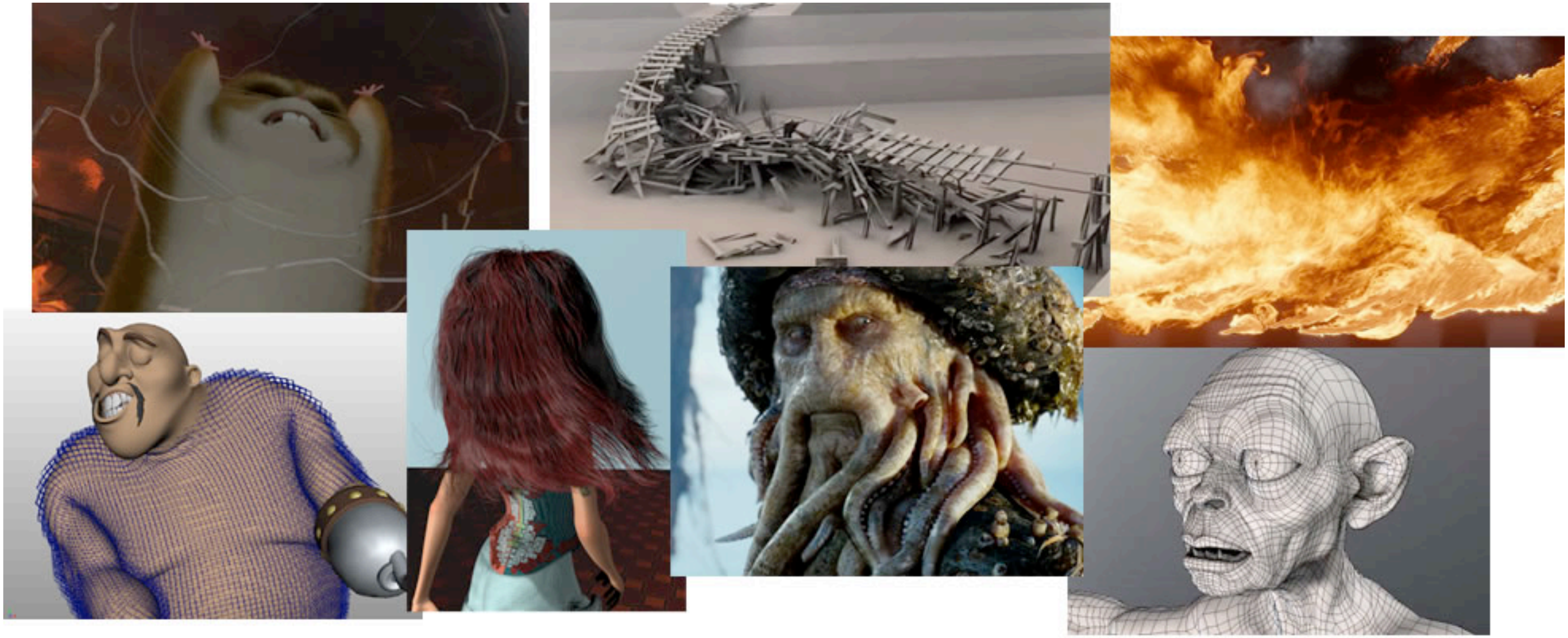


Welcome to CS838!

Topics In Computing: Introduction to Physics-based Modeling and Simulation

Fall 2012, 10:00-11:15 MWF

Today's lecture

- Explain our motivation for developing physics-based animation and modeling techniques.
- Discuss class logistics, project structure, software infrastructure and prerequisites
- Briefly introduce the various topics which will be covered in CS838

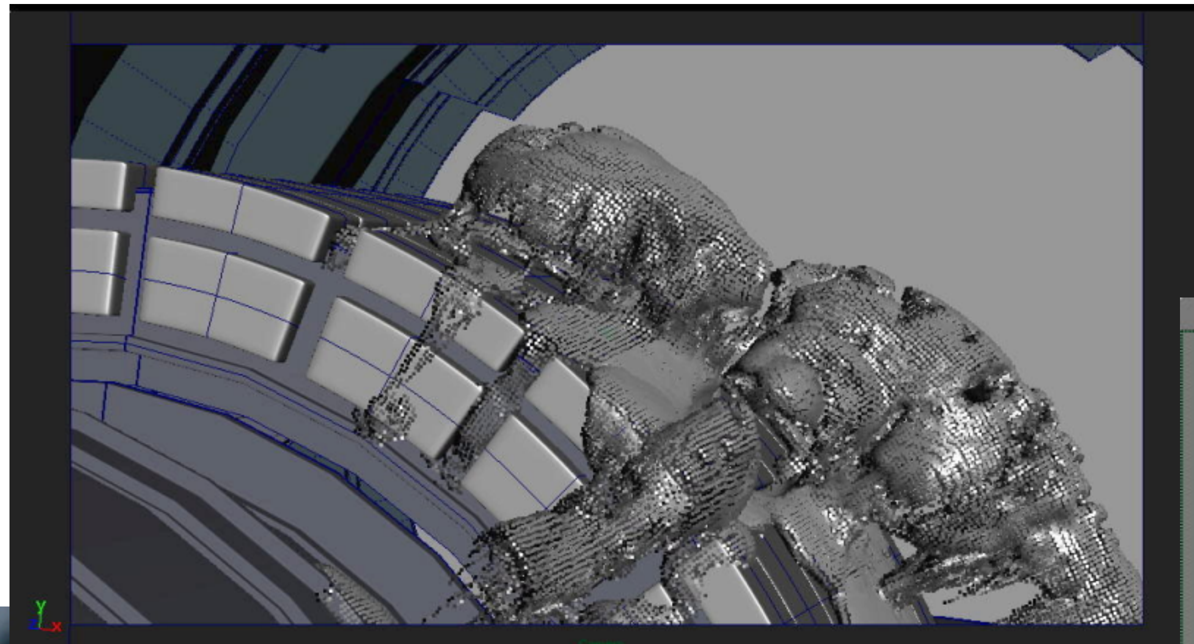
Next lecture : Get started with a discussion of geometric models and their various practical representations for the purposes of animation and simulation

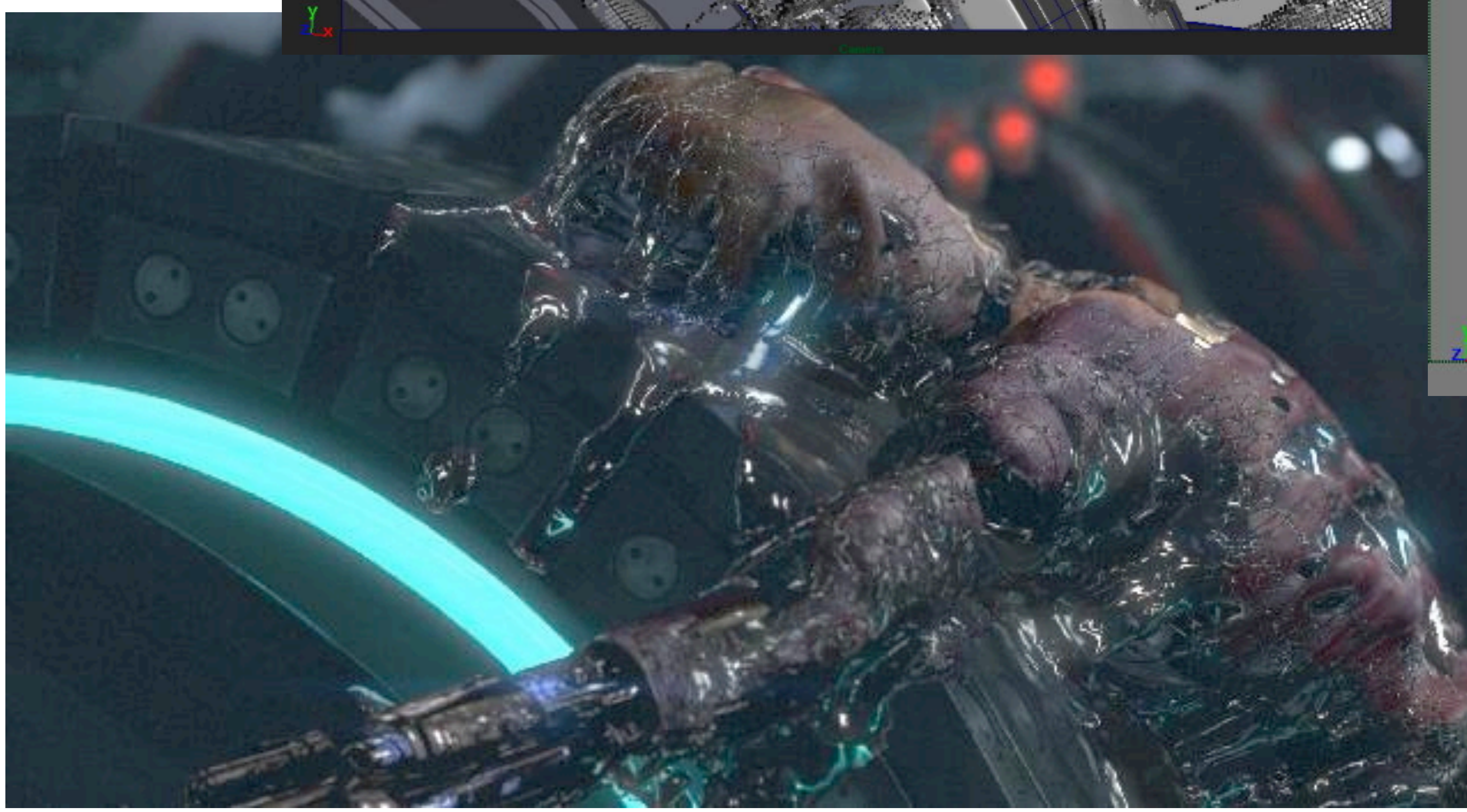
The objective of ...

