Position Free Monte Carlo for layered material rendering

ARPIT AGARWAL, Robotics Institute, CMU

Fig. 1. The project implementation used position free monte carlo for layered materials which allows to use anisotropic roughness, textured absorption and anisotropic medium with fast rendering times. Please see the image for the description of the materials.

Real world objects are often coated with multiple layers on top the base color to protect from scratches or to provide rich diversity of material appearance like spatially varying specular highlights. The idea of layered material modeling tries to circumvent explicitly calculating light transport with each coating layer and replace it with BSDF which gives equivalent reflection and transmission component, albeit with some assumption. Past approaches have tackled the problem of modeling layered material by either creating a concise representation of captured data or by precomputing a low-order statistics of the material. My project is based on [Guo et al. 2018] work on using position free Monte Carlo with plane parallel layers with participating media to provide the BRDF. In my project, I have implemented their unidirectional evaluation and approximate PDF approach to show the concept. The report highlights the main algorithm and benefits of the approach.

CCS Concepts: • **Computing methodologies → Rendering**; **Ray tracing**.

Additional Key Words and Phrases: ray tracing, layered materials, BSDF

ACM Reference Format:
Arpit Agarwal. 2017. Position Free Monte Carlo for layered material rendering. *ACM Trans. Graph.* 36, 4, Article 1 (July 2017), 5 pages. https://doi.org/http://dx.doi.org/10.1145/8888888.7777777

Author's address: Arpit Agarwal, Robotics Institute, CMU.

0730-0301/2017/7-ART1 $15.00
https://doi.org/http://dx.doi.org/10.1145/8888888.7777777

1 INTRODUCTION

Creating realistic images require 2 key inputs, namely, geometry and material. Material modeling is a key challenge in computer graphics. Various past works have characterized and compared materials surface appearance using BRDF, BSDF and sub-surface scattering.

My project implements following things

(1) Monte Carlo Position-Free sampling for plane parallel interface with volume scattering
(2) Unidirectional evaluation of layered BSDF with anisotropic volume scattering
(3) Approximate PDF estimation using rough interfaces
(4) BSDF with more than 2 layers with volume scattering in between each layer

2 PRIOR WORK

Discretized layered BSDFs: [Jakob et al. 2014] and [Zeltner and Jakob 2018] derived accurate BRDF representation using precomputation in direction in Fourier domain. There representation is very accurate and allows adding-doubling algorithms. [Zeltner and Jakob 2018] also introduced subtracting 2 BSDFs through their approach. However, these approaches require one to know the BSDF before the rendering begins and can be very computationally intensive.

Analytic layered BSDFs: [Weidlich and Wilkie 2007] combines multiple BSDFs where sub-surface scattering is absent to derive a multi-lobed BSDF. [Belcour 2018] track a low-order statistics of the BSDF across layers and fit an equivalent microfacet BSDF using the obtained combined statistics. These approaches are fast but can't handle anisotropic medium.

3 METHODOLOGY

3.1 Assumptions

The key assumption in the approach is as follows

(1) The light scattering inside the material is independent of position of intersection point. Only the depth inside the material is required to calculate the contribution
(2) The incoming and outgoing light leave from the same microfacet location and small displacement at the interface points could be ignored.

The proposed approach of using Position-Free Monte Carlo is proven to be unbiased, supports spatially varying parameters and is faster than global Monte Carlo methods.

3.2 Background

Using the path integral formulation from Veach's Thesis [Veach 1997], the pixel value is given as

$$I = \int_{\Sigma} f(\bar{x}) d\mu(\bar{x}) \tag{1}$$

where $\bar{x} = (\mathbf{x}_0, \cdots, \mathbf{x}_k)$ is the path with k segments and $k+1$ vertices on the surfaces or within participating media, Σ is the space of all the paths for $k \geq 0$. $f(\bar{x})$ is the path contribution and $\mu(\bar{x})$ is the area measure. The path contribution is given as follows

$$f(\bar{x}) = W_e(\mathbf{x}_0, \mathbf{x}_1) L_e(\mathbf{x}_k, \mathbf{x}_{k-1}\, T(\bar{x}) \tag{2}$$

where W_e and L_e are camera vertex and light vertex contribution respectively. $T(\bar{x})$ is the path throughput which contains BSDF(phase function) term, Geometry term and visibility term. Visibility in general scenario is the most costly process in the rendering. Geometry term introduces high variance as it includes inverse distance between the scene points.

