Lecture 1:
Introduction and Overview
Welcome!
Who Am I?

• Lingqi Yan
  - Pronunciation: ling—chi—yen
  - Assistant Professor
  - Web: www.cs.ucsb.edu/~lingqi/
    Email: lingqi@cs.ucsb.edu
  - Research: Computer Graphics — rendering — photorealism & speed
  - Hobbies: video games, piano and NBA

• What about you?
What is CS291A about?

Real-Time High Quality Rendering
What is CS291A about?

- Real-Time High Quality Rendering
  - What is Rendering?

3D scene (meshes, lights, etc.) → Image

Calculating light -> eye
What is CS291A about?

• So, we will not cover 3D modeling or game development using Unreal Engine (where can I learn them?)

Modeling character animation in Maya  

CSGO PoV Cam set up in Unreal Engine  
[https://www.youtube.com/watch?v=3TQ18SmQSv0]
What is CS291A about?

• And we will not cover physically-based animation/simulation (where can I learn this?)

Adaptive Anisotropic Remeshing for Cloth Simulation, Narain et al.
What is CS291A about?

- **Real-Time High Quality Rendering**
  - Speed: more than 30 FPS (frames per second), even more for Virtual / Augmented Reality (VR / AR): 90 FPS
  - Interactivity: Each frame generated on the fly

- **Real-Time High Quality Rendering**
  - Realism: advanced approaches to make rendering more realistic
  - Dependability: all-time correctness, no tolerance to (uncontrollable) failures
What is CS291A about?

- So, we will **not** cover expensive (but more accurate) light transport techniques in movies / animations (where can I learn this?)

Manifold Metropolis Light Transport
Jakob et al.

Gradient Domain Path Tracing
Kettunen et al.
What is CS291A about?

- And we will **not** cover Computer Vision / Deep Learning topics, e.g. XYZ-GAN (where can I learn this?)

GAN 2.0: NVIDIA’s Hyperrealistic Face Generator (both are fake)
Course Topics
Course Topics

• Shadow and Environment Mapping

Real-Time, All-Frequency Shadows in Dynamic Scenes
Annen et al.
Course Topics

• Interactive Global Illumination Techniques

Micro-Rendering for Scalable, Parallel Final Gathering
Ritschel et al.
Course Topics

- Precomputed Radiance Transfer

Global Illumination with Radiance Regression Functions
Ren et al.
Course Topics

- Sampling and Reconstruction

Fast 4D Sheared Filtering for Interactive Rendering of Distribution Effects
Yan et al.
Course Topics

- Real-Time Ray Tracing
Course Topics

- Image-based Rendering and Light Fields

Composition with input image and textures of virtual objects

Filtering Environment Illumination for Interactive Physically-Based Rendering in Mixed Reality
Mehta et al.
Course Topics

• Participating Media Rendering, Image Space Effects, etc.

Multiple scattering

Image space reflection
Questions?
Today’s Lecture
Outline

• Motivation

• Evolution of real-time rendering

• Technological and algorithmic milestones
  - Programmable graphics hardware
  - Precomputation-based methods
  - Interactive Ray Tracing

• Course Logistics

• Project 0
Motivation

• Today, Computer Graphics is able to generate **photorealistic** images
  - Complex geometry, lighting, materials, shadows
  - Computer-generated movies/special effects (difficult or impossible to tell real from rendered…)

[Artist: Teruyuki and Yuka]  [Artist: Hyun Kyung]
Motivation

- But accurate algorithms (esp. ray tracing) are very slow
  - So they are called offline rendering methods
  - Guess how long does it take to render one frame in Zootopia?
Motivation

- With proper approximations, we can generate plausible results but runs much faster

Toyota 2000GT, from TurboSquid (offline rendering)

Need for Speed: Payback (real-time rendering)
Evolution of Real-Time Rendering

- Interactive 3D graphics pipeline as in OpenGL
  - Earliest SGI machines (Clark 82) to today
  - Most of focus on more geometry, texture mapping
  - Some tweaks for realism (shadow mapping, accum. buffer)

SGI Reality Engine 93
(Kurt Akeley)
Evolution of Real-Time Rendering

• 20 years ago
  - Interactive 3D geometry with simple texture mapping, fake shadows (OpenGL, DirectX)

Final Fantasy VII (1997)

