CSCI 420 Computer Graphics
Lecture 1

Course Overview

Administrative Issues
Modeling
Animation
Rendering
OpenGL Programming
[Angel Ch. 1]

Jernej Barbic
University of Southern California

Course Information On-Line

http://viterbi-web.usc.edu/~jbarbic/cs420-s23/

- Schedule (slides, readings)
- Assignments (details, due dates)
- Software (libraries, hints)
- Resources (books, tutorials, links)

Submit assignments on Blackboard:

https://blackboard.usc.edu

Forum for questions is on Piazza:

https://piazza.com/usc/spring2023/csci420/home

About me

Full professor in CS

Post-doc at MIT

PhD, Carnegie Mellon University

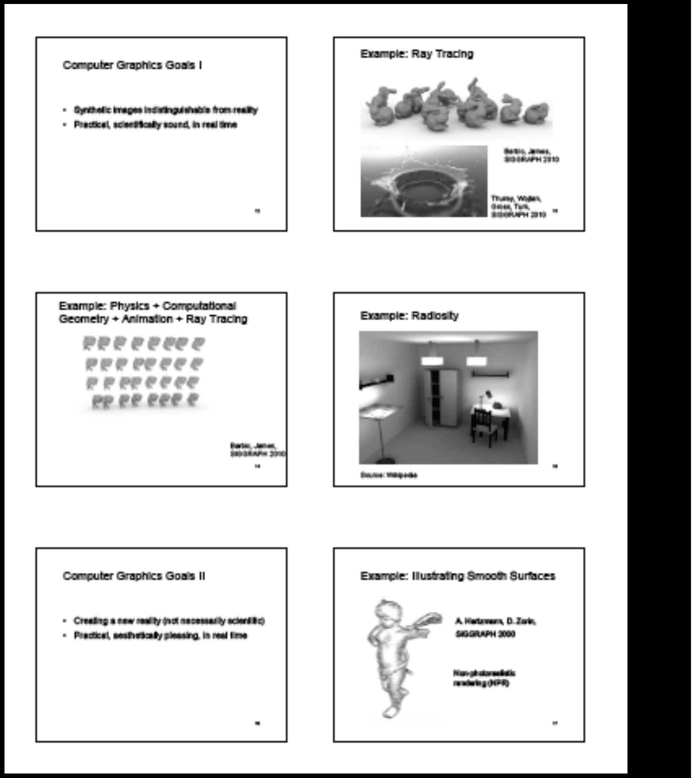
jnb@usc.edu

Mon 4:00-5:00, in person

Course slides

http://viterbi-web.usc.edu/~jbarbic/cs420-s23/

- Full-color version
- 6-slides-per-page B&W version
  -- good for printing
- Posted in advance of lectures
  -- bring to class & annotate
- Color viewing in Acrobat Reader:
  Disable “Replace Document Colors” in
  Preferences.Accessibility (if enabled)



Computer Graphics Goals I

- Synthetic images indistinguishable from reality
- Practical, scientifically sound, in real time

Example: Ray Tracing

Barbic, James, SIGGRAPH 2010

Thuerey, Wojtan, Gross, Turk, SIGGRAPH 2010

Example: Physics + Computational Geometry + Animation + Ray Tracing

Barbic, James, SIGGRAPH 2010

Example: Radiosity

Source: Wikipedia

Computer Graphics Goals II

- Creating a new reality (not necessarily scientific)
- Practical, aesthetically pleasing, in real time

Example: Illustrating Smooth Surfaces

A. Hertzmann, D. Zorin, SIGGRAPH 2000

Non-photorealistic rendering (NPR)

Background:

BSc Mathematics
PhD Computer Science

Research interests:

graphics, animation, real-time physics, control, sound, haptics

Practice:

Tech transfer, startup companies (Ziva Dynamics)

Teaching Assistant

Mianlun Zheng

Office hours:

Tuesday and Friday
4pm-5pm

Teaching Assistant

Jiahao Wen

Office hours:

Tuesday and Friday
4pm-5pm

Course Producer

TBA

Same office hours as TA

Prerequisites

- CSCI 104 (Data Structures and Object-Oriented Design)
- MATH 225 (Linear Algebra and Differential Equations)
- Familiarity with calculus and linear algebra
- C programming skills
- Junior, senior, MS or PhD student, or explicit permission of instructor
- See me if you are missing any and we haven’t discussed it

Grading

- 51% Programming Assignments (3x 17%)
- 19% Midterm Exam
- 30% Final Exam

Textbooks

- **Interactive Computer Graphics**

  A top-down approach with OpenGL, **Sixth** Edition

  Edward Angel, Addison-Wesley

- **OpenGL Programming Guide (“Red Book”)**

  Basic version also available on-line (see Resources)

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

- All assignments must be completed to pass the course

- Academic integrity policy applied rigorously

Computer Graphics

One of the “core” computer science disciplines:

Algorithms and Theory
Artificial Intelligence
Computer Architecture
Computer Graphics and Visualization
Computer Vision
Computer Security
Computer Systems
Databases
Networks
Programming Languages
Software Engineering

Course Overview

Theory: Computer graphics disciplines:

- Modeling: how to represent objects
- Animation: how to control and represent motion
- Rendering: how to create images of objects
- Image Processing: how to edit images

Practice: OpenGL graphics library

Not in this course:

- Human-computer interaction
- Graphic design
- User interface libraries

OpenGL Graphics Library

- Main focus:
  Core OpenGL Profile ("Modern OpenGL")
- OpenGL 3.2 and higher
- Shaders
- Homeworks use the Core Profile
- We will also study:
  Compatibility Profile ("Classic OpenGL")

Computer Graphics Disciplines

RENDERED USING DALI - HENRIK WANN JENSEN 2000

Source: Jensen

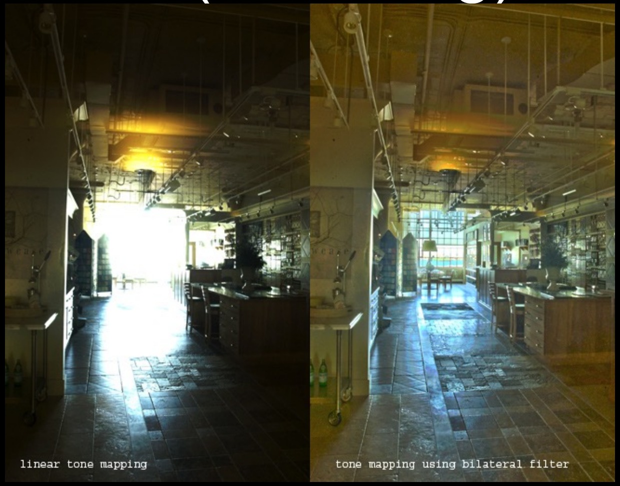
Rendering

Source: Botsch et al.

