Second-Order Occlusion-Aware Volumetric Radiance Caching

JULIO MARCO, Universidad de Zaragoza, I3A
ADRIAN JARABO, Universidad de Zaragoza, I3A
WOJCIECH JAROSZ, Dartmouth College
DIEGO GUTIERREZ, Universidad de Zaragoza, I3A

We present a second-order gradient analysis of light transport in participating media and use this to develop an improved radiance caching algorithm for volumetric light transport. We adaptively sample and interpolate radiance from sparse points in the medium using a second-order Hessian-based error metric to determine when interpolation is appropriate. We derive our metric from each point's incoming light field, computed by using a proxy triangulation-based representation of the radiance reflected by the surrounding medium and geometry. We use this representation to efficiently compute the first- and second-order derivatives of the radiance at the cache points while accounting for occlusion changes. We also propose a self-contained two-dimensional model for light transport in media and use it to validate and analyze our approach, demonstrating that our method outperforms previous radiance caching algorithms both in terms of accurate derivative estimates and final radiance extrapolation. We generalize these findings to practical three-dimensional scenarios, where we show improved results while reducing computation time by up to 30% compared to previous work.

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

Additional Key Words and Phrases: global illumination, rendering, irradiance caching, participating media, radiance derivatives

ACM Reference format:
Julio Marco, Adrian Jarabo, Wojciech Jarosz, and Diego Gutierrez. 2018. Second-Order Occlusion-Aware Volumetric Radiance Caching. *ACM Trans. Graph.* 37, 2, Article 20 (February 2018), 14 pages.
https://doi.org/10.1145/3185225

1 INTRODUCTION

Accurately simulating the complex lighting effects produced by participating media in the presence of arbitrary geometry remains a challenging task. Monte Carlo-based methods like path tracing numerically approximate the radiative transfer equation (RTE) [Chandrasekhar 1960] by stochastically sampling radiance in the medium. These approaches can handle complex geometry and general scattering properties, but since they lack memory and are largely blind to the radiance signal, they perform many redundant computations leading to high cost. A common strategy to increase efficiency is to adaptively sample radiance based on its frequency content, limiting the sampling density in regions where radiance barely changes, and placing more samples in regions with higher frequency variation [Zwicker et al. 2015].

Based on this principle, *volumetric radiance caching* [Jarosz et al. 2008] computes and stores radiance at sparse cache points in the medium, and uses these samples to reconstruct radiance at nearby locations whenever possible. The method is based on first-order translational derivatives of the radiance, which are used to i) determine how far away a cache point can be reused while controlling error, and ii) improve reconstruction quality by extrapolating the cached radiance values along their gradients. Unfortunately, since the gradient derivations ignore occlusion/visibility changes, the method fails in scenes containing occluders where changes in visibility are the dominant factor in local radiance behavior. Moreover, the reconstruction and error metric both rely on the same gradient estimates and ignore variations caused by higher-order derivatives. These factors lead to suboptimal cache point distributions, which fail to properly sample high-frequency features such as occlusions, while simultaneously oversampling other regions of the scene. This results in reduced efficiency and visible rendering artifacts.

Second-order illumination derivatives have proven to be a powerful and principled tool for sparsely sampling and interpolating surface irradiance [Jarosz et al. 2012; Schwarzhaupt et al. 2012], as well as controlling error in density estimation techniques [Belcour et al. 2014; Hachisuka et al. 2010; Kaplanyan and Dachsbacher 2013]. Inspired by these recent developments, we propose a new second-order, *occlusion-aware* radiance caching method for participating media which overcomes the limitations of current state-of-the-art methods.

To this end, we introduce a novel approach to compute first- and second-order occlusion-aware derivatives of both single and multiple scattering, and generalize the Hessian-based metric of Schwarzhaupt et al. [2012] for controlling the error introduced by first-order extrapolation of media radiance. In addition, we extend recent work on 2D radiometry, currently limited to surfaces [Jarosz et al. 2012], and derive a 2D theory of light transport in participating media. We use this framework to illustrate and analyze the limitations of the state of the art, as well as the benefits of our proposed method. We demonstrate the generality of our approach by deriving occlusion-aware derivatives of 3D media radiance and applying our Hessian-based metric to 3D cache distributions, showing that the benefits predicted by our 2D analysis hold equally in 3D. Our approach improves volumetric cache point distributions in isotropic homogeneous media, providing a significantly more accurate reconstruction of difficult high-frequency features, as Figure 1 shows.

2 RELATED WORK

We summarize here existing work on radiance caching methods as well as other techniques that leverage illumination derivatives to improve Monte Carlo rendering. For a general overview of scattering and existing adaptive sampling and reconstruction techniques, we refer the reader to other recent sources of information [Gutierrez et al. 2008; Zwicker et al. 2015].

Radiance caching: Irradiance caching was originally proposed by Ward et al. [1988] to accelerate indirect illumination in Lambertian scenes. The method computes and caches indirect irradiance

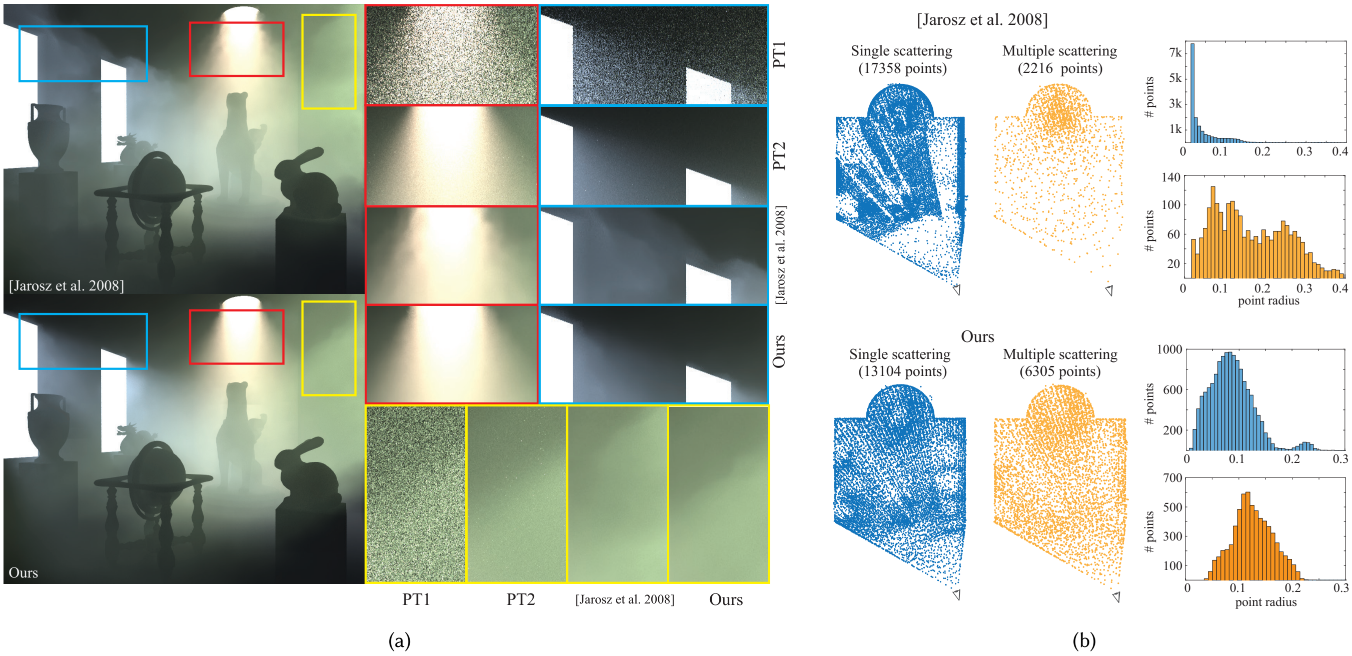

Fig. 1. *Statues* scene rendered with both single and multiple scattering. Radiance at surfaces is excluded for illustration purposes (please refer to the digital version for accurate visualization). (a) **PT1:** Path tracing, 2k samples/pixel, 2h. **PT2:** Path tracing, 500k samples/pixel, 500h. **[Jarosz et al. 2008]:** Occlusion-unaware, gradient-based error metric, ~19k cache points, 16k samples/cache, 155 minutes. **Ours:** Occlusion-aware, Hessian-based metric, ~19k cache points, 16k samples/cache, 154 minutes. (b) Cached point distribution as seen from above for both single and multiple scattering. Ignoring visibility derivatives fails at representing high-frequency shadows from the windows (a, blue and yellow) due to poor cache distribution, as well as other rapid radiance changes (a, red) in areas with good cache distribution, due to imprecise extrapolation during reconstruction. In contrast, our occlusion-aware Hessian-based method correctly handles these higher-frequency features by improving the sample distribution, as well as the reconstruction. The occlusion-unaware approach (b, top-left) concentrates the samples excessively near the surfaces, usually reaching the cache minimum radius (see top-right histograms), but ignoring occlusion changes throughout the scene. Using occlusion-aware first- and second-order derivatives, our method predicts the error introduced by gradient extrapolation more robustly, increasing cache density in regions where gradients change rapidly (b, bottom).

only at a sparse set of points in the scene, and extrapolates or interpolates these values whenever possible from cache points deemed to be sufficiently close by. Since indirect illumination changes slowly across Lambertian surfaces, the costly irradiance calculation can often be reused over large parts of the image, substantially accelerating rendering. There has been a wealth of improvements to irradiance caching, but we discuss only the most relevant follow-up work and refer to Křivánek and Gautron [2009] for a more complete survey.

Ward and Heckbert [1992] significantly improved reconstruction by leveraging gradient information, and Křivánek et al. [2006] incorporated heuristics to improve error estimation (and therefore quality) during adaptive caching. Křivánek and colleagues [2005; 2005] also extended irradiance caching to handle moderately glossy, non-Lambertian surfaces. Herzog et al. [2009] used anisotropic cache points based on the orientation of the illumination gradient. All these methods only considered *surface* light transport.

Jarosz et al. [2008] proposed *volumetric* radiance caching, which accelerates single and multiple scattering in participating media. They proposed an error metric based on the first-order derivative of the radiance, but their formulation ignored volumetric occlusion changes. In follow-up work, Jarosz et al. [2008] derived occlusion-aware gradients, but only of surface illumination in the presence of absorbing and scattering media, ignoring gradients of the media radiance itself. Both approaches are prone to suboptimal cache point distributions and visible artifacts since they ignore higher order derivatives or occlusion changes in media. Our work addresses both of these issues. Ribardière et al. [2011] proposed using anisotropic cache points and a second-order expansion for radiance reconstruction. Their approach, however, did not consider visibility changes due to their point-to-point computation of derivatives.

Recently, Jarosz et al. [2012] and follow-up work [Schwarzhaupt et al. 2012] made significant progress in heuristics-free error control for surface irradiance caching by formulating error in terms of second-order derivatives. In particular, Schwarzhaupt et al. [2012] proposed a novel radiometrically equivalent formulation of irradiance gradients and Hessians, which properly accounted for occlusions. The authors used these for extrapolation and principled error control, respectively. We extend these ideas and apply them to light transport in participating media, deriving first- and second-order occlusion-aware derivatives for improved reconstruction and principled error control in volumetric radiance caching.

Table 1. Notation for the optical properties of participating media, and their differences in 3D and 2D.

		3D		2D	
Quantity	Symbol	Expression	Units	Expression	Units
Particle density	ρ	Particles per unit volume	m^{-3}	Particles per unit area	m^{-2}
Cross-section	σ	Area	m^{2}	Length	m
Scattering coefficient	μ_s	Probability density per differential length	m^{-1}	Probability density per differential length	m^{-1}
Absorption coefficient	μ_a	Probability density per differential length	m^{-1}	Probability density per differential length	m^{-1}
Extinction coefficient	μ_t	$\mu_t = \mu_a + \mu_s$	m^{-1}	$\mu_t = \mu_a + \mu_s$	m^{-1}
Transmittance	T_r	$T_r(\mathbf{x}_1, \mathbf{x}_2) = \exp(-\int_{\mathbf{x}_1}^{\mathbf{x}_2} \mu_t(\mathbf{x})\mathrm{d}\mathbf{x})$	unitless	$T_r(\mathbf{x}_1, \mathbf{x}_2) = \exp(-\int_{\mathbf{x}_1}^{\mathbf{x}_2} \mu_t(\mathbf{x})\mathrm{d}\mathbf{x})$	unitless
Phase Function	$f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)$	Angular scattering of light at a point	sr^{-1}	Angular scattering of light at a point	rad^{-1}

Differential domain: Arvo [1994] derived closed form expressions for irradiance derivatives in polygonal environments, and Holzschuch and Sillion [1998, 1995] derived second-order illumination derivatives for error control in the radiosity algorithm. Local differentials have also proven useful for texture filtering [Igehy 1999; Suykens and Willems 2001], photon density estimation [Jarosz et al. 2011a; Schjøth et al. 2007], and spectral rendering [Elek et al. 2014]. Ramamoorthi et al. [2007] analyzed gradients of various surface lighting effects, including occlusions, and showed how these can be used for adaptive sampling and interpolation in image space. Lehtinen et al. [2013] and follow-up work [Manzi et al. 2014], proposed to compute image gradients instead of actual luminance values in Metropolis light transport (MLT), and feed a Poisson solver with these gradients to reconstruct the final image. Later work [Kettunen et al. 2015; Manzi et al. 2015] extended the applicability of this gradient domain idea to simpler Monte Carlo path tracing methods, and demonstrated how solving light transport in the gradient domain improves over primal space, while remaining unbiased. Rousselle et al. [2016] showed how such Poisson-based reconstruction approaches can be directly formulated as control-variate estimators. Kaplanyan and Dachsbacher [2013] leveraged second-order derivatives of irradiance to estimate optimal kernel bandwidth in progressive photon mapping, focusing on surface light transport only.

