Production Global Illumination

Yining Karl Li

Advisor: Norman Badler and Aline Normoyle

University of Pennsylvania

ABSTRACT

For my project, I propose building a production quality global illumination renderer supporting a variety of complex features. Such features may include some or all of the following: texturing, subsurface scattering, displacement mapping, deformational motion blur, memory instancing, diffraction, atmospheric scattering, sun&sky, and more. In order to build this renderer, I propose beginning with my existing massively parallel CUDA-based pathtracing core and then exploring methods to accelerate pathtracing, such as GPU-based stackless KD-tree construction, and multiple importance sampling. I also propose investigating and possible implementing alternate GI algorithms that can build on top of pathtracing, such as progressive photon mapping and multiresolution radiosity caching.

The final goal of this project is to finish with a renderer capable of rendering out highly complex scenes and animations from programs such as Maya, with high quality global illumination, in a timespan measuring in minutes rather than hours.

Project Blog: http://yiningkarlli.blogspot.com

1. INTRODUCTION

Global illumination, or GI, is one of the oldest rendering problems in computer graphics. In the past two decades, a variety of both biased and unbaised solutions to the GI problem have been proposed and implemented. Indeed, the GI problem today has been explored to the point that basic raytracers and pathtracers are now a regular staple in computer graphics curriculums in many universities. However, putting GI into use in a practical production setting, such as that in the animation industry, remains a significantly more complex problem than in academic settings due to a variety of concerns, such as speed and memory limitations. A basic student raytracer has to handle perhaps a collection of spheres in a Cornell Box, while a production GI renderer has to be able to handle potentially millions of objects and triangles, compute complex effects such as subsurface scattering, and do it all in a reasonable time frame within a reasonable memory bound.

Although the first GI solution, brute force monte-carlo pathtracing, was introduced over two decades ago by James Kajiya [1986], the practical computational complexity of the global illumination problem has meant that GI has only in the past few years finally begun to see utilization in CG animated movies and in special effects. In the past few years, a number of production quality GI renderers have come to market, including Solid Angle's Arnold, GI extensions to Pixar's Photorealistic Renderman, and Chaos Group's Vray. While Arnold uses highly optimized brute force pathtracing, Renderman uses more novel approaches such as point-based global illumination and multiresolution radiosity caching and Vray uses irradiance mapping and light caching.

This project will explore and implement various techniques required to efficiently render high quality images and animations of extremely complex scenes with potentially millions of polygons and multiple light sources. The project will focus on speed and feature richness above all. While competing with powerhouse commercial renderers like Arnold, Renderman, and Vray is beyond the scope of this project, hopefully the end result will still be feature-rich and fast enough for use as a production academic renderer suitable for usecases similar to those of Cornell's Mitsuba render, or PBRT.

1.1 Design Goals

The target users for the proposed renderer are CG professionals and researchers in need of a fast, feature rich, high quality rendering engine. Such users may include people in the architectural visualization or animation fields, or researchers in academia in need of a renderer for showcasing their research results. As such, a number of key design goals will be taken into consideration for this renderer: 1) Speed and quality. 2) Production features. 3) Support for both biased and unbaised techniques. 4) Some form of integration or interfacing with commercial CG packages such as Maya. 5) An accessible open-source codebase for use in research.

1.2 Projects Proposed Features and Functionality

The proposed renderer will implement unbaised brute force pathtracing in addition to at least one of the following baised GI techniques:

-Progressive photon mapping

-Multiresolution radiosity caching

-Point-based global illumination

-Virtual point lights

The proposed renderer will implement a number, although not necessarily all, of the following production-required features:

-Texturing

-PTEX

-Sun & Sky

-Displacement mapping

-Deformational motion blur

-Transformational motion blur

-Physically accurate depth of feild

-Diffraction and spectral rendering

-Multipass output

-Instanced geometry

-Procedural geometry and shading

-Subsurface scattering

-Volumetric rendering and scattering support

-Multiple importance sampling

-Stereo rendering

In order to facilitate easy use by artists and other CG professionals, the proposed renderer should have either direct Maya integration, or a method for exporting scenes from Maya into a format the renderer can understand.

The renderer will be entirely open source on Github, and will be written in C++ and, as much as possible, in CUDA.

