Eurographics Symposium on Rendering 2008
Steve Marschner and Michael Wimmer
(Guest Editors)

Volume 27 (2008), Number 4

Irradiance Gradients in the Presence of Participating Media and Occlusions

Wojciech Jarosz Matthias Zwicker Henrik Wann Jensen

University of California, San Diego

Abstract

In this paper we present a technique for computing translational gradients of indirect surface reflectance in scenes containing participating media and significant occlusions. These gradients describe how the incident radiance field changes with respect to translation on surfaces. Previous techniques for computing gradients ignore the effects of volume scattering and attenuation and assume that radiance is constant along rays connecting surfaces. We present a novel gradient formulation that correctly captures the influence of participating media. Our formulation accurately accounts for changes of occlusion, including the effect of surfaces occluding scattering media. We show how the proposed gradients can be used within an irradiance caching framework to more accurately handle scenes with participating media, providing significant improvements in interpolation quality.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.7]: Three-Dimensional Graphics and Realism—

1. Introduction

Some of the most visually compelling phenomena are caused by the intricate behavior of light within the natural world. However, faithful simulation of global light transport is expensive because it involves solving a recursive integral known as the rendering equation [Kaj86]. In the presence of scattering media the full three-dimensional radiative transfer equation [Cha60] must be considered, which is even more costly to compute. The most popular and successful algorithms to solve these equations are based on some form of stochastic ray tracing and Monte Carlo integration [Gla95]. Although these techniques tend to be very general and robust, they often suffer from noise.

Irradiance caching [WRC88] is one of the most successful strategies to improve the efficiency of Monte Carlo techniques. Irradiance caching computes an accurate solution of the light transport problem only at a sparse set of *cache points* and interpolates the cached values across the surfaces in the scene. In addition, the accuracy of the interpolation can be improved significantly by computing the gradients of illumination at the cache points. In Table 1 we summarize the most important previous techniques for computing illumination gradients for irradiance caching. However, none of these methods take into account the full radiative transport equation, which leads to increased interpolation artifacts in the presence of participating media.

In this paper we present extensions to these techniques to reduce these problems. We develop novel gradients that include absorption, emission, and scattering in volumetric media, and the effect of surfaces occluding media. In particular, we make the following contributions:

1. We derive gradients to handle indirect illumination from surfaces in the presence of **absorbing** media and occlusions. This generalization also allows us to handle non-Lambertian reflectors.
2. We derive gradients of irradiance in the presence of **scattering** or **emissive** media and account for surface-medium occlusion changes.

These two contributions enable, for the first time, the accurate computation of illumination gradients in the context of the full radiative transport equation. Without tracing any additional rays, our gradients are more accurate and contain significantly less noise than gradients computed using expensive numerical techniques such as finite differencing. The resulting gradient computations are easy to implement, and it is straightforward to include them in an irradiance caching framework. We show that our gradients produce higher quality interpolation than previous techniques, which do not take into account full radiative transport.

Method	PM	GCP	GIR	V	II	CO
[WH92]				✓	✓	✓
[Arv94]				✓	✓	
[KGPB05]		✓			✓	✓
[KGBP05]		✓		✓	✓	✓
[RMB07]		✓	✓	✓		✓
[JDZJ08]	✓	✓	✓		✓	✓
Our Method	✓	✓	✓	✓	✓	✓

Table 1: A comparison of the capabilities of illumination gradient techniques. Our gradients take into account the full radiative transport equation including the effects of absorbing, emissive, and scattering participating media (PM). We support cache points on glossy surfaces (GCP), and we consider effects of glossy indirect reflectors (GIR), visibility changes (V), indirect illumination (II), and curved objects (CO).

2. Previous Work

Irradiance Caching. Irradiance caching [WRC88] computes lighting using Monte Carlo ray tracing only at sparse locations and reuses these cached computations whenever possible through interpolation. This technique introduces bias, but exploits the smooth nature of illumination on Lambertian surfaces and significantly improves convergence, especially when combined with photon mapping [Jen01].

Irradiance Gradients. Ward and Heckbert [WH92] introduced a significant improvement to irradiance caching by deriving expressions for the first-order derivative of the incident illumination on Lambertian surfaces. The resulting irradiance gradients can be computed at negligible cost, yet they allow for much higher quality interpolation. Ward and Heckbert, however, do not consider that incident illumination may be subject to absorption in participating media. For a fixed point on a surface that reflects indirect illumination, the gradient computation assumes that if you translate a cache point by a small displacement, the incident radiance from the fixed point on the reflector does not change. This assumption also means that indirect reflectors are treated as diffuse materials for the purpose of gradient computation.

Below we discuss a number of extensions to the classical irradiance gradients by Ward and Heckbert. Please refer to Table 1 for a summary of the features of the different techniques.

Radiance Gradients. Křivánek et al. [KGPB05] further extended irradiance caching to store illumination on glossy surfaces, which is achieved by caching the hemispherical incident radiance field. Annen et al. [AKDS04] and Křivánek et al. [KGPB05] derived gradients of the incident radiance field. These formulation, however, ignored the effect of visibility changes and often produced inaccurate results. Křivánek et al. [KGBP05] later generalized the original irradiance gradient formulation. Their extension allows for both computing gradients of incident radiance on glossy surfaces and accurately taking into account occlusion changes. However, this approach still has the limitations that it does not consider participating media and it assumes Lambertian indirect reflectors for gradient computation.

Radiance Gradients in Participating Media. Jarosz et al. [JDZJ08] extended the radiance caching algorithm to cache inscattered lighting within participating media, and they presented analytic expressions for the gradient of inscattered light. Their gradient formulation takes into account scattering and absorption within anisotropic and heterogeneous media, and by replacing the phase function with a cosine-weighted BRDF it can be easily applied to *surfaces* within participating media. However, their approach does not consider a visibility gradient, thereby underestimating the true gradient when occlusions and volumetric shadow boundaries are present.

First-Order Analysis of Light Transport. Other researchers have investigated first-order forms of illumination. Arvo [Arv94] derived expressions for the Jacobian of irradiance for scenes composed of polygonal and Lambertian surfaces. Ramamoorthi et al. [RMB07] performed a first-order analysis of various shading processes, including a new visibility gradient derivation. Their formulation does not include participating media, and they only demonstrate their approach for evaluating direct illumination.

3. Overview

Our goal in this paper is to compute gradients of surface irradiance that take into account the full radiative transfer, including participating media. In general, irradiance E at a surface location $\mathbf{x}$ can be written as the cosine-weighted integral of incident radiance,

$$E(\mathbf{x}) = \int_{\Omega} L(\mathbf{x},\vec{\omega})(\vec{n}\cdot\vec{\omega})\, d\vec{\omega}, \tag{1}$$

where $\vec{n}$ is the surface normal at $\mathbf{x}$, and $L(\mathbf{x},\vec{\omega})$ is the incident radiance at $\mathbf{x}$ arriving from direction $\vec{\omega}$. We are interested in computing the gradient $\nabla E(\mathbf{x})$ with respect to a translation of $\mathbf{x}$.

