



HAL
open science

A Local Frequency Analysis of Light Scattering and Absorption

Laurent Belcour, Kavita Bala, Cyril Soler

► **To cite this version:**

Laurent Belcour, Kavita Bala, Cyril Soler. A Local Frequency Analysis of Light Scattering and Absorption. ACM Transactions on Graphics, 2014, 33 (5), pp.Article No. 163. 10.1145/2629490 . hal-00957242

HAL Id: hal-00957242

<https://inria.hal.science/hal-00957242>

Submitted on 18 Aug 2015

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.



Distributed under a Creative Commons Attribution 4.0 International License

A Local Frequency Analysis of Light Scattering and Absorption

LAURENT BELCOUR

Manao[†], Inria Bordeaux Sud-Ouest

and

KAVITA BALA

Cornell University

and

CYRIL SOLER

Maverick, Inria Rhône-Alpes

Rendering participating media requires significant computation, but the effect of volumetric scattering is often eventually smooth. This paper proposes an innovative analysis of absorption and scattering of local light fields in the Fourier domain, and derives the corresponding set of operators on the covariance matrix of the power spectrum of the light field. This analysis brings an efficient prediction tool for the behavior of light along a light path in participating media. We leverage this analysis to derive proper frequency prediction metrics in 3D by combining per-light path information in the volume.

We demonstrate the use of these metrics to significantly improve the convergence of a variety of existing methods for the simulation of multiple scattering in participating media. Firstly, we propose an efficient computation of second derivatives of the fluence, to be used in methods like irradiance caching. Secondly, we derive proper filters and adaptive sample densities for image-space adaptive sampling and reconstruction. Thirdly, we propose an adaptive sampling for the integration of scattered illumination to the camera. Finally, we improve the convergence of progressive photon beams by predicting where the radius of light gathering can stop decreasing. Light paths in participating media can be very complex. Our key contribution is to show that analyzing local light fields in the Fourier domain reveals the consistency of illumination in such media, and provides a set of simple and useful rules to be used to accelerate existing global illumination methods.

Categories and Subject Descriptors: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—*Raytracing*

Additional Key Words and Phrases: global illumination, participating media, adaptive sampling, frequency analysis

[†]Inria - LP2N (CNRS, Univ. Bordeaux, IOGS) - LaBRI (CNRS, Univ. Bordeaux)

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© YYYY ACM 0730-0301/YYYY/14-ARTXXX \$10.00

DOI 10.1145/XXXXXXXX.YYYYYYY

<http://doi.acm.org/10.1145/XXXXXXXX.YYYYYYY>

1. INTRODUCTION

Rendering participating media is challenging because of the high cost of simulating scattering events. But participating media mostly blur out details, and decrease the contrast of images: some image regions appear almost locally constant, and light beams are practically constant in the direction of light propagation. In this paper we introduce a new frequency analysis of local light fields in participating media. We show the effect that volumetric scattering has on lowering frequency content and contrast. We derive the associated theoretical framework and provide tools to optimize participating media rendering algorithms using the frequency content of light transport.

Scattering is a long standing problem in computer graphics where a range of techniques, with varying trade-offs between performance and accuracy, have been proposed to simulate the interaction of light with participating media. Unbiased methods such as path tracing [Lafortune and Willems 1993] and Metropolis light transport [Veach and Guibas 1997] provide accuracy, but often at a prohibitive cost. Photon mapping based approaches handle participating media [Jensen and Christensen 1998; Knaus and Zwicker 2011] with different trade-offs. Methods such as Photon Beams [Jarosz et al. 2011] efficiently simulate low order scattering, relying on the accumulation of illumination primitives (e.g., points or beams) to compute images. While some approaches exploit the lower frequency nature of lighting in participating media, to our knowledge, there is no existing literature on *a priori* frequency analysis of local light fields in volume transport.

For non-volumetric surface-based rendering, Durand et al. [2005] introduced a frequency analysis of light transport. We extend this framework to characterize the behavior, in the Fourier domain, of light traveling and scattering inside participating media. Methods exist that use the Fourier transform as a global transform operator in 3D to decouple the frequencies in the scattering equation [Ishimaru 1997]. Instead, our extension to the frequency analysis framework applies to 4D local light fields, along light paths in the medium.

We build on covariance analysis [Belcour et al. 2013], an efficient and practical representation of the covariance matrix of the frequency spectrum of the local light field. It was used to accelerate the rendering of motion and defocus blur. The covariance matrix representation conveys the required information on the Fourier transform of the light field, at a very small cost, making it tractable for path-tracing.

In this paper, we extend the covariance representation to global illumination in participating media, including multiple scattering. We show that our new formulae for participating media fit nicely in

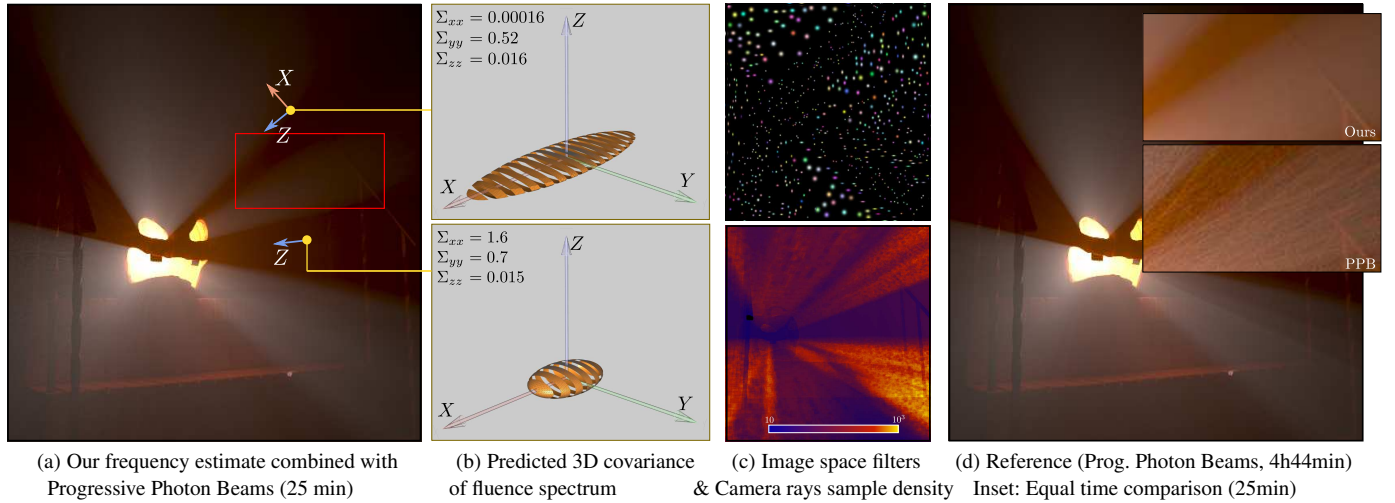


Fig. 1. We propose a frequency analysis of light transport in participating media, a broad theoretical tool that allows improvements in a wide variety of algorithms. From the predicted 3D covariance of the fluence in the Fourier domain (b), we derive three different sampling metrics in the 3D volume. We present multiple example uses of these metrics: to improve image-space adaptive sampling density and reconstruction, we provide sampling density and reconstruction filters (c-top); to improve light integration along camera rays, we evaluate a required number of samples along those rays (c-bottom); to improve progressive photon beams, we derive the optimal reconstruction radius based on the frequency content (a and d). In order to ease comparisons, we scaled the covariance graphs (b), and increased the luminosity of insets (d). This scene is composed of a pumpkin modeled by user Mundomupa of blendswap.com provided under creative common license CC-BY, and of a house front modeled by Jeremy Birn.

the existing framework when the medium is not optically thick (as in subsurface scattering). We use the covariance information of the local light field spectrum in 4D along light paths to predict the 3D covariance of the windowed spectrum of fluence in volumes with participating media. We propose four application scenarios where this quantity proves useful (see Figure 1). The contributions of this paper are:

- A local analysis of scattering and absorption in the Fourier domain along a light path, in heterogeneous participating media. The model is compatible with multiple scattering.
- A compact representation of attenuation and scattering in the Fourier domain, using covariance matrices.
- The combination of covariance from many light paths in the medium into usable sampling metrics in 3D.
- Four different computation scenarios that benefit from our analysis: computation of second derivatives of the fluence; image space adaptive sampling and reconstruction; adaptive sampling of scattered illumination along rays from the camera; and progressive photon beams.

Note that the term “spectral analysis” usually has multiple meanings; it is sometimes used to refer to the eigenanalysis of linear operators. In this paper the term spectrum refers to the frequency spectrum of the Fourier transform of light fields.

2. PREVIOUS WORK

We categorize related work into research on the frequency analysis of light transport, and on the volume rendering of participating media.

2.1 Fourier domain methods for scattering.

In these methods, the Fourier transform is used as a tool for solving the scattering equation at once in the entire domain [Duderstadt and Martin 1979; Ishimaru 1997]. Some methods use a different basis for certain dimensions, such as the Chebychev basis [Kim and Moscoso 2003], or spherical harmonics [Dave 1970]. These methods in general depend on a combination of very specific constraints: infinite or spherical domains [Dave and Gazdag 1970], periodic boundary conditions [Ritchie et al. 1997], isotropic scattering functions [Rybicki 1971], and mostly homogeneous scattering functions. These conditions make such methods not very suitable to computer generated images where the constraints of uniformity and periodicity can hardly be satisfied.

Our approach is fundamentally different: we use the Fourier transform as a local tool in the 4D ray space to predict bandwidth—as opposed to globally solving the equations—which allows us to handle non homogeneous participating media.

2.2 Volume rendering.

The field of rendering participating media has a long history. Volume rendering based on ray tracing techniques was first proposed for forward path tracing integration [Kajiya and Von Herzen 1984]. It has been expanded afterwards to other integration schemes: Laforge and Willems [1996] extended bidirectional path tracing; Pauly et al. [2000] adapted Metropolis for participating media. Photon mapping [Jensen and Christensen 1998] has been shown to be efficient in generating high frequency light patterns such as caustics. Cerezo et al. [2005] surveys the state-of-the-art, though it is a bit dated.

Recently, various extensions to photon mapping and progressive photon mapping use photon beams, rather than point sampling along rays, to greatly improve the performance of volume rendering [Jarosz et al. 2008; Jarosz et al. 2011; Jarosz et al. 2011; Knaus

and Zwicker 2011]. These methods however, remain unaware of image complexity, and rely on an accumulation of illumination primitives (e.g., points or beams) to compute an image that will eventually be very smooth.

Several virtual point light (VPL)-based algorithms for volumetric rendering trade-off quality and performance. Light cuts and variants [Walter et al. 2005; Walter et al. 2006; Walter et al. 2012] achieve scalable rendering of complex lighting with many VPLs for motion blur, depth of field, and volumetric media (including for oriented media [Jakob et al. 2010]). These scalable approaches couple error bounded approximations with simple perceptual metrics. For interactive VPL rendering of participating media, Engelhardt et al. [2010] introduce a GPU-friendly bias compensation algorithm. Novak et al. [2012] spread the energy of virtual lights along both light and camera rays, significantly diminishing noise caused by singularities.

Multiple approaches aim at efficiently computing low-order scattering in refractive media. Walter et al. [2009] compute single scattering in refractive homogeneous media with triangle boundaries. Ihrke et al. [2007] solve the eikonal equation with wavefront tracing for inhomogeneous media with varying refractive indices and Sun et al. [2010] develop a line gathering algorithm to integrate complex multiple reflection/refraction and single scattering volumetric effects for homogeneous media.

2.3 Efficient sampling and reconstruction methods

Some works perform adaptive sampling or local filtering using heuristics based on the frequency of light transport, without explicitly computing frequency information. Adaptive sampling for single scattering [Engelhardt and Dachsbacher 2010] permits re-sampling when detecting an occlusion. This approach finds epipolar lines, sparsely samples and interpolates along these lines, but finds occlusion boundaries to preserve high frequency details. An epipolar coordinate system [Baran et al. 2010; Chen et al. 2011] allows to interactively compute single scattering in volumetric media by exploiting the regularity of the visibility function.

The structure of the light field [Levoy and Hanrahan 1986; Gortler et al. 1986] can be exploited to perform adaptive sampling or reconstruction. For surface radiance computation, Lehtinen et al. [2011] exploits anisotropy in the temporal light field to efficiently reuse samples between pixels, and perform visibility-aware anisotropic reconstruction to indirect illumination, ambient occlusion and glossy reflections. Ramamoorthi et al. [2012] derived a theory of Monte Carlo visibility sampling to decide on the best sampling strategies depending on a particular geometric configuration. Mehta et al. [2012] derives the sampling rates and filter sizes to reconstruct soft shadows from a theoretical analysis to consider axis-aligned filtering.

Irradiance caching methods [Jarosz et al. 2008] inherently perform filtering in the space of the irradiance by looking at the irradiance gradient. For example, Ribardiere et al. [2011] perform adaptive irradiance caching for volumetric rendering. They predict variations of the irradiance and map an ellipsoid to define the non-variation zone with respect to a local frame.

