



HAL
open science

Volume Exploration with Light Diffusion

David Murray, Jérôme Baril, Xavier Granier

► **To cite this version:**

David Murray, Jérôme Baril, Xavier Granier. Volume Exploration with Light Diffusion. journées Française d'Informatique Graphique (j.FIG), Nov 2018, Poitiers, France. hal-03272284

HAL Id: hal-03272284

<https://inria.hal.science/hal-03272284v1>

Submitted on 28 Jun 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Volume Exploration with Light Diffusion

D. Murray^{1,2} and J. Baril¹ and X. Granier²

¹Thermo Fisher Scientific Bordeaux, France

²LP2N (UMR 5298 CNRS, Univ Bordeaux, IOGS) - Inria, Talence, France

Résumé

Exploring and understanding volumetric data is still a challenge in Computer Graphics. We believe that light-transport theory, originally dedicated to the computation of physically accurate visualization of light diffusion in the volume, is a key solution to offer the user a more accurate and more legible visualization. As the first step toward such an approach, we present in this paper an interactive solution of light propagation. We also demonstrate how it may be used to identify regions of interest in a volume.

Mots clé : Volume Rendering, Diffusion

1. Introduction

In volume visualization, conveying key features of a volume is an important, yet tedious, task. It can be done through accurate depiction of material appearance (through realistic renderings), or with expressive rendering techniques, to highlight specific features ([HKR⁺06]). Choosing the appropriate solution to highlight features in volumes requires both data knowledge and field knowledge.

Using physically-based rendering methods usually implies to balance between accuracy and interactivity ([AMHH⁺18]). Thus, a good knowledge of light transport operators is necessary to make this balance. These type of methods is well suited to faithfully represent material properties and appearance of participating media, like human tissues.

On the other hand, expressive techniques can depict a wider range of features, as they do not target faithful appearance. However, most of the expressive rendering techniques for volumes either operate on a subset of element defined by isosurfaces (e.g., [BGKG06]), or by operating per slice of the volume, usually to identify a specific part (e.g., [SA10]).

Concerning this problematic of conveying key features in volumes, our predicate is that we can adapt and tune light transport operators for an expressive purpose. Thus, in this paper, we introduce a new approach to assist the visualization, inspired by fluoroscopy imaging in medicine. This new metaphor is based on light transport to enhance depiction in volumes.

2. Related Work

In order to present an overview of the current techniques used to visualize participating media, we must first present

the optical model on which these techniques are based: the Radiance Transfer Equation [Cha50]. Next, we present a quick overview of rendering techniques based on this light transport model.

2.1. The Radiance Transfer Equation

We first consider a light beam and focus on its energy. When crossing a participating medium, this beam may be subject to energy losses or gains. These gains and losses are often separated in four phenomena:

- Emission: energy gains by if the media creates some light energy.
- Absorption: energy losses by an energy absorption and conversion. Quantified by absorption coefficient K_a .
- Out-scattering: energy losses by deflection toward another direction. Quantified by scattering coefficient K_s .
- In-scattering: energy gains by deflection from particles nearby. Quantified by scattering coefficient K_s .

By focusing on the influence of these phenomena on a small volume element, and for a small time step, one can establish the Radiance Transfer Equation (RTE), as introduced by [Cha50]:

$$\frac{1}{c} \frac{\partial L(p, t, \vec{\omega})}{\partial t} + \vec{\omega} \cdot \vec{\nabla}_p L(p, t, \vec{\omega}) = -K_t(p) \cdot L(p, t, \vec{\omega}) + Q_e(p, t, \vec{\omega}) + \int_{4\pi} K_s(p) \cdot \mathcal{P}(p, \vec{\omega}, \vec{\omega}') \cdot L_i(p, t, \vec{\omega}') d\vec{\omega}' \quad (1)$$

In Equation 1, K_t is the extinction coefficient (in m^{-1}), accounting for both losses by absorption and outward deflection, such as:

$$K_t = K_a + K_s$$

Q_e is the volumetric emission (in $W.m^{-3}$). The Phase Function $\mathcal{P}(p, \vec{\omega}, \vec{\omega}')$ quantifies the portion of energy that is deflected from an incoming direction $\vec{\omega}'$ toward the outgoing direction $\vec{\omega}$, expressed in sr^{-1} .

