Université de Montréal

Efficient Frequency-Space Methods for Light Transport Caching

par

Renaud Adrien Dubouchet

Département d'Informatique et de Recherche Opérationelle
Faculté des arts et des sciences

Thèse présentée à la Faculté des études supérieures et postdoctorales
en vue de l'obtention du grade de
Philosophiæ Doctor (Ph.D.)
en Informatique

avril 2021

Université de Montréal

Faculté des études supérieures et postdoctorales

Cette thèse intitulée

Efficient Frequency-Space Methods for Light Transport Caching

présentée par

Renaud Adrien Dubouchet

a été évaluée par un jury composé des personnes suivantes :

Mikhail Besmeltsev
(président-rapporteur)

Derek Nowrouzezahrai
(directeur de recherche)

Christopher Pal
(membre du jury)

Sheldon Andrews
(examinateur externe)

Jacques Bélair
(représentant du doyen de la FESP)

Thèse acceptée le :
10 septembre 2021

Sommaire

Le transport de la lumière permet de simuler physiquement le movement de photons dans un environnement virtuel. En rendu d'images, la lumière se propage une dernière fois vers un capteur virtuel la transformant en une image, affichée pour un observateur. Durant ce voyage la lumière peut être analysée fréquentiellement pour comprendre ses variations spatiales et angulaires afin d'accélerer le rendu. La génération d'images réalistes a subit de grandes avancées au cours des dernières années, réduisant l'écart entre simulation et réalité. Cependant les contraintes en terme de performance et de mémoire empêchent toujours aux applications interactives et en temps-réel de bénéficier des effets de rendu les plus complexes. Pour cela, les moteurs de rendu professionels modernes dépendent toujours de méthodes de pré-calculation de données et de procédures asynchrones de traitement.

Cette thèse par article présente deux projets traitant du transport de la lumière à travers une perspective fréquentielle dans le contexte d'applications interactives et en temps-réel. Nous proposons premièrement une méthode pour réutiliser efficacement le calcul préalable de chemins de lumière par méthode Monte Carlo pour des séquences animées. Nous prenons avantage de l'analyse fréquentielle du transport de la lumière réalisée dans des travaux antérieurs, étendue ici à l'échantillonement et reconstruction spatial, angulaire et temporel. Notre seconde méthode pré-calcule le transport de la lumière à travers les volumes participatifs jusqu'aux surfaces, que nous encodons comme réponse impulsive. Cet opérateur compacte et efficace nous permet d'accélerer le transport à travers des volumes jusqu'aux surfaces dans le contexte de diffusion multiple dans des conditions arbitraires de média participatifs.

Mots-clefs: Rendu réaliste, transport de la lumière, analyse fréquentielle, média participatif, échantillonage adaptatif, filtrage, harmoniques sphériques.

Summary

Light transport is the method of physically simulating the movement of photons in an environment. Applied to rendering, light travels one last time to a virtual sensor that captures it as an image displayed to an observer. As it travels, light is analysable frequentially to understand how it varies spatially and angularly to accelerate rendering. Recent advances in physically-based realistic rendering have been closing the gap between reality and simulation but the memory and performance costs still preclude the inclusion of the more computationally expensive effects in interactive and real-time applications. Because of this, modern production renderers rely on the ahead-of-time precomputation of data for efficient reuse in the form of offline computational processes and asynchronously distributed procedures.

This thesis by publication investigates with two papers the simulation of light transport from a frequency-based perspective for interactive and precomputed real-time applications. We first propose a method for efficiently reusing light path computations over time in interactive Monte Carlo path-traced animation sequences. We leverage to this end the frequency analysis of light transport introduced in previous works, extended to spatial, angular and temporal sampling and reconstruction. Our second method investigates the precomputation of participating volume-to-surface light transport as impulse responses, a compact and efficient frequency-based transport operator. In turn, these operators accelerate by orders of magnitude the computation of multi-scattered volume-to-surface transport in arbitrary, potentially heterogeneous media conditions.

Keywords: Rendering, light transport, frequency analysis, participating media, adaptive sampling, filtering, spherical harmonics.

Contents

List of figures

Remerciements

Je voudrais remercier Derek Nowrouzezahrai, mon superviseur de tant d'années. Il me semble que c'était encore hier que j'entrais dans son bureau pour la première fois. Merci pour ta grande patience et calme, mais aussi ton attention, discussions et explications, dérivations mathématiques au tableau. Merci de m'avoir poussé de l'avant tant de fois.

Je remercie ma famille, qui a toujours été là et le sera toujours. Mon père Jean-Michel et ma mère Laurence. Mes frères Pierre et Matthias. Mes grands-parents Michel et Marie-Thérèse. Mes oncles, tantes et cousines Cécile, Valérie, Marc, Marie, Claire, Bruno, Gilles, Karine.

Je remercie ma petite amie Oleksandra, qui m'a poussé à terminer lorsque je n'y pensais plus et qui est là tous les jours avec moi.

Je remercie mes amis et collègues, parce que lorsqu'on ne travaille pas, il faut bien rigoler. Ceux de la Côte Est: Arnaud, Cihan, Jonas, Zofia, Atyab, Madio, Mauricio. Ceux de la Côte-Ouest: Aleksandra, Andrew, Jeremy, Michal, Josiah, Ari. Les membres du Ligum: Laurent Belcour pour sa patience et entrain, Olivier, Luis, Melino, Aude, Alexandre, Gilles-Philippe, Joël, Chaitanya, Bruno, Jean-Philippe, Yangyang, David, Dabid, Cynthia, Antoine, Nicolas, Etienne, Sonia, Jean, Jonathan, Eric, et Pierre Poulin pour avoir été là du début à la fin.

Finalement je voudrais remercier Peter-Pike Sloan pour sa générosité et présence, ses connaissances sans fin et sa dédication.

Chapter 1

Introduction

Computer graphics – or the computer-assisted generation of images – is an endeavor with a decades-long history which product has become omnipresent. From line-drawing on CRT displays to interactive virtual reality and photorealistic rendering, the field has evolved drastically with the work of brilliant researchers, engineers and artists, bringing computer-generated imagery to every devices and to varied disciplines ranging from the medical to the entertainment industry.

Intersecting many fields, computer graphics relies on geometry, optics and physics among others, is tightly tied with the creative and artistic processes, and evolves alongside the hardware that allows its execution. The advances made in computing power have increased our ability to experiment with physical models of light transport, geometric processing and material synthesis, which in turn fuels always more efficient algorithms of rendering and visualization.

Rendering – which has jokingly been described as "extreme data compression" with the production of a final image as an advantageous side effect – is defined as the process of generating synthetic images from a descriptive model, or *scene* – a set of virtual lights, cameras, physical models, geometry and associated materials. The central problem that rendering seeks to solve is to generate the colors of the objects that an observer sees.

Ubiquitous to video games, which made their way into the mainstream over the past two decades, 3D CG has equally revolutionized film making. Computer-generated imagery(CGI) now dominates the field of modern visual effects. The resulting *photorealism* – or drive to imitate reality on artistic media – in many instances is enough to fool our visual senses: virtual objects and physical phenomena are seamlessly embedded in photographic shots and virtual worlds of increased geometric and lighting complexity, running over large-scale physical simulation models like ocean water and weather effects.

Death Stranding (©2019 Sony)

Blade Runner 2049 (©2017 Warner Bros.)

Red Dead Redemption 2 (©2018 Rockstar Studios)

Fig. 1. In recent years, CGI quality has constantly evolved to build realistic worlds. Movies (top-right) have always had lower frame time constraints and have increased the amount and quality of special effects. In contrast, video-games (top-left, bottom) have quickly caught up, exposing breathtaking worlds for players to explore in real-time.

More recently, *physically based rendering*(PBR) [**62, 17**] has aimed for models and algorithms more anchored in the underlying physical principles of light transport and material appearance, using strict regulations on measures and energy conservation that in turn consolidate rendering pipelines and data sets. Always striving for efficiency, *offline* CG animation rendering processes – which have the ability to last for hours or days for a frame – have now benefited for a half a decade by the transition, boasting complex *global illumination* effects of photogrammetric materials and geometry; while *interactive* or *realtime* applications such as games still heavily rely on *precomputation* methods to display the most expensive effects to their players. Thus many challenges remain and advances often take the shape of trade offs between performance, quality and interactivity.

Precomputation is one such method to offload data processing in an asynchronous way ahead of – or amortized over – time. Being such a general term, it takes many form such as pre-constructed hierarchical spatial data structures for scene management (e.g., for accelerating ray-tracing), temporal data caches allowing an algorithm to carry information over a sequence of frames, or light transport operators, that register in advance the physical behavior of light over surfaces and volumes for the purpose of reuse under varied illuminations. These methods, though ranging in use cases, are most often based on presumptive spatio-angular knowledge of the problem to be solved, such as – respectively to the above examples – the geometrical topology of a scene, the motion vectors of objects, and the reflectance of a material. Such precomputed cache designs are valid as long as the presuppositions made during their generation continue to hold true. This often is the case and exceptions to the rule are usually treated separately with specialized costlier solutions.

Ray-tracing has been the tried and true method of estimating the behavior of light in a virtual environment and an intuitive solution for light transport. Travelling at the speed of light, photon particles scatter around us until their absorption by matter. In ray-tracing, this behavior is abstracted by a segment between two locations, or ray originating from a position in a given direction. A sensor such as a camera lens or the human eye may be at the receiving end of such paths of light, which integrates contributing results as visible colors. What defines which color is seen or captured depends on various physical variables such as the power of the light source, the sensor mechanism and the many particles of matter the light has interacted with before reaching that destination. This multitude of variables and the infinite-dimensional nature of lighting interactions makes path-tracing and light transport vibrant fields of research.

Described above as scattering photon particles, light is more thoroughly defined as electromagnetic radiation over a wavelength spectrum. *Frequency analysis* of light transport aims to better understand the intricate behavior of light by examining the changes in its frequency spectrum as it travels and interacts with matter. This is an interesting topic, in that it can be used not only as a tool to retrieve prior knowledge about the complexity of the light integrand at any particular location, and thus the adaptive amount of work needed to solve it; but also as a mechanism for the encoding of lighting signals in the context of illumination caching and reuse. The frequency content of the lighting signal may thus be bounded as a mean of defining the variations within the domains of integration along its path, or to adequately adjust lighting proxy representations for storage and to reduce the processing cost of rendering algorithms.

This thesis presents two examples of rendering techniques solving complex light transport scenarios for interactive applications. The first work proposes an adaptive ray-traced shading computation amortized over time for the purpose of rendering interactive animations. The second work presents a novel perspective on precomputing the influence of participating media in light transport for real-time rendering. Both works study the behavior of light transport with the help of frequency-space based analyses and offer practical applications via the use of cache systems.

1.1. Summary of Contributions

The main contributions of this thesis include

Frequency Analysis of Sample Placement and Reuse. Chapter 1.

- We extend the frequency analysis of light transport by Durand [**23**] towards sample placement and reconstruction across animation frames, amortizing shading cost in the spatial, temporal and integration domains.
- We introduce an adaptive integration strategy taking cues from our frequency analysis based on material, lighting and visibility bandwidths.
- We present a lightweight sample caching framework for adaptively discard, reuse and recompute shading and occlusion over time according to our frequency criteria.

Spherical Impulse Responses of Scattered Light. Chapter 2.

- We propose a novel formulation and analysis of the impulse response of inscattered light in participating media for point, directional and differential area emitters.
- We extend the analysis to treat multiple scattering, heterogeneous media and anisotropic phase functions.
- We propose a compact, precomputed-once tabular Zonal Harmonics parameterization of our impulse responses to quickly apply volume-to-surface transport to precomputed lightmap, vertex and probe targets, and implemented in a modern AAA video game.

1.2. Organization of this Document

This thesis is divided in four chapters. The remainder of the introduction delves into the theoretical background material and state of the art in rendering related to the topics presented above and helful to the comprehension of the works exposed in later chapters. Section 1.3 introduces the fundamentals of light transport in vacuum, based on geometrical optics. Section 1.4 supplements these fundamentals to account for the presence of volumetric media and describes the various interactions between light and thin particulate mater. Section 1.5 explores the different manners in which frequency analysis is used as a mathematical tool to further study the behavior of light and its interactions in an environment. Section 1.6 presents various storage schemes that are used to cache illumination for further reuse, spatially and/or temporally. The following chapters present our published works. In Chapter 1 we present our first contribution, a ray tracing framework leveraging a frequency analysis of spatio-temporal sampling and reconstruction and using sample caching to preserve temporal coherency in the context of rendering animations. We describe in Chapter 2 a method to represent volume-to-surface transport during lighting precomputation built upon a novel model of the impulse response of scattered light: an efficient and compact representation of the influence of participating media in light transport. Finally, we conclude in the last Chapter by outlining our contributions and examine further avenues of research.

1.3. Fundamentals of Light Transport

Rendering considers the spatio-temporal behavior of light – or *light transport* – and how it interacts with matter before ultimately reaching a sensor (e.g., mechanical camera or biological eye), at which point an image resulting from the integration of the incoming light is presented to the observer.

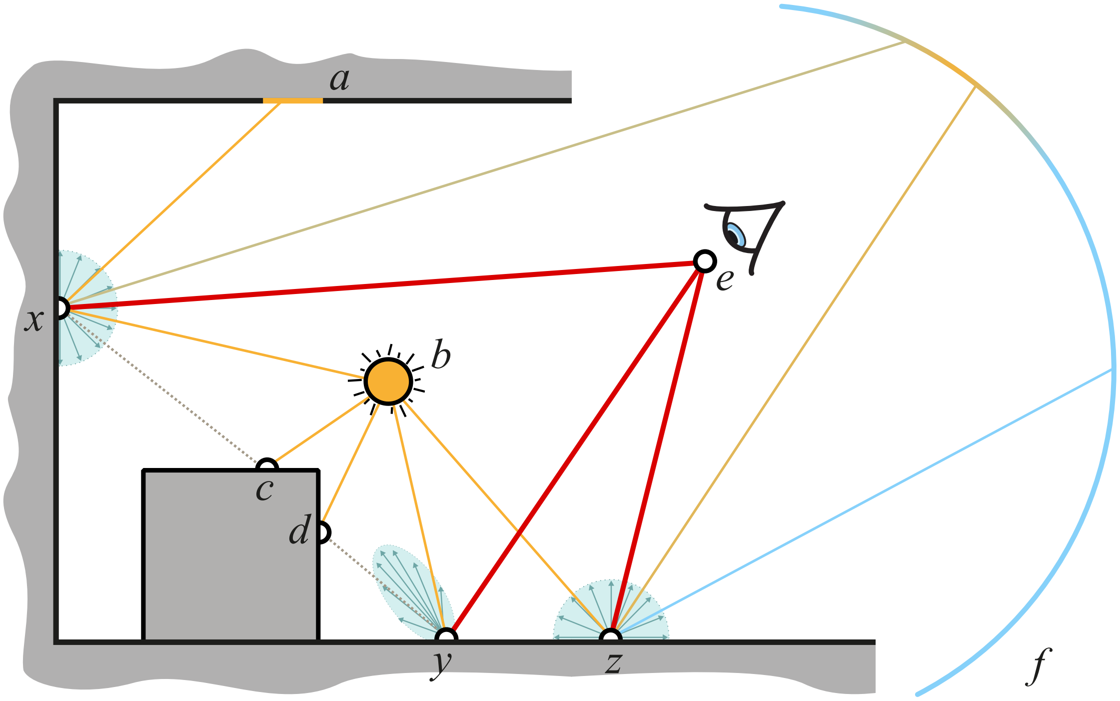


Fig. 2. A scene illustrating light transport from the point of view of an observer at point e. The scene is illuminated by three light sources, an area light (a), a point light (b) and an image-based environment light (f). Primary rays (in red) are cast from e and intersect the scene. Surfaces points x and z have a lambertian diffuse reflectance and y has a glossy reflectance. Thin solid colored lines denote direct lighting, where a direct connection is formed between a surface point and a light source, while dotted lines represent indirect lighting from other surface points a and b, which must themselves in turn be lit by direct and indirect light where the same process applies.

