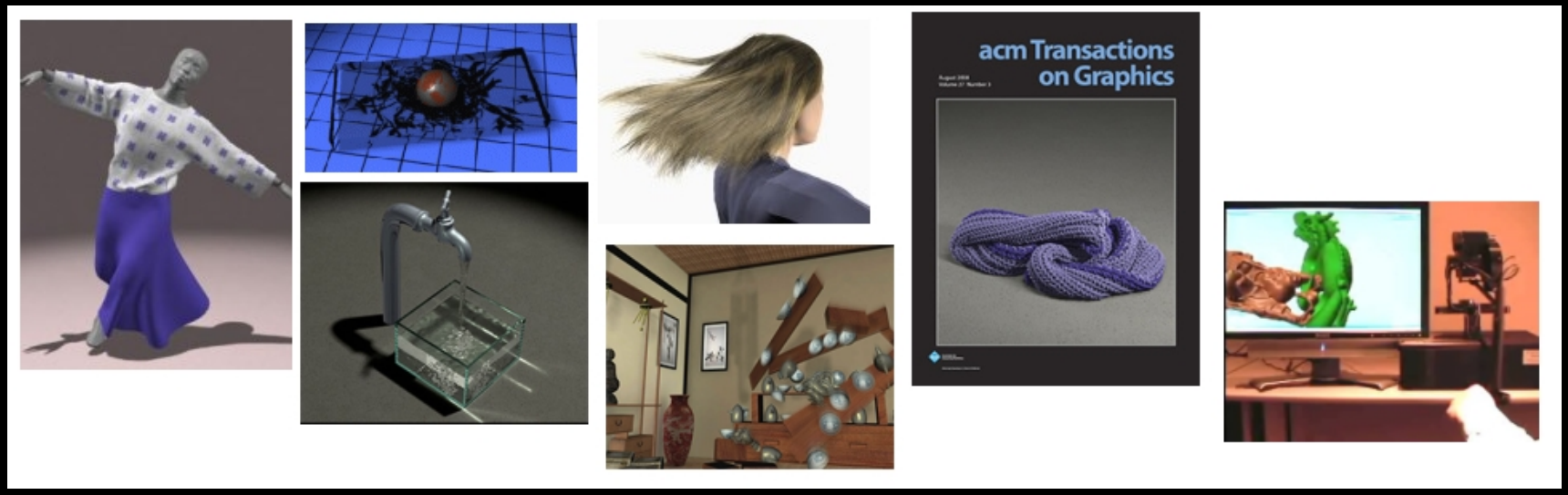
CSCI 520
Computer Animation and Simulation

Computer Animation and Simulation



About the teacher

- Assistant professor in CS
- Post-doc at MIT (2 years)
- PhD, Carnegie Mellon University
- jnb@usc.edu

About the teacher

- Background:
  BSc Mathematics
  PhD Computer Science

- Research interests:
  graphics, animation, real-time physics, control, sound, haptics

Teaching Assistant

- Yijing Li
- Tuesday, 1:00-3:00pm
- SAL 235

Who is the course for

- PhD students
- MSc students
- Advanced undergraduates

- CS 580 background will be very helpful !!

Why take this course

- Opens the door to jobs in computer graphics
- Make better games
- Put math and physics to use in the real world
- Real-time graphics is cool
- Impress your friends with demos

Prerequisites

- A grade of at least B in CS480 or CS580, or explicit permission of instructor
- Familiarity with calculus, linear algebra and numerical computation
- C/C++ programming skills
- See me if you are missing any and we haven't discussed it

Recommended Textbooks

- **Rick Parent: Computer Animation, Second Edition: Algorithms and Techniques**
- **OpenGL Programming Guide ("Red Book")**

  Basic version also available on-line (see Resources)

Evaluation

- Assignments: 3 x 21%
- Final Exam: 37%

Academic integrity

- No collaboration!
- Do not copy any parts of any of the assignments from anyone
- Do not look at other students' code, papers, assignments or exams
- USC Office of Student Judicial Affairs and Community Standards will be notified

Assignment Policies

- Programming assignments
  - Hand in via Blackboard by end of due date
  - Functionality and features
  - Style and documentation
  - Artistic impression
- 3 late days, usable any time during semester
- Academic integrity policy applied rigorously

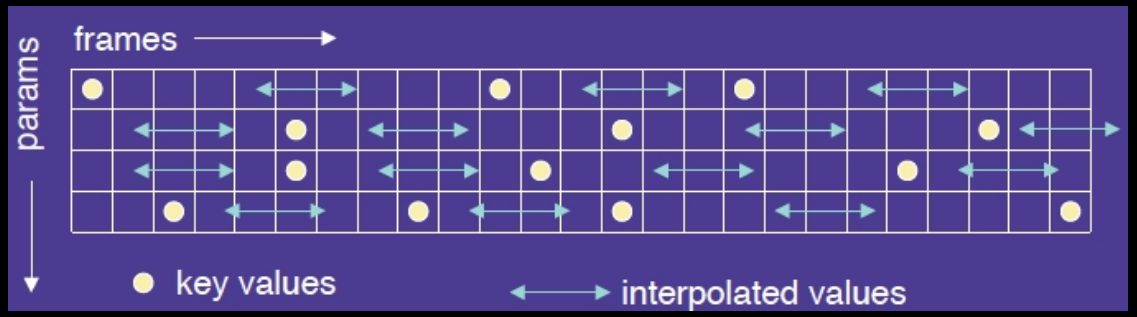
Class goals

- Gain ability to create animations and 3D simulations
- Learn a 3D graphics API (or improve skills)
- Improve code optimization skills

