Advanced Ray Tracing

Paper Summaries
• Any takers?

Assignments
• Checkpoint 5
  – Due Wednesday
• Checkpoint 6
  – To be given Wednesday
• RenderMan
  – Due May 3rd

Projects
• Project feedback
• Approx 22 projects
• Listing of projects now on Web
• Presentation schedule
  – Presentations (15 min max)
  – Last 3 classes (week 10 + finals week)
  – Sign up
    • Email me with 1st, 2nd, 3rd choices
    • First come first served.

Projects
• Exam day presentations
  – Tuesday, May 18th / 8-10am
    • or
  – Wednesday, May 19th / 10:15-12:15

Computer Graphics as Virtual Photography

Photography:
- real scene
- camera (captures light)
- photo processing
- Photographic print

Computer Graphics:
- 3D models
- camera model (focuses simulated lighting)
- tone reproduction
- synthetic image
Ray Tracing

- Integrated aspects of light and object interaction that had formerly been handled by separate algorithms:
  - Hidden surface removal
  - Reflection
  - Refraction
  - Shadows
  - Global specular interaction

Turner Whitted

Ray Tracing - Problems

- Object - ray intersection
- Ray traced images are point sampled
  - “Too sharp” (super real) - “wrong image”
  - Sharp shadows
  - Sharp Reflection/Refraction
  - Multiple reflections especially are too sharp
  - Aliasing
- Doesn’t handle major light transport functions
  - Diffuse interaction
  - Scattering of light
  - Caustics
- Computation time

Ray Tracing – Avoiding Ray Traced Look

- Avoiding that “ray traced look”, i.e., handle diffuse interaction
  - Ray tracing is point sampling
    - 1 ray per pixel
      - Assumes pixel is a single point
      - Assumes pin hole camera
    - 1 ray for transmission & reflection
      - Assumes all reflection is specular
      - Assumes that BRDF for all incoming directions is single out going direction

Ray Tracing – Avoiding Ray Traced Look

- “Backward”/ “Reverse” Ray Tracing
  - Three approaches
    - “Backward” ray tracing
    - Trace objects other than rays
    - Stochastic sampling

“Backward”/ “Reverse” Ray Tracing

This approach is (sort of) used in Photon Mapping also – later…
Tracing things other than rays

- Beam tracing – pyramidal beams
  - Recursively applies Weiler-Atherton hidden surface removal algorithm
  - Polygonal surfaces only
- Cone tracing – cones
- Pencil Tracing – bundle of rays
  - Paraxial rays -> 4D vectors
  - Requires smooth surfaces

[Heckbert84]

Tracing things other than rays

- Same as traditional ray tracing except
  - Intersection is a surface
  - Surfaces spawn more cones, beams, pencils
  - Advantage: More realistic
  - Disadvantage: Increase in complexity of geometric calculations
    - e.g., in cone tracing, calculations not only involve an intersection, but also the ratio between the cross-section of beam and area of intersection of object is needed

Cone Tracing Examples

http://www.cs.uaf.edu/~genetti/

Note the fuzzy reflections and soft shadows

Stochastic Ray Tracing

- Introduce randomness in ray spawning
- Kajiya86 used for solving rendering equation
  - Either reflection or refraction is spawned
  - Reflection rays can be spawned in diffuse direction
  - Multiple “paths” per pixel
Distributed Ray Tracing

- Unified
  - Blurred reflections
  - Blurred refraction
  - Soft shadows
  - Depth of field
  - Motion blur
- Uses stochastic sampling to get rid of ray traced look.
- Uses ray bundle, but size limited stochastically

Distributed Ray Tracing

- “Jitter” sampling
  - Trades aliasing for noise!
  - Use Poisson “noise” to jitter values from original “fixed” position
  - Using “jittering” in
    - Initial ray generation
    - Reflection/transmission ray generation
    - Shadow ray generation
    - Jitter over time.

Distributed Ray Tracing

- Initial ray generation
  - *Supersample* - instead of 1 ray per pixel, shoot 16 rays per pixel.
  - Initial ray positions are not evenly spaced, rather distributed stochastically within pixel using “jittering”

Distributed Ray Tracing

- Jittered ray generation
  - Using a lens/camera model
  - Simulation of depth of field
  - First use of non-pinhole camera model in ray tracing
Distributed Ray Tracing

Depth of Field

Notice different size circles of confusion caused by different sized lens

• Jittered ray generation

Distributed Ray Tracing

• Jittered Reflection
  – Send out multiple rays jittered about the real reflection direction
  – Contribution of each ray to intensity weighted by a predefined “importance” function - associated with the object’s reflectance properties

Distributed Ray Tracing

• Jittered Reflection

Jittered Reflection

1 reflection ray 10 reflection rays 20 reflection rays 50 reflection rays

Allen Martin, WPI
Jittered Transmission

- 1 transmission ray
- 10 transmission rays
- 20 transmission rays

Distributed Ray Tracing

- Jittered shadow rays
  - Multiple jittered rays sent out toward light sources
  - Contribution of each determined by a predefined importance function associated with the light
  - Shadow rays return fraction of light seen rather than yes/no
  - Results in soft shadows

Distributed Ray Tracing

Soft Shadows

[Cook84]

