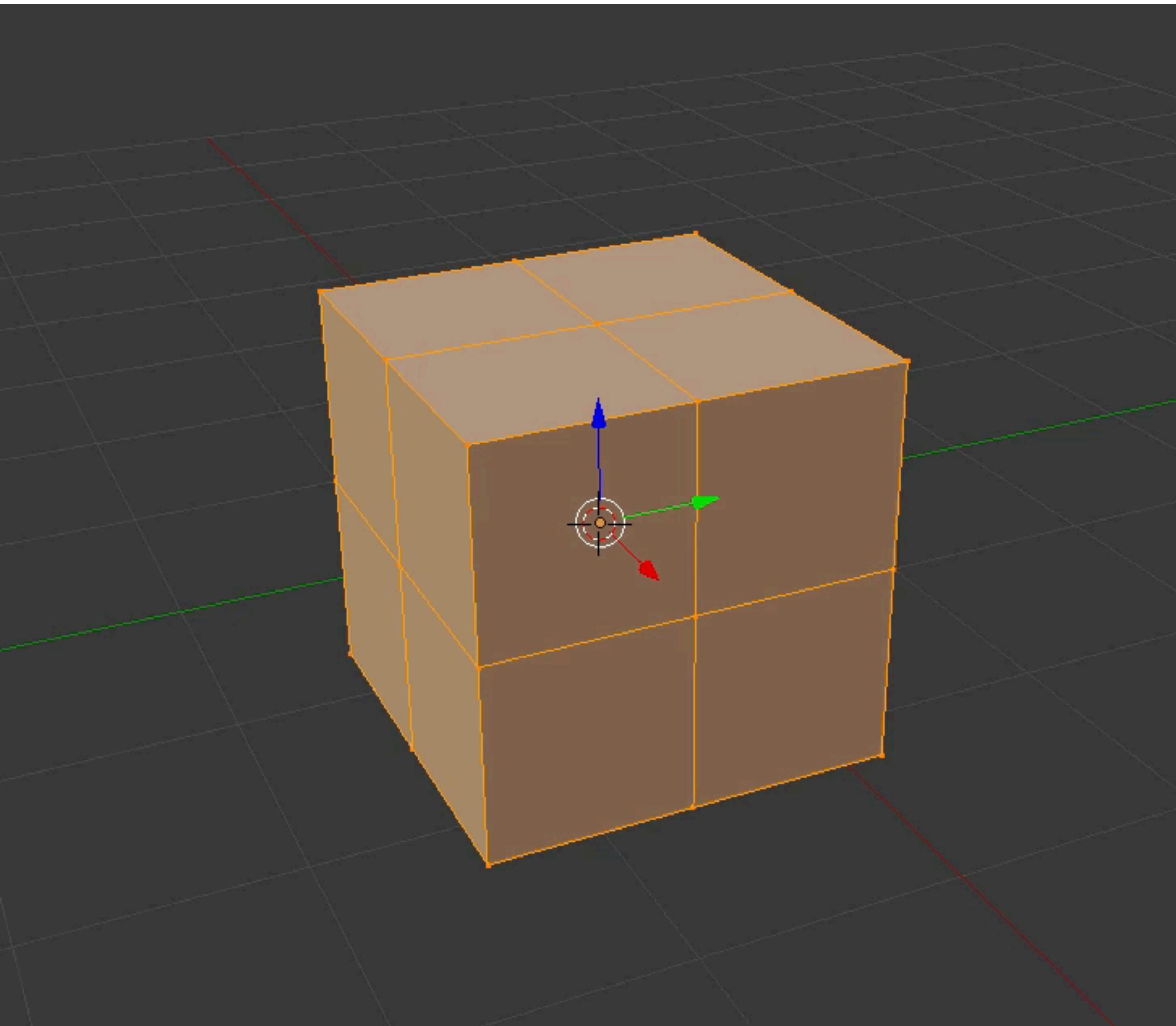


Editing is performed by moving control points and/or prescribing tangents



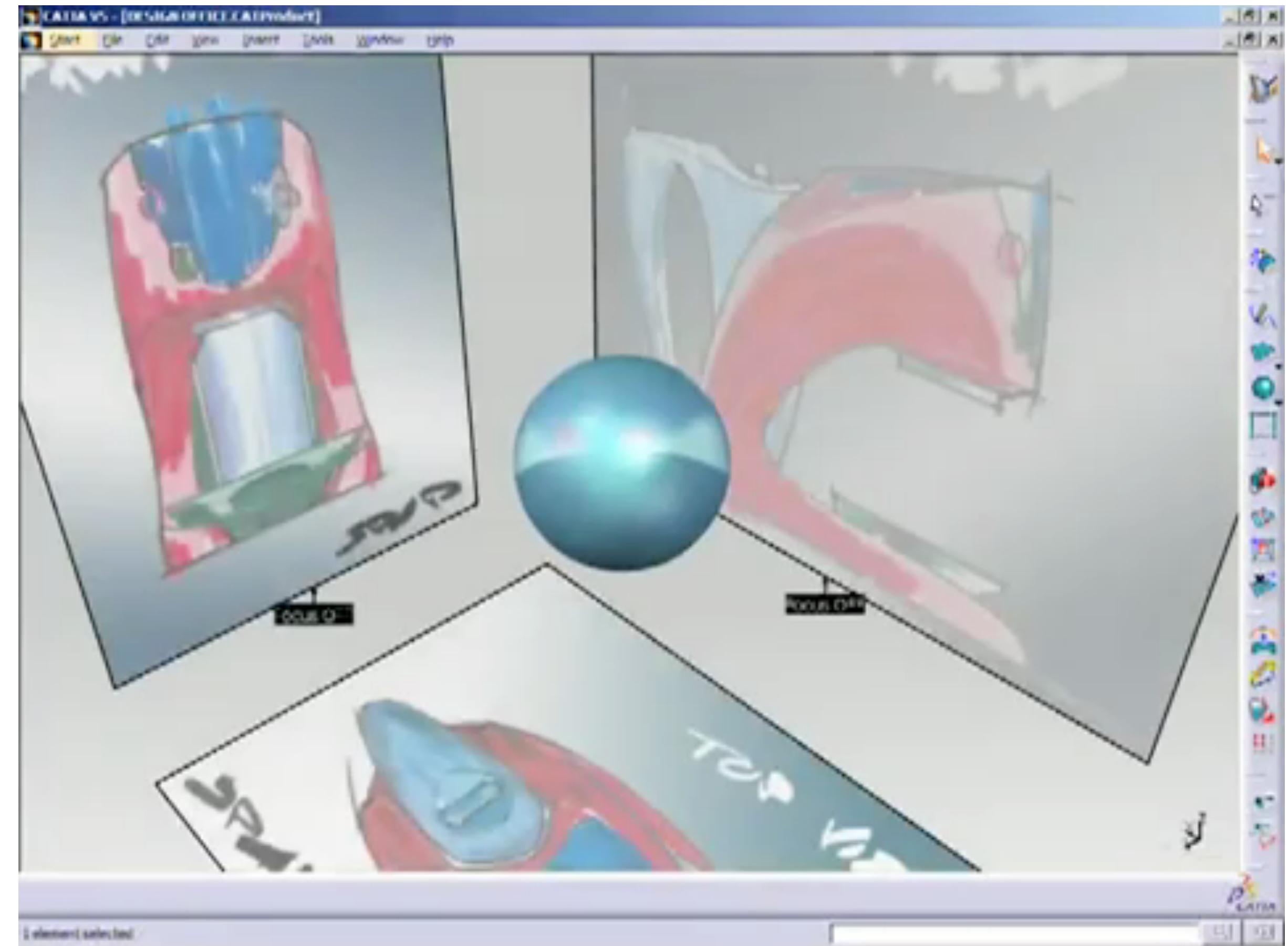
Patch-based construction of a surface

# Blender Demo



# Computer-Aided Geometric Design

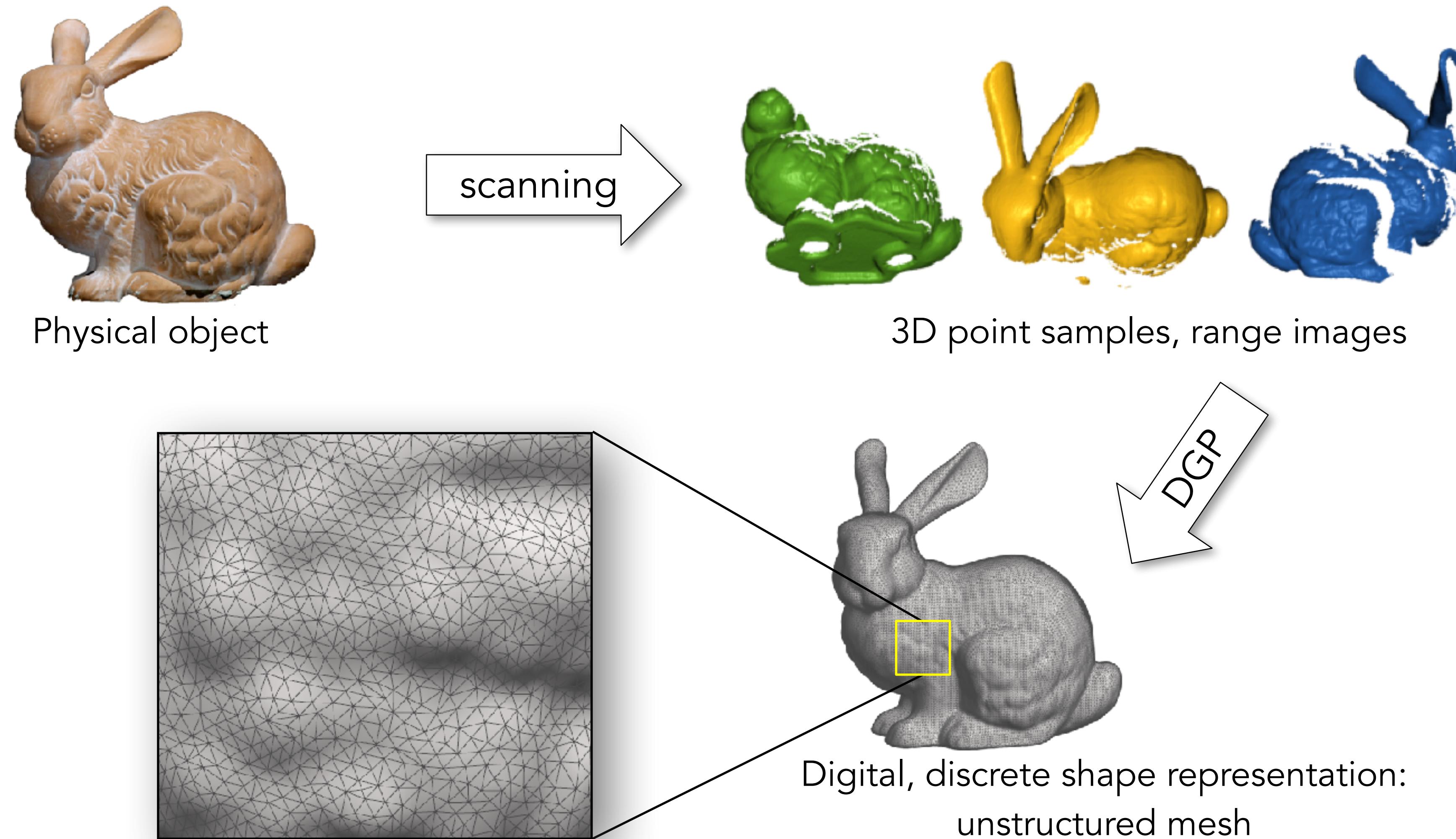
- High-quality surfaces
- Constrained modeling
- Requires a specific idea of the object first
  - Not easy to experiment and explore alternatives
- Requires extensive training



CATIA, Dassault Systems

<http://youtu.be/gTC5zMktMr0>

# Modern Geometry Acquisition Pipeline



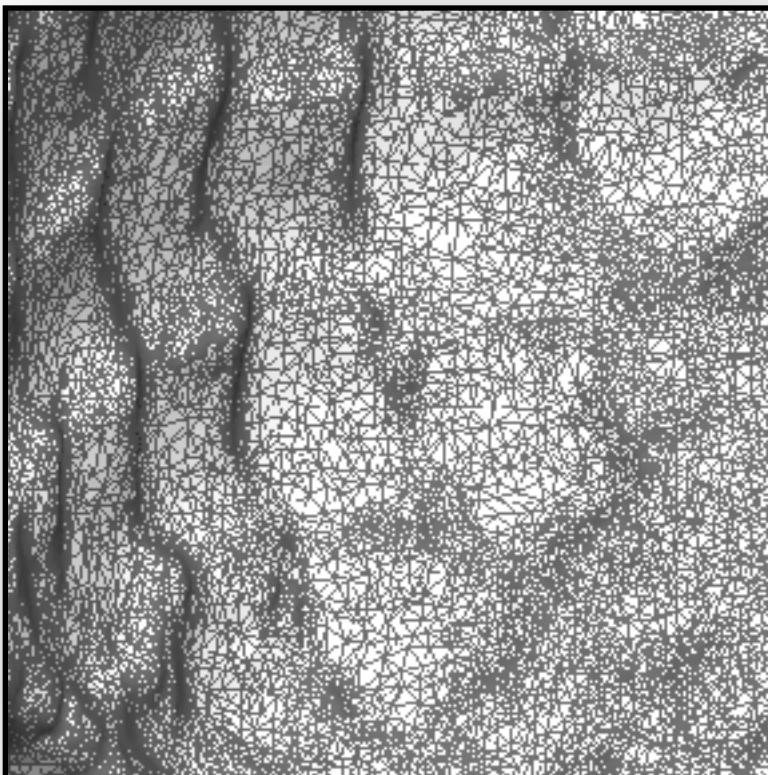
The Stanford Bunny, Stanford 3D Scanning Repository