Applications

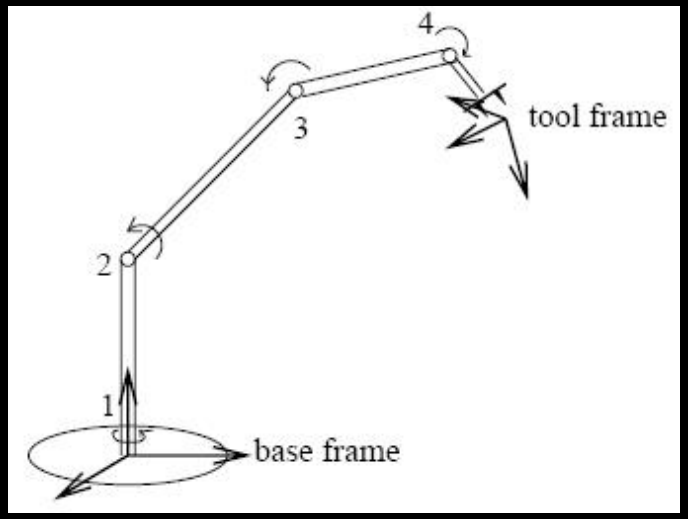
- Virtual reality
- Interactive computer animation
- Surgical simulation; preoperative planning
- Computational robotics; manipulation
- Video games
- Assembly planning
- Scientific visualization
- Education
- E-commerce