2.4 Frequency analysis of light transport.

In their frequency analysis of light transport, Durand et al. [2005] studied the frequency response of the radiance function to various radiative transport phenomena (such as transport, occlusion and reflection). Other works on this subject [Egan et al. 2009; Soler et al. 2009; Belcour and Soler 2011; Bagher et al. 2012] have enriched the number of effects to be studied (motion, lens) and showed that

filtering and adaptive sampling methods can benefit from frequency analysis. Yet, some radiative phenomena have not been studied in this framework, including refraction and scattering. We aim to fill a part of this gap by bringing comprehension of the frequency equivalent of volume scattering and attenuation operators, and showing the usefulness of such analysis with a few practical applications.

A frequency analysis has been carried out for shadows specifically by Egan et al. in 4D to build sheared reconstruction filters for complex visibility situations [Egan et al. 2011], or in the case of occlusion by distant illumination [Egan et al. 2011].

3. BACKGROUND: COVARIANCE OF LOCAL SPECTRUM

Our ultimate goal is to provide a general, efficient tool for predicting the variations of the illumination, at different stages of the calculation of global illumination, so as to make sampling and reconstruction methods the most efficient possible. In prior work [Belcour et al. 2013], it was demonstrated that the covariance of the spectrum of the local light field along rays does this job. In this paper, we perform the mathematical analysis to extend this approach to multiple scattering in participating media. This section recalls the basics about local light fields, Fourier analysis of light transport and the covariance representation of the spectrum as background.

3.1 Local light fields

We call the *local light field* the 4D field of radiance in the 4D neighborhood of a ray. Our space of study is the 4D domain of tangential positions around a central ray [Igehy 1999; Wand and Straßer 2003]. It is parameterized by two spatial and two angular coordinates, defined with respect to the plane orthogonal to the ray at a 3D position \mathbf{x} (See Figure 2).

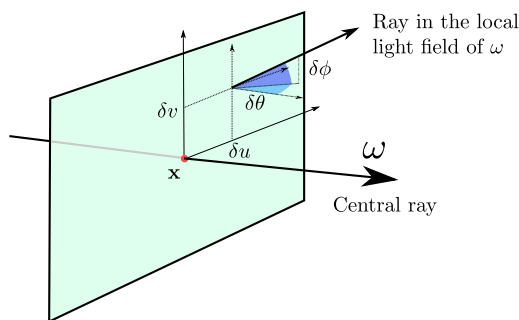


Fig. 2. Parameterization of a local radiance light field around a ray of direction ω . We use δu , δv as the transverse spatial coordinates of the ray and $\delta \theta$, $\delta \phi$ as its angular coordinates.

3.2 Fourier analysis

Durand et al. analyzed the various effects a local light field undergoes along a light path [Durand et al. 2005]. They showed that the effect of light transport operators such as reflection, free space transport, and occlusion, all correspond to simple operators on the Fourier spectrum of the light field. These operators and their equivalent operator in the Fourier domain are listed in Table I.

Space travel	Occlusion	BRDF	Rotation by angle φ	Scale $f(\lambda_1 x_1, \dots, \lambda_4 x_4)$
$\Sigma' = T_d^T \Sigma T_d$	$\Sigma' = \Sigma + O$	$\Sigma' = (\Sigma^{-1} + B)^{-1}$	$\Sigma' = R_\varphi^T \Sigma R_\varphi$	$\Sigma' = \Lambda \Sigma \Lambda$
$T_d = \begin{bmatrix} 1 & 0 & -d & 0 \\ 0 & 1 & 0 & -d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$O = \begin{bmatrix} O_{xx} & O_{yx} & 0 & 0 \\ O_{xy} & O_{yy} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & B_{\theta\phi}^{-1} & \\ 0 & 0 & & \end{bmatrix}$	$R_\varphi = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) & 0 & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 & 0 \\ 0 & 0 & \cos(\varphi) & -\sin(\varphi) \\ 0 & 0 & \sin(\varphi) & \cos(\varphi) \end{bmatrix}$	$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 \\ 0 & 0 & 0 & \lambda_4 \end{bmatrix}$

Fig. 3. Operations on the light field and their equivalence on the *covariance matrix of the light field's spectra* at various stages of light transport (see Belcour et al. [2013] for detailed derivations). The middle row shows each operator applied to the covariance Σ while the bottom row details the constant matrices involved. Occlusion adds spatial covariance (O is the 2D spatial covariance of the occluder), BRDF removes angular covariance ($B_{\theta\phi}$ is the angular covariance of the BRDF locally to the reflected direction), while rotation and free-space travel only cause a re-parameterization.

Table I. Operators on the light field function along a light path, and their Fourier equivalent.

Effect	Light space operator	Fourier operator
Space travel	Angles→space shear	Space→angles shear
Occlusion	Product with visibility	Convolution by occluder's spectrum
Projection	Scale	Inverse scale
BRDF integral	Convolution by BRDF	Product with BRDF's spectrum

3.3 Covariance representation

It is not practical to perform a complete calculation of the full spectrum, especially on a per light path basis. Fortunately, we do not need the full spectrum of local light fields to perform useful predictions about how the local light field behaves. The relevant information needed is how far and in which directions the spectrum spreads in the Fourier domain. It was demonstrated that the covariance matrix of the power spectrum of the local light field is sufficient to maintain this information [Belcour et al. 2013].

In the most general case, assuming non-static scenes, local light fields are defined over space, angle, and time, making the light field and its power spectrum a 5D function. The effect of motion is derived independently of the other 4D operators using a change of coordinates in time [Belcour et al. 2013]. In the present document, we chose to focus on static scenes only, and work with 4D covariance matrices.

For any zero-centered, non-negative real function f defined over the 4D domain, the covariance matrix of f is a 4×4 matrix, denoted as Σ , and defined by:

$$\forall (i, j) \in \{1, \dots, 4\}^2 \quad \Sigma_{i,j} = \frac{1}{\int_{\Omega} f} \int_{\mathbf{x} \in \Omega} (\mathbf{x} \cdot \mathbf{e}_i)(\mathbf{x} \cdot \mathbf{e}_j) f(\mathbf{x}) d\mathbf{x} \quad (1)$$

In this equation, \mathbf{e}_i is the i^{th} vector of the canonical basis of the 4D space Ω , while $(\mathbf{x} \cdot \mathbf{y})$ is the dot product of vectors \mathbf{x} and \mathbf{y} .

The covariance matrix has very interesting properties in terms of what we actually need:

- its eigenvectors indicate in which direction function f spreads the most and where it spreads the least;
- its eigenvalues are the variance of the function in all 4 principal directions;
- it is additive, which allows us to accumulate the covariance matrices of rays. More specifically, the Monte Carlo estimate of a composed covariance matrix of many rays is the weighted average of the individual covariance matrices with radiance as weights.

Therefore, the covariance of the *power spectrum* (amplitude of the spectrum) of the local light field can provide us information about sampling and integrating the light field function [Belcour et al.

2013]. In the remaining text, we use the term *spectral covariance* to mean the covariance of the power spectrum of a given quantity.

3.4 Relationship with a Gaussian approximation

When the covariance Σ is non-degenerate, it also coincides with the covariance matrix of the Gaussian g defined by

$$g(\mathbf{x}) = e^{-\mathbf{x}^T \Sigma^{-1} \mathbf{x}}$$

Therefore, representing a function by its covariance matrix Σ is just as good as approximating that function by the Gaussian above. Such an approximation appears very relevant for power spectra, which—just like Gaussians—are zero-centered radially symmetric functions.

However, in order to handle degenerate cases, using a covariance matrix is more practical than handling Gaussian functions. Besides, all transformations we perform over the local light field directly turn into algebraic operations on the covariance matrix, with no further approximation except for multiplication [Belcour et al. 2013].

3.5 Covariance algebra

All operators listed in Table I directly act on the covariance matrix as algebraic operations, most of which are left-and-right products with constant matrices (see Figure 3). Obviously missing in this list are the operators to model the effect of volumetric attenuation and scattering over the local light field along a light path. We derive them in the next section. We also need an efficient way of combining the 4D light path covariance information in the 3D domain, to predict how smooth the diffused illumination will eventually be, and where and how to sample it. This is done in Section 5.

3.6 Relationship with second derivatives

We show in Appendix A, that for any function f with good regularity in the neighborhood of a given point \mathbf{x}_0 , the covariance matrix of the *windowed spectrum* of f in the neighborhood of \mathbf{x}_0 corresponds to the Hessian matrix of f in the *primal domain* at \mathbf{x}_0 , in the coordinate system of the principal curvatures:

$$\forall i \quad \Sigma_{ii}(\hat{f}) = \frac{1}{4\pi^2 |f(\mathbf{x}_0)|} \left| \frac{\partial^2 f}{\partial x_i^2}(\mathbf{x}_0) \right| \quad (2)$$

This last property will be very useful later on in our analysis, as a practical method to compute covariance matrices of the windowed spectrum of some signals (e.g. the phase functions) around particular points.

4. FOURIER ANALYSIS FOR PARTICIPATING MEDIA

In this section, we extend the frequency analysis of light transport to participating media. We follow a light path, bouncing possibly multiple times, into the medium. Our model is therefore compatible with multiple scattering. We derive the equations to model the effect of two operators over the frequency spectrum of the local light field around this light path: absorption and scattering. We first perform a first order expansion of the phenomena. We then study the Fourier equivalent of the two operators, and show how to express them using algebraic operations on the spectral covariance matrices of the light field. Table II summarizes our notations.

Table II. Definitions and notations used in the paper.

$\delta u, \delta v, \delta \theta, \delta \phi$	Spatial and angular local coordinates
$l(\delta u, \delta v, \delta \theta, \delta \phi)$	local light field function
$\Omega_u, \Omega_v, \Omega_\theta, \Omega_\phi$	Spatial and angular variables in Fourier space
$\hat{l}(\Omega_u, \Omega_v, \Omega_\theta, \Omega_\phi)$	Fourier spectrum of the local light field
Σ	4D covariance of the local light field spectrum
Γ	3D covariance of windowed spectrum of fluence
δt	Coordinate along the central direction of travel
$\kappa(x, y, z)$	Volumetric absorption at 3D position (x, y, z)
$\kappa_{uv}(u, v)$	Volumetric absorption in plane orthogonal to a ray
$\omega, \omega_i, \omega_s$	directions of light (general, incident, scattered)
$\rho(\omega_i, \omega_s)$	phase function for directions ω_i, ω_s
$\bar{\rho}(\delta \theta, \delta \phi)$	phase function around ω_i, ω_s , i.e. $\bar{\rho}(0, 0) = \rho(\omega_i, \omega_s)$
ρ_g	Henyey-Greenstein function with parameter g
α	Finite angle for scattered direction
$\otimes_{\Omega_u \Omega_v}$	Convolution operator in the Fourier Ω_u, Ω_v plane

Although the behavior of individual light paths in participating media is potentially complicated, we will show that in the Fourier domain, absorption acts like visibility, and scattering acts like reflectance. Not only does this fit elegantly into the existing framework, but it also results in a very simple frequency prediction tool for efficiently rendering participating media.

4.1 Volumetric absorption

We first consider the effect of volumetric absorption. When the density of particles is not constant in space, energy is not uniformly absorbed as light travels through the medium. This creates an increase in spatial frequencies in the signal (similar to shadows), which further propagates to the angular domain because of the travel of light. We study the effect of volumetric absorption by a density function $\kappa(x, y, z)$ acting as an extinction coefficient along a ray, for a small travel distance δt along ω (see Figure 4).

The attenuated light obeys the following differential equation [Cerezo et al. 2005]:

$$\frac{\partial(l(\mathbf{x} + t\omega, \omega))}{\partial t}(0) = -\kappa(\mathbf{x})l(\mathbf{x}, \omega) \quad (3)$$

We perform a first order approximation of the absorption, considering κ to be constant for a small distance δt along ω . This allows us to integrate this equation as:

$$l(\mathbf{x} + \delta t\omega, \omega) = l(\mathbf{x}, \omega)(1 - \delta t\kappa(\mathbf{x})) \quad (4)$$

Let κ_{uv} be the restriction of κ to the plane, orthogonal to ω (Which means $\kappa_{uv}(\delta u, \delta v) = \kappa(\mathbf{x} + \delta u\mathbf{u} + \delta v\mathbf{v})$). We adopt the notation $p(\delta u, \delta v) = 1 - \delta t\kappa_{uv}(\delta u, \delta v)$. In the Fourier domain, Equation 4 turns into a convolution:

$$\hat{v} = \hat{l} \otimes_{\Omega_u \Omega_v} \hat{p} \quad (5)$$

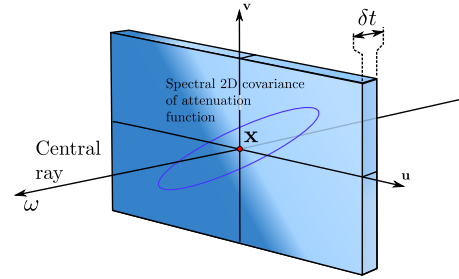


Fig. 4. Notations for the attenuation operator. We analyze the spectral covariance of the attenuation for a small travel distance δt along the central ray. Using small distances allows to assume that the attenuation is constant along ω .