In the presence of participating media, light transport obeys the *radiative transfer equation* (RTE) [Cha60]. In this case the incident radiance $L(\mathbf{x},\vec{\omega})$ is the sum of two separate contributions: the radiance L_s coming from surfaces that reflect illumination, and the radiance L_m contributed by the participating medium, i.e.,

$$L(\mathbf{x},\vec{\omega}) = L_s(\mathbf{x},\vec{\omega}) + L_m(\mathbf{x},\vec{\omega}). \tag{2}$$

The radiance L_s due to surfaces expands to

$$L_s(\mathbf{x},\vec{\omega}) = T_r(\mathbf{x}\leftrightarrow\mathbf{x}_s)L(\mathbf{x}_s,-\vec{\omega}), \tag{3}$$

where $\mathbf{x}_s$ is the surface point visible from $\mathbf{x}$ in direction $\vec{\omega}$. The outgoing radiance $L(\mathbf{x}_s,-\vec{\omega})$ is computed by integrating the lighting and BRDF at $\mathbf{x}_s$, and $T_r(\mathbf{x}\leftrightarrow\mathbf{x}_s)$ is the transmittance of the medium between $\mathbf{x}$ and $\mathbf{x}_s$.

The radiance L_m contributed by the medium is

$$L_m(\mathbf{x},\vec{\omega}) = \int_0^s T_r(\mathbf{x}\leftrightarrow\mathbf{x}_t)\sigma_s(\mathbf{x}_t)L_i(\mathbf{x}_t,-\vec{\omega})\, dt, \tag{4}$$

where $\mathbf{x}_t = \mathbf{x} - t\vec{\omega}$ with $t\in(0,s)$, s is the distance through the medium to the nearest surface at $\mathbf{x}_s$, and the remaining quantities are defined in Table 2.

Symbol	Description
$\sigma_s(\mathbf{x})$	Scattering coefficient at $\mathbf{x}$
$\sigma_a(\mathbf{x})$	Absorption coefficient at $\mathbf{x}$
$\sigma_t(\mathbf{x})$	Extinction coefficient at $\mathbf{x}$
$T_r(\mathbf{x} \leftrightarrow \mathbf{x}')$	Transmittance
$L_i(\mathbf{x}, \vec{\omega})$	Inscattered radiance from media
N	Number of divisions in azimuthal angle ϕ
M	Number of divisions in elevation angle θ
$\vec{\omega}_{j,k}$	Sample direction for cell (j,k), i.e.: $(\theta_{j,k}, \phi_{j,k})$
$L_s(\mathbf{x}, \vec{\omega}_{j,k})$	Incoming surface radiance through cell (j,k)
$A_{j,k}$	Area of cell (j,k), i.e.: $(MN)^{-1} pdf(\vec{\omega}_{j,k})^{-1}$
$(j_-,k),(j_+,k)$	Cell boundaries $(j \leftrightarrow j\text{-}1,k)$ and $(j \leftrightarrow j\text{+}1,k)$
$(j,k_-),(j,k_+)$	Cell boundaries $(j,k \leftrightarrow k\text{-}1)$ and $(j,k \leftrightarrow k\text{+}1)$
$\hat{u}_k$	Tangent vector in the ϕ_k direction
$\hat{v}_{k_-}$	Tangent vector in the $\phi_{k_-} + \frac{\pi}{2}$ direction

Table 2: Definitions of quantities used throughout the paper.

To improve the clarity of our derivations we split up the total irradiance in Equation 1 into irradiance from surfaces E_s and irradiance from the medium E_m such that

$$E_s(\mathbf{x}) = \int_\Omega L_s(\mathbf{x}, \vec{\omega})(\vec{n} \cdot \vec{\omega})\, d\vec{\omega}, \text{ and} \tag{5}$$

$$E_m(\mathbf{x}) = \int_\Omega L_m(\mathbf{x}, \vec{\omega})(\vec{n} \cdot \vec{\omega})\, d\vec{\omega}, \tag{6}$$

and $E(\mathbf{x}) = E_s(\mathbf{x}) + E_m(\mathbf{x})$. The total gradient is simply $\nabla E = \nabla E_s + \nabla E_m$. We derive gradients for the irradiance from surfaces in Section 4 and for irradiance from participating media in Section 5. We illustrate this process in Figure 2.

Note that in the following sections we only consider irradiance gradients for brevity, but all of these computations could easily be projected onto spherical harmonics to obtain full radiance gradients [KGBP05, JDZJ08]. Also, for clarity, we do not include media emission in our derivations. This can be trivially included since the emission and inscattered terms are nearly identical. We also restrict our derivations to translational gradients. We account for gradients on curved surfaces by including a rotational gradient computed as described by Ward and Heckbert [WH92]. Unlike the translational gradient, participating media does not influence the computation of the rotational gradient.

4. Irradiance Gradients from Surfaces

In this section we present a generalization of Ward and Heckbert's [WH92] and Křivánek's [KGBP05] irradiance gradients. Our generalization relaxes the restriction that radiance arriving at $\mathbf{x}$ from an indirect light source at $\mathbf{x}'$ is constant under translation of $\mathbf{x}$. This generalization allows us to correctly handle scenes where the radiance may change due either to absorption by participating media or glossy reflectors.

4.1. Irradiance Gradients

In the irradiance gradient computations of Ward and Heckbert [WH92] and Křivánek et al. [KGBP05] the irradiance integral from Equation 5 is estimated using stratified Monte Carlo integration,

$$E_s(\mathbf{x}) \approx \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} A_{j,k}\, L_s(\mathbf{x}, \vec{\omega}_{j,k})\, (\vec{n} \cdot \vec{\omega}_{j,k}), \tag{7}$$

where we define the relevant quantities in Table 2 and illustrate the stratified geometry in Figure 1.

The contribution of each sample is the product of the radiance through the cell, $L_s(\mathbf{x}, \vec{\omega}_{j,k})$, and the area of the cell, $A_{j,k}$, and this contribution is weighted by the cosine term $(\vec{n} \cdot \vec{\omega}_{j,k})$. The irradiance gradient considers how these terms change when translating the center of projection by an infinitesimal amount. By differentiating Equation 7 we get:

$$\begin{aligned} \nabla E_s(\mathbf{x}) &\approx \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} \nabla (A_{j,k}\, L^s_{j,k}\, (\vec{n} \cdot \vec{\omega}_{j,k})) \\ &= \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} (\nabla A_{j,k}\, L^s_{j,k} + A_{j,k}\, \nabla L^s_{j,k})\, (\vec{n} \cdot \vec{\omega}_{j,k}), \end{aligned} \tag{8}$$

where we have used $L^s_{j,k} = L_s(\mathbf{x}, \vec{\omega}_{j,k})$ for brevity. Translation in the tangent plane may induce both a change in the area of a cell $\nabla A_{j,k}$ and a change in the radiance seen through a cell $\nabla L^s_{j,k}$. Note that the cosine term does not change under translation, so the $\nabla(\vec{n} \cdot \vec{\omega}_{j,k})$ term drops out.

The total surface irradiance gradient can be expressed as a sum of two terms,

$$\nabla E_s(\mathbf{x}) \approx \nabla_A E_s(\mathbf{x}) + \nabla_L E_s(\mathbf{x}), \tag{9}$$

where $\nabla_A E_s(\mathbf{x})$ includes the gradient terms containing changing cell area $\nabla A_{j,k}$, and $\nabla_L E_s(\mathbf{x})$ incorporates the terms with changing cell radiance $\nabla L^s_{j,k}$. Previous work on irradiance and radiance gradients derived expressions for $\nabla_A E_s(\mathbf{x})$, which we review in Section 4.2. Our contribution is that we consider $\nabla_L E_s(\mathbf{x})$, which we derive in Section 4.3.

