Light propagation visualization as a tool for 3D scene analysis in lighting design

Edward A. Kopylov[†], Kirill A. Dmitriev
Computer Graphics Department, Keldysh Institute of Applied Mathematics, RAS, Moscow, Miusskaya sq., 4, 125047, Russia

Abstract

This paper is devoted to a designer tool, which is an extension of particle tracing algorithm for analysis of scenes, artifacts, image ghosts, and ray tracing mechanism itself. The global illumination problem here is considered at the level of light propagation visualization, which takes into account all cognitive benefits of visual representation. Physically accurate particle tracers randomly generate thousands of rays per second to simulate light propagation in a scene considering light as consisting of photons with a constant energy. The proposed tool utilizes and filters the information of how light photons propagate in the scene. Several examples in the paper demonstrate a practical value of obtained photon tracks for scene analysis.

Keywords: Program visualization; Lighting design; Light propagation; Global illumination; Particle tracing; Ghost analysis; Illumination analysis; Russian roulette.

1. Introduction

The lighting design in computer graphics is an expanding field, which involves architectural applications, systems for luminaire design, design of backlight devices, devices with light-conductive elements, etc. The main goal of such systems is physically accurate solution of global illumination problem. It means that a computer model of some real scene is given and systems calculates how light is distributed in the scene, how each object is illuminated, and finally how the scene is viewed from given camera.

By its very definition, computer graphics is a visual field, however, in order to understand its solution results, a user most likely envisions a virtual world full of additional geometric objects [1] such as viewer models, image planes and rays. If a lighting design system works correctly, effectively, and if the final result is all that is expected, then users probably have little interest in the visualization of light propagation in the scene. However, if the result seems to be incorrect, or if a user wishes to explore the behavior of the system or explain its behavior to others, then an appropriate image how light rays propagate has great practical value for him.

The cognitive benefits of visual representations have been well known for years. These benefits include the substitution of simple perceptual inferences for more difficult logical inferences, the reduction of search time for locating desired information, the reduction of short-term memory loads during problem solving, and an improved recall of data [2]. For example, in order to thoroughly understand the light simulation mechanism in backlight devices, where light propagates through transparent objects undergoing multiple reflections and refractions, a user must have a visual perception of light energy transfer.

Although the visualization of light propagation in the scene is very attractive for programmers for program debugging and algorithm optimization points of view [1], it gives great benefit also for users of lighting design software to detect problems in the scene description, to explain artifacts and ghosts in results, to analyze the efficiency of the calculation depending on scene specific features.

Any local reflectance model, which determines how light interacts with objects in the scene, is not conceptually difficult, however the comprehensive global reflectance model that includes all possible ways of light propagation in the scene dilutes the user's initial intuition about how light will spread and makes it difficult to track down incorrectness in

---

[†] Correspondence author.
E-mail address: oek@gin.keldysh.ru (E. Kopylov)

the scene specification. Anomalies in results are a fairly common occurrence, and even if a user can determine the general nature of the problem (e.g. "this object is too bright"), it would be difficult to isolate the cause of the problem without a proper visualization of light interaction with scene objects.

The information about possible ways of light propagation in the scene is available with Monte Carlo random walk algorithms, which are well known methods for physically accurate solution of global illumination problem. The first Monte Carlo ray tracing algorithm in computer graphics – called *distributed ray tracing* – was proposed by Cook at al. in [3], which spawned to a set of variations, including *particle tracing* [4] and *light tracing* [5]. An simple idea stated in [3] – to follow the global conception using a random choice everywhere – is implemented in particle tracing, where each event, which may change the energy of a ray originated from a light source is treated in the probabilistic way by the *Russian roulette* rule: either photon ray survives with the unchanged energy or it completely disappears. In essence this principle allows us to consider the space of all possible ways of light spread, everywhere densely covered by photon tracks generated by particle tracer.

It can be formally proved that most methods which operate with particle tracing approximate the well known rendering equation [6], but we will not stop this here, taking into account that there is a physical justification: the tracing of light photons is deduced from physics of light.

2. Light propagation visualization

The objective of this paper is to show a practical value of the visualization of light flow between surfaces in the scene. This information is available from methods of global illumination analysis such as particle tracing which traces thousands of rays per second to collect photon histories or account ray-surface interactions in some other way.

