Goals

- **Systems:** Write fairly complex programs for image processing, mesh algorithms, image synthesis
- **Theory:** Understand mathematical aspects and algorithms underlying modern 3D graphics

This course is a continuation of CSE 167, Introduction to Computer Graphics. It fills some gaps, provides a more advanced thorough overview.

Course Outline

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

Unit 1: Foundations of Signal and Image Processing
Understanding the way 2D images are formed and displayed, the important concepts and algorithms, and to build an image processing utility like Photoshop.
Weeks 1 – 3. Assignment 1

Unit 2: Meshes, Modeling
Weeks 3 – 5. Assignment 2

Unit 3: Advanced Rendering
Weeks 6 – 8. (Final Project)

Unit 4: Animation, Imaging
Weeks 9, 10. (Final Project)
Overview

- CSE 190, Advanced Computer Graphics
  - Prerequisite: Done well in CSE 167 or equivalent elsewhere
  - Strong interest in computer graphics
  - Advanced undergraduates, MS, PhD all welcome
  - Should count for relevant graphics/vision concentrations
  - Advanced topics in image processing, geometry, rendering, animation following on from CSE 167
  - Intended as a one quarter self-contained follow on
    - If you take only one advanced graphics course, full coverage
  - No significant overlap with CSE 168, 169
    - Image processing, meshes new material
    - Rendering coverage is real-time, image-based, not in 168

Overview

- Regular lecture class but less rigid than CSE 167
  - Advanced course, encourage class participation
  - But also need more independence, self discipline
  - Grading entirely based on 3 large programming projects
  - Can be done individually (same requirements) or group of 2
  - Given 3-4 weeks, no extensions/late days. Turn in what you have. Need to START EARLY and work steadily.
  - Homeworks usually turned in by creating website
    - Send e-mail of URL to instructor/TA to submit
    - Do not modify website after due date
    - May schedule demos
  - Encourage you to take other CS 16x, 2xx in graphics

Overview

- Workload: Challenging course: Lots of fun, rewarding but may involve significant work
  - However, given 3-4 weeks/project; work steadily
  - Grading will be generous. Assume did well in 167, here to have fun learn more graphics
  - Final project open ended (some detailed options)
  - More flexibility on pass-fail
    - Must still do one (of two) regular assignment
    - Final project can be waived given research etc.
  - Please see website for more details, assignments

Overview

- Website: http://cseweb.ucsd.edu/~ravir/190/2015/190.html
- Lectures Tu-Th 3:30-4:50pm in Soda WLH 2111
- E-mail instructor directly for questions, meetings …
  - Office hours at 10am on Mon/Tue in EBU3B, Room 4118
- Teaching Assistant: Chiwei Tseng
  c5tseng@eng.ucsd.edu
- Piazza newsgroup: please sign up
  (piazza.com/ucsd/spring2015/cse190)
- No books. Lecture slides online, reading as needed

Administrivia

- Look at website
- Various policies etc. for course. Send me e-mail if confused
- Skim assignments if you want. All are ready
- Assignment 1, due Apr 24. Start working on it immediately (START EARLY). For today, make sure download/compile
- Find partners for assignment 1 and possibly later (can switch partners or switch individual/group of two between assignments if you want). Tell instructor/TA if need help
- Questions?

To Do

- Brief history of significant developments in field
- End with a video showcasing graphics

History

- 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

From Text to GUIs

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

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


- 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)
Relevance to Course

- In 167, didn’t focus on 2D
- But relevant broadly (not just for 2D), since ultimately 3D scene displayed as 2D image
- In 190, we cover image processing and many photoshop functions [assign. 1 to write a mini-version]

Geometry

- Spline curves, surfaces: 70s – 80s
- Utah teapot: Famous 3D model
- More recently: Triangle meshes often acquired from real objects

Progressive Mesh Simplification

- Coarse mesh + subdivision rule
- Smooth surface = limit of sequence of refinements

Subdivision Surfaces

- Unit 2 is about mesh processing algs.
- Will learn to represent, work with meshes
- Do mesh simplification, progressive meshes for assignment 2

Rendering and Appearance

- Core area in computer graphics
- Efficiently and easily create visual appearance
- Long history (1960s to current time): Variety of old and new topics
- From basic visibility and shading, to global illumination, to image-based rendering, to data-driven appearance and light fields
- Many links to physics, math, computer science
- We focus on real-time, image-based (no overlap with 168 that focuses on basic offline rendering)
Rendering: 1960s (visibility)
- Roberts (1963), Appel (1967) - hidden-line algorithms
- Sutherland (1974) - visibility = sorting

Rendering: 1970s (lighting)
- 1970s - raster graphics
  - Blinn (1974) - curved surfaces, texture

Rendering (1980s, 90s: Global Illumination)
- early 1980s - global illumination
  - Whitted (1980) - ray tracing
  - Goral, Torrance et al. (1984) radiosity
  - Kajiya (1986) - the rendering equation
  - This is basically what 168 covers

Image-Based Rendering
- Apple’s QuickTime VR
  - Outward
  - Inward

Dual Interpretation of Light Field
- Plenoptic Light Field
  - Field radiance
- Surface Light Field
  - Surface radiance
- UV Array of ST Images
- ST Array of UV Images

Lytro Light Field Camera
Acquiring Reflectance Field of Human Face [Debevec et al., SIGGRAPH 00]

Illuminate subject from many incident directions

Example Images

Images from Debevec et al., 00

Precomputed Radiance Transfer

- Better light integration and transport
  - dynamic, area lights
  - self-shadowing
  - interreflections
- For diffuse and glossy surfaces
- At real-time rates
- Sloan et al. 02

Precomputation: Spherical Harmonics

Basis 16
Basis 17
Basis 18

Diffuse Transfer Results

No Shadows/Inter
Shadows
Shadows+Inter

Arbitrary BRDF Results

Anisotropic BRDFs
Other BRDFs
Spatially Varying
Imaging

- Processing of images important part of graphics
- Especially in context of photography: Combine photos, manipulate images
  - Recent video on automatic cinemagraph portraits
- Computational photography. Examples flash/no-flash, fluttered shutter, new light field cameras
- Community and Internet photo collections
- Basic ideas like HDR and Texture Synthesis

High Dynamic Range

- Photographs at multiple exposures
- Combine and tonemap

From Wikipedia,Debevec and Malik 97

Multiple Photographs

Combined and Tonemapped

Texture Synthesis

- From small image to larger (keep texture)
- Novel idea: Copy image patches (quilting)

History of Computer Animation

- 10 min clip from video on history of animation
  - http://www.youtube.com/watch?v=LzZwiLUVaKg
- Covers sketchpad, animation, basic modeling, rendering
- A synopsis of what this course is about
- (watch offline if short on time)
Summary

- Graphics is Modeling/Geometry, Rendering, Animation/Simulation, Imaging and much more
- Course looks at all of these. One stop follow on to CSE 167, no overlap with 168 or 169
- 3 programming assignments (groups of 2)
  - Image Processing
  - Progressive Meshes
  - Project (eg Real-Time / Image-Based Rendering)