Final Projects

CS114, Spring 2018

Shuang Zhao

Computer Science Department

University of California, Irvine

(Multiple slides courtesy to Steve Marschner)

Final Project Rules

- **Group size:** 2 to 4 students
  - Choose your own groups (utilizing Piazza)
  - Expected scope is larger with more people
- **Examples:**
  - A cool game (with a focus on graphics)
  - Implementation of advanced rendering/animation algorithms
- **Deliverables:**
  - Project proposal (due May 21)
  - Milestone presentation (early June)
  - Final presentation (during the final's week in June)

What Makes for Interesting Graphics?

- **Rendering**
  - Fancy shading/reflectance models
  - Translucency
  - Environment illumination
- **Animation**
  - Collision detection
  - Particle systems for smoke, fire, explosions, etc.
  - Procedurally animated water, wind, etc.

What Makes for Interesting Graphics?

- **Modeling**
  - Subdivision surfaces
  - Voxelized terrain
  - Procedural models (trees, cities, etc.)

- **Complexity Management**
  - Acceleration structures: Kd-trees, octrees
  - Level-of-detail techniques

Overlap with Other Projects

- In general, it is okay to build upon your own earlier work
  - But you need to talk to me about it!

- You have to disclose overlaps
  - Work that comes from projects you did for other courses
  - Work that comes from your own personal projects before this course
  - Submitting overlapping work without saying anything is *dishonest*

Final Project Proposal

- One-two page description of your project
  - Say what constitutes the technical “meat”
  - Tentative schedule with allocation of team-members to tasks
- Major areas of focus
  - One primary area for small groups
  - Two for large groups

Project Requirements

- Must go **significantly** beyond Projects 1--3
  - Combine multiple techniques in interesting ways
  - Implement significant new techniques

- **Quality** product expected:
  - Nice imagery
  - Correct implementations (demonstrated with experimental results)
  - How you achieve results is as important as the results themselves

Code Base

- Pick whatever code base you want
  - Build on codebase from Projects 1--3
  - Start from scratch
  - C++/Java/WebGL
  - …
- However, you need to implement key technical components yourself (instead of directly using existing libraries)

Resources

- Get 3D models off the web
  - E.g., www.turbosquid.com
  - Do NOT spend too much time modeling a person or an object
- Articles referenced in lecture
- Piazza, office hours