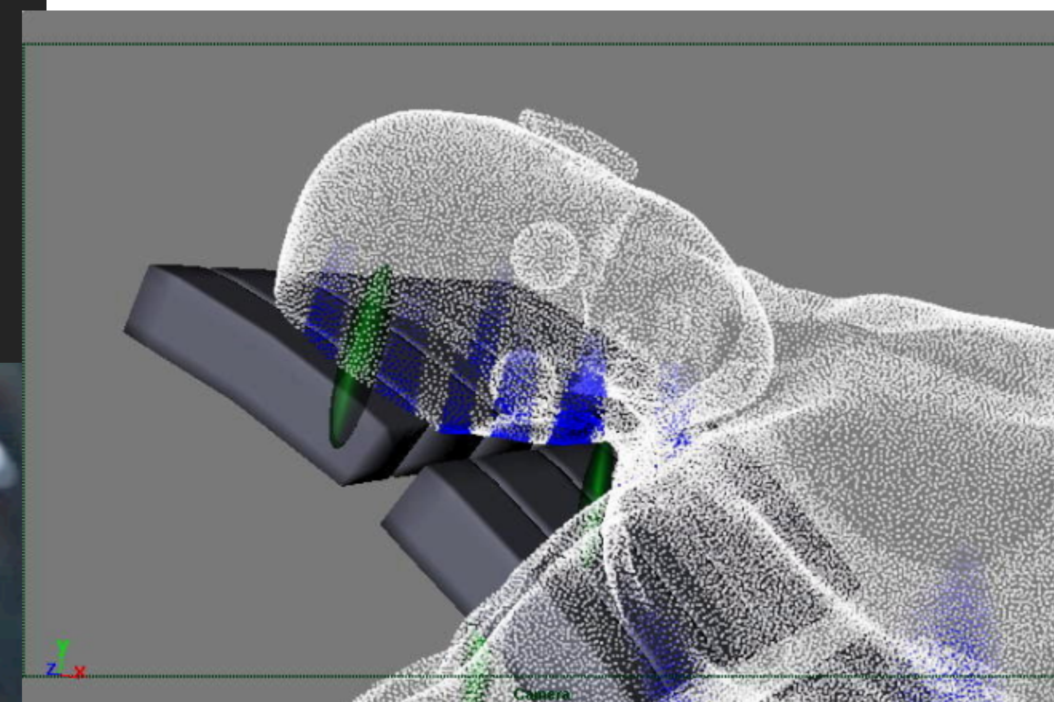
Computer Graphics : Making beautiful images

Simulation & Modeling : Create (dynamic) content which can be rendered into beautiful images

... and why use simulation?





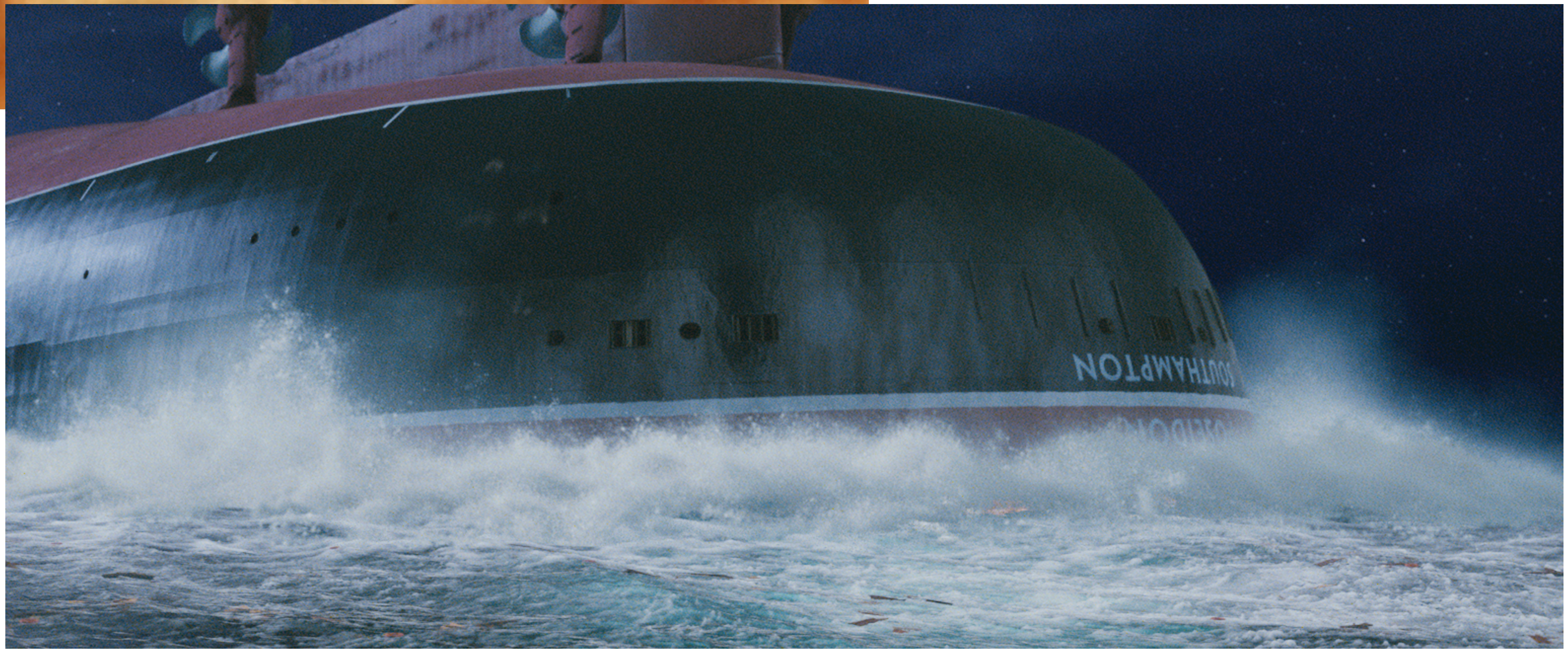


[Images © Industrial Light+Magic]

Motivation : Fictional objects, exotic materials

[Images © Industrial Light+Magic]

Motivation : Art-directable “natural” phenomena

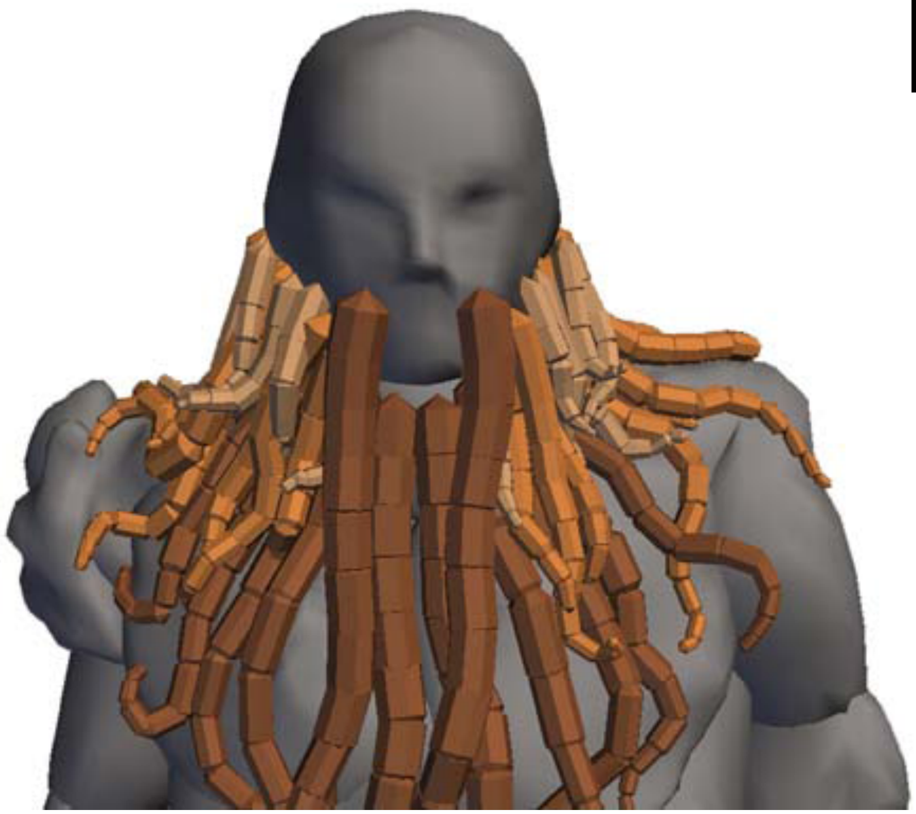


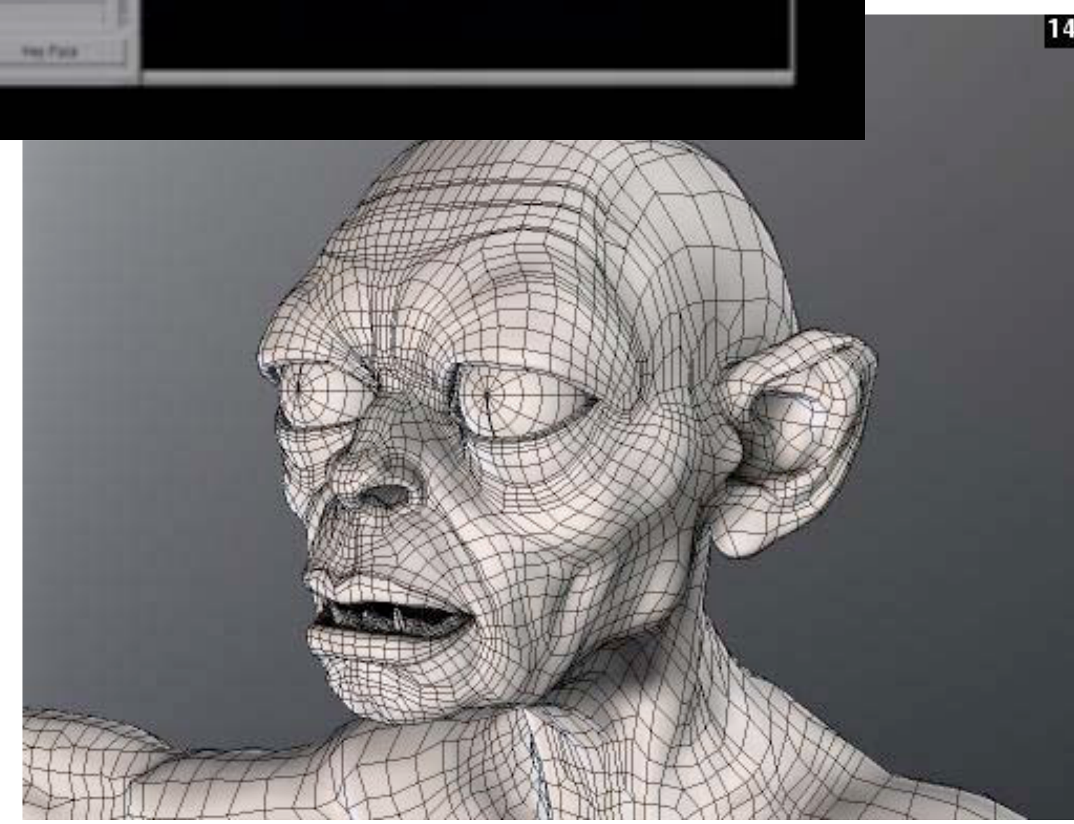
[Images © Industrial Light+Magic]

Motivation : Scenarios that cannot be replicated with scaled models

[Credits: E. Parker, J. O'Brien, Pixelux Entertainment Videos © LucasArts Ltd.]

