ICIC Express Letters
Part B: Applications
Volume **7**, Number **3**, March **2016**

ICIC International ©2016 ISSN 2185-2766
pp. 599–605

GPU-BASED RENDERING VIA IMPROVED DIFFUSION EQUATION AND GEOMETRIC FEATURE SAMPLING

Jie Xu[1], Wenjun Song[2] and Jinhua Fu[2,3,*]

[1]School of Software
[2]School of Computer and Communication Engineering
Zhengzhou University of Light Industry
No. 5, Dongfeng Road, Zhengzhou 450002, P. R. China
phdsciei@gmail.com; songwjzz@163.com

[3]State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou Information Science and Technology Institute
Zhengzhou 450000, P. R. China
*Corresponding author: fujinhuazz@qq.com

Received September 2015; accepted December 2015

Abstract. *In order to acquire a fast rendering of translucent materials via GPU (Graphic Processing Unit) rendering pipeline, we propose an effective method for realistic rendering based on improved diffusion equation and geometric feature sampling. Considering diffusion equation can simulate the complex subsurface scattering inside translucent materials, we improve and derive the diffusion equation as a discrete form to fit GPU evaluation easily. Using geometric features such as curvature and area, we improve classic Turk's point repulsion algorithm by the geometric feature sampling to spread out the sample points reasonably. The results show that our method could acquire a more realistic appearance by geometric feature sampling, better shading effect and relatively fast speed of rendering.*

Keywords: GPU, Improved diffusion equation, Geometric feature sampling, Realistic appearance

1. **Introduction.** There are many translucent materials such as milk, skin, marble and wax, which can be easily distinguished from others without any difficulty. This is because of the nature of their reflection properties of light. Objects made of translucent materials let light rays get inside the object, travel inside the material and leave the object at a different point of the surface. This effect is called *subsurface scattering*. So we have to simulate subsurface scattering based on GPU rendering pipeline to meet the requirement of realistic real-time rendering.

So far, many methods for subsurface scattering simulation have been proposed in the past. A simplified BSSRDF (Bidirectional Surface Scattering Reflectance Distribution Function) model is firstly proposed by Jensen et al. [1] and one prevailing way to handle translucency in rendering is realized in [1]. Then, hierarchical integration technique is presented in [2] in order to use an analytical model for subsurface scattering; unfortunately, this technique does not appear to allow real-time rendering, which is what the authors of this paper seek. Sundstedt et al. [3] state that the simulation of subsurface scattering is challenging because it must correctly simulate the light transport and capture its appearance. The light transport inside translucent materials is very complex and tends to become isotropic even if the incident light and phase function are anisotropic. The experiments show it can use diffusion equation to simulate the complex subsurface scattering. So Wang et al. [4] present the polygrid diffusion algorithm to render translucent materials, which solves the diffusion equation with a boundary condition defined by the given illumination environment. Soon after, Arbree et al. [5] describe a finite element solution of the heterogeneous diffusion equation that solves heterogeneous subsurface scattering. Later, corrected diffusion approximation and photon beam diffusion are studied by Lehtikangas et al. [6] and Habel et al. [7] respectively. They can capture the subsurface scattering of many common materials in the context of film-quality rendering, but they cannot meet the requirement of fast interactive rendering. Considering the development of GPU rendering technique, Chang et al. [8] present a novel approach for realistic real-time rendering of translucent surfaces. The computation of subsurface scattering via a hybrid scheme is able to steadily render the translucent effect in real time with a fixed amount of samples. They finally give the feasible GPU implementation and make real-time frame rate possible. However, the rendering speed of their method is still relatively low when the number of sample points is increasing greatly. More recently, Frisvad et al. [9] present an improved analytical model for subsurface scattering that captures translucency effects. Specially, for realistic rendering for human skin, Ma et al. [10] also design an algorithm based on subsurface scattering and gamma correction, and obtain real-time rendering via GPU rendering pipeline. Nowadays, for 3D real-time game or other fast interactive applications, however, these methods mentioned above still acquire a relatively low speed of rendering. Motivated by overcoming the limitation, our research focuses on faster GPU-based rendering while keeping rendering quality. We derive the diffusion equation to simulate the complex subsurface scattering and finally give a discrete form of BSSRDF to fit GPU evaluation easily. Also to speed up the rendering and acquire a comparable rendering quality, we improve classic Turk's point repulsion algorithm by the geometric feature sampling to spread out the sample points reasonably.