These countless interactions are studied within the theory of light transport as series of local modulations of an incident lighting signal. Understanding these local procedures is key to model light transport for the purpose of rendering. Different mathematical models help us understand these operations at different sections along the path of light, such as models of light emission, reflection or refraction on or through a surface, occlusion and integration. Furthermore, several models of transport exist when reasoning about light. The most common, geometrical optics, makes the assumption of light as photon particles traveling in a piecewise-linear manner, that is in straight lines (or rays) until an interaction with matter occur. This simple premise is the basis of path tracing algorithms, and the one we focus on in this dissertation. Other models include wave optics which additionally considers diffraction effects and quantum optics, which further widen the range of describable effects (e.g., phosphorescence). Having the benefit of targeting the limited human visual system, rendering gains from using the former, simpler one, thus producing photorealistic images at a reasonable cost. We make further assumptions for this section including the restriction in analysis to light at – or before – it reaches the sensor, and thus won't detail camera effects and settings; and model light as traveling in vacuum, without the presence of participating media, which is covered in Section 1.4. Light Transport is a comprehensive field of study, which in computer graphics was exposed through *Monte Carlo* path tracing (see Section 1.3.4) with the *rendering equation* [**41**] (see Section 1.3.3), based on previous ray tracing efforts [**2, 91, 20**], and subsequently extended [**4, 84**]. Before expanding on these concepts, we expose fundamentals of light emission in Section 1.3.1 and scattering in Section 1.3.2.

1.3.1. Light Emission

Light is measured with various radiometric quantities, with the wavelength-dependent photon $e_\lambda = hc/\lambda$ being the basic indivisible unit of radiometry, measured in watts (or Joules per second) $[W \equiv J \cdot s]$, where h is the Planck constant, c is the speed of light and λ is the particular wavelength. Light is quantified for a particular light wavelength by the amount of such photons, called the *radiant power* Φ_λ.

The radiance L defines the light received by a surface from a particular direction or emitted from a surface in a particular direction. This outgoing radiance at a point $\mathbf{x}$ in the direction $\boldsymbol{\omega}$ toward an observer is thus a good representation of the visible color of that surface point and is defined as

$$L(\mathbf{x},\boldsymbol{\omega}) = \frac{d^2\Phi}{dA^\perp d\boldsymbol{\omega}},$$

measuring the differential amount of light (power Φ, pre-integrated over the visible spectrum of wavelengths λ) emitted from a differential solid angle $d\boldsymbol{\omega}$ per unit differential projected area $dA^{\perp} \equiv \cos\theta dA$. Radiance is thus measured in watts per sterradian per unit area $[W \cdot sr^{-1} \cdot m^{-2}]$. Integrating radiance over the directional domain yields *irradiance* E, the incident light at a point from all directions, measured in watts per unit area $[W \cdot m^{-2}]$. Also of note is the *radiant intensity* I_{Φ} – measured in watts per steradian $[W \cdot sr^{-1}]$: the radiant power flowing per unit solid angle – as a quantity often used to angularly specified intensities over a spherical domain to model or mask various parts of the emission profile of a light source, such as *IES profiles*.

This allows us to represent physical quantities of radiation over the light spectrum, including radiations in ultraviolet, infrared and the visible range in between. A physically based renderer models radiance over the visible wavelength spectrum (between 380nm and 750nm). Some production renderers [**28**] have begun considering the whole visible spectrum, a process called spectral rendering. The majority though only consider a simpler model restricted to three wavelengths corresponding to a red, green and blue channel, forming the RGB color model based on human perception and the limitations of traditional display devices. Unless stated otherwise, we simplify later equations to abstract wavelength-dependence and assume per-channel solves.

Light sources define in rendering how light is emitted in a virtual scene. In our physical world light is (most often) emanation from converted heat radiation, a complexity that virtual light sources bypass while preserving the physical laws of light transport. They are classified in three categories: punctual, area based and image based light sources.

Punctual Lights. *Point* and *spot* lights are categorized as punctual lights, defined as spatially infinitesimal radially emitting sources. They follow the inverse square law for physical correctness: their intensity decreases proportionally to the square of the distance (or radius r) to the illuminated object, which yields

$$L = \frac{I_{\Phi}}{r^2}.$$

This implies a weak singularity in the limit as the distance tends to 0, which is often time prevented with bias by implying a minimum small radius. In the other direction, toward infinity, punctual lights never truly reach zero radiance and bias by radius bounding is again used for culling optimizations. This is commonly solved by interpolating to zero while taking care of preserving brighter closer radiance values. Point and spot differ by their angular emission profile. Point lights emit in a constant manner in all directions, so that $\Phi = 4\pi I_\Phi$, while spot lights rely on an outer radius θ_{out} defining the cone of emission, with an optional inner radius θ_{in} to model a smooth cone falloff from bright to dim, so that

$$\Phi = \int_\Omega I_\Phi d\boldsymbol{\omega} = \int_0^{2\pi} \int_0^{\theta_{out}} I_\Phi d\theta d\phi = 2\pi I(1 - \cos\theta_{out}/2),$$

while the actual radiance in a certain direction contained in the cone is modulated by a smoothing function depending on the angular ratio of θ_{out} and θ_{in}. Punctual light sources are resolved analytically and their visibility obtained with a single sample, making them a common emitter type.

Area Lights. Also referred to as *emissives*, they are sources with a physical shape, most commonly a triangle mesh representing an object, and more closely fit the description of everyday light sources, such as a light bulb, a neon tube, a tv screen or the sun, without the complex internal physical simulations. Though they follow regular surface-light interactions (described in Section 1.3.2) when lit themselves, the majority of the contribution of area lights come from emission. Additionally to being more relatable to by artists, they also behave more appropriately for rendering algorithms by not having the weak spatial singularity present in punctual lights, which cause both sharp shadows and shading aliasing for specular surfaces.

Area lights possess drawbacks in that they are not necessarily analytically solvable depending on their shape and relative position to the receiver. Additionally, each point to be lit must consider many sampling directions to capture both the spatially-varying lighting and visibility to the area emitter. For precomputation and offline rendering, this is often done with Monte Carlo estimation(see Section 1.3.4), while appealing closed-form solutions have been found for interactive applications [**3, 21, 32, 9**].

Image-based Lights. Based on the concept of directional lighting, which flows in parallel directions and have no spatial dimension, image based lighting(IBL) represents lighting from the complete sphere of directions flowing inward to a central point; and get its name from the textures they rely on to define angular intensities. Applying a 2D texture (or set of) to the sphere of direction is performed effectively via various mappings, such as lattitude-longitude or cubemap, each having advangages and drawbacks, especially concerning discontinuity errors [**49**].

IBL see uses for both distant (or global) and local lighting (usually represented by *light reflection probes*). Distant image based lighting work especially well for modeling global effects such as the sky or unreachable background environment, thus allowing variations over time (e.g., to change the color of the sky as the day sets) but not over space, by definition. For this reason, they are often defined by high resolution high-dynamic range *HDR* images, often captured from real life photographs in locations around the world. The sun (or moon) is often removed from such panoramas to be later modeled either as a directional light source shining equally on the entire scene or as a far away punctual light. Because of its high intensity and small solid angle when viewed from a surface, representing the sun as a disk area-light reduces aliasing artifacts on specular surfaces.

Local light probes similarly model a surrounding environment but at a sparse scene resolution to allow spatial changes in local lighting [**?**]. They are calculated by spherical projection (or multi-faceted prespective projection) of the scene around them. This information is later queried back by static or dynamic objects in the vicinity as a relighting method. Drawbacks from light probe relighting include limited resolution (set by the capture) and reprojection parallax between the probe itself, the source lighting information it captured, and the target to be lit. A spatially dense sampling of probes in a scene offers better interpolation and parallax management at the cost of a higher storage and processing cost. Probes model a statically captured illumination at important locations, and must be recalculated whenever the environment around it changes, e.g., dynamic objects and light sources.

Image-based lighting, whether from high resolution photographs or captured scene environment, are often preprocessed before query through the construction of a *mipmap* hierarchy, defining various levels of roughness between purely specular (i.e., the captured information) and diffuse (see Sections 1.3.2, 1.6) to accelerate queries for a particular solid angle.

1.3.2. Surface Interactions

As described previously, radiance is dependent on the amount of photons of each particular wavelength. Radiance stays invariant until an interaction with matter occur, at which point the physical microscopic and macroscopic characteristics of the surface in contact define the interplay between light and matter. These interactions are described by a *bidirectional* *scattering distribution function*(BSDF) [**55, 59**]. BSDF is an umbrella term for the mathematical class of functions defining the ratio of outgoing radiance to incident irradiance at a surface point and by extension, how light is absorbed, reflected and transmitted by a particular material. The perceived color of a surface is in a large part the result of the wavelength-dependent loss of energy of the incident lighting spectrum contingent to the aspects of the surface (e.g., pigment, geometric features, opaqueness) as it gets redirected toward an observer.

A scattering function f_s is in practice decoupled as the sum of its reflective part, the *bidirectional reflectance distribution function*(BRDF) f_r and its transmittive part, the *bidirectional transmittance reflection function*(BTDF) f_t. We consider the case of the BRDF, in which both the incident and outgoing directions, $\boldsymbol{\omega_i}$ and $\boldsymbol{\omega_o}$ resp., lie in the same hemisphere as defined by the normal direction $\mathbf{n}$ at the surface point $\mathbf{x}$, defined as

$$f_r(\mathbf{x},\boldsymbol{\omega_i},\boldsymbol{\omega_o}) = \frac{dL(\mathbf{x},\boldsymbol{\omega_o})}{dE(\mathbf{x},\boldsymbol{\omega_i})} = \frac{dL(\mathbf{x},\boldsymbol{\omega_o})}{dL(\mathbf{x},\boldsymbol{\omega_i})\cos\theta_i d\boldsymbol{\omega_i}},$$

where $\cos\theta_i = \langle \boldsymbol{\omega_i} \cdot \mathbf{n} \rangle$ is the foreshortening factor due to the tilt of the surface to the incident direction.

In practice, analytical reflectance models are used to model a 4-dimensional BRDF, as opposed to dense data sets measured in reflectrometry [**50, 89**]. As edge reflectance cases, the ideal specular (perfectly smooth) and the ideal diffuse (perfectly rough) have simple closed-form solutions: the mirror reflection – where light only scatters when incident and outgoing directions are reflections of each other with respect to $\mathbf{n}$ – and the Lambertian reflectance – which is constant for all viewing directions, respectively. Few surfaces follow these archetypes but instead range in roughness and appear glossy.

Physically based energy conservative microfacet models [**83, 18, 88**] studies macroscopic surfaces as statistical distributions of purely specular micro-surface normals governing reflection directions. It is defined as

$$f(\boldsymbol{\omega_i},\boldsymbol{\omega_o}) = \frac{D(\mathbf{h})G(\boldsymbol{\omega_i},\boldsymbol{\omega_o},\mathbf{h})F(\boldsymbol{\omega_i},\mathbf{h})}{4\langle \mathbf{n} \cdot \boldsymbol{\omega_i} \rangle \langle \mathbf{n} \cdot \boldsymbol{\omega_o} \rangle},$$

where $\mathbf{h} = \frac{\omega_i + \omega_o}{|\omega_i + \omega_o|}$ is the microfacet normal – or half-direction vector, $F(\boldsymbol{\omega_i},\mathbf{h})$ is the Fresnel term, $D(\mathbf{h})$ is the microfacet normal distribution and $G(\boldsymbol{\omega_i},\boldsymbol{\omega_o},\mathbf{h})$ is the geometric masking term.

1.3.3. The Rendering Equation

Rearranging the BSDF equation to solve for outgoing radiance and accounting for the light emitted by the surface $L_e(\mathbf{x},\boldsymbol{\omega_o})$ yields the rendering equation (illustrated in Figure 2), introduced alongside path-tracing in 1986 by Kajiya [**41**] as

$$L(\mathbf{x},\boldsymbol{\omega_o}) = L_e(\mathbf{x},\boldsymbol{\omega_o}) + \int_{\Omega} L_i(\mathbf{x},\boldsymbol{\omega}) f_s(\mathbf{x},\boldsymbol{\omega},\boldsymbol{\omega_o}) \langle \boldsymbol{\omega} \cdot \mathbf{n} \rangle d\boldsymbol{\omega},$$

which describes exiting radiance at a surface point as the sum of emitted light and scattered light – the integral of incoming light L_i over the sphere of directions around $\mathbf{x}$ modulated by the surface reflectance model.

Intuitive in its formulation, the rendering equation unfortunately has unbounded dimensionality in that the outgoing radiance $L(\mathbf{x},\boldsymbol{\omega_o})$ is dependent on all incoming radiance $L_i(\mathbf{x},\boldsymbol{\omega_i})$, itself having to be solved for as outgoing radiance $L(\mathbf{y},-\boldsymbol{\omega_i})$, i.e., from another point, in the opposite direction. This recursive nature makes statistical integration methods better apt at converging toward an approximation over time.

Various rendering techniques attempt to solve the rendering equation, approaching the dimensionality problem from different angles and navigating the trade-offs between quality, compute time and memory limitations. From its linearity, the scattered part of the rendering equation is often decoupled as the sum of direct and indirect incident lighting.

The direct lighting integral can be reformulated as a discrete sum of incident lighting from the light sources present in the scene. Solving for direct lighting is in itself a intricate problem as the complexity of scenes' lighting grow [**12**]. Various works in hardware-accelerated rendering has been proposed to extend or facilitate direct lighting computation [**81, 35**], shadows [**27, 33**] or glossy reflections [**93**].

Indirect lighting – or global illumination – is nowadays commonly the product of numerical estimation in the form of Monte Carlo path-tracing [**67**], discussed in Section 1.3.4, where memory and performance limitations are aleviated by spatial, angular and temporal caching mechanisms, which we present in Section 1.6. Various other branches of research on computing global illumination exist [**29, 40**], which we won't cover for the purposes of this document.

1.3.4. Monte Carlo Integration

Unidirectional path-tracing using recursive ray-tracing from sensor to light sources was proposed by Kajiya [**41**] to solve the unbounded rendering equation by building independent light paths. This can be seen as a succession of discrete integral estimation steps. Monte Carlo integration methods estimate such integrals statistically by evaluating the weighted average value of the integrand modeled as a *probability density function*(PDF). Successive averaged samples taken from a PDF will converge over time to the result, with an error reduction proportional to $\frac{1}{\sqrt{N}}$, with N the number of random samples drawn uniformly over the domain of integration V (e.g., 4π for the sphere). Reformulating the rendering equation using Monte Carlo integration yields

$$\hat{L}(\mathbf{x},\boldsymbol{\omega_o}) \simeq L_e(\mathbf{x},\boldsymbol{\omega_o}) + \frac{V}{N}\sum_{j}^{N} L_i(\mathbf{x},\boldsymbol{\omega_j}) f_s(\mathbf{x},\boldsymbol{\omega_j},\boldsymbol{\omega_o}) \langle \boldsymbol{\omega_j} \cdot \mathbf{n} \rangle.$$

Since Monte Carlo integration rely on independent samples, each step along a light path will both randomly pick a direction toward the next step and calculate the probability of that direction being taken, as well as contributing to convergence of the final value by adding a potential light path contribution to the estimate. Thus, Monte Carlo estimation explores the many dimensions of the problem in an independent fashion, decoupled from its dimensional complexity. These methods are classified as *unbiased* and will converge to the correct answer statistically as more samples are computed, until which point the error is perceived as *noise*. The rate of convergence is analysed from the *variance* of the estimate in relation to the final result – or *ground-truth*. Capturing the ground-truth result of such unbounded integrals is in theory unfeasible. In practice, the ground-truth is estimated via an unbiased integration method by drawing a large number of samples with no regard to performances in which the variance should tend toward zero, against which error analysis is then performed for the validation of another – potentially biased – method.

