

Real-Time High Quality Rendering

CSE 274 [Fall 2015], Lecture 1

Ravi Ramamoorthi

<http://www.cs.ucsd.edu/~ravir>



Motivation

- Today, create photorealistic computer graphics
 - Complex geometry, lighting, materials, shadows
 - Computer-generated movies/special effects (difficult or impossible to tell real from rendered...)



- CSE 168 images from rendering competition (2011)
- But algorithms are very slow (hours to days)**

Real-Time Rendering

- Goal: interactive rendering. Critical in many apps
 - Games, visualization, computer-aided design, ...
- Until 10-15 years ago, focus on complex geometry



- Chasm between interactivity, realism**

Evolution of 3D graphics 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)



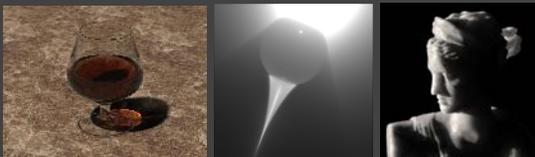
SGI Reality Engine 93
(Kurt Akeley)

Offline 3D Graphics Rendering

Ray tracing, radiosity, photon mapping

- High realism (global illum, shadows, refraction, lighting...)
- But historically very slow techniques

"So, while you and your children's children are waiting for ray tracing to take over the world, what do you do in the meantime?" Real-Time Rendering



Pictures courtesy Henrik Wann Jensen

15 years ago

- High quality rendering: ray tracing, global illumination
 - Little change in CSE 168 syllabus, from 2003 to today
- Real-Time rendering: Interactive 3D geometry with simple texture mapping, fake shadows (OpenGL, DirectX)
- Complex environment lighting, real materials (velvet, satin, paints), soft shadows, caustics often omitted in both
- No CSE 274: Realism, interactivity at cross purposes

Today: Real-Time Game Renderings



Why Still Teach This Course?

- Previously taught this course 11 years ago at Columbia (previous slide then said 5 years ago)
- Many key developments in last 11 years
 - Real-Time ray-tracing practical, hardware (Optix)
 - Programmable shaders even on your mobile phone
 - Massive increase in computational power (but mobile devices are more prominent, have less capability)
 - New algorithms: precomputation, fast global illum.
- Revolution over past decade since first taught
 - High quality real-time rendering practical, used
 - But true photorealism in real time still a few years away; that is the revolution we're pushing towards*

Outline of Lecture

- Background
- Motivation, effects for high quality real-time rendering*
- Recent technological, algorithmic developments
- Preview of results currently possible
- Logistics of course

Given need to teach the course step by step, many parts are same as 11 years ago, but also newer developments

High quality real-time rendering

- Photorealism, not just more polygons
- Natural lighting, materials, shadows



Interiors by architect Frank Gehry. Note rich lighting, ranging from localized sources to reflections off vast sheets of glass.

High quality real-time rendering

- Photorealism, not just more polygons
- Natural lighting, materials, shadows



Glass Vase

Glass Star (courtesy Intel)

Peacock feather

Real materials diverse and not easy to represent by simple parametric models. Want to support measured reflectance.

High quality real-time rendering

- Photorealism, not just more polygons
- Natural lighting, materials, shadows



small area light, sharp shadows
Agrawala et al. 00

soft and hard shadows
Ng et al. 03

Natural lighting creates a mix of soft diffuse and hard shadows.

Today: Full Global Illumination



Applications

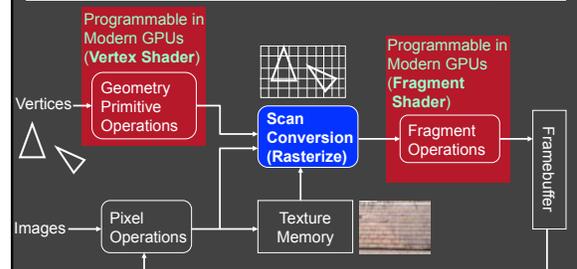
- Entertainment: Lighting design
- Architectural visualization
- Material design: Automobile industry
- Realistic Video games
- Electronic commerce



Outline of Lecture

- Background
- Motivation, effects for high quality real-time rendering
- Recent technological, algorithmic developments
- Preview of results currently possible
- Logistics of course

GPU Programmable Shaders



Traditional Approach: Fixed function pipeline (state machine)
New Development (2003-): Programmable pipeline

