cs348b: Image Synthesis Techniques
CS348b: Image Synthesis

Goal: How to generate photorealistic images?

Applications
- Movies
- Games
- Industrial design
- Architecture
- Commerce
- Cultural heritage
- Simulators (ML) and trainers

War for the Planet of the Apes (2017)
The Abyss (1989)
Jurassic Park (1993)
Bunny (1998)
Architecture: Impulse Arts, “Wochendhaus”
Simulation / ML Training: Aurora
Physically-Based Rendering
Modeling and Simulating the World
Titanic (1997)
War for the Planet of the Apes (2017)
NVIDIA Marbles At Night (2021)
Ray Tracing and Path Tracing
Ray Tracked Visibility and Shadows
Whitted Ray Tracing

Whitted, An Improved Illumination Model For Shaded Display, 1979.
Whitted Ray Tracing
Whitted Ray Tracing

Recursive Expression:
Whitted(ray) {
    L = 0;
    if (IntersectClosest(ray)) {
        if (SurfaceIsSpecular)
            return Whitted(specularRay);
        else if (LightIsVisible)
            L += scatteredIllumination;
    }
    return L;
}
The Rendering Equation

\[ L_o(\omega_o) = L_e(\omega_o) + \int_\Omega f_r(\omega_i \rightarrow \omega_o) L_i(\omega_i) \cos \theta_i \, d\omega_i \]
Recursive Expression:

PathTrace(ray) {
    L = 0;
    if (IntersectClosest(ray)) {
        L += surfaceEmission;
        ray = randomDirection();
        L += reflection * PathTrace(ray);
    }
    return L;
}
Random Walk—Path Tracing
Rendering as Physical Simulation

Models

- Light
- Light sources
- Materials and shapes
  - Surfaces: Reflection and texture models
  - Media: Atmospheric scattering models
- Cameras

Simulation

- Light transport algorithms
  - Efficient ray-surface queries for many shapes
  - Sampling ray paths using Monte Carlo integration
Measuring Light and Reflection
Photographic Images

We have made high-quality pictures of the Cornell box in its current configuration. The liquid-cooled Photometrics PXL1300L CCD camera has a precision of 12 bits. We used 7 narrow-band filters to get a coarse sampling across the visible spectrum. Dark current has been subtracted from the images, and flat-field correction has been applied to account for the cosine fall-off and the lens fall-off. The following data are available:

- **IPLab images** The images are collected in a compressed tar file (10 MB compressed, 18 MB uncompressed!). They are in IPLab format, which can be read using a simple Matlab function `iplab_read`.  
- **TIFF images** The images are available in TIFF files, with 16 bits per pixel. This format can be read by a number of applications, including Photoshop. There is a choice of either a compressed tar archive (about 10MB compressed, 18MB uncompressed) or a ZIP archive (about 10MB).  
- **OpenEXR images** The images are also available in the OpenEXR high-dynamic-range image format. Each pixel is represented as a 15-bit floating-point number with a 10-bit mantissa and 5-bit exponent. There is a choice of either an uncompressed tar archive (about 8MB, gzip compression shrinks it by only 0.4%) or a ZIP archive (about 8MB).  
- **Transmission spectra** of the 7 corresponding filters, the lens system, and the response spectrum of the camera are available from our page with measurement data.

Synthetic Images

A synthetic image file, generated from the data below, is available in two formats:

- As a .jpeg file
- As an .rge file. RGBE is a floating point image format for which the specifications and I/O functions are available.

Scene Data

The reflectance data and the geometric data for the Cornell Box (with diffuse objects) are presented below. The data are also available in two other formats:

- As an .mdla file. MDLA is a format defined at Cornell for scene descriptions, with specifications that are easily extendible.
- As an .inventor file. The RGB reflectance values in this file are for previewing purposes only; they do not correspond to the real spectral data.

http://www.graphics.cornell.edu/online/box/
Which is Real?
Which is Real?

Photograph

Rendering
Data-Driven Rendering

EPFL Goniophotometer
Data-Driven Rendering

EPFL RGL Material Database
Data-Driven Rendering

Greg Zaal: https://polyhaven.com/a/rathaus
Data-Driven Rendering?

Gaugan [Park et al. 2019]
Algorithms and Architectures
Full Movie: 100 million hours of CPU time!

29 hours/frame = 29*60*60 seconds/frame = 104,400 seconds/frame

100K seconds/frame * 10 GFLOPs = ~1 PFLOP

1 PFLOP / 1 Megapixel = ~ 1 GFLOP / pixel
Blinn’s Law

“As technology advances, rendering time remains constant”

Voyager 2 Saturn Flyby (1981)
Blinn’s Law in Action

Monsters, Inc. (2001)
Blinn’s Law in Action

Monsters University  (2013)
Blinn’s Law in Action

Monsters, Inc. (2001)
Blinn’s Law in Action

Monsters University (2013)
Modeling and Simulating the World...

With a Computational Budget
Interdisciplinary Topics

Computer science
- Computational geometry
- Computer architecture

Physics
- Bidirectional reflectance distribution functions
- Radiometry and light fields
- Radiative transport

Mathematics
- Integral equations
- Monte Carlo methods

Perception

Art and design
Course Mechanics
Logistics

Slide comments (10%)
- 2 per lecture, due before next lecture

4 Assignments (10% each)
- Lighting design
- Sphere tracing distance estimators
- Light field cameras
- Monte Carlo sampling

Final project (50%)
- Reproduce a scene or image
- Explore advanced rendering algorithms
Final Projects
The Rendering Competition
Artistic Art Glass
Georg Petschnigg and Inam Ur-Rahman Malik
Jellyfish

Kayvon Fatahalian and T. Foley
Lily Pads
Tom Brow and Ranjitha Kumar
Cotton Candy
Chenlin Meng, Hubert Teo, Jiren Zhu
Summertime Iridescence
Jennifer Tao and Cynthia Jia