4.2. Gradient of Cell Area

Ward and Heckbert's and Křivánek's formulations consider the rate of change of each cell area, $\nabla A_{j,k}$, but assume that $\nabla L^s_{j,k} = 0$. Hence, these formulations compute the irradiance gradient as

$$\nabla_A E_s(\mathbf{x}) \approx \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} \nabla A_{j,k}\, L^s_{j,k}\, (\vec{n} \cdot \vec{\omega}_{j,k}). \tag{10}$$

An illustration of this in 2D is shown in Figure 2(a,b,c).

Ward and Heckbert express the gradient of each cell area $A_{j,k}$ in terms of directional derivatives with respect to two ba-

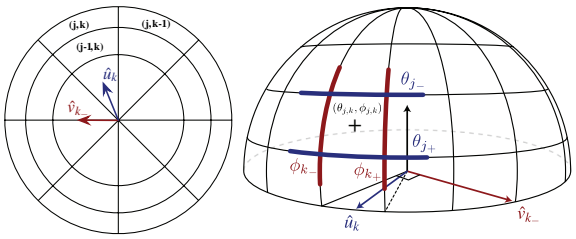

Figure 1: The stratified geometry used in the Ward and Heckbert [WH92] gradient computation illustrated from above (left) and the side (right).

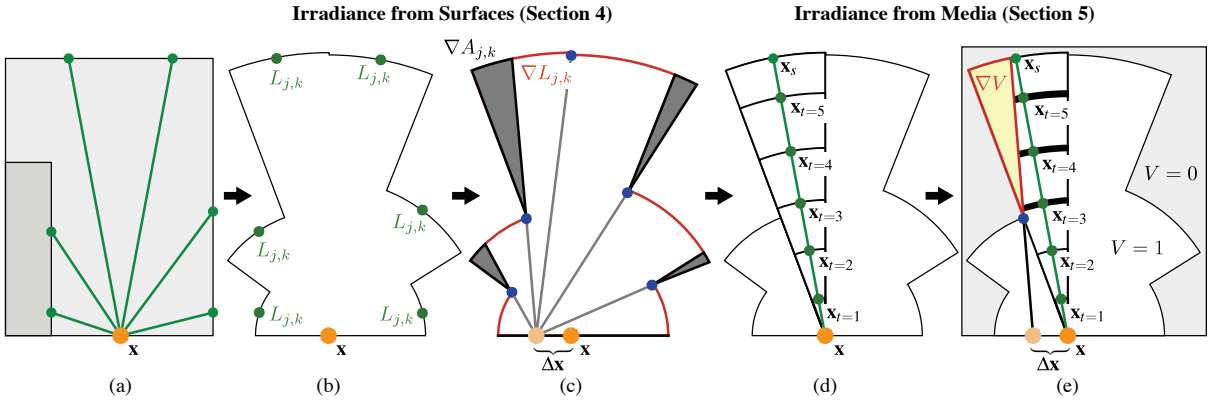

Figure 2: Irradiance caching approximates irradiance by tracing rays in a hemisphere (a) and calculating the radiance and distance to each hitpoint (b). Ward and Heckbert express irradiance gradients (c) by computing $\nabla A_{j,k}$ (grey) of each cell under a translation of $\mathbf{x}$, where the rate of motion of a cell boundary is determined by the distance (blue) to the closer surface. Our method accounts for absorbing media by additionally considering $\nabla L_{j,k}$ (red). We use ray marching (d) by sampling the inscattered radiance at discrete points $\mathbf{x}_t$. For each discrete "shell" we compute both a radiance gradient and a visibility gradient (e) to account for surfaces occluding the medium. Conceptually, ∇V reduces the contribution of radiance from shells (highlighted) beyond the occluder (blue).

sis vectors $\hat{u}_k$ and $\hat{v}_{k_-}$ in the tangent plane. Their final gradient expands to

$$\nabla_A E_s(\mathbf{x}) \approx \sum_{k=0}^{N-1} \left(\hat{u}_k \sum_{j=1}^{M-1} \nabla_{\hat{u}_k} A_{j,k} (L^s_{j,k} - L^s_{j\text{-}1,k}) \cos\theta_{j_-} + \hat{v}_{k_-} \sum_{j=0}^{M-1} \nabla_{\hat{v}_{k_-}} A_{j,k} (L^s_{j,k} - L^s_{j,k\text{-}1}) \cos\theta_j \right), \quad (11)$$

where,

$$\nabla_{\hat{u}_k} A_{j,k} = \frac{(\phi_{k_+} - \phi_{k_-}) \cos\theta_{j_-} \sin\theta_{j_-}}{\min(r_{j,k}, r_{j\text{-}1,k})}, \quad (12)$$

$$\nabla_{\hat{v}_{k_-}} A_{j,k} = \frac{\cos\theta_{j_-} - \cos\theta_{j_+}}{\sin\theta_{j,k} \min(r_{j,k}, r_{j,k\text{-}1})}. \quad (13)$$

The relevant quantities are defined in Table 2 and illustrated in Figure 1. We refer the interested reader to the original papers for a detailed derivation of this expression [WH92, KGBP05].

4.3. Gradient of Cell Radiance

By allowing the cell radiances to change with translation, we can include attenuation due to participating media as well as glossy reflectors in the gradients. We additionally compute a gradient $\nabla_L E_s$ that incorporates changes in cell radiance,

$$\nabla_L E_s(\mathbf{x}) \approx \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} A_{j,k} \nabla L^s_{j,k} (\vec{n} \cdot \vec{\omega}_{j,k}), \quad (14)$$

where the area of each cell is

$$A_{j,k} = \int_{\phi_{k_-}}^{\phi_{k_+}} \int_{\theta_{j_-}}^{\theta_{j_+}} \sin\theta \, d\theta \, d\phi, \quad (15)$$

$$= (\cos\theta_{j_-} - \cos\theta_{j_+})(\phi_{k_+} - \phi_{k_-}). \quad (16)$$

For a cosine-weighted distribution, this gradient term reduces to the average of the cell gradients,

$$\nabla_L E_s(\mathbf{x}) \approx \frac{\pi}{MN} \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} \nabla L^s_{j,k}. \quad (17)$$

Using the definition of surface radiance from Equation 3, the gradient $\nabla L^s_{j,k}$ is

$$\nabla L^s_{j,k} = \nabla T_r(\mathbf{x} \leftrightarrow \mathbf{x}_s) L(\mathbf{x}_s, -\vec{\omega}_{j,k}) + T_r(\mathbf{x} \leftrightarrow \mathbf{x}_s) \nabla L(\mathbf{x}_s, -\vec{\omega}_{j,k}). \quad (18)$$

We compute the gradient of the transmittance term using the technique developed by Jarosz et al. [JDZJ08] and described in Appendix A. Though we do not demonstrate this in our results, glossy reflectors can easily be handled in our framework by computing the radiance gradient using the gradient of the BRDF at $\mathbf{x}_s$.

The final irradiance gradient is simply the sum of Ward and Heckbert's original gradient $\nabla_A E_s(\mathbf{x})$ and our additional cell radiance gradient term $\nabla_L E_s(\mathbf{x})$ as expressed in Equation 9. In Figure 3 we demonstrate the impact of adding the gradients of cell radiance. The resulting irradiance gradients are more accurate, and the quality of interpolation in irradiance caching is significantly improved.