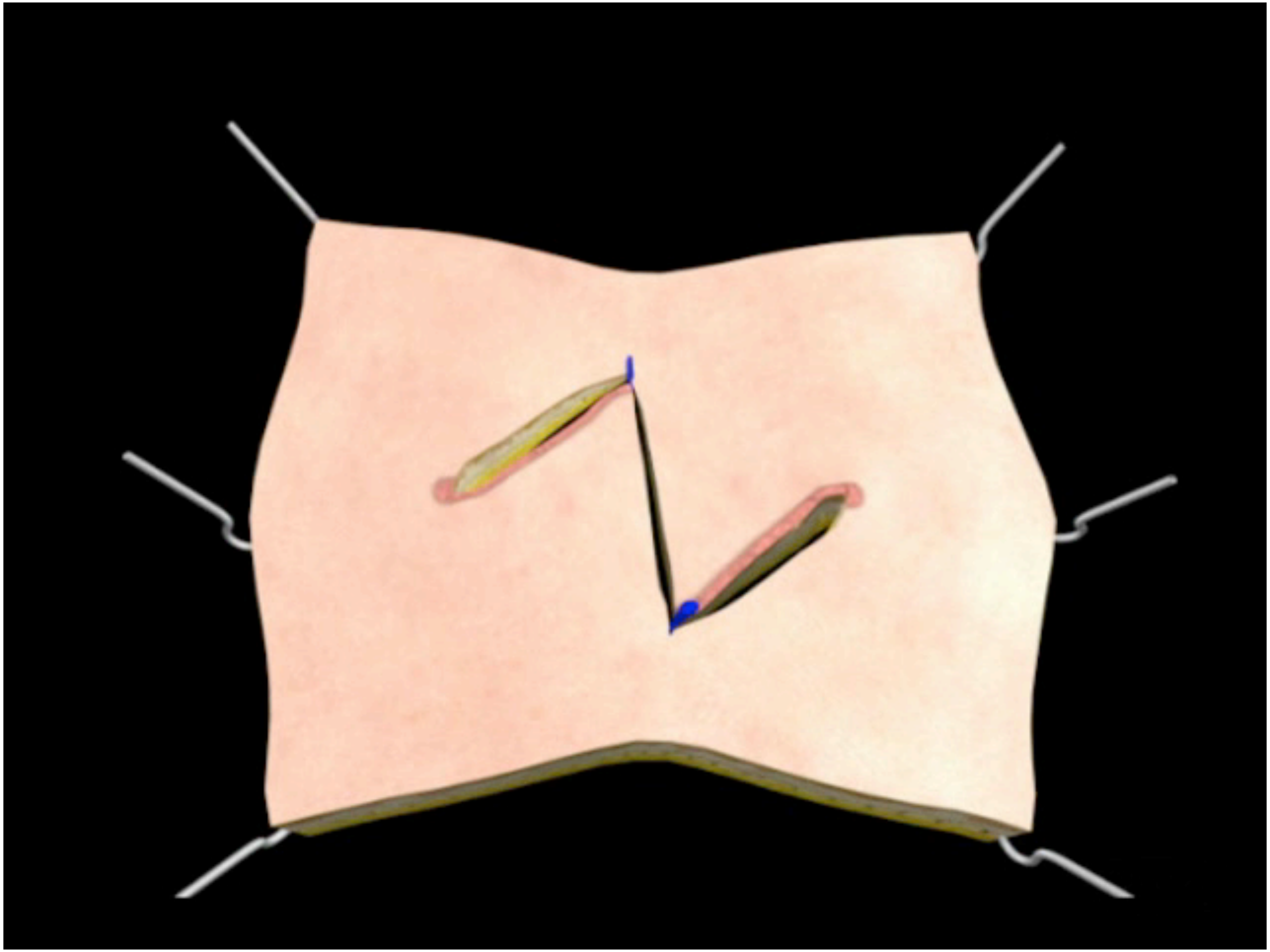
Motivation : Enhanced gameplay



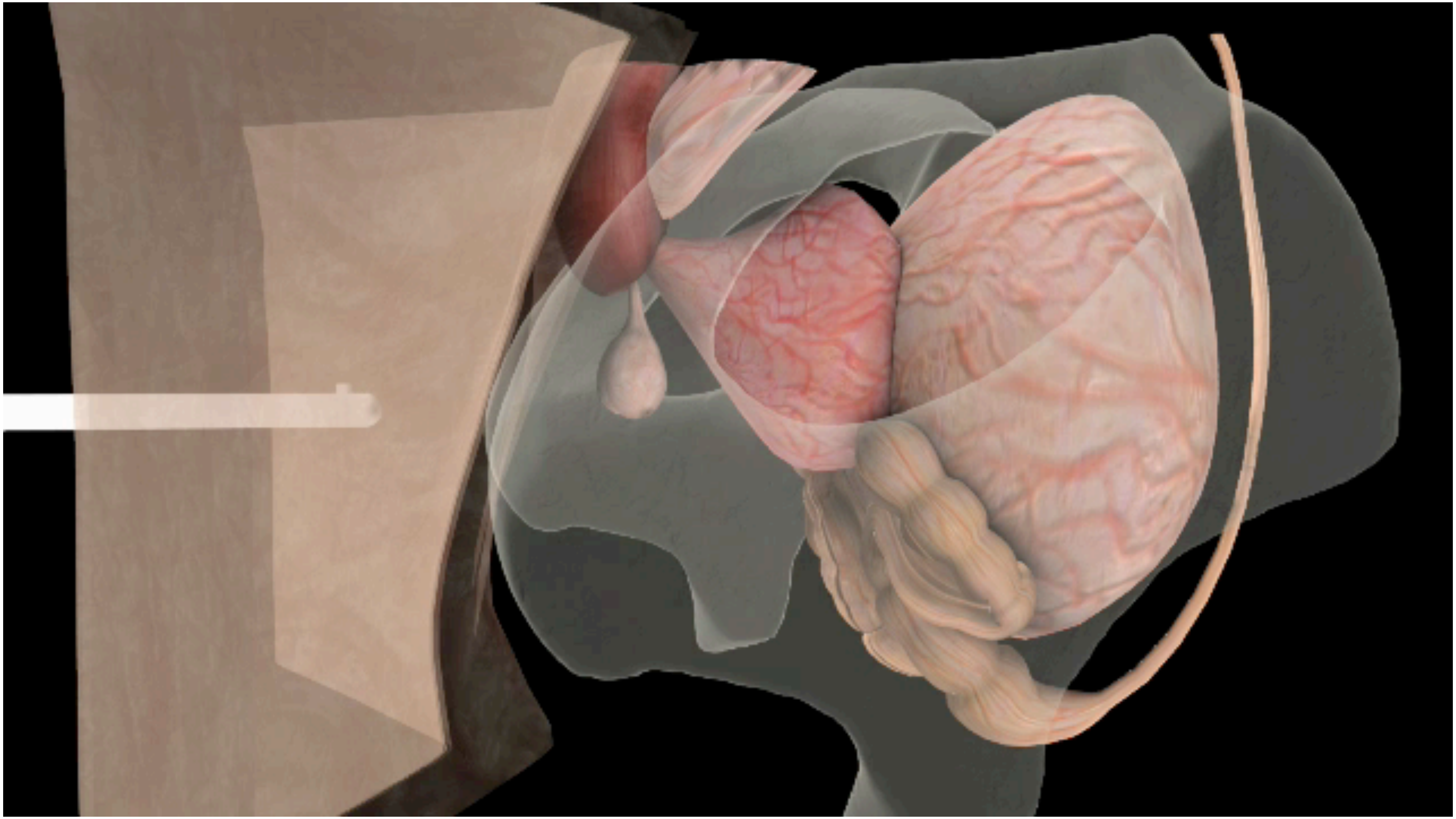


[Videos © Weta Digital, Industrial Light+Magic]