The rest of this paper is organized as follows. The derivation and discretization of improved diffusion equation are given in Section 2. Then implementation of geometric feature sampling is showed in Section 3. At last, experimental results are discussed in Section 4, and finally conclusions are given in Section 5.

2. Derivation and Discretization of Improved Diffusion Equation.

One of the most successful frameworks for approximating the diffuse reflectance profile R_d is the diffusion approximation. It is based on the observation that in highly scattering materials, the light transport and diffusion tend to become isotropic even if the incident light and phase function are anisotropic. The light diffusion process inside scattering materials can be shown in Figure 1.

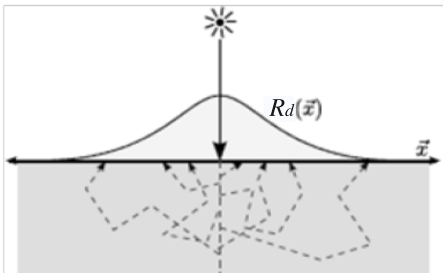

FIGURE 1. Process of light transport and diffusion

The diffusion approximation solves the *radiance transport equation* (RTE) by considering only a first-order spherical harmonic expansion of the radiance:

$$L(\overrightarrow{x}, \overrightarrow{\omega}) \approx \frac{1}{4\pi}\phi(\overrightarrow{x}) + \frac{3}{4\pi}\overrightarrow{E}(\overrightarrow{x}) \cdot \overrightarrow{\omega} \quad (1)$$

where the radiance $L(\overrightarrow{x}, \overrightarrow{\omega})$ is characterized by the current position along the ray $\overrightarrow{x}$ and the direction $\overrightarrow{\omega}$. Fluence ϕ and flux $\overrightarrow{E}$ are the first two moments of the radiance which can be denoted as:

$$\phi(\overrightarrow{x}) = \int_{4\pi} L(\overrightarrow{x}, \overrightarrow{\omega})\, d\overrightarrow{\omega} \tag{2}$$

$$\overrightarrow{E}(\overrightarrow{x}) = \int_{4\pi} L(\overrightarrow{x}, \overrightarrow{\omega}) \cdot \overrightarrow{\omega}\, d\overrightarrow{\omega} \tag{3}$$

By substituting Equation (1) into the radiative transport equation with Equation (2) and Equation (3), and integrating over all directions, we can derive the diffusion equation:

$$-D\nabla^2\phi(\overrightarrow{x}) + \sigma_a\phi(\overrightarrow{x}) = Q(\overrightarrow{x}) \tag{4}$$

where $D = 1/3\sigma_t'$ is diffusion constant coefficient and σ_t' is the reduced extinction coefficient. σ_a is absorption coefficient of one material. Q is an isotropic source term.

Additionally, we can formulate the diffuse reflectance profile R_d in terms of the fluence $\phi(\overrightarrow{x})$. R_d is defined as the radiant exitance divided by the incident flux. Using Fick's law, it states that (for isotropic sources) the vector flux is the gradient of the fluence:

$$\overrightarrow{E}(\overrightarrow{x}) = -D\nabla\phi(\overrightarrow{x}) \tag{5}$$

Since the radiant exitance on the boundary is the dot product of the vector flux with the surface normal, we have

$$R_d(\overrightarrow{x}) = \overrightarrow{E}(\overrightarrow{x}) \cdot \overrightarrow{n} = -D(\nabla \cdot \overrightarrow{n})\phi(\overrightarrow{x}) \tag{6}$$

where n is the normal at the point of incoming light.

For a unit power point light in an infinite medium, the fluence is given by the diffusion Green's function, and we can now derive the fluence for a semi-infinite medium. If the point light source is at a distance z_r below the surface, we place the image source at a distance $z_v = z_r + 4AD$ above the surface. A can be computed as $A = (1 + F_{dr})/(1 - F_{dr})$, where F_{dr} is the diffuse fresnel reflectance. The resulting *dipole fluence* $\phi(\overrightarrow{x})$ is then:

$$\phi(\overrightarrow{x}) = \frac{1}{4\pi D}\left(\frac{e^{-\sigma_{tr}d_r}}{d_r} - \frac{e^{-\sigma_{tr}d_v}}{d_v}\right) \tag{7}$$

where $\sigma_{tr} = \sqrt{\sigma_a/D}$ is the transport coefficient. $d_v = \| \overrightarrow{x} - \overrightarrow{x}_v \|$ and $d_r = \| \overrightarrow{x} - \overrightarrow{x}_r \|$ are distances of $\overrightarrow{x}$ to the image source and the point light source, respectively. The dipole model is illustrated in Figure 2.