This is the objective of noise reduction methods, the implementation of methods achieving equivalent qualitative results at a fraction of the samples and thus computing cost.

Importance sampling(IS) analyses the shape of the integrand to model a better fit density function that prioritizes sampling in regions of higher values, instead of uniformly distributing samples across the domain of integration. Convergence is achieved faster by privileging a larger density of samples in these important regions. An intuive example is the mirror reflection, with which only one direction of the hemispherical domain above the surface contains the information we require (i.e., delta function δ). Distributing samples uniformly (i.e., with a constant PDF) in the domain will theorically never converge to the solution, whereas a fitted delta density function reaches convergence with only one sample.

An importance sampling estimator $\hat{f}$ for a target function f sampled according to a probability density function $p(k) : k_1, \cdots, k_N \in V$ is given by

$$\hat{f} \simeq \frac{V}{N} \sum_{j}^{N} \frac{f(k_j)}{p(k_j)}, \quad f \sim p.$$

For an unbiased estimator, the chosen PDF must be strictly positive whenever the integrand is as well. Finding an efficient density function for an integrand so that it is proportional to it, i.e., $f \sim p$, is the major obstacle in importance sampling. Certain functions like analytical BRDFs tend to have identifiable PDFs which improve convergence, especially when the reflectance of the material gets glossier and the solid angle of potentially sampled directions shrinks. On the other hand, a distribution such as hemispherical visibility – or depth to first occluder – is a complex problem modeling a spatio-angularly-varying high-frequency function.

Importance sampling can be applied to the several independent terms of the rendering equation [**84, 68**]: the BRDF f_r and the lighting distribution L_i. Veach and Guibas [**85**] combine sampling and estimation of several density functions with *multiple importance sampling*(MIS). An inadequate choice of density functions in this case is however shown to perform poorly and to slow convergence relatively to regular importance sampling of the best of the two distribution [**60**].

Orthogonally, progressive sampling has become a way of aleviating time-to-first-frame by distributing sampling computation across time. A final image is not calculated once from a defined number of samples N. Instead, by keeping track of integration weights, an estimate is refined over time as more samples are drawn. This is especially convenient in the context of preview renderers used by artists or scene designers for fast feedback and iterative workflows.

Adaptive sampling uses the variance of the integrand itself as an oracle for adapting the sampling distribution [**61, 69**]. This driving criterion must first exist to control sample generation and these methods are often iterative and/or progressive. Adaptive sampling in image space results in a sparse set of samples that do not cover all the pixels and filtering techniques must be used to reconstruct the final image. We describe in Section 1.5 frequency-based methods to predict the sampling rate in 2-dimensional image space and the 3-dimensional spherical domains of the rendering integration. Filtering methods are reviewed in Section 1.6 as a way to reconstruct the final image from these spare sets of contributions.

The next sections of this chapter independently introduce some of the fundamentals necessary for the comprehension of the papers in the next chapters. We discuss in the next section the role of participative media in path-traced light transport and how volumetric media influences the rendering equation. We then review in Section 1.5 basis for the analysis of light transport in the frequency domain to facilitate gathering prior knowledge on rendering integrands and finding Monte Carlo adaptive sampling oracles. Finally, Section 1.6 describes strategies to cache and reuse illumination information to reduce memory and computing costs in interactive and real-time applications.

1.4. Influence of Participating Media

The term participating media (or volumetric media) in graphics applies to the set of phenomena that are volumetric in nature in that they can absorb, scatter or emit light. Though we have covered these behaviors in the context of hard surfaces in the previous section, the characteristics of a participating medium additionally describe the interactions of light with sparse particles of matter of microscopic scale, so that it may be simulated with geometric optics. The earth's atmosphere, smoke, fire, water and sand are examples of phenomena that are formulated and studied as participating media.

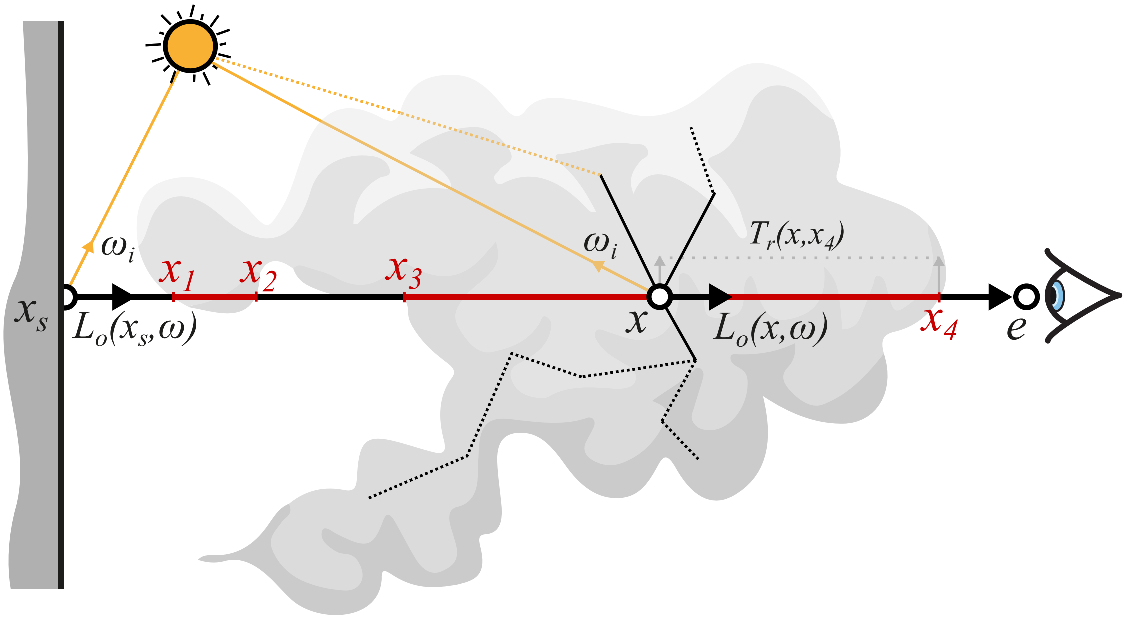


Fig. 3. A simple scene depicting light transport in participating media following the VRE. The total radiance incident to the eye e from direction $\boldsymbol{\omega}$ combines the attenuated outgoing surface radiance $L_o(\mathbf{x_s},\boldsymbol{\omega})$ and in-scattered radiance L_m along the ray. L_m is the integral of light scattered into $\boldsymbol{\omega}$ from other directions at all points in the medium along the ray, here pictured in **red**. The transmittance $T_r(\mathbf{x_a},\mathbf{x_b})$ depends on the extinction coefficient σ_t of the medium and governs the attenuation. Point $\mathbf{x}$ depicts a scattering event alongside $\boldsymbol{\omega}$. Here, the spherically in-scattered light is depicted as thin solid lines for single scattering, and dotted lines for multiple scattering. Each change in direction denotes a scattering event, treated in the same manner as at $\mathbf{x}$.

Light transport in paticipating media has first been studied in graphics through diffusion theory and the radiative transport problem [**16, 78**]. They define the *radiative transport equation*(RTE) defining the differential change in radiance due to the effects of media

$$(\boldsymbol{\omega} \cdot \nabla)L(\mathbf{x},\boldsymbol{\omega}) = \underbrace{L_e(\mathbf{x},\boldsymbol{\omega})}_{emisson} - \underbrace{\sigma_a(\mathbf{x})L(\mathbf{x},\boldsymbol{\omega})}_{absorption} + \underbrace{\sigma_s(\boldsymbol{\omega})L_i(\mathbf{x},\boldsymbol{\omega})}_{in-scattering} - \underbrace{\sigma_s(\boldsymbol{\omega})L(\mathbf{x},\boldsymbol{\omega})}_{out-scattering},$$

where L_e is the emitted radiance, L_i is the inscattered radiance and L is the general incoming radiance within the spherical domain S^2.

Volumetric media is defined in graphics as statistically independent particles, where the density of the medium ρ defines the size and number of particles that compose it. The absorption coefficient σ_a and scattering coefficient σ_s are derived from the density to specify the predisposition of photons to undergo both particular interactions, as we describe below.

Emission. Just like a surface, as described in Section 1.3.1, a medium can emit light. Emission in media is also a spatio-angularly varying source function and is fairly straightforward to model as a linearly additive term in the formulation.

Absorption. As photons flow through a medium, some of them are absorbed upon contact of volumetric particles and transformed in another source of energy. This decreases the total amount of light depending on the *absorption coefficient* σ_a of the medium.

Scattering. Instead of being absorbed, a photon in contact with a medium particle can scatter in another direction, just as with a surface. The *scattering coefficient* σ_s of a medium guides the rate of scattering of photons within it. When considering the change in radiance along a ray such as with the RTE, scattering is decoupled between in-scattering – or the amount of photons scattering in that direction – and out-scattering of photons from that direction to another. The directionality of scattering also depends on the phase function that characterises the medium as we discuss later on.

Extinction. Absorption and scattering are often combined as extinction in radiative transfer formulations and media parameterizations, yielding the *extinction coefficient* $\sigma_t = \sigma_a + \sigma_s$ of a medium. The extinction of a medium has various associated properties. The *mean free path* of a photon in media is the inverse of the extinction coefficient, and defines its average travel distance before being absorbed or scattered. When considering such light paths, the *optical thickness* $\tau(\mathbf{x},\boldsymbol{\omega},t) = \int_0^t \sigma_t(\mathbf{x} + t'\boldsymbol{\omega})dt'$ models the integrated density along a ray in media. The scattering *albedo* $\alpha = \sigma_s/\sigma_t$ is the ratio measuring the fraction of photons being scattered (as opposed to absorbed), where a value of unity implies that all extinction is due to scattering events.

When styding light paths in media, the RTE indicates that the energy along that path will be attenuated due to out-scattering and absorption. This effect is described by the Beer-Lambert law [**46**] and the transmittance T_r models this attenuation between two points as

$$T_r(\mathbf{x},\mathbf{x_t}) = e^{-\tau(\mathbf{x},\frac{\mathbf{x_t}-\mathbf{x}}{t},t)}, \quad t = ||\mathbf{x_t} - \mathbf{x}||.$$

A medium is considered homogeneous, as opposed to heterogeneous, when the extinction coefficient is spatially invariant, that is $\sigma_t(\mathbf{x}) = \sigma_t$. For homogeneous media, the transmittance is simplified to $T_r(\mathbf{x},\mathbf{x_t}) = e^{-\sigma_t t}$. This is a common assumption for certain types of thinner media such as fog [**47**] and in the field of subsurface scattering [**40**]. Closed-form heterogeneity simplifications are often employed as well, e.g., in the case of height-based media, where σ_t is a 1-dimensional function.

By treating the surface rendering equation defined in the previous section as a boundary condition of radiative transfer and by integrating the RTE, we define the *volume rendering equation*(VRE) (depicted in Figure 3) for the radiance L as

$$L(\mathbf{x},\boldsymbol{\omega}) = \underbrace{\int_0^s \sigma_t(\mathbf{x_t}) T_r(\mathbf{x},\mathbf{x_t}) L_o(\mathbf{x_t},\boldsymbol{\omega}) dt}_{L_m(\mathbf{x},\boldsymbol{\omega})} + \underbrace{T_r(\mathbf{x},\mathbf{x_s}) L_o(\mathbf{x_s},\boldsymbol{\omega})}_{L_s(\mathbf{x_s},\boldsymbol{\omega})},$$

where L_m is the radiance arriving from all points along $\boldsymbol{\omega}$ in the medium, L_s the attenuated radiance arriving from the surface intersected at the limit $\mathbf{x_s} = \mathbf{x} - s\boldsymbol{\omega}$ and L_o is the outgoing radiance

$$\begin{aligned} L_o(\mathbf{x},\boldsymbol{\omega}) &= L_e(\mathbf{x},\boldsymbol{\omega}) + \sigma_s(\boldsymbol{\omega}) L_i(\mathbf{x},\boldsymbol{\omega}) \\ &= L_e(\mathbf{x},\boldsymbol{\omega}) + \sigma_s(\boldsymbol{\omega}) \int_{S^2} f(\mathbf{x},\boldsymbol{\omega},\boldsymbol{\omega_i}) L(\mathbf{x},\boldsymbol{\omega_i}) d\boldsymbol{\omega_i}, \end{aligned}$$

in which f is the scattering distribution function that is either the BSDF f_s or the phase function f_p depending on whether x is located on the surface $\mathcal{A}$ or in the medium $\mathcal{V}$ such that

$$f(\mathbf{x},\boldsymbol{\omega},\boldsymbol{\omega_i}) = \begin{cases} \alpha(\mathbf{x}) f_p(\mathbf{x},\boldsymbol{\omega},\boldsymbol{\omega_i}) & \text{if } \mathbf{x} \in \mathcal{V}, \\ f_s(\mathbf{x},\boldsymbol{\omega},\boldsymbol{\omega_i}) & \text{if } \mathbf{x} \in \mathcal{A}. \end{cases}$$

Phase Functions. When a scattering event occur in media, the phase function f_p defines the spherical bidirectional PDF used to estimate post-interaction directionality. It possesses similar properties to the surface BSDF and the same propensity for closed-form approximations [**62**] and importance sampling [**10, 39**]. Two theories on particulate scattering approximations exist, which use depends on the size of the particles of the simulated media. The Rayleigh theory [**79**] considers particules smaller than the wavelength of light as are present in atmospheric scattering, while the Mie theory [**54**] is an approach for particles of arbitrary sizes which generalizes well for particles larger than the wavelenght of light, as in smoke and water. Complementary in nature, they indicate in aggregate that smaller wavelengths of light scatter more intensely for smaller particles (hence variations in sky color) and that the larger particles are, the more light they scatter forward albeit with reduced wavelength dependence. The Henyey-Greenstein phase function [**34**] is the most widely used approximation for Mie scattering and intuitively defines the *anisotropy* of scattering directionality with a unimodal parameter $g \in [-1,1]$, where $g = 0$ is the isotropic phase function $f_p = 1/(4\pi)$, a proclivity for forward scattering is indicated by a positive g and for backward scattering by a negative g.

Related to the distinction between direct and indirect lighting, a common simplification of participating media models restricts the VRE to at most a single scattering event in the medium, i.e., the vaccum formulation of radiance $L(\mathbf{x_t},\boldsymbol{\omega})$ replaces $L_o(\mathbf{x_t},\boldsymbol{\omega})$ in the definition of L_m. Single scattering convey more energy in thin media, while multiple scattering, i.e., two scattering events or more, is absolutely necessary for denser media where hundreds of scattering events are necessary to reproduce the appearance of phenomena such as dense clouds [**42**]. In the absence of participating media, i.e., $\sigma_t = 0$, the volume rendering equation defaults to the surface rendering equation denoted in the previous section.