Geometry (Modeling)

Source: Baraff and Witkin

Animation



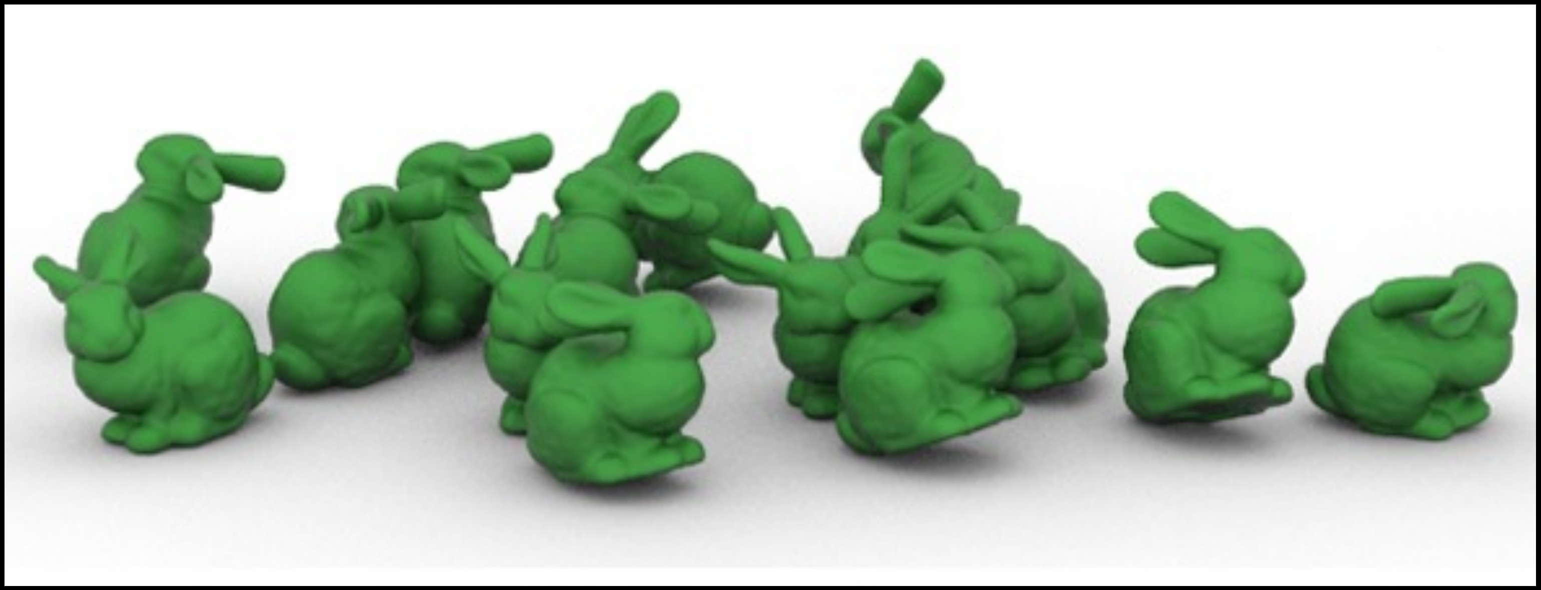
Source: Durand

Image Processing

Computer Graphics Goals I

- Synthetic images indistinguishable from reality
- Practical, scientifically sound, in real time