Keyframe Animation

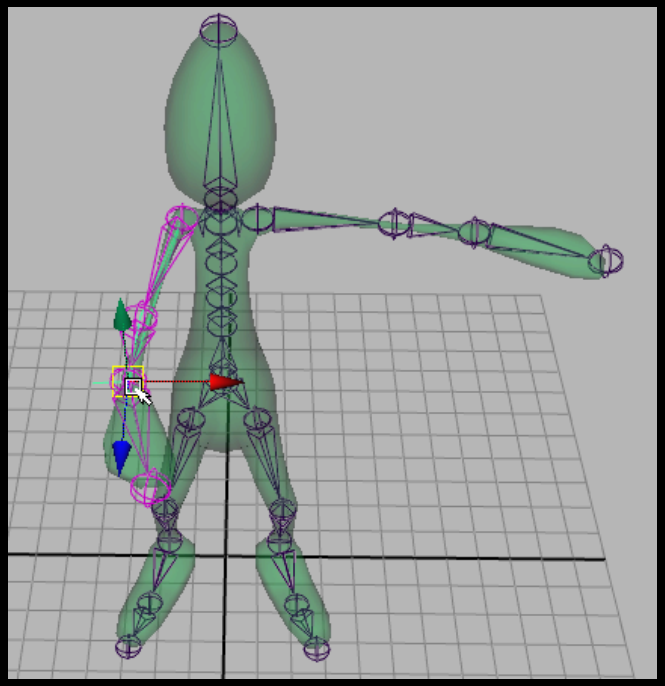


Motion Capture

Inverse Kinematics

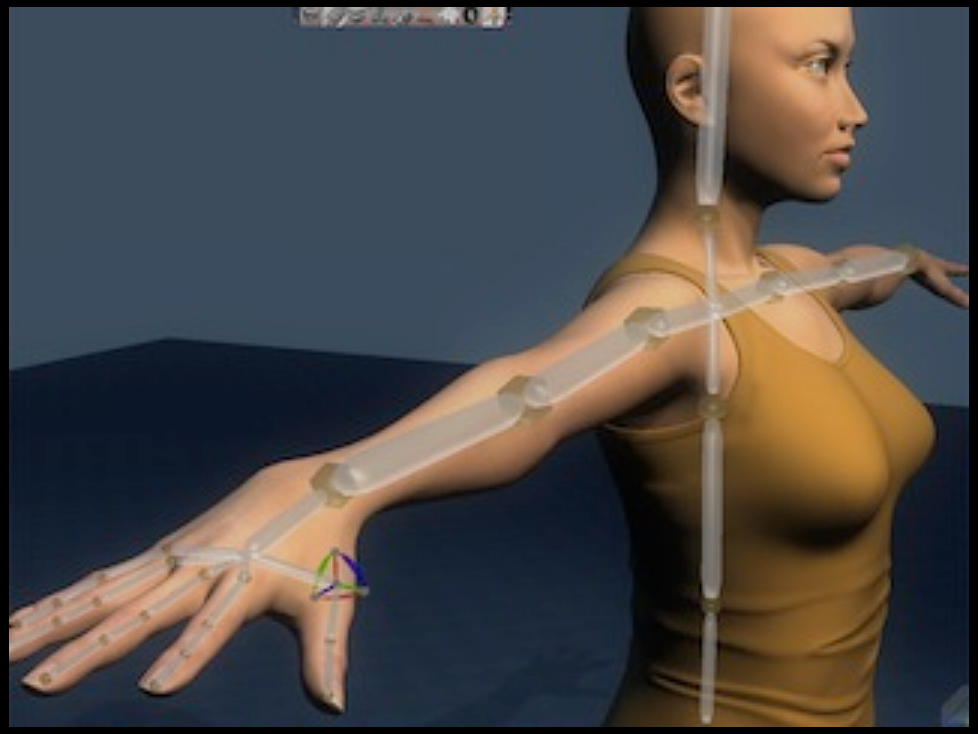


www.learnartificialneuralnetworks.com