3.3 Overview

The main quantity to calculate inside the layered material is $L(\mathbf{x}, \omega)$, where $\mathbf{x}$ is the point on each interface inside the material or in the volume and ω is the direction of outgoing light. Due to the assumption of position free, the problem reduces to calculation of $L(z, \omega)$ inside the layered media, which can be done very efficiently for anisotropic and spatially varying layer properties at run time. The following section describes the BSDF implementation which reduces to 3 key operations, namely, *sampling* incoming direction given outgoing direction, *evaluating* the BSDF of the layered material for given incoming ω_i and outgoing ω_o direction and *estimating probability* of sampling given incoming ω_i and outgoing ω_o direction. All the following equations are given with 2 interfaces with single medium combination. The implementation also implements this configuration. The multilayered setup is obtained by replacing the interfaces with another mutlilayered BSDF.

3.3.1 BSDF Sampling. This operation is required to generate the new direction for path generation for a given outgoing direction ω_o. This operation includes running position free volume rendering with simplified ray intersection operation in which only the depth of the point inside the medium matters. This process doesn't require any special data structure creation as the intersection point can be found in closed form given the input ray direction.

3.3.2 BSDF Evaluation. The BSDF contribution of sampled incoming ω_i and outgoing ω_o direction is calculated using this operation. The path contribution $f(\bar{x})$ for this case reduces to the following equation

$$f(\bar{x}) = v_1 s_1 v_2 s_2 \ldots s_{k-1} v_k \tag{3}$$

The vertex term is given as follows

$$v_i = v(z_i, -\omega_{i-1}, \omega_i) = \begin{cases} f_{\text{top}}(-\omega_{i-1}, \omega_i) & \text{if } z_i = 0 \\ f_{\text{bottom}}(-\omega_{i-1}, \omega_i) & \text{if } z_i = 1 \\ \sigma_s f_{\text{medium}}(-\omega_{i-1}, \omega_i) & \text{if } 0 < z_i < 1 \end{cases} \tag{4}$$

Let the transfer term $\tau(z, z', \omega)$ be given as follows

$$\tau(z, z', \omega) := exp\left(\frac{-\sigma_t |z' - z|}{|cos\,\omega\,|}\right) \cdot \mathbb{I}\left(\frac{z' - z}{cos\,\omega} > 0\right) \tag{5}$$

The segment term is defined as follows

$$s_i = s(z_i, s_{i+1}, \omega_i) =:= \tau(z_i, z_{i+1}, \omega_i) \times |cos\,\omega_i\,|^{\alpha_i} \tag{6}$$

where $\alpha_i = \mathbb{I}(z_i \in \{0,1\}) + \mathbb{I}(z_{i+1} \in \{0,1\}) - 1$. Please refer to the paper [Guo et al. 2018] for more discussion and derivation of the above equation. Paper specifically discusses the cosine terms in α to make it absolutely clear.

3.3.3 BSDF PDF Estimation. The operation of calculating probability for incoming ω_i and outgoing ω_o direction to calculate MIS weights during *Next Event Estimation* and weighting the contribution for the sampled path using BSDF sampling in standard volume path integral formulation. The paper proposes 2 approaches for this operation, namely, *unbiased PDF estimation* and *Approximate PDF estimation*. The theory for *unbiased PDF estimation* is similar to path contribution calculation with path contribution replaced with path pdf calculation. However, I implemented only the *Approximate PDF estimation* method as the paper showed that this method is exceedingly fast and leads to non-perceptible difference in the final images. Note: approximation is bad for grazing angles as can seen in Figure .

4 OTHER IMPLEMENTATIONS

For the final project, I also implemented following light models. However, I have not used them in my final rendering as the BSDF evaluation currently only supports unidirectional light transport. However, this is an implementation bug and should be resolved in future.

In the following sections, I describe the lights models and show a illustrative rendering using the lights models.

4.1 IES light model

Real light sources do not illuminate the scene as traditional light models like point light sources and area light sources, available in various renderers, emit light. Various manufacturers provide an IES profile, an Illuminating Engineering Society standard format, for real light sources. This becomes very important for fabrication and engineering related tasks in which light sources might be very close the scene to be illuminated. The IES light model assumes the light source to be a point light source with intensity along each outgoing direction parameterized by spherical coordinates (θ, ϕ) be given using a texture. Figure 2 shows examples of IES profile and corresponding illumination along the wall. Figure 3 shows the image illuminated with 2 IES profiles shown 2 from the top. The area of the head exactly below the light receives almost no light from the second IES light profile.