CPU and GPU Algorithms

- Vast increase in CPU power, modern instrs (SSE, SIMD, ...)
 - Real-time ray-tracing techniques are possible
 - Even on GPU (NVIDIA's OptiX)
 - Now used routinely in games etc.
 - <https://www.youtube.com/watch?v=h5mRREIXy-w>
- New classes of algorithms
 - Precomputation-Based methods (will study in course)
 - Real-Time Global Illumination techniques
 - Sparse Sampling and Filtering (will study in course)
- Goals for CSE 274
 - Understand basic ideas in high-quality real-time rendering (~2005)
 - Introduce newer concepts and goals of full photorealism (~present)
 - Not course about real-time rendering with complex geometry. Primarily about high-quality shading effects

Outline of Lecture

- Background
- Motivation, effects for high quality real-time rendering
- Recent technological, algorithmic developments
- Preview of results currently possible
 - Programmable graphics hardware
 - Precomputation-based methods
 - Interactive RayTracing
- Logistics of course

Programmable Graphics Hardware



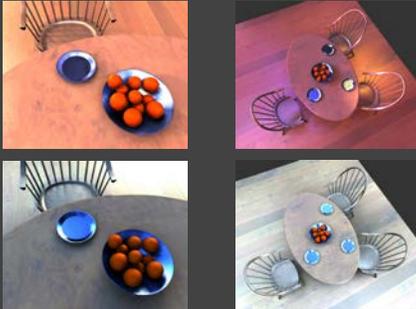
NVIDIA a new dawn demo
<http://www.geforce.com/games-applications/pc-applications/a-new-dawn/videos>

Precomputation-Based Methods

- Static geometry
- Precomputation
- Real-Time Rendering (relight all-frequency effects)
- Involves sophisticated representations, algorithms

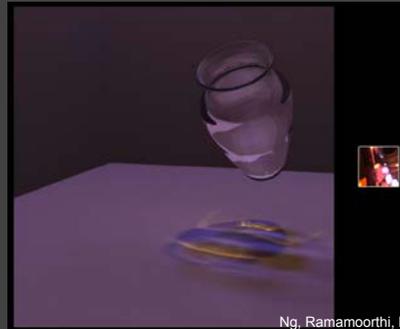


Relit Images



Ng, Ramamoorthi, Hanrahan 04

Demo: Real-Time Relighting



Ng, Ramamoorthi, Hanrahan 03

Spherical Harmonic Lighting



Avatar 2010, based on Ramamoorthi and Hanrahan 01, Sloan 02

Interactive RayTracing

Advantages

- Very complex scenes relatively easy (hierarchical bbox)
- Complex materials and shading for free
- Easy to add global illumination, specularities etc.

Disadvantages

- Hard to access data in memory-coherent way
- Many samples for complex lighting and materials
- Global illumination possible but expensive

Modern developments: Leverage power of modern CPUs, develop cache-aware, parallel implementations

<http://www.geforce.com/games-applications/pc-applications/design-garage/videos>

Project

- Wide flexibility if related to course. Can be done groups of 2
 - Default: Implement (part of) one of papers and produce an impressive real-time high quality rendering demo
 - See/e-mail me re ideas
 - Best projects will go beyond simple implementation (try something new, some extensions)
- Alternative (less desirable): Summary of 3+ papers in an area
 - Best projects will explore links/framework not discussed by authors, and suggest future research directions

Prerequisites

- Strong interest in graphics, rendering
- Computer graphics experience (167 or equivalent)
 - What if lacking prerequisites? Next slide
 - Experience with rendering (CSE 168) not required
- Course will move quickly
 - Covering recent and current active research
 - Some material quite technical
 - Assume some basic knowledge
 - Many topics. Needn't fully follow each one, but doing so will be most rewarding.

If in doubt/Lack prerequisites

- Material is deep, not broad
 - May be able to pick up background quickly
 - Course requirements need you to really fully understand only one/two areas (topics)
 - But if completely lost, won't be much fun
- If in doubt, see if you can more or less follow some of papers after background reading
- Ultimately, your call

Assignment this week

- E-mail me (ravir@cs.ucsd.edu)
 - Name, e-mail, status (Senior, PhD etc.)
 - Will you be taking course grades or P/F
 - Background in graphics/any special comments
 - Optional: Papers you'd like to present FCFS (only those that say "presented by students")
- Paper presenters for Oct 6 [You (may) get a one-paper reduction in load] (shadow and environment mapping)
 - L. Williams: Casting curved shadows on curved surfaces 78
 - T. Lokovic and E. Veach: Deep Shadow Maps 00
 - B. Cabral et al. Reflection Space Image-Based Rendering 99

Questions?