Closely related to our work, Belcour et al. [2014] performed a frequency analysis of light fields within participating media. They summarize the local light field using covariance matrices, which provides Hessians of fluence (up to sign) due to scattering and absorption. Their approach explicitly accounts for radiance changes only in the plane perpendicular to ray propagation, needing to average the per-light-path information from many rays to compute the 3D fluence spectrum. To account for visibility changes, they also require precomputing the covariance matrices in a finite neighborhood, sacrificing locality and incurring the cost of scene voxelization. In contrast, we provide a fully local method for computing first- and second-order derivatives of media radiance, without requiring voxelization, all while accounting for changes due to visibility, scattering, and transmittance.

2D spaces: Simplification to lower-dimensional spaces is a recurring tool used in problem analysis. In image synthesis, reduction to hypothetical 2D worlds has been used to obtain insights and illustrate the benefits of more complex 3D approaches [Heckbert 1992; Orti et al. 1996]. More recent analyses of derivative and frequency domains [Durand et al. 2005; Mehta et al. 2013; Ramamoorthi et al. 2007], as well as recent work on complex reflectance filtering [Yan et al. 2014, 2016] reduce the complexity of their derivations by performing them in 2D, before showing how the gained insights generalize to 3D. Jarosz et al. [2012] introduced a 2D surface radiometry and global illumination framework, and showed how this allows for a more practical analysis of 2D versions of standard rendering algorithms due to faster computation and simpler visualization. Other fields such as acoustic rendering have recently benefited from 2D reduction to provide interactive simulations [Allen and Raghuvanshi 2015]. Two-dimensional simulations have also been proved useful to synthesize higher-dimensional light transport, as in transient rendering [Bitterli 2016b; Jarabo et al. 2014]. In this paper we follow a similar methodology as Jarosz et al. [2012], providing a novel 2D radiometry framework for participating media.

3 2D AND 3D LIGHT TRANSPORT IN PARTICIPATING MEDIA

We describe here the main radiometric aspects of working in a two-dimensional domain, compared to 3D. Similar to Jarosz et al. [2012], we assume an *intrinsic model* where light is generated, scattered, and absorbed within a plane, thus ensuring energy conservation.

The outgoing radiance at a point $\mathbf{x}$ in a medium is defined as the angular integral of the incident radiance $L_i(\mathbf{x}, \vec{\omega}_i)$, modulated by the scattering phase function $f_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)$:

$$L(\mathbf{x}, \vec{\omega}_o) = \int_\Omega f_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)\, L_i(\mathbf{x}, \vec{\omega}_i)\, \mathrm{d}\vec{\omega}_i, \tag{1}$$

where $\vec{\omega}_i$ and $\vec{\omega}_o$ are directions over the spherical domain Ω pointing into and out of the point $\mathbf{x}$ respectively. The incident radiance $L_i = L_m + L_s$ is the sum of radiance arriving from the surrounding medium (L_m) and from surfaces (L_s):

$$L_m(\mathbf{x}, \vec{\omega}_i) = \int_0^s \mu_s(\mathbf{y}(t))\, T_r(\mathbf{x}, \mathbf{y}(t))\, L(\mathbf{y}(t), \vec{\omega}_i)\, \mathrm{d}t, \tag{2}$$

$$L_s(\mathbf{x}, \vec{\omega}_i) = T_r(\mathbf{x}, \mathbf{y}_s)\, L_o(\mathbf{y}_s, \vec{\omega}_i), \tag{3}$$

where $\mathbf{y}(t) = \mathbf{x} - t\vec{\omega}_i$ is a point in the medium, and $\mathbf{y}_s$ is a point on a surface at distance s with outgoing radiance L_o modeled by the rendering equation [Kajiya 1986]. The transmittance T_r models the attenuation due to scattering and absorption between two points, and $\mu_s(\mathbf{x}) = \rho\sigma_s$ is the scattering coefficient at $\mathbf{x}$, with ρ and σ_s the density and scattering cross-section in the medium, respectively. We detail our notation in Table 1, and highlight the main radiometric

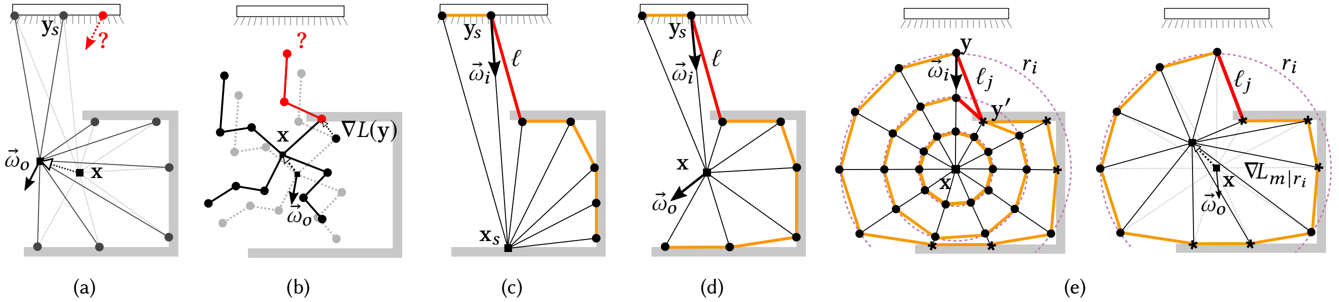

Fig. 2. Jarosz et al.'s [2008] point-to-point approach for computing first-order derivatives of single (a) and multiple (b) scattering ignores radiance that becomes occluded/disoccluded (red) as $\mathbf{x}$ is translated. Schwarzhaupt et al. [2012] compute occlusion-aware derivatives (c) of diffuse surface irradiance by considering the occlusion-free subdivision (orange) of surrounding geometry as seen from $\mathbf{x}_s$. We compute occlusion-aware first- and second-order derivatives (d,e) by constructing such occlusion-free subdivisions (orange) of the scene, both at surface locations for single scattering (similar to Schwarzhaupt's work), and also at ray-marched media locations for multiple scattering. Red segments represent approximations of both single and multiple scattering occlusions. Starred points $\star$ in (e) represent black samples at surfaces that occlude radiance from media.

differences between self-contained 2D and 3D worlds, described below.

Differences in 2D: When moving to a 2D world, the intrinsic radiometric model implies that all radiance travels within a planar medium, scattering therefore over angle instead of solid angle. This means that radiance falls off with the inverse distance instead of inverse squared distance [Jarosz et al. 2012]; this will become important in our analysis of first- and second-order derivatives.

The main changes when applying Equations (1–3) in 2D are:

- The integration domain Ω of Equation (3) becomes circular instead of spherical.
- The phase functions in 2D must be normalized over the circle, not the sphere, of incident directions.
- $L_o(\mathbf{y}_s)$ now indicates radiance from the closest curve (the 2D equivalent of a 3D surface).

In the next sections, we use this self-contained 2D world to better depict and reason about the improvements of our new occlusion-aware gradients and Hessians for media (Section 4), and our second-order error metric (Section 5), before extending them to a more practical three-dimensional world. Working in 2D also allows us to avoid collapsing a 3D scene into a 2D image for visualization, where information from many media points would contribute to a single image pixel. This allows us to illustrate the performance of our algorithm in a more intuitive way (Section 6) and to depict the introduced errors more clearly.

3.1 Radiance Caching in Participating Media

Before deriving our second-order, occlusion-aware volumetric radiance caching approach, we first summarize Jarosz et al.'s [2008] original formulation. To determine the radiance at any point $\mathbf{x}'$ in the medium[1], their algorithm first tries to approximate this value by extrapolating (in the log domain) the cached radiance L_k from nearby cache point locations $\mathbf{x}_k$ along their respective gradients:

$$L(\mathbf{x}', \vec{\omega}_o) \approx \exp\left[\frac{\sum_{k\in C}(\ln L_k + \nabla \ln L_k \cdot \Delta_{\mathbf{x}'})\,\mathrm{w}(\mathbf{x}_k, \mathbf{x}')}{\sum_{k\in C}\mathrm{w}(\mathbf{x}_k, \mathbf{x}')}\right], \quad (4)$$

with $\Delta_{\mathbf{x}'} = (\mathbf{x}' - \mathbf{x}_k)$. Here $\nabla \ln L_k = \nabla L_k / L_k$ is the log-space translational gradient of cache point $\mathbf{x}_k$, and $\mathrm{w}(\mathbf{x}_k, \mathbf{x}')$ is a weighting function that diminishes the influence of a cache point to zero as $\mathbf{x}'$ approaches the cache point's valid radius. The collection of nearby cache points C consists of all cache points whose valid radii contain $\mathbf{x}'$. If no nearby cache points are found, then the algorithm computes radiance using Monte Carlo sampling and inserts the value and its gradient into the cache for future reuse.

Jarosz et al. [2008] proposed to compute the valid radii using a metric based on the local log-space radiance gradient:

$$R = \varepsilon \frac{\sum L_j}{\sum \|\nabla L_j\|}, \quad (5)$$

where ε is a global error tolerance parameter and L_j and ∇L_j are the individual Monte Carlo samples of radiance and translational gradient respectively. Unfortunately, this error metric is an ad-hoc approximation of the error in the log-scale interpolation, which can lead to difficulty predicting the error in the sample distribution and suboptimal cache distributions.

Jarosz et al. maintain a separate cache for single/surface scattering and multiple scattering. They compute single-scattering gradients by Monte Carlo sampling the first translational derivative of Equations (1) and (3) in surface-area form. They trace out many rays in the sphere of directions around point $\mathbf{x}$ to obtain a number of surface hit points $\mathbf{y}_s$. Their gradient calculation, in essence, considers how the radiance L_s from each of these hit points would change (due to changes of transmittance and geometry terms, but not visibility) as $\mathbf{x}$ translates, but the surface hit points $\mathbf{y}_s$ remain fixed (see Figure 2a). For multiple-scattering gradients, they Monte Carlo sample the first derivative of Equations (1) and (2), where the whole set of sampled paths is assumed to move rigidly (see Figure 2b), accounting for translational derivatives at each scattering vertex.

This gradient formulation can efficiently compute the local change in radiance of any single Monte Carlo sample, but—by operating

[1] Throughout the text, $\mathbf{x}'$ represents points where we approximate radiance by interpolating the cache points, while $\mathbf{x}$ represents points where we compute radiance and its derivatives explicitly.

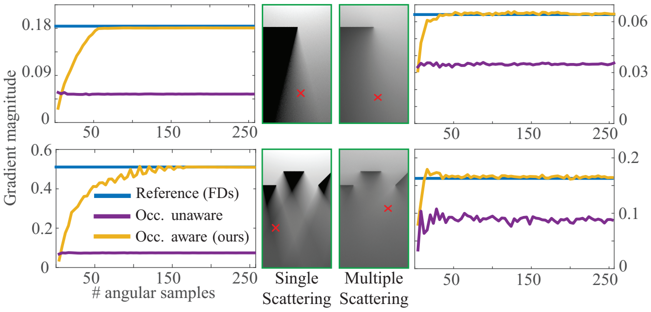

Fig. 3. Compared to prior occlusion-unaware gradients (purple), our gradients (yellow) converge to the reference solution (blue) with increasing angular sample count both for single scattering (left) and multiple scattering (right). The convergence plots are computed in the red crosses in the respective middle images.

independently on each radiance sample—it is not able to capture *global* effects such as visibility gradients. As a consequence, changes in radiance that becomes occluded/unoccluded as the shaded point is translated are not taken into account (see Figures 2a and 2b, red). As an illustrative example, Figure 3 shows how ignoring occlusions (purple line) leads to incorrect single- and multiple-scattering gradients in the penumbra region beneath the occluder.