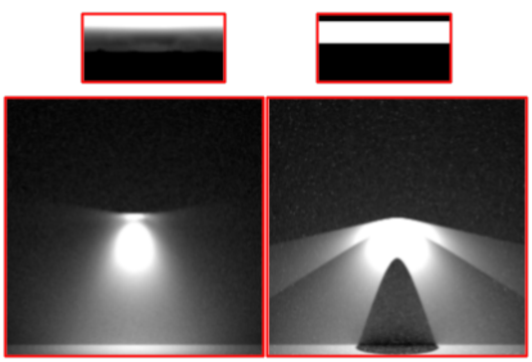

Fig. 2. *IES Profile*: The image show a wall illuminated by a IES light source pointing downwards with the IES profiles shown at the top. The images were rendering using BDPT in Mitsuba.

4.2 Directionally Varying Area light

Though IES light model allows one to model real light sources with fidelity, it does bode well for rendering algorithms which require connecting paths generated from points in the scene to connect to the light source. For this reason, we can combine the benefits of area light source and Spot light to simulate what I term as *Directionally Varying Area Light* or *DVAreaLight*. This light model has 2 parameters, namely, *cutoffAngle* and *beamWidth* which are used to control the cone of light emitted from each point on the area light source. Figure 4 shows the light model illuminating a wall. Figure 3 shows the monkey illuminated from the right using area light source and *DVAreaLight* respectively.

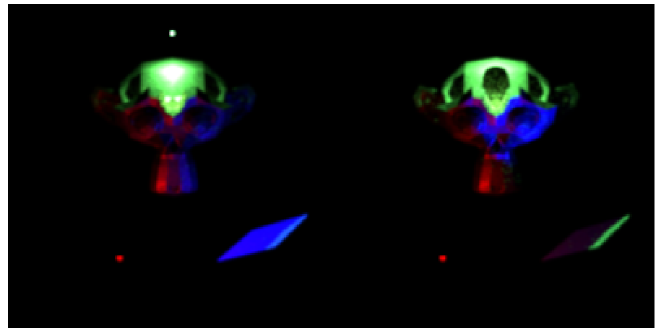

Fig. 3. Effect of light models on the illumination of monkey face.

Fig. 4. Directionally Varying Area Light Model: The left figure shows a wall illuminated by the normal area light source and right figure shows the wall illuminated using *DVAreaLight* with *cutoffAngle*=20 and *beamWidth*=15

5 RESULTS

For comparison, I used the reference implementation by the original authors.

5.1 BSDF Operation Validation

Figure 5 shows the polar plot of the unidirectional BSDF evaluation step from reference and my implementation for material as shown in Figure 8. There is a close match across channel.

Figure 6 shows the polar plot of the approximate PDF step from reference and my implementation for material as shown in Figure 8. There is a constant shift in the PDF estimation and potential source of bug.

5.2 Full Image Rendering Validation

Figure 7 shows the images of a flat plane coated with different layered materials illuminated by a big spherical light and camera pointing directly towards it. The exact description of the material

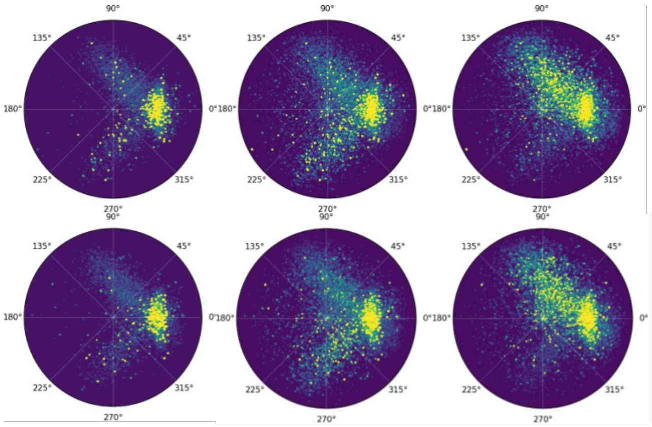

Fig. 5. **BSDF Evaluation comparison**: The top and bottom row shows the comparison of BSDF evaluation operation for image shown in 8 with reference implementation and my implementation. The left, middle and right column show the red, green and blue channel of the BSDF respectively.