5. Irradiance Gradients from Media

In this section we consider the contribution of scattering media to the irradiance gradient. Our derivation is based on a reformulation of the radiance from the medium in Equation 4. Instead of integrating to the closest visible surface, we introduce a visibility function and integrate to infinity. We write the media irradiance as

$$E_m(\mathbf{x}) = \int_\Omega \int_0^\infty T_r(\mathbf{x} \leftrightarrow \mathbf{x}_t) \sigma_s(\mathbf{x}_t) L_i(\mathbf{x}_t, -\vec{\omega}) V(\mathbf{x} \leftrightarrow \mathbf{x}_t) (\vec{n} \cdot \vec{\omega}) \, dt \, d\vec{\omega}, \quad (19)$$

where we ignore the emission terms. The visibility function V returns one if the two arguments are mutually visible, and zero otherwise. If we collect all factors except visibility in a new term $L_V(\mathbf{x}, \mathbf{x}_t, \vec{\omega}) = T_r(\mathbf{x} \leftrightarrow \mathbf{x}_t) \sigma_s(\mathbf{x}_t) L_i(\mathbf{x}_t, -\vec{\omega}) (\vec{n} \cdot \vec{\omega})$, which we call *unshadowed radiance*, the media irradiance

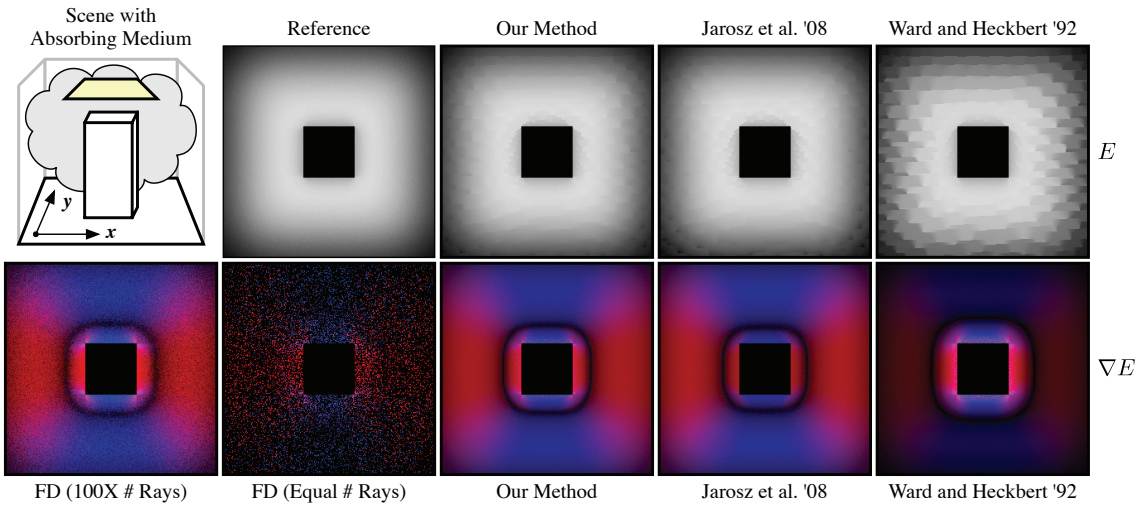

Figure 3: We compare irradiance caching and extrapolation using different gradient computation techniques in a scene with an **absorbing** medium. The top row shows irradiance *extrapolated* over the ground plane from a few cache points. Our gradients consider changes of visibility and absorption by the medium. Previous methods, which ignore either visibility [JDZJ08] or absorption by the medium [WH92], lead to increased artifacts. In the bottom row we visualize the irradiance gradients computed *per-pixel*. The red and blue color coding shows absolute values of the x and y components of the gradient, respectively. We computed gradients for each technique using 1.8K gather rays per pixel and also a "ground truth" gradient using finite differences (FD). Since finite difference gradients are extremely noisy, our reference uses 100 times more rays (180K rays/pixel). Our method produces accurate gradients with few gather rays and converges to the reference solution using finite differencing.

reduces to a product of unshadowed radiance and visibility,

$$E_m(\mathbf{x}) = \int_\Omega \int_0^\infty L_V(\mathbf{x}, \mathbf{x}_t, \vec{\omega}) V(\mathbf{x} \leftrightarrow \mathbf{x}_t)\, dt\, d\vec{\omega}. \qquad (20)$$

Differentiation using the product rule leads to the gradient of media irradiance,

$$\nabla E_m(\mathbf{x}) = \int_\Omega \int_0^\infty \nabla L_V V + L_V \nabla V\, dt\, d\vec{\omega}, \qquad (21)$$

where we have omitted the function arguments for brevity.

Jarosz et al. [JDZJ08] derived gradients of multiple scattering within participating media in a similar manner. In their derivations, however, visibility was assumed constant and the gradient of V was ignored. In effect, they only considered the first term of Equation 21. We use their technique to compute this term and refer to their paper for more details. In addition, we also compute the gradient of visibility as described in the following section.

5.1. Visibility Gradient

In order to account for the visibility gradient, we need to compute the following quantity:

$$\nabla_V E_m(\mathbf{x}) = \int_0^\infty \int_\Omega L_V \nabla V\, d\vec{\omega}\, dt, \qquad (22)$$

$$= \int_0^\infty \tilde{V}_\Omega(\mathbf{x}, t)\, dt, \qquad (23)$$

where we use ∇_V to denote that this is a gradient only of the visibility component, and we swap the order of integration to make the derivation more convenient. Additionally, we introduce the shorthand $\tilde{V}_\Omega$ for the hemispherical integral of $L_V \nabla V$.

If we consider a fixed t, $\tilde{V}_\Omega$ computes the gradient of the weighted visibility integral over the hemisphere. This is similar to Ward and Heckbert, except they compute the gradient of a weighted *radiance* integral over the hemisphere. Therefore, by substituting the radiance function with visibility, and replacing the weighting functions, we can compute this weighted visibility gradient using Ward and Heckbert's formulation. Intuitively, we are computing an irradiance gradient in a scene where the radiance function encodes V as black occluders in front of a distant environment map, L_V. Applying these modifications to Equation 11 results in

$$\tilde{V}_\Omega \approx \sum_{k=0}^{N-1} \left(\hat{u}_k \sum_{j=1}^{M-1} \nabla_{\hat{u}_k} A_{j,k} (V_{j,k} - V_{j\text{-}1,k}) L_V(j_\text{-}, k) + \hat{v}_{k_\text{-}} \sum_{j=0}^{M-1} \nabla_{\hat{v}_{k_\text{-}}} A_{j,k} (V_{j,k} - V_{j,k\text{-}1}) L_V(j, k_\text{-}) \right), \qquad (24)$$

where the directional derivatives $\nabla_{\hat{u}_k} A_{j,k}$ and $\nabla_{\hat{v}_{k_\text{-}}} A_{j,k}$ are computed using Equations 12 and 13 and utilize the distance to occluders in their denominators. $V_{j,k}$ is a binary function indicating for each direction $\vec{\omega}_{j,k}$ whether the distance s to the nearest surface is less than or greater than t:

$$V_{j,k}(t,s) = \begin{cases} 0 & \text{if } t \geq s, \\ 1 & \text{if } t < s. \end{cases} \qquad (25)$$

In order to integrate over t we perform ray marching, which discretizes the medium into "shells" as illustrated in Figure 2(d,e). Conceptually, we consider radiance from the medium as coming from expanding shells of radius t about the evaluation point $\mathbf{x}$ and compute the visibility gradient in