In this equation, $\otimes_{\Omega_u \Omega_v}$ denotes a convolution over the spatial component only. The effect of attenuation is therefore identical to occlusion, except that the mask $p = 1 - \delta t\kappa_{uv}$ is a function taking arbitrary values in $[0, 1]$ instead of a binary function. Let A be the covariance matrix of \hat{p} . Applying the covariance formula for occlusion (Figure 3), we write the covariance matrix of the convolution as the sum of the two covariance matrices:

$$\Sigma' = \Sigma + A \quad (6)$$

This equation shows that absorption transfers covariance into the spectrum of the local light field. Another way of seeing this is that the oscillations of absorption transfer into the light-field.

Computing matrix A in practice, is actually simple using Equation 2: we compute the 2D Hessian matrix $H(\mathbf{x})$ of κ in the (\mathbf{u}, \mathbf{v}) basis using finite differences, and diagonalize it using a 2D-rotation R . We apply the absolute value, and convert it back to the (\mathbf{u}, \mathbf{v}) coordinate system, using covariance rotation with the inverse rotation R^T (using Figure 3):

$$A = \frac{\delta t}{4\pi} \begin{bmatrix} \left[R^T | RH(\mathbf{x})R^T | R \right] & \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} \\ \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} & \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} \end{bmatrix} \quad (7)$$

It follows that if a ray crosses a region with transverse sharp transitions of the attenuation function (e.g., a transverse transition from opaque to non-opaque medium, such as the one depicted on Figure 4) the attenuation matrix will represent arbitrarily large frequencies in the direction of the discontinuity; this behavior is equivalent to binary occlusion.

Note that for locally constant and linearly varying volumes, the absorption does not affect the spectral covariance of the signal. In this case the effect of attenuation is simply the change of the weight we will use when combining covariance matrices from multiple light paths that we describe in Section 5.

4.2 Scattering

In this section we derive the matrix formulation of the change in spectral covariance of a local light field, along a ray that is scattered in a participating medium. Starting from the scattering equation, we perform a first order analysis of the integral, and compute the Fourier transform of the approximated local light fields.

The scattering equation used in raytracing expresses the local increase of radiance at \mathbf{x} , in direction ω_s due to light scattering from all incoming directions ω according to the phase function ρ [Cerezo

et al. 2005]:

$$\frac{\partial(l(\mathbf{x} + t\omega_s, \omega_s))}{\partial t}(0) = \frac{\kappa_s}{4\pi} \int_{\omega \in S^2} \rho(\omega, \omega_s) l(\mathbf{x}, \omega) d\omega \quad (8)$$

Integrating for a small traveling distance δt , we obtain:

$$l(\mathbf{x} + \delta t\omega_s, \omega_s) = l(\mathbf{x}, \omega_s) + \frac{\delta t \kappa_s}{4\pi} \int_{\omega \in S^2} \rho(\omega, \omega_s) l(\mathbf{x}, \omega) d\omega \quad (9)$$

When performing Monte-Carlo rendering in participating media, the sum on the right is handled by deciding with Russian Roulette whether the light path scatters or not. Consequently, to study scattering along a light path that is known to scatter, we need to deal with the integral term of the above equation only.

4.2.1 Scattering the local light fields. We study the implication of this equation in the 4D neighborhood of a couple of directions ω_i and ω_s , making a finite angle α (In other words, $\cos \alpha = \omega_i \cdot \omega_s$). We derive the scattering equation for small perturbations around the incoming and outgoing directions.

We consider the incoming and outgoing light fields to be defined in correlated angular frames, for which the first angular component lies in the plane containing ω_s and ω_i , as depicted in Figure 5. Let $(\delta\theta, \delta\phi)$ and $(\delta\theta', \delta\phi')$ be the angular coordinates of the incoming and outgoing light fields in these frames, $R_{\delta\theta, \delta\phi}$ the rotation that turns the central direction into the local light field direction $(\delta\theta, \delta\phi)$. We also denote by $(\delta u, \delta v)$ (resp. $\delta u', \delta v'$) the spatial components of the 4D frame around ω_i (resp. ω_s).

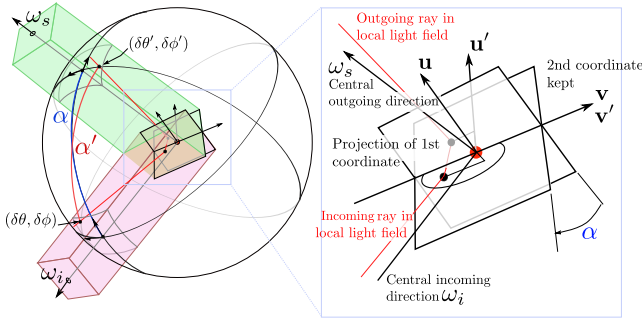


Fig. 5. Notations for scattering in 3D, with the input and output angular coordinate systems aligned. The local light field around the central incoming direction ω_i (resp. scattered direction ω_s) is parameterized with spatial and angular coordinates $\delta u, \delta v, \delta\theta, \delta\phi$ (resp. $\delta u', \delta v', \delta\theta', \delta\phi'$). The change in spatial coordinates implies a projection of the first spatial component only, scaling it by $\cos \alpha$.

With this notation in place, we express the scattered contribution of the light field around ω_i to the light field around ω_s , by restricting the integration domain in Equation 9 to local coordinates around the two vectors:

$$l_s(\delta u', \delta v', \delta\theta', \delta\phi') = \frac{\delta t \kappa_s}{4\pi} \int_{\delta\theta, \delta\phi} l_i(\delta u' \cos \alpha, \delta v', \delta\theta, \delta\phi) \rho(R_{\delta\theta, \delta\phi} \omega_i, R_{\delta\theta', \delta\phi'} \omega_s) \quad (10)$$

Note also that in the integrand l_i uses *outgoing* spatial coordinates $\delta u', \delta v'$, but *incoming* angular coordinates $\delta\theta, \delta\phi$. Reparameterizing spatial coordinates corresponds to a projection of the first spatial coordinate, resulting in the 1D scale by $\cos \alpha$ that expands the signal along the first spatial coordinate.

We suppose that $\rho(\omega_i, \omega_s)$ only depends on the relative position of the two vectors, in which case we can express it as $\rho(\cos \alpha)$. Given an input angular perturbation $(\delta\theta, \delta\phi)$ and output angular perturbation $(\delta\theta', \delta\phi')$ in the equatorial parametrization, the angle α' between those directions obeys the law of cosines (See for instance [Todhunter 1859] page 18, and Figure 5), which for small perturbations, boils down to:

$$\cos(\alpha') = \cos(\alpha + \delta\theta_i - \delta\theta_s) \cos(\delta\phi_i - \delta\phi_s)$$

We adopt the following notation for the phase function in the neighborhood of (ω_i, ω_s) :

$$\bar{\rho}(\delta\theta' - \delta\theta, \delta\phi' - \delta\phi) = \rho(\cos(\alpha + \delta\theta - \delta\theta') \cos(\delta\phi - \delta\phi'))$$

Equation 10 becomes:

$$l_s(\delta u', \delta v', \delta\theta', \delta\phi') = \frac{\kappa_s \delta t}{4\pi} \int_{\delta\theta, \delta\phi} l_i(\delta u' \cos \alpha, \delta v', \delta\theta, \delta\phi) \bar{\rho}(\delta\theta' - \delta\theta, \delta\phi' - \delta\phi)$$

In the angular domain, this is a 2D convolution between the incident light field and the phase function ρ in the neighborhood of directions ω_i, ω_s .

4.2.2 Spectrum covariance formula. Given this formulation, going to Fourier space then follows in a straightforward manner. We simply take the Fourier transform on both sides, to get:

$$\hat{l}_s(\Omega_u, \Omega_v, \Omega_\theta, \Omega_\phi) = \frac{\kappa_s \delta t}{4\pi \cos \alpha} \hat{l}_i \left(\frac{\Omega_u}{\cos \alpha}, \Omega_v, \Omega_\theta, \Omega_\phi \right) \hat{\rho}$$

To translate this relationship into covariance matrices, we apply the formulae summarized in Figure 3: the convolution adds angular bandwidth to the inverse of the covariance (and thus effectively removes angular bandwidth), and the scale by $1/\cos \alpha$ scales the covariance by $\cos^2 \alpha$. The outside factor is kept away for normalization according to Equation 1:

$$\Sigma_s = ((V_\alpha \Sigma_i V_\alpha)^{-1} + S)^{-1} \quad (11)$$

where S is the covariance matrix of the scattering operator. S depends on the 2D covariance matrix σ_ρ of the windowed Fourier spectrum of $\bar{\rho}$, and V_α is the scale matrix due to the spatial reparameterization:

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_\rho^{-1} \\ 0 & 0 & 0 & 0 \end{bmatrix} V_\alpha = \begin{bmatrix} \cos \alpha & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12)$$

To compute matrix σ_ρ , we use Equation 2, which directly gives the spectral covariance of the phase function around (ω_i, ω_s) from the Hessian matrix of $\bar{\rho}$ (we suppose that the Hessian is diagonal, without loss of generality. Otherwise, an additional rotation is needed just like what we did for absorption):

$$\sigma_\rho = \frac{1}{4\pi^2 \bar{\rho}(0,0)} \begin{bmatrix} \left| \frac{\partial^2 \bar{\rho}}{\partial \theta^2}(0,0) \right| & 0 \\ 0 & \left| \frac{\partial^2 \bar{\rho}}{\partial \phi^2}(0,0) \right| \end{bmatrix} \quad (13)$$

The effect of scattering is therefore very similar to what a BRDF does on the local light field: it removes frequencies.

It is also interesting to note that for $\alpha = \frac{\pi}{2}$, the spectrum covariance in Ω_u is totally removed by the above equation. This is because in this case, the incoming and outgoing directions are perpendicular, and therefore no variation along u on the incoming light affects the outgoing light field. Note also that for a general light

path scattering multiple times in a volume, Equation 11 needs to be interleaved with rotations to correctly align coordinate systems between two scatter events.

In summary, we proved that the effect of the scattering operator to the covariance matrix will be: a BRDF operator followed by a scaling of the spatial component. We will now give an example of how to compute S when ρ is the Henyey-Greenstein function.

4.2.3 Covariance of the Henyey-Greenstein phase function. There are multiple analytical models of phase functions available [Gutierrez et al. 2009]. As a practical example, we give the formulas of the spectral covariance matrix for the Henyey-Greenstein phase function, that is most common in the field. This function is defined using angle α between the incoming and outgoing directions, as

$$\rho_g(\omega_i, \omega_s) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2g \cos \alpha)^{\frac{3}{2}}} \text{ with } \cos \alpha = \omega_i \cdot \omega_s$$

We show in Appendix B that the spectral covariance of the Henyey-Greenstein function locally around ω_s is:

$$\text{cov}(\hat{\rho}_g) = \frac{1}{4\pi^2} \begin{bmatrix} |h_{11}| & 0 \\ 0 & |h_{22}| \end{bmatrix} \quad (14)$$

with

$$h_{11} = \frac{3g(2(g^2 + 1) \cos \alpha + g(3 \cos(2\alpha) - 7))}{2(g^2 - 2g \cos \alpha + 1)^2}$$

$$h_{22} = \frac{3g \cos \alpha}{g^2 - 2g \cos \alpha + 1}$$

As expected, the covariance is isotropic for $\alpha = 0$ (i.e. $h_{11} = h_{22}$) since the Henyey-Greenstein function is rotationally symmetric, and is null for $g = 0$, since the function is constant in this case. We validate these equations in Figure 6 with a comparison of ground truth and predicted covariance matrices for two different values of α .

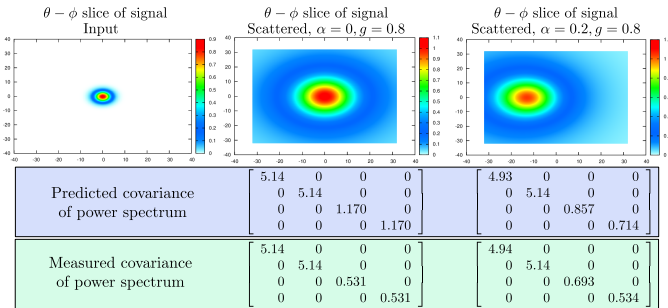


Fig. 6. Validation of Equations 11 and 14. We measure the covariance of the windowed spectrum of an input light beam (left) after scattering with two different angles (resp. $\alpha = 0$, middle, and $\alpha = 0.2$, right), for a Henyey-Greenstein parameter $g = 0.8$. Top row: Angular slice of the signal (in the primal domain). Middle row: Predicted covariance matrix of the spectrum. Bottom row: measured covariance matrix, from the 4D data sampled with 64^4 in $[-1, 1]^4$. Given the order of magnitude of numbers involved in the calculation, the measured and predicted values are very close (besides, the square root of the diagonals must be compared). Our calculation is conservative and well predicts the behavior of the measured data.