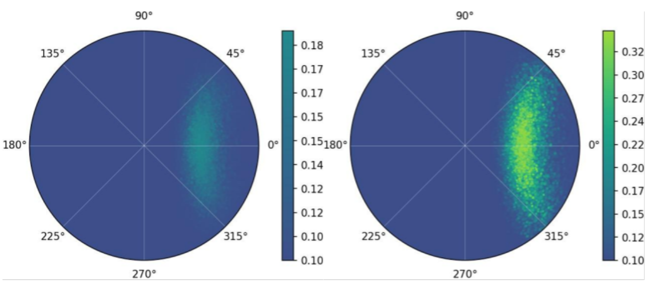

Fig. 6. **BSDF Approximate PDF comparison**: The left and right image shows the reference implementation and my implementation approximate PDF estimation. The trend looks the same. However, there is a constant shift.

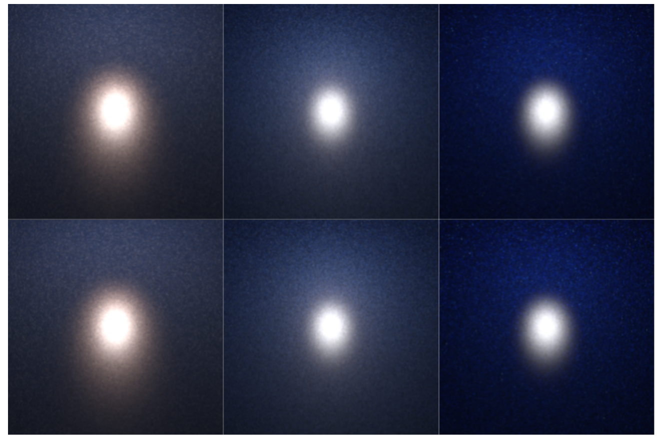

Fig. 7. The top and bottom row show the images rendered using reference and my implementation respectively. The images show a plane coated with layered materials. The medium scattering increases from left to right column. For exact description please refer to text.

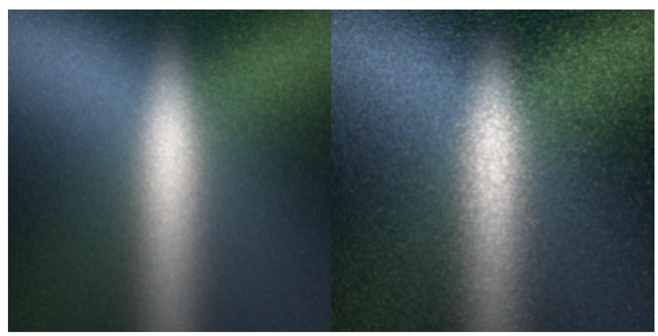

Fig. 8. The left and right images are rendering using reference and my implementation respectively.

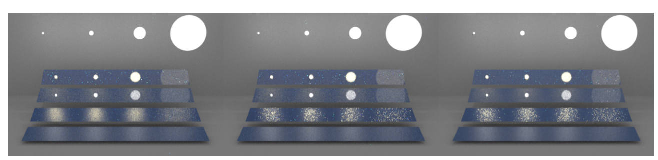

Fig. 9. Multiple Importance Sampling: The left and middle images are rendered using reference implemention with and without MIS for BSDF evaluation. The right image shows image from my implementation which does not have MIS implementation in BSDF evaluation. Note: this MIS is different than global path tracing MIS which is available in all the images shown above. All the images were rendered using 64spp

is available in the original paper supplementary. This scene was very helpful in debugging the implementation as it the only interaction happening is inside the BSDF itself and almost negligible intersection time. The figure shows that my implementation is able to match the scene with homogeneous and heterogeneous medium with the reference implementation. My implementation times were higher than the reference implementation.

Figure 8 shows the previous scene with flat slab coated with 3 layered material. Again, for the exact material details please refer to the original paper supplementary. This helped to test if the nested BSDF model implementation is correct.

Figure 9 shows the effect of *Multiple Importance Sampling* in the BSDF evaluation implementation. In my project, I did not implement MIS which leads to noisy images at the lower slabs. However, the rendering matches the reference implementation without MIS and is inferior to image with MIS. This part is very easy to implement. I expect to include this in the near future.