Let us consider the typical work of particle tracer in details. A photon is born at light source surface and initial direction of its propagation is chosen randomly in agreement with luminous intensity distribution of the light source. Then the photon generated is traced until ray intersection with some scene object. After intersection is found it is randomly determined what event happens with this photon (specular or diffuse, reflection or refraction). The *Russian Roulette* rule is used to avoid tracing of multiple successor rays. Only one is chosen and traced further until it is absorbed or leaves the scene.

In principle this method provides complete global light propagation model for scenes composed of specular mirrors, diffuse surfaces, dielectric boundaries, self-luminous objects and surfaces characterized by general BRDF (BTDF). This is a very important property of the method as it allows us to obtain photon tracks involving arbitrary long sequences of specular and diffuse interreflections, which are typical for some technical devices like backlight emitter devices. Of course it has no much sense to visualize all photon tracks. Particle tracer has a potential to generate and trace enormous number of photons in a very short period of time. Thousands of tracks representing traced photon paths will fill the screen very soon without giving any idea of how some interesting artifact of the image appears. So the necessary information is attenuated in tons of generated photons and the ability to specify a criterion which tracks should be visualized is crucial point here.

If we are looking for a technique to specify a photon visualization criterion, then one way to do it is to use AND-OR tree [7]. Almost any practically interested criterion can be decomposed into a set of elementary events like “intersection with picked surface” combined with AND and OR logic relations. With AND-OR tree we can have *and* nodes whose successors must *all* be achieved, and *or* nodes where *one* of the successors must be achieved. This allows us to specify visualization criterion of arbitrary complexity where all set of sub criterions must be satisfied for a photon being visualized and where there are alternative subcriterions, any of which could be satisfied.

The proposed light propagation visualization system is a combination of particle tracing and some kind of finite automation to recognize photon tracks under the interest. The incorporation with particle tracing looks very natural and claims to be named a process of method visualization. As it is described in [1] such process usually involves three general phases: 1) the program instrumentation phase, 2) the visualization building phase, and 3) the visualization usage phase. The implementation of these phases within the proposed light propagation visualization system is described below.

2.1 Instrumentation phase

During the instrumentation phase, the code of existing method of particle tracing is modified to recognize the interesting photon tracks and remember them for subsequent visualization. Here it should be noted that each photon is uniquely identified inside of particle tracer with status of pseudo random generator before photon birth. So, to replay some photon track from the start, it is required only to recover appropriate pseudo random generator state and run particle tracer.

At this phase visualization criterion is applied to each traced photon to filter only interesting photon tracks for the visualization. We introduced simple and intuitively understandable set of elementary events, which can compose AND-OR tree of visualization criterion:

- photon intersects with specified scene objects;
- photon experiences given number of particular transformations with surfaces (specular or diffuse refraction, reflection by BRDF, etc.);
- photon hits camera observer.

User interface at this phase allows a user to visually construct AND-OR tree for the specification of visualization criterion.

There are many possible ways to recognize if traced photon satisfied to visualization criterion or not. One of them is to effectively transform AND-OR criterion tree into OR tree where each node represents a whole set of goals to be satisfied.

2.2 Visualization building phase

Selected photon tracks are displayed by a 3D polylines in the scene shaded by Open GL. Internal polyline vertices coincident with intersection points found by particle tracing method. The last vertex of the polyline is either a point of intersection where the photon is happened to be absorbed or an estimated absorption point in scattering medium (if the photon is absorbed in the medium), or finally it is a point outside of scene domain if the photon goes away. No additional non-trivial programming is necessary at this phase because whole photon track is traced from beginning by particle tracer by recovery of random generator status stored on previous phase.

2.3 Using the visualization phase

In the final phase, a user can interact with the resulting visualization by manipulating the following useful controls:

- to trace next photon track segment or whole track;
- to query the system for more information about photon-surface intersection;
- to view statistic information about selected photons;
- some useful operations with photon tracks including arbitrary number of transaction recoils, group operations with selected photons etc.

Obtained photon tracks are view-independent so viewpoint manipulations and camera animation are available.

3. Practical use of light propagation visualization

The following subsections describe practical cases when visualization of light propagation really helps for scene analysis.

3.1 Parasitic light

The light propagation visualization system was applied to study the influence of parasitic light on the quality of image formed by lens (Fig. 1).