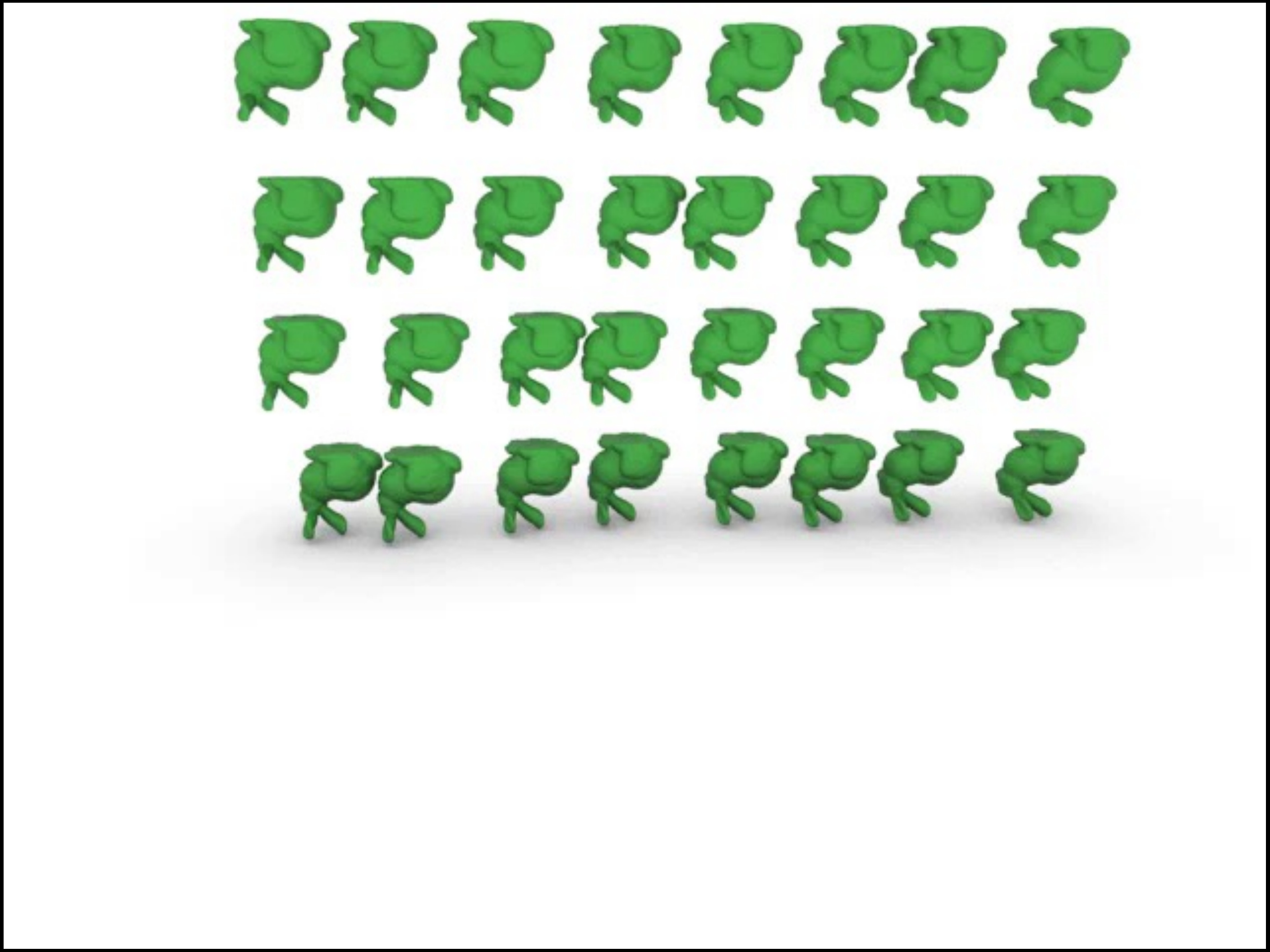
Example: Ray Tracing



Barbic, James,
SIGGRAPH 2010

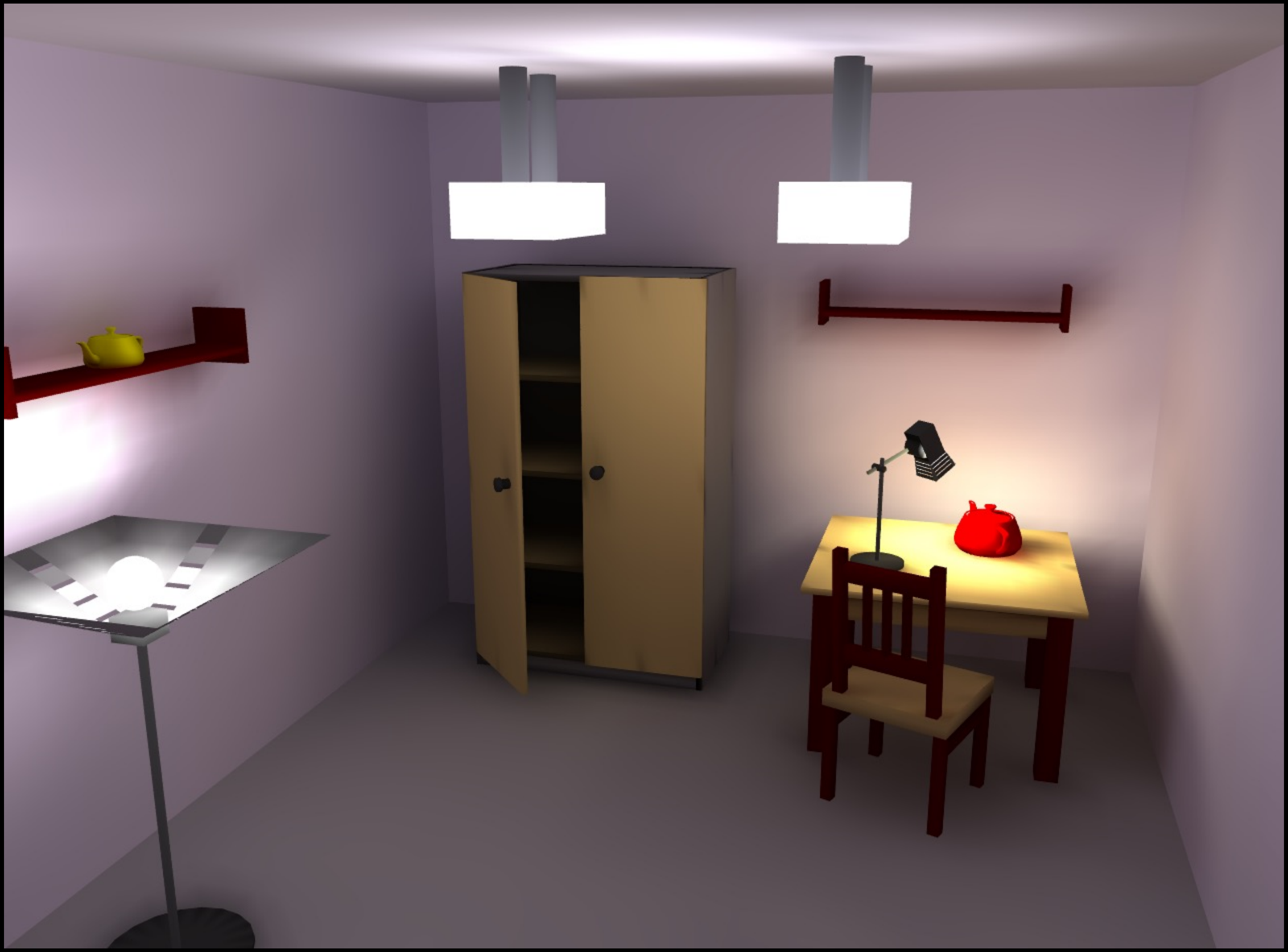
Thurey, Wojtan,
Gross, Turk,
SIGGRAPH 2010