In the remainder of this paper we describe our novel Hessian-based radiance caching method for participating media that overcomes the aforementioned limitations. In Section 4 we introduce our approach for computing occlusion-aware first- and second-order derivatives of media radiance. Then, in Section 5 we introduce our Hessian-based error metric and extrapolation method for volumetric radiance caching.

4 RADIOMETRIC DERIVATIVES IN MEDIA

Following the work of Schwarzhaupt et al. [2012] on global illumination on surfaces, we formulate the radiance at $\mathbf{x}$ as a piecewise linear representation of the incoming radiance. Conceptually, we build an approximated coarse representation of the scene as seen from the media point $\mathbf{x}$ by triangulating adjacent stochastic angular samples $\mathbf{y}_s$ (see Figures 2c and 2d). The interesting property of this triangulation is that the geometry term for each triangle (segments in 2D) models the attenuation due to the solid angle; as a consequence, changes in the geometry term (due to translation of $\mathbf{x}$) model changes in the observed radiance.

We extend Schwarzhaupt et al.'s [2012] formulation to handle not only light transport from surfaces, but also from media. In the case of surfaces, the sample points $\mathbf{y}_s$ are located at the first surface point as seen from $\mathbf{x}$ in direction $\overrightarrow{\mathbf{yx}}$ (Figure 2d). For points in a participating medium, however, radiance arrives from multiple distances along each direction. We therefore consider a set of concentric triangulations at increasing distances r_i, each representing the outgoing radiance at that particular distance in the medium. If occluding geometry exists closer than the distance r_i, we place a zero-radiance sample at the surface intersection (points marked with ⋆ in Figure 2e).

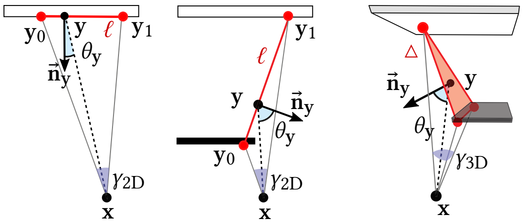

Fig. 4. Left and center: Visible and occluded cases for 2D surface-media radiance for an angle γ. Red segment represent the piecewise-linear construction as seen from $\mathbf{x}$. Right: 3D interpretation, where occlusions are represented by slanted triangles, and visibility changes are modeled as changes in the 3D geometry term between the triangle points $\mathbf{y} \in \triangle$, and $\mathbf{x}$.

Handling Occlusions and Transmittance: In essence, we are approximating the integration along Ω, by transforming the scene into a discrete set of *virtual* piecewise linear representations of the geometry and media around $\mathbf{x}$. As noted by Schwarzhaupt et al., this representation implicitly encodes changes in visibility by means of the geometry term. Our approach for media, however, requires taking transmittance into account and using different geometry terms (see Figure 2c), since surface-medium light transport only has a cosine term at the source $\mathbf{y}_s$. We illustrate this with a 2D example in Figure 4, left and center: Assuming a constant angle γ between vectors $\overrightarrow{\mathbf{xy}_0}$ and $\overrightarrow{\mathbf{xy}_1}$, occlusions generate segments $\ell = \mathbf{y}_1 - \mathbf{y}_0$ at grazing angles, with derivatives proportional to the steepness of the segment. When moving within the medium, the projected angle of ℓ towards $\mathbf{x}$ is proportional to $\cos\theta_\mathbf{y}$, and therefore the radiance from ℓ increases with $\cos\theta_\mathbf{y}$. This allows modeling the visibility changes as a change on the 2D geometry term $G = \cos\theta_\mathbf{y}/\|\overrightarrow{\mathbf{xy}}\|$. This principle holds also for 3D, as Figure 4, right, shows: Occlusions are represented by slanted triangular faces, and visibility changes are modeled as changes in the 3D geometry term between the triangle points $\mathbf{y} \in \triangle$ and $\mathbf{x}$. We leverage this equivalence to provide a unified formulation for radiance derivatives, applicable both to 2D and 3D[2].

Using the formulation presented before, we approximate $L(\mathbf{x}, \vec{\omega}_o)$ by discretizing the space into a set of concentric rings $\mathcal{R}$ as:

$$L(\mathbf{x}, \vec{\omega}_o) \approx \sum_{r_i \in \mathcal{R}} \frac{1}{\mathrm{pdf}(r_i)} \sum_{\ell_j \in \mathcal{L}_i} L_j(\mathbf{x}, \vec{\omega}_o), \tag{6}$$

where the last ring $r_s \in \mathcal{R}$ has all its vertices on surfaces, $\mathcal{L}_i$ is the set of segments for ring r_i, and $\mathrm{pdf}(r_i)$ is the probability of sampling a particular distance when building the ring (for the surface ring, we have $\mathrm{pdf}(r_s) = 1$). L_j is the radiance contributed by each segment $\ell_j \in \mathcal{L}_i$, defined by the integral:

$$L_j(\mathbf{x}, \vec{\omega}_o) = \int_{\ell_j} f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)\, G(\mathbf{x}, \mathbf{y})\, T_r(\mathbf{x}, \mathbf{y})\, L(\mathbf{y}, \vec{\omega}_i)\, \mathrm{d}\mathbf{y}. \tag{7}$$

By construction, the visibility between $\mathbf{x}$ and $\mathbf{y}$ is $V(\mathbf{x}, \mathbf{y}) = 1$, and $\mathbf{y}$ is a point on a *virtual* surface; we thus need to account for the foreshortening at $\mathbf{y}$. This allows for a unified formulation of both surface-to-medium and medium-to-medium radiance derivatives, using the same geometry term in both cases. Note that we have

[2] For convenience, we formulate all the equations in terms of 2D media and geometry subdivisions in segments ℓ, but all formulae are equally applicable in 3D by substituting segments ℓ by triangles $\triangle$.

merged together the phase function $f_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)$ and scattering coefficient $\mu_s(\mathbf{x})$ as a directional scattering function $f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = \mu_s(\mathbf{x}) f_s(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o)$, to make the following derivations simpler.

Differentiating Equation (6) with respect to $\mathbf{x}$ provides approximations for the first and second order derivatives:

$$\nabla L(\mathbf{x}, \vec{\omega}_o) \approx \sum_{r_i \in \mathcal{R}} \sum_{\ell_j \in \mathcal{L}_i} \frac{\nabla L_j(\mathbf{x}, \vec{\omega}_o)}{\mathrm{pdf}(r_i)}, \tag{8}$$

$$\mathbf{H}L(\mathbf{x}, \vec{\omega}_o) \approx \sum_{r_i \in \mathcal{R}} \sum_{\ell_j \in \mathcal{L}_i} \frac{\mathbf{H}L_j(\mathbf{x}, \vec{\omega}_o)}{\mathrm{pdf}(r_i)}, \tag{9}$$

which in turn require differentiating the radiance from each segment.

Unfortunately, we cannot compute Equation (7) and its derivatives analytically in closed-form, while computing it numerically would be prohibitively expensive. We instead introduce a set of assumptions to build a closed-form approximation:

- For a sufficiently fine subdivision the angle γ tends to 0, so $\vec{\omega}_i$ can be regarded as constant for the whole segment, and $f(\mathbf{x}, \vec{\omega}_i, \vec{\omega}_o) = f(\mathbf{x}, \vec{\omega}_\ell, \vec{\omega}_o)$, with $\vec{\omega}_\ell$ a fixed direction from $\mathbf{x}$ to a point in segment ℓ.
- For all $\mathbf{y} \in \ell$, we assume constant $T_r(\mathbf{x}, \mathbf{y}) = T_r(\mathbf{x}, \mathbf{y}_\ell)$, and $L(\mathbf{y}, \vec{\omega}_i) = L(\mathbf{y}_\ell, \vec{\omega}_i)$. Following existing approaches for surface irradiance, we choose $\mathbf{y}_\ell$ as the furthest point in the segment ℓ, which will be the first to be occluded/unoccluded.

These assumptions allow us to significantly simplify the integral in Equation (7) to:

$$\begin{aligned} L_j(\mathbf{x}, \vec{\omega}_o) &\approx f(\mathbf{x}, \vec{\omega}_{\ell_j}, \vec{\omega}_o)\, T_r(\mathbf{x}, \mathbf{y}_{\ell_j})\, L(\mathbf{y}_{\ell_j}, \vec{\omega}_i) \int_{\ell_j} G(\mathbf{x}, \mathbf{y})\, \mathrm{d}\mathbf{y} \\ &= f(\mathbf{x}, \vec{\omega}_{\ell_j}, \vec{\omega}_o)\, T_r(\mathbf{x}, \mathbf{y}_{\ell_j})\, L(\mathbf{y}_{\ell_j}, \vec{\omega}_i)\, F_{\ell_j}(\mathbf{x}), \end{aligned} \tag{10}$$

which now admits a closed-form solution in both 2D and 3D (see Appendices B and C). More importantly, this allows us to approximate the derivatives of L_j in closed form as:

$$\nabla L_j \approx L_F \nabla f + \nabla L_F f, \tag{11}$$

$$\mathbf{H}L_j \approx L_F \mathbf{H}f + \nabla L_F \nabla^{\mathsf{T}} f + \nabla f \nabla^{\mathsf{T}} L_F + \mathbf{H}L_F f, \tag{12}$$

where

$$\nabla L_F = L_r \nabla F_\ell + \nabla L_r F_\ell, \tag{13}$$

$$\mathbf{H}L_F = L_r \mathbf{H}F_\ell + \nabla L_r \nabla^{\mathsf{T}} F_\ell + \nabla F_\ell \nabla^{\mathsf{T}} L_r + \mathbf{H}L_r F_\ell, \tag{14}$$

$$\nabla L_r = L \nabla T_r + \nabla L T_r, \tag{15}$$

$$\mathbf{H}L_r = L\mathbf{H}T_r + \nabla L \nabla^{\mathsf{T}} T_r + \nabla T_r \nabla^{\mathsf{T}} L + \mathbf{H}L T_r. \tag{16}$$

For brevity we have omitted function parameters, and we express gradients and Hessians in terms of the scaled radiance $L_F = F_\ell L_r$, and the reduced radiance $L_r = LT_r$. While Equations (10–16) are general, we restrict our work to Lambertian surfaces and isotropic, homogeneous media (in Section 7 we discuss how to extend it to anisotropic and heterogeneous media). This means that both L and f are constant, and therefore their derivatives cancel out as $\nabla L = \mathbf{H}L = \nabla f = \mathbf{H}f = 0$, removing directional dependences; this allows us to simplify Equations (11) and (12) to:

$$\nabla L_j \approx Lf \left(T_r \nabla F_\ell + \nabla T_r F_\ell \right), \tag{17}$$

$$\mathbf{H}L_j \approx Lf \left(T_r \mathbf{H}F_\ell + \nabla T_r \nabla^{\mathsf{T}} F_\ell + \nabla F_\ell \nabla^{\mathsf{T}} T_r + \mathbf{H}T_r F_\ell \right). \tag{18}$$

We refer to Appendices A, B and C for all the terms.

By construction, our formulation in Equation (6) and its derivatives (Equations (8) and (9)) are biased but consistent estimators of $L(\mathbf{x}, \vec{\omega}_o)$, $\nabla L(\mathbf{x}, \vec{\omega}_o)$, and $\mathbf{H}L(\mathbf{x}, \vec{\omega}_o)$, respectively. In addition the assumptions imposed in Equation (10) introduce some additional bias due to the piecewise assumption in the scattering f, transmittance T_r, and radiance terms L. However, as shown in Figure 3 our formulation converges accurately to the actual derivatives. Note that we use this biased but consistent approximation only to compute first- and second-order derivatives of media radiance (Equations (8) and (9)), while computing actual radiance values (Equation (1)) using the standard unbiased Monte Carlo estimator. In the following, we describe how to use the derivatives in Equations (8) and (9) for interpolating radiance from a set of cache points, and define an error metric for such interpolation.

5 SECOND-ORDER ERROR CONTROL FOR MEDIA RADIANCE EXTRAPOLATION

The error in radiance caching is controlled by a tolerance value ε, and depends both on how radiance is extrapolated, and on the radiance moments at cache point $\mathbf{x}$. These moments define a valid bounding region $\aleph$ where a point $\mathbf{x}'$ can be used for extrapolation. We provide here the key ideas and resulting equations for the valid regions in the context of 2D and 3D participating media and provide detailed derivations in the supplementary material.

