Computer Graphics

Time: **Tuesday 5:00- somewhere around 7:00**

Instructor: **Denis Zorin,**
office hours Tue 11-12pm or by appointment
dzorin@mrl.nyu.edu

Requirements: **Programming and written assignments,**

no midterms, no finals

Web page:
http://www.mrl.nyu.edu/dzorin/intro-graphics

Computer Graphics

Prerequisites

- **Basic linear algebra and calculus**
- **Solid C/C++ programming**

Hardware/Software

- **Any recent PC or UNIX workstation ok**
- **OpenGL and GLUT**

Computer Graphics

Grading

- 80% programming, 20% written
- Programming
  - up to 30% quality of code
  - 50% off for crashes without assert
- Late policy:
  - 24 hours: 20% off
  - 48 hours: 40% off
  - 72 hours: 80% off

Extensions have to be requested in advance

Topics

Image processing

Scan conversion

Modeling basics

Lighting

Rendering algorithms

What is computer graphics?

Computer science: software and hardware systems, vision, computational geometry

Mathematics: transformations, curves, surfaces, PDEs, numerical integration

Physics: light, dynamics

Psychology: perception

Art

Applications

Entertainment

- **Animation and special effects**
- **Games**

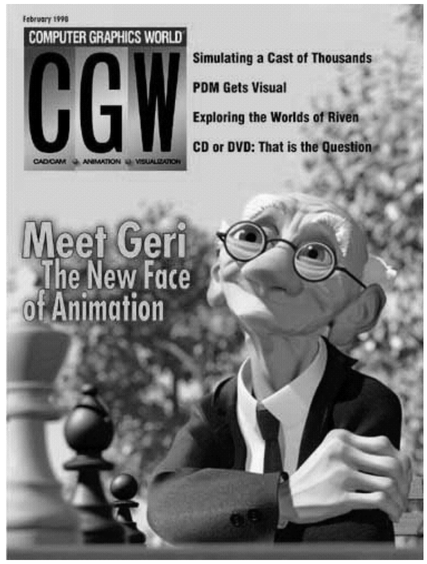
CAD

Scientific Visualization

Medicine

System Visualization

Animation



Games

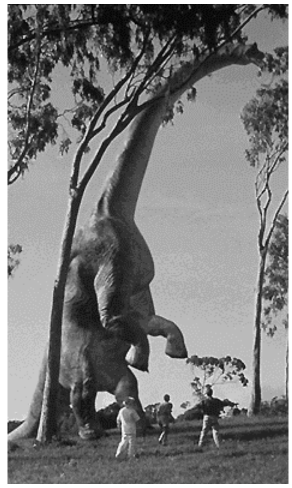
Adventure

Interactive

Special Effects

Merge real and virtual objects

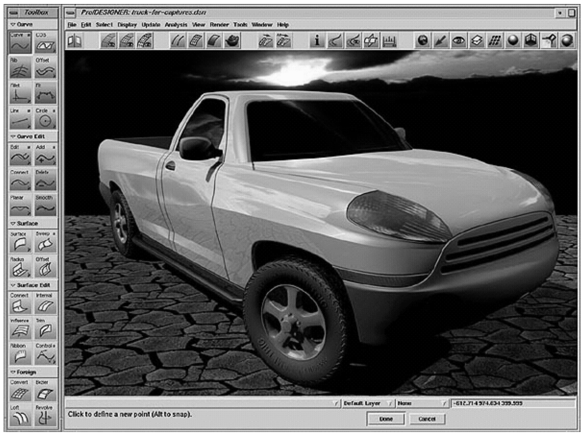
Create virtual scenes indistinguishable from reality