Example: Physics + Computational Geometry + Animation + Ray Tracing



Barbic, James, SIGGRAPH 2010

Example: Radiosity

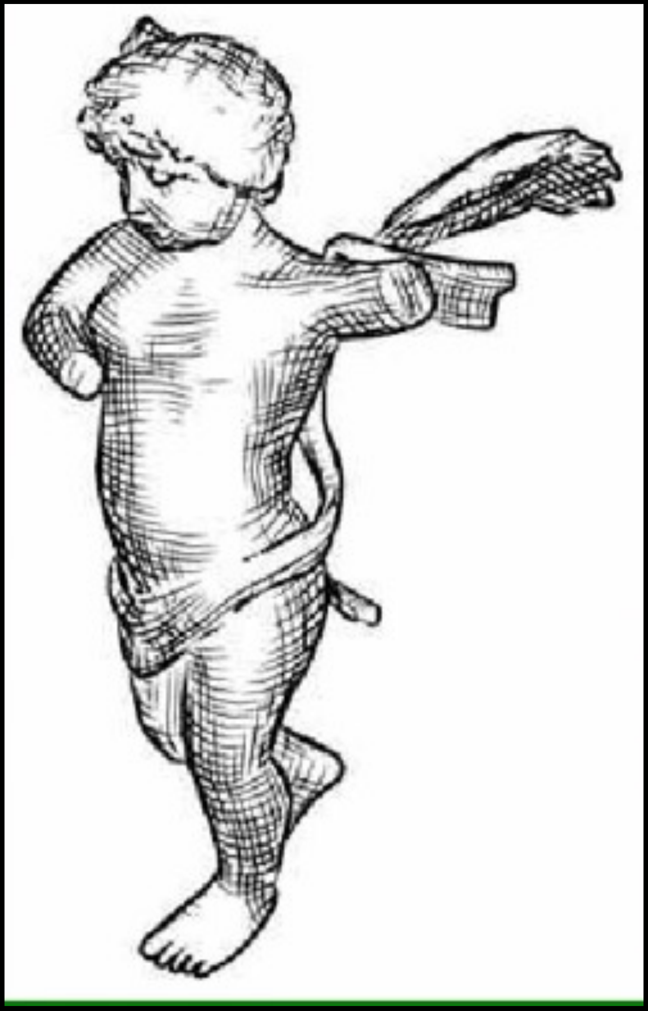


Source: Wikipedia

Computer Graphics Goals II

- Creating a new reality (not necessarily scientific)
- Practical, aesthetically pleasing, in real time

Example: Illustrating Smooth Surfaces

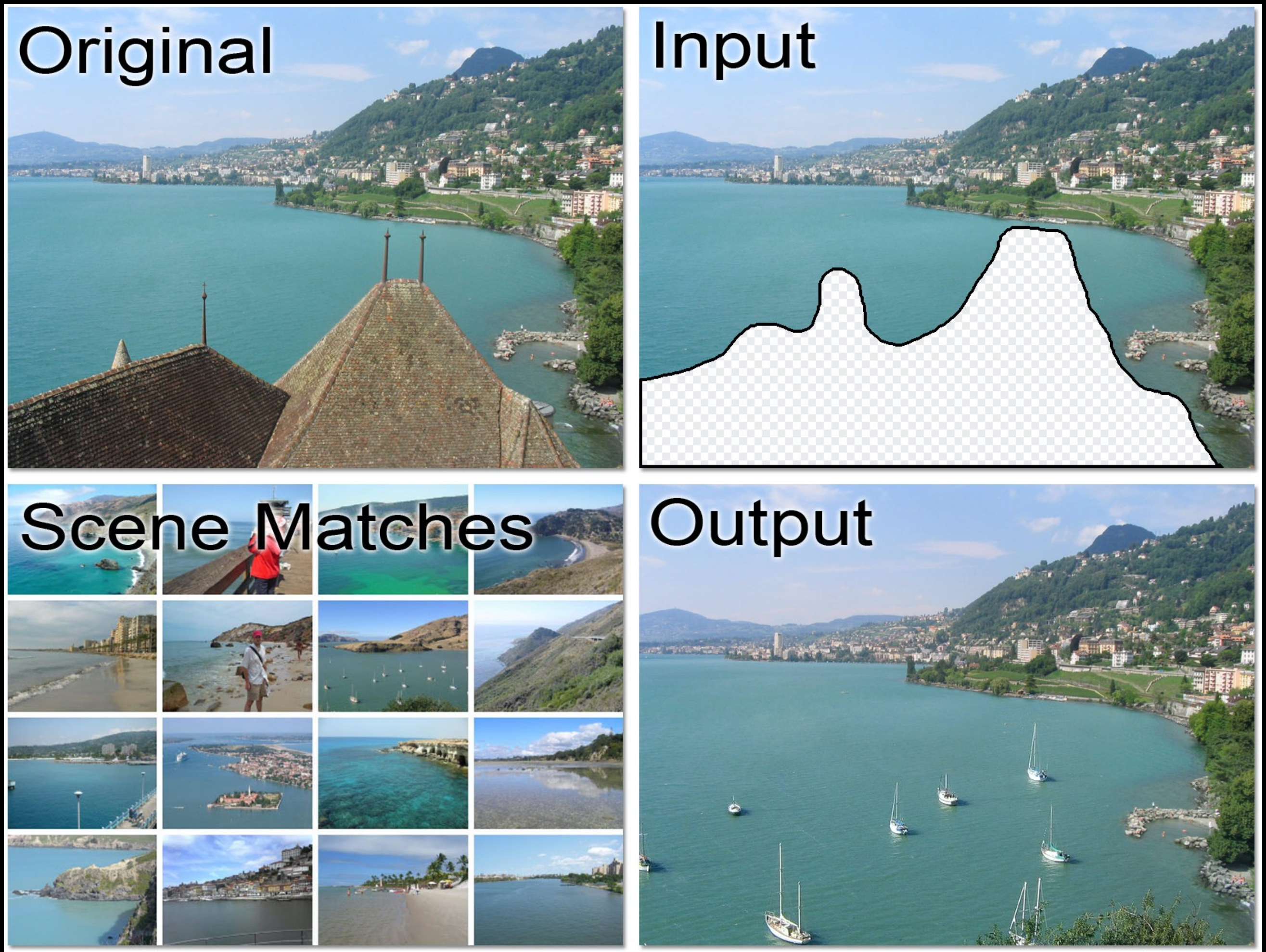


A. Hertzmann, D. Zorin,

SIGGRAPH 2000

Non-photorealistic rendering (NPR)

