Real-Time Image-Based Volume Lighting

Cyril Delalandre Pascal Gautron Jean-Eudes Marvie
Technicolor

(a) 512^3 cloud - 10.7fps (b) Animated thick smoke 128^3 - 20.8fps (c) Animated thin smoke 128^3 - 20.8fps

Figure 1: *Our solution computes single scattering due to distant lighting using an optimized spherical harmonics projection of the scattered radiance.*

1. Introduction

The interaction of translucent objects with light creates complex effects such as scattering, absorption and volumetric shadows. While most accurate rendering approaches resort to heavy computations, recent needs in interactive applications have led to new types of algorithms, trading quality or genericity for speed.

We propose a two-step, GPU-friendly technique for real-time rendering of heterogeneous participating media under distant environment lighting (Figure 1). First, our algorithm estimates the spherical scattered radiance at a number of points in the medium and projects this function into the spherical harmonics basis. In the second step we render use the scattered radiance information to compute single scattering by ray-marching. Our method is easy to implement using GPU shaders and does not require any precomputation, hence supporting dynamic lighting, animated media, dynamic optical properties of the volume, emission and self-shadowing.

2. Related Work

Rendering participating media remains a widely studied topic [CFP*05]. Some of these techniques generate high quality images using offline stochastic techniques such as Monte Carlo path tracing [Sta94] or variations on photon mapping [JNSJ11]. To achieve interactive or real-time performance, some techniques introduce restrictive assumptions by considering homogeneous media [WR08] or point lighting [DGMF11]. In the real world objects are lit by their entire environment. To enhance realism of rendered media some approaches [ZRL*08] simulate distant lighting to illuminate the medium. After a computationally-intensive volume analysis, this technique renders predetermined volumes in real-time. Another interactive technique [NGS09] leverages medium sparsity for interactive lighting from a constellation of point light sources. The proposed concepts of distance function and validity masks could be merged with our technique for further performance.

Spherical harmonics projection has been extensively used in computer graphics in the last decade. In particular, the Precomputed Radiance Transfer techniques initiated by Sloan *et al.* [SKS02] have given rise to numerous solution for real-time lighting under low-frequency environments. However, most methods share the need for long, model-dependent precomputations. Our method leverages techniques similar to PRT, while avoiding precomputations.

3. Technical Background

3.1. Single scattering

The interaction between light and participating media is fully described by the radiative transport equation [Cha50]. At each point $\mathbf{p}$ a participating medium is described by its coefficients of absorption $\sigma_a(\mathbf{p})$ and scattering $\sigma_s(\mathbf{p})$, as well as its extinction $\sigma_t(\mathbf{p}) = \sigma_a(\mathbf{p}) + \sigma_s(\mathbf{p})$. The amount of light

DOI: 10.2312/conf/EG2013/short/057-060

scattered at $\mathbf{p}$ from the incoming direction $\omega_\mathbf{i}$ towards $\omega_\mathbf{o}$ is given by the phase function $p(\mathbf{p}, \omega_\mathbf{o}, \omega_\mathbf{i})$.

The radiance reaching a point $\mathbf{c}$ (Figure 2) due to single scattering along a direction $\omega_\mathbf{o}$ is given by integrating the scattering events occurring between $\mathbf{p_{in}}$ and $\mathbf{p_{out}}$ as follows:

$$L(\mathbf{p_{in}}, \omega_\mathbf{o}) = \int_{\mathbf{p_{in}}}^{\mathbf{p_{out}}} R_s(\mathbf{p_n}, \omega_\mathbf{o}) e^{\int_{\mathbf{p_{in}}}^{\mathbf{p_n}} -\sigma_t(\mathbf{p})d\mathbf{p}} d\mathbf{p_n} \quad (1)$$

where $e^{\int_{\mathbf{p_{in}}}^{\mathbf{p_n}} -\sigma_t(\mathbf{p})d\mathbf{p}}$ is the attenuation along $|\mathbf{p_n p_{in}}|$. $R_s(\mathbf{p_n}, \omega_\mathbf{o})$ is the radiance scattered at $\mathbf{p_n}$ towards $\omega_\mathbf{o}$:

$$R_s(\mathbf{p_n}, \omega_\mathbf{o}) = \sigma_s(\mathbf{p_n}) \int_\Omega p(\mathbf{p_n}, \omega_\mathbf{o}, \omega_\mathbf{i}) L(\omega_\mathbf{i}) e^{\int_{\mathbf{p_n}}^{\mathbf{k_{in}}} -\sigma_t(\mathbf{k})d\mathbf{k}} d\omega_\mathbf{i} \quad (2)$$