Summary

In this section we have derived equations to compute the spectral covariance of the local light field along a light path inside participating media. We have shown that absorption increases frequency content and acts as the previously defined occlusion operator. Scattering low-pass filters the angular frequencies of the input local light-field with a bandwidth defined by the phase function. Thus, it acts like the BRDF operator.

5. SAMPLING AND INTEGRATION METRICS IN 3D

In Section 4, we performed an analysis of scattering and attenuation in the 4D space of local light fields along rays. In participating media, light bounces in all directions, and the covariance of a single ray cannot be used to predict the overall frequency characteristics of the light distribution. In this section we will see how to leverage the 4D local analysis to compute a set of sampling metrics in 3D, by combining the covariance from many rays.

We consider the following metrics: *image-space bandwidth* will enable efficient image space sampling and reconstruction; the *variance in the volume along a ray* will prove useful to optimize the placement of integration samples for integrating illumination along a ray; and finally, a prediction of *volumetric bandwidth* will predict the optimal size of density estimation kernels to improve volumetric integration techniques, such as progressive photon mapping.

Combining the 4D local covariance of many rays into a single 3D field also has favorable practical consequences: we favor reusing the covariance of a small subset of light paths by storing it in a buffer. However, since the spectral covariance of radiance is directional it would ideally need to be stored in a 5D buffer, a potentially computationally and memory intensive datastructure. Fortunately, a good compromise is to base our metrics on the frequency covariance of the volumetric fluence only, which requires a spatial (3D) buffer, at the cost of a very reasonable approximation.

5.1 The volumetric covariance

The volumetric covariance is the covariance of the power spectrum of the fluence in a volume. It bridges the gap between the light path formulation of covariance derived in Section 4, and our proposed practical improvements of sampling strategies in existing global illumination methods.

We define the volumetric covariance to be the 3×3 covariance matrix $\Gamma_{\mathbf{x}}$, where entry (i, j) is the ij -covariance of the Fourier transform of the fluence $F_{\mathbf{x}}$ in the neighborhood of \mathbf{x} :

$$(\Gamma_{\mathbf{x}})_{ij} = \frac{1}{F_{\mathbf{x}}(0)} \int \Omega_i \Omega_j \widehat{F_{\mathbf{x}}}(\Omega) d\Omega \quad (15)$$

The local fluence F at an offset \mathbf{s} around the point \mathbf{x} is defined as:

$$F_{\mathbf{x}}(\mathbf{s}) = \int_{\omega \in S^2} l(\mathbf{x} + \mathbf{s}, \vec{\omega}) d\omega$$

In practice, the volumetric covariance is computed and stored in a voxel grid, and the size of the neighborhood considered for each voxel is the size of the voxel itself (so, $\mathbf{x} + \mathbf{s}$ is restricted to lie in the voxel).

Computation. We compute the volumetric covariance by accumulating contributions of individual light paths traversing the volume. At a point \mathbf{x} , the 4D spectral covariance Σ of an incident light path in direction ω carries the illumination from a very localized set of directions around ω . The 2D spatial sub-matrix of the 4D covariance of the local light field around ω is therefore a slice of the

3D covariance of the integrated radiance, in the plane orthogonal to ω .

Consequently, we compute the covariance of the fluence at \mathbf{x} by summing up the 2D spatial slices of the covariance matrices of each incident light path, padded to 3D with zeroes, and rotated to match the world coordinate system. Since Σ lives in Fourier space, and the summation happens in the primal domain, submatrices need to be extracted from Σ^{-1} and inverted back to Fourier space after summation:

$$\Gamma_p = \left(\int_{\omega \in S^2} R_\omega \Sigma^{-1}|_{\delta x, \delta y} R_\omega^T I(\omega) d\omega \right)^{-1} \quad (16)$$

In this equation, the notation $\Sigma^{-1}|_{\delta x, \delta y}$ refers to the 2D spatial sub-matrix of matrix Σ^{-1} , while R_ω is the 3×2 matrix converting the two local spatial coordinates around ω into the three coordinates of the world. Finally, $I(\omega)$ is the normalized incident energy along incoming direction ω .

In practice, the integral in Equation 16 is computed using a classical Monte Carlo summation, as light paths in the volume cross voxels they contribute to. We do not need to explicitly compute $I(\omega)$ since it is naturally handled by the photon tracing approach: the number of path crossing a voxel is proportional to the fluence. We only record how many times each voxel was hit, for proper normalization.

5.2 Image-space covariance

We want to compute image-space covariances for adaptive sampling. The angular sub-matrix of the covariance Σ at the camera can be used to derive sampling densities and reconstruction filters for ray-tracing, at each pixel [Belcour et al. 2013].

The most straightforward method to obtain Σ for each pixel in the screen would be to accumulate covariance matrices from light paths reaching the camera, applying the theory of Section 4. While this eventually provides an unbiased estimate of the image-space covariance, it needs many light paths to obtain a reasonably noise-free estimate.

It is the spatial variations of “light intensity” in the participating medium that will show up as angular bandwidth at the camera, and after projection, as spatial bandwidth on the screen [Durand et al. 2005]. Consequently, we propose computing screen-space bandwidth from Γ . To compute Σ at the camera, we slice the volumetric

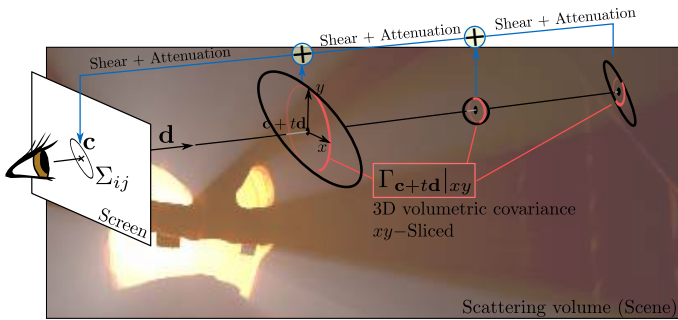


Fig. 7. To estimate the 2D spatial covariance matrix in image space, we accumulate slices of the volumetric covariance Γ along the ray using Equation 17, and the equations of attenuation 6 and 7.

covariance Γ orthogonally to camera rays, pad it to 4D with null angular covariance, and accumulate it using Equations 6 and 7 to account for the attenuation between points along the ray and the

camera. At pixel p , corresponding to a ray with origin c and direction d :

$$\Sigma(p) = \frac{1}{\int K(t) dt} \int_{t=0}^D K(t) T_d^T (A(t) + \Gamma_{c+td}|_{xy}) T_d dt \quad (17)$$

The normalization constant $K(t)$ accounts for how much energy is associated with each voxel in the covariance grid. The resulting matrix is a 4D covariance from which we extract the angular component at the camera. The process is illustrated in Figure 7.

Equation 17 is an approximation because it implicitly supposes that the last bounce of light before the camera has no angular covariance, meaning that the last scattering step is isotropic. In practice we found this approximation to have no visible effect.

5.3 Ray-space variance

When gathering energy in participating media, one needs to integrate illumination along rays. For each ray, we need to make sure that the spacing between integration samples avoids aliasing with respect to how much the illumination varies in the volume. That requires the variance $v(t)$ of the fluence function along the ray. The variance of the fluence in a particular direction ω is naturally given by a 1D slice of the volumetric covariance matrix, in direction ω (supposedly a unit vector):

$$v(\mathbf{x}, \omega) = \omega^T \Gamma_{\mathbf{x}} \omega \quad (18)$$

5.4 Optimal kernel size for density estimation

Methods based on density estimation such as photon mapping, need to carefully adapt the size of the kernel to collect photons: too small a kernel increases variance, while too large a kernel increases bias. Silverman gives an estimate of the mean integrated square error for density estimation (see Silverman’s monograph [1986], page 85) which depends on the Laplacian of the estimated function. In the case of photon mapping methods the function is the fluence, and Silverman’s estimate gives:

$$\text{bias}_h(\mathbf{x}) = \frac{1}{2} h^2 a \Delta F(\mathbf{x})$$

In this formula, h is the radius of the kernel used for density estimation, and a is a constant that only depends on the shape of the kernel. We use again equation 2, that links the diagonal of the covariance matrix to the absolute value of the partial second derivatives of $F(\mathbf{x})$, to obtain an upper bound on the absolute value of the Laplacian from the trace of the covariance matrix:

$$|\Delta F(\mathbf{x})| \leq 4\pi^2 F(\mathbf{x}) \text{tr}(\Gamma_{\mathbf{x}})$$

Equality holds when the second derivatives share the same sign. From this, we have an upper bound on the density estimation bias:

$$|\text{bias}_h(\mathbf{x})| \leq 2\pi^2 h^2 a F(\mathbf{x}) \text{tr}(\Gamma_{\mathbf{x}}) \quad (19)$$

This equation directly gives us the largest possible kernel radius h to always keep the bias below a given threshold.

Summary. In this section we have explained how to combine the 4D spectral covariance of many rays into volumetric 3D covariance. We have derived interesting sampling metrics from volumetric covariance. In the next section, we will explain how to use these metrics to improve existing rendering algorithms in three different application scenarios.

6. IMPROVEMENT OF EXISTING SAMPLING AND RECONSTRUCTION METHODS

In this section we demonstrate the usefulness of our analysis from Section 4, and the sampling prediction metrics we derived in Section 5. We examine four different calculation steps that are involved in computational methods of global illumination in participating media, and we show that our sampling metrics can be used to accelerate them.

Our sampling metrics need the computation of volumetric 3D covariance, as defined in Section 5. To compute and store volumetric covariance, we use a voxel grid, the *covariance grid*. In the use cases below, we always read the values of Γ in that grid to compute the required metrics. All results are computed on an Intel i7-3820 with four cores at 3.60GHz per core and an NVidia GeForce GTX 680. We use 8 threads to benefit from hyperthreading. Unless noted, we use a 64^3 covariance grid. The covariance grid population algorithms run on a single thread, while we use multi-processor capabilities for volume integration (Section 6.3 and 6.4 using OpenMP) and for density estimation (Section 6.5 using CUDA).

6.1 The covariance grid

We sample light paths, and populate the covariance grid using Equation 16. We also record how many times each voxel in the grid is visited by light paths, so as to maintain information for proper normalization.

This calculation is not view-dependent. Depending on the application, we populate the covariance grid using a fixed proportion of the light paths used for the simulation (in Section 6.5), or fill it up once before the simulation (Sections 6.3 and 6.4). Figure 9 references the values used for the different scenes. For the algorithms of Section 6.3 and 6.4, ray marching and filling the covariance grid with 100,000 light paths takes 21 seconds for a 64^3 covariance grid with the Halloween scene. We used as many as 10,000 light paths for the Sibenik scene, as the lights are spot lights. With this amount of light paths, it took 8 seconds for ray marching and filling for the 64^3 covariance grid.

Figure 8 shows the volumetric covariance predicted by our system in three different locations of a scene showing volumetric caustics and shadows.

6.2 Efficient prediction of the Hessian of the fluence

The covariance grid naturally enables a stable computation of second derivatives of the fluence. In this section we study the benefit of estimating second derivatives from the covariance in frequency space using Equation 2, rather than trying to estimate second derivatives in the primal domain using a finite differences scheme.

We present such a comparison in Figure 10, in a simple volume containing a point light source, and for two different integration schemes: a 3-points second derivative estimate, which naturally proves to be very noisy, and a heavily filtered estimate using the second derivative of a Gaussian over 21^3 neighbor voxels. This experiment not only proves that our model gives a correct estimate of the second derivatives, but also that it converges faster than methods based on the primal domain. This is not surprising, because our method does not require explicit differentiation over the path-traced illumination the way the primal domain estimate does.