As opposed to light transport in vacuum described in Section 1.3, solving the VRE necessitates sampling ray-particle interactions along light paths [**56**], that is, a distance along the ray and a scattering direction. While finding a scattering direction depends on the phase function analytically or through importance sampling, various sampling strategies have been conceived for the former problem. Luminaire sampling strategies work better in thinner media where single scattering dominates [**29**] while exponential distance sampling techniques, based on the Beer-Lambert law, better explore heterogeneous dense media. This is trivial in homogeneous media by inverting the transmission and solving for a distance t. In heterogeneous media, null-collision unbiased methods such as Woodcock – or delta – tracking [**92, 57**] use rejection sampling over the length of the ray by introducing ficticious matter to render the medium homogeneous. Distances can then be sampled analytically along the ray and rejected probabilistically depending on the ratio of real to fictitious density at that location. Upon rejection, the ray continues forward and a new distance is sampled. The performance of these techniques strongly correlate on how close the virtual homogeneous density is to the actual density along the ray, introducing many rejection steps in the worst case.

1.5. Frequency Analysis in Computer Graphics

In the previous sections, we have identified light transport as a series of integration steps along light-paths and how these discrete operative integrands inform us on the probabilistic behavior of light. Studying the local radiance variations over the many domains of integration we have identified can therefore communicate knowledge on the complexity of these signals. Frequency analysis and the use of frequency basis functions have been greatly successful in rendering for both the study of the lighting signal and its encoding.

The frequency content of a signal bounded to a specific domain in its primary space in part yields insight regarding its rate of change inside that domain. Knowledge of the spatial and angular variations of radiance, or other signals used in graphics, can greatly facilitate various tasks such as deriving density functions, adaptive sampling or determining the sizes of filtering kernels and the adequate band-limit for frequency based basis functions. We describe in Section 1.5.1 the Fourier analysis, which decomposes a general function in its oscillatory components, or frequency content. Section 1.5.2 briefly reviews spherical harmonics and their applications in rendering as a polyvalent basis function.

1.5.1. Fourier Analysis

Frequency analysis was introduced in the nineteenth century by Joseph Fourier as a method to solve the heat equation. In graphics, Cook [**19**] studies sampling distributions in frequency space to introduce stochastic sampling, a method to generate appropriate nonuniform sample locations which frequency content is similar to that of a Poisson disk distribution, to aleviate the aliasing that results from regular uniform point sampling. Durand and colleagues [**23**] later introduce the frequency analysis of light transport to derive knowledge from the frequency content of radiance as it propagates from its source to the sensor, to predict the characteristics of lighting interactions and drive adaptive sampling strategies. To this end, they study spatio-angular bandwidth of the local 4-dimensional *light field* and how it is altered by light transport, where the angular dimension of the light field is approximated by its tangent via the paraxial hypothesis.

Frequency analysis in rendering thus seeks to study Monte Carlo numerical integration and sampling patterns in the context of path tracing in terms of signal processing, where the source of numerical error in frequency space is well recognized, linearity of functions is preserved, and convolution and multiplication are symmetrical operations between the spatial and frequency spaces [**22**]. This field of study has since then been broadly researched in graphics, e.g., for direct [**5**] and indirect [**52, 11**] illumination, distribution effects [**77, 26, 8, 53, 94**], soft shadows [**24, 25, 51**] and participating media [**7**].

The Fourier transform is the mathematical mechanism that decomposes a function f – expressed in the spatio-temporal *primal* domain – into its constituent frequencies ξ defining a frequency spectrum F in the frequency domain. The Fourier transform F and its inverse F^{-1} are defined as

$$F(\xi) = F\{f\} = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx,$$
$$f(x) = F^{-1}\{F\} = \int_{-\infty}^{\infty} F(\xi) e^{2\pi i x \xi} d\xi.$$

where the decomposition comes from the relation $e^{i\theta} = \cos(\theta) + i\sin(\theta)$.

According to the Nyquist-Shannon theorem, an integral can be estimated exactly without aliasing with a sampling rate higher than its maximum frequency (bandwidth) – half the Nyquist rate. Conversely, the Nyquist-limit (band-limit) of a signal – or largest representable bandwidth – is one-half of the sampling rate. Deriving the band-limit of an integral function by studying the bounds of its bandwidth in the frequency domain can thus serve as a criterion for adaptive sampling. The bandwidth of an amplitude spectrum in frequency space can be bounded by taking the 95^{th} percentile along its orthonormal dimensions to encompass most of the signal. Tighter fits of the spectrum have been studied as well [**14, 26**].

Instead of defining the band-limit of the amplitude spectrum in terms of bounded shapes, Belcour and colleagues [**8**] approximate the extents of the spectrum as a gaussian parameterized by the covariance matrix Σ of the local light field. The gaussian $g(\mathbf{z}) = e^{-\mathbf{z}^T \Sigma^{-1} \mathbf{z}}$ bounds the spectrum and is used directly to measure variation and anisotropy of the signal. The covariance matrix is defined as

$$\Sigma_{i,j} \equiv \int_{\mathbf{z} \in \Omega} \langle \mathbf{z}, \mathbf{e}_i \rangle \langle \mathbf{z}, \mathbf{e}_j \rangle |F(\mathbf{z})| d\mathbf{z},$$

where the diagonal elements of Σ are the variances of the amplitude of the spectrum along each input dimension – the set Ω, with $\mathbf{e}_i, \mathbf{e}_j \in \Omega$ – of the local light field, and the off-diagonal elements are the covariances between the i^{th} and j^{th} dimensions. Similarly to the seminal work of Durand et al. [**23**], the covariance matrix here serves the purpose of band-limiting the spectrum for the purpose of deriving sampling densities and reconstruction filters sizes.

1.5.2. Spherical Harmonics

Spherical and hemispherical functions are ubiquitous in rendering, from distant and local illumination to bidirectional reflectance distribution functions. *Spherical Harmonics*(SH) are a natural basis functions for the representation of these angularly-defined functions. Harmonic functions are functions that solve Laplace's equation, and spherical harmonics are the set of orthogonal functions that satisfy this equation over the two dimensional spherical domain, in the same manner that the Fourier series solves it in one dimension over the circle. Several resources [**65, 66, 30, 63**] extensively explore the intricacies of Spherical Harmonics and their use in rendering.

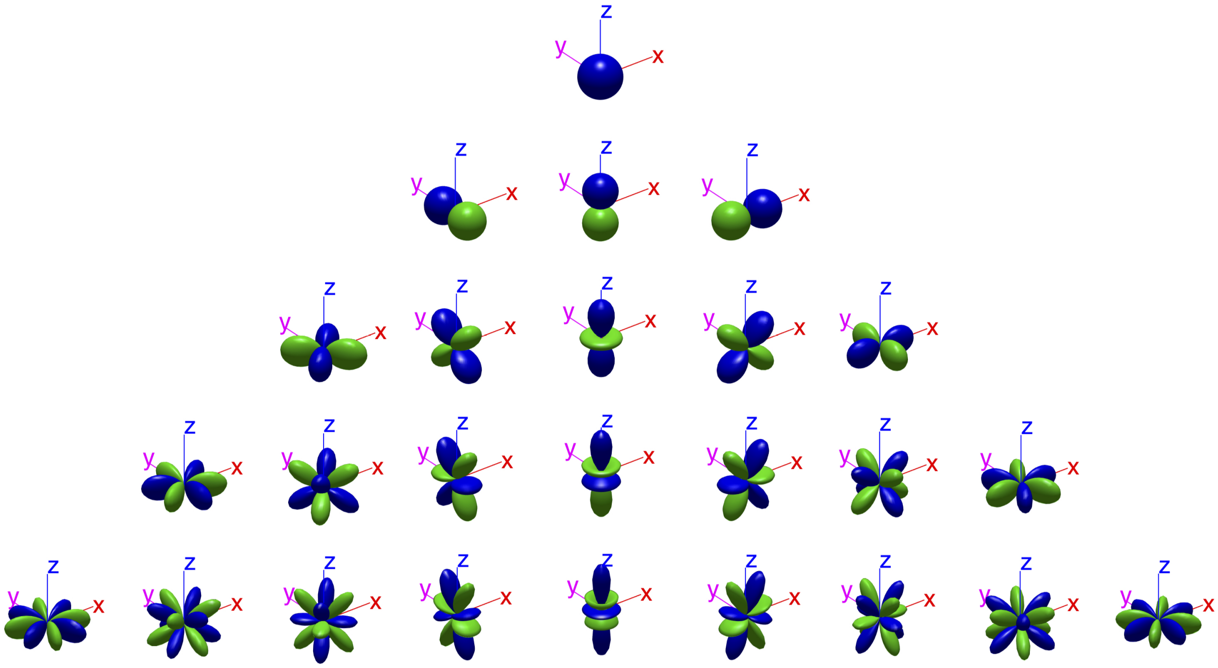


Fig. 4. Plot of the first bands of real valued spherical harmonic basis functions. Blue indicates positive values and green negative values. Top to bottom denotes the order l. Zonal harmonics are the z-aligned subset $m = 0$ represented in the central column. The off-center $m \neq 0$ bases denote the negative and positive degrees. A spherical function can be projected onto each basis function to decompose it into its frequency moments. Reprinted from [**64**].

The spherical harmonic basis functions possess several useful properties for rendering:

- As an orthonormal basis, it is straightforward to project a spherical function onto and from spherical harmonics .
- Similarly to the Fourier transform, a convolution in the primal domain is a multiplication in the SH domain and vice versa.
- Spherical harmonics are rotationally invariant, i.e., rotated sample sets of the primal function present no aliasing when projected onto SH.
- They possess efficient forms for rotation operations and integrated product of two SH functions.
- An isotropic spherical function projected to SH yields *Zonal Harmonics*(ZH) that further possess efficient rotational qualities and compactness [**58**].

A spherical function f can be *projected* onto each spherical harmonic basis function y_l^m to obtain a vector f_l^m of SH coefficients such that

$$f_l^m = \int_{S^2} f(\boldsymbol{\omega}) y_l^m(\boldsymbol{\omega}) d\boldsymbol{\omega},$$

and the original function f can be *expanded* as a linear combination of these coefficients and basis functions

$$f(\omega) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} f_l^m y_l^m(\boldsymbol{\omega}),$$

where the coefficients are indexed according to the *order* l and the *degree* m, where m indexes the $2l+1$ basis functions of each band, an order-N projection comprises the bands $0 \leq l \leq N-1$ and lower bands represent lower frequencies. It is often convenient to reformulate the expansion as a singly-indexed vector such that $i = l(l+1)+m+1$. Zonal harmonics form the SH subset with $m = 0$, reducing their size to one coefficient per band. Figure 4 represents the five first bands of spherical harmonic basis functions.

In computer graphics, the real-valued set of spherical harmonic basis functions is prevalent, defined as

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

using here the spherical coordinate of the projected direction vector $\boldsymbol{\omega} = (x,y,z) = (\theta,\phi) \in S^2$ and where P_l^m are the associated Legendre polynomials and K_l^m the normalization factors

$$K_l^m = \sqrt{\frac{(2l+1)}{4\pi}\frac{(l-|m|)!}{(l+|m|)!}}.$$

The set of spherical harmonics so defined forms a pyramid of basis functions which order represent the frequency of the basis over the sphere. Note that retrieving the function f in the primal domain from its frequency-space coefficients is a sum over an infinite number of bands, similar to the Fourier transform. In practice, an upper bound on the bandwidth of the signal is placed yielding a *band-limited* approximation of the primal function. Band-limiting the signal introduces *ringing* artifact of positive and negative values in the reconstruction that has to be dealt with accordingly [**74**]. Spherical harmonics are thus well tailored for the representation and encoding of lower frequency signals, i.e., rougher BRDFs, sky illumination and indirect local illumination. Inducing a band-limit to the signal is also commonly done for performance purposes in the context of real-time illumination with SH basis functions using vector instructions and parallel processing, e.g., the order-2 quadratic RGB SH uses 3 SSE vectors.

The trade-offs presented by spherical harmonics are attractive for the encoding of various illumination and visibility functions of lower frequency content, while higher frequency effects might be more appropriately represented by other frequency-space bases such as *wavelets* or non-frequency-based functions [**37, 9**] such as spherical gaussians, radial basis functions or cosine power lobes. Spherical harmonics are also commonly used as an intermediate bases for further processing [**36, 9**].

1.6. Illumination Caching and Reuse

Computing global illumination at interactive or real-time performances and manageable memory cost is still a subject of active research. Knowing these limitations, such algorithms rely instead on differed or asynchronous execution. Parts of the lighting, sampling or data conversion is computed partially ahead of time and stored in memory to be efficiently fetched at run-time. Abrash [**1**] illustrated this pattern in the early days of 3D games by precomputing diffuse lighting, stored in texture maps then applied on surfaces. This technique known as *light mapping* still lives on in modern production renderers today [**72**].

Precomputed Radiance Transfer. Following the theory of locally low dimensional light transport [**48**], *precomputed radiance transfer*(PRT) methods [**76, 75, 47**] approximate spatially coherent lighting signals such as diffuse and low frequency glossy interreflections, smooth direct lighting and self-shadowing in a process called *baking.* Techniques based on PRT precompute the transferred incident lighting at a sparse number of locations such as mesh vertices [**43**], per-instance local probes [**16**] or global spatially distributed probes [**31, 73**], encoded with basis functions, most often as spherical harmonics coefficients.

By assuming a locally static model in the case of diffuse interreflections, PRT encodes transfer vectors $[t(\mathbf{p})]$ at locations $\mathbf{p}$ representing how the local geometry shadows and scatters light onto itself. The transfer vectors encode the linear transformation between SH projected incident radiance $[L]_i$ and outgoing radiance L_o for the n^{th} order SH projection as

$$L_o(\mathbf{p}) = \sum_{i}^{n^2} [t(\mathbf{p})]_i [L(\mathbf{p})]_i.$$

A single pass is necessary to encode self-shadowing from direct lighting, where the transfer vector encode the cosine-weighted hemispherical kernel around normal $\mathbf{n}$ and binary visibility function at a diffuse surface point of albedo $\rho(\mathbf{p})$ such that

$$[t(\mathbf{p})]_i = \frac{\rho(\mathbf{p})}{\pi} \int_{\Omega} \lfloor \langle \boldsymbol{\omega} \cdot \mathbf{n} \rangle \rfloor V(\mathbf{p},\boldsymbol{\omega}) y_i(\boldsymbol{\omega}) d\boldsymbol{\omega}.$$

The same paradigm is extended to diffuse interreflections by iteratively considering the per-bounce transfer vectors $[t(\mathbf{p})]_i^b$ now computed by summing the contributions from previous-bounce transfer vectors $[t(\mathbf{p})]_i^{b-1}$. The sum of all bounce transfer vectors account for direct and indirect shadowed diffuse lighting. Although needing higher order projections to account for higher frequency content, PRT derives a similar procedure for glossy reflections via sparse transfer matrices to account for directionality. At run-time, the linear transformation of an arbitrary incident SH-projected lighting function occurs through a dot product (resp. matrix-vector multiplication) with the transfer vectors (resp. matrices) to retrieve the outgoing radiance at a point.

Filtering Methods. As described in Section 1.3, modern production renderers adaptively sample regions of the final image or other integration domains to reduce the cost of shading and filter or denoise the resulting sparse sample sets to fill-in empty or under-sampled regions. As denoising algorithms get more performant [**87, 15**], reducing the amount of shaded points and reconstructing the image becomes an attractive alternative. Though we discuss frequency based sampling criteria in Section 1.5, alternative sampling strategies exist based on gradient [**90, 45**] and hessian [**71**] analysis, on *deep* Monte Carlo renders [**86**] or on tools from the machine learning community [**6, 15**].