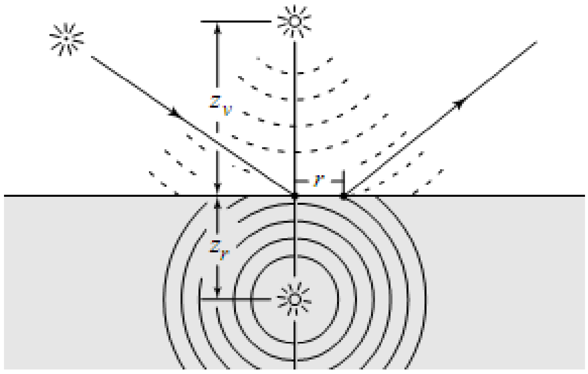

FIGURE 2. Dipole model

Finally, substituting the resulting dipole fluence into Equation (6) with Equation (7), we yield a closed form approximation for the diffuse reflectance:

$$R_d(\overrightarrow{x}) = \frac{\alpha'}{4\pi}\left\{z_r\left(\sigma_{tr} + \frac{1}{d_r}\right)\frac{e^{-\sigma_{tr}d_r}}{d_r^2} + z_v\left(\sigma_{tr} + \frac{1}{d_v}\right)\frac{e^{-\sigma_{tr}d_v}}{d_v^2}\right\} \tag{8}$$

where $\alpha' = \sigma_s'/\sigma_t$ is the reduced albedo. σ_s' is the reduced scattering coefficient and σ_t is the extinction coefficient.

Using Equation (8), the Fresnel reshaping term results in the diffusion dipole BSSRDF model as follows, which has been found successful adaption in realistic rendering.

$$S_d(\overrightarrow{x}, \overrightarrow{\omega}_i, \overrightarrow{\omega}_o) = \frac{\rho}{\pi} F_t(\overrightarrow{x}, \overrightarrow{\omega}_i) R_d(\overrightarrow{x}) F_t(\overrightarrow{x}, \overrightarrow{\omega}_o) \tag{9}$$

where F_t represents the Fresnel transmission term, and ρ is an approximate normalization factor.

We can now obtain the total reflected radiance L_r at surface point $\overrightarrow{x}$ due to the BSSRDF through integration:

$$L_r(\overrightarrow{x}, \overrightarrow{\omega}_o) = \int_A S_d(\overrightarrow{x}, \overrightarrow{\omega}_i, \overrightarrow{\omega}_o) I(\overrightarrow{x}, \overrightarrow{\omega}_i)(\overrightarrow{\omega}_i \cdot n) d\overrightarrow{x} \tag{10}$$

where I is the initial irradiance.

In the actual implementation of real-time rendering, the model is represented by the discrete 3D mesh. Accordingly, Equation (10) could be rewritten as a discrete sum function which fits GPU evaluation easily as follows.

$$L_r(\overrightarrow{x}, \overrightarrow{\omega}_o) = \sum_{x_i \in VS} S_d(\overrightarrow{x}, \overrightarrow{\omega}_i, \overrightarrow{\omega}_o) I(\overrightarrow{x}, \overrightarrow{\omega}_i) G(\overrightarrow{x}) \tag{11}$$

where VS is the vertex set of model surface. G is the micro-facet geometry function.

3. **Implementation of Geometric Feature Sampling.** The diffuse approximation is generally computed in a single pass which consumes a fair amount of time. To speed up the computation, we have presented an effective evaluation by sampling the irradiance. To compute the irradiance on the surface, sample points have to be effectively distributed over the object surface. In this paper, considering geometric features such as curvature and area, we improve classic Turk's point repulsion algorithm by using a *geometric feature sampling* to reasonably spread out the sample points over the surface.