Motivation : Expressive fictional characters



Motivation : Virtual training environments

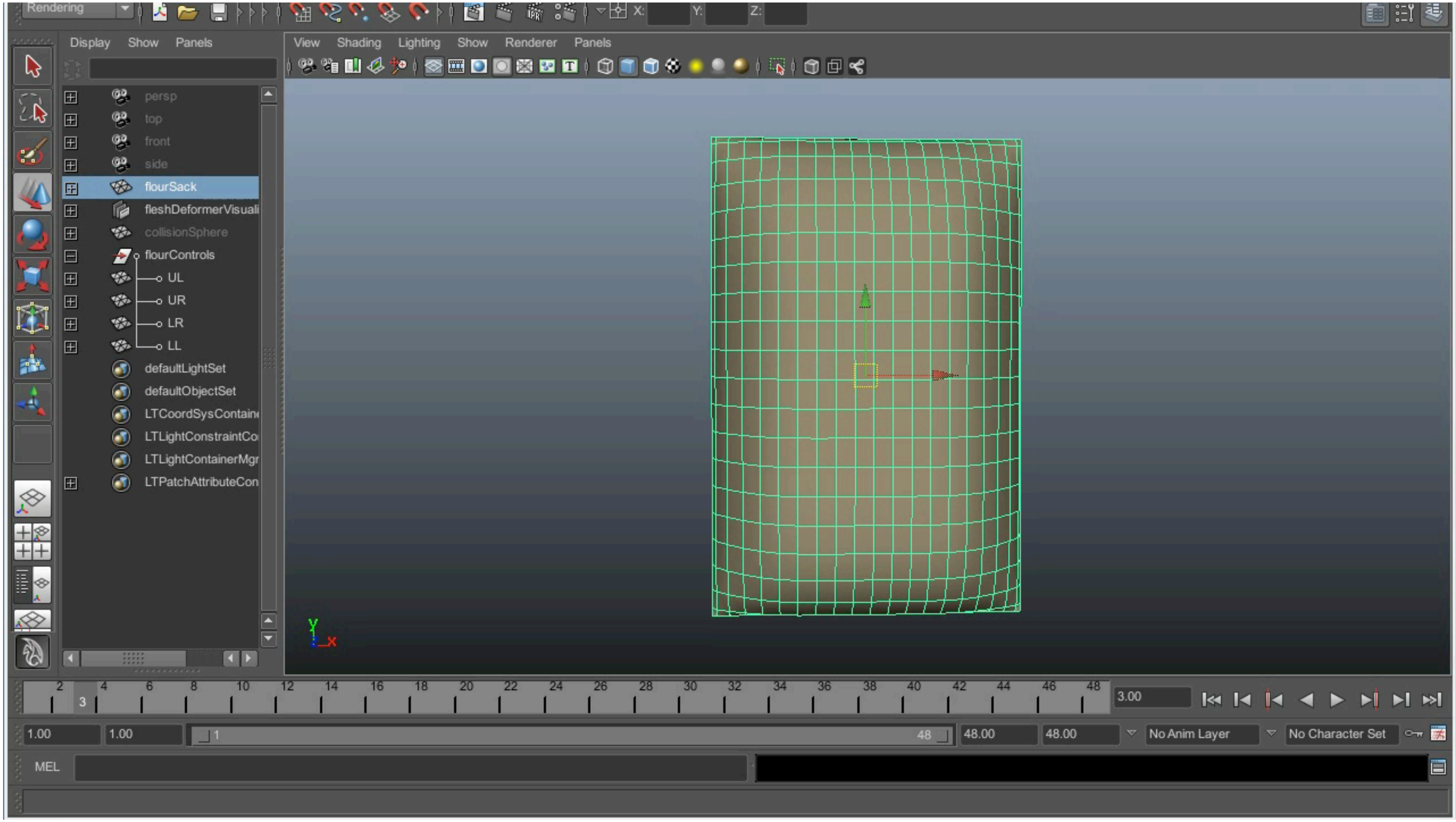


[Credits: Chentanez et al. "Interactive Simulation of Surgical Needle Insertion and Steering", SIGGRAPH2009]

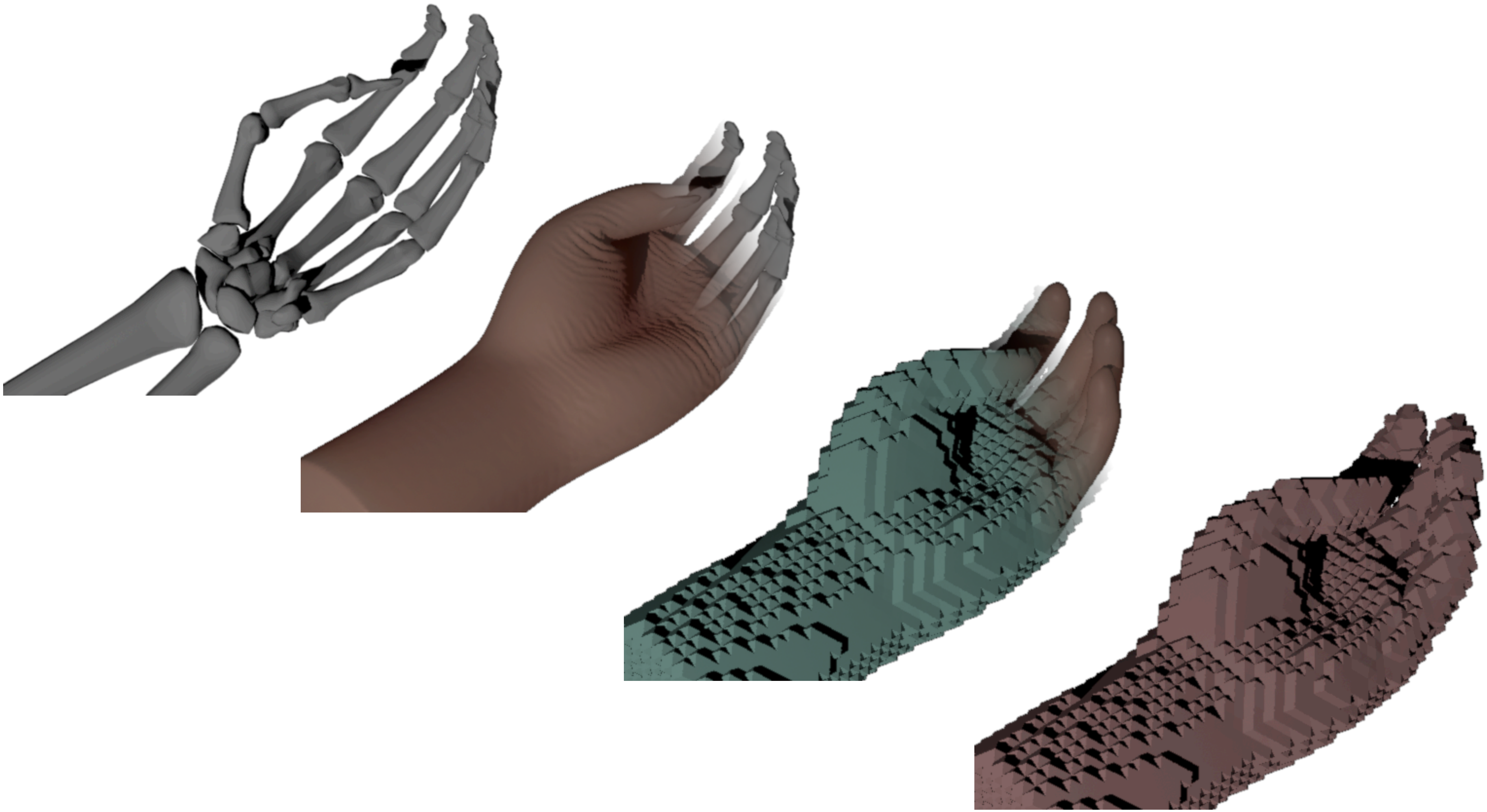
Motivation : Virtual training environments

[Credits: Chentanez et al. "Interactive Simulation of Surgical Needle Insertion and Steering", SIGGRAPH2009]

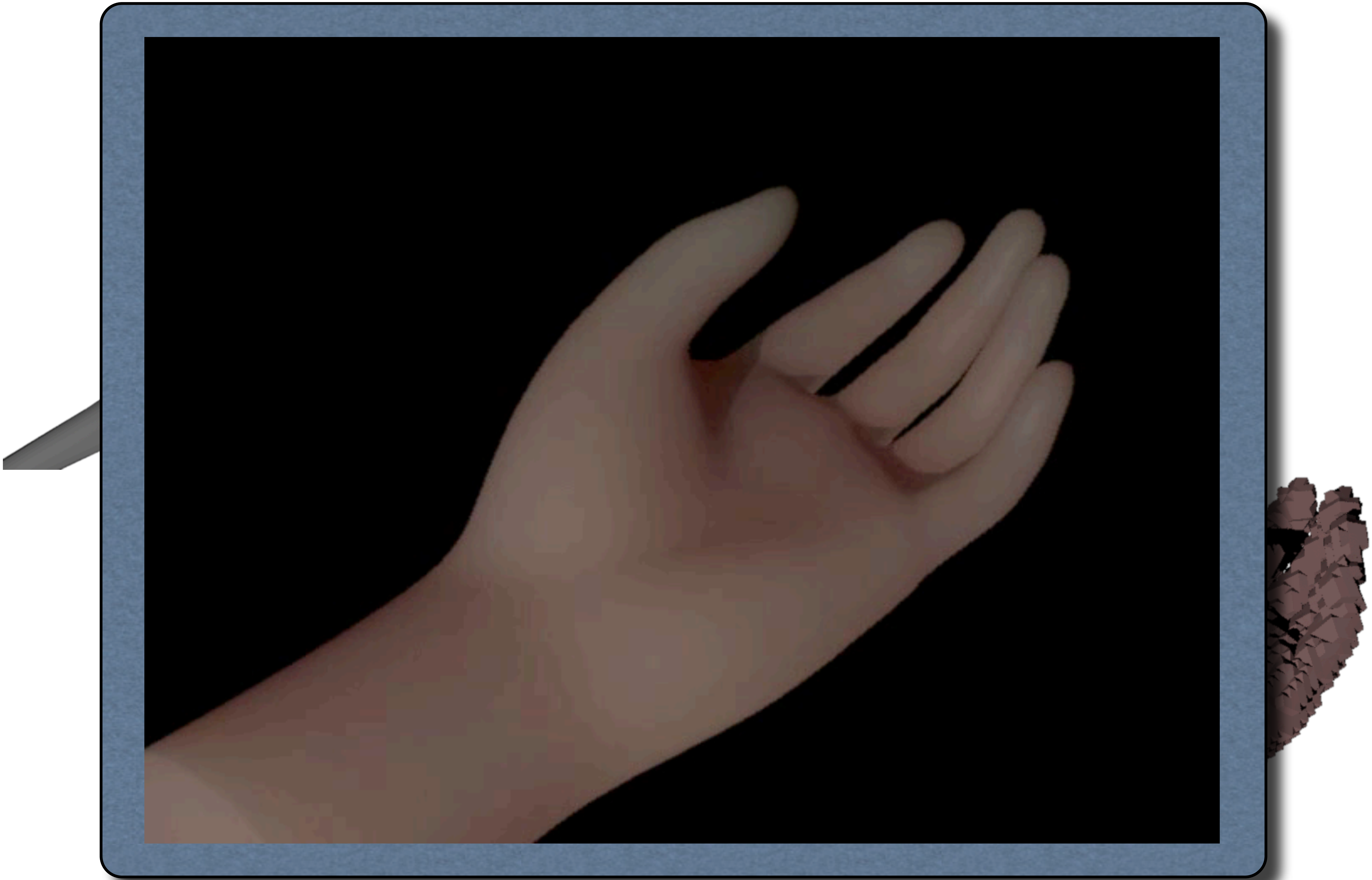
Motivation : Virtual training environments