Figure 10, shows rendering of a 3 layered BSDF with following properties

(1) *interface0* rough dielectric layer with $\eta = 1.5$ and $\alpha = 0.05$
(2) anisotropic *medium0* with density=0.5, albedo=(0.5 0.7 0.95) and orientation=(0.5,0.5,0) and microflake phase function with stddev=0.05
(3) *interface1* rough dielectric layer with $\eta_{exterior} = 1.5$, $\eta_{interior} = 1.2$ and $\alpha = 0.01$

(4) anisotropic *medium1* with density=0.5, albedo=(0.5 0.95 0.5) and orientation=(0.5,-0.5,0) and microflake phase function with stddev=0.05
(5) *interface2* rough conductor Cu with roughness=0.05

Note this case is not handled well by data-driven [Zeltner and Jakob 2018] and analytic [Belcour 2018] approaches.

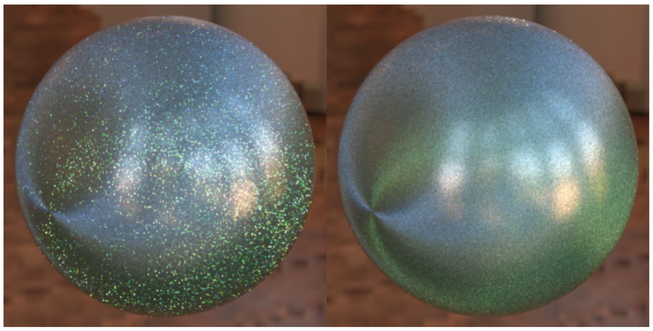

Fig. 10. The left and right images are rendering using reference and my implementation respectively. Please refer to text for detailed description of the material.

6 FINAL RENDERING

Figure 1 shows my entry into the rendering competition. The image was inspired by original paper's teaser image [Guo et al. 2018]. The image contains 4 objects highlighting various strengths of the approach.

First object on the left shows a sphere coated with a 3 layered material with anisotropic roughness at interface and anisotropic medium with *microflake* phase functions.

Second object on the right was designed to show the specular highlights and simplest case of 2 layer material with rough interfaces and homogeneous medium with *HG* phase function.

Third object geometry was taken from [Zeltner and Jakob 2018] paper supplementary material coffee scene. The layered material has textured absorption at the last conductor layer due to which a greenish color starts to appear on the left part of the object.

Fourth object contains the anisotropic rough interface dielectric at the top, homogeneous media and very rough conductor at the bottom.

7 CONCLUSIONS AND FUTURE WORK

The current approach works well for layered materials with anisotropy and spatially varying arbitrary BSDF property without any expensive precomputation. The proposed approach is also shown to be unbiased and has a bidirectional extension. One of the assumption of ignoring the displacement of light scattering leaving the medium was violated noticeable number of times during implementation debugging and rendering test scenes. This case occurred specifically during volume rendering or if a large number of light bounces occurred. The second case would have negligible contribution to the BSDF value. However, the careful inspection of the first might be useful. Together with future work mentioned in the original paper, the paper has been extended to include sampling using multiple terms BSDF is explored in [Xia et al. 2020] with the assumption of modeling BSDF using gaussians. This is an exciting area of future research.

ACKNOWLEDGMENTS

The reference implementation from the original authors [Guo et al. 2018] and their corresponding scene files were very helpful in debugging my implementation. My implementation used Mitsuba [Jakob 2010] for global light transport and single layer BSDFs.

REFERENCES

Laurent Belcour. 2018. Efficient rendering of layered materials using an atomic decomposition with statistical operators. *ACM Transactions on Graphics* 37, 4 (2018), 1.

Yu Guo, Miloš Hašan, and Shuang Zhao. 2018. Position-free monte carlo simulation for arbitrary layered bsdfs. *ACM Transactions on Graphics (ToG)* 37, 6 (2018), 1–14.

Wenzel Jakob. 2010. Mitsuba renderer. http://www.mitsuba-renderer.org.

Wenzel Jakob, Eugene d'Eon, Otto Jakob, and Steve Marschner. 2014. A comprehensive framework for rendering layered materials. *ACM Transactions on Graphics (ToG)* 33, 4 (2014), 1–14.

Eric Veach. 1997. *Robust Monte Carlo methods for light transport simulation*. Vol. 1610. Stanford University PhD thesis.

Andrea Weidlich and Alexander Wilkie. 2007. Arbitrarily layered micro-facet surfaces. In *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*. 171–178.

Mengqi Xia, Bruce Walter, Christophe Hery, and Steve Marschner. 2020. Gaussian Product Sampling for Rendering Layered Materials. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 420–435.

Tizian Zeltner and Wenzel Jakob. 2018. The layer laboratory: a calculus for additive and subtractive composition of anisotropic surface reflectance. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–14.