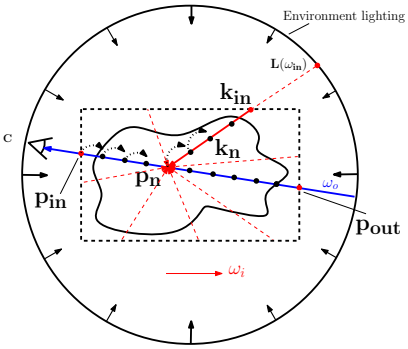


Figure 2: *Notations and principle of a ray-marching single scattering estimation within a bounded a participating medium. The light reduced intensity of the environment map is estimated for each direction $\omega_\mathbf{o}$ along $[\mathbf{k_{in}}, \mathbf{p_n}]$.*

As those integral equations cannot be solved analytically nor numerically in a reasonable time, a classical approach consists in caching directional data using spherical harmonics.

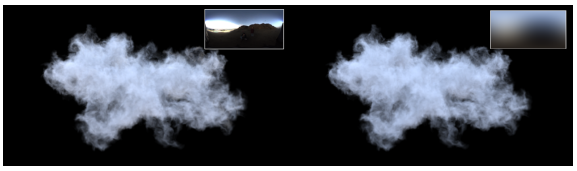
3.2. Spherical Harmonics

Real spherical harmonics define an orthonormal functional basis over the sphere:

$$Y_l^m(\theta, \phi) = \begin{cases} \sqrt{2} K_l^m \cos(m\phi) P_l^m(\cos(\theta)) & m > 0, \\ \sqrt{2} K_l^m \sin(|m|\phi) P_l^{|m|}(\cos(\theta)) & m < 0, \\ \sqrt{2} K_l^0 P_l^0(\cos(\theta)) & m = 0 \end{cases} \quad (3)$$

where P_l^m are the associated Legendre polynomials and K_l^m is the normalization constant. This extends the principle of Fourier transforms to the spherical domain, allowing the representation of a spherical function f as a vector of coefficients $\mathbf{f}$. The orthonormality of the SH basis reduces the dot product of two functions f and g to:

$$\int f(s)g(s)ds = \sum_{l=0}^{N} \sum_{m=-l}^{m=l} \mathbf{f}_l^m \mathbf{g}_l^m \quad (4)$$



(a) (b)

Figure 3: *Offline rendering of a volume under a fully detailed environment map (a) and its reconstruction from 9 SH coefficients (b). As scattering smoothes out high frequency lighting details both images remain visually similar.*

Similarly, the *triple product* of functions f, g, and h represented by their SH vectors $\mathbf{f}$, $\mathbf{g}$ and $\mathbf{h}$ is:

$$\int f(s)g(s)h(s)ds = \sum_i \sum_j \sum_k \mathbf{f}_i \mathbf{g}_j \mathbf{h}_k C_{ijk} \quad (5)$$

where $C_{ijk} = \int_\Omega Y_i(\omega) Y_j(\omega) Y_k(\omega) d\omega$. Despite the sparsity of C, the costs of the triple product quickly become prohibitive in real-time applications.

4. Our algorithm

The scattered radiance equation features three spherical functions: the incoming radiance $L(\omega_\mathbf{i})$, the radiance attenuation $e^{(\int_{\mathbf{p_n}}^{\mathbf{k_{in}}} -\sigma_t(\mathbf{k})d\mathbf{k})}$ and the phase function $p(\mathbf{p}, \omega_\mathbf{o}, \omega_\mathbf{i})$.

Using spherical harmonics for each component, the scattered radiance equation is equivalent to the costly triple product shown in Equation 5. Based on this observation we divide Equation 2 into two factors to reduce the computation to a simple *dot product* (Equation 4). The volume-independent factor F_{Ind}, which does not change during rendering and the volume-dependent factor F_{Dep} which varies over time and must be recomputed for each frame.