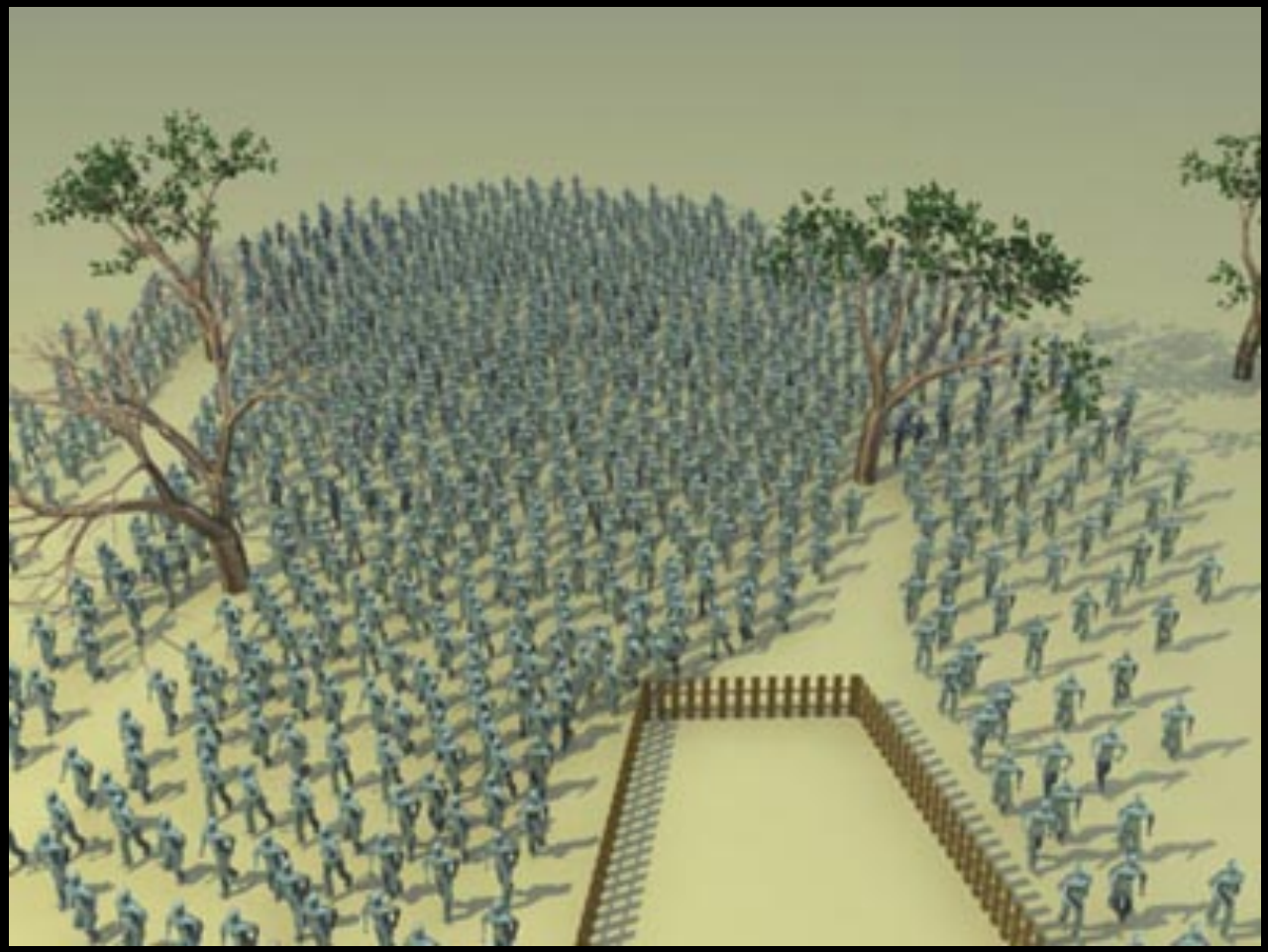
source: Autodesk

Character Rigging



Facial Animation

Crowd Animation



Crowd Animation

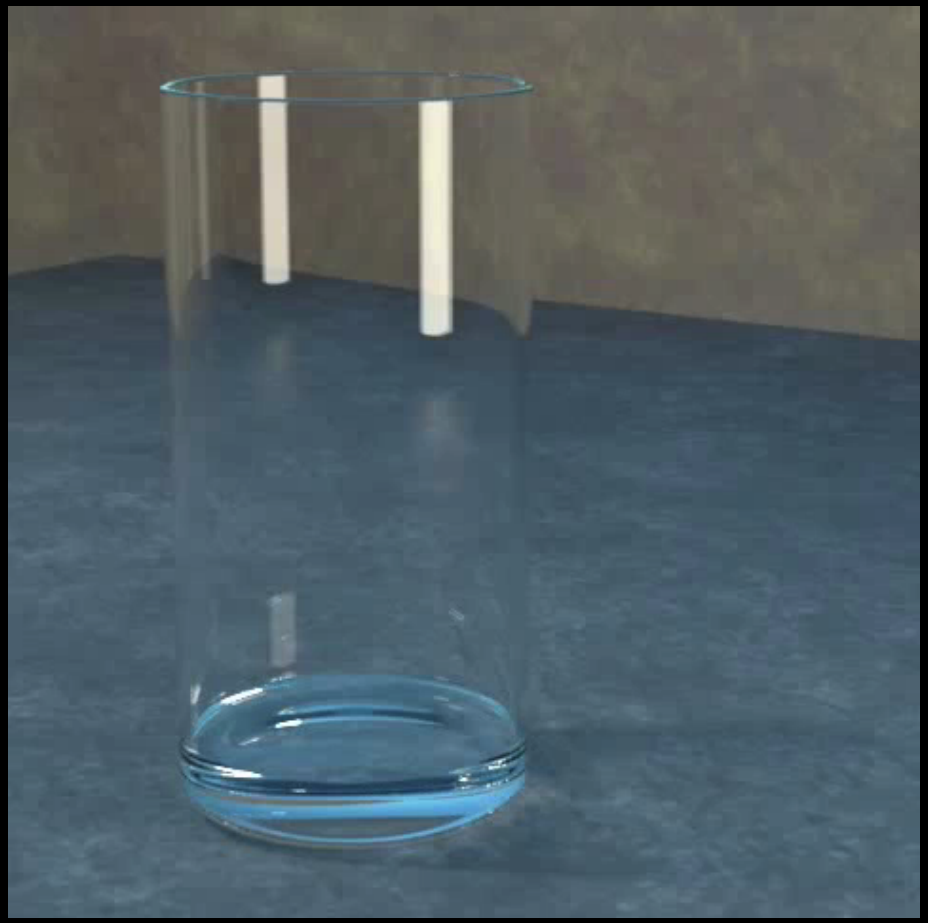
Continuum Crowds