Filtering methods are employed to smooth out noisy Monte Carlo images (denoising) or reconstruct missing spatial information from a sparse set of point contributions. These operations are dual of each other and respectively referred to as gathering and splatting methods. Gathering methods apply a filter kernel $w(\mathbf{p}, \mathbf{q})$ iteratively on each element $\mathbf{p}$ of the domain to determine by weighted average the contribution received from each surrounding element $\mathbf{q}$ covered by the kernel $w(\mathbf{p}, \mathbf{q})$. Conversely, splatting methods iterate over each input sample to spread their contribution to the neighboring area under the kernel. While these techniques are by definition biased since each element becomes correlated to others from their local neighborhood, the bias tends to zero in the limit as kernel sizes skrink to the size of the element.

A denoising process evaluates the filtered value $\hat{g}$ from input value g as

$$\hat{g}(\mathbf{p}) = \frac{1}{W(\mathbf{p})} \sum_{q \in \mathcal{N}_{\mathbf{p}}} w(\mathbf{p}, \mathbf{q}) g(\mathbf{q}),$$

where $w(\mathbf{p}, \mathbf{q})$ is the kernel function, $W(\mathbf{p}) = \sum_{q \in \mathcal{N}_{\mathbf{p}}} w(\mathbf{p}, \mathbf{q})$ is the normalization factor for the weighted average and $\mathcal{N}_{\mathbf{p}}$ is the neighborhood around $\mathbf{p}$.

Edge-preserving smoothing functions are a popular pick for denoising kernels, such as the bilateral [**82, 44**] and non-local means filters [**70**]. They are defined to smooth the domain while preserving spatial and radiometric high frequencies but can trivially be extended with *joint filtering* to any complementary feature set available prior to filtering and defined over the domain. A typical gaussian bilateral filter with feature vector $\mathbf{f} = f_1,...,f_k$ has the form

$$w(\mathbf{p}, \mathbf{q}) = \prod_{i=1}^{k} \exp\left(\frac{-||f_i(\mathbf{p}) - f_i(\mathbf{q})||^2}{2\sigma_i^2}\right),$$

where σ_i^2 is the adjusting gaussian parameter of feature f_i. Such kernels imply large weights for elements that are closer over the set of feature spaces.

Precomputing filtered elements from a high frequency dataset is known as pre-filtering [**13, 49**] to avoid run-time processing at the cost of memory storage of the pre-filtered hierarchy. This technique is often used for static elements such as triangular meshes in the case of level-of-detail (LOD), albedo textures, BRDFs and image based lighting. The correct footprint is then derived at run-time from the solid angle of integration to query the pre-filtered hierarchy.

References

[1] Michael Abrash. Quake's lighting model: Surface caching, 2000.

[2] Arthur Appel. Some techniques for shading machine renderings of solids. In *Proceedings of the April 30–May 2, 1968, Spring Joint Computer Conference*, AFIPS '68 (Spring), pages 37–45, New York, NY, USA, 1968. ACM.

[3] James Arvo. Stratified sampling of spherical triangles. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, page 437–438, New York, NY, USA, 1995. Association for Computing Machinery.

[4] James Arvo and David Kirk. Particle transport and image synthesis. *SIGGRAPH Comput. Graph.*, 24(4):63–66, September 1990.

[5] Mahdi M. Bagher, Cyril Soler, Kartic Subr, Laurent Belcour, and Nicolas Holzschuch. Interactive rendering of acquired materials on dynamic geometry using bandwidth prediction. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '12, pages 127–134, New York, NY, USA, 2012. ACM.

[6] Steve Bako, Thijs Vogels, Brian McWilliams, Mark Meyer, Jan Novák, Alex Harvill, Pradeep Sen, Tony DeRose, and Fabrice Rousselle. Kernel-predicting convolutional networks for denoising monte carlo renderings. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2017)*, 36(4), 2017.

[7] Laurent Belcour, Kavita Bala, and Cyril Soler. A Local Frequency Analysis of Light Scattering and Absorption. *ACM Transactions on Graphics*, 33(5):Article No. 163, August 2014. (presented at SIGGRAPH 2014).

[8] Laurent Belcour, Cyril Soler, Kartic Subr, Nicolas Holzschuch, and Fredo Durand. 5d covariance tracing for efficient defocus and motion blur. *ACM Trans. Graph.*, 32(3):31:1–31:18, July 2013.

[9] Laurent Belcour, Guofu Xie, Christophe Hery, Mark Meyer, Wojciech Jarosz, and Derek Nowrouzezahrai. Integrating Clipped Spherical Harmonics Expansions. *ACM Trans. Graph*, 37(2), March 2018.

[10] Laurent Belcour, Guofu Xie, Christophe Hery, Mark Meyer, Wojciech Jarosz, and Derek Nowrouzezahrai. Integrating clipped spherical harmonics expansions. *ACM Transactions on Graphics*, March 2018.

[11] Laurent Belcour, Ling-Qi Yan, Ravi Ramamoorthi, and Derek Nowrouzezahrai. Antialiasing Complex Global Illumination Effects in Path-space. *ACM Transactions on Graphics*, 36(1), January 2017.

[12] Benedikt Bitterli, Chris Wyman, Matt Pharr, Peter Shirley, Aaron Lefohn, and Wojciech Jarosz. Spatiotemporal reservoir resampling for real-time ray tracing with dynamic direct lighting. *ACM Trans. Graph.*, 39(4), July 2020.

[13] Eric Bruneton and Fabrice Neyret. A survey of nonlinear prefiltering methods for efficient and accurate surface shading. *IEEE Transactions on Visualization and Computer Graphics*, 18(2):242–260, 2012.

[14] Jin-Xiang Chai, Xin Tong, Shing-Chow Chan, and Heung-Yeung Shum. Plenoptic sampling. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, page 307–318, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[15] Chakravarty R. Alla Chaitanya, Anton S. Kaplanyan, Christoph Schied, Marco Salvi, Aaron Lefohn, Derek Nowrouzezahrai, and Timo Aila. Interactive reconstruction of monte carlo image sequences using a recurrent denoising autoencoder. *ACM Trans. Graph.*, 36(4), July 2017.

[16] S. Chandrasekhar. *Radiative Transfer*. Dover Books on Intermediate and Advanced Mathematics. Dover Publications, 1960.

[17] Per H. Christensen and Wojciech Jarosz. The path to path-traced movies. *Found. Trends. Comput. Graph. Vis.*, 10(2):103–175, October 2016.

[18] R. L. Cook and K. E. Torrance. A reflectance model for computer graphics. *ACM Trans. Graph.*, 1(1):7–24, January 1982.

[19] Robert L. Cook. Stochastic sampling in computer graphics. *ACM Trans. Graph.*, 5(1):51–72, January 1986.

[20] Robert L. Cook, Thomas Porter, and Loren Carpenter. Distributed ray tracing. *SIGGRAPH Comput. Graph.*, 18(3):137–145, January 1984.

[21] Michal Drobot. Physically based area lights. *GPU Pro*, 5:67–100, 2014.

[22] Frédo Durand. A frequency analysis of monte-carlo and other numerical integration schemes. *Technical Report*, 2011.

[23] Frédo Durand, Nicolas Holzschuch, Cyril Soler, Eric Chan, and François X. Sillion. A frequency analysis of light transport. *ACM Trans. Graph.*, 24(3):1115–1126, July 2005.

[24] Kevin Egan, Frédo Durand, and Ravi Ramamoorthi. Practical filtering for efficient ray-traced directional occlusion. *ACM Trans. Graph.*, 30(6):180:1–180:10, December 2011.

[25] Kevin Egan, Florian Hecht, Frédo Durand, and Ravi Ramamoorthi. Frequency analysis and sheared filtering for shadow light fields of complex occluders. *ACM Trans. Graph.*, 30(2):9:1–9:13, April 2011.

[26] Kevin Egan, Yu-Ting Tseng, Nicolas Holzschuch, Frédo Durand, and Ravi Ramamoorthi. Frequency analysis and sheared reconstruction for rendering motion blur. *ACM Trans. Graph.*, 28(3):93:1–93:13, July 2009.

[27] Elmar Eisemann, Ulf Assarsson, Michael Schwarz, Michal Valient, and Michael Wimmer. Efficient real-time shadows. In *ACM SIGGRAPH 2013 Courses*, SIGGRAPH '13, pages 18:1–18:54, New York, NY, USA, 2013. ACM.

[28] Luca Fascione, Johannes Hanika, Mark Leone, Marc Droske, Jorge Schwarzhaupt, Tomáš Davidovič, Andrea Weidlich, and Johannes Meng. Manuka: A batch-shading architecture for spectral path tracing in movie production. *ACM Trans. Graph.*, 37(3):31:1–31:18, August 2018.

[29] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modeling the interaction of light between diffuse surfaces. *SIGGRAPH Comput. Graph.*, 18(3):213–222, January 1984.

[30] Robin Green. Spherical harmonic lighting: The gritty details. 2003.

[31] Gene Greger, Peter Shirley, Philip M. Hubbard, and Donald P. Greenberg. The irradiance volume. *IEEE Comput. Graph. Appl.*, 18(2):32–43, March 1998.

[32] Eric Heitz, Jonathan Dupuy, Stephen Hill, and David Neubelt. Real-time polygonal-light shading with linearly transformed cosines. *ACM Trans. Graph.*, 35(4):41:1–41:8, July 2016.

[33] Eric Heitz, Stephen Hill, and Morgan McGuire. Combining analytic direct illumination and stochastic shadows. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '18, pages 2:1–2:11, New York, NY, USA, 2018. ACM.

[34] L. G. Henyey and J. L. Greenstein. Diffuse radiation in the Galaxy. *The Astrophysical Journal*, 93:70–83, January 1941.

[35] Stephen Hill, Stephen McAuley, Cyril Jover, Sébastien Lachambre, Angelo Pesce, Xian-Chun Wu, Roger Cordes, Christophe Hery, Sébastien Hillaire, Naty Hoffman, Jorge Jiménez, Brian Karis, Sébastien Lagarde, Dan Lobl, and Ryusuke Villemin. Physically based shading in theory and practice. In *ACM SIGGRAPH 2016 Courses*, SIGGRAPH '16, New York, NY, USA, 2016. ACM.

[36] JT Hooker. Volumetric global illumination at treyarch.

[37] MichałIwanicki and Peter-Pike Sloan. Ambient dice. In *Proceedings of the Eurographics Symposium on Rendering: Experimental Ideas & Implementations*, EGSR '17, pages 19–29, Goslar Germany, Germany, 2017. Eurographics Association.

[38] Michał Iwanicki and Peter-Pike Sloan. Precomputed lighting in Call of Duty: Infinite Warfare. In *Advances in Real-Time Rendering in Games*, SIGGRAPH Courses, 2017.

[39] Wojciech Jarosz, Nathan A. Carr, and Henrik Wann Jensen. Importance sampling spherical harmonics. page 577–586, apr 2009.

[40] Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A practical model for subsurface light transport. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, pages 511–518, New York, NY, USA, 2001. ACM.

[41] James T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143–150, August 1986.

[42] Simon Kallweit, Thomas Müller, Brian McWilliams, Markus Gross, and Jan Novák. Deep scattering: Rendering atmospheric clouds with radiance-predicting neural networks. *ACM Trans. Graph. (Proc. of Siggraph Asia)*, 36(6), November 2017.

[43] Ladislav Kavan, Adam W. Bargteil, and Peter-Pike Sloan. Least squares vertex baking. In *Proceedings of the Twenty-second Eurographics Conference on Rendering*, EGSR '11, pages 1319–1326, Aire-la-Ville, Switzerland, Switzerland, 2011. Eurographics Association.

[44] Johannes Kopf, Michael F. Cohen, Dani Lischinski, and Matt Uyttendaele. Joint bilateral upsampling. *ACM Trans. Graph.*, 26(3):96–es, July 2007.

[45] Jaroslav Křivánek, Pascal Gautron, Greg Ward, Henrik Wann Jensen, Per H. Christensen, and Eric Tabellion. Practical global illumination with irradiance caching. In *ACM SIGGRAPH 2008 Classes*, SIGGRAPH '08, pages 60:1–60:20, New York, NY, USA, 2008. ACM.

[46] J.H. Lambert. *Photometria sive de mensura et gradibus luminis, colorum et umbrae*. Nineteenth Century Collections Online (NCCO): Photography: The World through the Lens. sumptibus vidvae E. Klett, typis C.P. Detleffsen, 1760.

[47] Jaakko Lehtinen. A framework for precomputed and captured light transport. *ACM Trans. Graph.*, 26(4), October 2007.

[48] Dhruv Mahajan, Ira Kemelmacher Shlizerman, Ravi Ramamoorthi, and Peter Belhumeur. A theory of locally low dimensional light transport. In *ACM SIGGRAPH 2007 Papers*, SIGGRAPH '07, page 62–es, New York, NY, USA, 2007. Association for Computing Machinery.

[49] Josiah Manson and Peter-Pike Sloan. Fast filtering of reflection probes. *Computer Graphics Forum (EGSR)*, 35(4):119–127, 2016.

[50] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. *ACM Trans. Graph.*, 22(3):759–769, July 2003.

[51] Soham Uday Mehta, Brandon Wang, and Ravi Ramamoorthi. Axis-aligned filtering for interactive sampled soft shadows. *ACM Trans. Graph.*, 31(6):163:1–163:10, November 2012.

[52] Soham Uday Mehta, Brandon Wang, Ravi Ramamoorthi, and Fredo Durand. Axis-aligned filtering for interactive physically-based diffuse indirect lighting. *ACM Trans. Graph.*, 32(4):96:1–96:12, July 2013.

[53] Soham Uday Mehta, JiaXian Yao, Ravi Ramamoorthi, and Fredo Durand. Factored axis-aligned filtering for rendering multiple distribution effects. *ACM Trans. Graph.*, 33(4):57:1–57:12, July 2014.

[54] Gustav Mie. Beiträge zur optik trüber medien, speziell kolloidaler metallösungen. *Annalen der Physik*, 330(3):377–445, 1908.

[55] Fred E. Nicodemus. Directional reflectance and emissivity of an opaque surface. *Appl. Opt.*, 4(7):767–775, Jul 1965.

[56] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte Carlo methods for volumetric light transport simulation. May 2018.

[57] Jan Novák, Andrew Selle, and Wojciech Jarosz. Residual ratio tracking for estimating attenuation in participating media. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, 33(6), November 2014.

[58] Derek Nowrouzezahrai, Patricio Simari, and Eugene Fiume. Sparse zonal harmonic factorization for efficient sh rotation. *ACM Transactions on Graphics*, 2012.

[59] F O. Bartell, Eustace Dereniak, and W L. Wolfe. The theory and measurement of bidirectional reflectance distribution function (BRDF) and bidirectional transmittance distribution function (BTDF). *Proceedings of SPIE - The International Society for Optical Engineering*, 257:154–160, 01 1980.

[60] Art Owen and Yi Zhou. Safe and effective importance sampling. *Journal of the American Statistical Association*, 95:135–143, 1998.

[61] J. Painter and K. Sloan. Antialiased ray tracing by adaptive progressive refinement. *SIGGRAPH Comput. Graph.*, 23(3):281–288, July 1989.

[62] M. Pharr, W. Jakob, and G. Humphreys. *Physically Based Rendering: From Theory to Implementation.* Elsevier Science, 2016.

[63] Peter pike Sloan. Stupid spherical harmonics (sh) tricks. In *In GDC'08*, 2008.

[64] Archontis Politis. Real/complex spherical harmonic transform, gaunt coefficients and rotations, 2021.

[65] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 497–500, New York, NY, USA, 2001. ACM.

[66] Ravi Ramamoorthi and Pat Hanrahan. Frequency space environment map rendering. *ACM Trans. Graph.*, 21(3):517–526, July 2002.