Example: Scene Completion



J. Hays, A. Efros, SIGGRAPH 2007

SIGGRAPH

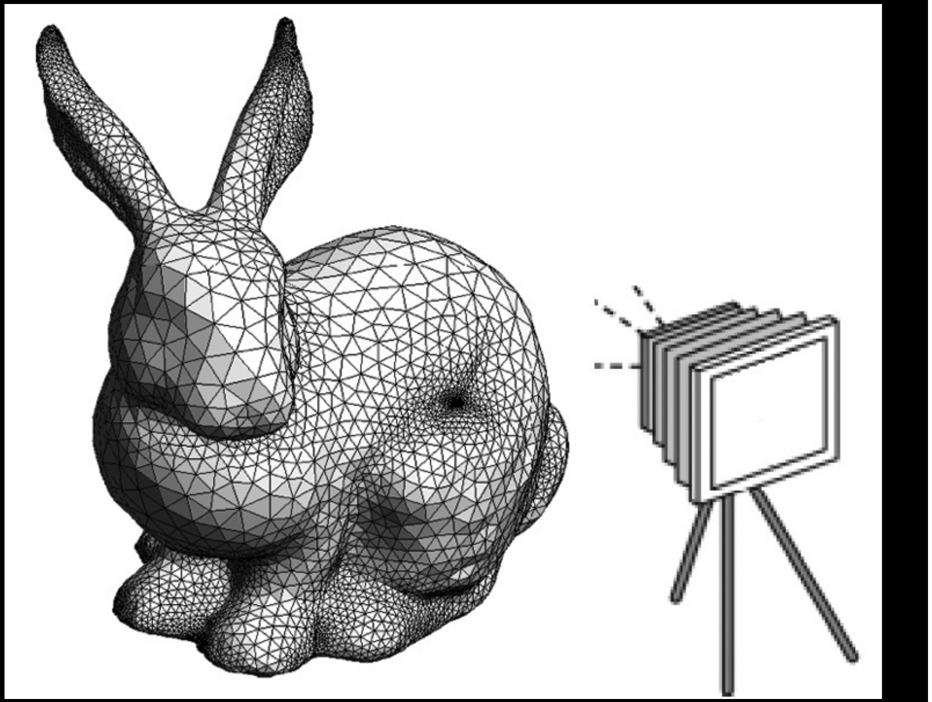
- Main computer graphics event in the world
- Once per year
- 30,000 attendees
- Academia, industry

1. Course Overview

- Administrative Issues
- Topics Outline (next)

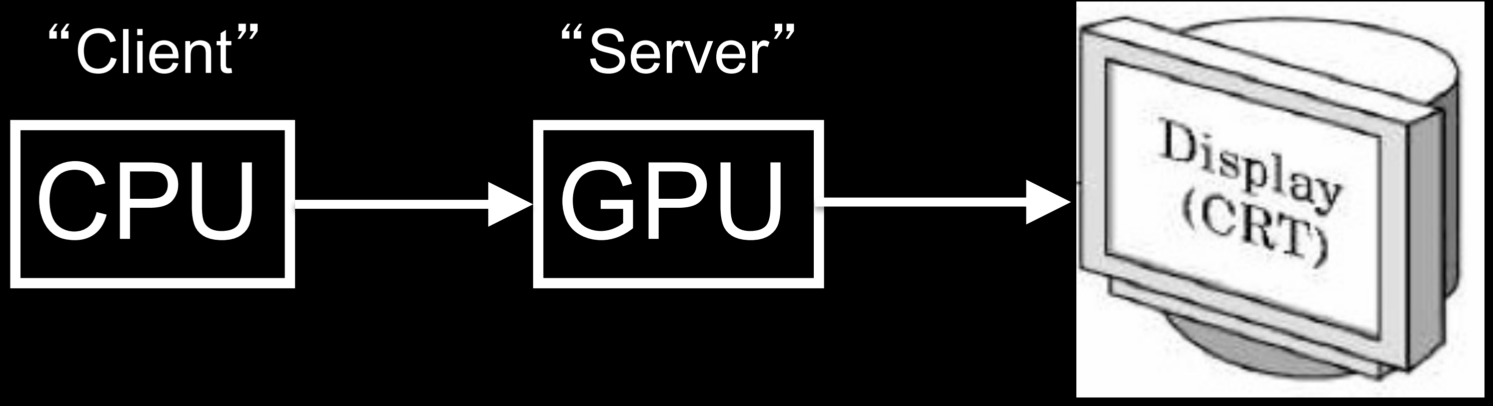
2. OpenGL Basics

- Graphics pipeline
- Primitives and attributes
- Color
- OpenGL core and compatibility profiles
- [Angel, Ch. 1, 2]



3. Input and Interaction

- Clients and servers
- Event driven programming
- Hidden-surface removal
- [Angel, Ch. 2]



“Client”

CPU

“Server”

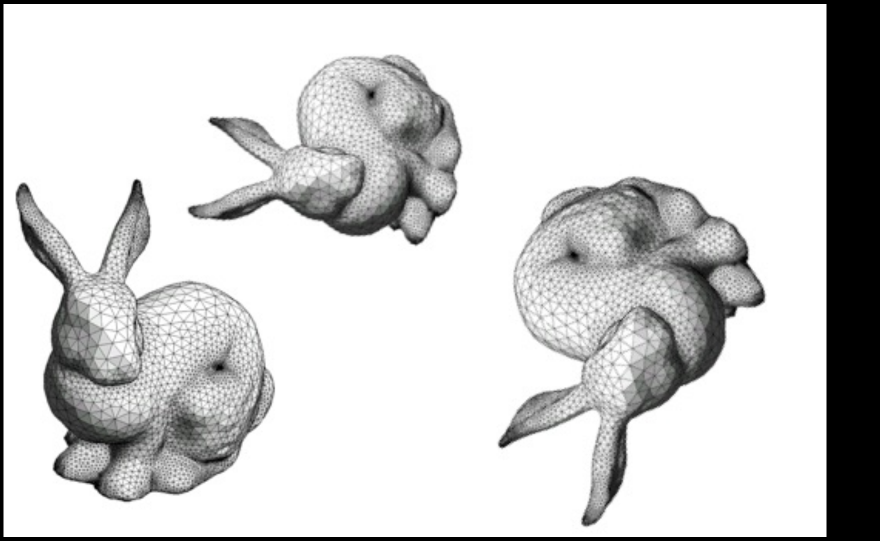
GPU

4. GPU Shaders

- Vertex program
- Fragment program
- Pipeline program
- Shading languages
- GLSL shading language
- Interaction with OpenGL