Adrien Treuille
Seth Cooper
Zoran Popović

Maya

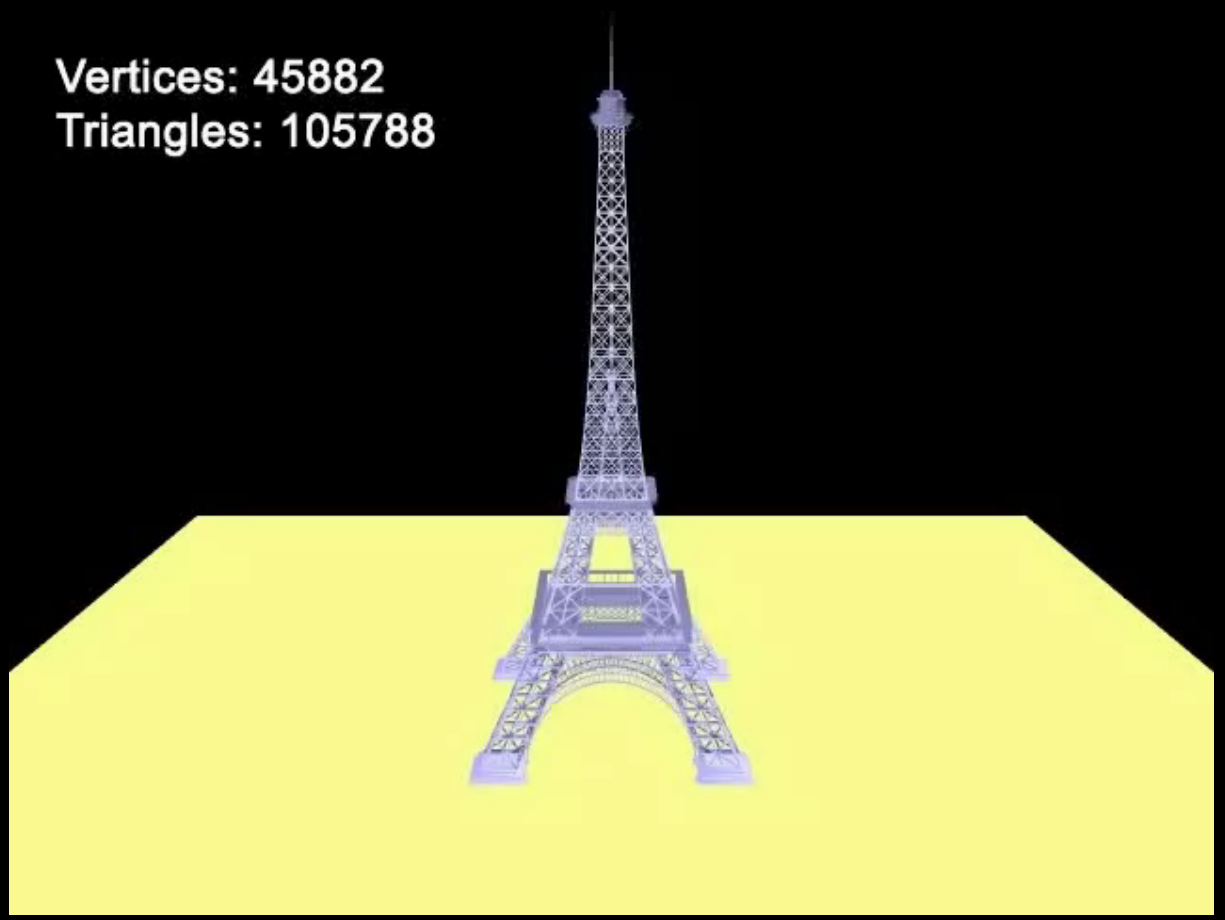

Muscle Spline Attributes
Displacement Map
Render Stats
Object Display
Node Behavior
Extra Attributes
persp