Methods that rely on linear interpolation between 3D illumination samples in the volume theoretically have an interpolation error that is proportional to the second derivatives of the illumination. This has been proven for surface irradiance caching meth-

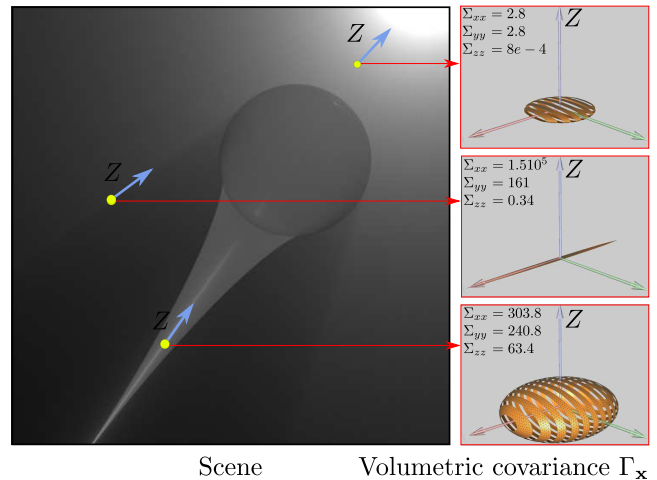


Fig. 8. The volumetric covariance Γ is the covariance of the fluence around each point in the volume. We show values of Γ from the *covariance grid*, as an ellipsoid (iso-value of the 3D Gaussian with same covariance). *Top*: in the region of diffusion, Γ is really small. *Middle*: on the shadow boundary, Γ is large along the normal to the shadow volume. *Bottom*: in the caustic, Γ is large orthogonally to the direction of the caustic (Caution: All graphs are normalized to help demonstrate the shape of the covariance. To compare the covariance quantitatively, look at the eigenvalues listed for each dimension).

		Halloween	Sibenik
Image Space	Cov. grid samples	100 000	10 000
	Step size	0.01	0.002
Eye Path	Cov. grid samples	100 000	10 000
	Step sizes	0.1 – 0.01	0.1 – 0.002
Naive	Step size	0.01	0.002
Prog. Photon Beams	α	0.7	0.7
	Samples per pass	5 000	5 000

Fig. 9. We list the different parameters used for our results section. We report the number of light paths used to fill the covariance grid, the distance between samples along the eye ray for the integration in the volume, and Progressive Photon Beams [Jarosz et al. 2011] parameters (radius reduction ratio α , and the number of light paths sampled per pass). We report only scenes that are common to the three algorithms. We used $\sigma_s = 0.1$, $\sigma_a = 0.1$ and $g = 0.5$ for the parameters of the volume. For our adaptive sampling and image space filtering algorithms, we used 8 jittered supersampling samples per pixel to obtain anti-aliasing.

ods [Schwarzaupt et al. 2012], and such an error estimate outperforms existing heuristics [Jarosz et al. 2008].

Although extending our work to volumetric irradiance caching is beyond the scope of this paper, we believe that the irradiance caching error estimate based on the Hessian can be extended to volumes, with an error estimate that is proportional to the Laplacian of the fluence (see, for instance, Equations 5 and 6 in [Schwarzaupt et al. 2012]), and where the shape of the influence regions of irradiance cache samples will be ellipsoids aligned with the eigenvectors of the Hessian of the fluence in the volume. This opens interesting research avenues toward the replacement of existing heuristics for error and influence regions for volumes [Ribardière et al. 2011], especially removing the need for special treatment of occlusion [Schwarzaupt et al. 2012] that the covariance analysis naturally handles.

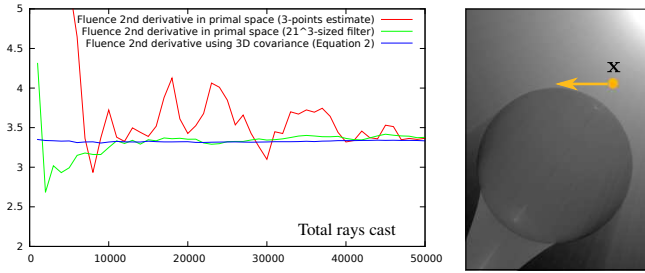


Fig. 10. Comparative estimate of the second derivative of the fluence in the primal domain (red and green curves) versus Equation 2 (blue curve), for point x in the direction of the arrow, as a function of the number of beams cast. In both cases we used grids of 128^3 voxels to store the covariance and the fluence. In the primal space, the second derivative is estimated in two ways from the fluence grid. *Red curve*: using a finite difference scheme between immediate neighbor voxels. *Green curve*: using a very large Gaussian filter around the measured voxel (averaging the nearby 21^3 voxels). For the *blue curve*, we simply applied Equation 2 to the covariance matrix picked at that voxel in the covariance grid, without filtering. With as low as 5000 total rays cast in the scene our estimate outperforms the costly filtered estimate in the primal domain.

6.3 Image adaptive sampling and reconstruction

An effective method for rendering images with varying local bandwidth is to compute image space converged illumination samples, and filter these samples using an appropriate 2D reconstruction kernel. For an optimal result, it is necessary to know in advance the optimal sampling density (in samples per square pixels), and the shape of the reconstruction filter at each pixel [Belcour et al. 2013] or to compute it from a subset of the light paths used in the image [Overbeck et al. 2009; Rousselle et al. 2011].

The optimal sampling densities $N(p)$ and the shape of the 2D image reconstruction filter f_p can be derived for each pixel p from the covariance $\Sigma(p)$ of the local light field at that particular pixel [Belcour et al. 2013]:

$$N(p) = k\sqrt{|\Sigma(p)|} \quad \text{and} \quad f_p(x) = e^{-\frac{1}{2}x^T(\Sigma(p)^{-1})|_{x,y}x} \quad (20)$$

In this expression, $\Sigma(p)^{-1}|_{x,y}$ is the spatial slice of the inverse of the spectrum covariance matrix in the image plane at pixel p . In other words, $\Sigma(p)$ is the covariance of the Gaussian whose variance matches the bandwidth of the image according to the Nyquist rate. In practice, for each pixel, we trace a single ray through the covariance grid and apply Equation 17 to compute $\Sigma(p)$.

The number of samples per square pixel is proportional to the determinant of the screen-space spectrum covariance, and the shape of the filter is obtained by slicing the covariance of the signal along the spatial dimensions at each pixel. The constant k lets us express $N(p)$ as a fraction of the total number of samples allocated for the entire image.

To compute the image, we obtain the required number of samples per pixel using Equation 20 and form an *image sampling density map*. We use this map as a probability density to draw pixels to be computed, using rejection sampling. For each pixel to compute, we estimate the radiance using path tracing. Each image sample is therefore a converged illumination value. Finally, we reconstruct the image by filtering the image samples around each pixel p with the filter f_p that is given by Equation 20.

This computation is efficient because it samples the image very sparsely when the resulting image is predicted to be smooth. Figure 11 shows the result of such a computation for the pumpkin

scene. The number of computed pixels is 43% of the total number of pixels in the image. The results also show that we correctly predict the shape and size of reconstruction filters. For more trivial scenes, the gain is even better.

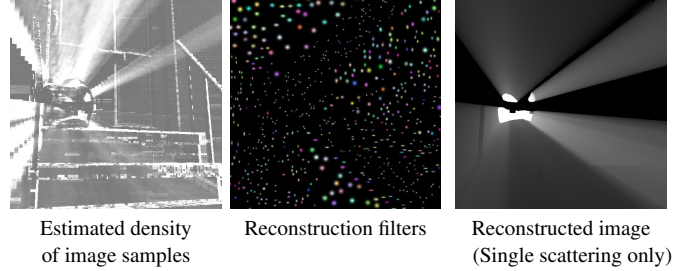


Fig. 11. We demonstrate the use of our prediction metrics for image space filtering and reconstruction of *single scattering*. We predict the varying density of image samples to compute using Equation 20 (*left*) as well as the shape of the Gaussian filter to reconstruct from these samples at each pixel (*middle*), to reconstruct the image (*right*).

6.4 Adaptive sampling along a ray

For each pixel that we compute, we need to integrate the illumination that is scattered towards the camera. For this we integrate the radiance after a last scattering event. Instead of using uniform samples, we adapt the sampling density to the variance of the fluence along the ray to the camera, as computed in Section 5.3. The variance is directly computed from the covariance grid using Equation 18. This allows us to place samples in regions where the path crosses rapid changes in illumination, which can be caused by volumetric shadows, for example, as illustrated in Figure 12. In practice we first uniformly sample the radiance along the eye path, and then we use additional samples in proportion to the local variance estimate along the path.

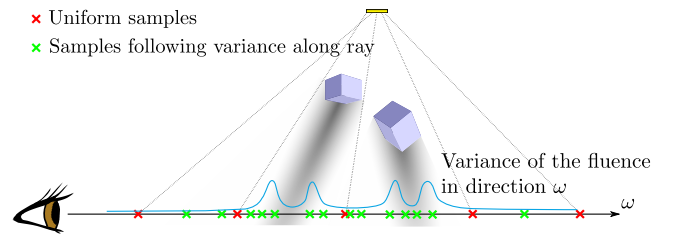


Fig. 12. Computing the amount of light reaching the camera through a given pixel requires integrating the radiance scattered by the participating medium towards the camera. Using an arbitrary number of uniformly spaced integration samples (red) might do a bad job if not accounting for the distribution of light into the volume. We propose instead to distribute samples according to the local variance of the fluence along the ray (green), in order to adaptively sample high frequency regions.

We provide a simplified algorithm for the adaptive integration used with Algorithm 1. It uses Equation 18 to evaluate the required distance between two samples in the volume.

Since our algorithm takes advantage of adaptive sampling on shadow boundaries, we are able to reduce the aliasing of light shafts caused by undersampling high frequency regions. Figure 13 shows

Algorithm 1 Our adaptive sampling algorithm compute the single scattering radiance for an eye ray defined by its position x in space and direction ω . It returns the light scattered by the volume in the interval $[0, T_{max}]$ along the ray. Our variance estimate v (Equation 18) provides a step size in the volume. Note that the last step requires special treatment as the step might be larger than the remaining integration distance. We do not show it in this example to keep a compact formulation.

```

function ADAPTIVEINTEGRATION( $\mathbf{x}$ ,  $\omega$ ,  $T_{max}$ ,  $step_{min}$ ,
 $step_{max}$ )
     $rad = 0$ 
     $t = 0$ 
    while  $t \in [0..T_{max}]$  do
         $\mathbf{x}_t = \mathbf{x} + t\omega$ 
         $dt = \frac{1}{2\sqrt{v(\mathbf{x}_t, \omega)}}$ 
         $dt = \text{clamp}(dt, step_{min}, step_{max})$ 
         $rad = rad + dt \text{ integrateRadiance}(\mathbf{x}_t, -\omega)$ 
         $t = t + dt$ 
    end while
    return  $rad$ 
end function

```

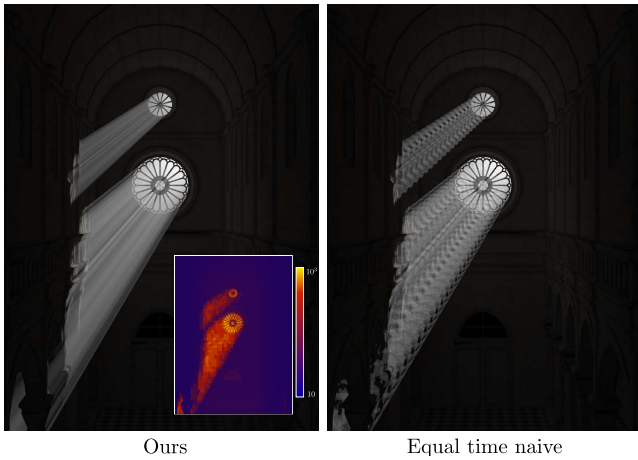


Fig. 13. We compare our variance-driven integration method to naive uniform sampling, at equal computation time (14 minutes). Our adaptive sampling clearly removes aliasing caused by the shaft from the Rose window. Inset: total number of samples used per pixel for our algorithm. (Model of the Sibenik cathedral by Marko Dabrovic).

that our results on the Sibenik cathedral model outperforms uniform sampling at equal computation time, in the detailed shafts.

We summarize the timings of the image space adaptive sampling and the eye path adaptive sampling algorithm compared with an equal quality naive raytracing approach in Figure 14. Both algorithms save computation time by adapting the workload to high frequency regions.

We investigated different resolutions for the covariance grid (Figure 15). A size of 64^3 for the grid was sufficient for all our scenes. Coarser grids will provide a conservative estimation of frequencies and lead to poor performances, while finer grids will naturally increase the cost of ray marching in this structure. The cost of filling the grid is linear with the grid edge size. For the Sibenik scene using 10K light-paths, ray marching and filling took 4s for a

scene	Image space	Eye space	Naive
Sibenik	19m	14m	1h 40m
Halloween	7m	6m 30s	22m

Fig. 14. Our adaptive sampling and image space filtering approaches save computation time compared to a naive raytracing approach for the same quality. Eye path adaptive sampling and the naive implementation use 8 samples per pixel for antialiasing. The image space method adapts the number of samples up to this limit.

32^3 grid, 8s for a 64^3 grid, and 17s for a 128^3 grid. We found that a 64^3 grid provides a good trade-off between construction time, quality and time required to ray march during rendering (see Figure 15), in all our tests, except for San Miguel where we needed a 256^3 grid.