Existing work on radiance caching for participating media estimates the relative error using radiance gradients at $\mathbf{x}$. However, ignoring higher-order derivatives creates suboptimal cache distributions that often oversample regions near surfaces and light sources. Given the radiance and the first n derivatives at a media point $\mathbf{x}$, we can approximate radiance at point $\mathbf{x}' \in \aleph$ using an n^{th}-order Taylor expansion. Following previous work [Schwarzhaupt et al. 2012] we truncate to order one, approximating $L(\mathbf{x}', \vec{\omega}_o)$ as:

$$L(\mathbf{x}', \vec{\omega}_o) \approx L(\mathbf{x}, \vec{\omega}_o) + \nabla L(\mathbf{x}, \vec{\omega}_o) \Delta_{\mathbf{x}'}. \tag{19}$$

Since we focus on isotropic media, we remove the directional dependence in the following derivations to simplify notation. By using a second order expansion of $L(\mathbf{x})$ as our oracle, we can approximate the relative error $\hat{\epsilon}'(\mathbf{x}')$ of the extrapolation as:

$$\hat{\epsilon}'(\mathbf{x}') \approx \frac{\left| \Delta_{\mathbf{x}'}^{\mathsf{T}} \mathbf{H}L(\mathbf{x}) \Delta_{\mathbf{x}'} \right|}{2\, L(\mathbf{x})}, \tag{20}$$

with $\mathbf{H}L(\mathbf{x})$ the Hessian matrix of $L(\mathbf{x})$. This expression is similar to the second-order error metric proposed by Jarosz et al. [2012] and follow-up work by Schwarzhaupt et al. [2012], although these works dealt with surfaces only.

By integrating Equation (20) in the neighborhood of $\mathbf{x}$ for a given error threshold ε, we can express the valid region in two-dimensional media as an ellipse with principal radii $R_{2\mathrm{D}}^{\lambda_i}$ (see Equations (S.9)-(S.12) in the supplemental for the complete derivation):

$$R_{2\mathrm{D}}^{\lambda_i} = \sqrt[4]{\frac{4L(\mathbf{x})\varepsilon}{\pi |\lambda_i|}}, \tag{21}$$

where λ_i is the i-th eigenvalue of the radiance Hessian $\mathbf{H}L(\mathbf{x})$. This formula is analogous to the relative error metric presented by Schwarzhaupt and colleagues [2012] for surfaces, but here the

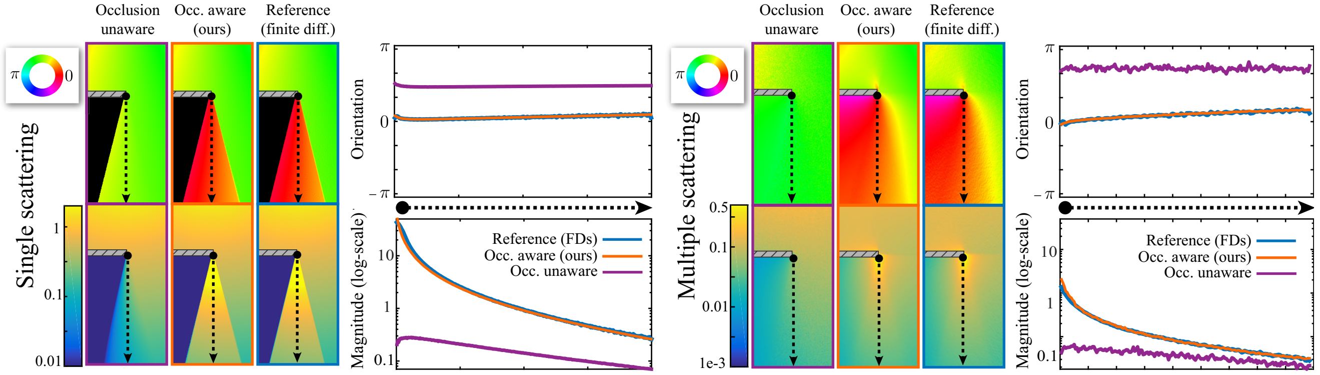

Fig. 5. 2D single and multiple scattering gradients in similar setup to Figure 3, top. Compared against an occlusion-unaware reference solution [Jarosz et al. 2008], our method correctly captures both gradients orientation (color-coded angle), and magnitude. The graphs show the evolution of the gradient across the dotted black line for both methods (purple, orange), and the reference solution (blue).

radii are computed by taking the principal components of the *volumetric* radiance Hessian. Adding the third dimension, the valid region for a cache point becomes a 3D ellipsoid, whose principal radii are:

$$R^{\lambda_i}_{3D} = \sqrt[5]{\frac{15L(\mathbf{x})\varepsilon}{4\pi|\lambda_i|}}. \qquad (22)$$

Our second-order error metric and its derived radius assume knowledge of the radiance and its derivatives at $\mathbf{x}$. In practice, these are usually computed by Monte Carlo techniques, which lead to other sources of error such as variance (inherent to Monte Carlo sampling), or bias (due to inaccuracies computing the derivatives).

The presented metric describes the error introduced by extrapolation from a single cache point in participating media. However, at render time, we compute radiance at each shaded point by interpolating from *multiple* cached points, as:

$$L(\mathbf{x}') \approx \frac{\sum_{k\in C}\left[L(\mathbf{x}_k) + \nabla L(\mathbf{x}_k)\cdot\Delta_{\mathbf{x}'}\right]\mathrm{w}(\mathbf{x}_k,\mathbf{x}')}{\sum_{k\in C}\mathrm{w}(\mathbf{x}_k,\mathbf{x}')}, \qquad (23)$$

with C the set of cache points whose radii include $\mathbf{x}'$, and $\mathrm{w}(\mathbf{x}_k,\mathbf{x}')$ the interpolation kernel. Following Jarosz et al. [2008], we use a cubic interpolation kernel $\mathrm{w}(\mathbf{x}_k,\mathbf{x}') = 3d^2 - 2d^3$ with $d = 1 - \|\mathbf{x}' - \mathbf{x}_k\|\, R_k^{-1}$. Since Equation (23) only interpolates from cache points which predict a maximum error $\hat{\epsilon}' < \varepsilon$ at $\mathbf{x}'$, the error of the weighted sum is equally upper-bounded by ε. Note that, as opposed to Jarosz et al. [2008] (4), we interpolate in linear space, where the error is more accurately predicted by our Hessian-based metric described in Equation (20).

6 RESULTS

In the following we illustrate the accuracy and benefits of our method. We start showing our results in a two-dimensional world, and compare it against a 2D version of the current state-of-the-art method [Jarosz et al. 2008]. We refer the reader to the supplementary material for the additional expressions to compute two-dimensional occlusion-unaware gradients. Then, we move to 3D, to demonstrate that our results are also consistent in a more practical three-dimensional scenario. For comparison purposes, all 3D insets show only single and multiple scattering in media, discarding surface radiance. Unless it is explicitly mentioned, we use isotropic points with the smallest principal axis of the Hessian. This is the most costly scenario for our method in comparison to previous work, since we cannot adapt to the signal as faithfully as with anisotropic points, and therefore require more points.

Implementation: We compute both radiance and derivatives at point $\mathbf{x}$ by stratified sampling uniformly in the sphere, with equal solid angle strata (in the case of 2D, this stratification is in the circle, using equal angle stratification). This reduces variance compared to pure uniform sampling. More importantly, it allows to very simply build the subdivision using the angular samples, by just connecting samples from adjacent strata [Schwarzhaupt et al. 2012]. This stratification is used for both media and surfaces, including area light sources, while other direct light sources (such as directional or point lights) must be handled separately. The accuracy of the subdivision for computing the derivatives relies on a dense sampling of the angular domain, and, as in any sampling problem, our sampling rate limits the amount of radiance changes that we can recover. This is especially important when capturing fine details such as small light sources, which are not computed using next event estimation (NEE), but could also be important in high-frequency fluctuations of radiance in media. However, in practice our method presents much better convergence than previous work [Jarosz et al. 2008] with increasing number of angular samples, as shown in Figure 3. Introducing a NEE-aware subdivision combined with the standard angular one via multiple importance sampling could significantly improve the performance of the derivative computation, although we leave this to future work. We perform the subdivision within the medium by uniformly ray-marching the medium at discrete distances around $\mathbf{x}$, and joining adjacent angular samples within each marching step (see Figure 2e).

Unless stated otherwise, single scattering in all compared methods refers to radiance emitted or reflected (first bounce) by surfaces. We limit multiple scattering to the second bounce for all methods. We did this mainly to reduce excessive variance when computing reference derivatives with finite differences. Note that both occlusion-unaware and occlusion-aware methods are equally applicable to

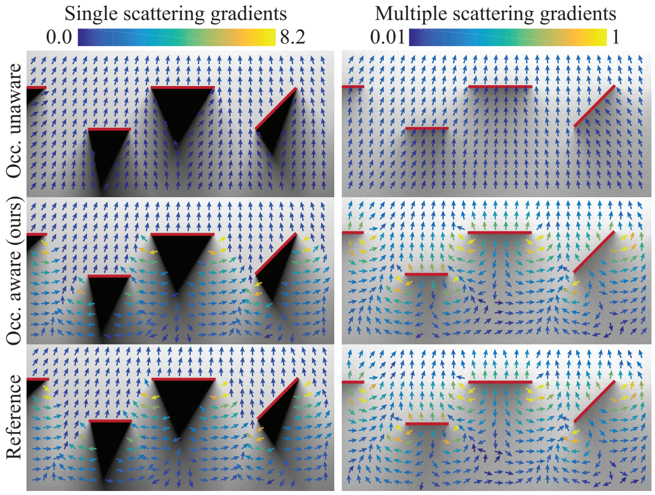

Fig. 6. Radiance gradients at discrete locations in 2D computed with occlusion unaware, and our occlusion aware methods, compared against reference gradients (bottom row) computed with path traced finite differences using 4M samples/gradient. Left column, top and middle, show single scattering gradients computed with 256 angular samples/gradient. Right column, top and middle, show multiple scattering gradients computed with 65536 samples/gradient (256 angular × 256 ray samples).

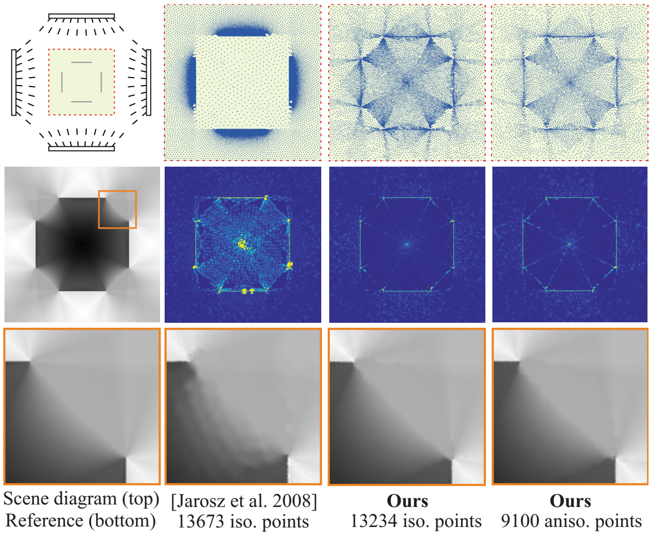

Fig. 7. Single scattering in a 2D setup with four line lights and four occluders. Point distributions (top row) show how a occlusion-unaware gradient metric [Jarosz et al. 2008] fails to estimate the correct radiance changes in complex shadows, while tending to concentrate cache points near reflecting geometry. In contrast, our algorithm distributes points according to occlusion-aware, second-order derivatives of radiance, capturing complex light patterns more accurately. Leveraging curvature information in the Hessians enables anisotropic cache points that further reduce the number of required cache points while maintaining quality (see error maps).

higher number of media bounces, although they usually require a high number of samples to obtain noiseless solutions.

Following previous methods [Jarosz et al. 2008], we first prepopulate the cache by uniformly sampling a ray from the camera, and ray-marching along the media, placing cache points in case they do not fulfill our error metric (Section 5). At render time, we evaluate Equation (23) at ray-marched points $\mathbf{x}'$ in the medium, extrapolating radiance from the surrounding valid cache points. If no valid cache points are found for $\mathbf{x}'$ then we compute its radiance and derivatives, and add it to the cache. As in previous methods [Jarosz et al. 2008], we separate single and multiple scattering caches, each in a different octree for efficient cache query.

All results were computed on a desktop PC with an Intel Core i7 3.4 GHz CPU and 16GB RAM. Note that all methods used for rendering comparisons of the complex 3D scenes *Whiteroom* and *Staircase* were accelerated with Embree ray-tracing kernels [Wald et al. 2014], and therefore the performance with respect to the other 3D scenes is higher.

6.1 Results in 2D

To evaluate the error introduced by our occlusion-aware computations of derivatives in a clear, intuitive way, we rely on their two-dimensional versions. In Figure 3 we showed the convergence of gradient computation with the number of angular samples. Previous approaches not taking into account visibility changes fail to estimate the gradient. In contrast, our derivative formulation converges to the actual gradient, even in areas of penumbra for both single and multiple scattering. The quality of our estimated derivatives increases with the number of angular samples, since the approximations introduced by our assumptions vanish as the strata size diminishes.