2. RELATED WORK

In the past twenty years, a number of GI methods have been proposed, and many of these methods have been implemented in a number of commercial production rendering packages. This section will examine a brief overview of GI methods, and then summarize some of the major commercial production rendering packages on the market today.

2.1 Global Illumination Algorithms

The roots of global illumination can be traced back to Arthur Appel's 1968 raycasting algorithm [App1968], which was subsequently extended into raytracing by Turner Whitted in 1980 [Whi1980]. In 1984, Cook. et. all proposed Distributed Raytracing, which enabled soft shadows, area lights, depth of field, motion blur, and other features now typical in GI renderers [CPC1984]. Cook's 1986 paper on stochastic sampling in CG laid the groundwork for the monte-carlo techniques that are now a staple in GI rendering [Coo1986].

The first real major breakthrough in the GI problem came with Kajiya's 1986 description of the rendering equation, an integral describing the total amount of light emitted from a point, also known as the radiance, within a given solid angle, given an incoming light function and a surface bidirectional reflectance distribution function [Kaj1986]. Along with his description of the rendering equation, Kajiya also proposed a method known today as brute force monte-carlo pathtracing. Although slow, pathtracing is a pure unbiased technique and used today as the reference point for all other GI methods. Extensions to pathtracing include bidirectional pathtracing, proposed by Lafortune and Willems in 1993 [LW1993], and Metropolis Light Transport, presented by Veach in his 1997 thesis [Vea1997].

In 1995, Henrik Wann Jensen introduced the first biased approach to global illumination in the form photon mapping [Jen1995]. A two pass algorithm, photon mapping begins by tracing photons throughout a scene starting from light sources to construct a photon map. The algorithm then performs a second pass where the radiance of every pixel is estimated by gathering radiance values from the N radiance photons. Photon mapping has since been extended by Hachisuka et. all into progressive photon mapping, in which increasing numbers of photon tracing passes are used to bring the final result closer and closer to an unbaised result [HOJ2008].

Another biased approach commonly used in production today is point-based global illumination, or PBGI, first proposed by Per Christensen in 2008 [Chr2008]. PBGI begins by construction a point cloud representation of all directly illuminated geometry and clusters these points into an octree. Radiance from each cluster is then approximated using spherical harmonics and then indirect illumination is approximated through a combination of rayracing, single disk approximation, and clustering.

In the past few years two new GI methods have been proposed and seen adoption in the industry. The first, virtual point lights (VPL), proposed by Segovia in his 2007 dissertation [Seg2007]. VPL approximates global illumination through the use of numerous point lights throughout a scene, which are automatically placed through techniques such as light tracing, and then clamped to produce a final result. The second, multiresolution radiosity caching, proposed by Christensen et. all is a single pass algorithm that separately evaluating view-independent shading operations from view-dependent ones, and caching the resultant radiosity in three resolutions, which are then referenced based on a given ray's differential [CHS*2012].

In addition to these techniques, in recent years, GI methods suitable for lower quality realtime applications such as voxel cone tracing by Crassin et. all have also been proposed [CNS*2011]. However, these methods are not within the scope of this project.

2.2 Commercial Production Global Illumination Renderers

The dominant GI renderer in the industry today is Pixar's Photorealistic Renderman. Originally implemented as a scanline rasterizing REYES based system, Photorealistic Renderman, or PRMan, has in recent years gained raytracing capabilities and GI cababilities through PBGI, multiresolution radiosity caching, and brute force monte-carlo pathtracing. PRMan is used on all of Pixar's films, as well as on a number of visual effects projects from studios such as ILM, Digital Domain, and Double Negative.

In recent years, Solid Angle's Arnold renderer has rapidly gained popularity among animation and visual effects studios. Arnold is a from-the-ground-up brute force monte-carlo pathtracer. Normally pathtracing is an extremely slow algorithm, but through a number of optimizations and so far undisclosed discoveries, Arnold is capable of producing pathtraced images at a speed comparable to or faster than any other production renderer on the market. Arnold today is Sony Imageworks' primary in-house renderer and has also seen adoption from visual effects studios such as Framestore.