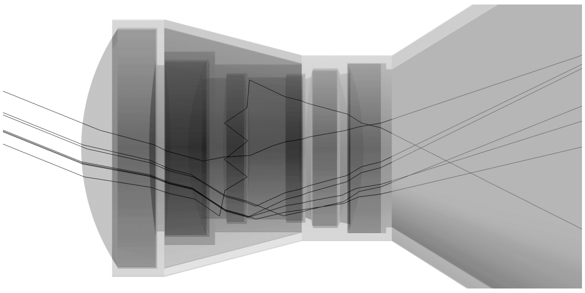
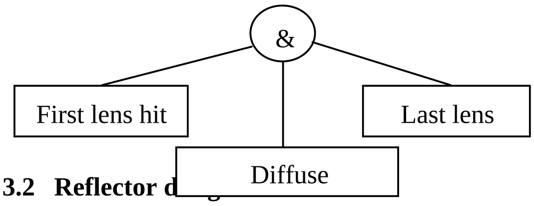


Fig. 1. Parasitic light inside of camera objective.

The whole construction is an objective consisting of 6 lenses cylindrical butt-ends and lens holders. The lens holders and lens butt-ends have the slight diffuse properties. These diffuse parts are the source of parasitic light when some amount of incident light is reflected from them. The figure with light propagation shows a formation of parasitic photon tracks originated from the sun at elevation angle 22 deg.

The visualization criterion for Fig. 1 is AND tree with three events:



3.2 Reflector d

The next case shown on Fig.2 relates to a problem with lamp reflector design. During lighting simulation a designer notices a drawback in the light distribution from parabolic reflector. The visualization of photon tracks with simple criterion to visualize only specular reflected photons gives the source of the problem – inaccurate shape representation in the reflector model.

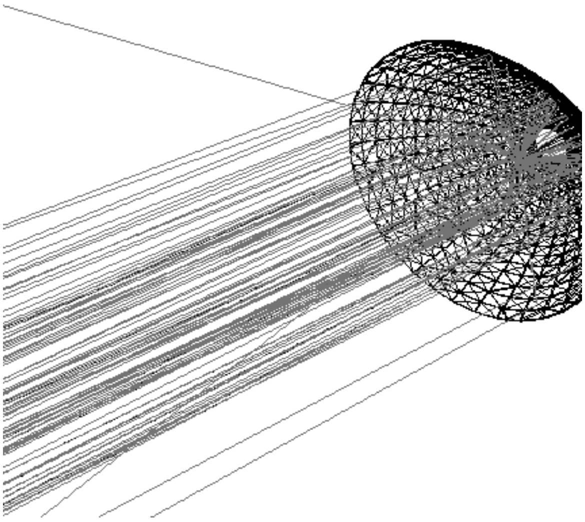


Fig. 2. Investigation of parabolic reflector.

3.3 Color bleeding

Light, which is diffusively reflected from a surface, is attenuated by the reflectivity of the surface, which is closely associated with the color of the surface. The reflected light energy often is colored, to some small extent, by the color of the surface from which it was reflected. This reflection of light energy in an environment produces a phenomenon known as *color bleeding*, where a brightly colored surface will 'bleed' onto adjacent surfaces. The first image on Fig. 3 illustrates this phenomenon, as green floor ‘bleeds’ its color onto the white walls and ceiling. For a contrast the next image in the same row is especially generated without *color bleeding* phenomenon.

The row below is a visualization of photon tracks participated in the creation of this phenomenon. The colored triangles are rough result of illumination maps produced during tracing of these photons.

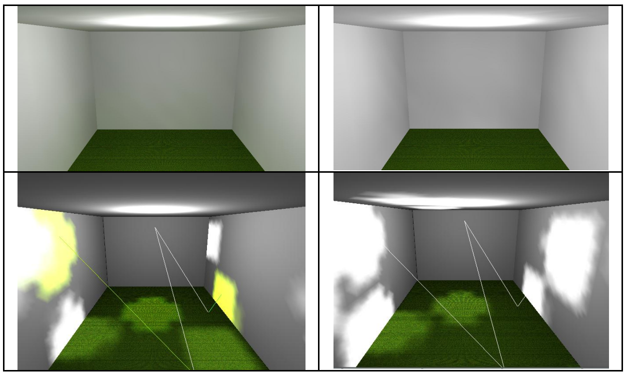


Fig. 3. Investigation of color bleeding effect.

3.4 Caustic analysis

The physically accurate particle tracers provide really accurate calculation of illuminance distribution in the scene including *caustics*.