Augmentation of animation/modeling tools



Augmentation of animation/modeling tools



Augmentation of animation/modeling tools

Augmentation of animation/modeling tools

Physics-based geometrical modeling

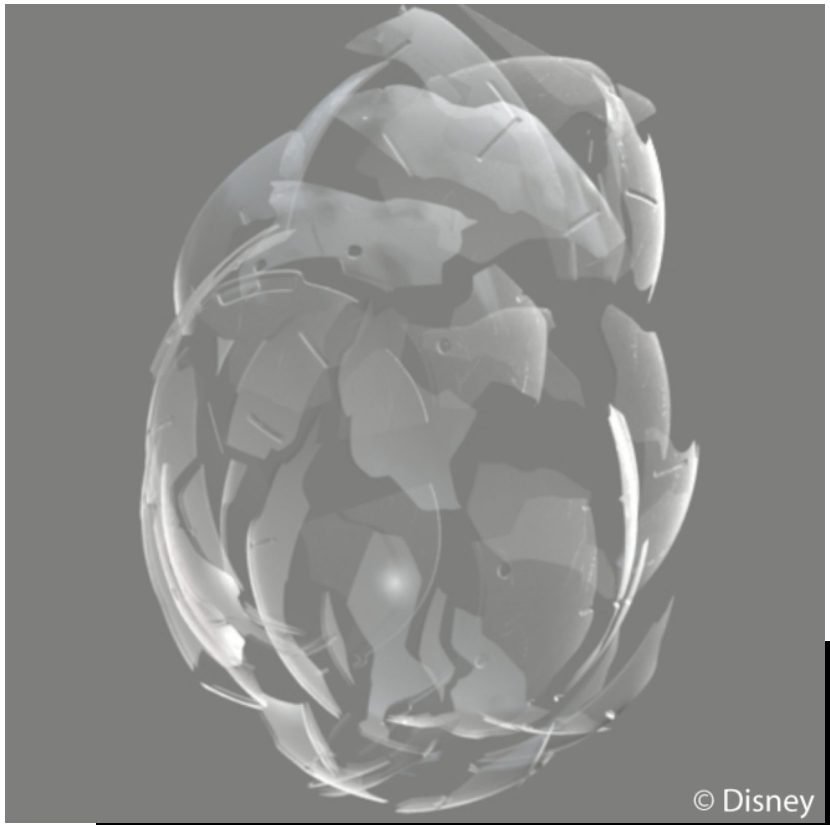
Physics-based geometrical modeling (fracture)

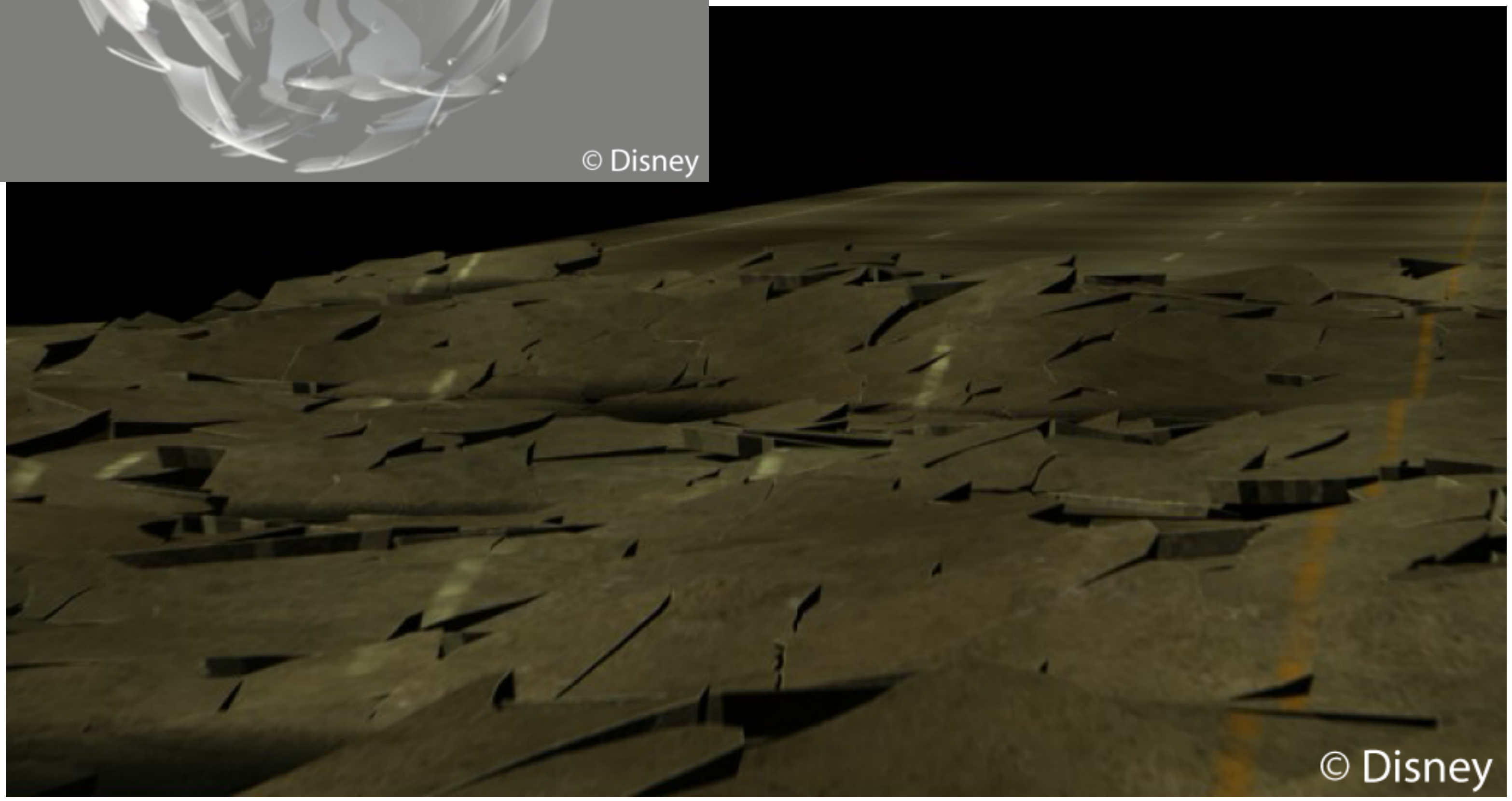
- Geometric modeling of fracture and material failure
- Crack propagation animation

[Images © Disney]

Physics-based geometrical modeling (fracture)

Physics-based geometrical modeling (fracture)

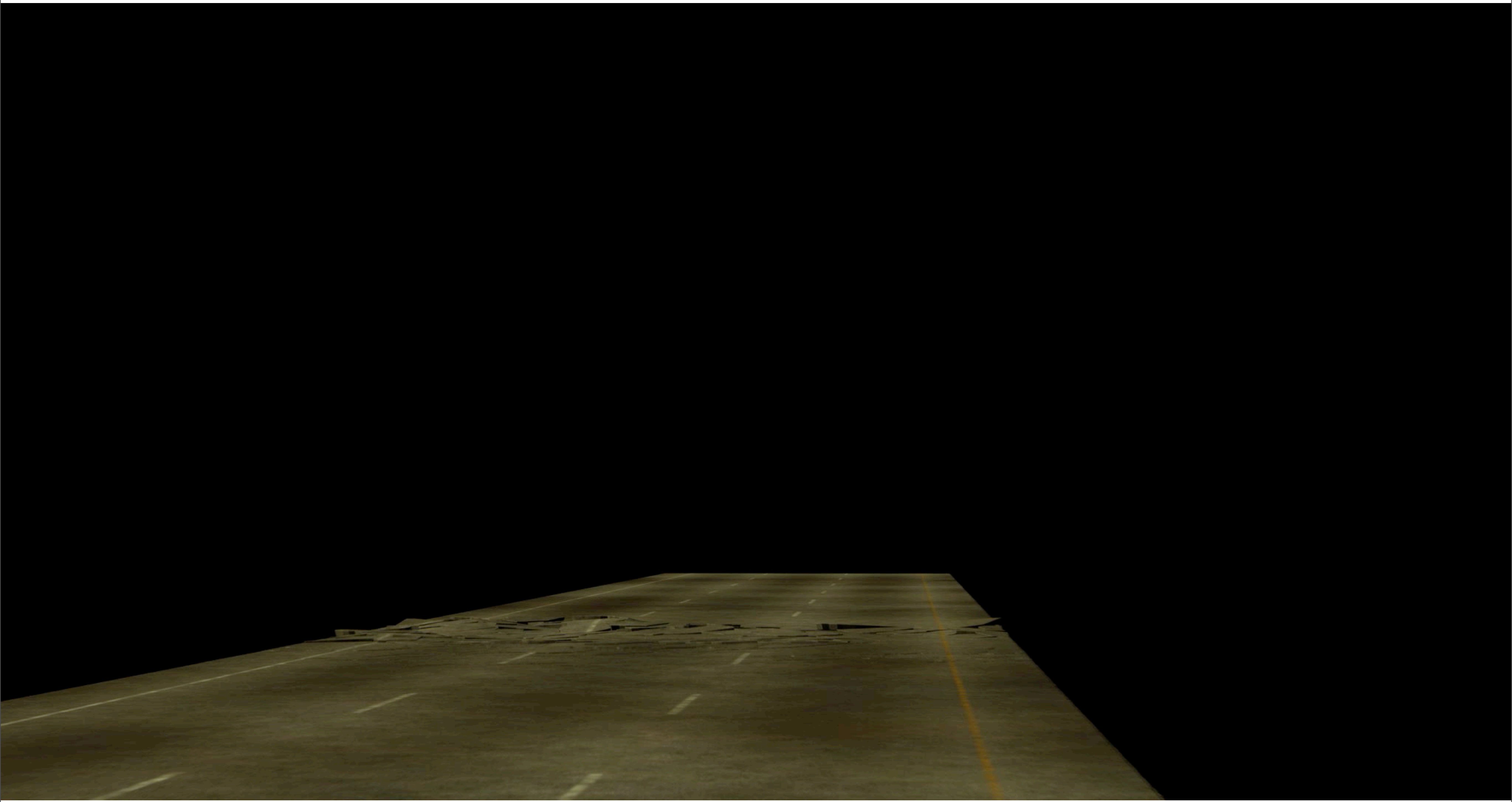




- *Geometric modeling of fracture and material failure*
- *Crack propagation animation*

[Images © Disney]

Physics-based geometrical modeling (fracture)



Physics-based geometrical modeling (fracture)

Physics-based geometrical modeling (hair)

Course information

- Location : Engineering Hall 2321
- 3 units, core credit eligible
- Class meets : MWF 10:00am - 11:15am
  - Class will meet approximately 29 out of 43 MWF days
  - Scheduled lecture cancellations will be posted on course website at least 2 weeks ahead of time
  - Lecture density will be highest in beginning of semester, and reduced towards the end.
- Course website : http://pages.cs.wisc.edu/~cs838-2
  - Check for lecture notes, assignments & reading materials

Course information

- Office Hours : CS6355, MWF 2:30-3:30pm
  - Regular office hours only on days with planned lectures
  - Other times by appointment
- Mailing list : compsci838-2-f12@lists.wisc.edu
  - Please email the instructor if you are not automatically enrolled in this list (e.g. if auditing the class)
- Email : sifakis@cs.wisc.edu
  - Please add "CS838" on subject line! (to be prioritized)
  - Email policy : Feel free to email as frequently as you need. Typically you will receive a response within 24hrs. However, be prepared to wait until next office hours (worst case) to get a comprehensive answer. If urgent, ask for an appointment.