For clarity our exposition considers isotropic media only. In this case F_{Ind} is the light intensity $L(\omega_i)$ and F_{Dep} corresponds to the product of the reduced intensity by the phase function. We relieve this limitation in Section 4.4.

4.1. Light intensity projection

As in [SKS02] the distant incoming lighting $L(\omega_i)$ is provided by a HDR image projected into SH. As scattering effects tend to act as a low-pass filter on the environment light, our experiments show that even using a low number of coefficients (typically 9) the loss of high frequency details of the environment map remains visually innocuous (Figure 3).

4.2. Reduced Intensity Projection

The remaining factor F_{Dep} represents the *dot product* of the phase function and the transmittance. This function depends on two directions:

$$F_{Dep}(\mathbf{p}, \omega_\mathbf{i}, \omega_\mathbf{o}) = p(\mathbf{p}, \omega_\mathbf{o}, \omega_\mathbf{i}) e^{\int_\mathbf{p}^{\mathbf{k_{in}}} -\sigma_t(\mathbf{k})d\mathbf{k}} \quad (6)$$

The phase function being constant in isotropic media, F_{Dep} collapses to:

$$F_{Dep}(\mathbf{p}, \omega_\mathbf{i}) = \frac{1}{4\pi} e^{\int_\mathbf{p}^{\mathbf{k_{in}}} -\sigma_t(\mathbf{k})d\mathbf{k}} \quad (7)$$

The remainder of Equation 7 is the transmittance between the entry $\mathbf{k_{in}}$ and $\mathbf{p}$. For heterogeneous media, this factor cannot be computed analytically. We thus use a ray-marching based algorithm for each direction $\omega_\mathbf{i}$. The main idea of our method is to cache this reduced intensity factor to amortize the evaluation costs during the rendering step.

Reduced intensity record The projection of the volume-dependent factor F_{Dep} into spherical harmonics must be performed for each frame and may constitute a bottleneck. The numerical integration is then performed using a very low number of samples. The *HEALPix* distribution [GHB*04] not only provides evenly distributed samples around the sphere, but also generates elements along discrete rings of constant latitude. The associated Legendre polynomials can then be evaluated once per row instead of per sample.

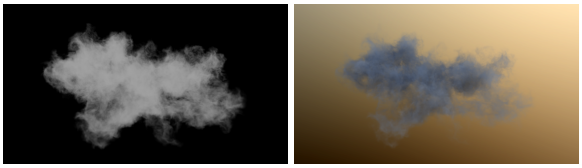
Considering a point $\mathbf{p}$ inside the medium, we compute the intersection point $\mathbf{k_{in}}$ between the bounding box of the medium and each sample ray $(\mathbf{p}, \omega_\mathbf{i})$. The reduced intensity is obtained by sampling the medium along the path $||\mathbf{pk_{in}}||$. The spherical reduced intensity is then projected into spherical harmonics. The coefficients along with the location of $\mathbf{p}$ are then packed into a *reduced intensity record*.

Record grid Generating a reduced intensity record containing 9 half-precision coefficients for each voxel of a 512^3 voxel grid would occupy an impractical 2.25 GB of graphics memory. Instead, we subdivide the bounding box of the medium into a coarse uniform grid. As light scattering can be considered as a diffusion process [Sta95], the reduced intensity tends to vary relatively slowly across a medium. We thus assign the same set of spherical harmonics coefficients for each point in a grid cell, the center of the cell being the origin of the projection. We implement the cell grid using multiple 2D-floating point textures where each texel of the texture contains the set of coefficient of a cell.
The size of the cell influences the rendering quality: using small cells size converges towards the reference solution at the expense of computational efficiency (Figure 4).

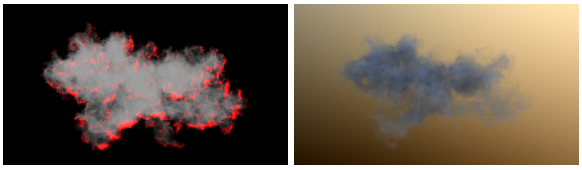
Depending on the configuration of the medium several cells may only cover empty voxels. Further speedup is then achieved by aggressively eliminating empty cells from the grid. Each cell is probed by evaluating the volume density at a number of sample points, and marked as empty if the overall density is null. As shown in Figure 5 agressive cell elimination does not introduce visible artifacts.