Computer Science

# Unstructured Digital Shapes

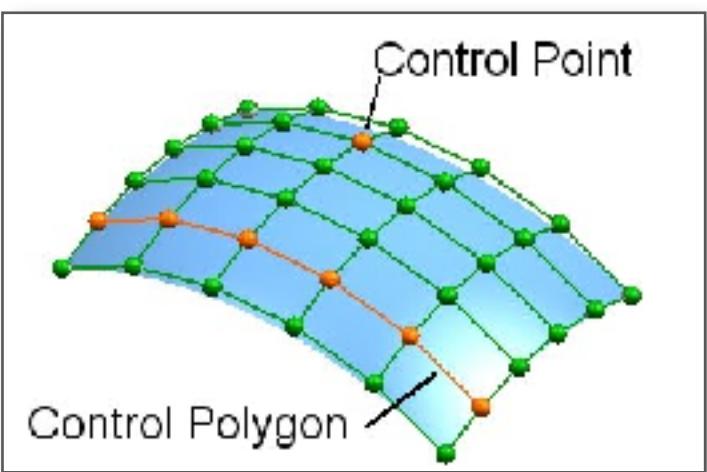
- How to **edit** and **animate**?
- How to convert to a **structured representation**?
- Computational challenge:  
very large amounts  
of data, yet modeling has to  
remain interactive



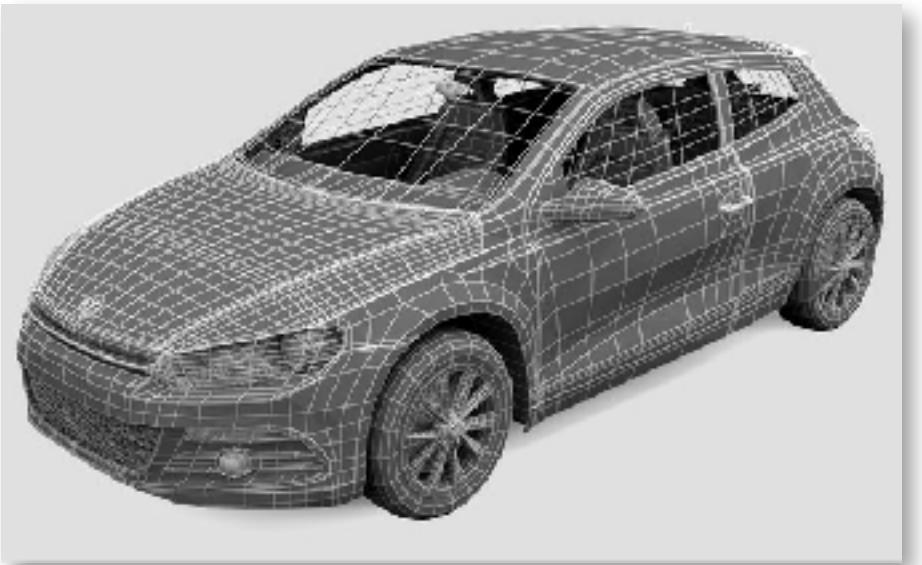
Thai statue, 10M triangles, Stanford 3D Scanning Repository

# Traditional CAD vs Modern Mesh Modeling

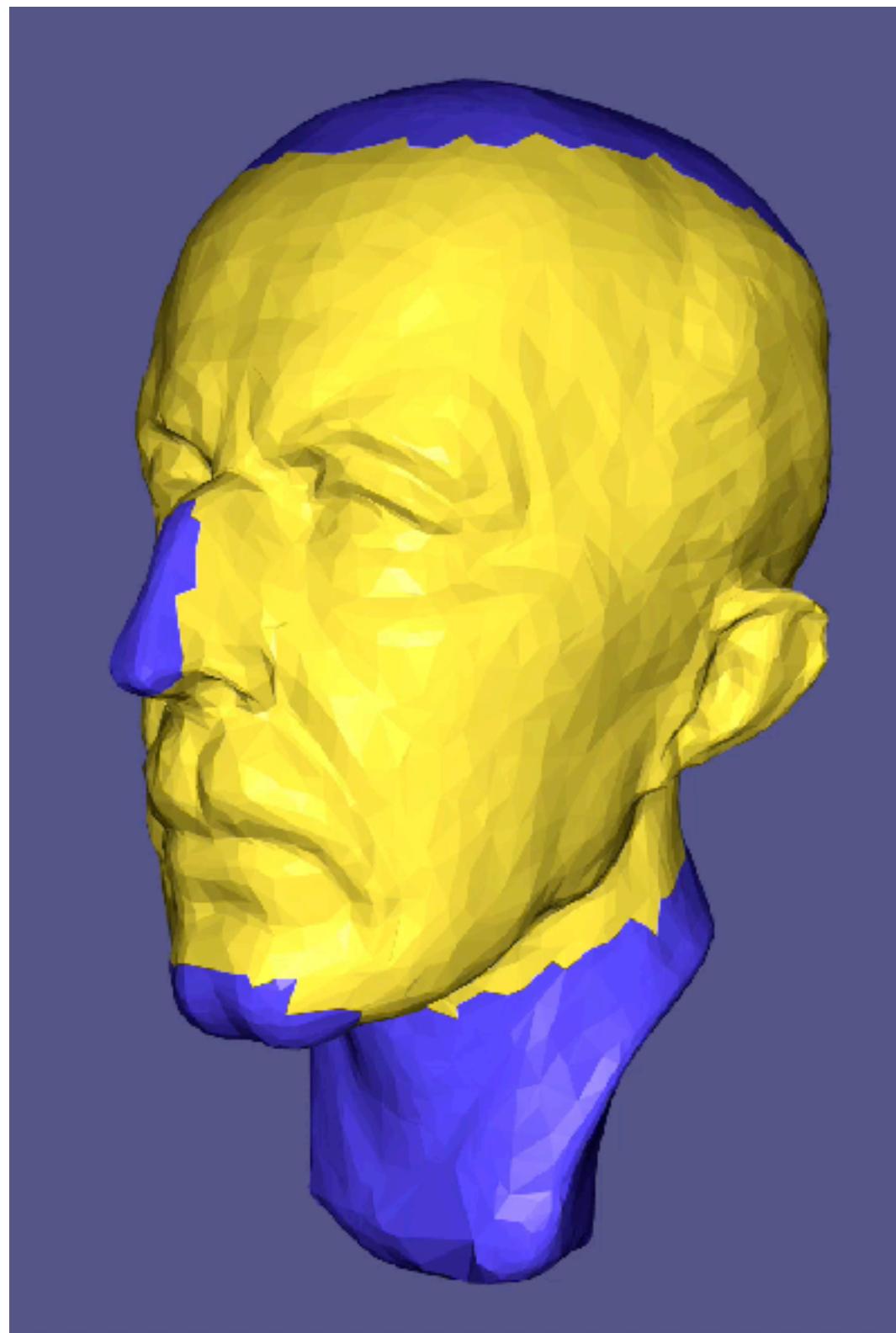
## Traditional CAD



$$\mathbf{x}(u, v) = \sum_{i,j} \mathbf{p}_{i,j} B_i(u) B_j(v)$$



## Modern mesh modeling

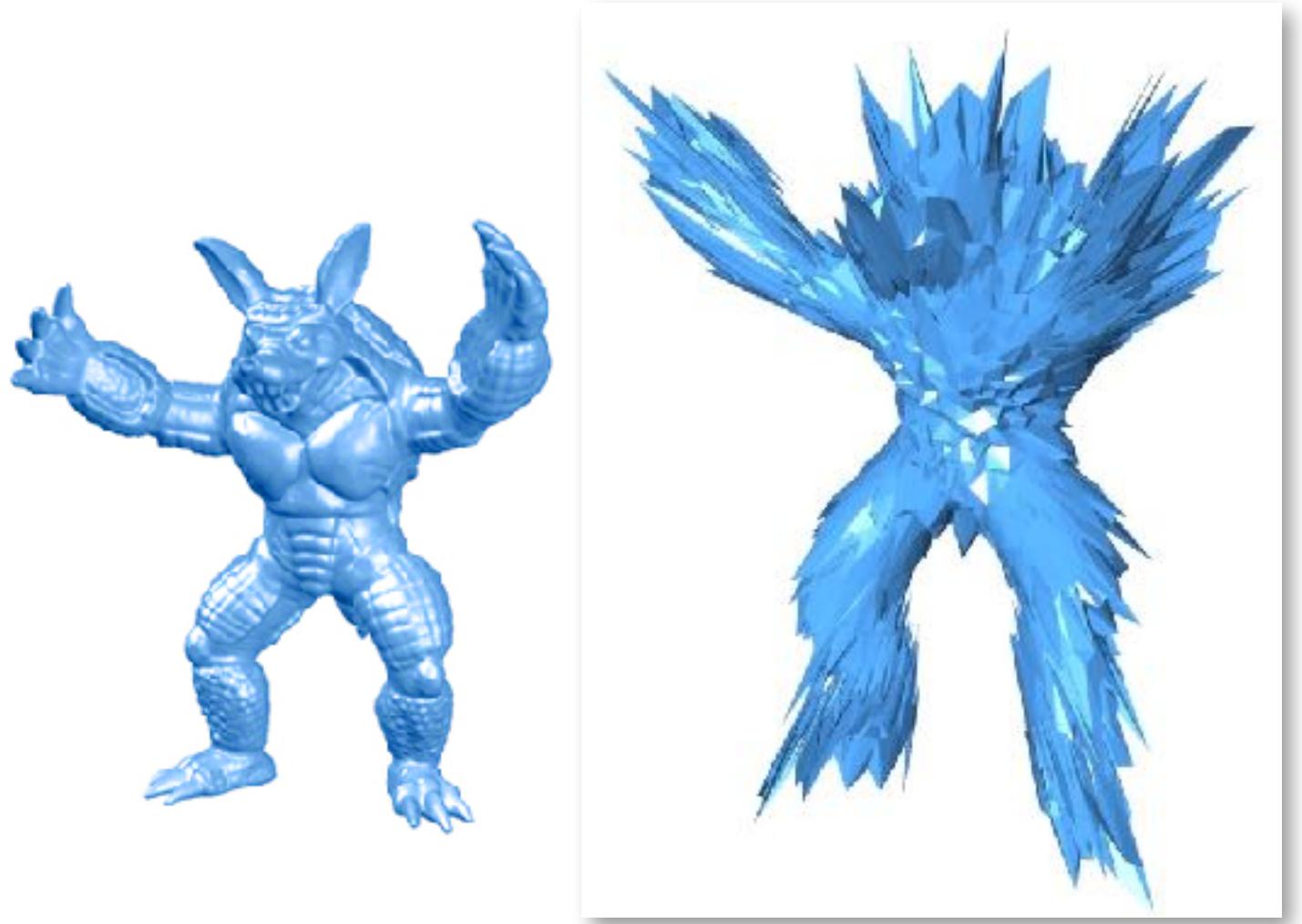


$$\min_{\mathbf{x}} \int_S E(\mathbf{x}) \quad s.t. \quad \mathbf{x}|_C = \mathbf{x}_{\text{fixed}}$$

User has more freedom!  
Select and manipulate  
arbitrary regions.

# Tools?

- Use techniques from both CS & Math
  - PDEs
  - Discrete differential geometry
  - Numerical linear algebra
  - Graph theory
  - ...
- ...combined with intuition and creativity ...
- work on real data = write/use code



# Prerequisites

- Linear Algebra
  - We **will not** cover the concepts that you need. If you are not familiar with basis, points, vectors, matrices, and linear systems, the course will be hard to follow.
- Python
  - We **will not** review the basic concepts of Python and Jupyter notebooks. There will be plenty of examples given, it should be easy to catch up if you never used it before.
  - Alternatively, you can use C++ for the assignments. However, no starting code will be provided.