[67] Tobias Ritschel, Carsten Dachsbacher, Thorsten Grosch, and Jan Kautz. The state of the art in interactive global illumination. *Comput. Graph. Forum*, 31(1):160–188, February 2012.

[68] Fabrice Rousselle, Petrik Clarberg, Luc Leblanc, Victor Ostromoukhov, and Pierre Poulin. Efficient product sampling using hierarchical thresholding. *The Visual Computer*, 24, 07 2008.

[69] Fabrice Rousselle, Claude Knaus, and Matthias Zwicker. Adaptive sampling and reconstruction using greedy error minimization. *ACM Trans. Graph.*, 30(6):1–12, December 2011.

[70] Fabrice Rousselle, Claude Knaus, and Matthias Zwicker. Adaptive rendering with non-local means filtering. *ACM Trans. Graph.*, 31(6), November 2012.

[71] Jorge Schwarzhaupt, Henrik Wann Jensen, and Wojciech Jarosz. Practical hessian-based error control for irradiance caching. *ACM Transactions on Graphics (TOG)*, 31, 11 2012.

[72] Dario Seyb, Peter-Pike Sloan, Ari Silvennoinen, Michał Iwanicki, and Wojciech Jarosz. The design and evolution of the uberbake light baking system. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 39(4), July 2020.

[73] Ari Silvennoinen and Jaakko Lehtinen. Real-time global illumination by precomputed local reconstruction from sparse radiance probes. *ACM Trans. Graph.*, 36(6):230:1–230:13, November 2017.

[74] Peter-Pike Sloan. Deringing spherical harmonics. pages 1–4, 11 2017.

[75] Peter-Pike Sloan, Jesse Hall, John Hart, and John Snyder. Clustered principal components for precomputed radiance transfer. *ACM Trans. Graph.*, 22(3):382–391, July 2003.

[76] Peter-Pike Sloan, Jan Kautz, and John Snyder. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Trans. Graph.*, 21(3):527–536, July 2002.

[77] Cyril Soler, Kartic Subr, Frédo Durand, Nicolas Holzschuch, and François Sillion. Fourier depth of field. *ACM Trans. Graph.*, 28(2):18:1–18:12, May 2009.

[78] Jos Stam. Multiple scattering as a diffusion process. In Patrick M. Hanrahan and Werner Purgathofer, editors, *Rendering Techniques '95*, pages 41–50, Vienna, 1995. Springer Vienna.

[79] John William Strutt. *On the Light from the Sky, its Polarization and Colour*, volume 1 of *Cambridge Library Collection - Mathematics*, page 87–103. Cambridge University Press, 2009.

[80] Bo Sun, Ravi Ramamoorthi, Srinivasa G. Narasimhan, and Shree K. Nayar. A practical analytic single scattering model for real time rendering. *ACM Transactions on Graphics*, 24(3):1040–1049, July 2005.

[81] Natalya Tatarchuk, Sébastien Hillaire, Tomasz Stachowiak, Huw Bowles, Beibei Wang, Ari Silvennoinen, and Ville Timonen. Advances in real time rendering, part i. In *ACM SIGGRAPH 2015 Courses*, SIGGRAPH '15, New York, NY, USA, 2015. ACM.

[82] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 839–846, 1998.

[83] Kenneth Torrance and E. Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of The Optical Society of America*, 57, 09 1967.

[84] Eric Veach. *Robust Monte Carlo Methods for Light Transport Simulation.* PhD thesis, Stanford, CA, USA, 1998. AAI9837162.

[85] Eric Veach and Leonidas J. Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, page 419–428, New York, NY, USA, 1995. Association for Computing Machinery.

[86] D. Vicini, D. Adler, J. Novák, F. Rousselle, and B. Burley. Denoising deep monte carlo renderings. *Computer Graphics Forum*, 0(0).

[87] Thijs Vogels, Fabrice Rousselle, Brian Mcwilliams, Gerhard Röthlin, Alex Harvill, David Adler, Mark Meyer, and Jan Novák. Denoising with kernel prediction and asymmetric loss functions. *ACM Trans. Graph.*, 37(4):124:1–124:15, July 2018.

[88] Bruce Walter, Stephen R. Marschner, Hongsong Li, and Kenneth E. Torrance. Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*, EGSR'07, pages 195–206, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.

[89] Gregory J. Ward. Measuring and modeling anisotropic reflection. *SIGGRAPH Comput. Graph.*, 26(2):265–272, July 1992.

[90] Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. A ray tracing solution for diffuse interreflection. *SIGGRAPH Comput. Graph.*, 22(4):85–92, June 1988.

[91] Turner Whitted. An improved illumination model for shaded display. *Commun. ACM*, 23(6):343–349, June 1980.

[92] E Woodcock, T Murphy, P Hemmings, and S Longworth. Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry. In *Applications of Computing Methods to Reactor Problems*, volume 557, 1965.

[93] Chao Xu, Rui Wang, and Hujun Bao. Realtime rendering glossy to glossy reflections in screen space. *Comput. Graph. Forum*, 34(7):57–66, October 2015.

[94] Ling-Qi Yan, Soham Uday Mehta, Ravi Ramamoorthi, and Fredo Durand. Fast 4d sheared filtering for interactive rendering of distribution effects. *ACM Trans. Graph.*, 35(1):7:1–7:13, December 2015.

Premier article.

Frequency Based Radiance Cache for Rendering Animations

par

Renaud Adrien Dubouchet[1], Laurent Belcour[2], and Derek Nowrouzezahrai[3]

(1) Université de Montréal
(2) Unity Technology
(3) McGill University

Cet article a été soumis à la revue Eurographics Symposium on Rendering 2017, en mars 2017. Il a été reformaté afin de suivre convenablement le format de cette thèse.

Renaud Adrien Dubouchet est l'auteur principal de cet article, et a dévelopé en collaboration avec Laurent Belcour le modèle mathématique derrière la dérivation des oracles fréquentiels. Il a également implémenté la technique sur un outil de tracé de chemins existant et a été l'auteur principal de la rédaction de cet article, qu'il a présenté à la conférence EGSR 2017 à Helsinki, Finlande.

Cet article étend aux séquences animées le travail de recherche de Frédo Durand et collègues de 2005 sur l'utilisation de l'étude fréquentielle du transport de la lumière pour l'échantillonnage et la reconstruction. De ce fait, un nouvel oracle fréquentiel est dérivé pour sous-échantillonner adéquatement l'image, mais aussi commander le cache de radiance déployé pour aider à la cohérence temporelle nécessaire de l'image à travers l'animation. Cette analyse fréquentielle aide également à l'intégration spatio-angulaire adaptative de la radiance, dépendante de statistiques du matériau, de l'illumination et de la visibilité en chaque point.

Résumé. Nous proposons une méthode pour le rendu de scènes animées et illuminées par éclairage distant direct, qui n'a besoin de calculer qu'une fraction des pixels de l'écran. Nous prenons avantage d'analyses en espace fréquentiel du transport de la lumière afin de déterminer les taux d'échantillonnage nécessaires au fil d'une animation par le biais d'un cache d'échantillons. Pour cela, nous dérivons des bandes passantes fréquentielles prenant en compte la complexité de la lumière distante, de la BRDF, et de la cohérence temporelle durant l'animation. Nous appliquons finalement un filtre bilatéral pour la reconstruction de l'image finale, utilisant les sets épars de points de radiance calculés et placés en accordance avec nos oracles fréquentiels–habituellement $< 25\%$ des pixels, par image.

Mots clés : Techniques Monte Carlo, Echantillonnage et reconstruction, Mise en cache de radiance

Abstract. We propose a method to render animation sequences with direct distant lighting that only shades a fraction of the total pixels. We leverage frequency-based analyses of light transport to determine shading and image sampling rates across an animation using a samples cache. To do so, we derive frequency bandwidths that account for the complexity of distant lights, visibility, BRDF, and temporal coherence during animation. We finally apply a cross-bilateral filter when rendering our final images from sparse sets of shading points placed according to our frequency-based oracles (generally $< 25\%$ of the pixels, per frame).

Keywords: Monte Carlo Techniques, Sampling and Reconstruction, Radiance Caching

1. Introduction

In physically based rendering, pixel colors are computed by estimating a multi-dimensional integral. Pixels are usually computed independently, leading to potential redundancies. In the case of animations, where variation *across* frames may only change sparsely, the probability of performing redundant computation only increases. To exploit this redundancy, previous work has targeted more efficient computation of *individual* pixels by improving the underlying numerical integration routines (e.g., path-integral and density-estimation approaches [**15, 14**] and importance sampling [**37**]), or by amortizing computation *across image regions* using adaptive methods based on predictive models of light transport (e.g., caching, interpolants and filters derived from frequency-space [**10**] or first-order [**29**] analyses).

We propose an adaptive approach to render animations with complex direct reflection and shadows from environment lighting. We perform adaptive sampling in image space, as well as during the numerical integration of each image sample. We further amortize shading cost by reusing previously computed image samples across the animation in a conservative

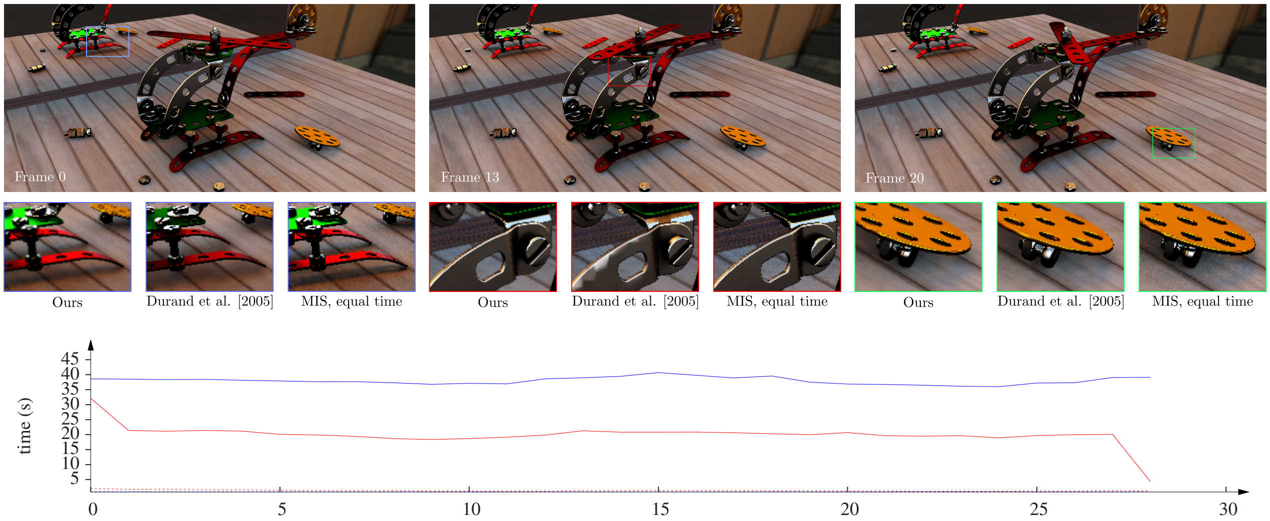


Fig. 5. For an equal quality target, our method (red curve) achieve almost two time faster rendering time for a better rendering quality than the method of Durand et al. [**10**] (blue curve). Insets from different images of the sequences (top) show how our method allows to reconstruct images with much less artifacts and less noise than Durand et al. [**10**], and an equal-time MIS integrator [**37**]

manner, based on frequency analysis of light transport. To do so, we devise a caching scheme to store image samples using spatial, directional and temporal frequency bandwidths. Our intuition is that frequency bandwidths of moving objects or viewpoints can be translated into static spatio-angular bandwidths. For cases where this assumption fails (i.e., moving occluders), we derive tailored conservative estimates.

Our oracles builds on frequency-based light transport analysis [**10**], and we additionally leverage them during final image reconstruction. We are able to render images using few image samples. Our frequency-analysis extends previous work to additionally treat temporally-varying occlusion changes, to support animation sequences using a lightweight caching scheme.

We perform rendering by applying three simple concepts :

(1) we shade only a carefully chosen *sparse* subset of pixels,

(2) when doing so, we adapt the spherical integration sampling rate according to variations in lighting, BRDF, and occlusion, and

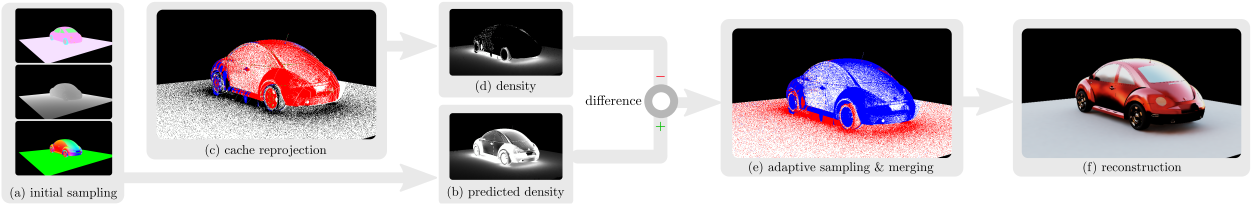


Fig. 6. From an initial sparse image-space sampling ((a), showing part of our extended G-buffer) we estimate the necessary sampling density for artifact free reconstruction (b). We then reprojection & validate samples from our temporal cache ((c), in red) and compute the sampling density from this first set (d) and subtract it from (b). We distribute samples according to this difference in density (e) and reconstruct the final shading (f) using our frequency-space oracles. New samples are added to the cache for the next frame.

(3) for animations, we further reduce the image sampling rates by reusing samples from previous frames according to our frequency analysis, properly accounting for shading and occlusion variations over time.

Specifically, our work consists of the following contributions :

- a frequency analysis of sample placement and reuse, within and across frames, that amortizes shading computations over space, time, and integration domains,
- an adaptive integration scheme based on material, lighting, and visibility statistics devised from our new frequency-analysis,
- a practical lightweight caching scheme that reuses, discards, and recomputes shading and occlusion information over time, according to our frequency-analysis, in order to control error.

2. Previous Work

Frequency Analysis of Light Transport. Durand et al. [**10**] proposed a frequency analysis of local lightfields for surface based shading. They applied it to a proof-of-concept adaptive image sampling and reconstruction application, where the numerical integration of the shading integral at each sample location had already been computed to (visual) convergence. This seminal work has promoted significant work on adaptive sampling and filtering techniques that rely on local frequency analysis for rendering various effects including unshadowed environmental shading [**1**], distribution effects [**13, 34, 5, 22**], soft shadows from geometric sources and ambient occlusion [**11, 12, 23**], and diffuse indirect illumination [**24**]. We are also motivated by this seminal work and try to extend it to the problem of adaptively rendering animation sequences. Like most of these work, we limit ourselves to the rendering of opaque surface and let aside the problem of rendering participating media.

Lightfield Reconstruction. Lehtinen et al. [**19, 20**] reconstruct static images from a sparse set of lightfield samples, leveraging the structure of continuous lightfield space. These methods assume smoothness along reconstruction directions in the lightfield and rely on the user provided sampling rate to be adapted to the integrand's bandlimit. On the contrary, we determine adequate sampling rates, and as such are better suited for higher frequency variations.

Deep Image Filtering. A related set of techniques [**30, 9, 32, 21, 6**] apply denoising filters directly to rendered images, using custom feature-space metrics, computed from unconverged path traced simulation. Instead of leveraging structure in the high-dimensional lightfield, these approaches formulate final rendering as a signal reconstruction problem ignoring, for the most part, higher-order structures, coherence or frequency-content of the shading. Still, they perform well in many complex scenarios. In certain cases, a limited form of temporal filtering is supported, but only between adjacent frames; our caching scheme adapts over an entire animation sequence.