In Computer Graphics, a common consideration is that the thermodynamic equilibrium is instantly reached. Thus, in most volume visualization applications, the RTE is used with its steady version.

2.2. Volume Rendering Techniques

To render volumes in real-time, one either uses approaches like Direct Volume Rendering (front-to-back ray-marching) or texture slicing (back-to-front compositing). These approaches use mostly a simplified model for the RTE (Emission-Absorption model), with eventually direct lighting techniques. For a more elaborate overview of the problems and solutions of real-time volume rendering, we refer the reader to Hadwiger et al. [HKRs⁺06].

The other category of approaches is offline rendering: the rendering is done more accurately but on the other hand, the computation time is hardly compatible with interactivity. We present here a brief overview of these techniques, for more information and in-depth analysis, we refer the reader to [CPP⁺05] and [JSYR13].

A well-known approach for offline rendering is a stochastic method called ray-tracing, introduced for volume by Kajjiya [Kaj86]. To generate an image, a ray is launched for each pixel of the camera toward the scene in order to find the closest intersection and evaluate the radiance. This process is repeated for as many bounces as desired. An alternative method, called Path Tracing, consists in building the path that the light follows inside the scene. A recent study about this kind of methods is presented by Novak et al. [NGHJ18].

An alternative method for offline rendering is Photon Mapping, introduced by [JC98] for volumes. It consists in shooting a set of photons from the light sources. If a photon reaches a voxel, it is stored at its position. In a second (gathering) pass, rays are cast from the camera. As these rays interact with volume, the photons in a neighboring area are gathered to estimate the radiance. The first version presented by Jensen was biased (the converged results may not be the correct solution) and was improved by [HOJ08] to greatly reduce the bias.

Finally, it is possible to evaluate the light propagation in a medium with an implicit representation. The problem of solving the RTE is reduced to a linear system, solved either with finite elements methods or finite differences methods. A recent example of implicit representation being used for lighting simulation in volumes is presented by Zhang and Ma [ZM13].

3. Our Objective and its Problematic

Focusing on light transport, our hypothesis is that we could use light propagation to assist the visualization, with exploration features.

This idea is strongly inspired by fluoroscopy imaging. In fluoroscopy imaging, a liquid or a gas, fluorescent when observed using X-Ray imaging, is injected in the body and tracked using a fluoroscopy microscope. This medical imaging process is used to visualize the propagation inside organs like the esophagus, the intestine or inside the veins.

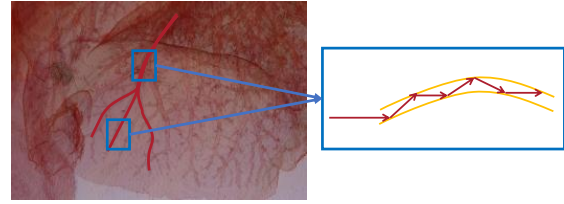


Figure 1: An illustration of a biased light transport algorithm. By using biased parameters to create or amplify discontinuities, we could enforce a confined light propagation, as illustrated on the schematic view.

We want to mimic this type of imaging process by altering the volume properties (absorption and scattering) to modify the output of the RTE resolution. This way, we can constrain, and thus guide, the light propagation in specific structures of the volume. A conceptual representation of this idea is presented in Figure 3.

In order to perform this task, we considered using Ray-tracing and Path-tracing techniques, for the rendering quality they offer with a small impact on memory. However, the resulting image is generally computed according to the point of view, thus, it must be recomputed whenever the camera is moved or when parameters are modified. The same goes for Photon-Mapping, but with a non-negligible additional memory cost as a photon map must be stored for the full volume.

For these reasons, we chose to use an implicit representation. Furthermore, we focus on iterative resolution, as it can be efficiently implemented with per-voxel kernel.

To implement our solution, we have conducted a study to evaluate several numerical schemes for both the steady and unsteady formulation of the RTE. In particular, we have experimented using finite element with main axis projection and finite differences with a first and second order angular moment representation. In all cases, the numerical scheme is either instable, or leads to artifacts (mostly grid patterns), a problem also stated recently by Koerner et al. [KPJ18].