# Organization

- Course Website
- Weekly Lecture
- Office Hours

# Organization

- Weekly live lectures
  - TWF 10:30am - 11:20am MacLaurin Building D116
- Office hours:
  - Wednesdays 4:00pm- 5:00pm ECS 612
  - By appointment
- Final exam
  - TBA

# Lectures

- I will upload the slides on the website, so that you can directly annotate them
- For every class, I will always add references in the end to the textbook and/or external resources
- You are encourage to take a look at the material **before** I present it in class

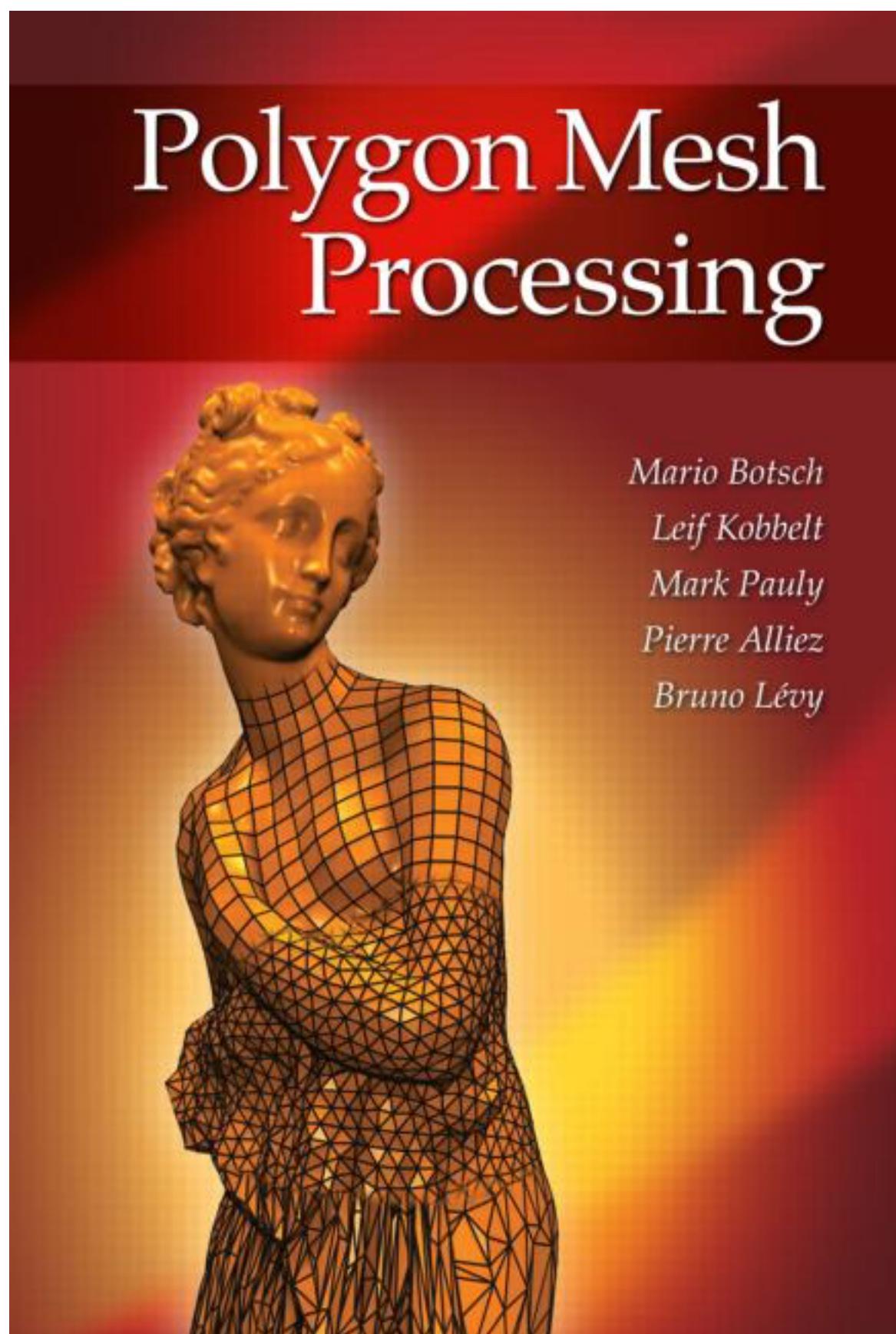
# Lectures

- Please interrupt me at any time to ask questions

# Final Coding Project

- Individual project, we will publish the rules later but you are essentially free to do whatever you want, as long as it requires geometry processing
- The project will be presented in a fast-forward session at the end of semester (3 minutes presentation per project)

# Material



Polygon Mesh Processing



<https://libigl.github.io/libigl-python-bindings/>



<https://www.wikipedia.org>

# Grading

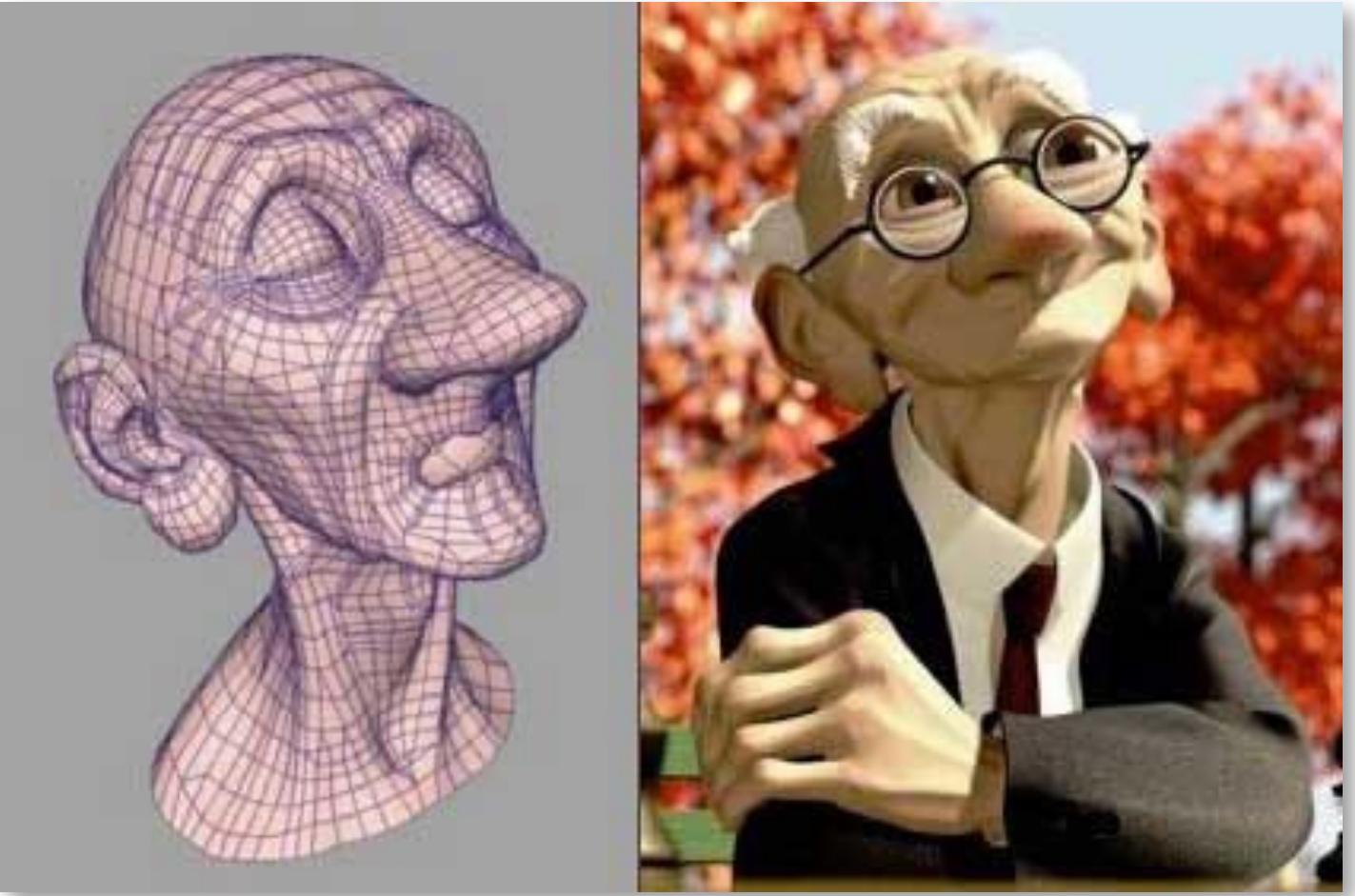
- **Assignments 60% (50% Grad)**
  - Late assignments count 70%
- **Project: 25% (35% Grad)**
- **Final Exam: 15%**
- There will be optional tasks, that will allow you to recover points lost in the assignments

# Policy

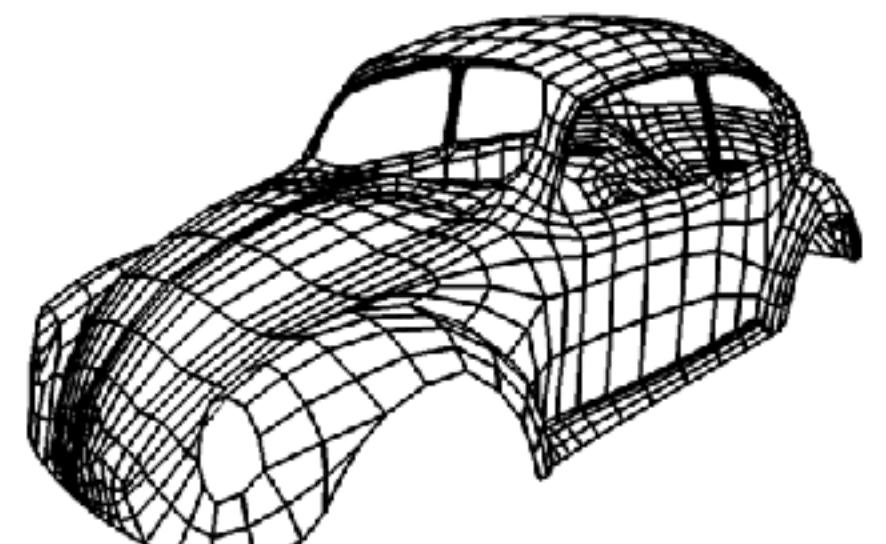
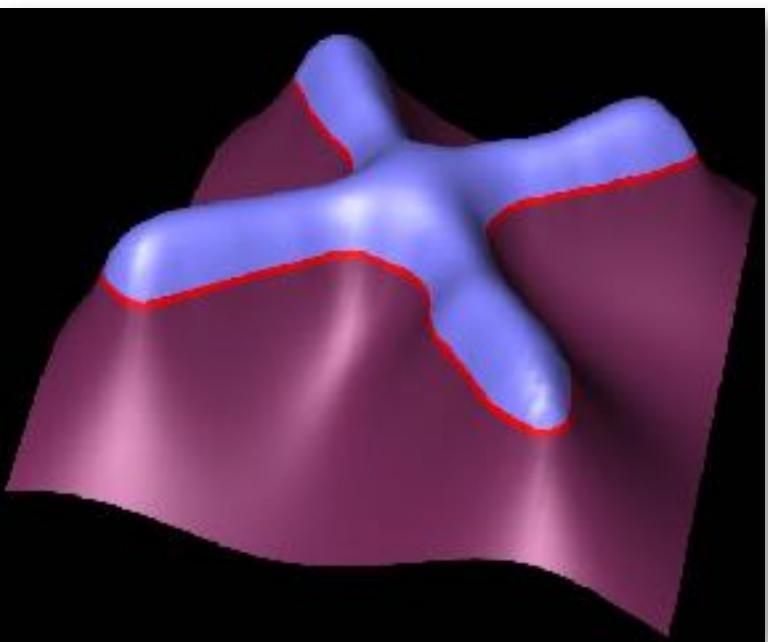
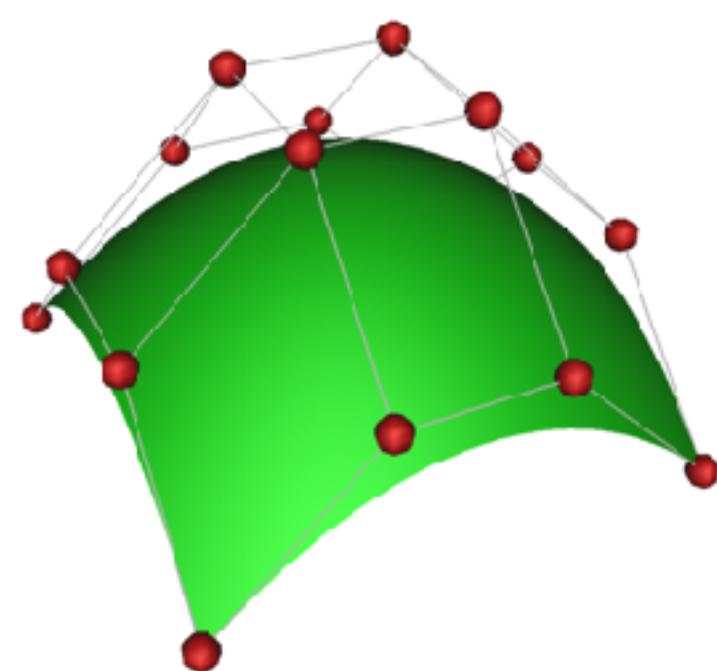
- You are encouraged to consult with your classmates/friends but collaboration in the assignments is **NOT** allowed.
- You are **NOT** allowed to copy code online or use external libraries (except those provided in the class) for the first 3 assignments.
- We will use plagiarism tools to validate all homework. Plagiarism will be punished with a zero-tolerance policy: **the minimal penalty will be a score of 0 points on the assignment, the reduction of 1 letter grade in the final score, and a permanent ban on university-related jobs.**