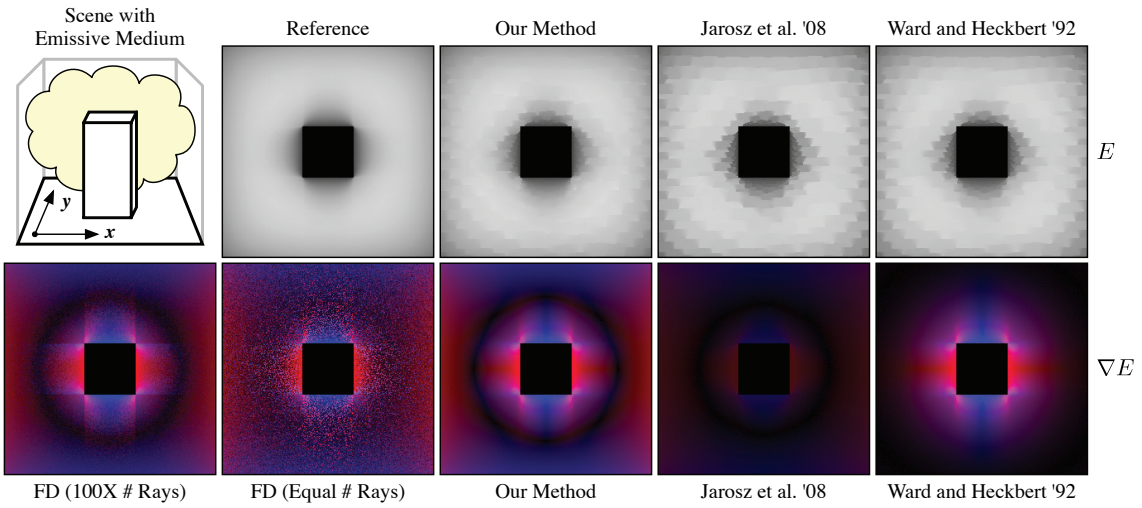

Figure 4: This modification of the scene from Figure 3 contains an **emissive** medium. Our gradient formulation obtains accurate results by taking into account the medium and surface-medium occlusions, whereas previous methods ignore these effects and suffer from artifacts.

Equation 23 as

$$\nabla_V E_m(\mathbf{x}) \approx \sum_{t=0}^{S-1} \tilde{V}_\Omega \Delta t. \tag{26}$$

In practice, however, ray marching is performed independently for each hemispherical direction, which swaps the order of the summations back again.

In summary, for a fixed distance t we compute L_V, ∇L_V as in Jarosz et al. [JDZJ08], and we compute a Ward and Heckbert-style gradient for $L_V \nabla V$ by evaluating the unshadowed radiance along visibility boundaries.

6. Implementation

Our new irradiance gradients can easily be added to a Monte Carlo renderer which uses irradiance caching and supports participating media. The irradiance caching algorithm stays the same, just the gradient computation needs to be modified to account for the medium. Our procedure is illustrated in Figure 2. We provide pseudo-code for computing E_s and E_m and the gradients ∇E_s and ∇E_m in Algorithm 1, and describe the procedure in the following sections.

Initialization. The irradiance and gradient computation starts by creating a $M \times N$ array where each element stores a `HemiSample`. This array is used to store the hemispherical samples needed to compute the irradiance and the irradiance gradient. Each `HemiSample` stores the distance `s`, the surface radiance and gradient `Ls` and `gLs`, and the media radiance and gradient `Lm` and `gLm`. All these values are initialized to zero.

Sampling Surfaces. The algorithm starts by looping over each cell (j,k) and tracing a ray in the $\vec{\omega}_{j,k}$ direction, generated using a cosine-weighted distribution [WH92]. We save the hit distance to the nearest surface, `s`, within the `HemiSample` and store the radiance $L(\mathbf{x}_s, -\vec{\omega}_{j,k})$ in `Ls`. If the intersected surface at $\mathbf{x}_s$ is glossy we also compute a gradient $\nabla L(\mathbf{x}_s, -\vec{\omega}_{j,k})$ and store this in `gLs`. At the end of this stage we have a full hemispherical discretization of the scene, with a radiance, gradient, and distance to the nearest surface in each cell. The cell radiances and gradients do not yet take into account participating media.

Algorithm 1 Calculate E_s, E_m, ∇E_s, and ∇E_m

for each cell (j,k) **do**
  Determine distance `s` by tracing ray towards $\vec{\omega}_{j,k}$
  Compute $L(\mathbf{x}_s, -\vec{\omega}_{j,k})$ and $\nabla L(\mathbf{x}_s, -\vec{\omega}_{j,k})$
  `Ls(j,k)` $= L(\mathbf{x}_s, -\vec{\omega}_{j,k})$
  `gLs(j,k)` $= \nabla L(\mathbf{x}_s, -\vec{\omega}_{j,k})$
end for
$A \Leftarrow \frac{\pi}{M \cdot N}$
for each cell (j,k) **do**
  $T_r \Leftarrow 1$
  $\nabla T_r \Leftarrow 0$
  for each ray-marching step through the medium **do**
    Update T_r and ∇T_r
    Compute L_V, and ∇L_V as in Jarosz et al. [JDZJ08]
    Compute $L_V \nabla V$ using Equation 24
    `Lm(j,k)` $+= L_V A \Delta t$
    `gLm(j,k)` $+= (L_V \nabla V + \nabla L_V A)\Delta t$
  end for
  `gLs(j,k)` $=$ `gLs(j,k)` $T_r +$ `Ls(j,k)` ∇T_r
  `Ls(j,k)` $*= T_r$
end for
$E_s = A \sum_{j,k}$ `Ls(j,k)`
$\nabla E_s =$ Equation 11 $+ A \sum_{j,k}$ `gLs(j,k)`
$E_m = \sum_{j,k}$ `Lm(j,k)`
$\nabla E_m = \sum_{j,k}$ `gLm(j,k)`

Sampling Media. In order to account for the media, we perform ray marching individually for each cell (j,k). As mentioned previously in Section 5, the integration over the hemisphere is the outer loop. For each cell, at each step in the medium we compute $L_V(\mathbf{x}_t)$ and $\nabla L_V(\mathbf{x}_t)$ using the techniques developed by Jarosz et al. [JDZJ08]. We also

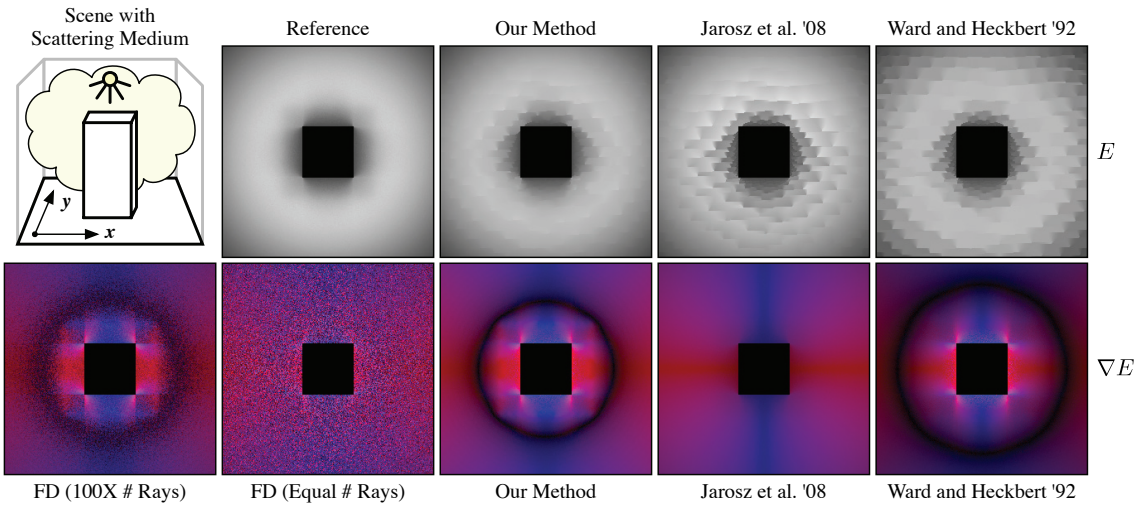

Figure 5: This modification of the scene from Figure 3 contains a spot light and a **scattering** medium. Our gradient formulation obtains accurate results by taking into account the medium and surface-medium occlusions, whereas previous methods ignore these effects.