Fig. 4(a). An image with caustics.

Let us recall what *caustics* are [8]. A *caustic surface* is formed as a result of light interaction (reflection or refraction) with the boundary of a medium that has nonlinear geometrical and/or optical properties. After such interaction, light photons change their initial direction, which leads to redistribution of the reflected or refracted light energy . *Caustic surfaces* usually subsequently intersect with some surface, producing bright curves on it, known as *caustics*. An everyday example is brightly lit-up curves on the inner surface of a cup or a mug. Fig.4(a) illustrates an image with *caustics* calculated by bi-direction ray tracing, which is a combination of particle tracing that produces illumination maps [9] and backward ray tracing.

Fig. 4(b). Photon tracks form caustics.

The Fig.4(b) uncovers the mechanism of the forming of two caustics near of glasses by light photons. While the source of left caustics is obvious, the right one has sufficiently complex origin.

3.5 Scene correctness analysis

It is the most applicable area of light propagation visualization. Scenes in computer graphics consist from geometry data, which defines shapes of the objects to be rendered, optical properties of the materials of which the objects are built and light sources illuminating the model. And each of these scene components can be specified incorrectly that lead to problems in lighting simulation.

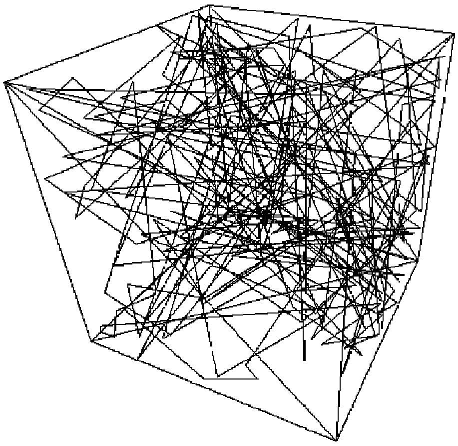


Fig. 5. Photon cycling inside of diffuse cube.

For example, let us imagine a closed diffuse cube with no absorption and a light source inside. The any physically accurate particle tracer will have a problem with such scene and, we know it from practice, the analysis of such trivial problem can take non-trivial time. On Fig. 5 a single photon track is visualized inside of absolutely diffuse cube until it was interrupted by hand. The visualized track helps to find the reason of the problem immediately.

3.6 Scene incomprehensibility analysis

The common problem for designers is to explain some strange effects visible on calculated images. Even if the scene is specified correctly this problem can be not trivial. Let us consider an example of strange ring reflection discontinuity on the ceiling shown on Fig. 6.

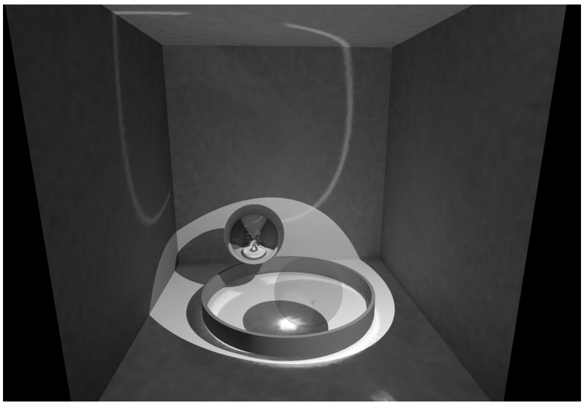


Fig. 6. The discontinuity of ring reflection on the ceiling.

The following Fig. 6(a) shows photon tracks that participate in the forming of ring reflection. The reason of the discontinuity is not clearly understandable on this step, however here we can recognize a ring segment, which is not involved in reflection generation.

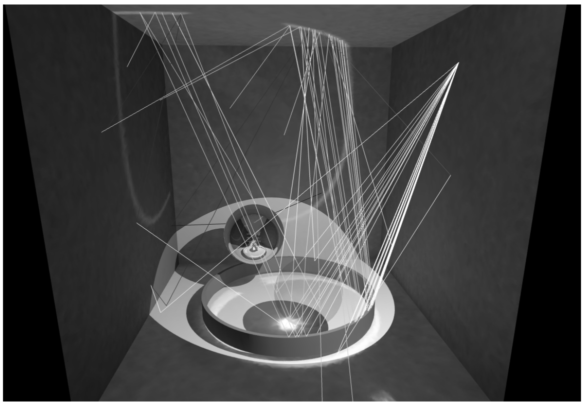


Fig. 6(a). These photon tracks generate ring reflection.