Caching and Temporal Coherence. Irradiance and radiance caching [**41, 18**] model local variations in indirect lighting to place sparse cache samples in a scene, and then compute smooth shading from the samples using first- or second-order [**31**] interpolants. Again, here the shading integral is computed independently at each sample (without any adaptivity), and temporal coherence is not handled. Bala et al. [**2**] devise radiance interpolants and cache re-use oracles based on error estimate bounds on (potentially reusable) shading samples, and the *render cache* system [**39, 3, 38**] reprojects previous shading samples for interactive preview. We also reproject cache samples for temporally coherent animations, but instead leverage frequency bandwidth estimates that take local geometry variation, spatial and temporal (spherical and camera) occlusion variations, reflectance and lighting changes into account. Meyer and Anderson [**25**] used a smooth basis to reproject stochastic samples both in space and time for smooth indirect illumination effects; our approach instead treats much sharper features not amenable to smooth, generalized basis-space techniques.

3. Overview

Our goal is to render still images and animated sequences, with all-frequency shadows and reflections from direct environmental illumination, without dense sampling of pixels nor of spherical integration samples for the *outgoing radiance at each pixel* (later refered as "*radiance points*"). To do so, we leverage frequency analysis to adapt the sampling in both of these spaces. For animations, we also build a lightweight cache of radiance points on-the-fly in object space in order to further reduce sampling cost. This cache is maintained (discarding / adding elements) based on frequency criterion. At each frame, we reproject in screen space radiance points from the cache and ensure that our target sampling rate, in screen space, is met by resampling only were needed. Our approach works in four steps (see Figure 6 and Algorithm 1):

1. we perform an initial light sampling of radiance points to estimate the frequency bandwidth metrics we will use to drive our final sampling rates (Figure 6(a));
2. we reproject radiance points stored in a cache, according to their spatial, angular and occlusion-aware temporal frequency bandwidths, to further prime our adaptive sampling scheme (Figure 6(c));
3. using the information gathered in #1 and #2, we adaptively sample screen space (Figure 6(e)) according to the difference between the conservative sampling density determined by our frequency analysis (Figure 6(b)) and the density of radiance points reprojected from previous frames (Figure 6(d)), potentially adding new radiance points in undersampled pixel regions. Newly sampled radiance points are computed using an adaptive spherical integration approach; and,
4. finally, we reconstruct the final image(s) using a frequency bandwidth-driven cross-bilateral filter (Figure 6(f)).

We distinguish ourselves from previous techniques by:

- coupling adaptive image/object-space sampling (Section 5), spherical sampling (Section 4) and final image reconstruction, using oracles devised from our frequency analysis (Section 6),
- accounting for shading variations due to camera and object motion in our bandwidth computation to, e.g., adaptively sample all-frequency shadows and reflection effects,

accounting for complex occlusion and appearance variations in animated sequences (Section 5), and by

- exploiting coherence in the spatio-angular outgoing radiance with a lightweight cache of frequency-space metrics (Section 5).

Algorithm 1: Our adaptive sampling & reconstruction method.

```
input  : Empty deep image $img$; Current cache $cache$; Viewpoint $v$
output : Reconstructed image buffer $img$
// Uniformly sample screen space (Section 5)
$pts$ = initialPixelSampling($v$)
// Compute radiance with adaptive integration at these positions
// and accumulate visibility statistics $\{\mu_v, \sigma_v\}$ (Section 4)
computePointsRadiance($pts$, $v$)
// Save points for cache insertion after image generation
savePointsInCache($cache$, $pts$)
// Cache point verification and reprojection (Section 5)
for $c$ in $cache$ do
    if isValidEntry($c$, $v$) do $pts$.add(c)
    else $cache$.discard($c$)
end
// For each pixel, reconstruct occlusion statistics and estimate
// the required sampling density (Section 4)
$N$ = computeSamplingDensity($pts$, $v$)
$npts$ = sampleImageFromDistribution($N$, $N_{\max}$)
// Compute the radiance at sampled positions and accumulate
// visibility statistics (Section 4)
computePointsRadiance($npts$, $v$)
// Reconstruct final image (Section 6)
$img$ = upsample($npts$, $v$)
```

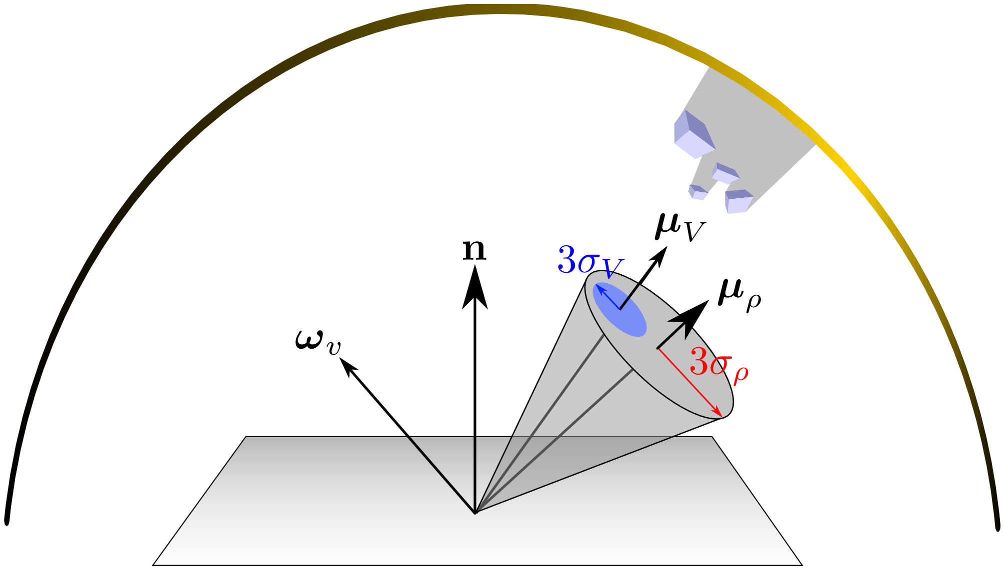


Fig. 7. We only consider the frequency content of the (distant) incident lighting and visibility within the view-evaluated *BRDF footprint* (in yellow), defined as a cone centered about the mean reflection direction of the BRDF slice $\boldsymbol{\mu}_\rho$ with radius equal to three standard deviation σ_ρ of the BRDF lobe. Our visibility statistics, comprised of its mean $\boldsymbol{\mu}_V$ and three standard deviation σ_V, are also accumulated exclusively within this footprint.

4. Adapting the Spherical Sampling Rate

Our approach adaptively places *radiance points* in a scene (and across animation frames) and, for each one of these point we use a spherical integration scheme that adapts the sampling rate when computing the outgoing radiance estimate.

A *radiance point* corresponds to the outgoing radiance towards a viewing direction $\boldsymbol{\omega}_v$, at a 3D position $\mathbf{p}$, based on the reflection equation [**16**]:

$$L(\mathbf{p}, \boldsymbol{\omega}_v) = \int_{\mathcal{H}^2} L_\infty(\boldsymbol{\omega}_l) V(\mathbf{p}, \boldsymbol{\omega}_l) \rho(\mathbf{p}, \boldsymbol{\omega}_v, \boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n}) d\boldsymbol{\omega}_l, \tag{4.1}$$

computed using a Monte Carlo estimator (see details below). Here, we restrict ourselves to direct illumination from distant environment/area sources L_∞, where ρ is the BRDF, V is the binary visibility function and $\mathcal{H}^2$ is the hemisphere of unit directions at $\mathbf{p}$ about the surface normal $\boldsymbol{n}$.

One of our goals is to accurately compute radiance points, with minimum computation, by adapting the hemispherical sampling rate to the complexity of the integrand. Given a bandwidth estimate B for our integrand, we apply Shannon's sampling theorem to determine the spherical sampling rate $N_\Omega = 4B^2$ used for Monte Carlo integration[1]. We remark that the integral in Equation 4.1 can be interpreted as a windowing of the incident lighting by a filter comprised of the product of the BRDF and cosine term [**27**]. Consequently, incident light and visibility frequencies outside of this window will not contribute to the integral and, so, should not be considered when determining the spherical sampling rate.

BRDF Cone. When computing Equation 4.1 we first estimate the BRDF's footprint defined by its view-dependent lobe centered about its mean reflection direction $\boldsymbol{\mu}_\rho$ and of standard deviation σ_ρ, simplifying its formulation by assuming the lobe radially symmetric (see Figure 7). For a Lambertian BRDF, $\boldsymbol{\mu}_\rho$ is the surface's normal and the whole hemisphere is considered. For rough microfacet-based BRDFs, a good approximation for $\boldsymbol{\mu}_\rho$ is the mirror reflection of the view direction $\boldsymbol{\omega}_v$ and σ_ρ should be proportional to the surface roughness e [**4, 35**]. We derive σ_ρ for a Phong BRDF of roughness e from the variance of its signal in the 2D plane orthogonal to $\boldsymbol{\mu}_\rho$ defined as :

$$Var[e] = \int_{t \in \mathcal{R}} \frac{t^2}{(1+t^2)^{e/2}} dt = \frac{\sqrt{\pi}\ \Gamma[\frac{e-3}{2}]}{2\ \Gamma[\frac{e}{2}]} \text{ for } e > 3 \tag{4.2}$$

In practice, we approximate the gamma functions in Equation 4.2 with the convergent version of Stirling's formula. We found that the first three terms of the serie were enough for the precision we need. The approximation is computed once per BRDF at initialization time. For microfacet BRDFs, we first find the Beckmann-equivalent shininess such as $e_{MF} = \sqrt{2/(2 + e_{Phong})}$ [**40**].

It is reasonable to treat any BRDF variation within the cone as negligible compared to that of the incident light (see Figure 7), and so we must estimate the incident lighting's frequency content *within* the directional cone footprint in order to adapt the spherical sampling of our Monte Carlo estimator of Equation 4.1.

Incident Lighting Bandwidth. To estimate the lighting bandwidth within the BRDF's cone, we pre-compute the environment light's local bandwidth for several discrete cone (window) sizes σ_ρ and for several discrete cone directions $\boldsymbol{\mu}_\rho$ [**1**]: this data is stored in a mip-like hierarchy of spherical textures (see Figure 8). We used cosine-windowed Fourier transforms to estimate the bandwidth since they most closely match our BRDF-windowed integration footprint profile. For different levels in the hierarchy, each cosine window corresponds to

---

[1] We square the 1D bandwidth to obtain a conservative 2D sampling rate.

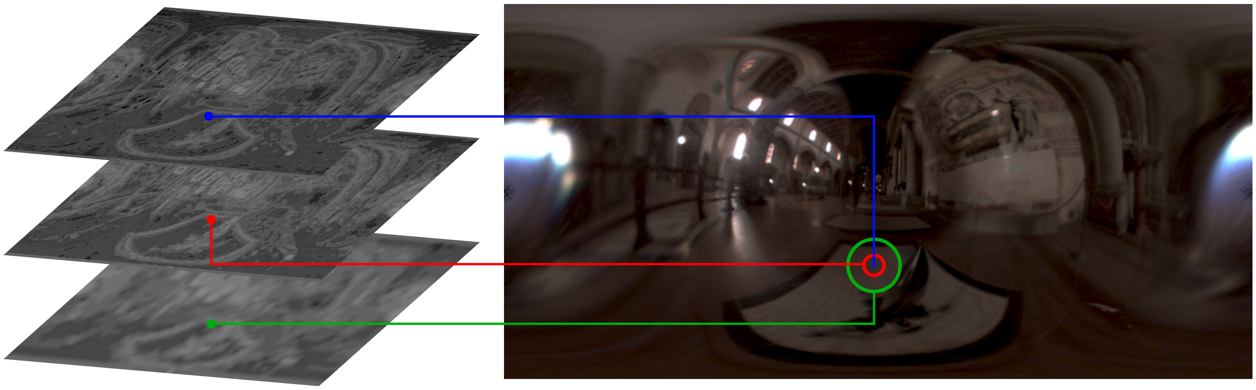


Fig. 8. We precompute the distant environment light's bandwidth for multiple footprint sizes (left), at discrete footprint directions. During rendering, we query this structure to obtain the lighting's bandwidth within the BRDF's footprint (right).

Phong lobe with a shininess proportional to the window size, and computing this structure must only be done once for each environment map. During spherical integration, we query this structure based on the BRDF footprint direction $\boldsymbol{\mu}_\rho$ and size $3\sigma_\rho$ in order to obtain the lighting bandwidth B_{L_∞}.

While we could proceed with $B = B_{L_\infty}$ and use pre-filtering methods similar to previous work [**27, 28, 17, 1**] to query the incident lighting with a sole sample, we want to additionally account for the occlusion in the integration cone, and thus have to use actual sampling to estimate our bandwidth.

Accounting for Visibility. It is impractical to *directly* incorporate the visibility's bandwidth into the integrand's bandwidth estimate B, since piece-wise constant (binary) functions in the primal (i.e., spherical) domain have infinite frequency bandwidth.

Instead, we motivate our solution by considering two spherical regions of integration, one that is largely occluded and another that is not: we note that the spherical sampling rate should not be modified in the unoccluded region since the sample rate already accounts for the frequency of the integrand, and these regions will contribute the most to the integral; we do, however, want to increase the sampling rate in occluded regions within the footprint, since any occlusions will likely increase the frequency bandwidth. As such, we opt to modulate the sampling rate determined by B_{L_∞} by a factor f_V that accounts for the amount of occlusion in the solid angle subtended by the cone of integration, as follows:

$$N_\Omega = f_V + 4B_{L_\infty}^2. \tag{4.3}$$

To determine the amount of occlusion, we accumulate a statistical representation of visibility during a first Monte Carlo estimation of Equation 4.1, which we compute using multiple importance sampling (MIS) [**36**] and $N_L = 4B_{L_\infty}^2$ spherical samples. Our MIS implementation distributes samples evenly according to the *pdf*s of the environment light $p_L(\boldsymbol{\omega}) \propto L_\infty(\boldsymbol{\omega})$ and the view-evaluated BRDF $p_\rho(\boldsymbol{\omega}) \propto \rho(\mathbf{p}, \boldsymbol{\omega}_v, \boldsymbol{\omega})$. During integration, we progressively accumulate the weighted arithmetic mean visibility vector $\boldsymbol{\mu}_o$ and 3×3 visibility covariance matrix Σ_o within the BRDF's footprint,

$$\begin{aligned}\boldsymbol{\mu}_o &= \frac{1}{A_o}\sum_{i=1}^{N_L} \boldsymbol{p}_i \rho_i (1 - v_i),\\ \Sigma_o &= \frac{N}{(N_L - 1){A_o}^2}\sum_{i=1}^{N_L} (\rho_i(1 - v_i))^2 (\boldsymbol{p}_i - \boldsymbol{\mu}_o)(\boldsymbol{p}_i - \boldsymbol{\mu}_o)^T,\end{aligned} \tag{4.4}$$

which we will use to adapt our spherical sampling rate. Here, the $\boldsymbol{p}_i$ are the first hit-points visible from the shading position to the i^{th} Monte Carlo (2D spherical) integration sample's direction $\boldsymbol{\omega}_i$ (expressed in the local coordinate frame about the BRDF cone's central direction

Car scene

without f_V

with f_V

Fig. 9. We increase the number of spherical integration samples to properly account for shading variations due to occlusion. This reduces the error (right insets) with respect to the ground truth.