Course information

- Objectives
  - Familiarize yourselves with the concepts and techniques of physics-based modeling and simulation
  - Obtain an understanding of available techniques, and know what papers to refer to for deeper insights and technical details
  - Acquire **hands-on** experience creating software for physics based simulation. Work with (often large) software libraries and implement reusable, efficient and well-structured code
  - Learn how to obtain information through literature search
  - Exercise your presentation skills

Evaluation

- 10% Attendance & Paper presentation
  - Every student will present one research paper, in a 20-minute in-class presentation
  - Must prepare PowerPoint/Keynote/OpenOffice slides, which will be subsequently posted on course website
  - List of papers will be posted on course website
  - Contact instructor directly for time/paper preference

Evaluation

- 40% (20% + 20%) Individual assignments
  - To be fully completed within the first half of the semester
  - Assignments will ask you to implement an established process, and familiarize yourselves with fundamental modeling and simulation techniques
  - Sample topics:
    - Author a procedural animation for a set of deformable bodies (e.g. parts of a virtual character, or various inanimate objects)
    - Implement a time integration scheme for simulating cloth or volumetric solids
    - Implement a simple collision detection and/or response algorithm

Evaluation

- 50% : Final project
  - Either individual, or in groups of 2 (your choice)
  - Deliverable at end of semester; checkpoints and milestones scheduled throughout the semester
  - The goal of the project is to extend somewhat beyond what is a standard practice, and ideally experiment with an original idea.
  - Sample topics:
    - Accelerate the performance of a specific simulation, by using a more advanced or specialized algorithmic technique.
    - Model a nonstandard trait for a virtual material (e.g. viscoelasticity, stiction, surface tension)
    - Apply a known technique to an original application

Prerequisites

No formal prerequisites, but ...

- Course will have a heavy mathematical component:
  - A basic knowledge of calculus and linear algebra is assumed (more details in next slides)
  - Specialized mathematical topics will be adequately reviewed in class, and supplemented with notes and/or literature references.
- Course will be implementation-intensive:
  - You should be comfortable with C++ programming
  - You will need to study and use third-party libraries
  - Experience with C++ templates will be useful (but not essential)
- Experience with parallel programming (threads, MPI, OpenMP, CUDA) is not required, but if you have it you are encouraged to pick projects that leverage this exposure.

Prerequisites (mathematical background)

You are expected to have working knowledge of:

- Vector valued functions, and functions of several variables
- Computing derivatives (and partial derivatives) of moderately complex functions, e.g.

$$\text{Find} \;\; \partial f/\partial x, \; \partial f/\partial y \;\; \text{when} \;\; f(x,y) = \frac{x}{\sqrt{x^2+y^2}}$$

Prerequisites (mathematical background)

You are expected to have working knowledge of:

- Vector valued functions, and functions of several variables
- Computing derivatives (and partial derivatives) of moderately complex functions, e.g.

$$\text{Find} \quad \partial f/\partial x,\ \partial f/\partial y \quad \text{when} \quad f(x,y) = \frac{x}{\sqrt{x^2+y^2}}$$

But, we will review in class topics such as:

- Numerical approximations of derivatives and integrals
- Computing certain types of exotic derivatives, e.g.

$$\text{Find} \quad f'(t) \ \text{when} \quad f(t) = \det(\mathbf{A} + t\mathbf{B})$$

$$[\text{Answer: } f'(t) = \det(\mathbf{A} + t\mathbf{B}) \cdot \text{tr}\{(\mathbf{A} + t\mathbf{B})^{-1}\mathbf{B}\}]$$

Prerequisites (mathematical background)

You are expected to have working knowledge of:

- What an n-dimensional vector and an nxn matrix is
- The concept of a trace and a determinant of a matrix
- The concept of eigenvalues and eigenvectors of a matrix
- Simple methods for solving linear systems of equations $\mathbf{A}\mathbf{x} = \mathbf{B}$ (for example, Gauss elimination)

Prerequisites (mathematical background)

You are expected to have working knowledge of:

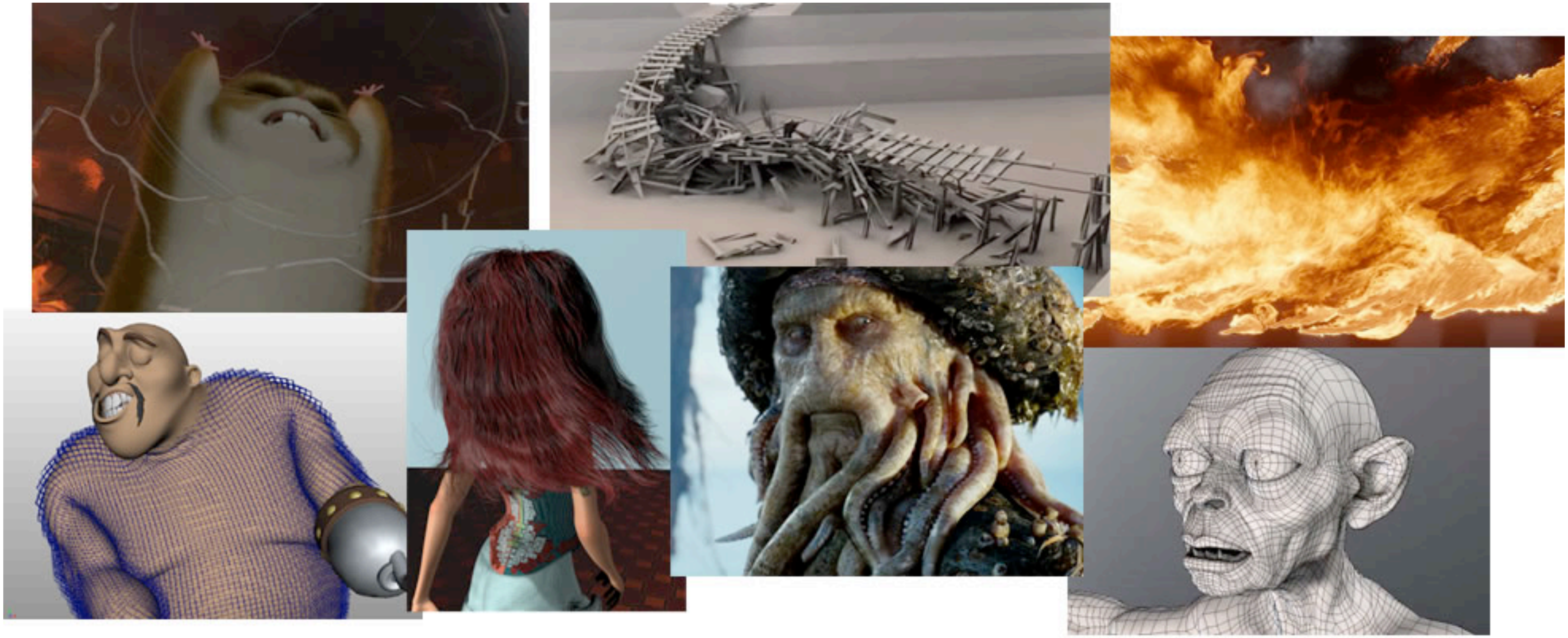
- What an n-dimensional vector and an nxn matrix is
- The concept of a trace and a determinant of a matrix
- The concept of eigenvalues and eigenvectors of a matrix
- Simple methods for solving linear systems of equations $\mathbf{Ax} = \mathbf{B}$ (for example, Gauss elimination)

But, we will review in class topics such as:

- How to solve linear systems of equations very efficiently (e.g. often at cost O(n) or O(nlogn), when considering n equations and n unknowns)
- Generalizations of matrices (2-dimensional arrays) to tensors (higher dimensional arrays)

Software infrastructure

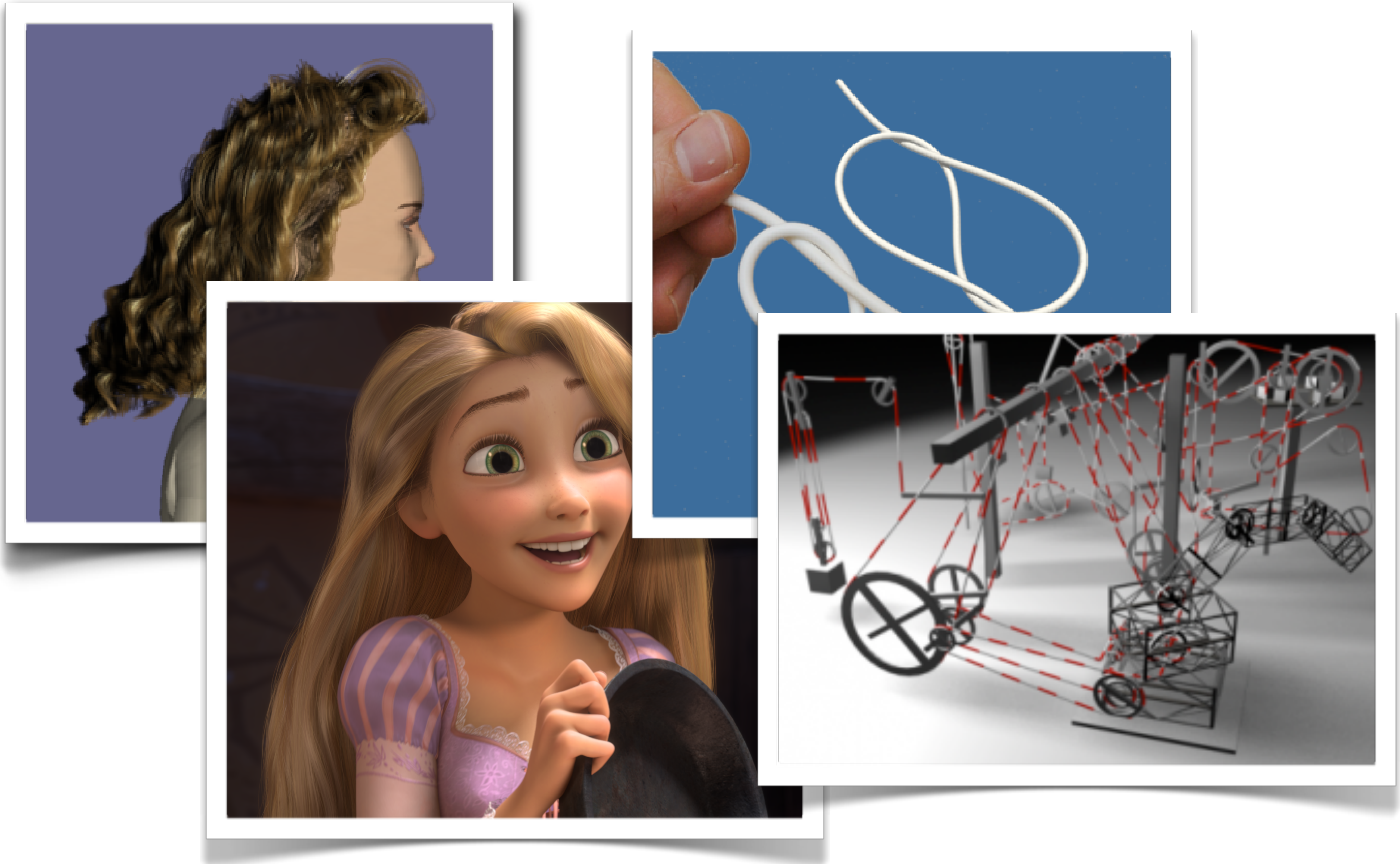
- Encouraged (but not strictly required) use of the PhysBAM open-source physics-based modeling library
  - Cross-platform, but predominantly used under Linux
  - Used at Walt Disney Animation studios, Pixar, Intel Corporation, Industrial Light+Magic
  - Includes visual debugging tools, and rendering support
  - Full suite (not the open-source version) includes both modeling infrastructure and dynamic simulation tools
    - The open source version includes all the modeling and visualization tools, as well as numerical tools and data structures
    - No dynamics; we shall implement those from scratch