# Course Topics

- Overview of shape representations
  - Parametric curves/surfaces
  - Implicit
  - Polygonal meshes

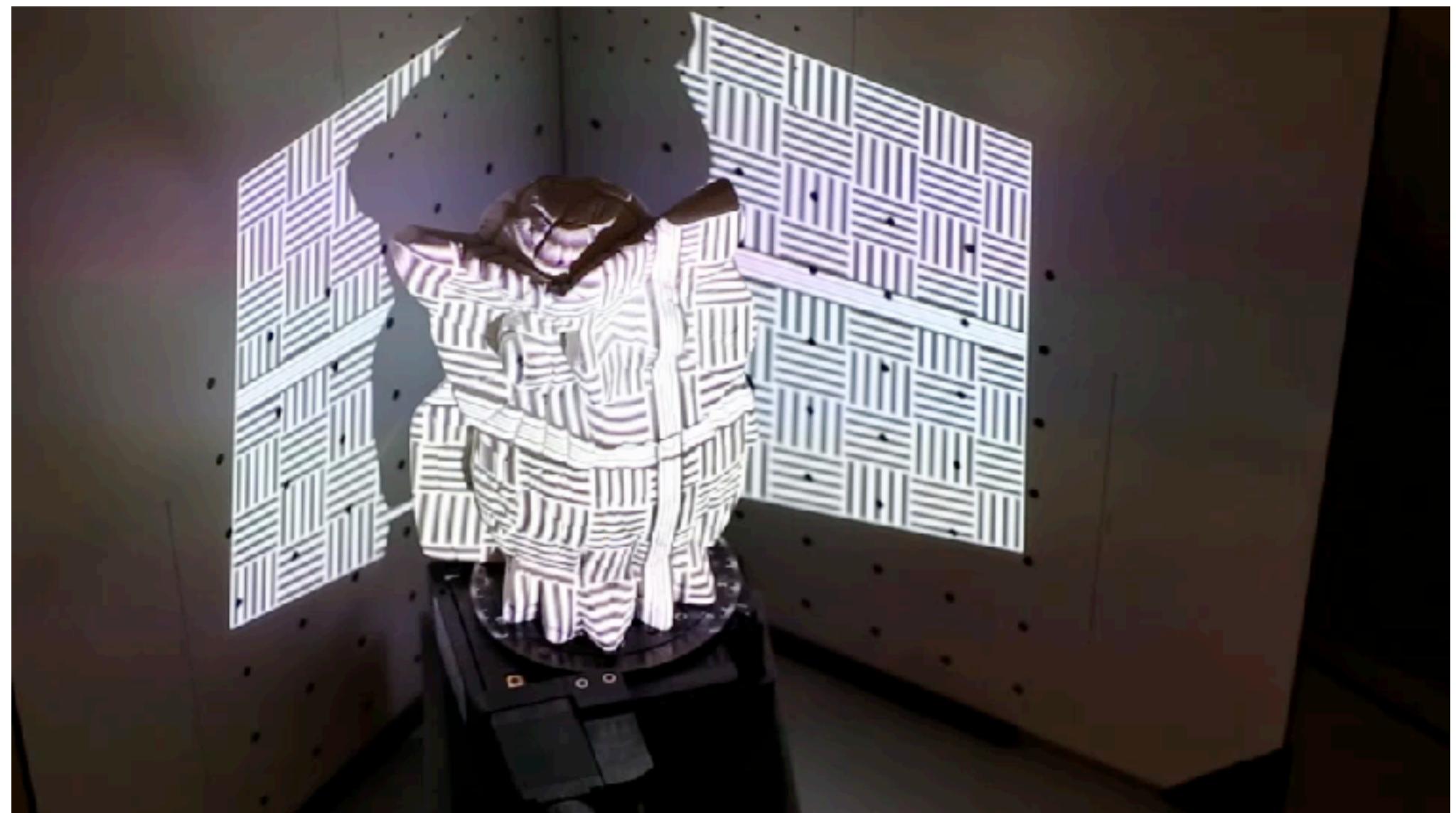
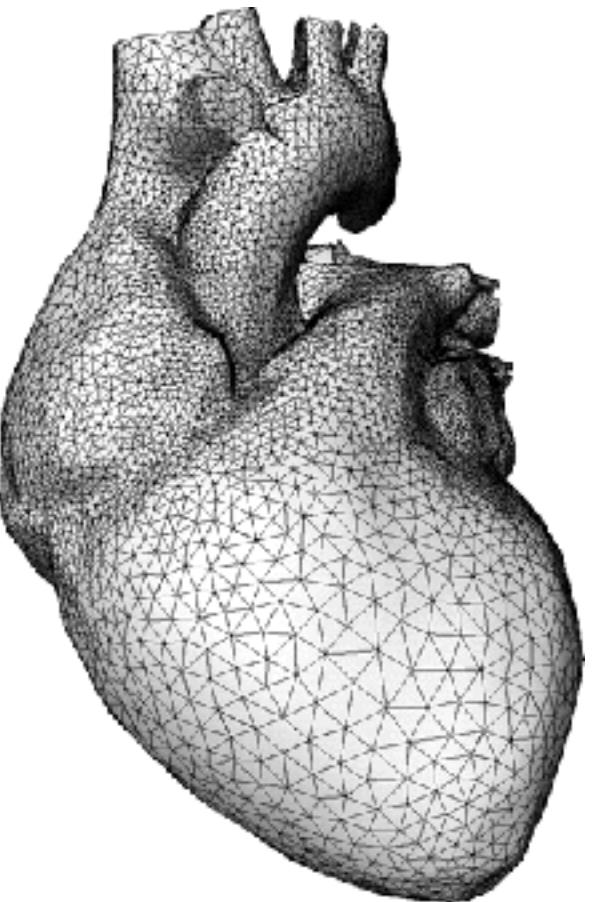
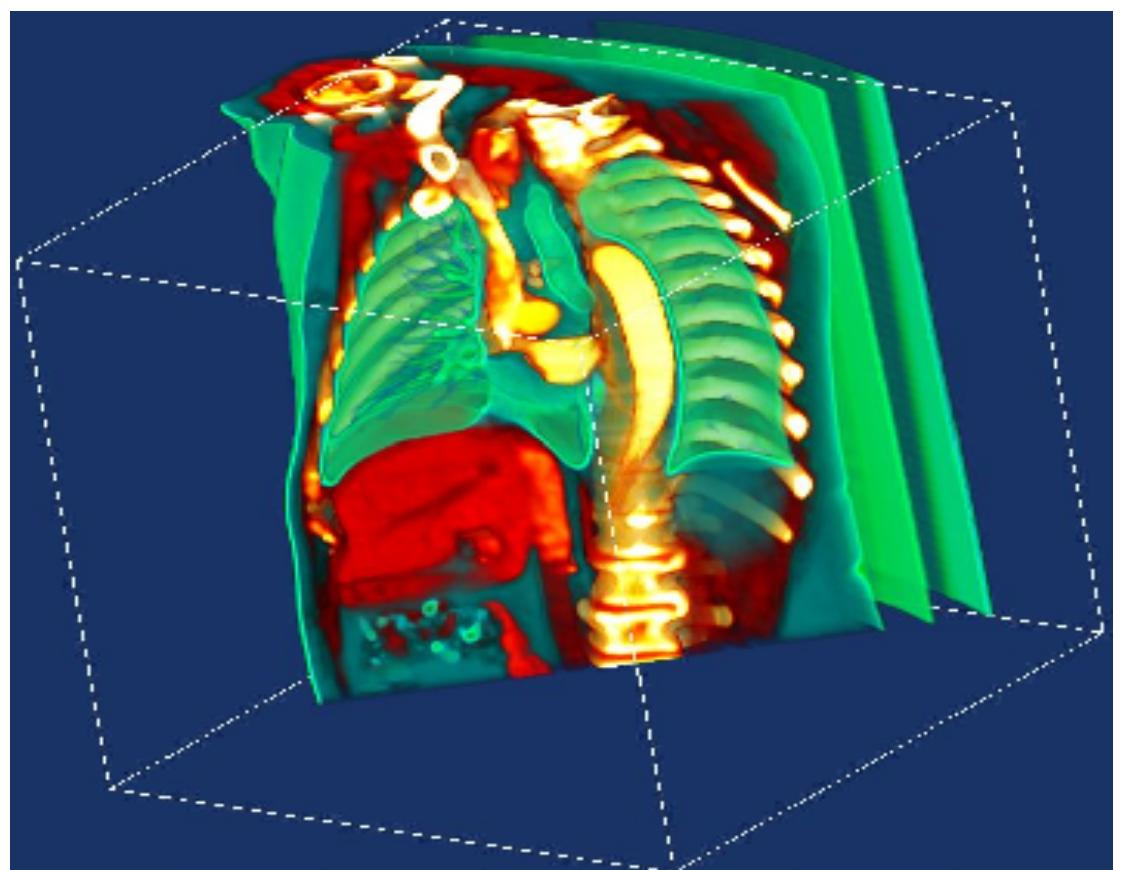
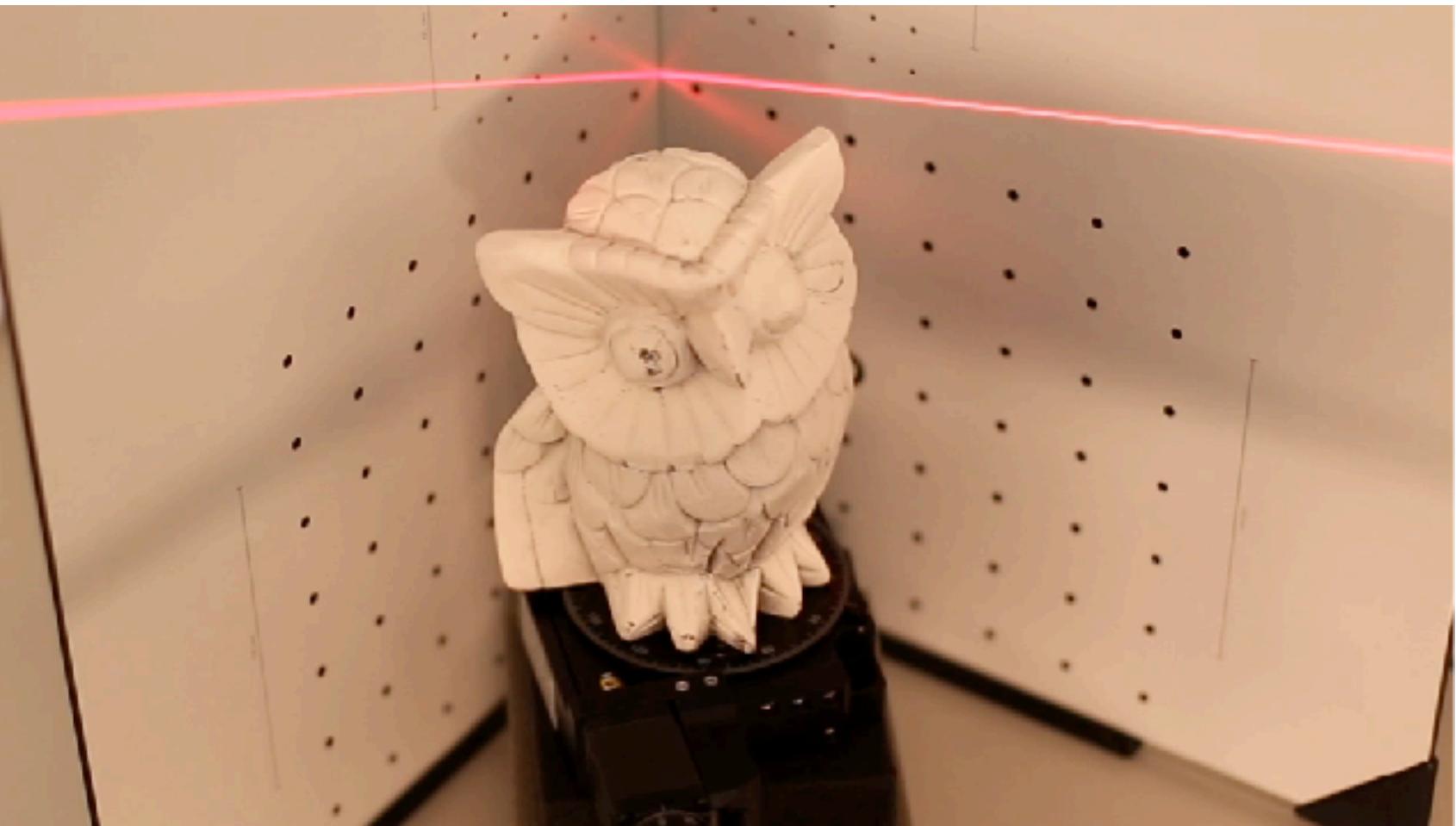