Distributed Ray Tracing

Jittering in time
  - Do ray tracing over a duration of time in which objects may move.
  - Uses same rays repeatedly over time interval
  - Results in motion blur

Distributed Ray Tracing

Motion Blur

[Cook84]
Distributed Ray Tracing - Summary

- Stochastic sampling ("jitter sampling")
- Apply jittering to:
  - Initial ray selection
    - Jittering on image plane + use of camera/lens model
  - Reflection / Transmittance
  - Shadow Rays
  - Over time

Photon Mapping - Motivation

- Combines “backward”/ “reverse” ray tracing with stochastic ray tracing
- Used to simulate the interaction of light with a variety transparent substances (caustics)
  - Glass
  - Water
  - Diffuse Inter-reflections between illuminated objects
  - Effects of particulate matter
    - Smoke
    - Water vapor

Photon Mapping

- Henrik Wann Jensen 95/96
- Simulates the transport of individual photons emitted from light sources
- Photons bounce off specular surfaces
- Photons deposited on diffuse surfaces
- Photons collected by ray tracing from eye

Photon Mapping - Caustics

- Pattern of light focused on a surface after having original light path bent by intermediate surface.
- For example, a glass of wine on a table changes the pattern and the color of light

Photon Mapping

What is a Photon?

- A photon $p$ is a particle of light that carries flux $\Delta \Phi_p (\mathbf{x}_p, \omega_p)$.
  - Power: $\Delta \Phi_p$ - magnitude (in Watts)
  - and color of the flux it carries, stored as an RGB triple
  - Position: $\mathbf{x}_p$ - location of the photon
  - Direction: $\omega_p$ - the incident direction used to compute irradiance
- Photons vs. rays
  - Photons propagate flux
  - Rays gather radiance
Photon Mapping Algorithm

Photon Map

- **Photon tracing**
  - Light packets sent from light sources
  - When intersect surface, store in *photon map*:
    - 3D coordinate of intersection
    - Incoming direction
    - Energy
    - Energy absorbed on bounces and refraction
  - Similar to reverse ray tracing
- **Rendering**
  - Ray tracing for direct illumination
  - Object intersected by ray
    - Check if in range of stored photons – yes, add energy
  - Photon map visualization
    - Indirect bounce

- **A photon map** is a data structure that stores all distributed photons
- Often three photon maps are used
  - One for caustics
  - One for indirect illumination
  - One for volume caustics
- For efficiency, photon direction is often constrained
  - Only sent in direction of objects known to cause caustics

Photon Mapping - Summary

- Combines “backward”/“reverse” ray tracing with stochastic ray tracing to generate caustics
- Cost effective compared to other methods

The Light of Mies van der Rohe

Parallel Computing

- **What is it?**
  - Solving a problem by dividing it into tasks to be handled by separate processors
- **Why use it?**
  - Performance
  - Goal linear speedup

Parallel Hardware

- **SIMD**
  - Same instruction executed in lockstep fashion on different data
- **MIMD**
  - Different instructions executed on different data
    - MIMD-SM
    - MIMD-DM
Parallel vs. Distributed

Issues in Parallel Computing: Communication

- More processors means more organization is required (Consider working on a team of 10 people vs. 1000 people)
- For performance, each processor must have a lot of work to do in comparison with any need to communicate to another processor.
- Keep all processors busy (load balancing)

RIT Ray Tracing 1991

- 975 objects
- 3 light sources
- 2048 X 2048 pixels
- Sun 4/490 – 1991
- 1.5 hours CPU time

RIT Fractals and Ray Tracing 1991

- 87,000 objects
- 5 light sources
- 1024 X 1024
- Sun 4/490 – 1991
- 18 hours CPU time

Ray Tracing

- Ray tracing lends itself nicely to parallel programming because the work can easily be sub-divided
  - Per pixel calculation, i.e., not dependent on other pixels
  - Potentially large calculation time
- How to distribute work?

Image Space Partitioning

Strip = Section of Viewing Plane per Processor
Data Parallel strip division

Object Space Partitioning

Voxel = Section of scene data for processor

Object Partitioning

Reinhard’s Parallel Example
Adaptive Partitioning to Load Balance

Partitioning for Ray Tracing

- Object Space Partitioning
  - May be necessary if very complex scene!
  - Less local processor memory needed for scene data
  - High delay when passing rays between processors means crossing machines
- Object Partitioning
  - Same as object space (just divided differently)
- Image Space Partitioning
  - Each local processor must have complete scene data
  - Messages are only sent to divide and reassemble, not while doing the work
Partitioning for Parallel Ray Tracing

- Image space partitioning is the best choice
- When might we use the others?
  - Object Space Partitioning: when there is not enough local processor memory for the scene data
  - Object Partitioning: same as above when object complexity is similar

Advanced Ray Tracing

- Summary
  - Object-Ray Intersection
    - Adaptive Depth Control
    - Bounding Volume
    - Spatial Subdivision
  - Sampling problem
    - Backward Ray Tracing
    - Cone / Pencil / Beam tracing
    - Stochastic Ray Tracing
      - Distributed Ray Tracing
    - Photon Mapping
  - Parallel Ray Tracing