ICIC Express Letters ICIC International ©2016 ISSN 1881-803X
Volume **10**, Number **4**, April **2016** pp. **943–948**

GPU-BASED RENDERING OF GLOBAL ILLUMINATION USING DISCRETIZATION OF RADIOSITY EQUATION

Wenjun Song[1,*], Jinhua Fu[1] and Yang Zhang[2]

[1]School of Computer and Communication Engineering
[2]College of Mathematics and Information Science
Zhengzhou University of Light Industry
No. 5, Dongfeng Road, Zhengzhou 450002, P. R. China
[*]Corresponding author: songwjzz@163.com; zhangyangzzuli@163.com

Received October 2015; accepted January 2016

Abstract. *Global illumination forms a fundamental part of the realistic image synthesis discipline. In order to obtain a real-time rendering of global illumination, we present discretization of radiosity equation to implement simulation of global illumination on GPU (Graphics Processing Unit). We analyze the direct and indirect illumination components to reveal the global illumination, and give the rendering equation to simulate it. Then the rendering equation could be derived into discrete and simple formula by discretization of radiosity equation, which fits GPU calculation easily. The result shows that we could acquire better effects of global illumination such as caustics, soft shadows and color bleeding, while implementing it at a relatively higher FPS (Frames Per Second).*
Keywords: Global illumination, Real-time rendering, Radiosity equation, Graphics Processing Unit

1. **Introduction.** Global illumination (GI) enhances the realism in rendered images by simulating all light inter-reflection effects. It generates realistic effects such as color bleeding, caustics and soft shadow. So global illumination is an important factor in creating realistic scenes and provides visual cues for understanding scene geometry.

Several rendering methods, such as path tracing [1], Monte-Carlo algorithm [2] and photon mapping [3,4], can simulate the global illumination efficiently. However, these methods are general in that they can handle arbitrary illumination, geometry and materials in an unbiased manner, they are typically computationally expensive and may take long to converge, making them infeasible for practical rendering. Recently it has become viable to render scenes with global illumination effects at interactive frame rates by exploiting the parallelism and programmability of modern GPUs. Nijasure et al. [5] have developed a system for computing a plausible global illumination solution on GPU, which runs on DirectX 9.0 and computes a global illumination solution for reasonably complex scenes at frame rates of 10-40 FPS. Then Barcelona [6] designs the Virtual Light Field (VLF) method, and enables real-time global illumination rendering in VR (Virtual Reality). The VLF has also been integrated with the Extreme VR system for real-time GPU-based rendering. With the limited computing resources in mobile devices, Ahn et al. [7] use the multi-resolution sampling method with 3D geometry and attributes simultaneously, and realize the real-time global illumination. These recent GPU-based algorithms enable the computation of global illumination solutions for fully dynamic scenes and are of interest to both the academic research community and practitioners of interactive computer graphics. Now sophisticated rendering techniques and increase in computational power have also made it feasible to incorporate global illumination effects into many practical applications. In skin rendering, Yang et al. [8] design an interactive rendering platform based on three-layer material model and realize subsurface scattering of global illumination. Improving the off-line raytracing [1,4], Dahlin et al. [9] present a GPU-based rendering system to enable real-time raytracing of scenes illuminated with video environment maps. Considering the focus point in Mixed Reality applications is the merging of objects from different realities into a new, visibly homogeneous scene, Thӧner and Kuijper [10] propose an algorithm based on voxel cone tracing [11] to provide global illumination. With the increasing requirements of faster interactive applications, the main challenge of rendering is not only to achieve global illumination rendering but also it should be easily integrated with existing GPU rendering pipelines at higher FPS.

We are motivated by the requirement, and seek to give faster rendering of global illumination. We derive discretization of the radiosity equation which fits GPU evaluation easily, and acquire an efficient simulation of global illumination. The result shows that we could acquire better visual effects of global illumination while implementing the rendering at higher FPS.

The rest of this paper is organized as follows. Global illumination and its rendering equation are given in Section 2. Then discretization of diffusion model for GPU implementation is showed in Section 3. Finally, experimental results are discussed in Section 4, and conclusions are drawn in Section 5.

2. **Global Illumination and Its Rendering Equation.** Global illumination is the simulation of physically based light transport. It is a means to simulate how light is transferred between surfaces in your 3D scene and it will greatly increase the realism of your scene. Not only that, it is also a means to convey mood and with clever use can be used to improve the rendering applications. GI algorithms take into account not only the light which comes directly from a light source (the direct illumination), but also subsequent cases in which this light is reflected by surfaces in the scene using different materials (indirect illumination).