Pixar



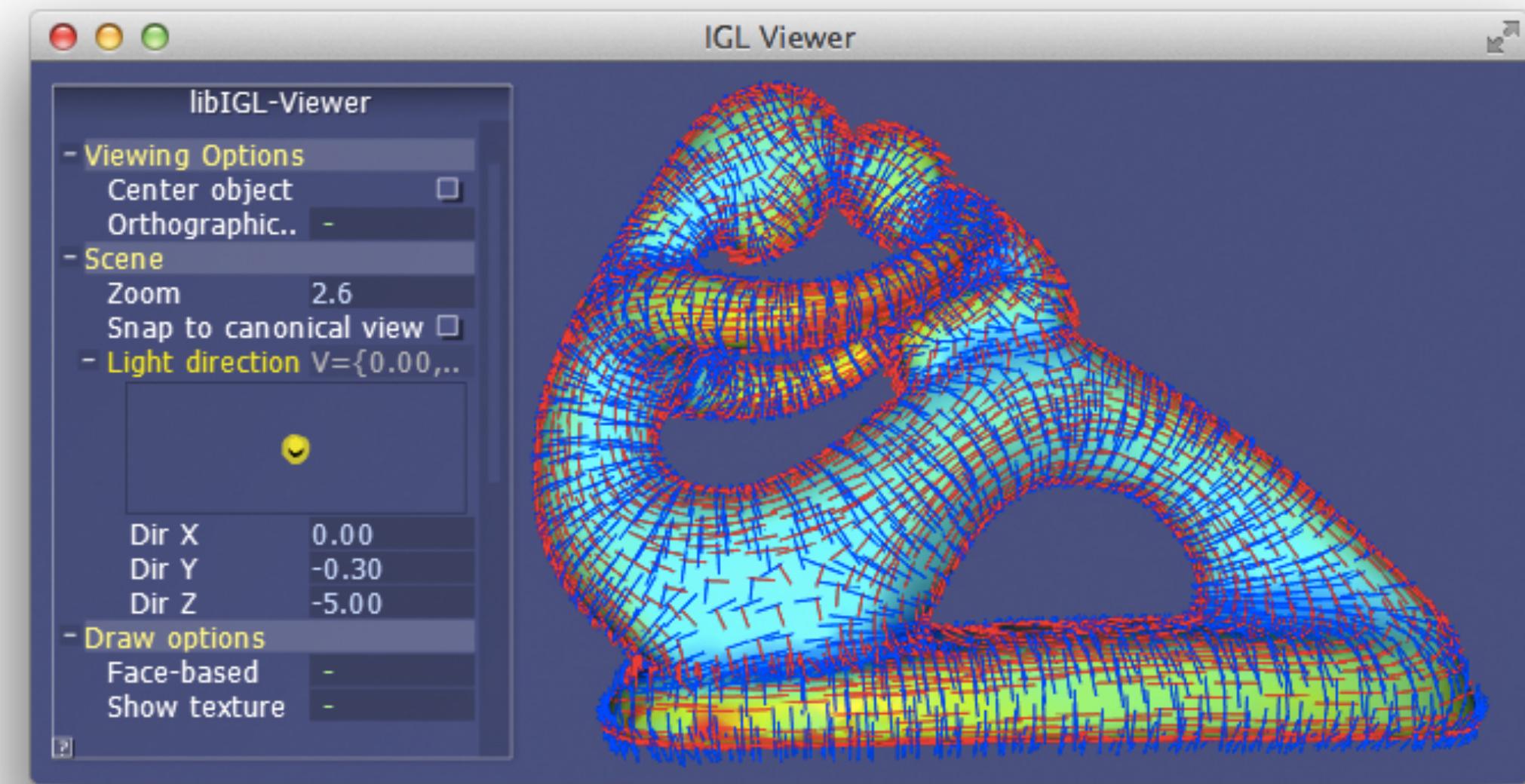
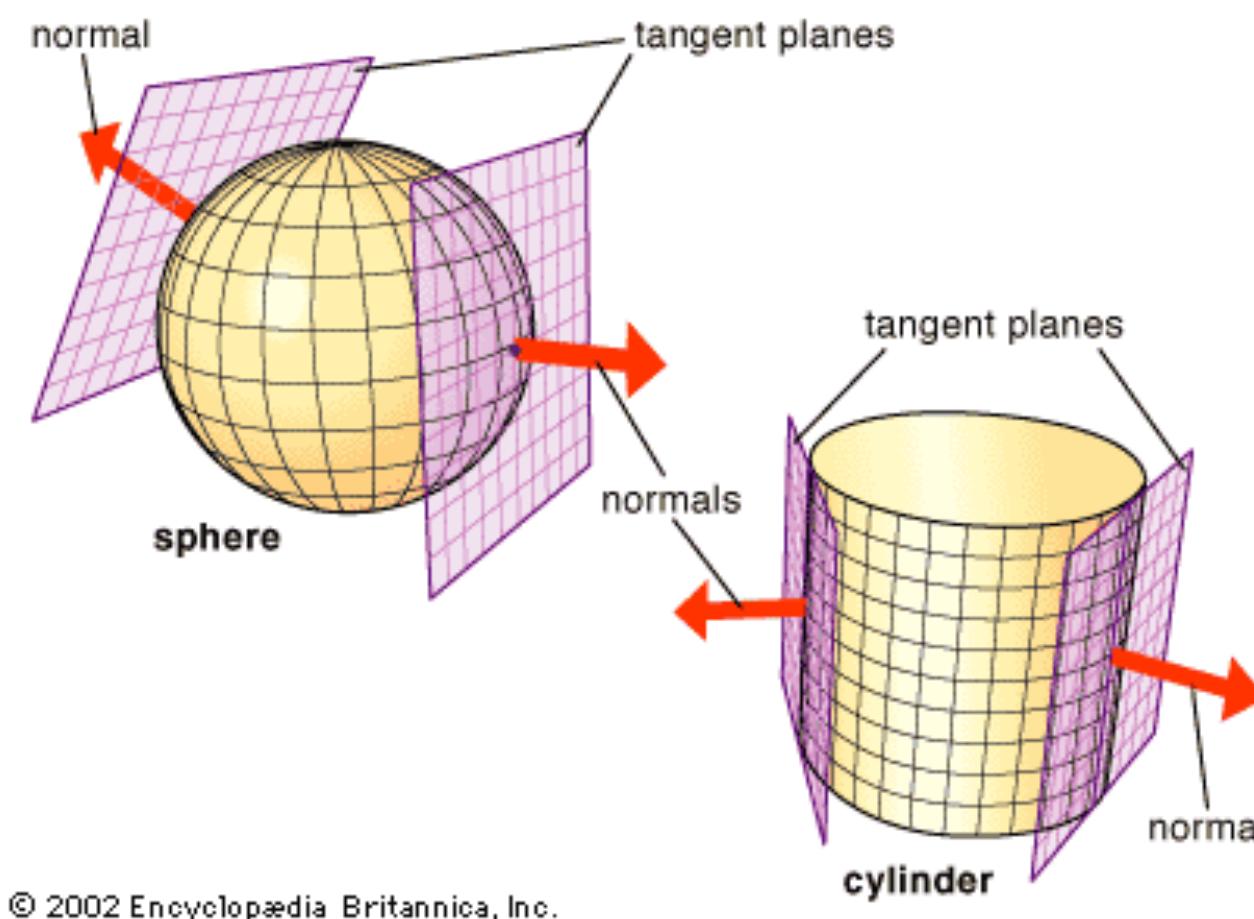
# Course Topics

- Shape acquisition
  - Scanning/imaging
  - Reconstruction



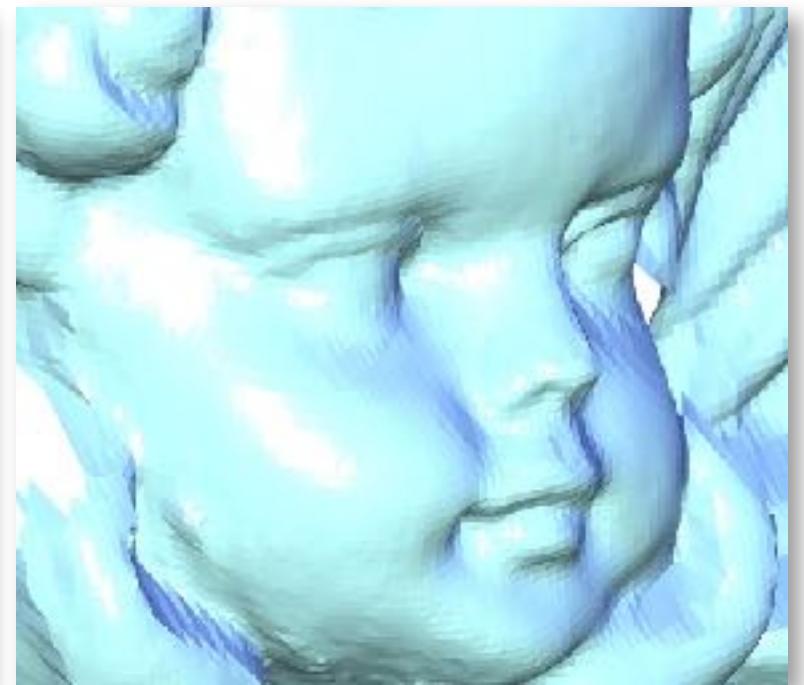
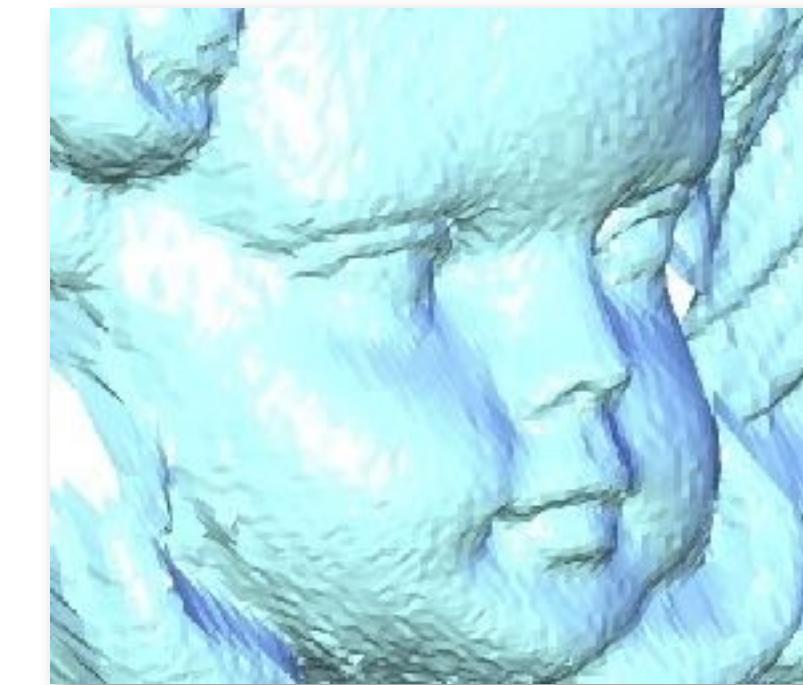
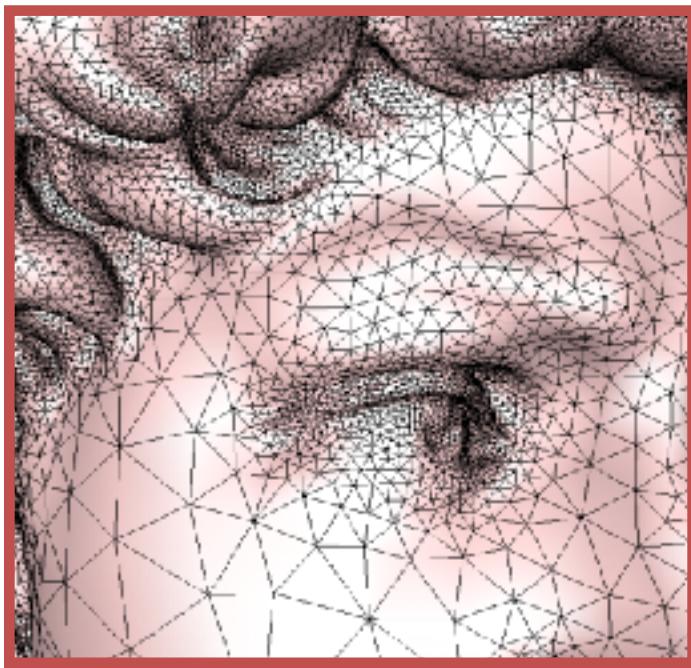
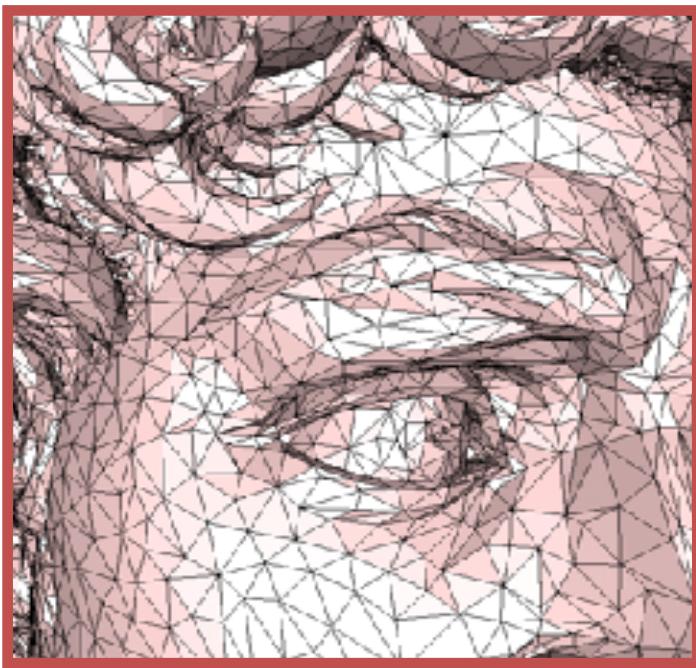
# Course Topics