We firstly set a random sampling of points on the mesh, and compute the curvature and area of each mesh triangle. Then we sort triangles of 3D model according to their curvature and area. Later we select triangles with larger curvature and area to distribute new points on the center of gravity. If curvatures of three vertexes of the triangle are $Curvature[i]$ and their coordinate vectors are $Coord[i]$, the coordinate of new point is as follows.

$$newCoord = (Curvature[i] \times Coord[i]) \Big/ \sum_{i=1}^{3} Curvature[i] \tag{12}$$

The geometric feature method distributes amounts and locations of sample points by depending on the surface area and curvature of each triangle, which can be implemented as shown in Algorithm 1.

Once the sample points are effectively spread on the surface, we can use global illumination techniques such as photon mapping or irradiance caching to obtain the irradiance at these particular points.

4. **Rendering Results.** Realistic rendering of translucent materials such as 3D rabbit model has been implemented in this section. We tested our method based on a SAMSUNG370R5V PC with Intel Core 2 i5-3210M @ 2.50GHz CPU and AMD Radeon HD 8750M graphics card. Our final rendering clearly shows the effective implementation of our technique.

To obtain the irradiance at specific points on the objects surface, we improve classic Turk's point repulsion method via geometric feature method firstly. We vary both sampling method as well as the number of samples used for each illumination to demonstrate their influence on the 3D rabbit model. Figure 3 and Figure 4 show two rendering scenes via different sampling methods.

Algorithm 1 New points distribution using geometric feature method

Require:

Original 3D mesh model, $originalMesh$;

Numbers of new distribution point, $2n$;

Ensure:

New mesh model after points distribution, $newModelMesh$;

1: read the original mesh of our model, $originalMesh$;
2: evaluate $Curvature$ of each point;
3: select n triangles with bigger $Curvature$ value;
4: evaluate $Area$ of each point;
5: select n triangles with bigger $Area$ value;
6: distribute a new point on the weighted gravity center;
7: for{i=1 to N}{
8: $newCoord \leftarrow (Curvature[i] \times Coord[i]) / \sum Curvature[i]$;
9: }
10: obtain the $newModelMesh$ after new point distribution;
11: **return** $newModelMesh$.

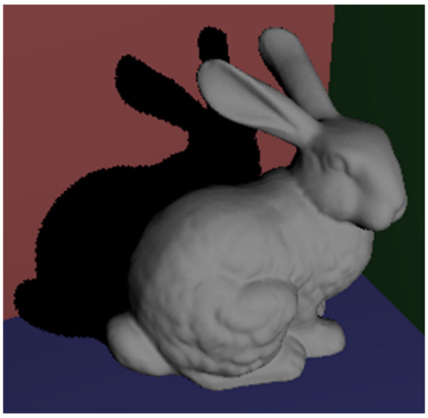

Figure 3. Rendering via Turk's method

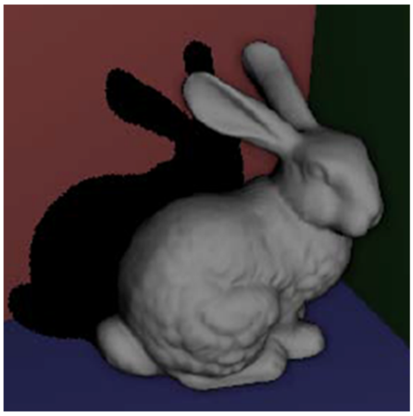

Figure 4. Rendering via our method

From Figure 3 and Figure 4, we can observe that the rendering result using our sampling method is more realistic than Turk's sampling method. The effect of lighting and shading via our method is more obvious and clearer, which shows our sampling technique is more effective.

After the sample points have been effectively selected, the irradiance can be calculated by sampling all the light sources and integrating over the hemisphere. Figure 5 and Figure 6 show two rendering scenes via Chang et al.'s method [8] and our method.

From Figure 5 and Figure 6, using our method we can clearly observe that the environment light and diffused light can truly affect the appearance of the model, and the model is more naturally integrated into the surroundings. Also the soft shadow can be rendered effectively, which gives a much more realistic scene.

In the performance of rendering speed, using GPU rendering pipeline we have achieved about 66 frames per second (FPS) when sampling 400 points; however, the real-time rendering method such as Chang et al.'s method [8] only acquires 22 FPS. When increasing sample points greatly, the difference of rendering time between our method and Chang et al.'s method [8] is more obvious as shown in Figure 7, which shows that we are able to achieve better quality with faster speed.