Special Effects

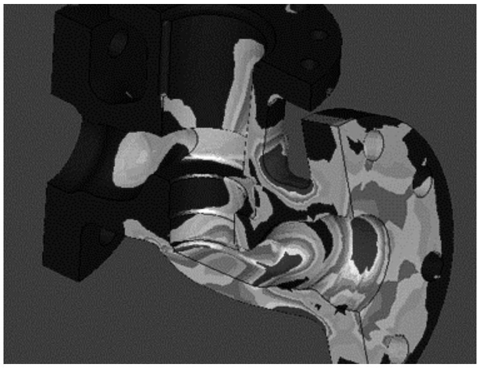
Computer-Aided Design

Conceptual design

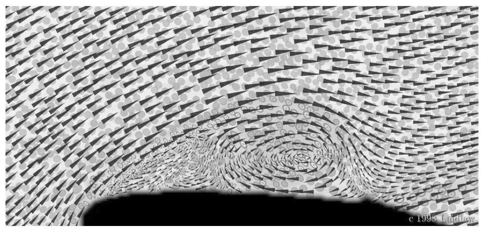


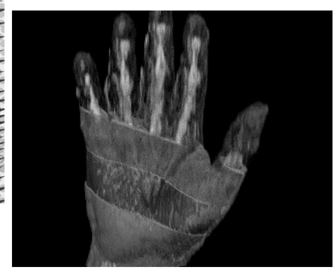
Computer-Aided Design

Simulation



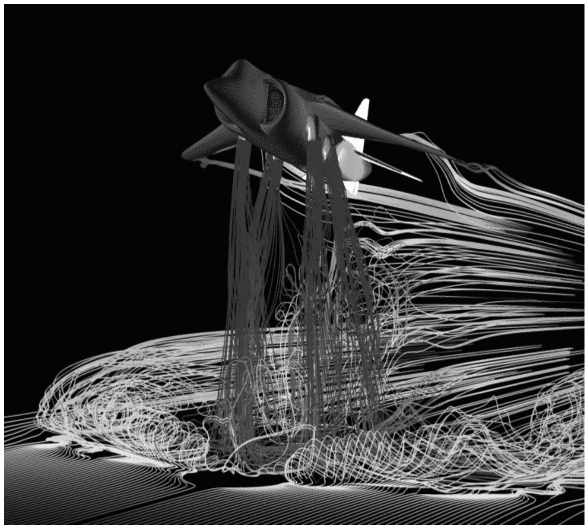
Visualization





Thunderstorm Simulation

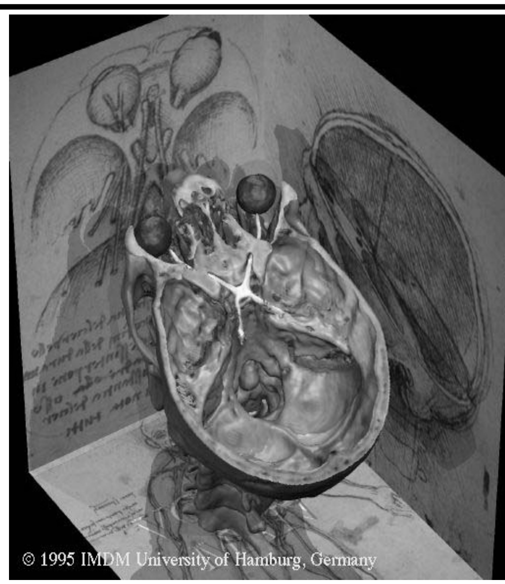
NASA's FAST



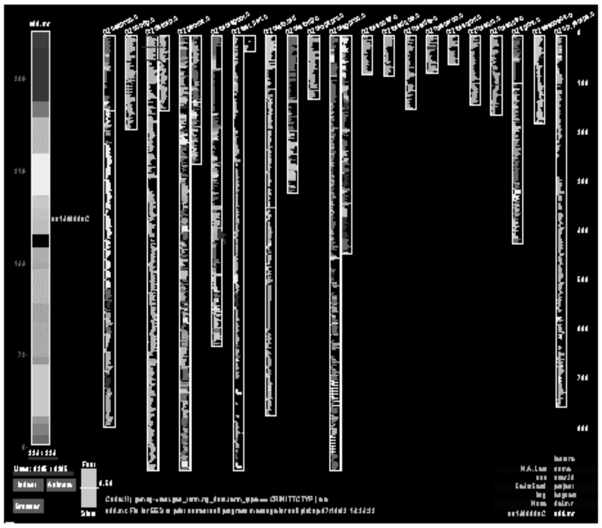
Airflow around a Harrier Jet
FAST System

Nasa Ames

Visible Human



Software Visualization



SeeSoft

S. Eick
Lucent

Shows code age:
new -> red
old -> blue

Modeling

Create the environment

- shapes
- appearance
- views

make it move

- define parameters
- compute how object shape, position, appearance changes

Rendering

Physics-based: simulate light propagation

Empirical: use trial and error to get pictures that are good enough

Nonphotorealistic: imitate artistic styles

Image-based: generate directly from photos or video (no modeling)

Image Processing

Output: typically a raster device

(CRT, LCD, printer)

Discrete colors/intensities

Need to convert continuous data to discrete

Combine real and synthetic

Basic Graphics System

Hardware

- input: mouse, keyboard, trackers, gloves …
- output: monitor, printer, haptic feedback, 3D display…
  graphics hardware (some common algorithms
  implemented in silicon, parallelized)

Sofware

- graphics libraries (OpenGL, Performer, Inventor, Optimizer etc.)
- applications

Graphic Software