In Figure 5 we compare the evolution of single and multiple scattering gradients across a penumbra region, computed with our method and previous work. We illustrate them in polar coordinates (magnitude and orientation) in a simple scene with a medium illuminated by an area light on top, and a line acting as an occluder within the medium. We compute reference gradients with path traced finite differences. Our approach manages to correctly compute both gradients magnitude and orientation in the penumbra region. The right graphs show a progression of gradients along the dotted line. The graphs show that our method is able to match the ground-truth, while the occlusion-unaware method both underestimates the magnitude of the gradient and computes an incorrect direction.

Figure 6 shows a comparison of gradients (shown as a vector field) with the occlusion-unaware method, our technique, and a ground-truth solution computed with finite differences. Our method correctly captures complex radiance changes, including strong changes near occluder boundaries, closely matching the ground-truth reference.

Our error metric takes into account second-order derivatives to drive sample-point density in the scene. Since we use the estimated occlusion-aware Hessians as an oracle of the error, this allows us to place more cache points in areas with higher frequencies. Additionally, our improved gradients allow for a more accurate extrapolation within the valid region of the cache points. Figure 7 (top) shows a scene with overlapping shadows, created by four lights and four

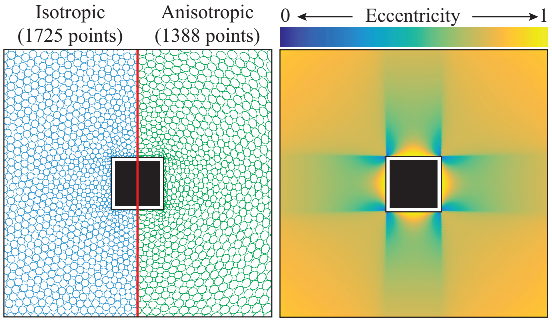

Fig. 8. 2D point distributions in a medium illuminated by a square-shaped light, using our Hessian-based error metric with isotropic and anisotropic points (left) using the same relative error threshold. Eccentricity of radiance curvature (right) determines anisotropy of cache points (left, green), stretching circular points to ellipses along the direction of lower change.

occluders (top-left diagram indicates the shaded region in green). Previous work (second column) drives point density based on the log-space gradient of radiance; in practice this tends to drastically increase point density near light-reflecting geometry, failing to efficiently sample shadowed regions. This can only be mitigated by radius-clamping heuristics (in this case based on the pixel size), thus breaking the principled properties of the approach. In contrast, our method (last two columns) does not rely on heuristics and manages to correctly capture shadows by placing more points near shadow boundaries.

By computing principal components of radiance Hessians, we can use the *radiance eccentricity* (i.e. the eccentricity of the ellipse defined by the Hessian of the radiance) to stretch media cache points along the components with lower radiance variation, obtaining elliptic (2D) or ellipsoidal (3D) cache points. In Figure 7 (bottom) we compare previous work with our isotropic and anisotropic cache distributions. Even with a similar number of isotropic points (∼13k), our improved derivatives manage to capture the overlapping shadows much better; using our anisotropic technique, we manage to reduce cache size by 32%, while keeping the same error threshold. Figure 8 illustrates eccentricity across a 2D scene with a square light emitter in the center. By keeping the same error threshold, our anisotropic cache reduces the number of cache points by up to 20%.

6.2 Results in 3D

Here we further analyze occlusion-unaware gradients and our occlusion-aware Hessians on four 3D scenes: *Strips*, *Statues*, *Patio* and *Cornell holes*. Unless stated otherwise, all renders are taken using 16 samples per pixel, and performing uniform ray marching with a step size of 0.1.

The *Statues* scene shown in Figure 1 combines both surface-to-media single scattering, and media-to-media (two-bounce) multiple scattering. The scene includes distant and local light sources (side windows and ceiling, respectively). Occlusion-unaware single and multiple scattering gradients lead to big splotches on the boundaries of light beams coming through the windows. In the case of light coming through the ceiling, while the point distribution captures shadow contours fairly well, extrapolation fails since occlusion-unaware gradients ignore light effects produced in the penumbra region. Moreover, occlusion-unaware techniques concentrate most cache points near light sources and reflecting surfaces (Figure 1, middle), as seen previously in 2D. Since the gradients are large in these areas, this results in very small valid radii for the cache points. Histograms (Figure 1, right) show how for previous work nearly 8000 points (leftmost bin, top blue histogram) on single scattering reach the minimum radius, which is close to a 40% of the total number of points. This implies that the performance of this approach is highly dependent on the value of such minimum radius, which undermines the principled basis of its error metric. In contrast our method generates better point distributions, which correctly capture light gradients while avoiding additional heuristics to control oversampling in certain regions.

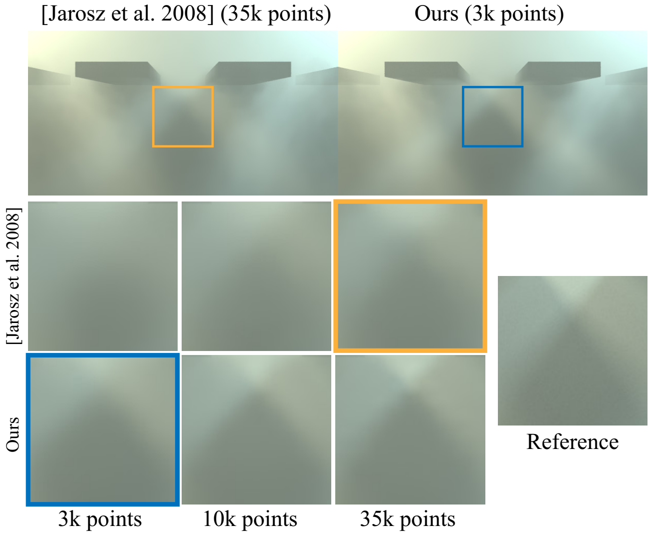

Fig. 9. *Strips* scene comparing single scattering for an increasing number of cache points, and showing relative error with respect to the reference image. While occlusion-unaware gradients method requires 35k cache points to fairly capture occlusions, our occlusion-aware Hessians produce similar results with just 3k points.

The *Strips* scene (Figure 9) shows surface-to-medium single scattering, for an increasing number of cache sizes. Surface radiance is excluded for illustration purposes. The occlusion-unaware method needs an order of magnitude more cache points to get comparable results to ours (see progression insets). This implies that we have to significantly drop the tolerance parameter to create sufficiently fine point distributions in occluded regions. As we can observe in Figure 9, top row, our method yields better sampling density and extrapolation from the sampled points, achieving similar results with an order of magnitude less points.

Computing derivatives of surface-to-medium form factor involves operating with 3×3 matrices (see Appendix C). Including the cost of scene subdivision, this introduces an overhead per cache point of just 9%, compared to computing only point-to-point first derivatives (see Table 2 for the *Patio* scene). Nevertheless, as we can see in

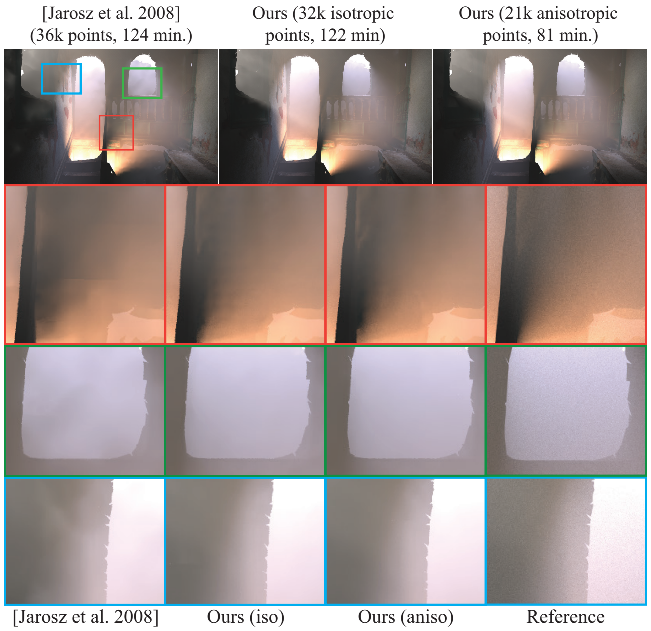

Fig. 10. *Patio* scene with single scattering. Our method outperforms existing occlusion-unaware techniques on an equal-time comparison. Moreover, our anisotropic cache manages to significantly reduce total time under the same error tolerance $\varepsilon = 1.5\text{e}{-4}$ than our isotropic cache, while still retaining shadow details on window boundaries and near thin handrails as shown in the insets.

Table 2. Computation data for the *Patio* scene. For the isotropic case, our method yields better results in equal time. Using anisotropic points provides a further 30% computation time reduction at the same low error threshold due to the improved point distributions and larger valid regions.

Method	Error tol.[3]	Cache gen.	Time / point	Total time
Jarosz et al. 2008	0.3	124 min / 36k pts	206 ms	136 min
Ours (isotropic)	$\varepsilon = 1.5\text{e}{-4}$	122 min / 32k pts	225 ms	135 min
Ours (anisotropic)	$\varepsilon = 1.5\text{e}{-4}$	81 min / 21k pts	225 ms	94 min

Figure 10, our method yields better equal-time results with isotropic points. Moreover, our anisotropic approach stretching spherical cache points along the principal components of radiance, allows to reduce both the number of points and the total computation time by 30% for the same error tolerance.

The *Cornell Holes* scene (Figure 11) shows how our method successfully resolves difficult, high-frequency occlusions due to light coming out of the box. Our method provides a built-in mechanism to significantly reduce error in two ways: additional samples reduce variance but also create finer subdivisions, thus improving accuracy when detecting occlusions.

We also demonstrate the benefits our method in scenes of higher complexity. In the *Staircase* scene (Figure 12) we show an equal-time comparison with a render time of 90 minutes. Path tracing has not fully converged to the reference solution in that time, and while the point distributions of occlusion-unaware methods manage to capture the main shadow boundaries, occlusion-unaware gradients still create visible artifacts on the shadow patterns created by light coming from different windows. Progressive photon beams [Jarosz et al. 2011b] manages to capture high frequency changes, but fails to densely sample the medium due to distant lighting. In equal time, our method manages to get the closest match to the reference by correctly capturing complex shadow configurations. In Figure 13 we also illustrate convergence of our occlusion-aware gradients in the same scene by analyzing the changes on a XZ-aligned slice of the media crossing through the light shafts. We compare our gradients against finite differences gradients on two orthogonal scanlines that cross through the shadows, and demonstrate how our method converges to the reference gradients by creating finer subdivisions with higher number of angular samples.

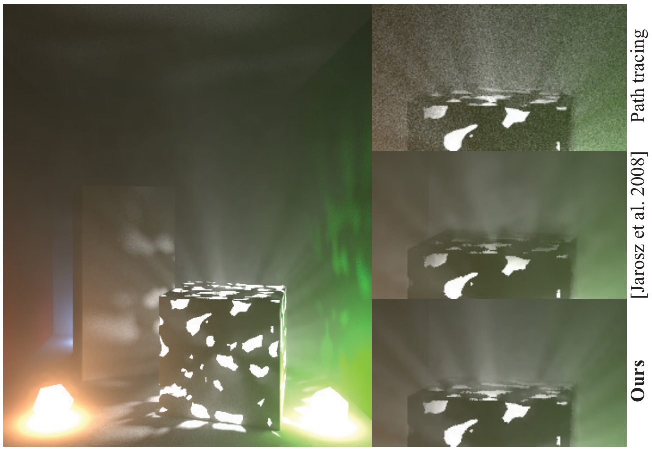

Fig. 11. *Cornell Holes* scene showing complex high-frequency shadows handled by our method, compared against equal-time path traced, and occlusion-unaware gradients solutions.

Finally we perform comparisons up to equal-quality in the *Whiteroom* scene, which presents high scattering due to bright white walls and furniture. In a sequence of insets with increasing render time, we show how our method manages to recover high-frequency shadows in much less time than other methods, which also fail to capture thin shadows near window boundaries.

7 CONCLUSIONS

We have presented a new occlusion-aware method for efficiently computing light transport in homogeneous isotropic media, including both single and multiple scattering. At the core of our method lies an efficient computation of radiance derivatives for both surface-to-medium and medium-to-medium light transport. Our radiance derivatives, including visibility changes for single and multiple scattering, improve both the placement of cache points, as well as their interpolation using a Taylor expansion.