- Differential geometry
  - Continuous and (mostly) discrete
  - Powerful tool to analyze and model shapes



# Course Topics

- Digital geometry processing
  - Denoising, smoothing, simplification, remeshing, parameterization, compression



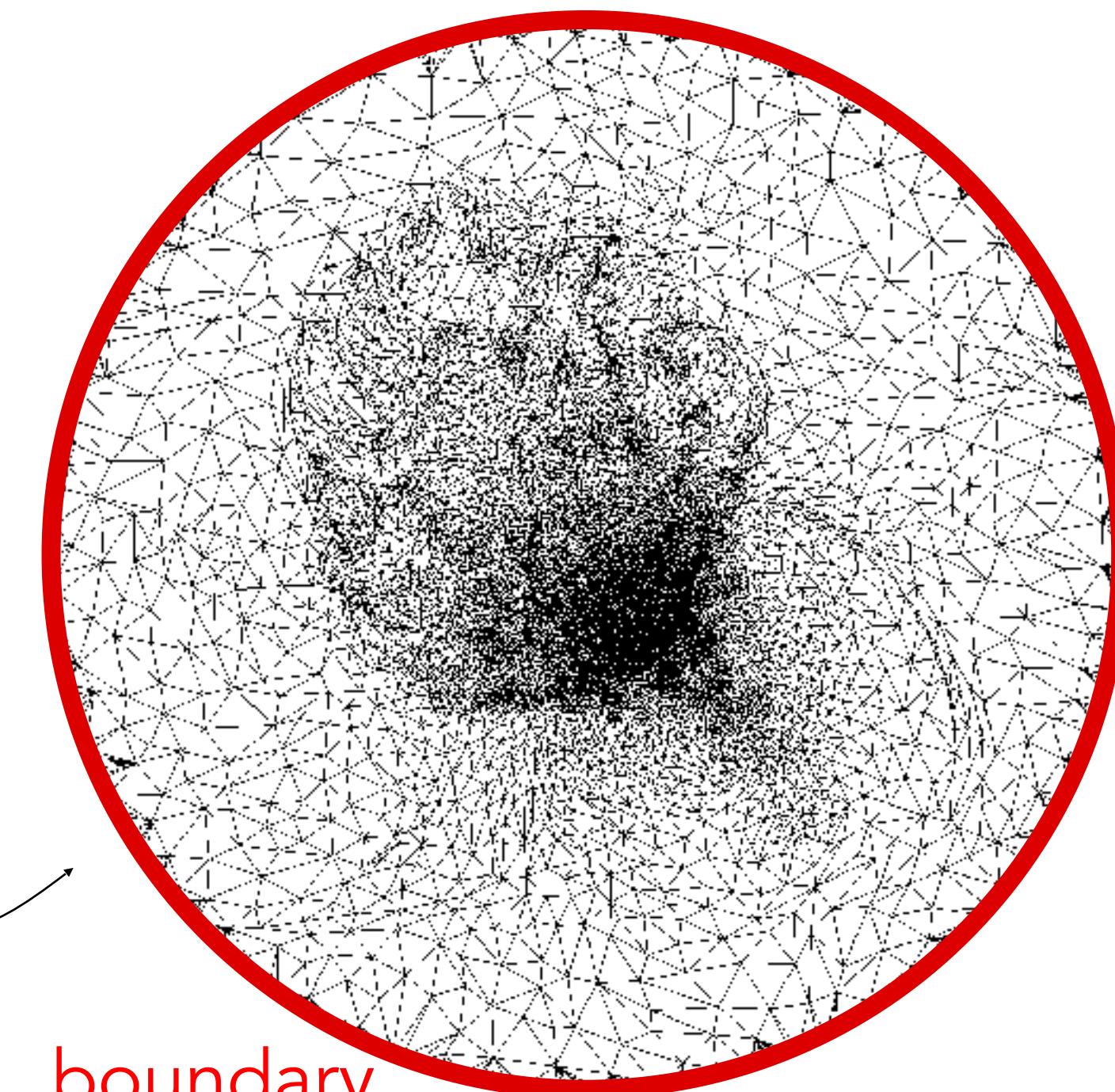
# Course Topics

- Parameterization

3D space ( $x,y,z$ )



2D parameter domain ( $u,v$ )

