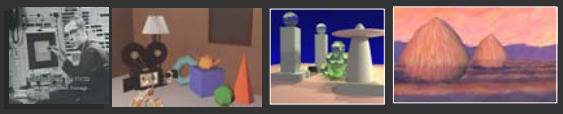
Computer Graphics (Spring 2008)

COMS 4160, Lecture 1: Overview and History

Ravi Ramamoorthi

http://www.cs.columbia.edu/~cs4160



Goals

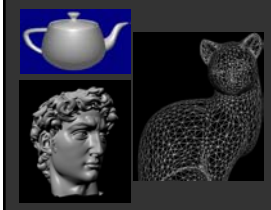
- **Systems:** Be able to write complex 3D graphics programs (real-time in OpenGL, offline raytracer)
- **Theory:** Understand mathematical aspects and algorithms underlying modern 3D graphics systems
- This course is ***not*** about the specifics of 3D graphics programs and APIs like Maya, Alias, AutoCAD, DirectX but about the concepts underlying them.

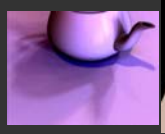
Demo: Surreal and Crazy World (HW 3)

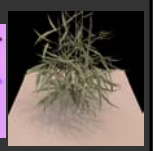
Course Outline

- 3D Graphics Pipeline

Modeling (Creating 3D Geometry) → **Rendering** (Creating, shading images from geometry, lighting, materials)







Course Outline

- 3D Graphics Pipeline

Modeling (Creating 3D Geometry) → **Rendering** (Creating, shading images from geometry, lighting, materials)

Unit 1: Transformations
Resizing and placing objects in the world. Creating perspective images.
Weeks 1 and 2
Ass 1 due Feb 14 (Demo)

Course Outline

- 3D Graphics Pipeline

Modeling (Creating 3D Geometry) → **Rendering** (Creating, shading images from geometry, lighting, materials)

Unit 1: Transformations
Weeks 1,2. Ass 1 due Feb 14

Unit 2: Spline Curves
Modeling geometric objects
Weeks 3,4
Ass 2 due Feb 26 (Demo)

Course Outline

- 3D Graphics Pipeline

Modeling (Creating 3D Geometry) → **Rendering** (Creating, shading images from geometry, lighting, materials)

Unit 1: Transformations
Weeks 1,2. Ass 1 due Feb 14

Unit 2: Spline Curves
Weeks 3,4. Ass 2 due Feb 26

Unit 3: OpenGL
Weeks 5-7.
Ass 3 due Apr 1

Midterm on units 1-3: Mar 10

Course Outline

- 3D Graphics Pipeline

Modeling (Creating 3D Geometry) → **Rendering** (Creating, shading images from geometry, lighting, materials)

Unit 1: Transformations
Weeks 1,2. Ass 1 due Feb 14

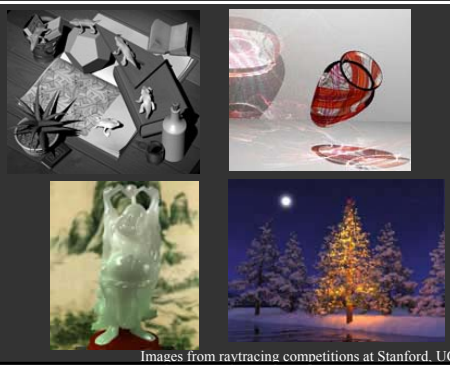
Unit 2: Spline Curves
Weeks 3,4. Ass 2 due Feb 26

Unit 3: OpenGL
Weeks 5-7.
Ass 3 due Apr 1

Unit 4: Shading, Ray Trace
Weeks 8,9.
Ass 4 due May 4

Midterm on units 1-3: Mar 10

Image Synthesis Examples



Images from raytracing competitions at Stanford, UCSD, UVa

Course Outline

- 3D Graphics Pipeline

Modeling (Creating 3D Geometry) → **Rendering** (Creating, shading images from geometry, lighting, materials)

Unit 1: Transformations
Weeks 1,2. Ass 1 due Feb 14

Unit 2: Spline Curves
Weeks 3,4. Ass 2 due Feb 26

Unit 3: OpenGL
Weeks 5-7.
Ass 3 due Apr 1

Unit 4: Lighting, Shading
Weeks 8,9.
Ass 4 due May 4
Written Ass 1 due Apr 16

Unit 5: Advanced Render
Weeks 11,12.
Written Ass 2 due May 6

Midterm on units 1-3: Mar 12

Logistics

- Website http://www1.cs.columbia.edu/~cs4160 has most of information (look at it)
- Office hours: after class (or just send me e-mail)
- TA: Ryan Overbeck, CEPSR 6LE4?
- Course bulletin board, cs4160@lists.cs.columbia.edu
- Textbook: Fundamentals of Computer Graphics by Shirley (2nd edition) , OpenGL Programming Guide 5th ed by Woo
- Website for late, collaboration policy, etc
- Questions?