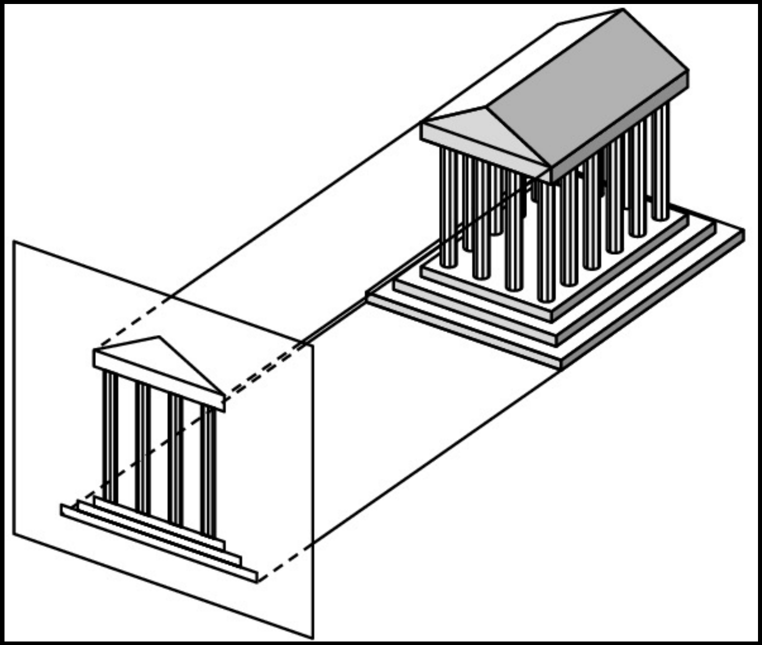
5. Objects & Transformations

- Linear algebra review
- Coordinate systems and frames
- Rotation, translation, scaling
- Homogeneous coordinates
- OpenGL transformation matrices
- [Angel, Ch. 3]



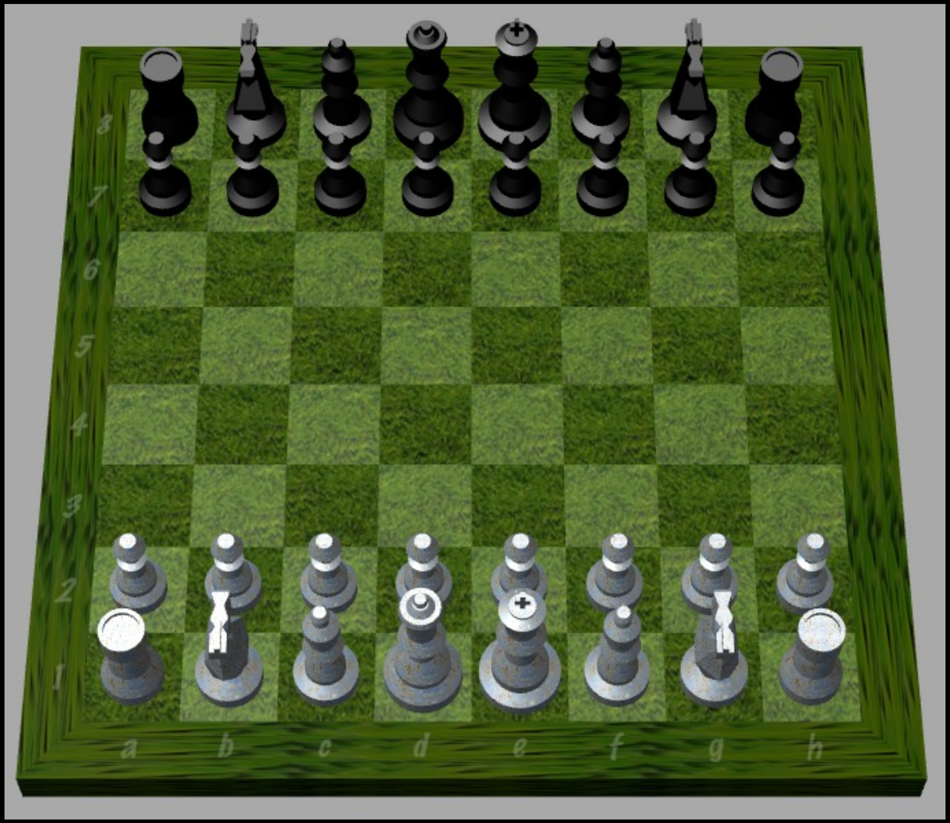
6. Viewing and Projection

- Orthographic projection
- Perspective projection
- Camera positioning
- Projections in OpenGL
- [Angel, Ch. 4]



7. Hierarchical Models

- Re-using objects
- Animations
- OpenGL routines
- Parameters and transformations
- [Angel, Ch. 8]

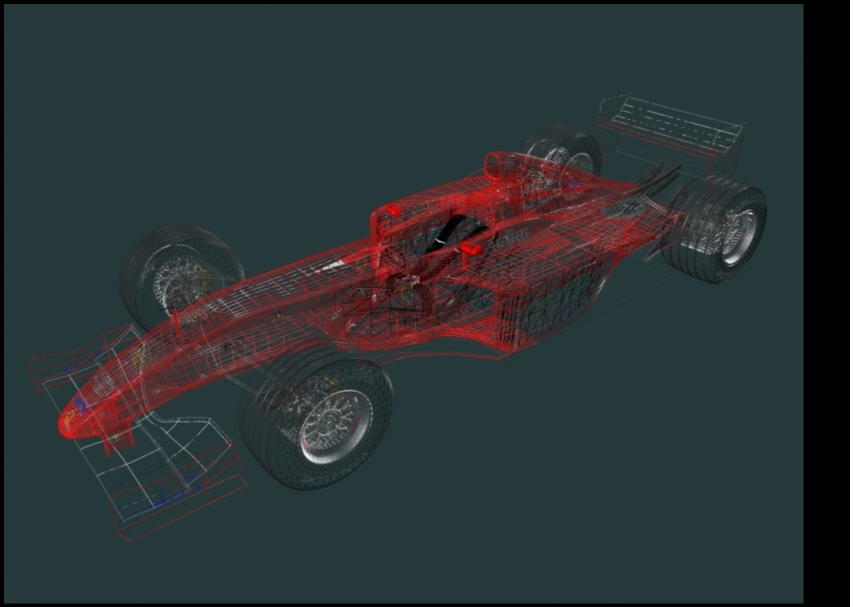


8. Light and Shading

- Light sources
- Ambient, diffuse, and specular reflection
- Normal vectors
- Material properties in OpenGL
- Radiosity
- [Angel, Ch. 5]