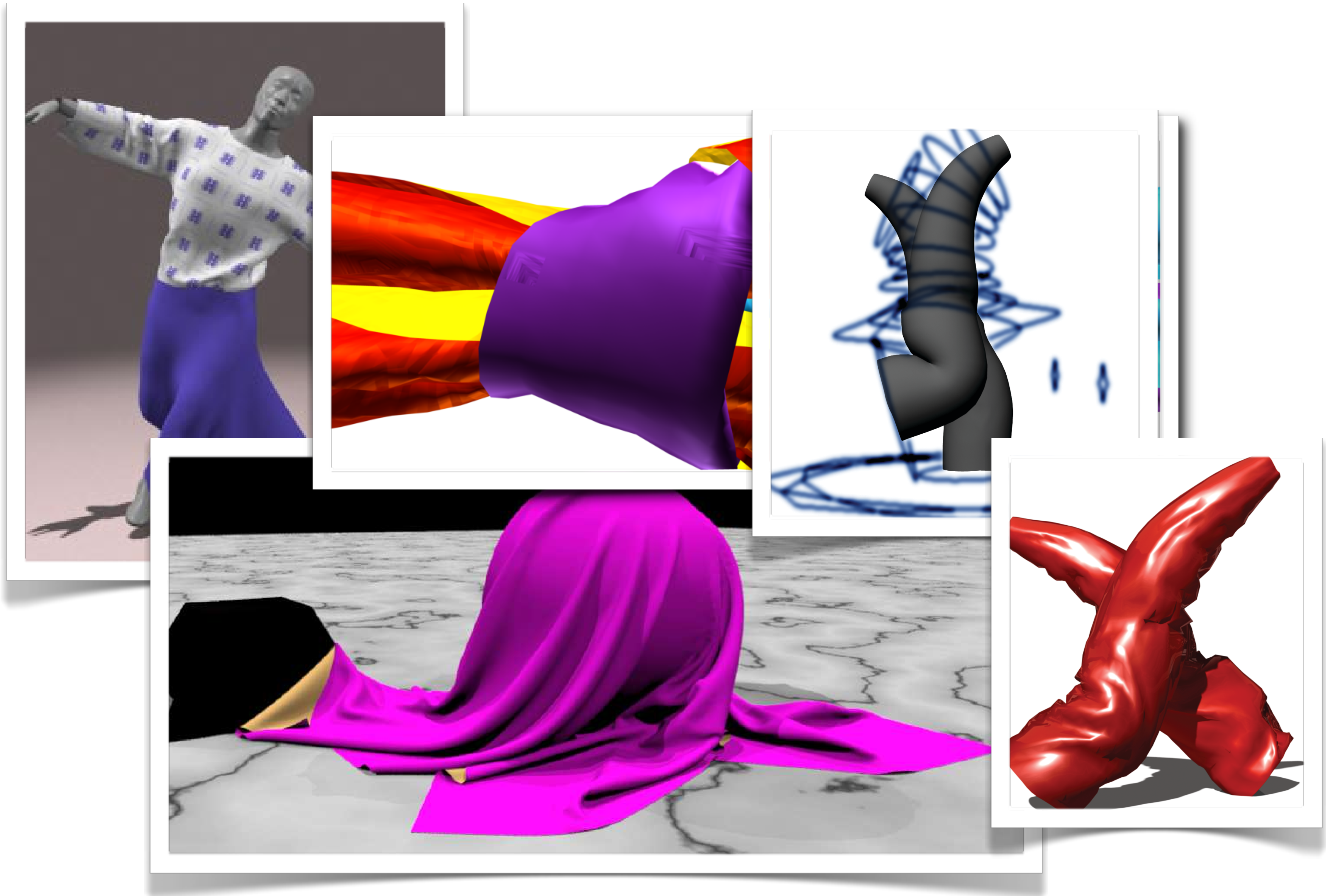
Welcome to CS838!
Topics In Computing: Introduction to Physics-based Modeling and Simulation