4.3. Rendering

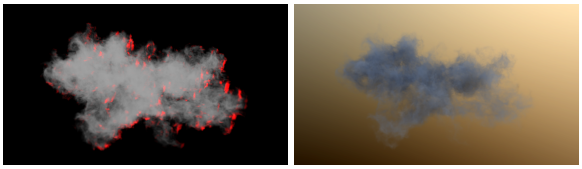
After projecting volume-dependent and -independent factors we render the medium using a ray-marching algorithm to solve Equation 1. For each pixel we cast a ray in the direction $\omega_\mathbf{o}$ and intersect the bounding box of the medium to determine the entry point $\mathbf{p_{in}}$ and the exit point $\mathbf{p_{out}}$. We then sample the ray along the path $|\mathbf{p_{in}p_{out}}|$. For each sample $\mathbf{p_n}$ we compute the scattered radiance. We first determine the cell containing $\mathbf{p_n}$ and fetch the corresponding set of coefficients from the record array. We then compute the *dot product* between these two sets of coefficients to get the scattered radiance. For all rays intersecting the medium we sum the scattered radiance contributions from the samples $\mathbf{p_n}$ attenuated by the medium along the path $|\mathbf{p_{in}p_n}|$. For isotropic static media, F_{Dep} can then be projected only once.



(a) No elimination - Rendering at 29fps



(b) 3^3 samples - Rendering at 48fps



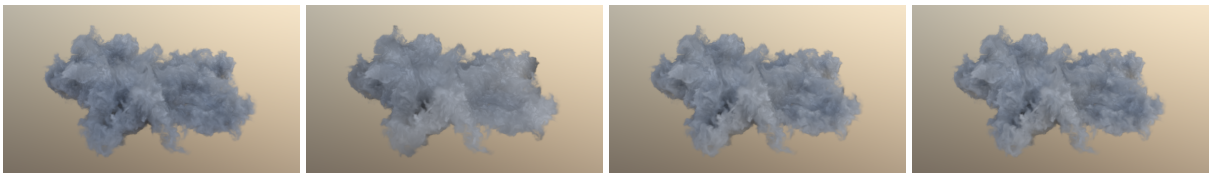
(c) 10^3 samples - Rendering at 10fps

Figure 5: *Cells density with eliminated non-empty cells shown in red: reference (a), 3^3 (b), and 10^3 samples (c).*

4.4. Anisotropic Media

In anisotropic media (Figure 6(b)) the phase function depends on both $\omega_\mathbf{i}$ and $\omega_\mathbf{o}$. A solution for dynamic media directly uses our grid subdivision: Given a viewing direction $\omega_\mathbf{o}$ we evaluate the phase function p for each direction sample $\omega_\mathbf{i}$. We then store the product of p and L into F_{Dep}.

Static media could benefit from another formulation: F_{Ind} can represent the phase function while F_{Dep} stores the product of the reduced intensity and the incoming lighting. The phase function can typically be projected for a number of outgoing directions, as proposed in [KSS02]. This second method allows us to recompute the dynamic factor for static media only when the lighting conditions change.



(a) Thick medium - Ref - 0.07fps (b) 10^3 - 28.00fps (c) 50^3 - 10.70fps (d) 100^3 - 3.81fps

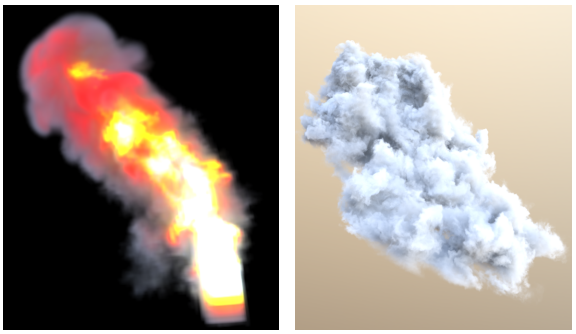
Figure 4: *Comparison between a reference solution and our method with different grid resolutions using a* 512^3 *volume.*

Grid resolution	Ref.	10^3	50^3	100^3
Dynamic lighting	0.07	28.00	10.70	3.81
Static lighting	0.07	31.90	31.90	31.90

Table 1: *Rendering speed (fps) using 9 SH coefficients to render* 1280×720 *images.*

4.5. Emissive media

Our solution easily extends to emissive media by simply accumulating self emission during the evaluation of radiance attenuation. This evaluation comes as a by-product of the reduced intensity computation, hence simulating emissive media at no additional cost.