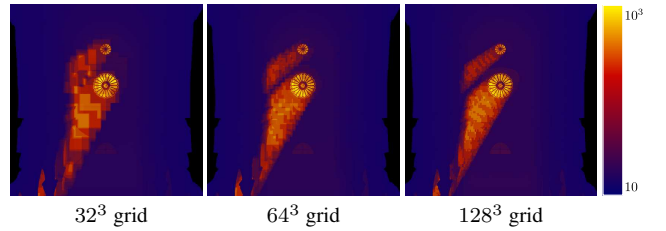


Fig. 15. We analyse the impact of various resolutions of the covariance grid (32^3 , 64^3 and 128^3) on the prediction of the required number of samples along camera rays. Smaller grid sizes bring more conservative results while larger grids are more costly to handle. A size of 64^3 performs well for most scenes.

6.5 Improved multiple scattering photon beams

We study the possible improvement of the convergence rate of *progressive photon beams* (PPB), to illustrate the benefits of frequency analysis.

In the original algorithm, photons are traced in the scene containing a participating medium and the paths of propagation (called beams) are stored [Jarosz et al. 2011]. Then, for each pixel, rays are shot from the camera, and the density of beams along the ray is estimated using a 2D circular kernel. This is repeated while decreasing kernel size until convergence is satisfactory.

Just like any other density estimation technique, the PPB algorithm fights between too small a reconstruction kernel—causing variance—and too large a reconstruction kernel—causing bias. Whereas the original algorithm keeps reducing the kernel sizes as more beams come along, we can afford to stop reducing it as soon as this size ensures that the density estimation bias is below a certain threshold. We know exactly when this happens from Equation 19.

During the gathering pass, for each eye ray, we test for its distance d to the beams stored (Figure 16). At the closest point to each beam along the ray, we look into the covariance matrix, and estimate the ideal gathering radius r_σ using Equation 19. We gather that beam only if:

$$d < \max(r_i, r_\sigma)$$

where, r_i is the radius given by the photon beam method for pass #i. In other words, we replace the gathering radius of progressive photon mapping by a specific radius for each pair (eye-ray, photon beam) that is adapted to the local variations of the signal. This

Table III. Images resolutions used for our rendering tests.

scene	San Miguel	Cornell Box	Sibenik	Halloween
resolution	1000 × 1024	512 × 512	714 × 968	512 × 512

adaptive radius computation stops us from decreasing the radius in regions of low bandwidth, and therefore significantly reduces variance, while keeping the error uniformly controlled. We implemented this gathering in a CUDA kernel.

We follow a pure ray tracing approach in our PPB implementation. We generate light paths and eye paths on the CPU and transfer them on the GPU. It lets us simplify the generation of rays, to follow specular paths from the camera and to avoid duplicating the scene data on the GPU. This explains the timing differences between our PPB implementation and the one of Jarosz et al. [2011].

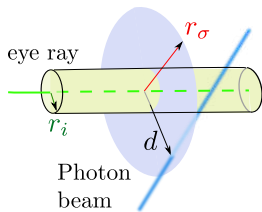


Fig. 16. Given a camera ray (green) and a beam, we use the radius r_σ , estimated by the covariance analysis, instead of the radius r_i of the progressive photon mapping, when r_i is smaller. Using this, we gather more beams in low frequency regions and decrease the variance of the image.

We validate our improvement of progressive photon beams using the San Miguel scene (Figure 17), Halloween scene (Figure 18), Cornell box (Figure 19), and Sibenik cathedral (Figure 20) scenes. In all cases, our covariance framework correctly estimates the high frequency regions due to the illumination. San Miguel, Halloween, and Sibenik (Figures 17,18, and 20) scenes have significant indirect illumination; they prove that our method converges faster than PPB. San Miguel also demonstrate the scalability of our technique. The non-homogeneous Cornell box scene (Figure 19) validates that our analysis and filtering methods correctly handle non-constant scattering parameters. In this example, the scattering parameters are varied based on Perlin noise. Image resolutions are reported in Table III.



(a) Our improved PPB, 9.7M beams, (b) Standard PPB, 10M beams, 25min

Fig. 18. The Halloween scene combines high frequency white shafts with a low frequency orange multiple scattering. Our covariance prediction allows to filter out the noise due to the diffuse component while preserving edges of shafts.

At equal computation time, we achieve a much better convergence in smoother regions of the image, while we approximately keep the same convergence in the highest frequency regions such

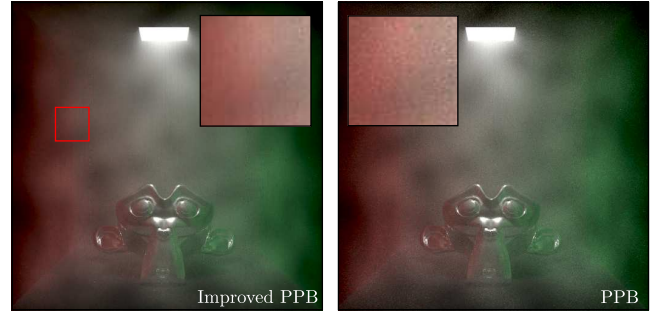


Fig. 19. In this figure, a non-homogeneous volume is illuminated. The walls of the Cornell box provide indirect color bleeding from diffuse reflections. We compare our algorithm after a run of 31 minutes (8.5M beams) with an equal time run of Progressive Photon Beams (10M beams). Our algorithm reduces noise thanks to our adaptive density estimation.

as shaft borders and caustics, as predicted. However, our method will always eventually stop reducing the kernel size, contrary to the classical photon beam approach. It just happens later for higher frequency regions of the image.

7. IMPLEMENTATION AND DISCUSSION

7.1 Implementation

Taking advantage of symmetry, 4D covariance matrices only need 10 floats to store (instead of 16) into light rays equipped with covariance information. An additional float is also needed to keep track of light intensity for proper normalization, since our definition of covariance has no units (see Equation 1). The combined covariance matrix Σ_{12} of two different paths with covariances Σ_1 and Σ_2 and intensities I_1 and I_2 , after proper alignment, is:

$$\Sigma_{12} = \frac{1}{I_1 + I_2} (I_1 \Sigma_1 + I_2 \Sigma_2)$$

We also store, along with the covariance information, the tangent frame of the local parametrization (two 3D normalized vectors).

A photon that carries covariance is initialized at the light source and covariance is updated as the photon path is sampled [Belcour et al. 2013]. For instance, a square diffuse light source will produce rays with null angular covariance and spatial covariance that depends on the size of the square.

During light propagation, the covariance matrix Σ of rays is updated when the light is reflected, refracted, or occluded using the Equations of Figure 3 (see Algorithm 2). Each time a ray is scattered we transform its covariance using Equation 11, and for each absorption event, we apply Equations 6 and 7. Eventually, the various operators involved boil down to sums, products and inverses (or pseudo-inverses) of 4D covariance matrices, which is carried on very efficiently using explicit formulas.

The covariance grid uses 6-float arrays to store the individual 3D spatial covariances Γ_p and an additional float for the light intensity normalization factor per voxel. We populate the covariance grid using Equation 16, basically summing up matrices multiplied by intensity. Depending on the application, the covariance grid is either populated once, using a fixed number of rays (for e.g., adaptive reconstruction and ray-space integration) or updated during the computation (for progressive photon beams, where only 10% of the light paths carry covariance information).

We did not use ray differentials in our implementation of PPB (nor in our improvement). Jarosz et al. [2011] showed that using ray

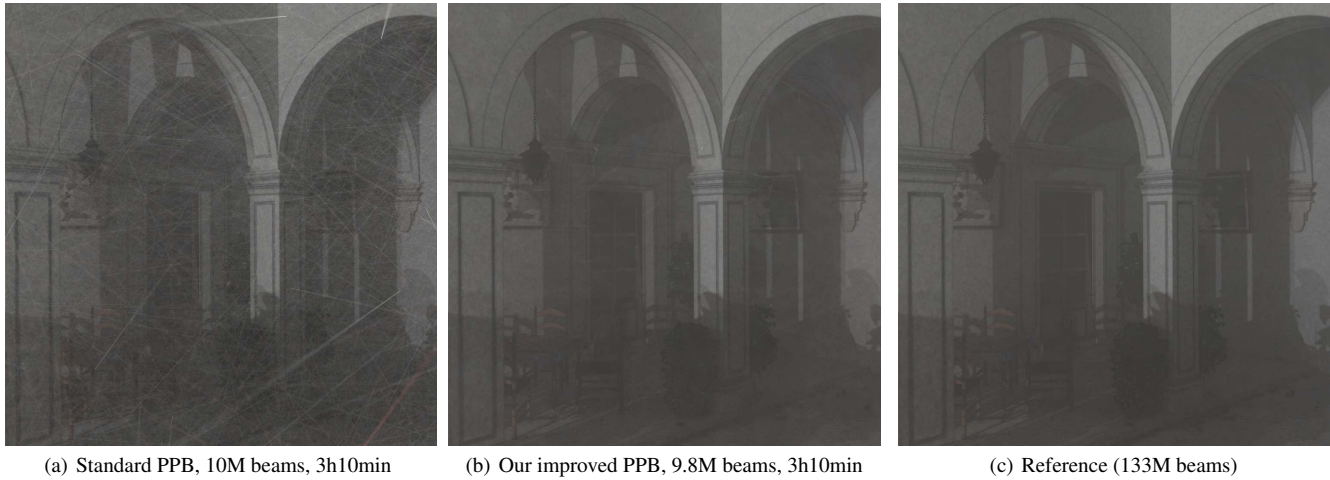


Fig. 17. The full San Miguel scene with a rather indirect illumination, shows a very smooth volumetric environment, in a significantly complex geometry (2.5M polygons). In this context, our frequency prediction method makes the progressive photon beams converge much faster. Because of the size of this scene, we used 256^3 -wide covariance and occlusion grids. The most severe limitation of covariance tracing in this scene was due to the occlusion grid size. Too small a grid would cause over-estimation of visibility, and conservatively populate the covariance grid with high frequencies. We used Morgan McGuire export of the San Miguel scene. As he noted on his website, <http://graphics.cs.williams.edu/data>, some geometry (chairs) and textures (walls and columns) are missing.

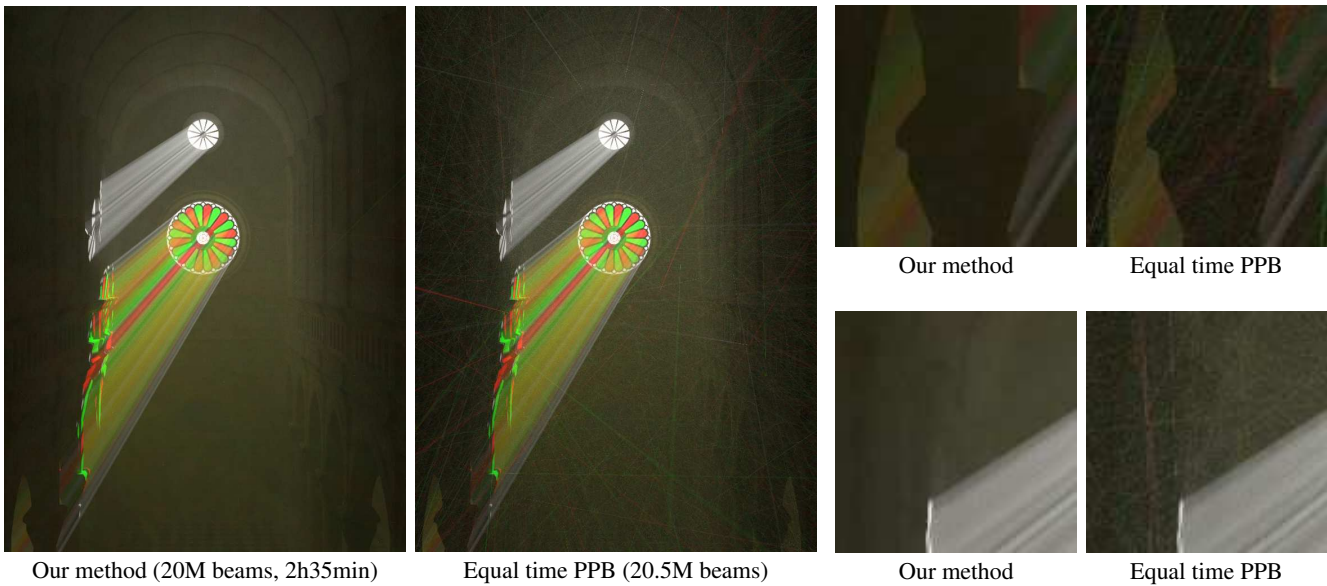


Fig. 20. Using our analysis metric our algorithm predicts the low frequency part of the volume where the progressive photon beam can afford to keep large collection radii, while controlling the bias. As a result, we provide a much better converged image at equal computation time than classical progressive photon beams; here, in a scene with multiple scattering.

differentials was beneficial for the convergence of specular paths from the light. But they use a constant beam radius for diffusely reflected or scattered beams. Since we compare the convergence of both algorithms for non-specular regions, this improvement of PPB was not necessary. Note that both algorithms would benefit equally from adding ray differentials.

We use a sufficiently large starting radius to improve the convergence of indirect effects while still keeping convergence of the

direct part. In all our examples, specular paths from the light are converged.