For a more natural iterative resolution, the unsteady RTE is more suited as, in theory, each step give radiance results that are consistent with this quantity, even before convergence. Furthermore, by making some reasonable approximations, we can reduce the RTE to a diffusion equation, more stable and less subject to artifacts. In the next section, we present how we can obtain a diffusion equation from the RTE, and how we use it to achieve our goal of performing selective diffusion.

4. Our Solution in Details

4.1. Diffusion Equation

Introduced in Computer Graphics by Stam [Sta95] (and later used by [JMLH01]), the Diffusion Approximation is widely used to address scattering problems. As stated by Stam [Sta95], in highly dispersive media (low absorption and high scattering) there are many occurrences of scattering events (multiple scattering). Thus, even if the phase function

is highly anisotropic, if the media is wide and dense enough, the radiance tends to become isotropic due to these repeated deflection. We can then approximate the radiance as a combination of an isotropic contribution, $\phi(p)$, and a direction-dependent correction term, $\langle \vec{\omega}, \vec{E}(p) \rangle$, such as:

$$L(p, \vec{\omega}) = \frac{1}{4\pi} \phi(p) + \frac{3}{4\pi} \langle \vec{\omega}, \vec{E}(p) \rangle \quad (2)$$

The isotropic term is linked to the spherical integral of the outgoing radiance, called the fluence rate:

$$\phi(p) = \int_{\Omega^2} L(p, \vec{\omega}) d\omega \quad (3)$$

The direction-dependent term corresponds the total flux crossing a surface, called the irradiance vector:

$$\vec{E}(p) = \int_{\Omega^2} L(p, \vec{\omega}) \vec{\omega} d\omega \quad (4)$$

The factors $\frac{1}{4\pi}$ and $\frac{3}{4\pi}$ are normalization constants.

Now, to obtain a diffusion equation from the RTE, we first focus on its moment of order 0 and 1, which are respectively:

$$\begin{aligned} \frac{1}{c} \frac{\partial \phi(p, t)}{\partial t} &= -\text{div}(\vec{E}(p, t)) - K_a(p) \phi(p, t) + \phi_e(p, t) \\ \frac{1}{c} \frac{\partial \vec{E}(p, t)}{\partial t} &= -\frac{1}{3} \vec{\nabla}_p \phi(p, t) - K_{t'}(p) \vec{E}(p, t) + \vec{E}_e(p, t) \end{aligned}$$

For more details about these terms, we refer the reader to

We also have to make another hypothesis, in addition to the Diffusion Approximation. As it is done in some diffusion problem (e.g., [HST⁺94]), we consider that the irradiance vector (the flux) is approximately steady, meaning that:

$$\frac{\partial \vec{E}}{\partial t} \approx 0$$

Note that this hypothesis can also be inferred directly from the Diffusion Approximation. As the latter is valid only for long time scales, it implies that the flux (the irradiance vector \vec{E}) can not present significant variations, to ensure that we are in a diffusive state. This is also inherent from the formulation of Equation 2, as, to ensure that the radiance is positive, we must have $\phi \gg \|\vec{E}\|$ ([Ish97]). By using the approximation of a steady flux, we obtain a diffusion equation such as:

$$\begin{aligned} \frac{1}{c} \frac{\partial \phi(p, t)}{\partial t} &= D(p) \Delta_p \phi(p, t) - \langle \vec{\nabla}_p D(p), \vec{\nabla}_p \phi(p, t) \rangle \\ &\quad - K_a(p) \phi(p, t) + \phi_e(p, t) \end{aligned} \quad (5)$$

With $D(p) = \frac{1}{3K_{t'}(p)}$, the diffusion coefficient. To use this model, we use a finite difference scheme, with forward time and central space pattern.

4.2. Controlling the Diffusion

Equation 5 has a general form that suit our needs. However, we propose to use variables more adapted to implementing the diffusion, and more user-friendly. Based on the formulation we have chosen, the controllable parameters in our algorithm are: D , K_a and dt . D is controlled using either predefined functions or using a custom transfer function. K_a can be controlled indirectly using either an automatic value based on D or by using the scattering albedo α ([vdH81]),

which quantifies the ratio between scattering and absorption in a medium, such as:

$$\alpha = \frac{K_s}{K_t}$$

As for dt , it should be set accordingly to the discretization spatial step to ensure convergence of the method. More information about the tools we use to control this diffusion will be available a document that has yet to be published (expected by December 2018).