compute $L_V(\mathbf{x}_t)\nabla V(\mathbf{x}_t)$ by evaluating Equation 24 for the (up to) 4 boundaries of the current cell: (j_-,k_-), (j_-,k_+), (j_+,k_-), and (j_+,k_+). We then multiply the terms by the cell area $A_{j,k}$ and accumulate their contributions into `Lm` and `gLm`.

At each step through the medium we maintain the current value of the transmittance T_r and its gradient ∇T_r. Once ray marching terminates, these values correspond to the transmittance $T_r(\mathbf{x}\leftrightarrow\mathbf{x}_s)$ and transmittance gradient $\nabla T_r(\mathbf{x}\leftrightarrow\mathbf{x}_s)$ from the evaluation location $\mathbf{x}$ to the surface at $\mathbf{x}_s$. We use these transmittance values to augment the `Ls` and `gLs` values computed in the previous stage in order to properly account for absorption by applying Equations 3 and 18.

Integrating. At the end of this process we integrate the irradiance and its gradient by summing over all the `HemiSamples`. Additionally, we add Equation 11 to the surface irradiance gradient to account for the changing cell areas.

7. Results

We implemented our irradiance gradients within a global illumination renderer written in C++. We compare our gradient method to Ward and Heckbert [WH92] for a number of scenes. We also compare to a modification of Jarosz et al.'s [JDZJ08] gradients, which computes irradiance gradients on surfaces by integrating over the hemisphere (instead of the whole sphere) and considering the cosine-weighted BRDF (instead of the phase function). In all of our results comparing irradiance caching using our gradients to other techniques, we only change the gradient method used. All other parameters, including the number of evaluation rays, and the resulting cache point locations are kept exactly the same. All timings are for an Intel Core 2 Duo 2.4 GHz machine using one core.

In Figure 3, 4, and 5 we demonstrate the importance of considering effects from the medium for variations of a scene containing absorbing, emitting, and scattering media, respectively. We directly visualize the x,y gradient components in

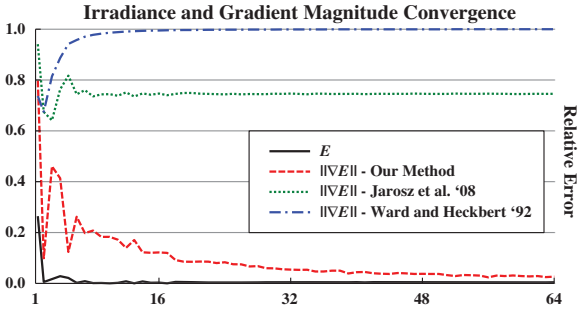

Figure 7: Relative errors for computing the irradiance and irradiance gradient for a single pixel in the scene from Figure 4. The x-axis plots M, the number of divisions in elevation angle, and N is set to $2M$. Previous gradient methods converge to incorrect values, leading to high error even with many samples.

these figures and compare them to "reference" solutions computed using finite differences. These renderings were computed at a resolution of 256×256 using $30\times60=1,800$ rays per evaluation. Our gradients match the reference solutions more faithfully than previous techniques and contain less noise than finite difference gradients computed using 180,000 evaluation rays per pixel. We demonstrate the accuracy of our gradients in Figure 7 by visualizing the convergence rate for the irradiance and gradient for a single pixel from the scene in Figure 4. Previous techniques do not correctly capture the true gradient whereas our technique quickly converges to the correct solution.

Figure 6 features a room with a strong volumetric light beam entering an open window. The walls and most of the floor in this scene are indirectly illuminated by this single beam of light. Using Ward and Heckbert's gradient computation, this scene renders in 3:17 minutes at a horizontal resolution of 1K. Since this gradient method ignores the media, all light is incorrectly assumed to come from surfaces during gradient computation. This results in significant artifacts. Performing an "overture" re-rendering of the image is

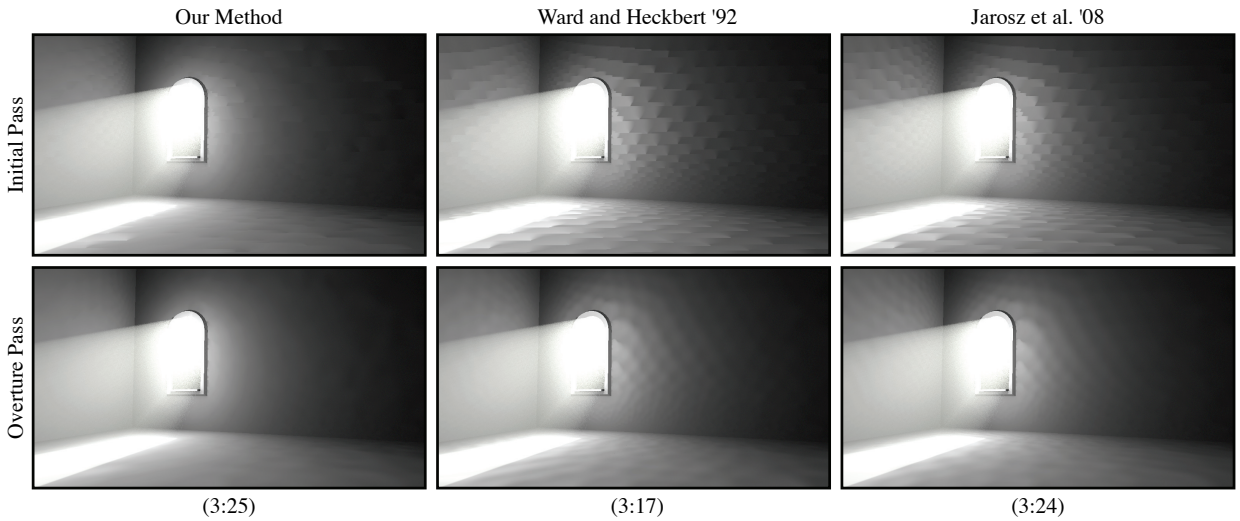

Figure 6: All the illumination on the walls of this room has first scattered off of the floor or the bright beam of light through the window. Our gradient computation significantly reduces extrapolation artifacts (top). Using an "overture" pass, where the scene is re-rendered once all cache points are computed, previous techniques still suffer from interpolation artifacts (bottom) while our method produces smooth reconstruction.

a common technique for improving the quality of irradiance cache renderings. This allows for interpolation, instead of extrapolation, to be used for the entire image. Even though the overture pass improves the quality of the rendering by reducing sudden discontinuities, the inaccurate gradients result in distracting "ripples" on the walls and floor. The gradients from Jarosz et al. do account for the medium but fail to handle visibility changes, which also leads to significant artifacts. Our method, on the other hand, can correctly handle this scene and obtains noticeably smoother reconstruction of the indirect illumination. The overhead of our gradients is fairly negligible, and our method is able to render the same image in 3:25 minutes. The overture pass takes a fraction of the total render time for each method, and completes in under 4 seconds.