7.2 Limitations

The various techniques we present, based on our frequency analysis, effectively improve convergence in regions of the volume where under-sampling can be performed without loss of accuracy. If frequency is high everywhere—such as in a very highly varying medium, or in a volume where a large number of small shadow

Algorithm 2 The tracing of frequency photons is straightforward to implement. It only requires that we update the covariance matrix at specific steps. Note that it requires the ray tracing engine to compute specific information for intersection with geometry (such as local curvature). The R matrix is the factorized matrix of projection, alignment and curvature, before and after reflection. T is the covariance of the texture matrix.

```

function TRACEFREQUENCYPHOTON
   $\{p, \omega\} \leftarrow \text{sampleLight}()$ 
   $\Sigma \leftarrow \text{computeLightCovariance}()$ 
  while russianRoulette() do
     $p \leftarrow \text{traceRay}(p, \omega)$ 
    for all voxels  $v$  until hit do
       $\text{updateVoxelCovariance}(v, \Sigma)$ 
       $\Sigma \leftarrow T_d^T \Sigma T_d$ 
       $\Sigma \leftarrow \Sigma + O$ 
       $\Sigma \leftarrow \Sigma + A$ 
    end for
     $\omega \leftarrow \text{sampleBRDF}()$ 
     $\Sigma \leftarrow R_i^T \Sigma R_i$ 
     $\Sigma \leftarrow \Sigma + T$ 
     $\Sigma \leftarrow R_o^T (\Sigma^{-1} + B)^{-1} R_o$ 
  end while
end function

```

rays are cast—our *a priori* analysis naturally predicts that the sampling needs to be uniformly dense. In this case, the computation of covariance information would naturally not help improve convergence.

Using volumetric covariance implies an approximation, since it neglects the angular covariance of the incident light. Our method captures variations in the volumetric fluence which, for reasonably non-specular phase functions, remains close to the variance of the radiance, while only requiring a small storage cost. In the case of a very specular phase function at the last bounce before the camera, a 3D covariance grid is likely to produce a conservative over-estimation of the bandwidth. A more accurate approach would require also storing directional information into the covariance grid, and does not invalidate our frequency analysis.

The size of the covariance grid is another important limitation as it determines the scale at which we can optimize the radius reduction. A coarse grid will conservatively estimate small kernel sizes in low varying regions since high frequencies will leak outside of the region where they actually take place (This is illustrated in Figure 21).

The paraxial approximation used by the frequency analysis limits the capacity of our predictions to describe the full directional variations of light with a few photons. The paraxial approximation is valid for angles below one degree. However, using our estimates, based on this assumption, to estimate light variations works in practice.

7.3 Discussion

Performing our analysis in the Fourier domain around light paths brings us a local characterization of the signal’s bandwidth. Wavelets also bring bandwidth estimation of a signal. However, they perform a local analysis at the cost of a making operations like convolution and scaling much more complex. In our case, localization is already brought by windowing our analysis around a particular point of the signal, and the Fourier transform appears to be the most simple approach to characterize bandwidth. Polyno-

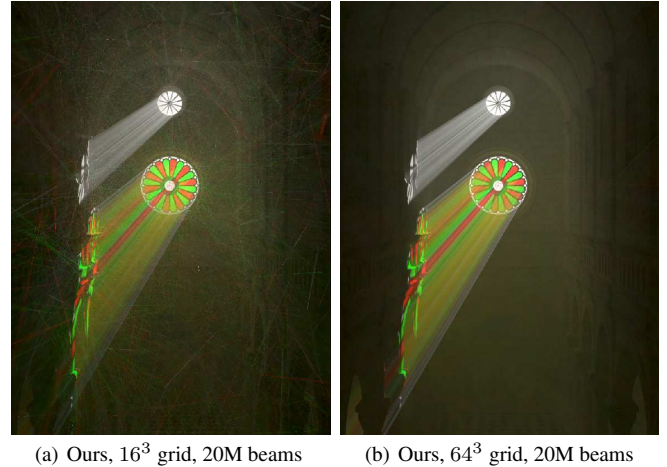


Fig. 21. Influence of the size of the covariance grid over the gain in convergence. A very coarse grid (16^3 in this example), will conservatively spread high frequency values into regions where the radiance is actually low frequency, failing to improve the convergence in these regions. Note: we chose to use an overly coarse grid in this example to make the effect more apparent.

mial bases in turn, don’t offer simple expressions for convolution and scale.

Our theoretical derivations perform first order approximations of the scattering and absorption operators, as opposed to first order approximations of the spectra. Linearly approximating the spectra would be meaningless. In our framework, spectra contain all possible frequencies. The only assumption made is the paraxial approximation [Durand et al. 2005].

As a comparison to the work of Yue et al. [2010], our adaptive sampling strategy is based on the variance of the illumination, whereas Yue’s algorithm is based on the maximum variance of the density in the medium along a light path. Therefore we avoid wasting time oversampling highly varying regions with low energy. While adaptive sampling techniques are usually based on an *a posteriori* estimation of the energy (sometimes the bandwidth) of the signal, we base our sampling on an *a priori* prediction of the variance of the signal.

Kulla et al. [2012] propose strategies for importance sampling participating media. Our approach is complementary since we believe that importance sampling metrics can be derived from the volumetric covariance. Combining our covariance prediction tool with their importance sampling metrics would allow to drive the importance sampling by the actual distribution of energy in the solution.

Keeping a large radius for progressive photon beams could slow down the selection process, if using an acceleration structure (such as a KD-tree [Sun et al. 2010]), since this increases the branching factor of the KD-tree search. In our CUDA implementation, which follows the method described by Jarosz et al., not having an acceleration structure removes this penalty.

When filling the covariance grid, we do not need to record a directional distribution of incident illumination to weight the covariance contributions from incident rays, since those rays are path-traced and arrive with a probability that is already proportional to the illumination. Consequently, even in its current and simple implementation, the covariance grid allows us to perform anisotropic filtering of light beams. This is visible in the caustic scene (Fig-

ure 8) where we can see that the covariance estimates capture the directional discontinuities of the light distribution.

We chose to apply our framework to accelerate progressive photon beams rather than the more recent progressive virtual beam lights [Novák et al. 2012]. The two methods, however, use a similar iterative radius reduction scheme. Therefore, one can expect to improve on the latter the same way we improve on progressive photon beams, using our radius reduction stopping criterion.

Similar to Novak et al. [2012], we could use our method to only reconstruct the multiple scattering component of the illumination and not store the specular contribution into the covariance grid. Although did not do that, we still produce convincing results as our method adapts to any effect (be it direct or indirect). Treating indirect scattering independently would enhance the speedup factor of our method as the reconstructed signal would be of even lower frequency. But this would increase the required engineering for the implementation (multiple photon maps would be required).

Our improved progressive photon beams method removes the need to finely tune the parameters of the radius reduction. This, in a way, is similar to the Adaptive Progressive Photon Mapping of Kaplanyan and Daschbacher [2013]. One notable difference is that our method does not need to evaluate the Hessian of the radiance using density estimation, and as shown in Section 6.2 our estimate is much more robust.

Adding the time analysis [Belcour et al. 2013] is straightforward but currently limited to rigid motion. The analysis of time varying media is also possible, but beyond the scope of this paper. Time analysis could be implemented using the 3D covariance grid by integrating the time dimension. This way, motion events are blurred according to the resulting appearance. A 4D grid would be necessary to perform temporal coherent filtering. Adding depth of field is also orthogonal to this work, but we expect it would not cause particular issues.

8. CONCLUSION

We proposed the first extension to participating media, of the Fourier analysis of local light fields. This is a very important problem that is amenable to performance acceleration, since participating media typically has lower frequencies.

We show how to extend the use of covariance matrices in a principled manner to represent the spectrum of the light field including scattering and absorption. We derive the equations to combine the information carried by each light path into a set of 3D frequency prediction metrics, and to compute them from a common quantity: the volumetric covariance, stored in a grid. We used these metrics to improve the convergence and efficiency of image-space adaptive sampling and reconstruction, camera ray integration, and for accelerating the progressive photon beam approach.

Several future avenues of research exist. While we demonstrated the use of this analysis for the gather part of photon mapping, our frequency estimate could also be used to drive photon tracing, like, for example, using a function of frequency as the acceptance function of Metropolis photon sampling [Fan et al. 2005]. Another interesting research avenue would be to extend our analysis to oriented media [Jakob et al. 2010].

We did not use an adaptive covariance grid. To do that, the local resolution of the covariance grid needs to be adapted to the variations of the covariance information to be stored, and would spare the need to specify an initial resolution. We leave this question to future work.

ACKNOWLEDGMENTS

The authors thank user Mundomupa from blendswap.com for the design of the pumpkin model (used in the Halloween scene) and Jeremy Birn for the design of the house front (used in the Halloween scene).

The authors also thank Romain Pacanowski for this helpful comments during the redaction of this work.

This work was supported in part by a grant ANR-11-BS02-006 "ALTA".

REFERENCES

- BAGHER, M., SOLER, C., SUBR, K., BELCOUR, L., AND HOLZSCHUCH, N. 2012. Interactive rendering of acquired materials on dynamic geometry using bandwidth prediction. In *Proceedings of the 2012 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games - I3D*. 127–134.
- BARAN, I., CHEN, J., RAGAN-KELLEY, J., DURAND, F., AND LEHTINEN, J. 2010. A hierarchical volumetric shadow algorithm for single scattering. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2010)* 29, 6 (Dec.), 178:1–178:10.
- BELCOUR, L. AND SOLER, C. 2011. Frequency based kernel estimation for progressive photon mapping. In *SIGGRAPH Asia 2011 Posters*. ACM, 1.
- BELCOUR, L., SOLER, C., SUBR, K., HOLZSCHUCH, N., AND DURAND, F. 2013. 5D Covariance Tracing for Efficient Defocus and Motion Blur. *ACM Transactions on Graphics* 32, 3 (July), 31:1–31:18.
- CEREZO, E., PEREZ-CAZORLA, F., PUEYO, X., SERON, F., AND SILLION, F. X. 2005. A survey on participating media rendering techniques. *The Visual Computer* 21, 5 (June), 303–328.
- CHEN, J., BARAN, I., DURAND, F., AND JAROSZ, W. 2011. Real-time volumetric shadows using 1D min-max mipmaps. In *Symposium on Interactive 3D Graphics and Games on - I3D 2011*. ACM, 39–46.
- DAVE, J. V. 1970. Coefficients of the Legendre and Fourier series for the scattering functions of spherical particles. *Applied Optics* 9, 8, 1888–1896.
- DAVE, J. V. AND GAZDAG, J. 1970. A modified Fourier transform method for multiple scattering calculations in a plane parallel Mie atmosphere. *Applied Optics* 9, 6, 1457–1466.
- DUDERSTADT, J. AND MARTIN, W. 1979. *Transport Theory*. Wiley-Interscience Publications. Wiley.
- DURAND, F., HOLZSCHUCH, N., SOLER, C., CHAN, E., AND SILLION, F. X. 2005. A frequency analysis of light transport. *ACM Transactions Graphics (Proceedings of ACM SIGGRAPH 2005)* 24, 3 (July), 1115–1126.
- EGAN, K., DURAND, F., AND RAMAMOORTHY, R. 2011. Practical filtering for efficient ray-traced directional occlusion. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2011)* 30, 6 (Dec.), 180:1–180:10.
- EGAN, K., HECHT, F., DURAND, F., AND RAMAMOORTHY, R. 2011. Frequency analysis and sheared filtering for shadow light fields of complex occluders. *ACM Transactions on Graphics* 30, 2 (Apr.), 1–13.
- EGAN, K., TSENG, Y.-T., HOLZSCHUCH, N., DURAND, F., AND RAMAMOORTHY, R. 2009. Frequency analysis and sheared reconstruction for rendering motion blur. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)* 28, 3, 93:1–93:13.
- ENGELHARDT, T. AND DACHSBACHER, C. 2010. Epipolar sampling for shadows and crepuscular rays in participating media with single scattering. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games - I3D*. 119–125.