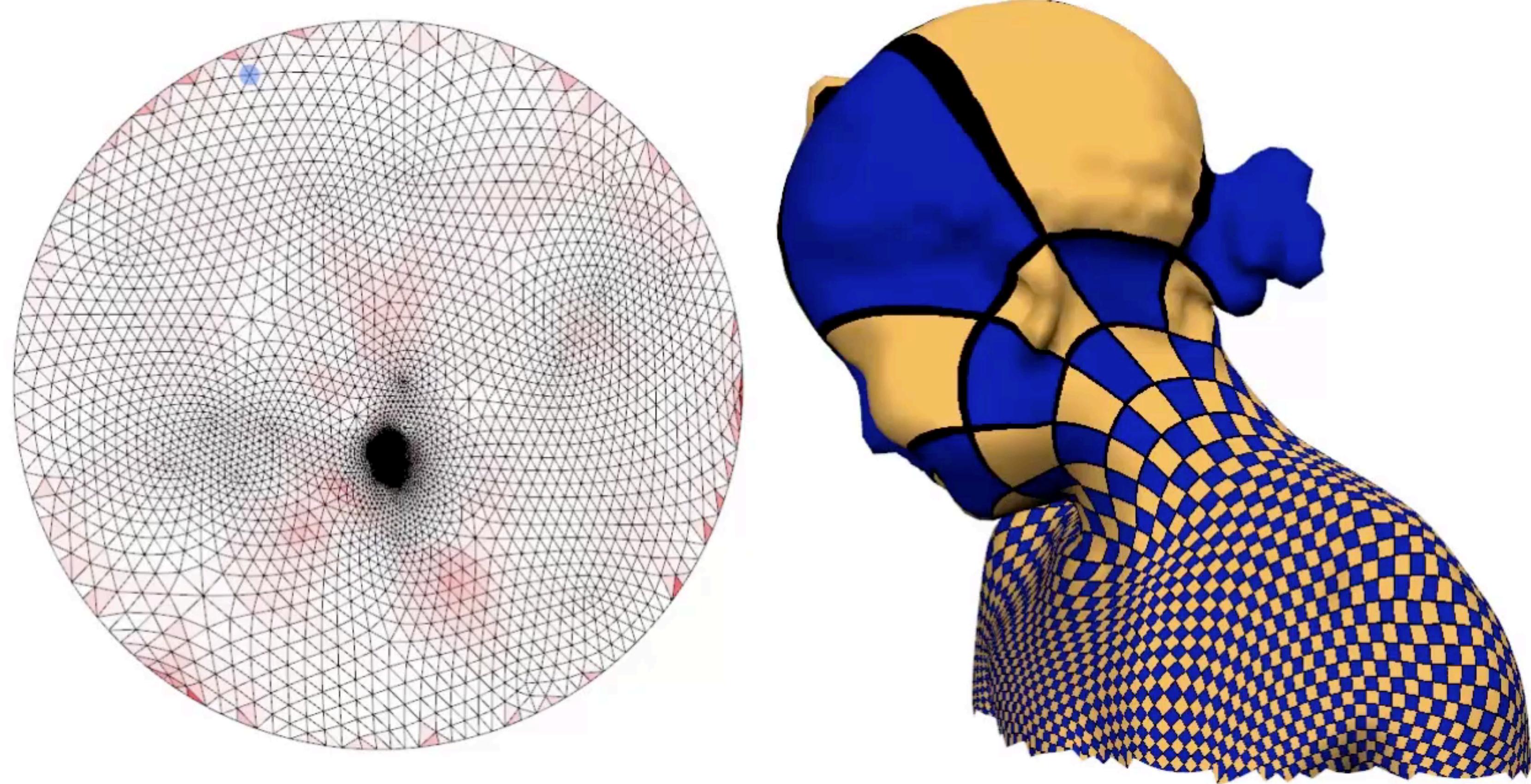


boundary

boundary

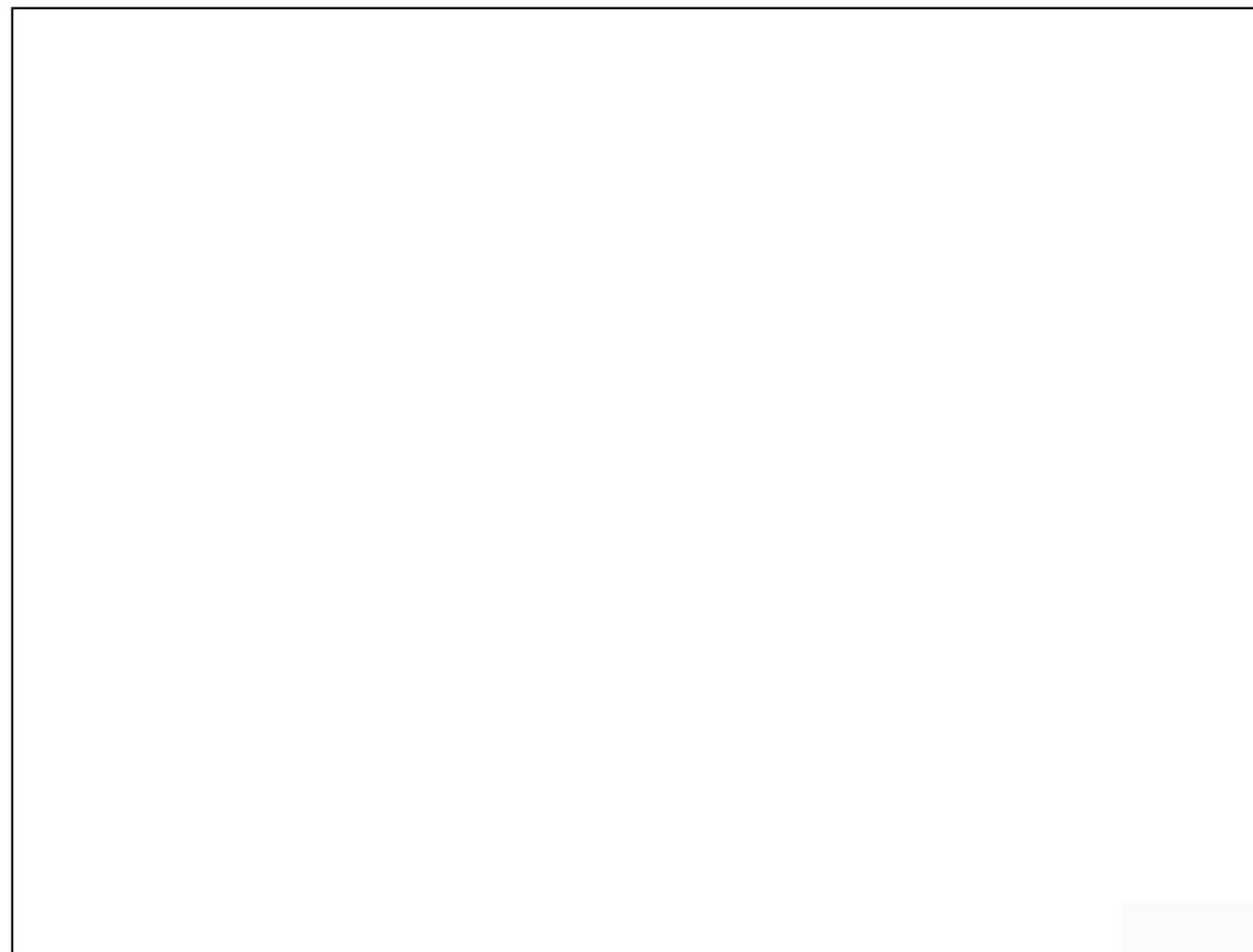
# Course Topics

- Parameterization
- 

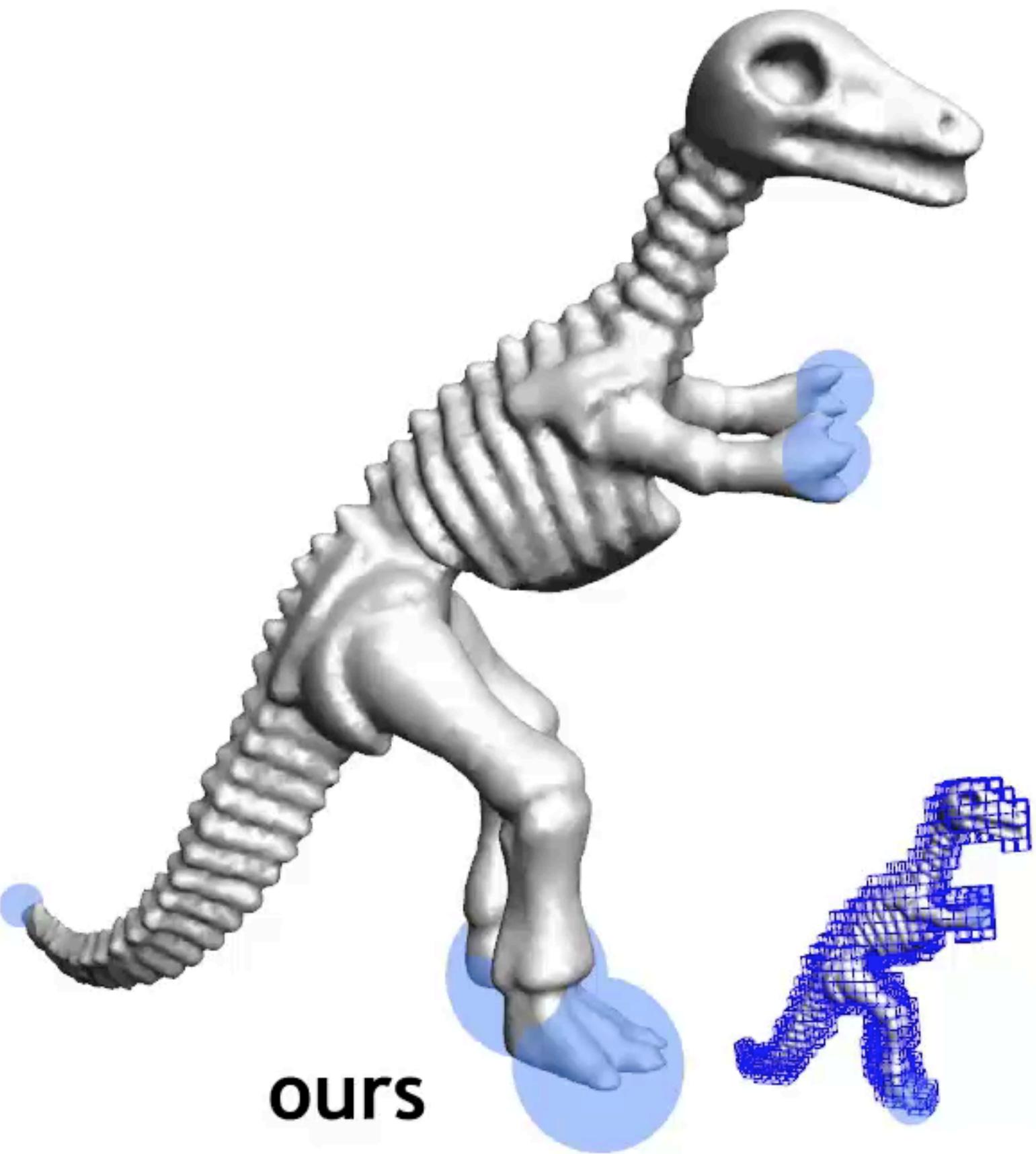


# Course Topics

- Shape creation and editing

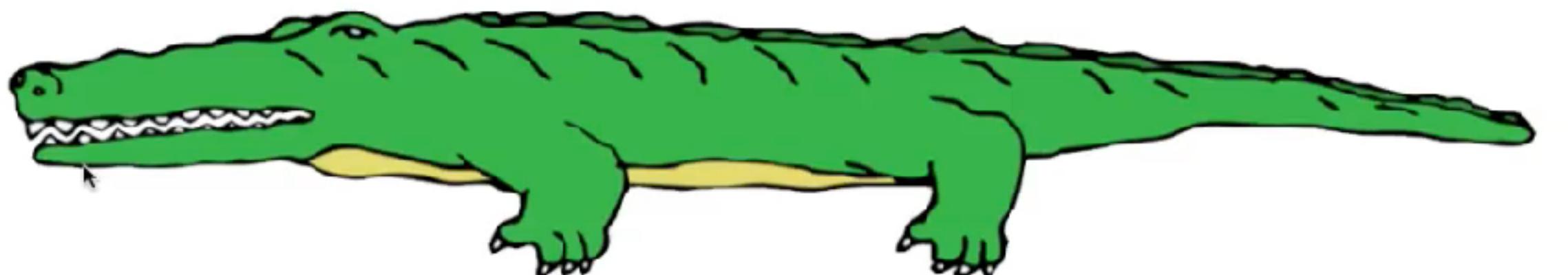


<http://youtu.be/W0XGkS7zebo>



# Course Topics

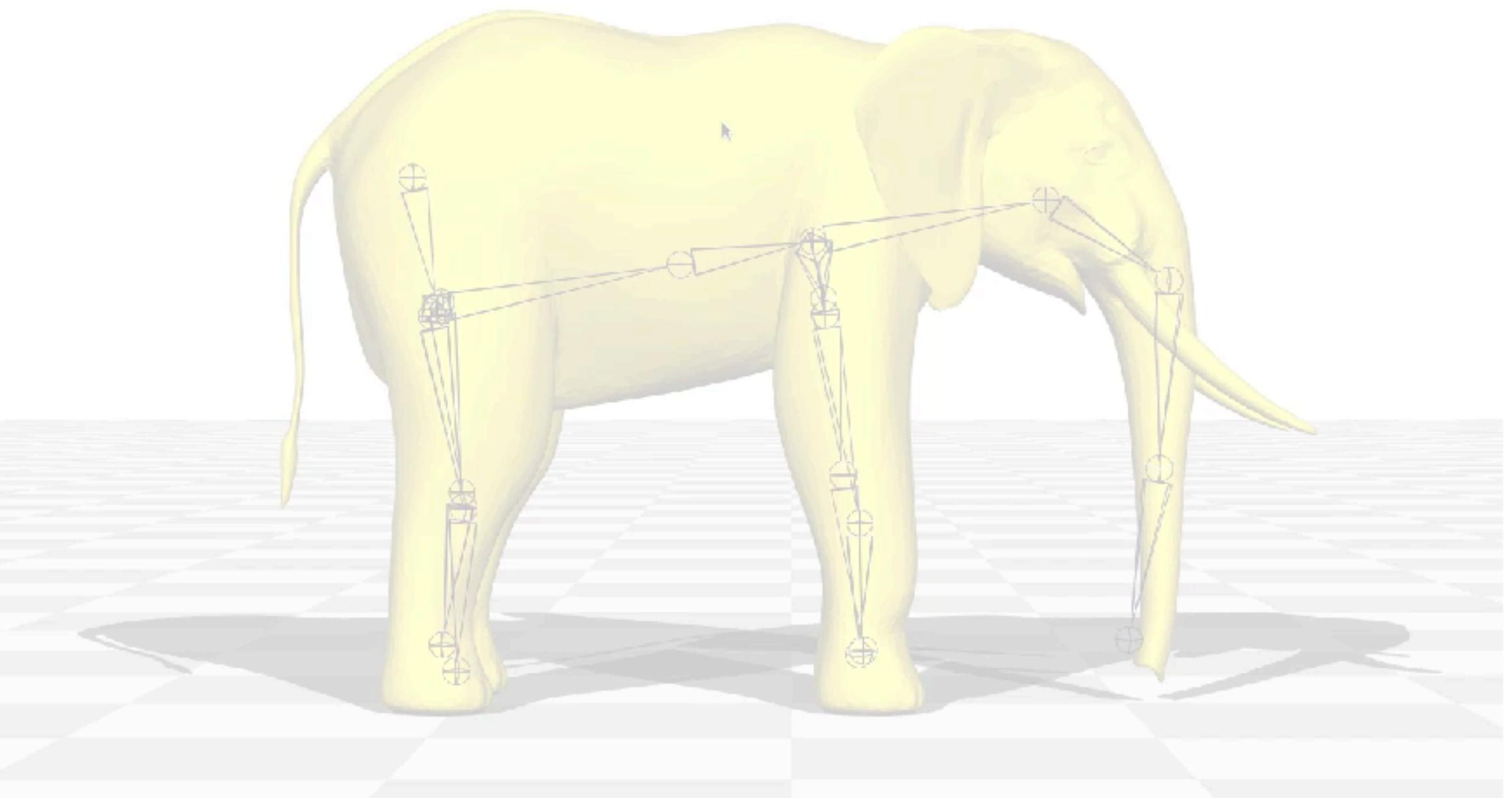
- Skinning, animation



<http://youtu.be/P9fqm8vgdB8>

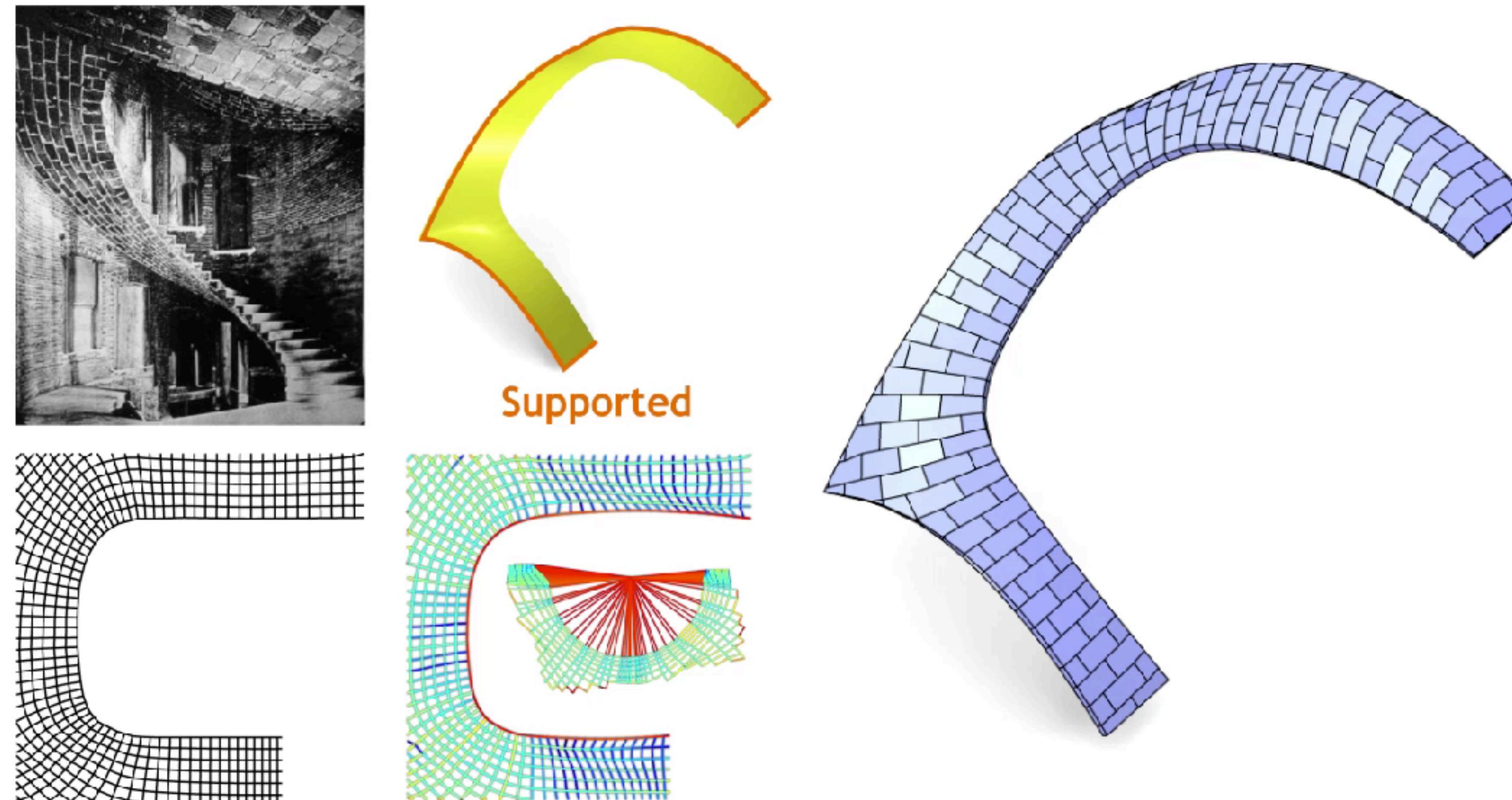


<http://youtu.be/Pjg33pH9RKO>



# Course Topics

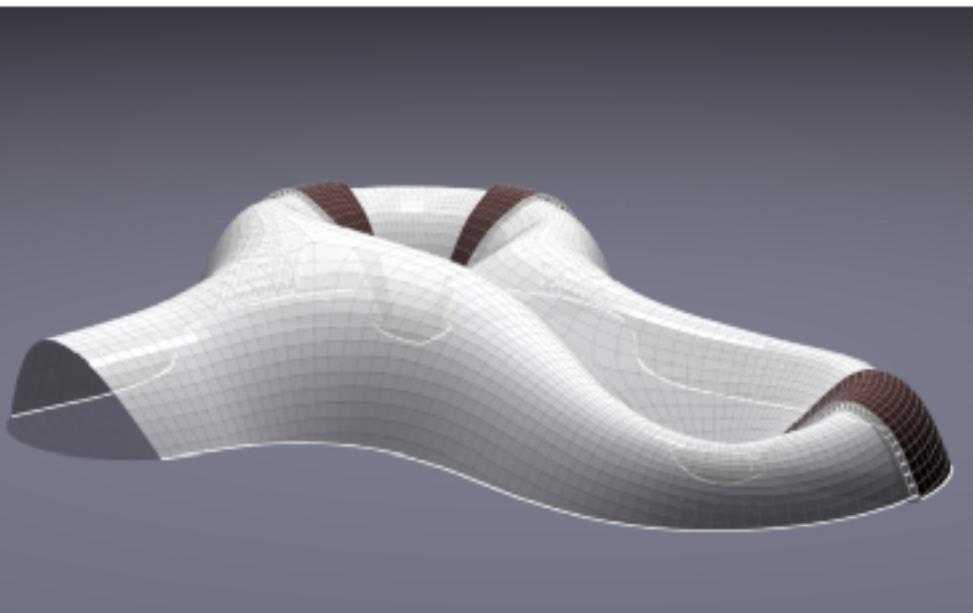
- Architectural geometry and structure-aware modeling