We have additionally formalized light transport in participating media in a self-contained 2D world; we hope that this framework becomes a valuable contribution for the graphics community as a

[3] Note that Jarosz et al.'s metric (Equation (5)) is different from our Hessian-based integrated error ε, thus tolerance values of both metrics have different meaning.

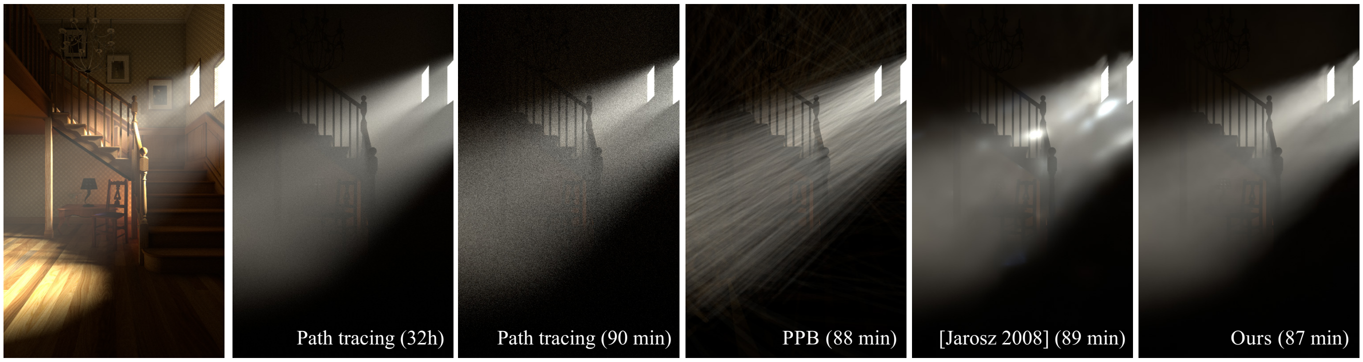

Fig. 12. *Staircase* scene showing equal-time comparisons of path tracing, progressive photon beams (PPB), occlusion-unaware gradients [Jarosz et al. 2008], and our second-order occlusion-aware solution. We include a fully converged solution for path tracing. Each cache in both our method and Jarosz et al. is computed using 16k stratified angular samples, and rendered using 16 samples per pixel. The progressive photon beams solution was obtained using the publicly available Tungsten rendering engine [Bitterli 2016a]. Note how the occlusion unaware method creates visible artifacts in the patterns created by the shadows crossing from different windows, while our method correctly captures those details in equal time. Due to distant lighting, progressive photon beams fails to densely sample the light shafts coming through the windows, resulting in visible variance after 400 iterations of 1M beams/iteration.

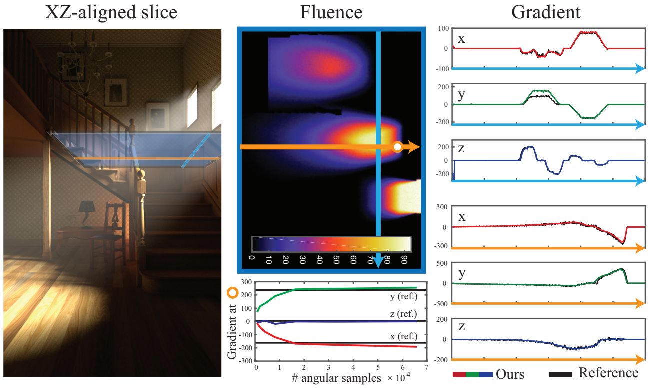

Fig. 13. We demonstrate the convergence of our occlusion aware derivatives in complex 3D scenarios like *Staircase*. We illustrate this using an XZ-aligned slice of the media that captures the occlusion changes produced by the light shafts through the windows. Right graphs show our computed gradients across two orthogonal scan lines of the slice, where we can observe how our method matches the reference derivatives computed with finite differences. In the bottom graph we also illustrate convergence at the white dot respect to the number of angular samples. Higher number of angular samples create finer scene subdivisions and increase the precision of our derivatives, which provide a very good estimation of the actual derivatives.

testbed for novel algorithms. Our results (2D and 3D) demonstrate a significant improvement over the current state of the art, both in equal-time and equal-error comparisons.

Limitations & Future Work. Our work shares some of the limitations of traditional radiance caching algorithms, namely the assumption of relatively low frequency transport with finite derivatives. High-frequency illumination due to e.g. small light sources would require a very fine-grained subdivision to accurately find shadow boundaries. Other high-frequency effects such as caustics would additionally require departing from the assumption of constant angular radiance L_o in Equation (10), which would in turn require computing its translational derivatives.

In our implementation we have assumed isotropic media, which helps reduce the complexity and storage requirements of the cache points. By using an angularly-resolved caching of radiance and its derivatives (by using e.g. spherical harmonics [Jarosz et al. 2008; Křivánek et al. 2005]) anisotropic phase functions could be added. Incorporating heterogeneous media would break the assumption of constant scattering term (i.e. $\nabla f \neq \mathbf{H} f \neq 0$) given the variability of μ_s and f_s within the media. This would require us to use the full radiance derivatives (Equations (11) and (12)), instead of the simplified Equations (17) and (18). Moreover, it would require changing our derivatives of transmittance T_r; given our marching procedure for subdividing the media, a similar approach to Jarosz et al.'s [2008] for single scattering could be used. Finally, high-frequency heterogeneity in the medium would require a very fine subdivision, which would potentially make our approach impractical.

Our error metric assumes that the error is due to extrapolation only, with perfect radiance samples and derivatives. However, both are computed stochastically, which introduces variance (in the case of radiance), and bias (on the derivatives). Developing new metrics taking into account these additional sources of error, as well as accurately characterizing them, are interesting avenues of future work. In this regard, analyzing other consistent approaches to compute derivatives (e.g. using photon mapping [Kaplanyan and Dachsbacher 2013]) might be helpful. Evaluating whether using our biased estimator of radiance (Equation (6)) instead of our Monte Carlo estimate of Equation (1) would be interesting too, making our cache points more robust by reducing variance (at the price of additional bias). Finally, it may be possible to use our first- and second-order derivatives to accurately estimate the optimal kernel in density estimation algorithms for participating media [Hachisuka et al. 2013], as well as to guide sampling in media or to improve quadrature-based ray-marching methods [Muñoz 2014].

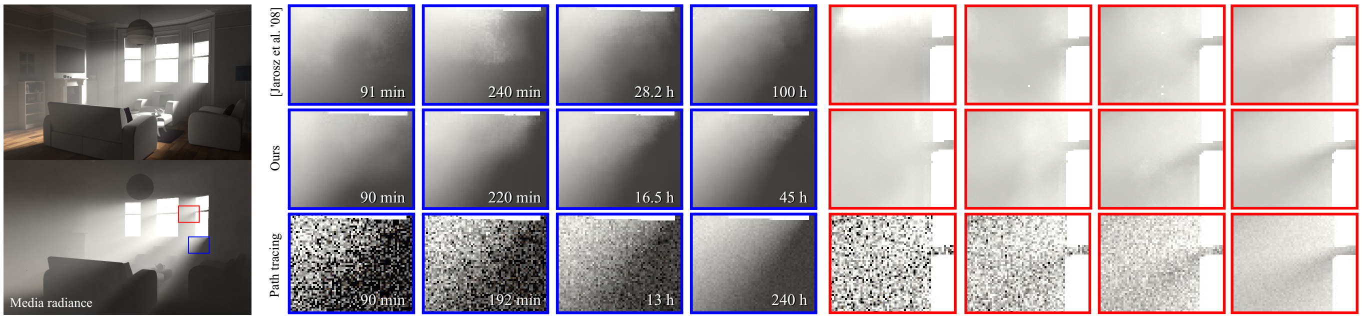

Fig. 14. We illustrate convergence to an equal-quality reference solution with our algorithm, Jarosz et al. method, and path tracing for the *Whiteroom* scene. In 3D scenes of higher complexity, our method presents much better convergence properties than previous ones, being able to reconstruct the shadow boundaries near the window frames in much less time.

ACKNOWLEDGEMENTS

We want to thank the reviewers for their insightful comments. We also want to thank M. S. Suraj for discussions and help during early stages of the project, Ibón Guillén for helping with figures, and Benedikt Bitterli for his help on the comparisons with progressive photon beams. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (CHAMELEON project, grant agreement No 682080), DARPA (project REVEAL), and the Spanish Ministerio de Economía y Competitividad (projects TIN2016-78753-P and TIN2014-61696-EXP). Julio Marco was additionally funded by a grant from the Gobierno de Aragón.

REFERENCES

Andrew Allen and Nikunj Raghuvanshi. 2015. Aerophones in Flatland: Interactive Wave Simulation of Wind Instruments. *ACM Trans. Graph.* 34, 4, Article 134 (July 2015), 11 pages. https://doi.org/10.1145/2767001

James Arvo. 1994. The Irradiance Jacobian for Partially Occluded Polyhedral Sources. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '94)*. ACM, New York, NY, USA, 343–350. https://doi.org/10.1145/192161.192250

Laurent Belcour, Kavita Bala, and Cyril Soler. 2014. A Local Frequency Analysis of Light Scattering and Absorption. *ACM Trans. Graph.* 33, 5, Article 163 (Sept. 2014), 17 pages. https://doi.org/10.1145/2629490

Benedikt Bitterli. 2016a. Rendering resources. (2016). https://benedikt-bitterli.me/resources/

Benedikt Bitterli. 2016b. Virtual femto-photography. (2016). https://benedikt-bitterli.me/femto.html

Subrahmanyan Chandrasekhar. 1960. *Radiative Transfer*. Dover.

Frédo Durand, Nicolas Holzschuch, Cyril Soler, Eric Chan, and François X Sillion. 2005. A frequency analysis of light transport. In *ACM Transactions on Graphics (TOG)*, Vol. 24. ACM, 1115–1126.

Oskar Elek, Pablo Bauszat, Tobias Ritschel, Marcus Magnor, and Hans-Peter Seidel. 2014. Spectral Ray Differentials. *Computer Graphics Forum (Proceedings of EGSR)* 33, 4 (2014). http://people.mpi-inf.mpg.de/~oelek/Papers/SpectralDifferentials/

Diego Gutierrez, Srinivasa G. Narasimhan, Henrik Wann Jensen, and Wojciech Jarosz. 2008. Scattering. In *ACM SIGGRAPH ASIA 2008 Courses*.

Toshiya Hachisuka, Wojciech Jarosz, Iliyan Georgiev, Anton Kaplanyan, and Derek Nowrouzezahrai. 2013. State of the Art in Photon Density Estimation. In *ACM SIGGRAPH Asia Courses*.

Toshiya Hachisuka, Wojciech Jarosz, and Henrik Wann Jensen. 2010. A Progressive Error Estimation Framework for Photon Density Estimation. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 29, 6, Article 144 (December 2010), 144:1–144:12 pages. https://doi.org/10.1145/1882261.1866170

Paul S. Heckbert. 1992. Radiosity in Flatland. *Computer Graphics Forum* 2 (1992), 181–192.

Robert Herzog, Karol Myszkowski, and Hans-Peter Seidel. 2009. Anisotropic Radiance-Cache Splatting for Efficiently Computing High-Quality Global Illumination with Lightcuts. 28, 2 (2009), 259–268.

Nicolas Holzschuch and Francois Sillion. 1998. An Exhaustive Error-Bounding Algorithm for Hierarchical Radiosity. In *Computer Graphics Forum*, Vol. 17. Wiley Online Library, 197–218.

Nicolas Holzschuch and François X. Sillion. 1995. Accurate Computation of the Radiosity Gradient with Constant and Linear Emitters. In *Rendering Techniques 1995 (Eurographics Symposium on Rendering)*. Dublin, Ireland, 186–195. https://hal.inria.fr/inria-00379406

Homan Igehy. 1999. Tracing ray differentials. *Proceedings of SIGGRAPH* 33 (1999), 179–186.

Adrian Jarabo, Julio Marco, Adolfo Muñoz, Raul Buisan, Wojciech Jarosz, and Diego Gutierrez. 2014. A Framework for Transient Rendering. *ACM Transactions on Graphics (SIGGRAPH Asia 2014)* 33, 6 (2014).