(a) 128^3, 7.14 fps (b) $813 \times 460 \times 324$, 8.16fps

Figure 6: *Emissive medium (a) and anisotropic production medium (b) rendered using a* 100^3 *grid.*

5. Results

We implemented our algorithm on a Intel Xeon X5680 3.36GHz processor running a Geforce GTX580 GPU. All scenes are rendered with a resolution of 1280×720 (Figure 1). We render a 512^3 resolution volume at 10.70 fps for a full dynamic rendering using a 50^3 subdivision grid. The rendering speed increases to 31.90 fps when the F_{Dep} is evaluated only once (Table 1).

6. Conclusion

We proposed a set of solutions for rendering single scattering due to distant lighting. By splitting the scattered radiance into volume-independent and -dependent factors and using agressive caching, our solutions provide high performance for static and dynamic volumes with arbitrary optical properties. The subdivision of the medium and the projection in spherical harmonics yield interactive to real-time frame rates. The next steps will enhance our technique for efficient multiple scattering evaluation by using spherical harmonics interpolations to simulate light exchanges between neighboring cells. Another goal is the optimization of the medium subdivision for a more efficient cell elimination.

References

[CFP*05] CEREZO E., FRANCISCO S., PEREZ F., SILLION F., PUEYO X.: A survey on participating media rendering techniques. *Visual Computer 21*, 5 (2005), 303–328.

[Cha50] CHANDRASEKHAR S.: *Radiative Transfer*. Clarendon Press, Oxford, 1950.

[DGMF11] DELALANDRE C., GAUTRON P., MARVIE J.-E., FRANÇOIS G.: Transmittance function mapping. In *Proceedings of the Symposium on Interactive 3D Graphics and Games* (2011), pp. 31–38.

[GHB*04] GORSKI K. M., HIVON E., BANDAY A. J., WANDELT B. D., HANSEN F. K., REINECKE M., BARTELMAN M.: Healpix – a framework for high resolution discretization, and fast analysis of data distributed on the sphere, 2004.

[JNSJ11] JAROSZ W., NOWROUZEZAHRAI D., SADEGHI I., JENSEN H. W.: A comprehensive theory of volumetric radiance estimation using photon points and beams. In *Proceedings of SIGGRAPH* (2011), vol. 30, pp. 5:1–5:19.

[KSS02] KAUTZ J., SLOAN P.-P., SNYDER J.: Fast, arbitrary BRDF shading for low-frequency lighting using spherical harmonics. In *Proceedings of the Eurographics Workshop on Rendering* (2002), pp. 291–296.

[NGS09] NAVARRO F., GUTIERREZ D., SERÓN F. J.: Interactive HDR lighting of dynamic participating media. *Visual Computer 25* (2009), 339–347.

[SKS02] SLOAN P.-P., KAUTZ J., SNYDER J.: Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. In *Proceedings of SIGGRAPH* (2002), pp. 527–536.

[Sta94] STAM J.: Stochastic rendering of density fields. In *Proceedings of Graphics Interface* (1994), pp. 51–58.

[Sta95] STAM J.: Multiple scattering as a diffusion process. In *Proceedings of the Eurographics Workshop on Rendering* (1995), pp. 41–50.

[WR08] WYMAN C., RAMSEY S.: Interactive volumetric shadows in participating media with single-scattering. *Proceedings of IEEE Symposium on Interactive Ray Tracing* (2008), 87–92.

[ZRL*08] ZHOU K., REN Z., LIN S., BAO H., GUO B., SHUM H.-Y.: Real-time smoke rendering using compensated ray marching. In *Proceedings of SIGGRAPH* (2008), pp. 1–12.