Chaos Group's V-Ray is another massively popular GI renderer. Used primarily by smaller effects houses and architectural visualization studios, V-Ray supports brute force monte-carlo pathtracing, but primarily relies on a combination of irradiance caching and a modified version of photon mapping called light caching.

Arguably the fastest and most advanced GI renderer in the industry today is Blue Sky Animation's CGIStudio, which was one of the earliest GI renderers. However, extreme secrecy and the proprietary nature of CGIStudio means that information about CGIStudio is extremely scarce.

OTOY's CUDA-based pathtracers, Octane and Brigade, represent a new generation of massively parallel GPU based GI renderers capable of generating unbiased GI renders with unprecendented rapidity. Octane and Brigade are the main sources of inspiration for this project's use of CUDA.

Other major commercial GI renderers today include Maxwell Render, Mental Ray, Mantra, and others. Major academic research renderers include PBRT and Cornell University's Mitsuba Render.

3. PROJECT PROPOSAL

This project will implement a robust, feature rich, fast production renderer capable of rendering massive scenes with full global illumination.

3.1 Anticipated Approach

The implementation will build upon my existing CUDA pathtracing core, named TAKUA Render. The existing pathtracing core supports totally brute force, unbiased pathtracing with no optimizations whatsoever. The existing pathtracing core will have to be extended with bidirectional pathtracing and multiple importance sampling at the very least in order to serve as a plausible starting point for techniques such as progressive photon mapping.

Since the existing pathtracing core was originally built specifically with extensibility in mind, once extensions have been completed, adding a large numbers of features rapidly should not be a relatively straightforward task, from the perspective of integration into the renderer. Certain features will almost certainly be more complex than others, however. I anticipate features like memory-efficient displacement mapping and deformational motion blur to be significantly more complex to implement than features like transformational motion blur and depth of field.

Once a number of geometry and shading specific features have been implemented, I'd like to move on to implementing at least one biased technique, such as progressive photon mapping or multiresolution radiosity caching. These methods all begin with raytracing operations, so the extended pathtracing core will prove useful here as well.

Once the renderer has reached a performance and feature level suitable for rendering scenes such as Sponza, comparisons can be made between my renderer and other renderers such as Mitsuba, V-ray, and Octane and further areas of future research can be identified.

3.2 Target Platforms

The renderer will be implemented with C++ and CUDA, meaning that the renderer will target workstations with NVIDIA graphics hardware. The renderer should be able to compile and run without modification on Windows, OSX, and Linux, and will be available open-source on Github. Maya will be used for scene modeling and will either integrate with or export to the renderer.

3.3 Evaluation Criteria

The evaluation criteria for this project will be comparing feature-sets between production renderers such as Arnold and V-ray, research renderers such as PBRT and Mitsuba, and my renderer. This project expects to produce images with quality comparable to existing production renderers, in a timeframe comparable to production renderers, with a similar, but possibly more limited, feature set.

4. RESEARCH TIMELINE

Project Milestone Report (Alpha Version)

- Completed all background reading (Oct. 1)
- Fully functional CUDA pathtracing core (Oct. 1)
- Fully functional stackless spatial acceleration system, such as a stackless KD-tree (Oct. 1)
- Multiple importance sampling and other pathtracing optimizations (Oct. 19)
- Implemented "low hanging" features such as texturing, transformational motion blur, etc (Oct. 19)
- First fully rendered scenes for comparison with existing renderers (Oct. 19)
- Decide on biased algorithm for implementation (Oct. 19)

Project Beta Review

- Implemented "complex" features such as displacement, deformational motion blur(Nov. 1)
- Implemented biased algorithm (Nov. 30)
- Speed optimizations

Project Final Deliverables

- Fully functional renderer
- Maya integration
- Comparisons with existing renderers

- Fully rendered demonstrations scenes and animations
- Final polish

Project Future Tasks

- Implement additional biased techniques
- Implement methods to blend different techniques (enable first bounces through pathtracing, second bounces via progressive photon mapping, etc.)
- Further speed improvments

5. METHOD

6. RESULTS

7. CONCLUSIONS and FUTURE WORK

8. REFERENCES

[App1968] ARTHUR APPEL, "Some Techniques for Shading Machine Renderings of Solids," AFIPS 1968 Conference Proceedings, pp. 33–45, Apr. 1968.