Counter Strike (1999)
Evolution of Real-Time Rendering

• 20 -> 10 years ago
  - A giant leap since the emergence of programmable shaders (2000)
  - Complex environment lighting, real materials (velvet, satin, paints), soft shadows

Assassin’s Creed II (2009)  
Resident Evil 5 (2009)
Evolution of Real-Time Rendering

• Today
  - Extended to Virtual Reality (VR) and even movies
  - “Stunning graphics”

Beat Saber, VR Game

Zafari, animation series rendered completely using Unreal game engine
Evolution of Real-Time Rendering

• Today
Evolution of Real-Time Rendering

• Today
Evolution of Real-Time Rendering

- In the future

The Matrix (1999 movie)
Evolution of Real-Time Rendering

- In the future

Ready Player One (2018 movie)
Technological and Algorithmic Milestones

- Programmable graphics hardware (shaders) (20 years ago)

A New Dawn demo, NVIDIA
https://www.geforce.com/games-applications(pc-applications/a-new-dawn/videos)
Technological and Algorithmic Milestones

- Programmable graphics hardware (shaders)
Technological and Algorithmic Milestones

• Precomputation-based methods (15 years ago)
  - Complex visual effects are (partially) pre-computed
  - Minimum rendering cost at run time
Technological and Algorithmic Milestones

- Precomputation-based methods

All-Frequency Rendering of Dynamic, Spatially-Varying Reflectance
Wang et al.
Technological and Algorithmic Milestones

• Precomputation-based methods: Relighting
  - Fix geometry
  - Fix viewpoint
  - Dynamically change lighting

[Ng, Ramamoorthi, Hanrahan 04]
Technological and Algorithmic Milestones

• Interactive Ray Tracing (8-10 years ago: CUDA + OptiX)
  - Hardware development allows ray tracing on GPUs at low sampling rates (~1 samples per pixel (SPP))
  - Followed by post processing to denoise

Car interactively rendered using NVIDIA OptiX
Pixar’s real-time previewer
Technological and Algorithmic Milestones

• What do you think [is / will be] the milestone of CG today and in the future?
  - Deep learning?
  - Real-Time Ray Tracing?
  - VR with realistic graphics?

• Let’s re-look at this slide several years later!
Questions?
Course Logistics
Class Philosophy

• We want a very active class

• Come to class

• Follow, think, discuss and question
Prerequisites

- **Strong interest** in graphics, rendering

- Computer graphics experience (CS180 or equivalent)
  - A brief survey
  - What if lacking prerequisites? Next slide

- Course will move quickly
  - Covering recent and current active research
  - Some material quite technical
  - Many topics.
    
    Need not to fully follow each one, but doing so will be most rewarding.
If Lacking Prerequisites

• You have this week to catch up with basic OpenGL (and general Computer Graphics) knowledge this week

• The next lecture will briefly review related topics

• Project 0 will help you warm up quickly

• Practice makes perfect!
Course Logistics

• Course Website
  - Everything’s there!

• No required textbooks
  - Related papers will available online before lectures
  - Lecture slides will be available right after class
  - The book “Real-Time Rendering (3rd ed or later)” by Moller and Haines may be helpful (we will not follow it, though)
Course Logistics

• No TA for this class
  - Sign up on Piazza for student discussion

• Office hour
  - Tuesdays 1PM - 2PM, HFH 5102
  - Please don’t ask me to debug…

• Academic integrity
  - Work alone except for the final project (no copy-pasting from others)
  - Do not publish your code (on Github, etc.)
  - Other details on the website, strictly enforced
Assignments and Grading

• Assignments
  - One paper presentation (15%)
  - Projects 0 - 3 (15% each)
  - Final project (25%, in groups of two)
  - NO EXAMS!

• Grading
  - Submit your project by 11:59PM on/before the due dates via Gauchospace
  - Each late day = 10% off
Assignments and Grading

• More words about projects
  - Getting a “working example” is the most important
  - Minimize distractions w.r.t. C++ syntax and OpenGL usage, and focus on core shader implementations
  - Software engineering is not necessary
  - Hard code is definitely acceptable
  - START EARLY!
Questions?
Announcement

- Project 0 is out today

- Next lecture: recap some important concepts
  - Hardware Graphics Pipeline
  - Shader Language
  - Radiometry
  - Rendering Equation
  - etc.
Thank you!