Workload

- Lots of fun, rewarding but may involve significant work
- 4 programming projects; latter two are time-consuming (but you have > 1 month, groups of two, intermediate milestones). START EARLY !!
- Course will involve some understanding of mathematical, geometrical concepts taught (explicitly tested on midterm, open book take home written assignments at end)
- Prerequisites: Solid C/C++/Java programming background. Linear algebra (review on Mon) and general math skills
- Should be a difficult, but fun and generously graded course

Related courses

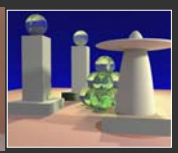
- COMS 4162, follow on to 4160 taught by me alternate years in the spring.
- Many 6000-level courses (e.g. COMS 6160 High Quality Real-Time Rendering, Representations of Visual Appearance)
- Part of Vision and Graphics track in BS and MS programs. Columbia Vision and Graphics Center
- Other related courses: Computer Vision, Robotics, User Interfaces Computational Geometry, …

To Do

- Look at website
- Various policies etc. for course. Send me e-mail if confused.
- Skim assignments if you want. All are ready
- Assignment 0, Due Jan 29 Tue (see website). Send e-mail to cs4160@lists.cs.columbia.edu telling us about yourself and sending us a digital photo (so we can put names to faces).
- Any questions?

History

- Brief history of significant developments in field
- Couple of animated shorts for fun
- Towards end of course: movie, history of CG

What is Computer Graphics?

- Anything to do with visual representations on a computer
- Includes much of 2D graphics we take for granted
- And 3D graphics modeling and rendering (focus of course)
- Auxiliary problems: Display devices, physics and math for computational problems

The term Computer Graphics was coined by William Fetter of Boeing in 1960
First graphic system in mid 1950s USAF SAGE radar data (developed MIT)

2D Graphics

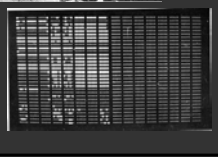
Many of the standard operations you're used to:

- Text
- Graphical User Interfaces (Windows, MacOS, ..)
- Image processing and paint programs (Photoshop, …)
- Drawing and presentation (Powerpoint, …)

How far we've come: TEXT


Manchester Mark I

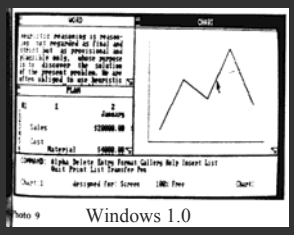
Display →



From Text to GUIs

- Invented at PARC circa 1975. Used in the Apple Macintosh, and now prevalent everywhere.


Xerox Star


Windows 1.0

Drawing: Sketchpad (1963)

- Sketchpad (Sutherland, MIT 1963)
- First interactive graphics system (VIDEO)
- Many of concepts for drawing in current systems
  - Pop up menus
  - Constraint-based drawing
  - Hierarchical Modeling

Paint Systems

- SuperPaint system: Richard Shoup, Alvy Ray Smith (PARC, 1973-79)

- Nowadays, image processing programs like Photoshop can draw, paint, edit, etc.

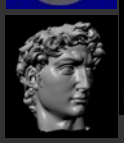
Image Processing

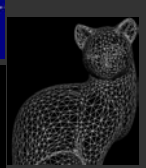
- Digitally alter images, crop, scale, composite
- Add or remove objects
- Sports broadcasts for TV (combine 2D and 3D processing)

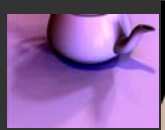
3D Graphics

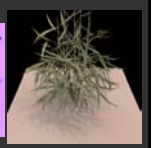
- 3D Graphics Pipeline

Modeling
(Creating 3D Geometry) → **Rendering**
(Creating, shading images from geometry, lighting, materials)



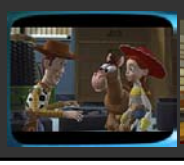






Applications

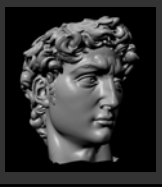
- Entertainment (Movies), Art
- Design (CAD)
- Video games
- Education, simulators, augmented reality

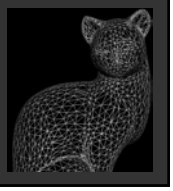


Modeling

- Spline curves, surfaces: 70^s – 80^s
- Utah teapot: Famous 3D model

- More recently: Triangle meshes often acquired from real objects





Rendering: 1960s (visibility)

- Roberts (1963), Appel (1967) - hidden-line algorithms
- Warnock (1969), Watkins (1970) - hidden-surface
- Sutherland (1974) - visibility = sorting

Images from FvDFH, Pixar's Shutterbug
Slide ideas for history of Rendering courtesy Marc Levoy

Rendering: 1970s (lighting)

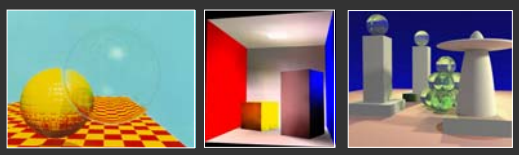
1970s - raster graphics

- Gouraud (1971) - diffuse lighting, Phong (1974) - specular lighting
- Blinn (1974) - curved surfaces, texture
- Catmull (1974) - Z-buffer hidden-surface algorithm

Rendering (1980s, 90s: Global Illumination)

early 1980s - global illumination

- Whitted (1980) - ray tracing
- Goral, Torrance et al. (1984) radiosity
- Kajiya (1986) - the rendering equation



Short Videos