The next step is to visualize photon tracks reflected from this ring segment (see Fig 6(b)) and the reason of discontinuity become obvious – a refracting glass ball is encountered on the way of these photons.

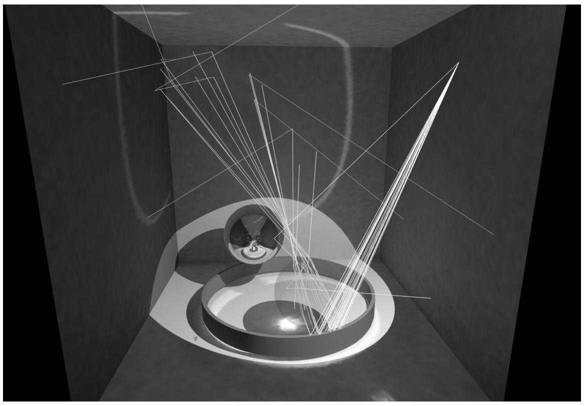


Fig. 6(b). These photon tracks are refracted by glass ball.

3.7 Photon statistic analysis

In principle small number of incorrectly traced photons does not affect the produced result. E.g. as result of floating point calculation error several photons can leak between adjacent triangles. The correspondent criterion can be specified as lack of intersection of photons originated inside of a solid body with its surface.

Simple division of found photon number by total photon number can obtain the simplest statistical information about interesting photons in light visualization system. This information can be also used for estimation of optical device efficiency. E.g. ratio of photons transmitted by lamp reflector in the given direction is efficiency of this device.

Also particle tracer efficiency can be investigated. In mediums with refraction properties the phenomenon of total internal reflection can significantly increase the photon life length. Large number of such photons will slow down the particle tracer performance.

4. Conclusion

The light propagation visualization systems are by no means new or unique. For example, Photopia distributed by Lighting Technologies, Inc. allows importing light rays back into the luminaire model to see how light moves through the fixture. Also the backward visualization of light propagation is popular to demonstrate how color of a single pixel is dependent upon tracing of viewing rays and shadow feeler rays. However, it seems to be there is no system like our one that has the capacity to allow user querying for light propagation path of arbitrary complexity for providing comprehensive analysis of the light distribution over the scene.

The proposed light propagation visualization is unique designer tool in computer graphics. It fully utilizes the information of how light photons propagate in the scene with the help of cognitive benefits of visual representation and power instrument of photon selection. It is our hope that this tool in the lighting design domain will find deserved application, maybe one from described above.

Acknowledgments

We acknowledge the support in part of the Russian Foundation for Basic Research under a grant entitled "Physically accurate solution of global illumination analysis" (98-01-00547).

Also we would like to thank our colleagues Andrei Khodulev and Alexander Trifonov for a great help and assistance in the implementation of visualization system, Vadim Sokolov and Aleksey Garbul for examples of practical usage.

References

1. E. Wernert. A unified environment for presenting, developing and analyzing graphics algorithms. In Computer Graphics (August 1997).

2. J. Mackinlay. Automating the design of graphical presentations and relational informations. ACM Transactions on Graphics, Vol.5, No.2 (1986).

3. R. Cook, T. Porter, L. Carpenter. Distributed ray tracing. In Computer Graphics, SIGGRAPH ’84 Proceedings, 137-145 (1984).

4. J. Arvo. Backward ray tracing. In Developments in Ray Tracing, ACM Siggraph ’86 Course Notes

(1986).

5. P. Dutre, E. Lafortune, Y.D. Willems. Monte Carlo light tracing with direct computation of pixel intensities. Compugraphics ’93, Alvor, 128-137 (1993).

6. J.T. Kajiya. The rendering equation. In Computer Graphics, SIGGRAPH'86 Proceedings, v.20, 143-150 (August 1986).

7. I. Bratko. Prolog Programming for Artificial Intelligence. Addison Wesley, London (1990).

8. A. Watt, M. Watt. Advanced Animation and Rendering Techniques. Theory and Practice, Addison Wesley, London, 455p (1992).

9. V. Volevich, K. Myszkowski, A. Khodulev, E. Kopylov. Perceptually informed progressive global illumination solution. Technical Report 99-1-002, The University of Aizu, Japan (1999).