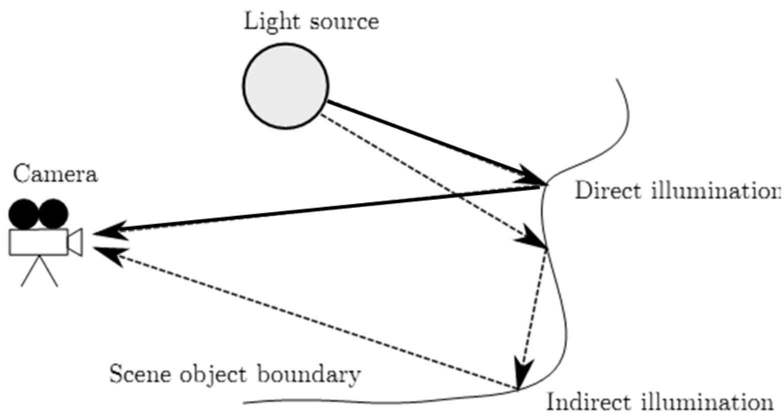


FIGURE 1. Direct illumination (full line) and indirect illumination (dashed line)

Figure 1 shows two examples of illumination types that we distinguish for points that are visible from the camera. The path shown in solid line with arrow denotes the direct illumination, since it leads directly from the light source, bouncing off the surface and then to the camera. The path shown in dashed line with arrow adds one more light bounce. The surface is then lit indirectly; we call global illumination algorithms combine both direct and indirect illumination.

The fundamental equation which expresses the global illumination within the scene is called radiance equation as follows

$$L_o(x,\omega) = L_e(x,\omega) + \int_\Omega \frac{\rho(x,\omega,\omega')}{\pi} L_i(x',\omega')(\omega'\cdot N)d\omega' \tag{1}$$

where Equation (1) shows that the radiance L_o coming from the scene point x in the direction ω, is given by a sum of two terms. L_e is the emitted light energy from the point x in ω. ρ is the BRDF (Bidirectional Reflectance Distribution Function) factor which expresses the diffuse reflectivity (also known as albedo) of the scene point x. L_i is the incoming radiance at point x' in direction ω'. N is the normal at point x'.

Rewriting the equation to integrate over all scene elements (surfaces) instead of angles we receive:

$$L_o(x,\omega) = L_e(x,\omega) + \int_S f_r(x,\omega) L_i(x',\omega')(\omega' \cdot N) dA' \tag{2}$$

where the $\frac{\rho(x,\omega,\omega')}{\pi}$ term can be substituted with f_r. The integral over the surface S sums the reflected radiance. A' is the microfacet around the exiting point.

3. **Discretization of the Radiosity Equation.** Based on the radiosity method, we derive the discretized version of the rendering Equation (2) for GPU computation easily as follows. Given radiosity's assumption that all surfaces are perfectly diffuse, we can first drop the angular dependence of all terms in our equation and this also allows us to move the BRDF f_r outside the integral entirely:

$$L_o(x) = L_e(x) + f_r(x) \int_S L_i(x') G(x,x') V(x,x') dA' \tag{3}$$

where V is the visibility factor which is equal to either 0 or 1. G is the geometric factor which expresses the mutual pose of the point x and x'.

$$G(x,x') = \frac{\cos\theta \cos\theta'}{\|x - x'\|^2} \tag{4}$$

where the mutual pose of θ and θ' is shown in Figure 2.

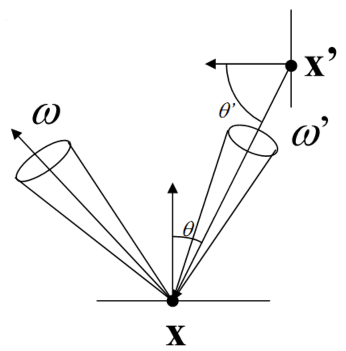


FIGURE 2. The mutual pose of θ and θ'

As the radiance outgoing from the scene point x is independent of the direction ω and equal to radiosity of the point x divided by π, the radiance equation can be multiplied by π and rewritten to express the radiosity:

$$B(x) = E(x) + \rho(x) \int_S \frac{B(x') G(x,x') V(x,x')}{\pi} dA' \tag{5}$$

If we deal with a scene which is made of a finite number of diffuse surfaces we can substitute the integral with the sum:

$$B(x) = E(x) + \rho(x) \sum_{j=1}^{N} \int_{A_j} \frac{B_j G(x,x') V(x,x')}{\pi} dA_j \tag{6}$$

The mean radiosity of the i-th scene element B_i can be computed as:

$$B_i = \frac{1}{A_i}\int_{A_i} B(x)dA_i = E_i + \rho_i \sum_{j=1}^{N} B_j \int_{A_i}\int_{A_j} \frac{G(x,x')V(x,x')}{\pi} dA_j dA_i \tag{7}$$

We now take the plunge and discretize this equation over surfaces i, j in the scene:

$$B_i = E_i + \rho_i \sum_{j=1}^{N} B_j F_{j\to j} \tag{8}$$

The newly introduced term $F_{j\to j}$ is the *form factor* between i and j. Simply, it is the fraction of light energy leaving surface patch j that arrives anywhere on patch i shown in Figure 3.