- ENGELHARDT, T., NOVAK, J., AND DACHSBACHER, C. 2010. Instant Multiple Scattering for Interactive Rendering of Heterogeneous Participating Media. Tech. rep., Karlsruhe Institut of Technology.
- FAN, S., CHENNEY, S., AND CHI LAI, Y. 2005. Metropolis photon sampling with optional user guidance. In *Proceedings of the 16th Eurographics Symposium on Rendering Techniques 2005*. Eurographics Association, 127–138.
- GIRAUD, B. AND PESCHANSKI, R. 2006. On positive functions with positive Fourier transforms. *Acta Physica Polonica B* 37, 2 (February), 331–343.
- GORTLER, S. J., GRZESZCZUK, R., SZELISKI, R., AND COHEN, M. 1986. The Lumigraph. In *Proceedings of ACM SIGGRAPH 1996*. ACM, 43–54.
- GUTIERREZ, D., JENSEN, H. W., JAROSZ, W., AND DONNER, C. 2009. Scattering. In *ACM SIGGRAPH Asia 2009 Courses*. 15:1–15:620.
- IGEHY, H. 1999. Tracing ray differentials. In *Proceedings of ACM SIGGRAPH 1999*. ACM, 179–186.
- IHRKE, I., ZIEGLER, G., TEVS, A., THEOBALT, C., MAGNOR, M., AND SEIDEL, H.-P. 2007. Eikonal rendering: efficient light transport in refractive objects. 26, 3 (July).
- ISHIMARU, A. 1997. *Wave Propagation and Scattering in Random Media*. IEEE Press and Oxford University Press.
- JAKOB, W., ARBREE, A., MOON, J. T., BALA, K., AND MARSCHNER, S. 2010. A radiative transfer framework for rendering materials with anisotropic structure. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2010)* 29, 4 (July), 53:1–53:13.
- JAROSZ, W., DONNER, C., ZWICKER, M., AND JENSEN, H. W. 2008. Radiance caching for participating media. *ACM Transactions on Graphics* 27, 1 (Mar.), 7:1–7:11.
- JAROSZ, W., NOWROUZEZAHRAI, D., SADEGHI, I., AND JENSEN, H. W. 2011. A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Transactions on Graphics* 30, 1 (Feb.), 5:1–5:19.
- JAROSZ, W., NOWROUZEZAHRAI, D., THOMAS, R., SLOAN, P.-P., AND ZWICKER, M. 2011. Progressive photon beams. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2011)* 30, 6 (Dec.), 181:1–181:12.
- JAROSZ, W., ZWICKER, M., AND JENSEN, H. W. 2008. The beam radiance estimate for volumetric photon mapping. *Computer Graphics Forum* 27, 2 (Apr.), 557–566.
- JENSEN, H. W. AND CHRISTENSEN, P. H. 1998. Efficient simulation of light transport in scenes with participating media using photon maps. In *Proceedings of ACM SIGGRAPH 1998*. 311–320.
- KAJIYA, J. T. AND VON HERZEN, B. P. 1984. Ray tracing volume densities. *ACM Computer Graphics (Proceedings of ACM SIGGRAPH 1984)* 18, 3 (July), 165–174.
- KAPLANYAN, A. S. AND DACHSBACHER, C. 2013. Adaptive progressive photon mapping. *ACM Transactions on Graphics* 32, 2 (Apr.), 16:1–16:13.
- KIM, A. D. AND MOSCOSO, M. 2003. Radiative transfer computations for optical beams. *Journal Computational Physics* 185, 1 (Feb.), 50–60.
- KNAUS, C. AND ZWICKER, M. 2011. Progressive photon mapping: A probabilistic approach. *ACM Transactions on Graphics* 30, 3 (May), 25:1–25:13.
- KULLA, C. AND FAJARDO, M. 2012. Importance sampling techniques for path tracing in participating media. *Computer Graphics Forum (Proceedings of EGSR 2012)* 31, 4 (June), 1519–1528.
- LAFORTUNE, E. P. AND WILLEMS, Y. D. 1993. Bi-directional path tracing. In *Proceedings of third international conference on computational graphics and visualization techniques*. 145–153.
- LAFORTUNE, E. P. AND WILLEMS, Y. D. 1996. Rendering participating media with bidirectional path tracing. In *Proceedings of the Eurographics Workshop on Rendering Techniques*. Springer-Verlag, 91–100.
- LEHTINEN, J., AILA, T., CHEN, J., LAINE, S., AND DURAND, F. 2011. Temporal light field reconstruction for rendering distribution effects. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2011)* 30, 4 (July), 55:1–55:12.
- LEVOY, M. AND HANRAHAN, P. 1986. Light field rendering. In *Proceedings of ACM SIGGRAPH 1996*. ACM, 31–42.
- MEHTA, S., WANG, B., AND RAMAMOORTHI, R. 2012. Axis-aligned filtering for interactive sampled soft shadows (proceedings of acm siggraph asia 2012). *ACM Transactions on Graphics* 31, 6 (Nov), 163:1–163:10.
- NOVÁK, J., NOWROUZEZAHRAI, D., DACHSBACHER, C., AND JAROSZ, W. 2012. Progressive virtual beam lights. *Computer Graphics Forum (Proceedings of EGSR 2012)* 31, 4 (June), 1407–1413.
- NOVÁK, J., NOWROUZEZAHRAI, D., DACHSBACHER, C., AND JAROSZ, W. 2012. Virtual ray lights for rendering scenes with participating media (proceedings of acm siggraph 2012). *ACM Transactions on Graphics* 31, 4 (July), 60:1–60:11.
- OVERBECK, R. S., DONNER, C., AND RAMAMOORTHI, R. 2009. Adaptive wavelet rendering. In *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2009)*. Vol. 28. 140:1–140:12.
- PAULY, M., KOLLIG, T., AND KELLER, A. 2000. Metropolis Light Transport for Participating Media. *Proceedings of the Eurographics Workshop on Rendering Techniques*.
- RAMAMOORTHI, R., ANDERSON, J., MEYER, M., AND NOWROUZEZAHRAI, D. 2012. A theory of monte carlo visibility sampling. *ACM Transactions on Graphics* 31, 5 (Sept.), 121:1–121:16.
- RIBARDIÈRE, M., CARRÉ, S., AND BOUATOUCH, K. 2011. Adaptive records for volume irradiance caching. *The Visual Computer* 27, 6-8 (Apr.), 655–664.
- RITCHIE, B., DYKEMA, P. G., AND BRADY, D. 1997. Use of fast-Fourier-transform computational methods in radiation transport. *Physical Review E* 56, 2 (Aug), 2217–2227.
- ROUSSELLE, F., KNAUS, C., AND ZWICKER, M. 2011. Adaptive sampling and reconstruction using greedy error minimization. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2011)* 30, 6 (Dec.), 159:1–159:12.
- RYBICKI, G. B. 1971. The searchlight problem with isotropic scattering. *Journal of Quantitative Spectroscopy and Radiative Transfer* 11, 6.
- SCHWARZHAUPT, J., JENSEN, H. W., AND JAROSZ, W. 2012. Practical hessian-based error control for irradiance caching. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2012)* 31, 6 (Nov.), 193:1–193:10.
- SILVERMAN, B. 1986. *Density Estimation for Statistics and Data Analysis*, 1 ed. Chapman and Hall/CRC.
- SOLER, C., SUBR, K., DURAND, F., HOLZSCHUCH, N., AND SILLION, F. X. 2009. Fourier depth of field. *ACM Transactions on Graphics* 28, 2, 18:1–18:12.
- SUN, X., ZHOU, K., LIN, S., AND GUO, B. 2010. Line space gathering for single scattering in large scenes. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2010)* 29, 4 (July), 54:1–54:8.
- TODHUNTER, I. 1859. *Spherical Trigonometry, for the use of Colleges and Schools*. Macmillan and Company.
- VEACH, E. AND GUIBAS, L. J. 1997. Metropolis light transport. In *Proceedings of ACM SIGGRAPH 1997*. ACM, 65–76.
- WALTER, B., ARBREE, A., BALA, K., AND GREENBERG, D. P. 2006. Multidimensional lightcuts. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2006)* 25, 3 (July), 1081–1088.

WALTER, B., FERNANDEZ, S., ARBREE, A., BALA, K., DONIKIAN, M., AND GREENBERG, D. P. 2005. Lightcuts: a scalable approach to illumination. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2005)* 24, 3 (July), 1098–1107.

WALTER, B., KHUNGURN, P., AND BALA, K. 2012. Bidirectional Lightcuts. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2012)* 31, 4 (July), 59:1–59:11.

WALTER, B., ZHAO, S., HOLZSCHUCH, N., AND BALA, K. 2009. Single scattering in refractive media with triangle mesh boundaries. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)* 28, 3 (July), 92:1–92:8.

WAND, M. AND STRASSER, W. 2003. Multi-resolution point-sample ray-tracing. In *Proceedings of Graphics Interface 2003*.

YUE, Y., IWASAKI, K., CHEN, B.-Y., DOBASHI, Y., AND NISHITA, T. 2010. Unbiased, adaptive stochastic sampling for rendering inhomogeneous participating media. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2010)* 29, 6 (Dec.), 177:1–177:8.

APPENDIX

A. LAPLACIAN FROM COVARIANCE OF SPECTRUM

Let f be a function defined over a 4D domain. Following basic Fourier theory, the DC of the second derivative is the integral of its spectrum:

$$\forall i, j \quad \frac{\partial^2 f}{\partial x_i \partial x_j}(0) = \int \widehat{\frac{\partial^2 f}{\partial x_i \partial x_j}}(\Omega) d\Omega$$

...where the integral is carried over the entire 4D Fourier domain. We expand the Fourier transform of the derivative over each variable:

$$\begin{aligned} \forall i, j \quad \frac{\partial^2 f}{\partial x_i \partial x_j}(0) &= \int (2\pi i \Omega_i)(2\pi i \Omega_j) \widehat{f}(\Omega) d\Omega \\ &= -4\pi^2 \int \Omega_i \Omega_j \widehat{f}(\Omega) d\Omega \end{aligned}$$

This formulation almost fits the covariance of the power spectrum. It actually does when $\widehat{f}(\Omega)$ is real and positive, and we miss the normalization factor (which is $\int \widehat{f} = f(0) = 1$). There exist a wide class of such functions [Giraud and Peschanski 2006], the simplest of which are Gaussians. In this case we have:

$$-4\pi^2 f(0) \Sigma_{ij}(\mathbf{x}_0) = \frac{\partial^2 f}{\partial x_i \partial x_j}$$

In practice, that means that we can approximate the covariance of the power spectrum of a function in a small window around a point, as soon as the function is close to the osculating Gaussian at that point. For 4D functions, the covariance matrix of the windowed power spectrum of the function around a particular point (corresponding to $\mathbf{x} \rightarrow f(\mathbf{x}_0 + \mathbf{x})$) is therefore well approximated by the following diagonal matrix in the frame of the principal curvatures:

$$\Sigma(\mathbf{x}_0) \approx \frac{1}{4\pi^2 |f(0)|} \left\| \left\| \frac{\partial^2 f}{\partial x_i^2}(\mathbf{x}_0) \right\| \right\|_{ii}$$

In the most general case, the Hessian matrix must be diagonalised before taking the absolute values on the diagonal. Similarly, we

have:

$$\sum_i \left| \frac{\partial^2 f}{\partial x_i^2} \right| \approx 4\pi^2 f(\mathbf{x}_0) \text{Tr}(\Sigma(\mathbf{x}_0)) \quad (21)$$

B. LOCAL FOURIER COVARIANCE OF H-G FUNCTION

The Henyey-Greenstein phase function is defined by

$$\rho_g(\omega_i, \omega_s) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2gc)^{\frac{3}{2}}} \quad \text{with } c = \omega_i \cdot \omega_s = \cos \alpha$$

We are interested in the 2D covariance of the windowed spectrum of function $\bar{\rho}$, defined by

$$\bar{\rho}(\delta\theta, \delta\phi) = \rho_g(\cos(\alpha + \delta\theta) \cos(\delta\phi))$$

We recall that the 2D covariance matrix of a phase function ρ , noted σ_ρ , is defined by Equation 13:

$$\sigma_\rho = \frac{1}{4\pi^2 \bar{\rho}(0, 0)} \begin{bmatrix} \left| \frac{\partial^2 \bar{\rho}}{\partial \theta^2}(0, 0) \right| & 0 \\ 0 & \left| \frac{\partial^2 \bar{\rho}}{\partial \phi^2}(0, 0) \right| \end{bmatrix}$$

Taking the second derivatives of $\bar{\rho}$ and evaluating them at $\delta\theta = \delta\phi = 0$ gives:

$$\begin{aligned} \frac{\partial^2 \bar{\rho}}{\partial \delta \theta^2}(0, 0) &= \frac{3g(g^2 - 1)(2(g^2 + 1) \cos \alpha + g(3 \cos(2\alpha) - 7))}{8\pi(g^2 - 2g \cos \alpha + 1)^{7/2}} \\ \frac{\partial^2 \bar{\rho}}{\partial \delta \theta \partial \delta \phi}(0, 0) &= 0 \\ \frac{\partial^2 \bar{\rho}}{\partial \delta \phi^2}(0, 0) &= \frac{3g(g^2 - 1) \cos \alpha}{4\pi(g^2 - 2g \cos \alpha + 1)^{5/2}} \end{aligned}$$

The Hessian being diagonal, we don't need additional rotations before taking the absolute value. When $\alpha = 0$, the second derivatives match (to $3g(g+1)/(4\pi(g-1)^4$), as it is expected for a rotationally symmetric function. To get the covariance in equation 14, we further divide by $4\pi^2 \bar{\rho}(0, 0) = 4\pi^2 \rho_g(\alpha)$.