Fall 2012, 10:00-11:15 MWF

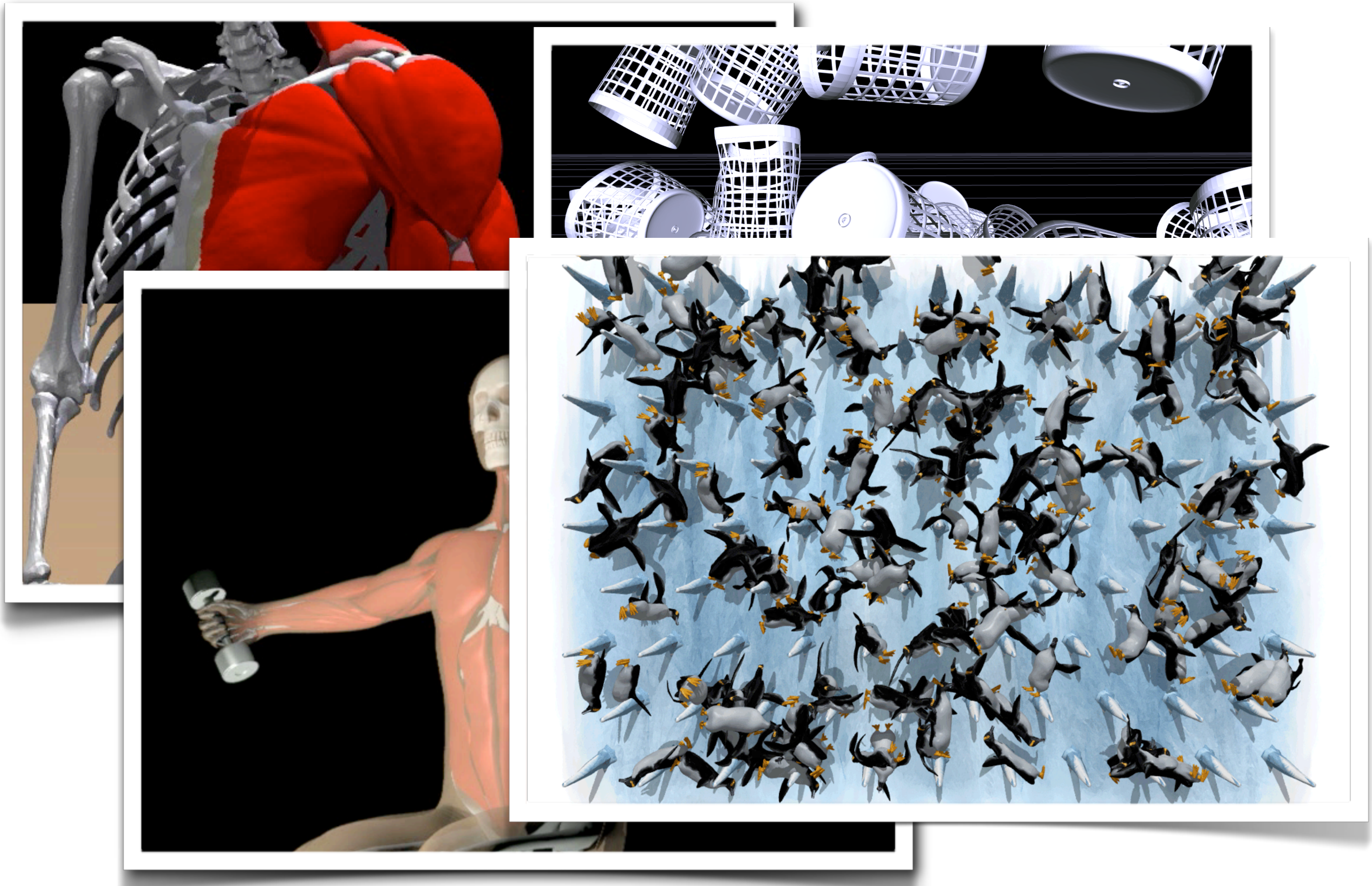
Topics : 1D elasticity



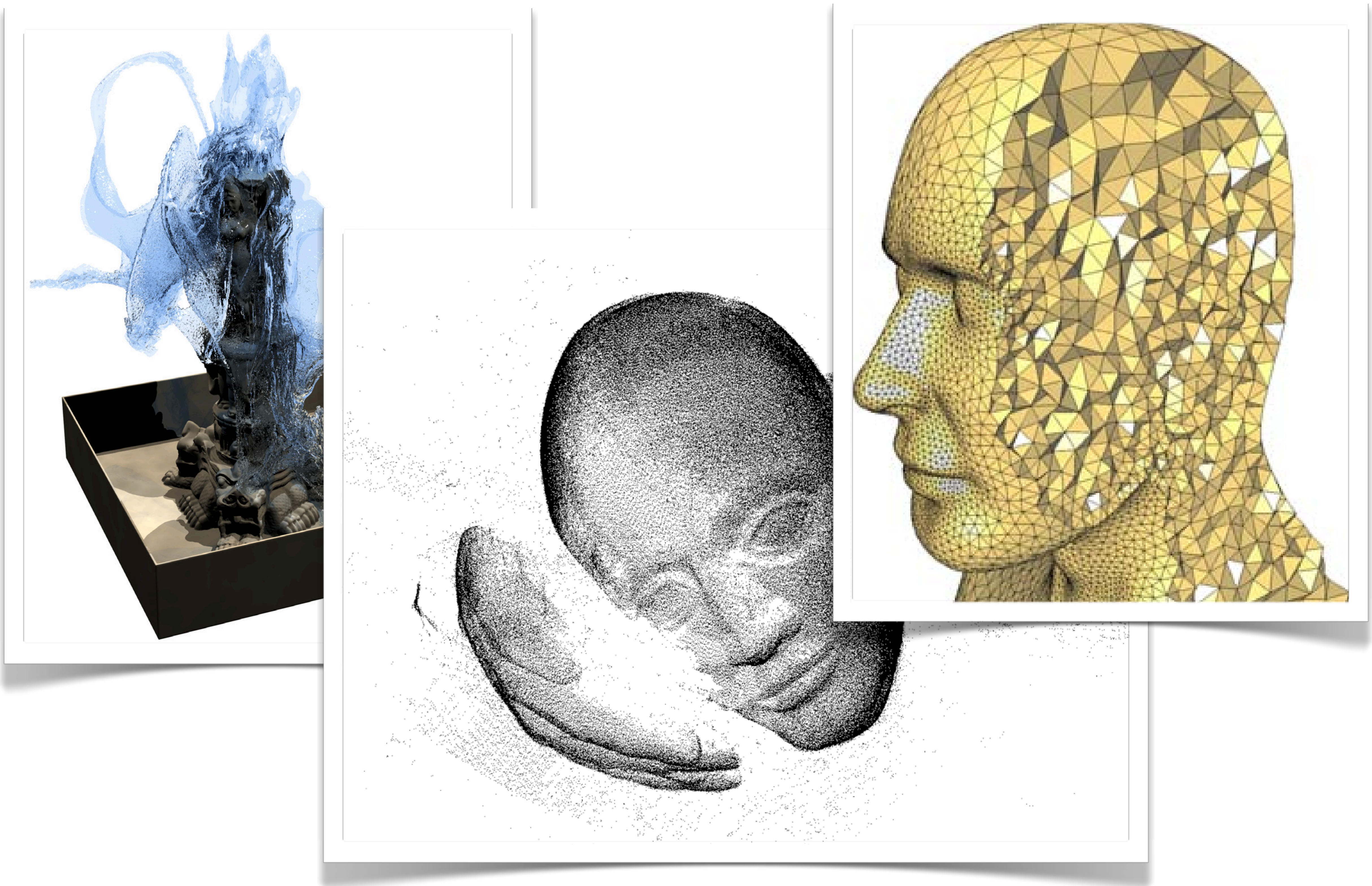
Topics : 2D elasticity



Topics : 3D elasticity

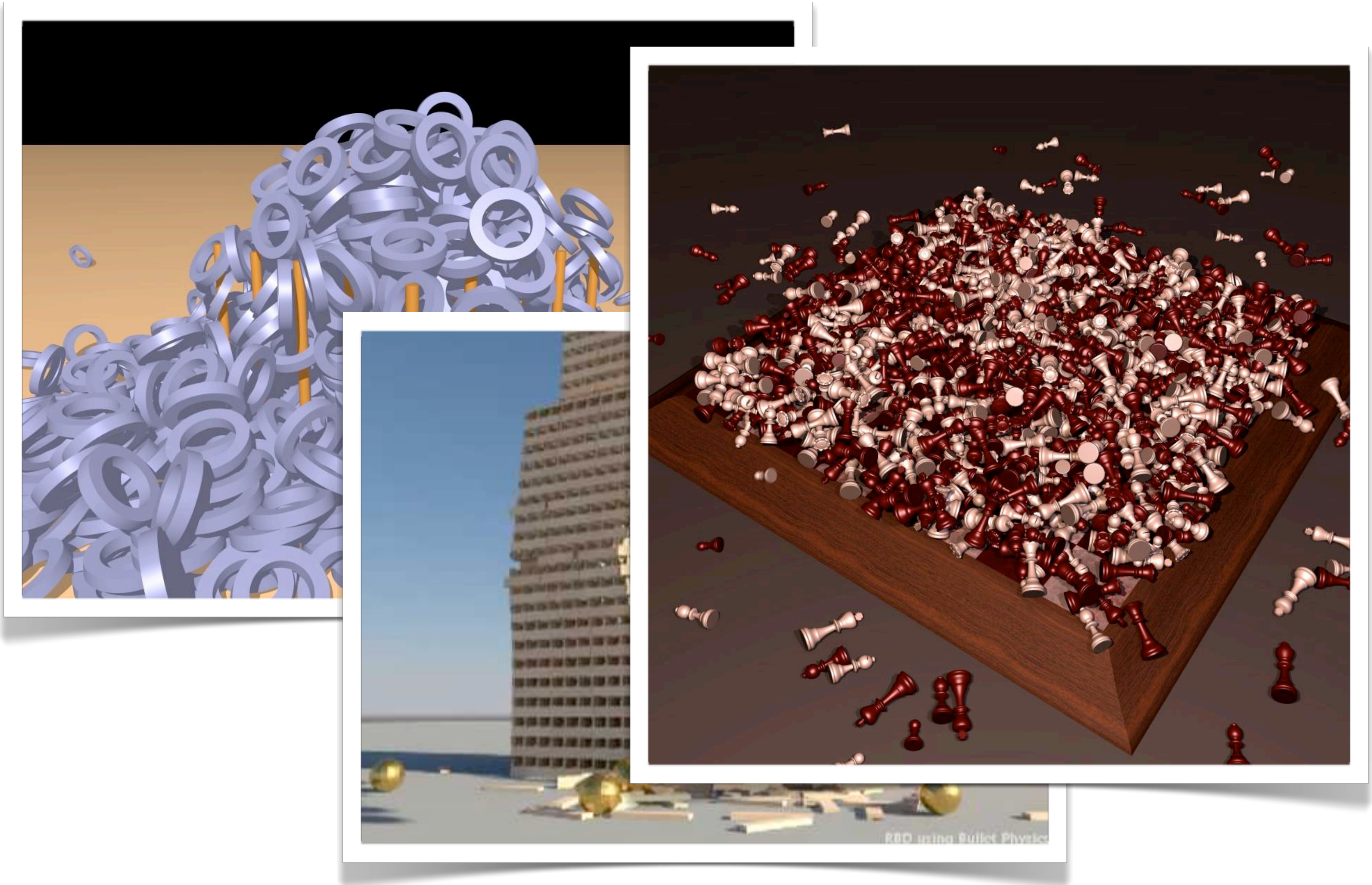


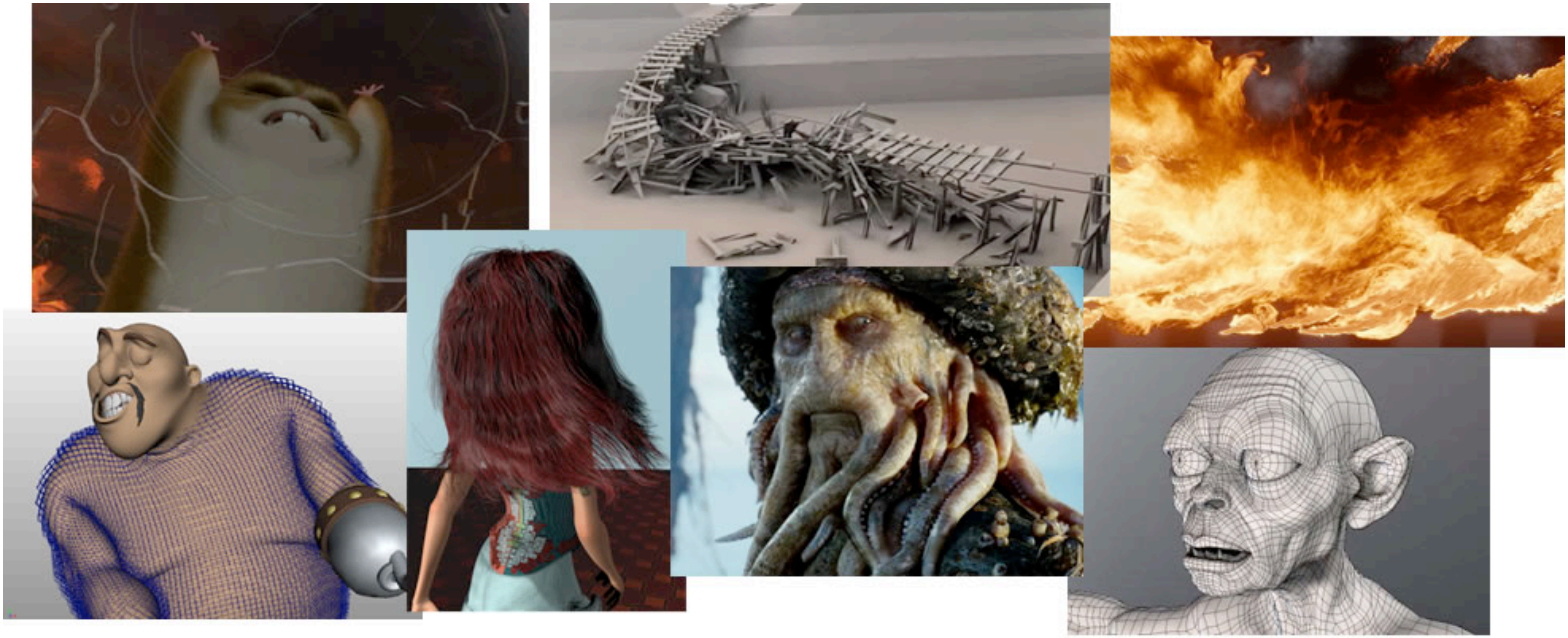
Topics : Discrete geometry representations



Topics : Introduction to fluid dynamics

Topics : Rigid bodies





Welcome to CS838!
Topics In Computing: Introduction to Physics-based Modeling and Simulation

Fall 2012, 10:00-11:15 MWF