Fluids



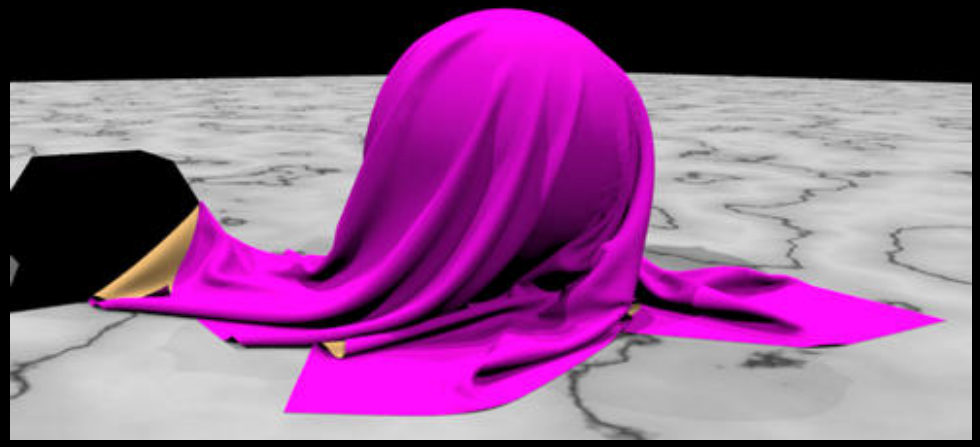
Source:
Stanford University

Deformations



Source:
CMU

Cloth

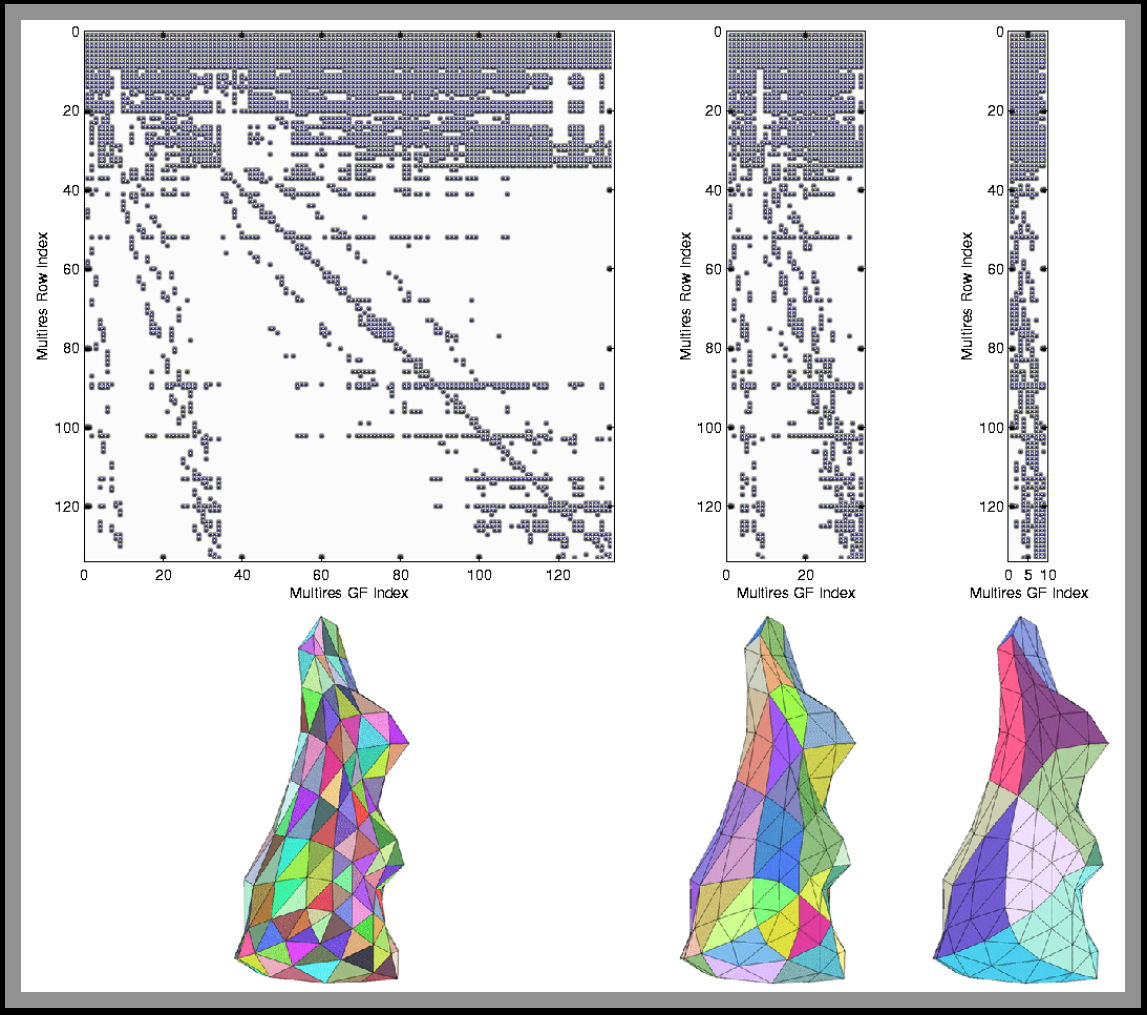


Source:
ACM SIGGRAPH

Simulating Large Models

Source:
Cornell University

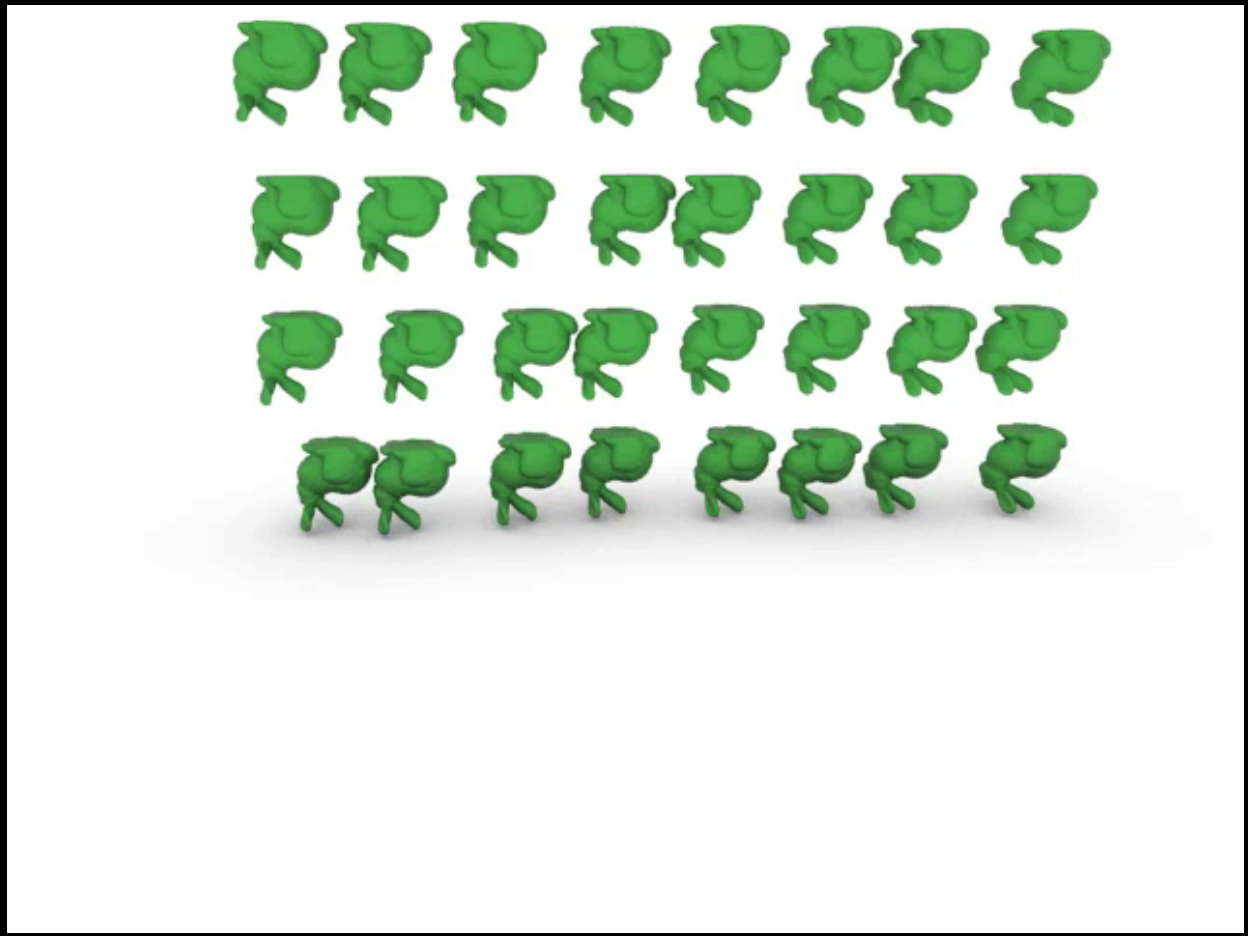
Simulating Large Models



Sound

Source:
CMU

Self-collision detection



Source:
USC

GPU programming

- Vertex shader
- Fragment shader
- CUDA
- OpenCL

Physics in games

Real-Time Deformation and Fracture
in a Game Environment

Eric Parker
Pixelux Entertainment

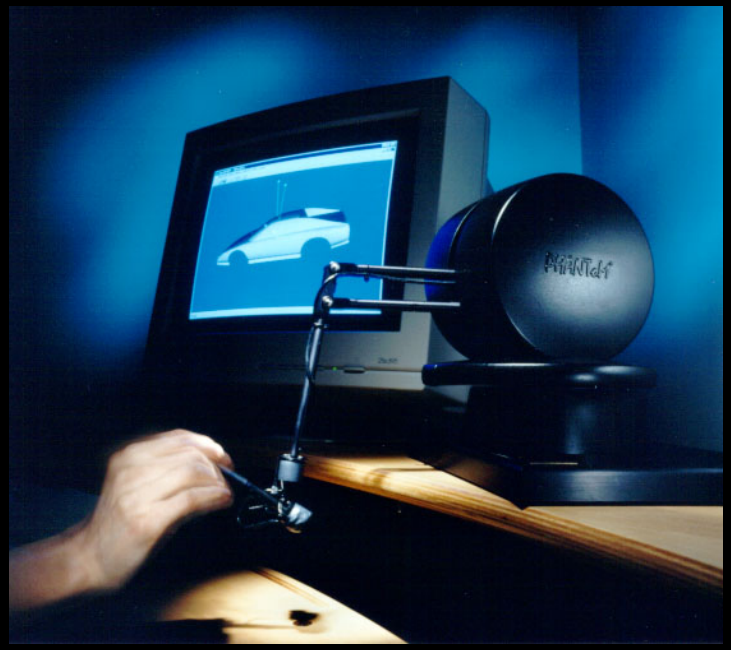