Figure 8 contains a modification of the classic Cornell box rendered at a resolution of 1K. Ward's method produces distracting artifacts even after an overture pass. Gradients computed using the techniques from Jarosz et al. [JDZJ08] do a much better job in this scene, but still contain artifact in areas with occlusions changes. Our method produces smooth results even in the initial pass. The overture pass takes less than 5 seconds for each method.

The disco light scene in Figure 9 presents a particular challenge for conventional gradient methods since all lighting on surfaces is indirect lighting from the scattering medium. Due to this, Ward and Heckbert's gradients suffer from significant artifacts both on the surface of the sphere and the ground plane. The gradients by Jarosz et al., since they ignore visibility changes, also suffer from artifacts. This is particularly noticeable on the ground near the base of the disco light where visibility changes are most prominent. Our method is able to obtain a smooth reconstruction of the irradiance on the first pass and can render this scene at 1K resolution in 10:33 minutes. This is only a slight overhead on top of the 10:30 minute render time using Ward and Heckbert's gradients. The overture pass takes less than 7 seconds to compute for each method.

8. Discussion and Future Work

Our work on computing accurate irradiance gradients in scenes containing participating media exposes a number of limitations of current techniques and in the process, suggests several exciting possibilities for future work.

Error Metric. In this paper we were only concerned with improving the quality of interpolation by computing more accurate gradients. However, another significant contributor to the efficiency and quality of the irradiance caching method is the error metric used to compute valid radii of cache points. The split-sphere model which drives the error metric is geometry-driven and completely ignores lighting and all effects from participating media. This can lead to suboptimal cache point distributions. A more general error metric is desirable, but it is not immediately clear how to extend the split-sphere model to incorporate these effects. Jarosz et al. [JDZJ08] derived an error metric specifically for participating media, and Křivánek et al. [KBPv06] introduced a neighbor clamping technique. Creating a robust error metric that takes into account the local geometry as well as lighting from surfaces and media is a difficult problem which warrants further work.

Radiance Gradients. Though we presented all of our derivations within the context of irradiance, it is trivial to apply our approach to radiance caching on surfaces. Křivánek [KGBP05] showed that, to compute radiance gradients, the cosine terms in Equations 7 and 8 can be replaced by a set of basis functions. This projection enables efficiently storing the full radiance function as a vector of coefficients and the radiance gradient as a corresponding set of gradient coefficients.

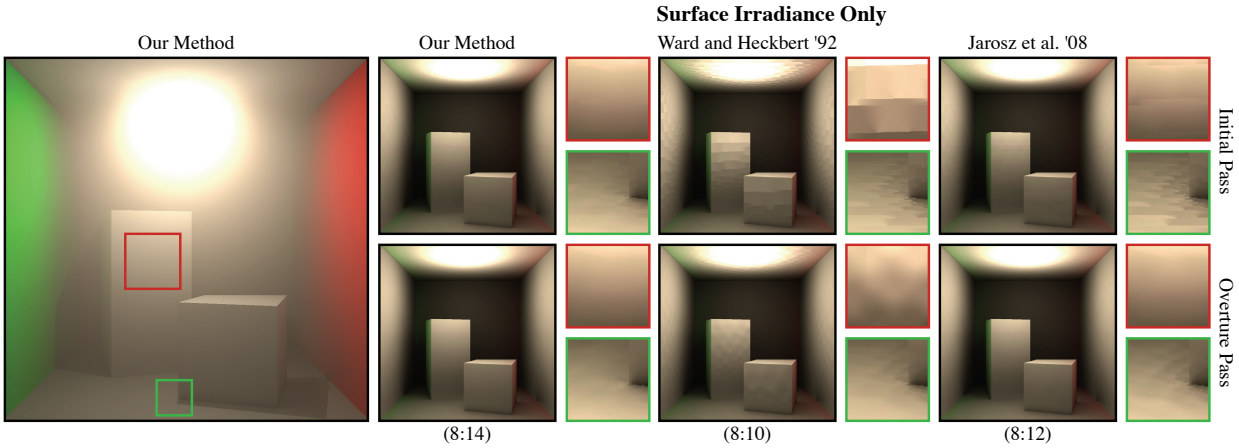

Figure 8: The classic Cornell box scene with a scattering medium. We compare the quality of irradiance caching and provide two zoomed in regions for each result. Ward's gradient formulation does not consider the medium, which results in inaccurate gradients and significant extrapolation artifacts. The participating media gradients from Jarosz et al. do a much better job, but still suffer from artifacts due to occlusion changes. Our gradients obtain the smoothest results by taking into account the media and occlusions.

Radiance Gradients in Participating Media. Jarosz et al. [JDZJ08] compute radiance gradients of single scattering from lights, single scattering from surfaces, and multiple scattering, but ignore visibility in all of these computations. Our work provides the first step in computing visibility-aware radiance gradients for caching *within* participating media.

Computing single scattering from surfaces is very similar to the surface irradiance presented in Section 4. Ward and Heckbert, however, only consider gradients in 2D along the tangent plane. Our surface irradiance gradients could be extended to work in participating media by additionally deriving expressions for the gradients of cell area with respect to motion along a third axis.

Similarly, the definition of media irradiance in Equation 19 is nearly identical to the computation of multiple scattering. Extending Ward and Heckbert's stratified gradients to consider motion along all three dimensions would also enable the use of a visibility gradient within the multiple scattering gradient.

Gradients of single scattering from light sources cannot be handled using Ward and Heckbert's stratified gradients since these effects are typically not computed using hemispherical integration. For point light sources, however, visibility gradients can be efficiently approximated without tracing additional rays by using shadow maps and performing finite differencing. We in fact implemented this extension and use it when computing the gradient of single-scattered radiance embedded within the ∇L_V term of Equation 21.

Visibility Gradient. Ward and Heckbert's stratified gradient formulation is intuitive and works well in practice; however, it is difficult to quantify its mathematical correctness since the gradient is performed after discretization. More recently, Ramamoorthi et al. [RMB07] presented an elegant new visibility gradient formulation. Their approach is more mathematically rigorous since it presents an analytic expression for the gradients, and only then performs discretization of this analytic expression. In Section 5.1 we modify Ward and Heckbert's gradients to estimate visibility gradients; however, an obvious alternative to this approach would be to directly use the gradients presented by Ramamoorthi et al. This approach would allow for a completely analytic expression for the media irradiance gradient. We explored this avenue, but Ramamoorthi's visibility integration suffers from a weak singularity. Though this singularity can be avoided by using adaptive sampling over the hemisphere, it makes it more cumbersome to integrate into the stratified ray marching process needed within our context. Nevertheless, we still believe this is a promising alternative and plan to investigate this approach further.

9. Conclusion

In this paper we presented a method for accurately computing irradiance gradients on surfaces in the presence of participating media and occlusions. We applied our gradient derivations to the irradiance caching method and demonstrated that incorporating participating media and visibility information within the gradient is important for high quality irradiance interpolation in scenes containing these effects.

10. Acknowledgments

This work was supported in part by NSF grant CPA 0701992 and the UCSD FWGrid Project, NSF Research Infrastructure Grant Number EIA-0303622.

A. Transmittance Gradient

The gradient of transmittance is needed for computing the cell radiance gradient in Section 4. Jarosz et al. [JDZJ08] derived this gradient, and we include it here for reference.