4.3. Implementation Details

Our implementation is done using Open Inventor (version 9.8), an SDK from Thermo Fisher Scientific, and OpenGL (up to version 4.3) to handle features not directly supported by Open Inventor. We first present how we implement the diffusion algorithm and then, how we store the results.

Our implementation relies on two 3D buffers to store the result of the diffusion, using a Ping Pong scheme ([GR92]). This is necessary as we use an iterative resolution, meaning that during the computation we access data that comes from a previous step. As for the computation step, it is done with compute shaders as they allow using a buffer format that can be directly interfaced with the other programmable stages of the graphic pipeline.

5. Results and Limitations

In section we present different results of our method in term of exploration capabilities and we also discuss about the limitations of our method.

Results on Figures 2(a) and 2(b) are achieved by using a mapping for D that is based on the scalar gradient of the original data. We can see that such a mapping is efficient to perform constrained diffusion, as the light remain in the structure it was emitted in.

As for the limitations, an significant one is the tuning of the constraints we apply. If it is not sufficiently constrained, the diffusion overflows (Figure 2(c)), if it is too much constrained, the diffusion is limited (Figure 2(d)). A trade-off can hardly be found, so we should design additional tools to control this aspect.

Also, due the prototype state of our solution, the memory and computation costs are currently limiting. Indeed, it has yet to be optimized, as we use, in addition to the original data, 256 bits per voxel to store the result of the diffusion. As for the computation time, our method currently scales linearly with the number of voxel. It requires an additional 2.5ms per iteration for a data of $128 \times 128 \times 128$ voxel. Thus for bigger volume, it hinders the interactivity.

6. Conclusion

Our goal was to provide a solution to visualize structures and regions of interest in volumes without using global segmentation. To address this problem, we have proposed a model to solve the RTE iteratively, with its stability condition. This method fulfills the initial goal as it effectively provides a fluid-like diffusion while being based on light transport. We

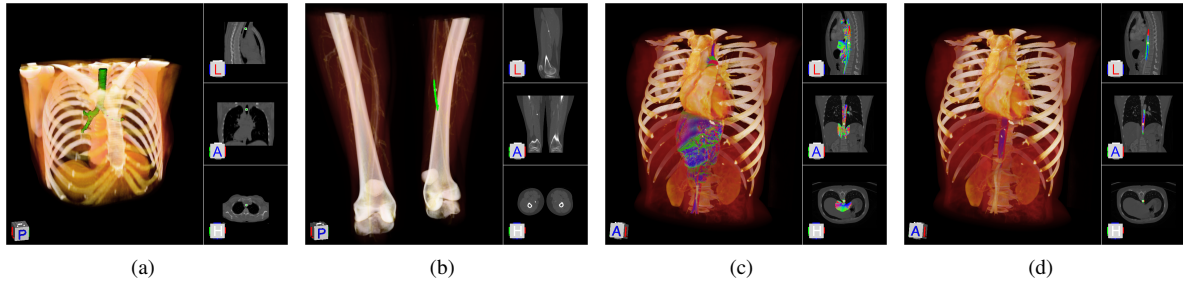


Figure 2: Examples of rendering using our methods. (a) A source placed at the top of the trachea, diffusing in it. (b) A source placed inside a vein. (c,d) Examples of ill-controlled parameters by visualizing the normalized irradiance vector (\vec{E}). The vector is also displayed on the slices for understanding. (c) The diffusion is done without enough absorption, leading to overflowing. (d) The diffusion is done with too much absorption, leading to an over-constrained diffusion, as the vein is not fully identified.

have shown that the solution can be used to identify regions of interest using diffusion.

However, the method remains limited, in particular, the visualization aspect has not been properly studied in terms of rendering styles and transfer functions. As we focused on the diffusion model, the aspect of visualization is treated here and should be addressed in future work.