James O'Brien
U.C. Berkeley

Video Edited by Sebastian Burke

From the proceedings of SCA 2009, New Orleans

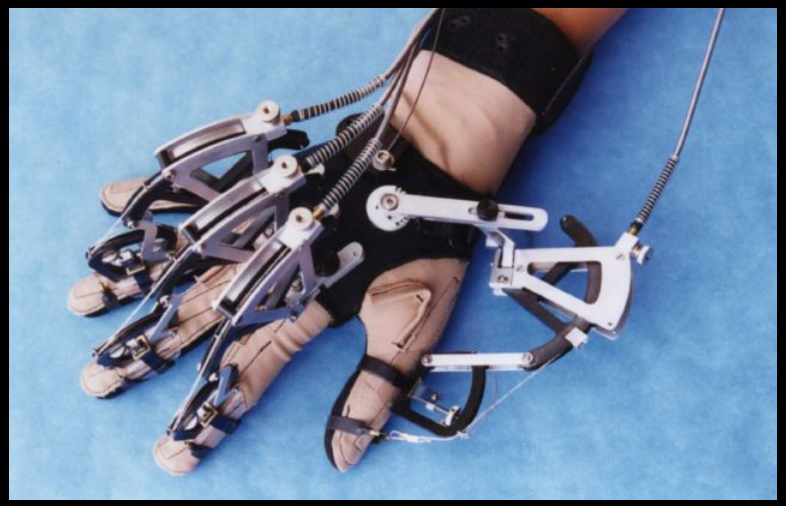
Source:
Symposium on
computer animation

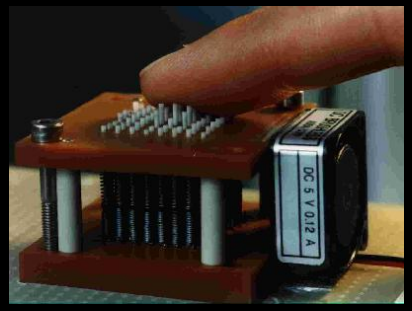
Force-feedback Rendering



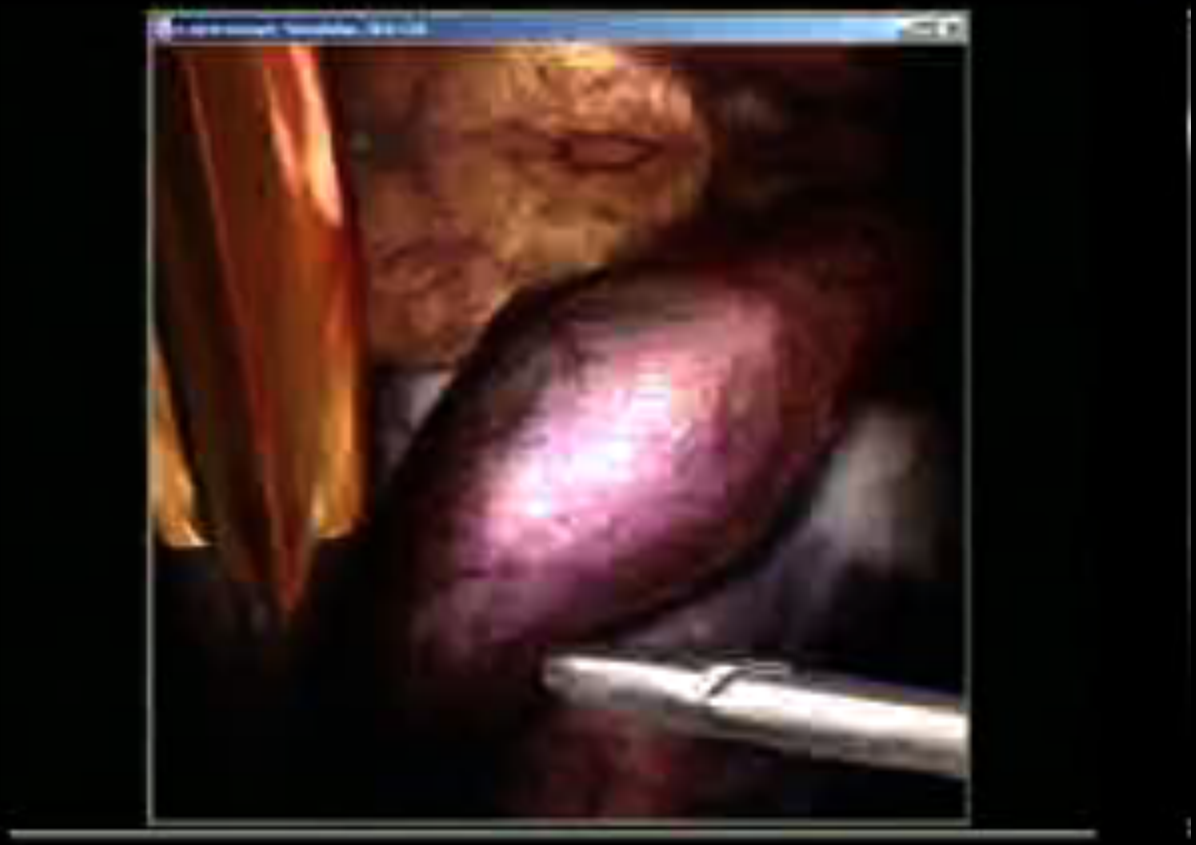
Haptic Interfaces

- hap·tic ('hap-tik)
  adj.
  Of or relating to the sense of touch; tactile.





Surgical Simulation



Source:
Cornell University

Multibody dynamics

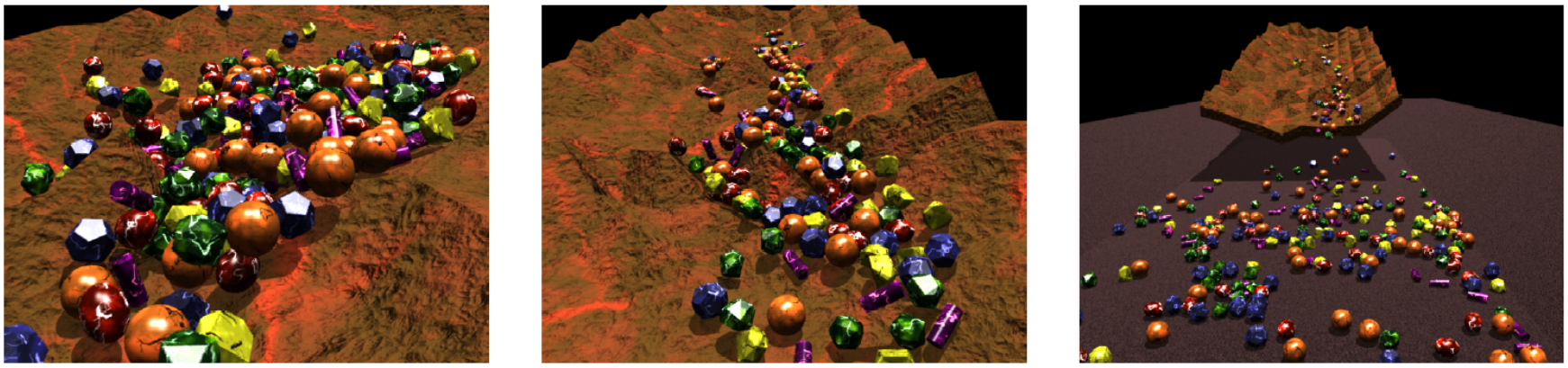


Figure 1: *Avalanche:* 300 rocks tumble down a mountainside.

TOPICS TO BE COVERED:

- Overview of computer animation
- Primer on numerical linear algebra
- Dynamical systems, numerical integration of ODEs
- Constraints and contact
- Character Rigging
- Inverse Kinematics
- Maya
- Crowds
- Rigid body dynamics
- Collision detection
- Structured deformable objects (solids, cloth, hair)
- Fracture and cutting
- Fluids (Navier-Stokes)
- Haptics
- Sound simulation (acoustics)
- Programmable graphics hardware (GPUs)
- Case study: Havok engine for physics in games
- Motion capture

CSCI 520

Computer Animation and Simulation

http://www.jernejbarbic.com