[Whi1980] TURNER WHITTED, "An Improved Illumination Model for Shaded Display," Communications of the ACM, vol. 23, no. 6, pp. 343–349, Jun. 1980.

[CPC1984] ROBERT L. COOK, THOMAS PORTER, LOREN CARPENTER, "Distributed Ray Tracing," Computer Graphics, vol. 18, no. 3, pp. 137–145, Jul. 1984.

[Coo1986] ROBERT L. COOK, "Stochastic Sampling in Computer Graphics," ACM Transactions on Graphics, vol. 5, no. 1, pp. 51–72, Jan. 1986.

[Kaj1986] JAMES T. KAJIYA, "The Rendering Equation," SIGGRAPH 1986: Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques, vol. 20, no. 4, pp. 143–150, 1986.

[LW1993] ERIC P. LAFORTUNE, YVES D. WILLEMS, "Bidirectional Path Tracing," presented at the Proceedings of Compugraphics 1993, 1993, pp. 145–153.

[Jen1995] HENRIK WANN JENSEN, "Importance Driven Path Tracing using the Photon Map," Rendering Techniques 1995: Proceedings of the Sixth Eurographics Workshop on Rendering, pp. 326–335, 1995.

[Vea1997] ERIC VEACH, "Robust Monte Carlo Methods for Light Transport Simulation," Stanford University PhD Dissertation, pp. 1–432, Dec. 1997.

[JB2002] HENRIK WANN JENSEN, JUAN BUHLER, "A Rapid Hierarchical Rendering Technique for Translucent Materials," presented at the SIGGRAPH 2002: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, 2002, pp. 1–6.

[CLF*2003] PER CHRISTENSEN, DAVID LUAR, JULIAN FONG, WAYNE L. WOOTEN, DANA BATALI, "Ray Differentials and Multiresolution Geometry Caching for Distribution Ray Tracing in Complex Scenes," presented at the Computer Graphics Forum (Eurographics 2003 Conference Proceedings), 2003, pp. 543–552.

[CB2004] PER CHRISTENSEN, DANA BATALI, "An Irradiance Atlas for Global Illumination in Complex Production Scenes," presented at the Proceedings of the Eurographics Symposium on Rendering 2004, 2004, pp. 113–141.

[JC2007] HENRIK WANN JENSEN, PER CHRISTENSEN, "High Quality Rendering using Ray Tracing and Photon Mapping," SIGGRAPH 2007 Course Notes, pp. 1–130, May 2007.

[Chr2008] PER CHRISTENSEN, "Point-Based Approximate Color Bleeding," Pixar Technical Memos, pp. 1–9, Jul. 2008.

[Seg2007] BENJAMIN SEGOVIA, "Interactive Light Transport with Virtual Point Lights," University Claude Bernard PhD Dissertation, pp. 1–169, Oct. 2007.

[HOJ2008] TOSHIYA HACHISUKA, SHINJI OGAKI, HENRIK WANN JENSEN, "Progressive Photon Mapping," SIGGRAPH Asia 2008: Proceedings of ACM SIGGRAPH Asia 2008, pp. 1–8, Sep. 2008.

[HJ2009] TOSHIYA HACHISUKA, HENRIK WANN JENSEN, "Stochastic progressive photon mapping," ACM Transactions on Graphics, vol. 28, no. 5, p. 1, Dec. 2009.

[Chr2010] PER CHRISTENSEN, "Point-Based Global Illumination for Movie Production," SIGGRAPH 2010 Course Notes, pp. 1–21, May 2010.

[CNS*2011] CYRIL CRASSIN, FABRICE NEYRET, MIGUEL SAINZ, SIMON GREEN, ELMAR EISEMANN, "Interactive Indirect Illumination Using Voxel Cone Tracing," Computer Graphics Forum, vol. 30, no. 7, pp. 1921–1930, Nov. 2011.

[CHS*2012] PER CHRISTENSEN, GEORGE HARKER, JONATHAN SHADE, BRENDEN SCHUBERT, DANA BATALI, "Multiresolution Radiosity Caching for Efficient Preview and Final Quality Global Illumination in Movies," Pixar Technical Memos, pp. 1–10, Jul. 2012.