The transmittance T_r is defined as

$$T_r(\mathbf{x} \leftrightarrow \mathbf{x}') = e^{-\tau(\mathbf{x} \leftrightarrow \mathbf{x}')}, \tag{27}$$

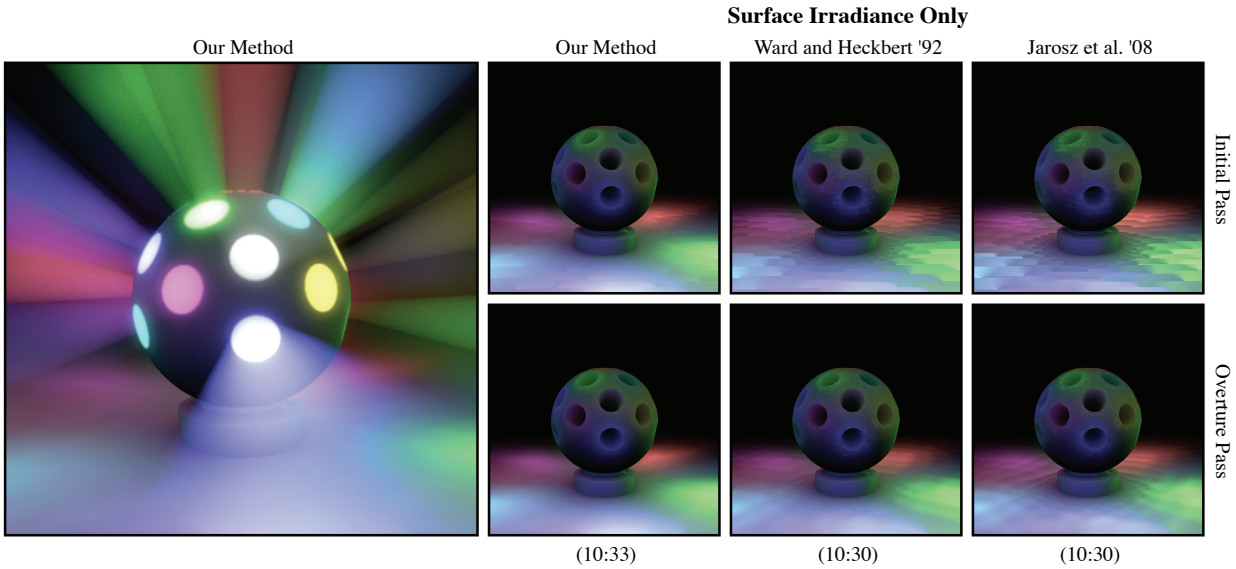

Figure 9: This disco light scene contains 21 light sources, but all lighting on surfaces is indirect. Ward's gradient formulation assumes all lighting (even from the media) arrives from surfaces which results in inaccurate gradients and significant extrapolation artifacts. The participating media gradients from Jarosz et al., however, do not take the visibility term into account. Our gradients obtain the smoothest results by taking into account the media and occlusions.

and the gradient of the transmittance is

$$\nabla T_r(\mathbf{x}\leftrightarrow\mathbf{x}') = -\nabla\tau(\mathbf{x}\leftrightarrow\mathbf{x}')T_r(\mathbf{x}\leftrightarrow\mathbf{x}'). \tag{28}$$

In the general case of heterogeneous media, the optical thickness τ can be expressed as:

$$\tau(\mathbf{x}\leftrightarrow\mathbf{x}') = \|\mathbf{x}-\mathbf{x}'\|\overline{\sigma}_t(\mathbf{x}\leftrightarrow\mathbf{x}'), \tag{29}$$

where $\overline{\sigma}_t = \int_0^1 \sigma_t(\mathbf{y}(t))\,dt$ is the average extinction over the line segment between $\mathbf{x}$ and $\mathbf{x}'$, which we have parameterized as $\mathbf{y}(t) = \mathbf{x}+t(\mathbf{x}'-\mathbf{x})$. The corresponding gradient is

$$\nabla\tau(\mathbf{x}\leftrightarrow\mathbf{x}') = \nabla(\|\mathbf{x}-\mathbf{x}'\|)\overline{\sigma}_t(\mathbf{x}\leftrightarrow\mathbf{x}')+ \|\mathbf{x}-\mathbf{x}'\|\nabla(\overline{\sigma}_t(\mathbf{x}\leftrightarrow\mathbf{x}')). \tag{30}$$

A more detailed explanation is available in Jarosz et al. [JDZJ08].

References

[AKDS04] ANNEN T., KAUTZ J., DURAND F., SEIDEL H.-P.: Spherical Harmonic Gradients for Mid-Range Illumination. In *Rendering Techniques* (June 2004), Keller A., Jensen H. W., (Eds.), Eurographics Association. 2

[Arv94] ARVO J.: The Irradiance Jacobian for Partially Occluded Polyhedral Sources. In *SIGGRAPH* (1994). 2

[Cha60] CHANDRASEKHAR S.: *Radiative Transfer*. Dover Publications, New York, 1960. 1, 2

[Gla95] GLASSNER A. S.: *Principles of Digital Image Synthesis*. Morgan Kaufmann, 1995. 1

[JDZJ08] JAROSZ W., DONNER C., ZWICKER M., JENSEN H. W.: Radiance Caching for Participating Media. *ACM Trans. Graph. 27*, 1 (2008), 1–11. 2, 3, 4, 5, 6, 7, 8, 9, 10

[Jen01] JENSEN H. W.: *Realistic Image Synthesis Using Photon Mapping*. A. K. Peters, Ltd., Natick, MA, USA, 2001. 2

[Kaj86] KAJIYA J. T.: The Rendering Equation. In *SIGGRAPH* (New York, NY, USA, 1986), ACM Press, pp. 143–150. 1

[KBPv06] KŘIVÁNEK J., BOUATOUCH K., PATTANAIK S. N., ŽÁRA J.: Making Radiance and Irradiance Caching Practical: Adaptive Caching and Neighbor Clamping. In *Rendering Techniques* (June 2006), Akenine-Möller T., Heidrich W., (Eds.), Eurographics Association. 8

[KGBP05] KŘIVÁNEK J., GAUTRON P., BOUATOUCH K., PATTANAIK S.: Improved Radiance Gradient Computation. In *SCCG '05: Proceedings of the 21st spring conference on Computer graphics* (New York, NY, USA, 2005), ACM Press, pp. 155–159. 2, 3, 4, 8

[KGPB05] KŘIVÁNEK J., GAUTRON P., PATTANAIK S., BOUATOUCH K.: Radiance Caching for Efficient Global Illumination Computation. *IEEE Transactions on Visualization and Computer Graphics 11*, 5 (2005). 2

[RMB07] RAMAMOORTHI R., MAHAJAN D., BELHUMEUR P.: A First Order Analysis of Lighting, Shading, and Shadows. *ACM Transactions on Graphics 26*, 1 (Jan 2007), 2. 2, 9

[WH92] WARD G. J., HECKBERT P.: Irradiance Gradients. In *Third Eurographics Workshop on Rendering* (Bristol, UK, 1992), pp. 85–98. 2, 3, 4, 5, 6, 7

[WRC88] WARD G. J., RUBINSTEIN F. M., CLEAR R. D.: A ray tracing solution for diffuse interreflection. In *SIGGRAPH* (New York, NY, USA, 1988), ACM Press, pp. 85–92. 1, 2