Also, the method could probably be optimized to reduce the memory footprint as well as the computation cost. Concerning memory, we believe this amount could be reduced by studying the impact of using lower resolutions and by optimizing the bit occupancy. For computation time, one solution is to use a more adapted pattern for the compute groups, another aspect not treated, in order to fully exploit compute shaders capabilities.

Finally, user interactions are essential when using such a method, and this aspect has not been addressed in the paper. However, as we believe that they are keys to a potential success of the method, it must be tackled in future work. In a document that has yet to be published (expected by December 2018), we address some of these interaction problems.

Références

- [AMHH⁺18] Tomas Akenine-Möller, Eric Haines, Naty Hoffman, Angelo Pesce, Michał Iwanicki, and Sébastien Hillaire. *Real-Time Rendering 4th Edition*. A K Peters/CRC Press, 2018.
- [BGKG06] Stefan Bruckner, Sören Grimm, Armin Kanitsar, and Meister Eduard Gröller. Illustrative context-preserving exploration of volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1559–1569, 2006.
- [Cha50] S. Chandrasekhar. *Radiative Transfer*. Dover books on physics and engineering. Clarendon Press, 1950.
- [CPP⁺05] Eva Cerezo, Frederic Pérez, Xavier Pueyo, Francisco J. Seron, and François X. Sillion. A survey on participating media rendering techniques. *Vis. Comput.*, 21(5):303–328, June 2005.
- [GR92] Jim Gray and Andreas Reuter. *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., 1st edition, 1992.
- [HKRs⁺06] Markus Hadwiger, Joe M. Kniss, Christof Rezk-salama, Daniel Weiskopf, and Klaus Engel. *Real-time Volume Graphics*. A. K. Peters, Ltd., 2006.
- [HOJ08] Toshiya Hachisuka, Shinji Ogaki, and Henrik Wann Jensen. Progressive photon mapping. In *ACM Transactions on Graphics (TOG)*, volume 27, page 130. ACM, 2008.
- [HST⁺94] Richard C. Haskell, Lars O. Svaasand, Tsong-Tseh Tsay, Ti-Chen Feng, Matthew S. McAdams, and Bruce J. Tromberg. Boundary conditions for the diffusion equation in radiative transfer. *J. Opt. Soc. Am. A*, 11(10):2727–2741, Oct 1994.
- [Ish97] A. Ishimaru. *Wave Propagation and Scattering in Random Media*. An IEEE OUP classic reissue. IEEE Press, 1997.
- [JC98] Henrik Wann Jensen and Per H. Christensen. Efficient simulation of light transport in scenes with participating media using photon maps. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '98*, pages 311–320. ACM, 1998.
- [JMLH01] Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A practical model for subsurface light transport. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '01*, pages 511–518. ACM, 2001.
- [JSYR13] Daniel Jönsson, Erik Sundén, Anders Ynnerman, and Timo Ropinski. A survey of volumetric illumination techniques for interactive volume rendering. *Computer Graphics Forum*, 33(1):27–51, 2013.
- [Kaj86] James T Kajiya. The rendering equation. In *ACM SIGGRAPH Computer Graphics*, volume 20, pages 143–150. ACM, 1986.
- [KPJ18] David Koerner, Jamie Portsmouth, and Wenzel Jakob. PN-Method for Multiple Scattering in Participating Media. In Wenzel Jakob and Toshiya Hachisuka, editors, *Eurographics Symposium on Rendering - Experimental Ideas & Implementations*. The Eurographics Association, 2018.
- [NGHJ18] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte carlo methods for volumetric light transport simulation. *Computer Graphics Forum (Proceedings of Eurographics - State of the Art Reports)*, 37(2), may 2018.
- [SA10] Neeraj Sharma and Lalit M Aggarwal. Automated medical image segmentation techniques. 35, 2010.
- [Sta95] Jos Stam. Multiple scattering as a diffusion process. In Patrick M. Hanrahan and Werner Purgathofer, editors, *Rendering Techniques '95*, pages 41–50. Springer Vienna, 1995.
- [vdH81] H.C. van de Hulst. Light scattering by small particles. 10, 1981.
- [ZM13] Yubo Zhang and Kwan-Liu Ma. Fast global illumination for interactive volume visualization. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, I3D '13*, pages 55–62. ACM, 2013.