[Panizzo et al. SIGGRAPH 2013](#)

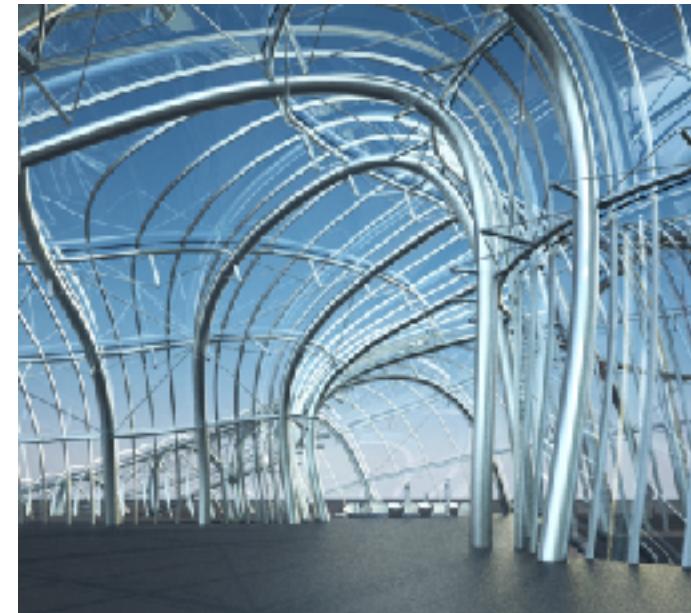
# Course Topics

- Architectural geometry and structure-aware modeling



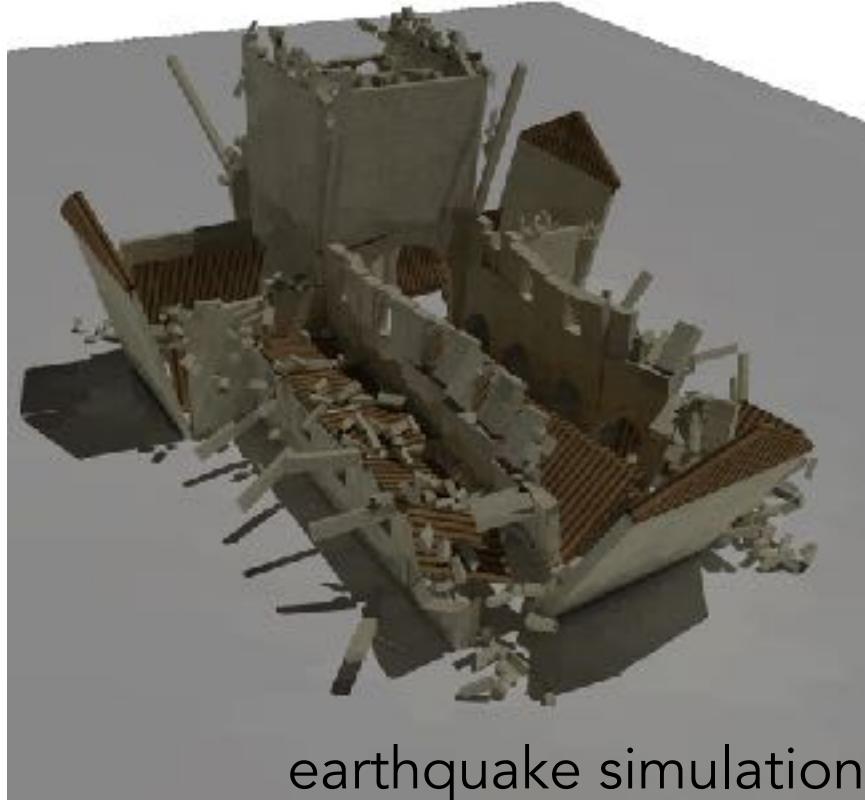
[Pottmann 08]

<http://www.dmg.tuwien.ac.at/pottmann/2008/panels08/panels.html>

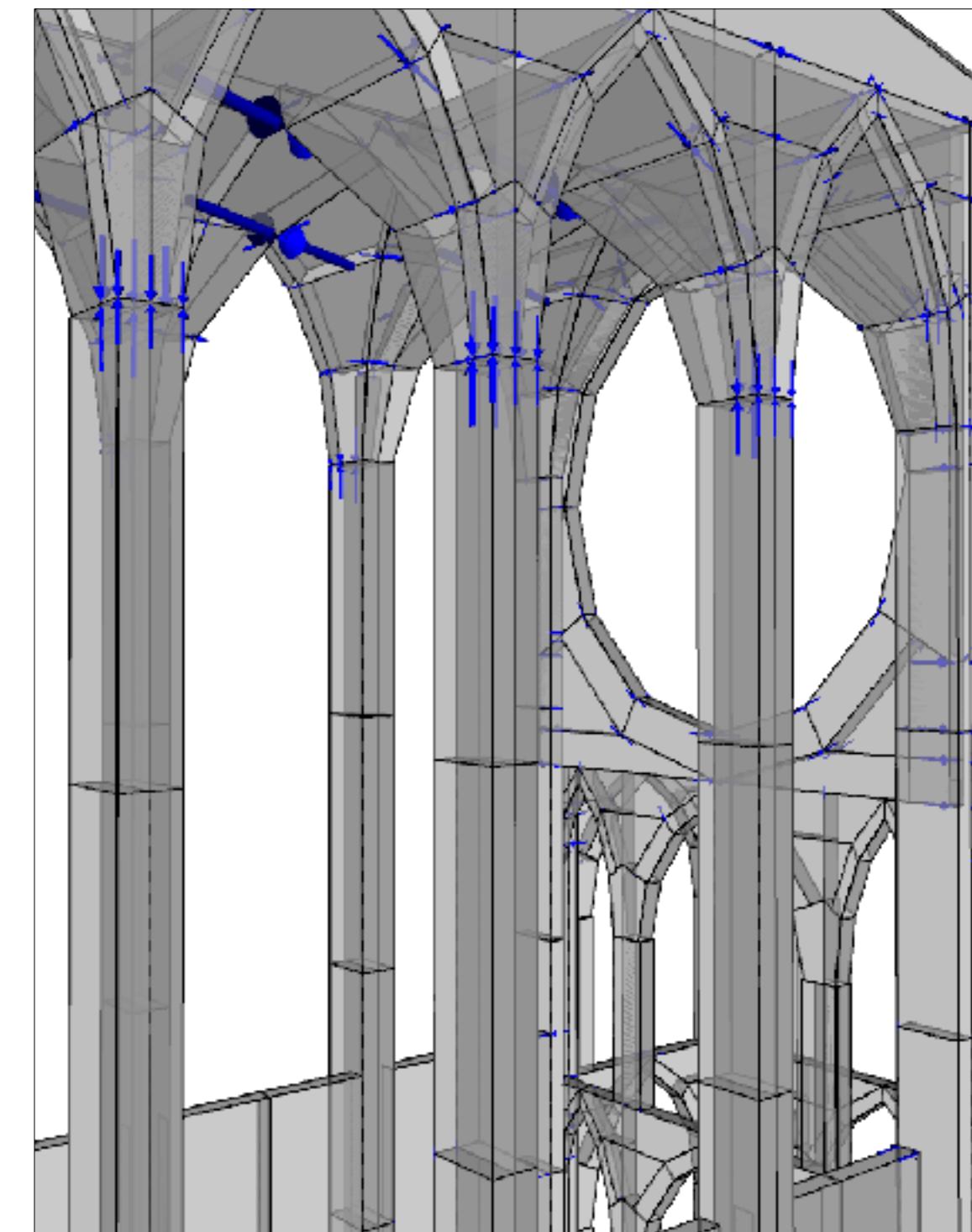


[Whiting 09]

<http://www.inf.ethz.ch/personal/whitinge/projects/siggraph09.html>

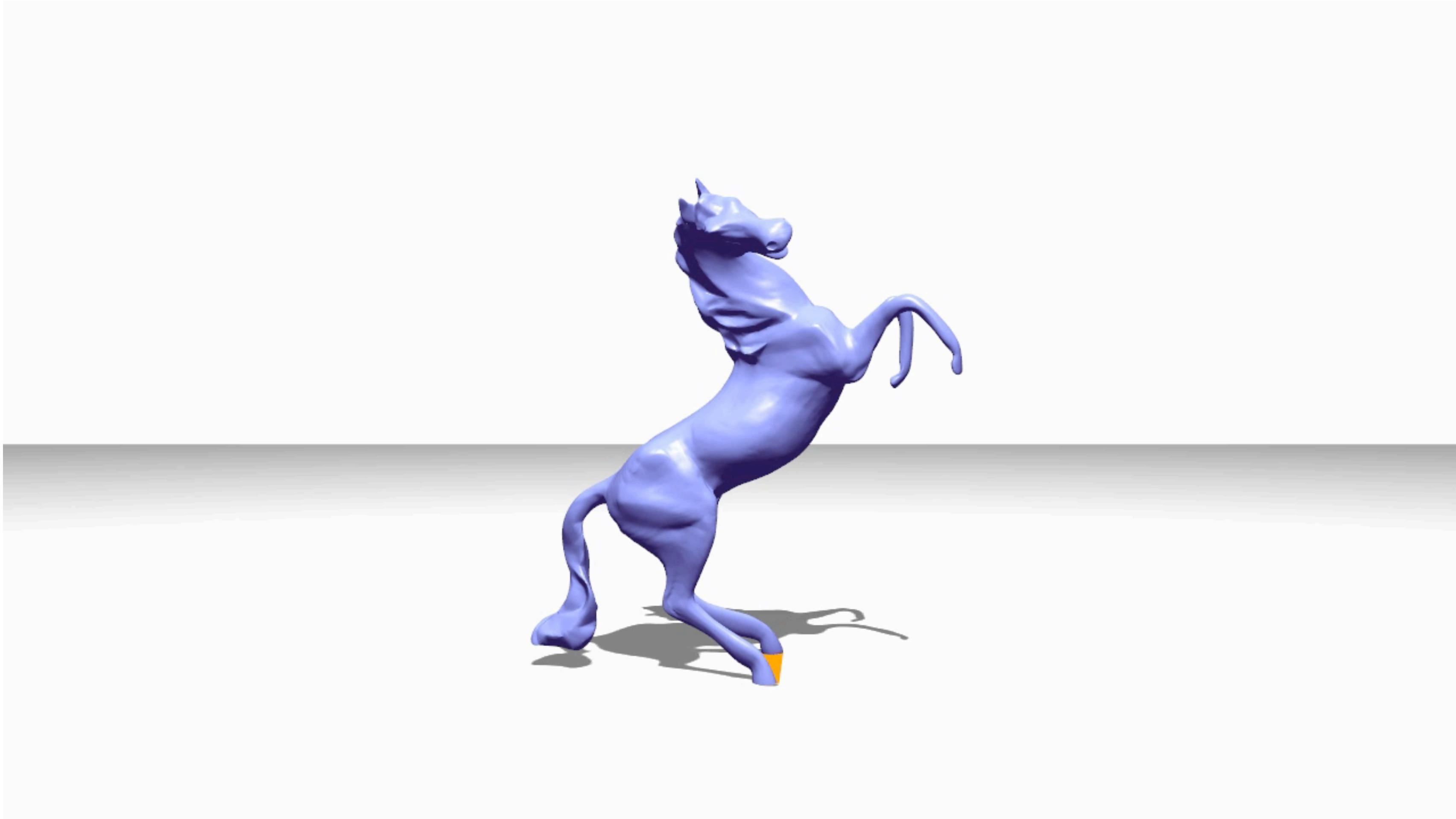


earthquake simulation



# Course Topics

- 3D fabrication-aware shape modeling



# Course Topics

- 3D fabrication-aware shape modeling



**Vacuum Forming**

# Thank you