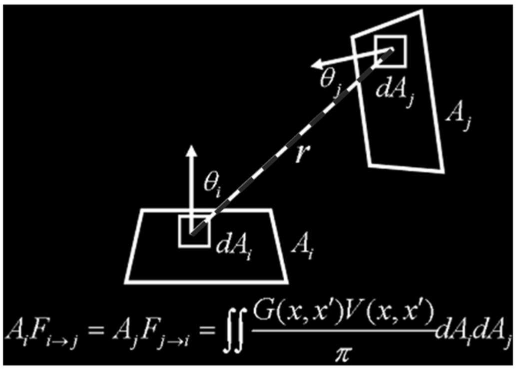


FIGURE 3. Form factor

The final step is to formulate this discretized version of the rendering equation as a matrix equation:

$$B_i - \rho_i \sum_{j=1}^{N} B_j F_{j\to j} = E_i \tag{9}$$

Based on $M_{ij} = I_{ij} - \rho_i F_{i\to j}$, we can acquire

$$\sum_j M_{ij} B_j = E_i \tag{10}$$

Finally, we give the discretization of the radiosity equation which fits to GPU computation:

$$MB = E \tag{11}$$

where M and E are known and so we easily solve B.

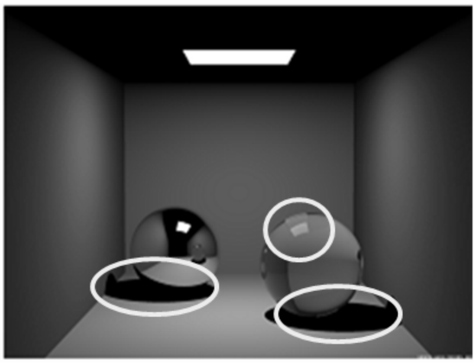
4. **Results.** We have implemented the global illumination rendering using discretization of radiosity equation on GPU. With Intel(R) Core(TM) i7-4910MQ CPU @ 2.90GHz and NVIDIA GeForce GTX 970, we have realized the method using GLSL shader and OpenGL programming in VS2012. In implementation, about 30KB-80KB mesh data is rendered using different methods. We have achieved relatively high speed of rendering and our results clearly show the effective implementation of our technique.

In rendering phase, we first push 3D object meshes to OpenGL, and then implement rendering equation using discretization of radiosity equation on GPU. In order to compare difference of three different illuminations and prove global illumination can do better rendering effect as related above, firstly, we render 3D budda model in the Cornell Box under direct illumination, indirect illumination and global illumination as shown in Figure 4.

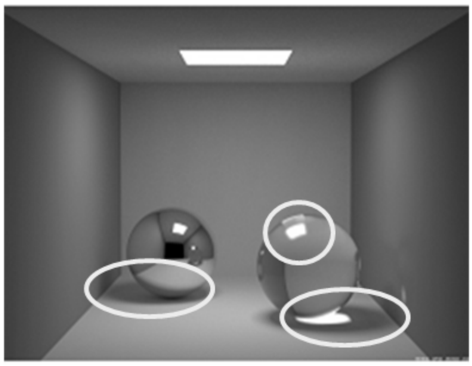
From Figure 4, direct illumination and indirect illumination cannot give realistic effects such as soft shadows and color bleeding, especially paying attention to rectangle area. However, visual effects of realistic rendering can be clearly observed using our method of global illumination.

Then, we render two balls in the Cornell Box and compare the rendering effects betwwen real-time raytracing [9] and our method, which is shown in Figure 5.