Tobias R. Metoc

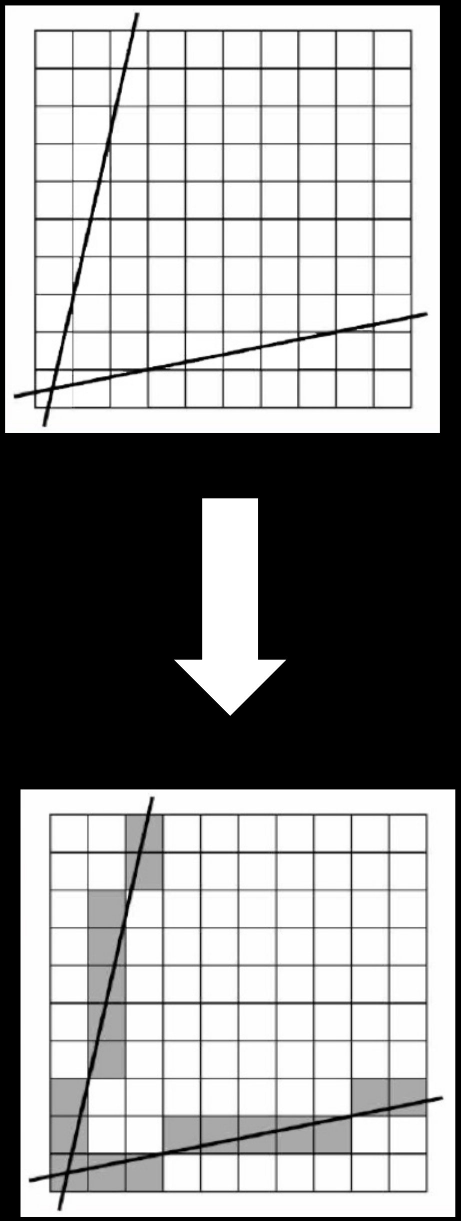
9. Curves and Surfaces



- Review of 3D-calculus
- Explicit representations
- Implicit representations
- Parametric curves and surfaces
- Hermite curves and surfaces
- Bezier curves and surfaces
- Splines
- Curves and surfaces in OpenGL
- [Angel, Ch. 10]

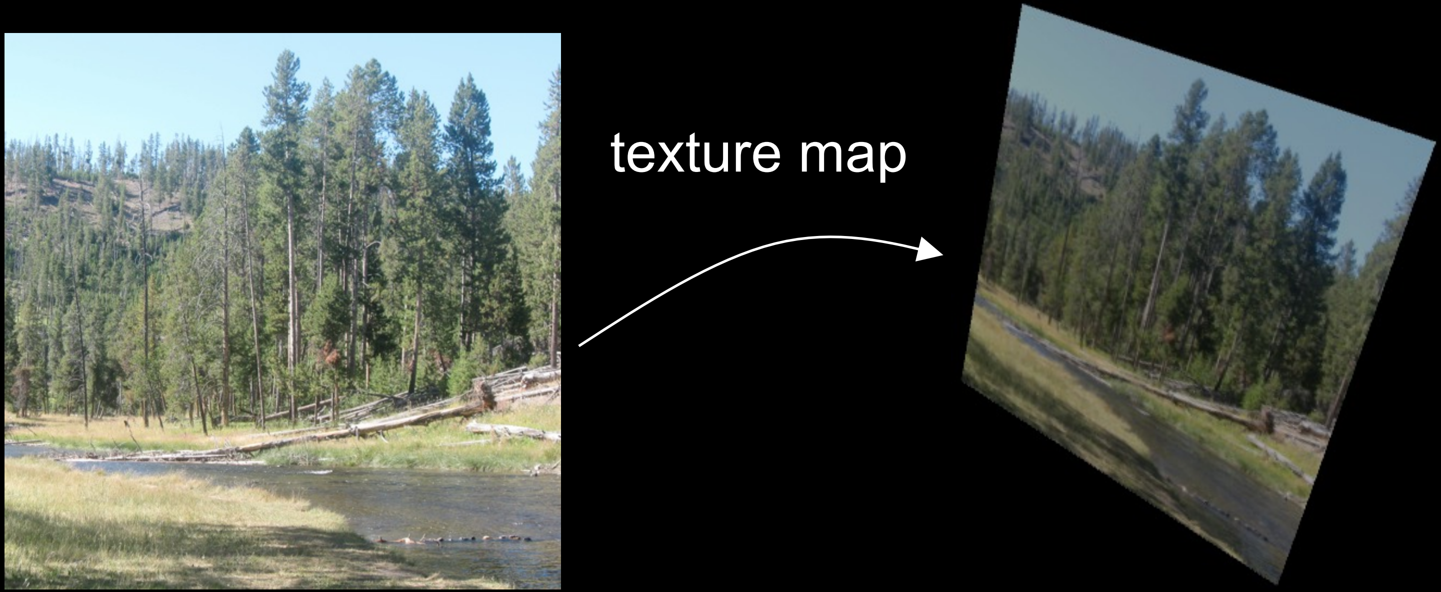
10. Rendering

- Clipping
- Bounding boxes
- Hidden-surface removal
- Line drawing
- Scan conversion
- Antialiasing
- [Angel, Ch. 6]