5. **Conclusions.** Using improved diffusion equation and geometric feature sampling, we have implemented a fast rendering via GPU rendering pipeline and given a more realistic

FIGURE 5. Rendering via Chang et al.'s method

FIGURE 6. Rendering via our method

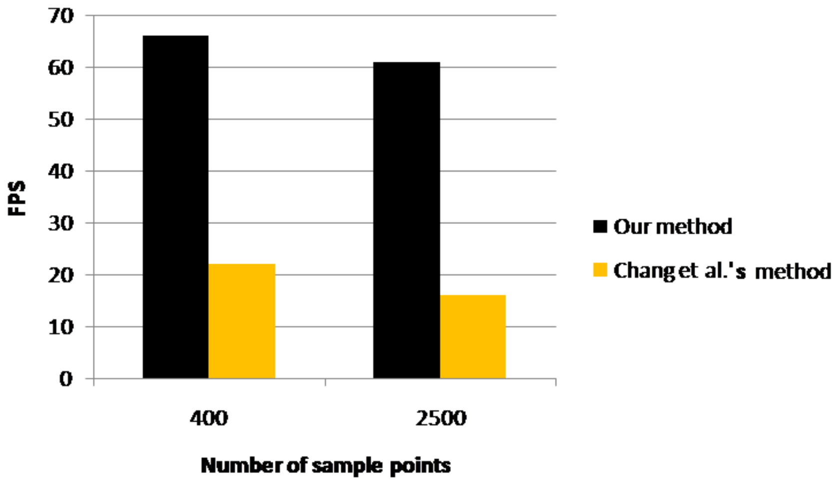

FIGURE 7. FPS comparison between different methods when sampling different points

appearance. The rendering results show that our sampling algorithm can acquire more obvious effect of lighting and shading, which proves our sampling technique is more effective than Turk's sampling algorithm. Also we can clearly observe that the rendering appearance of our method is more naturally integrated into the surroundings. In the performance of rendering speed, we are able to achieve better quality with faster speed when increasing sample points greatly.

In future, to bring together physically plausible illumination of dynamic complex scenes and real-time rates, the optimal combination of improved approaches such as dynamic environment lighting is an interesting research direction.

REFERENCES

[1] H. W. Jensen, S. R. Marschner, M. Levoy and P. Hanrahan, A practical model for subsurface light transport, *Proc. of SIGGRAPH*, pp.511-518, 2001.

[2] H. W. Jensen and J. Buhler, A rapid hierarchical rendering technique for translucent materials, *Proc. of SIGGRAPH*, pp.576-581, 2002.

[3] V. Sundstedt, J. Jimenez and D. Gutierrez, Screen-space perceptual rendering of human skin, *ACM Trans. Applied Perception*, vol.6, no.4, pp.1-15, 2009.

[4] J. Wang, S. Zhao, X. Tong, S. Lin, Z. Lin and Y. Dong, Modeling and rendering of heterogeneous translucent materials using the diffusion equation, *ACM Trans. Graphics*, vol.27, no.1, pp.329-339, 2008.

[5] A. Arbree, B. Walter and K. Bala, Heterogeneous subsurface scattering using the finite element method, *IEEE Trans. Visualization and Computer Graphics*, vol.17, pp.956-969, 2011.

[6] O. Lehtikangas, T. Tarvainen and A. D. Kim, Modeling boundary measurements of scattered light using the corrected diffusion approximation, *Biomedical Optics Express*, vol.3, no.3, pp.552-571, 2012.

[7] R. Habel, P. H. Christensen and W. Jarosz, Photon beam diffusion: A hybrid monte carlo method for subsurface scattering, *Computer Graphics Forum*, pp.27-37, 2013.

[8] C. Chang, W. Lin, T. Ho, T. Huang and J. Chuang, Real-time translucent rendering using GPU-based texture space importance sampling, *Computer Graphics Forum*, vol.27, pp.517-526, 2008.

[9] J. R. Frisvad, T. Hachisuka and T. K. Kjeldsen, Directional dipole model for subsurface scattering, *ACM Trans. Graphics*, vol.34, no.1, pp.1-12, 2014.

[10] Z. Ma, N. Li and M. Yang, Realistic rendering of human skin based on subsurface scattering and gamma correction, *ICIC Express Letters, Part B: Applications*, vol.6, no.6, pp.1485-1492, 2015.