FIGURE 4. Rendering comparison and from left to right: direct illumination, indirect illumination and global illumination



(a) Realistic rendering via real-time raytracing [9]



(b) Global illumination rendering via our method

FIGURE 5. Realistic rendering via different methods

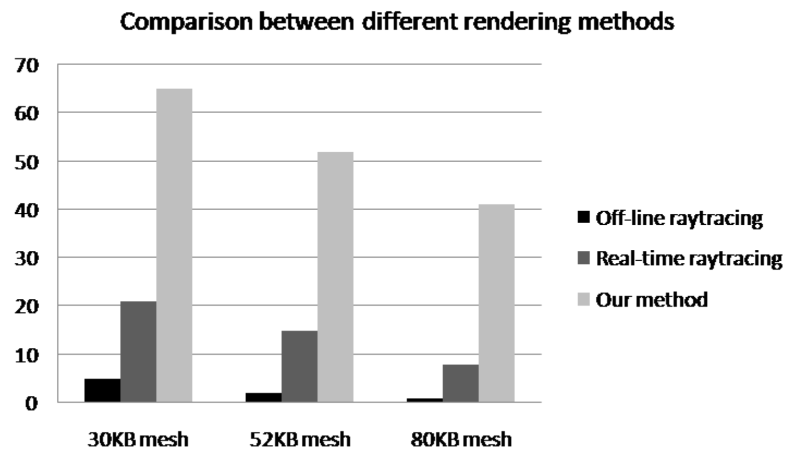


FIGURE 6. FPS comparison between different rendering methods

From Figure 5, we can observe both circle area and whole effect of global illumination effects such as caustics, color bleeding and soft shadows in the right Figure 5(b) are more obvious than Figure 5(a). So we can efficiently realize the global illumination rendering via discretization of radiosity equation. However, real-time raytracing is not physically correct and thus it cannot offer enough visual scene.

To give an objective evaluation, we use FPS as performance parameter which is evaluated based on different mesh quantity of rendering scene. The comparison is shown in Figure 6. From Figure 6, we can obtain a relatively higher FPS while producing highly plausible images. Especially, when increasing mesh quantity, rendering speed of our method is more obvious over other two methods. So our method is more efficient over off-line raytracing [4] and real-time raytracing [9].

5. **Conclusions.** We have implemented the real-time rendering of global illumination based on discretization of radiosity equation. The result shows that global illumination effects of budda model is better than direct illumination and indirect illumination in visual effects such as soft shadows and color bleeding. Also we can give realistic rendering of global illumination over real-time raytracing [9] in caustics and other effects. Finally, we give an objective evaluation and compare rendering speed among off-line raytracing [4], real-time raytracing [9] and our method, which shows higher FPS acquired by our method while producing highly plausible images.

Future improvements include improving the rendering equation with an importance sampling framework to properly enhance the rendering speed. Another interesting feature to consider is integration of subsurface scattering into the application to increase the visual fidelity of the rendered images.

REFERENCES

[1] M. Abramovici, P. R. Menon and D. T. Miller, Critical path tracing: An alternative to fault simulation, *IEEE Design and Test of Computers*, vol.1, no.1, pp.83-93, 1984.

[2] L. Szirmay-Kalos, Monte-Carlo global illumination methods – State of the art and new developments, *Spring Conference on Computer Graphics*, 1999.

[3] H. W. Jensen, *Realistic Image Synthesis Using Photon Mapping*, AK Peters Publisher, 2001.

[4] H. W. Jensen, A practical guide to global illumination using ray tracing and photon mapping, *ACM SIGGRAPH Course Notes*, no.5, 2004.

[5] M. Nijasure, S. Pattanaik and V. Goel, Real-time global illumination on GPUs, *Journal of Graphics GPU and Game Tools*, vol.10, no.2, pp.55-71, 2005.

[6] U. D. Barcelona, Real-time global illumination for VR applications, *IEEE Computer Graphics and Applications*, vol.28, no.6, pp.56-64, 2008.

[7] M. Ahn, I. Ha, H. E. Lee and J. D. K. Kim, Real-time global illumination on mobile device, *Proc. of the SPIE*, vol.9030, no.1, pp.271-283, 2014.

[8] M. Z. Yang, K. Q. Wang and W. M. Zuo, Interactive rendering for realistic face via three-layer material model, *Acta Photonica Sinica*, vol.42, no.7, pp.857-863, 2013.

[9] J. Dahlin, D. Jonsson, M. Kok, T. B. Schon and J. Unger, Real-time video based lighting using GPU raytracing, *Proc. of the 22nd European Signal Processing Conference*, pp.1627-1631, 2014.

[10] M. Thöner and A. Kuijper, Delta global illumination for mixed reality, *Lecture Notes in Computer Science*, vol.9179, pp.108-118, 2015.

[11] C. Crassin, F. Neyret, M. Sainz, S. Green and E. Eisemann, Interactive indirect illumination using voxel cone tracing, *Computer Graphics Forum*, pp.1921-1930, 2011.