Wojciech Jarosz, Craig Donner, Matthias Zwicker, and Henrik Wann Jensen. 2008. Radiance Caching for Participating Media. *ACM Transactions on Graphics (Presented at SIGGRAPH)* 27, 1 (March 2008), 7:1âĂŞ7:11. https://doi.org/10.1145/1330511.1330518

Wojciech Jarosz, Derek Nowrouzezahrai, Iman Sadeghi, and Henrik Wann Jensen. 2011a. A Comprehensive Theory of Volumetric Radiance Estimation Using Photon Points and Beams. *ACM Transactions on Graphics (Presented at SIGGRAPH)* 30, 1 (Jan. 2011), 5:1âĂŞ5:19. https://doi.org/10.1145/1899404.1899409

Wojciech Jarosz, Derek Nowrouzezahrai, Robert Thomas, Peter-Pike Sloan, and Matthias Zwicker. 2011b. Progressive Photon Beams. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 30, 6 (December 2011). https://doi.org/10.1145/2070781.2024215

Wojciech Jarosz, Volker Schönefeld, Leif Kobbelt, and Henrik Wann Jensen. 2012. Theory, Analysis and Applications of 2D Global Illumination. *ACM Transactions on Graphics (Presented at SIGGRAPH)* 31, 5 (Sept. 2012), 125:1âĂŞ125:21. https://doi.org/10.1145/2231816.2231823

Wojciech Jarosz, Matthias Zwicker, and Henrik Wann Jensen. 2008. Irradiance Gradients in the Presence of Participating Media and Occlusions. *Computer Graphics Forum (Proceedings of EGSR)* 27, 4 (June 2008), 1087–1096. https://doi.org/10.1111/j.1467-8659.2008.01246.x

James T. Kajiya. 1986. The Rendering Equation. *SIGGRAPH Comput. Graph.* 20, 4 (Aug. 1986), 143–150. https://doi.org/10.1145/15886.15902

Anton S. Kaplanyan and Carsten Dachsbacher. 2013. Adaptive Progressive Photon Mapping. *ACM Transactions on Graphics* 32, 2 (April 2013), 16:1–16:13.

Markus Kettunen, Marco Manzi, Miika Aittala, Jaakko Lehtinen, Frédo Durand, and Matthias Zwicker. 2015. Gradient-Domain Path Tracing. *ACM Trans. Graph.* 34, 4 (2015).

Jaroslav Křivánek, Kadi Bouatouch, Sumanta N. Pattanaik, and Jiří Žára. 2006. Making Radiance and Irradiance Caching Practical: Adaptive Caching and Neighbor Clamping. In *Rendering Techniques 2006, Eurographics Symposium on Rendering*. Eurographics Association, Nicosia, Cyprus, 127–138.

Jaroslav Křivánek and Pascal Gautron. 2009. *Practical Global Illumination with Irradiance Caching*. Morgan & Claypool. https://doi.org/10.2200/S00180ED1V01Y200903CGR010

Jaroslav Křivánek, Pascal Gautron, Kadi Bouatouch, and Sumanta Pattanaik. 2005. Improved radiance gradient computation. In *Proceedings of the 21st spring conference on Computer graphics (SCCG '05)*. ACM, New York, NY, USA, 155–159. https://doi.org/10.1145/1090122.1090148

Jaroslav Křivánek, Pascal Gautron, Sumanta Pattanaik, and Kadi Bouatouch. 2005. Radiance caching for efficient global illumination computation. *Visualization and Computer Graphics, IEEE Transactions on* 11, 5 (sept.-oct. 2005), 550 –561. https://doi.org/10.1109/TVCG.2005.83

Jaakko Lehtinen, Tero Karras, Samuli Laine, Miika Aittala, Frédo Durand, and Timo Aila. 2013. Gradient-Domain Metropolis Light Transport. *ACM Trans. Graph.* 32, 4 (2013).

Marco Manzi, Markus Kettunen, Miika Aittala, Jaakko Lehtinen, Fredo Durand, and Matthias Zwicker. 2015. Gradient-Domain Bidirectional Path Tracing. In *Proc. of EGSR.*

Marco Manzi, Fabrice Rousselle, Markus Kettunen, Jaakko Lehtinen, and Matthias Zwicker. 2014. Improved Sampling for Gradient-domain Metropolis Light Transport. *ACM Trans. Graph.* 33, 6, Article 178 (2014).

Soham Uday Mehta, Brandon Wang, Ravi Ramamoorthi, and Fredo Durand. 2013. Axis-aligned Filtering for Interactive Physically-based Diffuse Indirect Lighting. *ACM Trans. Graph.* 32, 4, Article 96 (July 2013), 12 pages. https://doi.org/10.1145/2461912.2461947

Adolfo Muñoz. 2014. Higher Order Ray Marching. *Computer Graphics Forum* 33, 8 (2014), 167–176. https://doi.org/10.1111/cgf.12424

Rachel Orti, Stephane Riviere, Fredo Durand, and Claude Puech. 1996. Radiosity for dynamic scenes in flatland with the visibility complex. In *Computer Graphics Forum*, Vol. 15. 237–248.

Ravi Ramamoorthi, Dhruv Mahajan, and Peter Belhumeur. 2007. A First-order Analysis of Lighting, Shading, and Shadows. *ACM Trans. Graph.* 26, 1, Article 2 (Jan. 2007). https://doi.org/10.1145/1189762.1189764

Mickaël Ribardière, Samuel Carré, and Kadi Bouatouch. 2011. Adaptive records for volume irradiance caching. *The Visual Computer* 27, 6 (2011), 655–664.

Fabrice Rousselle, Wojciech Jarosz, and Jan Novák. 2016. Image-space Control Variates for Rendering. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 35, 6 (December 2016), 169:1–169:12. https://doi.org/10.1145/2980179.2982443

Lars Schjøth, Jeppe Revall Frisvad, Kenny Erleben, and Jon Sporring. 2007. Photon Differentials. In *Proceedings of the 5th International Conference on Computer Graphics and Interactive Techniques in Australia and Southeast Asia (GRAPHITE '07).* ACM, New York, NY, USA, 179–186. https://doi.org/10.1145/1321261.1321293

Jorge Schwarzhaupt, Henrik Wann Jensen, and Wojciech Jarosz. 2012. Practical Hessian-Based Error Control for Irradiance Caching. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 31, 6 (Nov. 2012). https://doi.org/10.1145/2366145.2366212

Frank Suykens and Yves D. Willems. 2001. Path differentials and applications. In *Rendering Techniques 2001 (Proceedings of the Eurographics Workshop on Rendering).* Eurographics Association, London, United Kingdom, 257–268. https://doi.org/10.1007/978-3-7091-6242-2_24

Adriaan Van Oosterom and Jan Strackee. 1983. The Solid Angle of a Plane Triangle. *IEEE Transactions on Biomedical Engineering* BME-30, 2 (1983), 125 – 126. https://doi.org/10.1109/TBME.1983.325207

Ingo Wald, Sven Woop, Carsten Benthin, Gregory S Johnson, and Manfred Ernst. 2014. Embree: a kernel framework for efficient CPU ray tracing. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 143.

Gregory J. Ward and Paul S. Heckbert. 1992. Irradiance Gradients. *1992 Eurographics Workshop on Rendering*, 85–98. http://radsite.lbl.gov/radiance/papers/erw92/paper.html

Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. 1988. A Ray Tracing Solution for Diffuse Interreflection. In *Proceedings of the 15th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '88).* ACM, New York, NY, USA, 85–92. https://doi.org/10.1145/54852.378490

Ling-Qi Yan, Miloš Hašan, Wenzel Jakob, Jason Lawrence, Steve Marschner, and Ravi Ramamoorthi. 2014. Rendering Glints on High-Resolution Normal-Mapped Specular Surfaces. *ACM Trans. Graph. (Proc. SIGGRAPH)* 33, 4 (2014).

Ling-Qi Yan, Miloš Hašan, Steve Marschner, and Ravi Ramamoorthi. 2016. Position-Normal Distributions for Efficient Rendering of Specular Microstructure. *ACM Trans. Graph. (Proc. SIGGRAPH)* 35, 4 (2016).

Matthias Zwicker, Wojciech Jarosz, Jaakko Lehtinen, Bochang Moon, Ravi Ramamoorthi, Fabrice Rousselle, Pradeep Sen, Cyril Soler, and Sung-Eui Yoon. 2015. Recent Advances in Adaptive Sampling and Reconstruction for Monte Carlo Rendering. *Computer Graphics Forum (Proceedings of Eurographics)* 34, 2 (May 2015), 667–681. https://doi.org/10.1111/cgf.12592

APPENDICES

In the following we summarize 2D and 3D expressions of translational derivatives of transmittance and form factors needed for our method. We box all relevant final expressions that to the best of our knowledge are new to the literature. We define column vectors as $\vec{\mathbf{v}}$ and row vectors as $\vec{\mathbf{v}}^\intercal$. Expressions such as $\vec{\mathbf{r}}_1 \cdot \vec{\mathbf{r}}_2$ denote dot (inner) products, while expressions such as $\vec{\mathbf{r}}_1 \vec{\mathbf{r}}_2{}^\intercal$, $\nabla(\ldots)\nabla^\intercal(\ldots)$, and $(\ldots)(\ldots)^\intercal$ denote vector *outer* products.

A HOMOGENEOUS TRANSMITTANCE DERIVATIVES

Homogeneous transmittance is modeled by the exponential decay due to extinction,

$$T_r = e^{-\mu_t \|\overrightarrow{\mathbf{yx}}\|} \tag{24}$$

where $\|\overrightarrow{\mathbf{xy}}\|$ denotes distance between source $\mathbf{y}$ and shaded point $\mathbf{x}$. Its gradient and Hessian with respect to a translation of $\mathbf{x}$ are

$$\nabla T_r = -\mu_t \frac{\overrightarrow{\mathbf{yx}}}{r} T_r, \tag{25}$$

$$\boxed{\mathbf{H} T_r = -\mu_t \Big(\frac{\mathbf{J}(\overrightarrow{\mathbf{yx}})}{r} - \frac{1}{r^3} \overrightarrow{\mathbf{yx}}\,\overrightarrow{\mathbf{yx}}^\intercal - \frac{\mu_t}{r^2} \overrightarrow{\mathbf{yx}}\,\overrightarrow{\mathbf{yx}}^\intercal\Big)\, T_r. \qquad (26)}$$

B 2D SEGMENT-MEDIA FORM FACTOR DERIVATIVES

The form factor between a 2D segment ℓ and a media point $\mathbf{x}$ (Figure 15, left) is defined as the integrated curve-media geometry term along all segment points. This is equivalent to the angular ratio covered by ℓ as seen from $\mathbf{x}$

$$F_\ell(\mathbf{x}) = \frac{1}{2\pi} \int_{\mathbf{y}_0}^{\mathbf{y}_1} \frac{\cos\theta_{\mathbf{y}}}{\|\mathbf{x} - \mathbf{y}\|} \mathrm{d}\ell(\mathbf{y}) = \frac{1}{2\pi} \arccos\left(\frac{\overrightarrow{\mathbf{xy}}_0}{r_0} \cdot \frac{\overrightarrow{\mathbf{xy}}_1}{r_1}\right).$$

where $r_i = \|\overrightarrow{\mathbf{xy}}_i\|$. The form factor gradient and Hessian become

$$\boxed{\nabla F_\ell(\mathbf{x}) = -\frac{1}{2\pi} \frac{\nabla \cos\theta'}{\sqrt{1 - \cos^2\theta'}} \qquad (27)}$$

$$\boxed{\begin{aligned} \mathbf{H} F_\ell(\mathbf{x}) = -\frac{1}{2\pi} \Bigg(& \frac{\mathbf{J}(\nabla \cos\theta')}{\sqrt{1 - \cos^2\theta'}} \\ & + \frac{\cos\theta'}{(1 - \cos^2\theta')^{3/2}} \nabla \cos\theta' \nabla^\intercal \cos\theta' \Bigg) \qquad (28) \end{aligned}}$$

where $\mathbf{J}$ is the Jacobian operator, and:

$$\nabla \cos\theta' = \frac{\cos\theta'}{r_0^2} \overrightarrow{\mathbf{xy}}_0 + \frac{\cos\theta'}{r_1^2} \overrightarrow{\mathbf{xy}}_1 \tag{29}$$

$$- \frac{(\overrightarrow{\mathbf{xy}}_0 + \overrightarrow{\mathbf{xy}}_1)}{r_0 r_1}, \tag{30}$$

$$\begin{aligned} \mathbf{J}(\nabla \cos\theta') = & -\mathbf{J}\left(\frac{\overrightarrow{\mathbf{xy}}_0}{r_0 r_1}\right) - \mathbf{J}\left(\frac{\overrightarrow{\mathbf{xy}}_1}{r_0 r_1}\right) \\ & + \mathbf{J}\left(\frac{\cos\theta'}{r_0^2} \overrightarrow{\mathbf{xy}}_0\right) + \mathbf{J}\left(\frac{\cos\theta'}{r_1^2} \overrightarrow{\mathbf{xy}}_1\right), \end{aligned} \tag{31}$$

$$\mathbf{J}\left(\frac{\overrightarrow{\mathbf{xy}}_i}{r_0 r_1}\right) = \frac{\mathbf{J}(\overrightarrow{\mathbf{xy}}_i)}{r_0 r_1} + \frac{\overrightarrow{\mathbf{xy}}_i \overrightarrow{\mathbf{xy}}_0^\intercal}{r_0^3 r_1} + \frac{\overrightarrow{\mathbf{xy}}_i \overrightarrow{\mathbf{xy}}_1^\intercal}{r_0 r_1^3}, \tag{32}$$

$$\begin{aligned} \mathbf{J}\left(\frac{\cos\theta'}{r_i^2} \overrightarrow{\mathbf{xy}}_i\right) = & \frac{\cos\theta'}{r_i^2} \mathbf{J}(\overrightarrow{\mathbf{xy}}_i) + \frac{\overrightarrow{\mathbf{xy}}_i}{r_i^2} \nabla^\intercal \cos\theta' \\ & + \frac{2\cos\theta'}{r_i^4} \overrightarrow{\mathbf{xy}}_i \overrightarrow{\mathbf{xy}}_i^\intercal. \end{aligned} \tag{33}$$