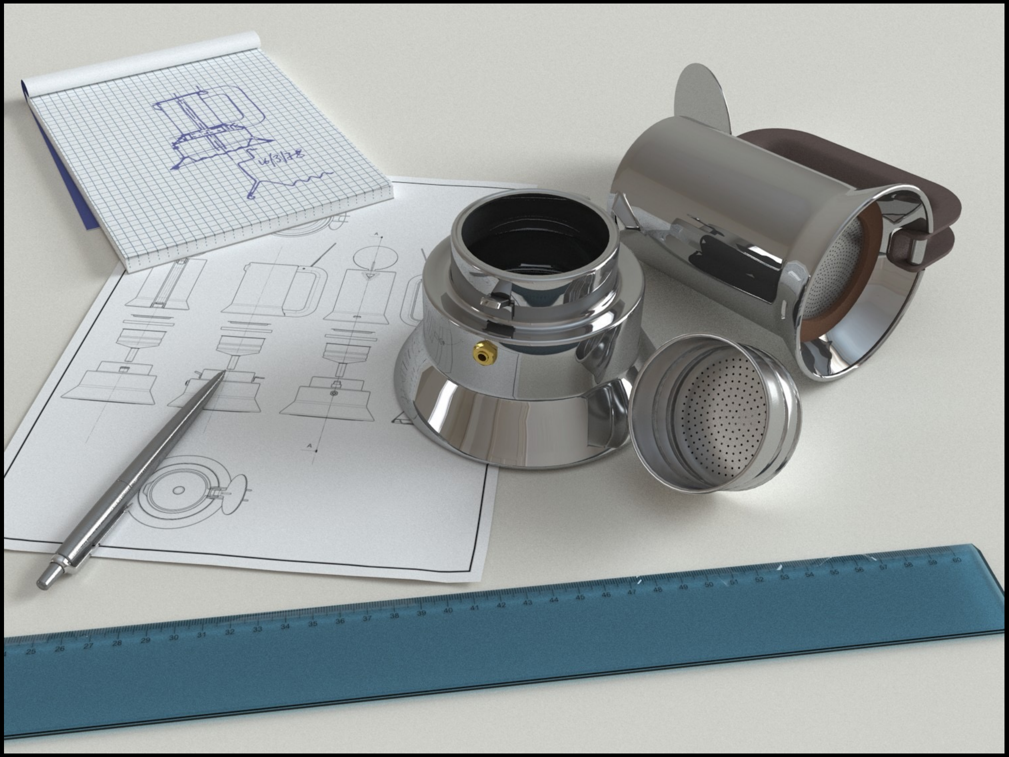
11. Textures and Pixels

- Texture mapping
- OpenGL texture primitives
- Bump maps
- Environment maps
- Opacity and blending
- Image filtering
- [Angel, Ch. 7]



12. Ray Tracing

- Basic ray tracing [Angel, Ch. 11]
- Spatial data structures [Angel, Ch. 8]
- Motion Blur
- Soft Shadows



www.yafaray.org

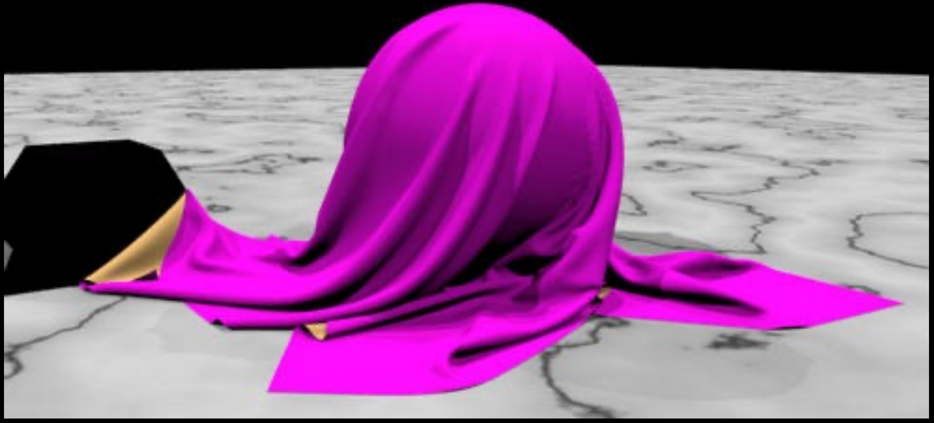
13. Radiosity

- Local vs global illumination model
- Interreflection between surfaces
- Radiosity equation
- Solution methods
- [Angel Ch. 11]

Cornell University

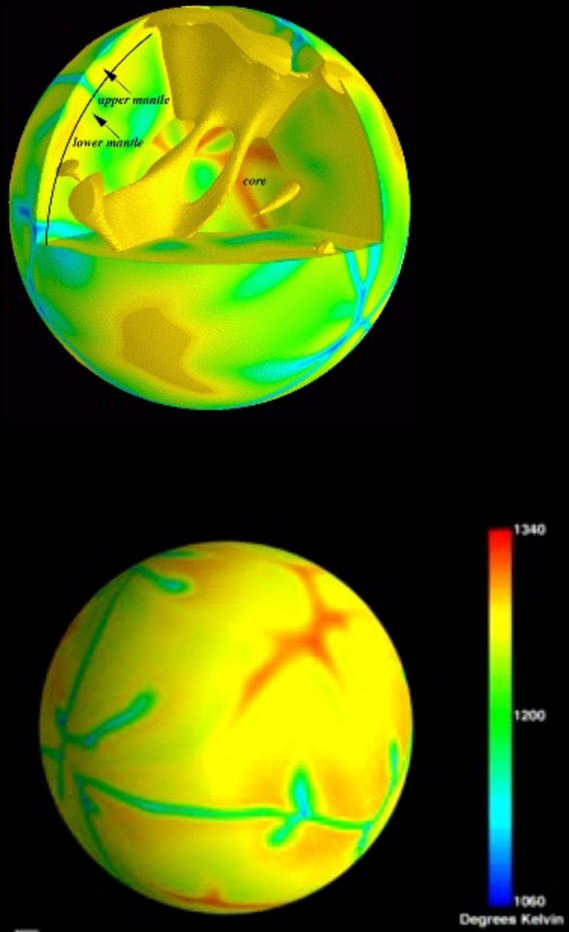
14. Physically Based Models

- Particle systems
- Spring forces
- Cloth
- Collisions
- Constraints
- Fractals
- [Angel, Ch. 9]



15. Scientific Visualization

- Height fields and contours
- Isosurfaces
- Volume rendering
- Texture mapping of volumes
- [Angel Ch. 11]



Earth Mantle Heat Convection
University of Utah

Guest Lecture:

TBA

“Wildcard” Lectures:

- Graphics hardware
- More on animation
- Motion capture
- Virtual reality and interaction
- Special effects in movies
- Video game programming
- Non-photo-realistic rendering

Hot Application Areas

- Film visual effects
- Feature animation
- Virtual reality
- PC graphics boards
- Video games
- Visualization (science, architecture, space)

Hot Research Topics

- Modeling
  - getting models from the real world
  - multi-resolution
- Animation
  - physically based simulation
  - motion capture
- Rendering:
  - more realistic: image-based modeling
  - less realistic: impressionist, pen & ink

Acknowledgments

- Jessica Hodgins (CMU)
- Frank Pfenning (CMU)
- Paul Heckbert (Nvidia)