C 3D TRIANGLE-MEDIA FORM FACTOR DERIVATIVES

The form factor between a 3D triangular face △ and a media point $\mathbf{x}$ (see Figure 15, right) is defined as the integrated surface-media geometry term along all points in the triangle. Analogous to 2D,

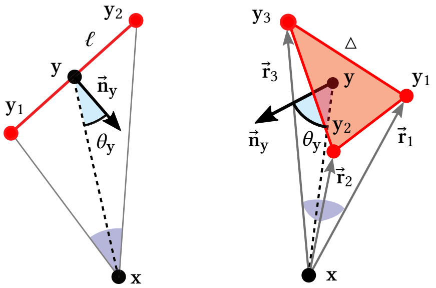

Fig. 15. Setups for segment-to-media (2D, left) and triangle-to-media (3D, right) form factors.

this has analytical solution equal to the ratio of solid angle covered by the triangle as seen from $\mathbf{x}$,

$$F_\triangle(\mathbf{x}) = \frac{1}{4\pi}\int_{\mathbf{y}\in\triangle} \frac{\cos\theta_\mathbf{y}}{\|\mathbf{x}-\mathbf{y}\|^2}\,\mathrm{d}\triangle(\mathbf{y}) = \frac{\Omega}{4\pi}. \tag{34}$$

Solid angle Ω of a triangle can be computed as [Van Oosterom and Strackee 1983],

$$\Omega = 2\arctan\frac{|A|}{B} \tag{35}$$

with

$$A = \vec{\mathbf{r}}_1\cdot(\vec{\mathbf{r}}_2\times\vec{\mathbf{r}}_3) \tag{36}$$

$$B = \mathrm{r}_1\mathrm{r}_2\mathrm{r}_3 + (\vec{\mathbf{r}}_1\cdot\vec{\mathbf{r}}_2)\,\mathrm{r}_3 + (\vec{\mathbf{r}}_2\cdot\vec{\mathbf{r}}_3)\,\mathrm{r}_1 + (\vec{\mathbf{r}}_1\cdot\vec{\mathbf{r}}_3)\,\mathrm{r}_2 \tag{37}$$

where $\vec{\mathbf{r}}_i = \overrightarrow{\mathbf{xy}_i}$, and $\mathrm{r}_i = \|\vec{\mathbf{r}}_i\|$ (see Figure 15, right). Note that the numerator A requires an absolute value to ensure positive vector order (i.e. triangle winding) with respect to $\mathbf{x}$. Also, when obtaining negative arctangent values, π must be added to the obtained solid angle.

The gradient of the form factor with respect to a translation of $\mathbf{x}$ becomes

$$\nabla F_\triangle(\mathbf{x}) = \frac{1}{2\pi}\nabla\arctan\frac{|A|}{B} \tag{38}$$

$$= \frac{1}{2\pi}\frac{B\nabla|A| - |A|\nabla B}{|A|^2+B^2}, \tag{39}$$

and its Hessian yields

$$\mathbf{H}F_\triangle(\mathbf{x}) = \frac{1}{2\pi}\left(\frac{\nabla(|A|)\nabla^\intercal B - \nabla B\nabla^\intercal(|A|)}{|A|^2+B^2} + \frac{B\mathbf{J}(\nabla(|A|)) - |A|\mathbf{J}(\nabla B)}{|A|^2+B^2} - \frac{(B\nabla(|A|)-|A|\nabla B)\left(\nabla(|A|^2)+\nabla(B^2)\right)^\intercal}{\left(|A|^2+B^2\right)^2}\right). \tag{40}$$

Note that for computing the terms $\nabla(|A|)$ and $\mathbf{J}(\nabla|A|)$, we can apply the derivatives of the absolute value of a vector function:

$$\nabla(|A|) = \frac{A}{|A|}\nabla A, \tag{41}$$

$$\mathbf{J}(\nabla|A|) = \frac{A\mathbf{J}(\nabla A) + \nabla A\nabla^\intercal A}{|A|} - \frac{A^2(\nabla A\nabla^\intercal A)}{|A|^3}. \tag{42}$$

The gradient of A becomes

$$\nabla A = \mathbf{J}(\vec{\mathbf{r}}_2\times\vec{\mathbf{r}}_3)\vec{\mathbf{r}}_1 + \mathbf{J}(\vec{\mathbf{r}}_1)(\vec{\mathbf{r}}_2\times\vec{\mathbf{r}}_3). \tag{43}$$

By the Jacobi identity we have that

$$\mathbf{J}(\vec{\mathbf{r}}_2\times\vec{\mathbf{r}}_3) = \vec{\mathbf{r}}_2\times\mathbf{J}(\vec{\mathbf{r}}_3) - \vec{\mathbf{r}}_3\times\mathbf{J}(\vec{\mathbf{r}}_2) \tag{44}$$

where any vector-matrix cross product $\vec{\mathbf{v}}\times\mathbf{J}(\bullet)$ can be expressed by means of the matrix multiplication form

$$\vec{\mathbf{v}}\times\mathbf{J}(\bullet) = \langle\vec{\mathbf{v}}\rangle\mathbf{J}(\bullet) \tag{45}$$

$$\vec{\mathbf{v}} = \begin{pmatrix} v_{(1)} \\ v_{(2)} \\ v_{(3)} \end{pmatrix}, \qquad \langle\vec{\mathbf{v}}\rangle = \begin{pmatrix} 0 & -v_{(3)} & v_{(2)} \\ v_{(3)} & 0 & -v_{(1)} \\ -v_{(2)} & v_{(1)} & 0 \end{pmatrix}. \tag{46}$$

Since $\mathbf{J}(\vec{\mathbf{r}}_1)=\mathbf{J}(\vec{\mathbf{r}}_2)=\mathbf{J}(\vec{\mathbf{r}}_3)=-I_3$, we have that

$$\nabla A = \left(\langle\vec{\mathbf{r}}_2\rangle\mathbf{J}(\vec{\mathbf{r}}_3) - \langle\vec{\mathbf{r}}_3\rangle\mathbf{J}(\vec{\mathbf{r}}_2)\right)\vec{\mathbf{r}}_1 - (\vec{\mathbf{r}}_2\times\vec{\mathbf{r}}_3)$$

$$= \langle\vec{\mathbf{r}}_3 - \vec{\mathbf{r}}_2\rangle\vec{\mathbf{r}}_1 - (\vec{\mathbf{r}}_2\times\vec{\mathbf{r}}_3). \tag{47}$$

Note that $\langle\vec{\mathbf{r}}_3 - \vec{\mathbf{r}}_2\rangle = \langle\mathbf{y}_3 - \mathbf{y}_2\rangle$ and therefore does not depend on $\mathbf{x}$, and $\langle\vec{\mathbf{v}}\rangle^\intercal = \langle-\vec{\mathbf{v}}\rangle$ (see Equation (46)). As a result, the Jacobian of ∇A becomes a zero matrix

$$\mathbf{J}(\nabla A) = \mathbf{J}(\vec{\mathbf{r}}_1)\langle\vec{\mathbf{r}}_3 - \vec{\mathbf{r}}_2\rangle^\intercal - \mathbf{J}(\vec{\mathbf{r}}_2\times\vec{\mathbf{r}}_3)$$

$$= \langle\vec{\mathbf{r}}_3 - \vec{\mathbf{r}}_2\rangle - \langle\vec{\mathbf{r}}_3 - \vec{\mathbf{r}}_2\rangle$$

$$= 0. \tag{48}$$

The gradient of B becomes

$$\nabla B = \nabla(\mathrm{r}_1\mathrm{r}_2\mathrm{r}_3) + \nabla\left((\vec{\mathbf{r}}_1\cdot\vec{\mathbf{r}}_2)\,\mathrm{r}_3\right) + \nabla\left((\vec{\mathbf{r}}_2\cdot\vec{\mathbf{r}}_3)\,\mathrm{r}_1\right) + \nabla\left((\vec{\mathbf{r}}_1\cdot\vec{\mathbf{r}}_3)\,\mathrm{r}_2\right) \tag{49}$$

where

$$\nabla(\mathrm{r}_1\mathrm{r}_2\mathrm{r}_3) = \mathrm{r}_2\mathrm{r}_3\nabla\mathrm{r}_1 + \mathrm{r}_1\mathrm{r}_3\nabla\mathrm{r}_2 + \mathrm{r}_1\mathrm{r}_2\nabla\mathrm{r}_3 \tag{50}$$

$$\nabla\left((\vec{\mathbf{r}}_i\cdot\vec{\mathbf{r}}_j)\,\mathrm{r}_k\right) = (\vec{\mathbf{r}}_i\cdot\vec{\mathbf{r}}_j)\,\nabla\mathrm{r}_k - \mathrm{r}_k(\vec{\mathbf{r}}_i + \vec{\mathbf{r}}_j) \tag{51}$$

$$\nabla\mathrm{r} = -\frac{\vec{\mathbf{r}}}{\mathrm{r}}. \tag{52}$$

Jacobian of ∇B yields

$$\mathbf{J}(\nabla B) = \mathbf{J}(\nabla(\mathrm{r}_1\mathrm{r}_2\mathrm{r}_3)) + \mathbf{J}\left(\nabla\left((\vec{\mathbf{r}}_1\cdot\vec{\mathbf{r}}_2)\,\mathrm{r}_3\right)\right) + \mathbf{J}\left(\nabla\left((\vec{\mathbf{r}}_2\cdot\vec{\mathbf{r}}_3)\,\mathrm{r}_1\right)\right) + \mathbf{J}\left(\nabla\left((\vec{\mathbf{r}}_1\cdot\vec{\mathbf{r}}_3)\,\mathrm{r}_2\right)\right) \tag{53}$$

where

$$\begin{aligned}\mathbf{J}(\nabla(\mathrm{r}_1\mathrm{r}_2\mathrm{r}_3)) = {}& \mathrm{r}_2\mathrm{r}_3\mathbf{J}(\nabla\mathrm{r}_1) + \mathrm{r}_1(\nabla\mathrm{r}_3\nabla^\intercal\mathrm{r}_2 + \nabla\mathrm{r}_2\nabla^\intercal\mathrm{r}_3) \\ & + \mathrm{r}_1\mathrm{r}_3\mathbf{J}(\nabla\mathrm{r}_2) + \mathrm{r}_2(\nabla\mathrm{r}_3\nabla^\intercal\mathrm{r}_1 + \nabla\mathrm{r}_1\nabla^\intercal\mathrm{r}_3) \\ & + \mathrm{r}_1\mathrm{r}_2\mathbf{J}(\nabla\mathrm{r}_3) + \mathrm{r}_3(\nabla\mathrm{r}_2\nabla^\intercal\mathrm{r}_1 + \nabla\mathrm{r}_1\nabla^\intercal\mathrm{r}_2)\end{aligned} \tag{54}$$

$$\begin{aligned}\mathbf{J}\left(\nabla\left((\vec{\mathbf{r}}_i\cdot\vec{\mathbf{r}}_j)\,\mathrm{r}_k\right)\right) = {}& (\vec{\mathbf{r}}_i\cdot\vec{\mathbf{r}}_j)\,\mathbf{J}(\nabla\mathrm{r}_k) + 2\mathrm{r}_k I_3 \\ & - \nabla\mathrm{r}_k(\vec{\mathbf{r}}_i + \vec{\mathbf{r}}_j)^\intercal - (\vec{\mathbf{r}}_i + \vec{\mathbf{r}}_j)\nabla^\intercal\mathrm{r}_k\end{aligned} \tag{55}$$

$$\mathbf{J}(\nabla\mathrm{r}) = \frac{I_3}{\mathrm{r}} - \frac{\vec{\mathbf{r}}\,\vec{\mathbf{r}}^\intercal}{\mathrm{r}^3}. \tag{56}$$