$\boldsymbol{\mu}_\rho$) hitting a finite distance occluder, $v_i = V(\boldsymbol{\omega}_i)$ is the binary visibility evaluated at $\boldsymbol{\omega}_i$ and we weight the value by the view-evaluated BRDF's *pdf* evaluated in the sampling direction $\rho_i = p_\rho(\boldsymbol{\omega}_i)$ to avoid considering visibility samples in regions outside the BRDF footprint (i.e., regions of little contribution to the final integral); finally, $A_o = \sum_{i=1}^{N} \rho_i(1 - v_i)$ is the normalization of BRDF weights for the occluded samples ($v_i = 0$).

The visibility's standard deviation σ_V corresponds roughly to an occlusion coverage measure *within* the BRDF's footprint (see Figure 7) and is derived as

$$\sigma_V = max(\sqrt{(\widehat{\boldsymbol{\mu}_{\rho_t}})^T\, \Sigma_o\, \widehat{\boldsymbol{\mu}_{\rho_t}}}, \sqrt{(\widehat{\boldsymbol{\mu}_{\rho_b}})^T\, \Sigma_o\, \widehat{\boldsymbol{\mu}_{\rho_b}}}\ ), \tag{4.5}$$

where $\widehat{\boldsymbol{\mu}_{\rho_t}}$ and $\widehat{\boldsymbol{\mu}_{\rho_b}}$ are the unit tangent and binormal vector of the local coordinate frame around $\boldsymbol{\mu}_p$. We use the ratio of this coverage to the size of the BRDF footprint to estimate how many more samples are needed as $f_V = f({}^{\sigma_V}/_{\sigma_\rho})$. We experimented with various easing functions and found that a Gaussian profile, $f(x) = (N_{\max}^{\Omega} - N_\Omega)\, e^{-1/2(x-1/2)^2}$ yields good results, where $N_{\max}^{\Omega}$ is our maximum spherical sampling budget (Figure 9). We use the remaining integration samples to improve our Monte Carlo estimate of Equation 4.1 (with the same MIS estimator).

We use the frequency content of the BRDF and lighting, as well as occlusion statistics, to adapt the spherical integration cost for each radiance point. To further reduce render time, we will both reuse information from radiance points computed in previous frames (Section 5), as well as reconstructing the final image using only a sparse set of image-space radiance points (Section 6).

5. A Sparse, Adaptive Radiance Cache

When rendering an image, we first uniformly distribute a small number of radiance points in image-space in order to gather the required information to compute the optimal sampling density: we estimate the outgoing radiance's spatio-angular frequency bandwidth (Section 5.1). We then reuse radiance points from previous frames, leveraging an object-space cache (Section 5.2). Shading variations caused by camera and/or object motion can invalidate radiance points stored in the cache, and so we yet again leverage our frequency analysis to appropriately handle cache sample invalidation, resampling, and reuse (Section 5.3) in a manner that maintains image fidelity and temporal coherence.

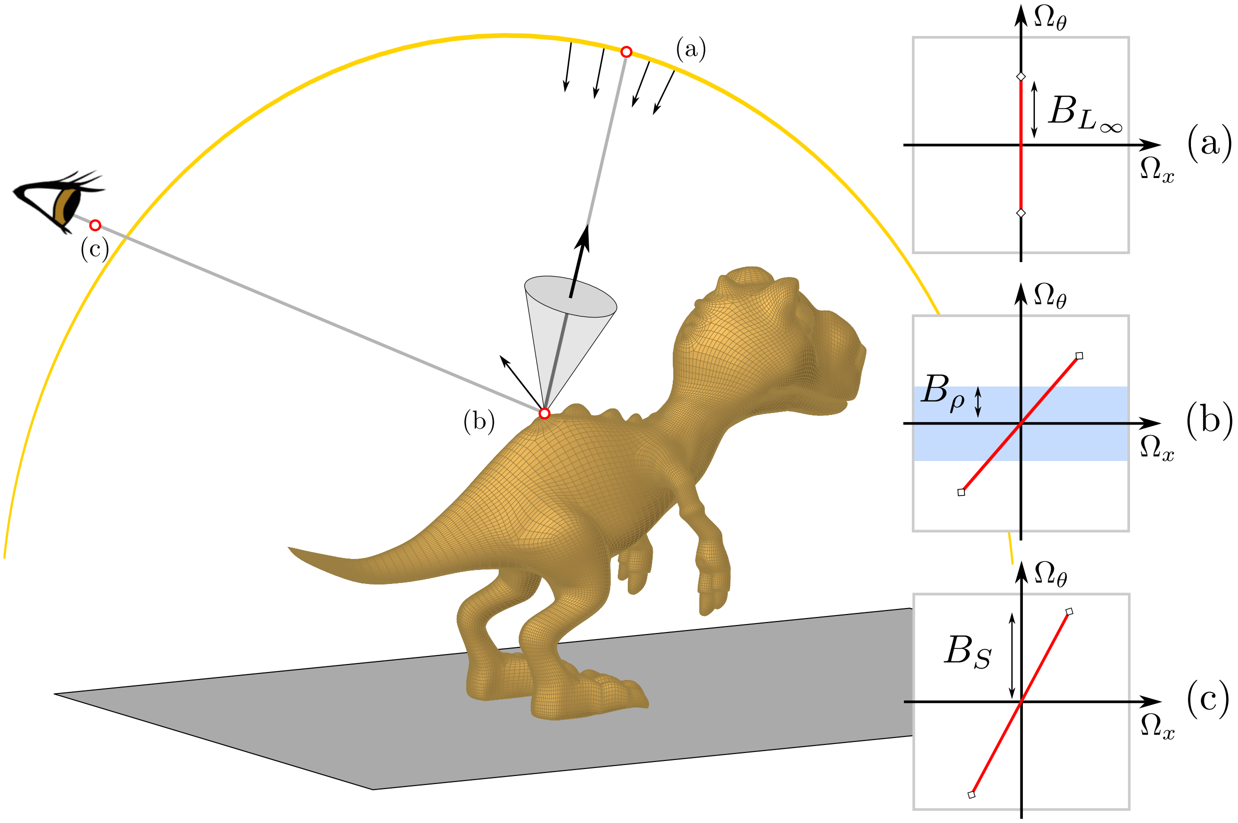


Fig. 10. To estimate the screen-space bandwidth B_S, we start with the emitted (angular) bandwidth of the environment light B_{L_∞} reduced to the BRDF's footprint (a), we then apply the BRDF bandlimit B_ρ (b) and modulate it according to the curvature, foreshortening, and spatial transport [**10**] (c). The angular bandwidth serves directly as the screen-space bandwidth estimate.

5.1. Adaptive Image Space Sampling

The spatio-angular bandwidth of the outgoing radiance in image-space will be used to determine the (screen-space) density of radiance points required for artifact-free image reconstruction (discussed in Section 6). To compute this bandwidth we proceed similarly to previous work [**10, 1**] as illustrated in Figure 10: beginning with the bandwidth B_{L_∞} emitted within the solid angle of the (distant) light subtended by the BRDF's footprint (Figure 10(a)), we analyze the reflected bandwidth (Figure 10(b)) accounting for the local visiblity and BRDF's band-limit B_ρ, to estimate the bandwidth of the shade point at the sensor location B_S (Figure 10(c)).

Frequency analysis methods typically operate either on the bandwidth B or the variance σ^2 of the spectrum. We interchange between these two measures in our discussion using 3 standard deviations (corresponding to the 99$^{\text{th}}$ percentile of a Gaussian) as the bandwidth of the spectrum from its variance: $B \approx 3\sqrt{\sigma^2}$. We use a compact 2D variance representation [**5**] for our spatio-angular bandwidths, $\sigma^2 = \{\sigma_x^2, \sigma_\theta^2\}$, where σ_x and σ_θ are the spatial and (isotropic) angular bandwidths of the outgoing radiance field.

While accumulating the mean occlusion direction $\boldsymbol{\mu}_o$ and covariance Σ_o during radiance point integration (Section 4), we also compute the mean and variance of occluders' distance in the BRDF cone, similarly to Equations 4.4. We project the occluders' hit information on the mean reflection direction of the BRDF $\boldsymbol{\mu}_p$ instead of on its local tangents. We thus get a mean occlusion distance of $\mu_t = \|\boldsymbol{\mu}_o\|$ and a standard deviation of $\sigma_t = \sqrt{(\widehat{\boldsymbol{\mu}_\rho})^T \, \Sigma_o \, \widehat{\boldsymbol{\mu}_\rho}}$. We approximate the *minimum* distance to occluders as

$$t_{min} = \mu_t - 3 \times \sigma_t. \tag{5.1}$$

Moreover, assuming a pinhole camera, the bandwidth incident on the aperture is equal to the screen-space bandwidth we will use for sampling. And so, the screen-space spatio-angular 2×2 covariance matrix is computed as

$$\Sigma_{S,\theta} = \mathbf{T}_{\mathbf{x}\to\mathbf{v}} \Big(\mathbf{C}_v \circ \mathbf{B}_{\rho_s} \circ \mathbf{C}_L \Big) \Big(\sigma_{L_\infty}^2 + \mathbf{T}_{V\to\mathbf{x}}(t_{min}) \Big) \ , \tag{5.2}$$

where each of the five operators above are simply 2×2 matrices [**1**] (see Appendix 9 for details): $\mathbf{C}_L$ accounts for the local curvature and cosine factor between $\boldsymbol{n}$ and the mean BRDF direction $\boldsymbol{\mu}_\rho$; $\mathbf{B}_{\rho_s}$ bandlimit the bandwidth according to the view-evaluated BRDF; $\mathbf{C}_v$ accounts for the mirror reflection, curvature and cosine foreshortening between the shade point and the viewpoint; $\sigma_{L_\infty}^2$ is estimated using the bandwidth queried from our hierarchical structure; and $\mathbf{T}_{\mathbf{x}\to\mathbf{v}}$ transports the bandwidth from the shade point to the eye. When accounting for occlusion, an additional transport operator $\mathbf{T}_{V\to\mathbf{x}}$ warps the spatio-angular occlusion bandwidth according to the *minimum occluder distance* between the shade point and the light as derived in Equation 5.1 [**10, 11**].

From the covariance matrix $\Sigma_{S,\theta}$ we retrieve the angular part σ_θ^2 defining our final scalar image-space variance, and derive its bandwidth as stated above as

$$B_S \approx 3\sqrt{\sigma_\theta^2}. \tag{5.3}$$

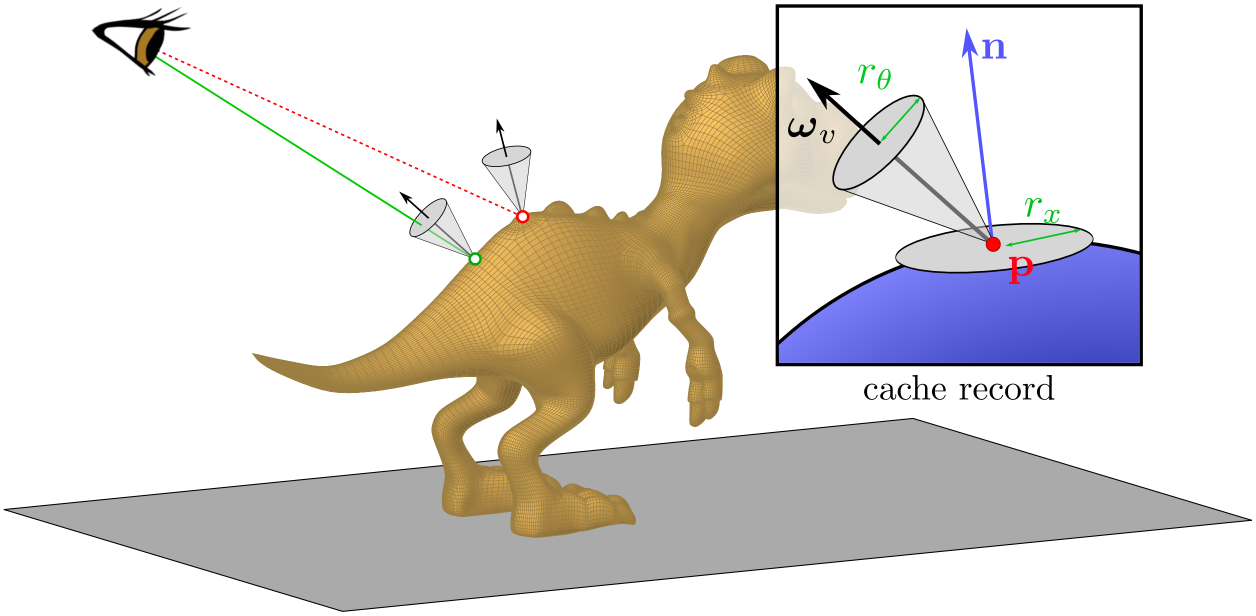


Fig. 11. During reprojection, we test whether the new viewing direction remains inside the BRDF's footprint and, if so, we project the sample onto the screen (plain green line); if not, we (optionally) discard the sample from the cache (dashed red line).

Sampling Density and Radiance Point Placement. We first uniformly sample pixels in the image plane with a low number of samples per pixel to compute per pixel visibility statistics. The number of directional samples used is proportional to the BRDF cone apex. We apply a simple depth- and normal-aware bilateral upsampling filter [**33**] with filter sizes proportional to the pixel's unoccluded angular variance (see Appendices 9.1, 9.2) to remove noise from the bandwidth and visibility statistic values, $\{\boldsymbol{\mu}_o, \Sigma_o\}$. We use these values to compute the screen-space bandwidth B_S using Equations 5.2 and 5.3, and determine the final pixel sampling densities by clamping the bandwidth between zero and one (see Section 7) which we use to sample the image space by rejection sampling : each pixel receiving one unique sample if a random value between zero and one falls lower or equal than the corresponding pixel's sampling density, ensuring that a density of one always yields one and only one sample and one of zero will never be sampled.

As detailed below in Section 5.2, we can further reduce the rendering cost by only generating new radiance points in screen-space where reprojected (valid) radiance point samples from our cache do not satisfy our pixel sampling density requirement. When introducing a new radiance point, we always perform spherical integration using the approach in Section 4.

5.2. Caching and Reusing Radiance Points

During animation, the number of new radiance points computed each frame can be significantly reduced if we carefully reuse radiance points from previous frames. As such, we propose an object-space cache to store, update, invalidate, and reuse (when suitable) radiance points across frames of an animation. Every scene object gets an associated cache for easier management of its associated samples. Each cache entry consists of a tuple comprising a 3D position, 2D viewing direction, surface normal, radiance value, time at which the sample was originaly created, boolean flag identifying moving samples, spatial bandwidth B_x, angular bandwidth B_θ, a temporal bandwidth B_t (see Section 5.3), and the occlusion ratio used to define f_V. The spatial, angular and temporal bandwidths (B_x, B_θ, B_t) (see derivation in Appendix 9) are used to define a circular surface patch, a cone of directions (see Figure 11, inset), and a time interval within which the sample's radiance value is known to remain close to the correct value. In the case where the associated object is a specular mirror, the tuple receives an additional 3D position, normal and object ID describing the reflected object from the view direction it was sampled from. Three render-time parameters $q_{\epsilon_{\{x,\theta,t\}}}$ are used (see below) to control the accuracy of our method.

After our initial sampling, but before computing the current pixel sampling density, we update (based on object motion) and reproject radiance points stored in our cache onto the image plane. We only reuse these points for final image reconstruction if:

(1) they pass a z-buffer camera visibility test, and

(2) their radiance value remains *valid* after reprojection.

The *validity* of reprojected radiance points is determined according to spatial, angular and temporal bandwidth tests in order to control the bias introduced in the final rendering (as detailed below and in Figure 11). Reprojected radiance points that do not pass these tests are not considered for reprojection nor reconstruction, and are flagged for discard from their respective cache (as illustrated in Figure 12(a)).

Namely, object motion affects cache point validity in four ways:

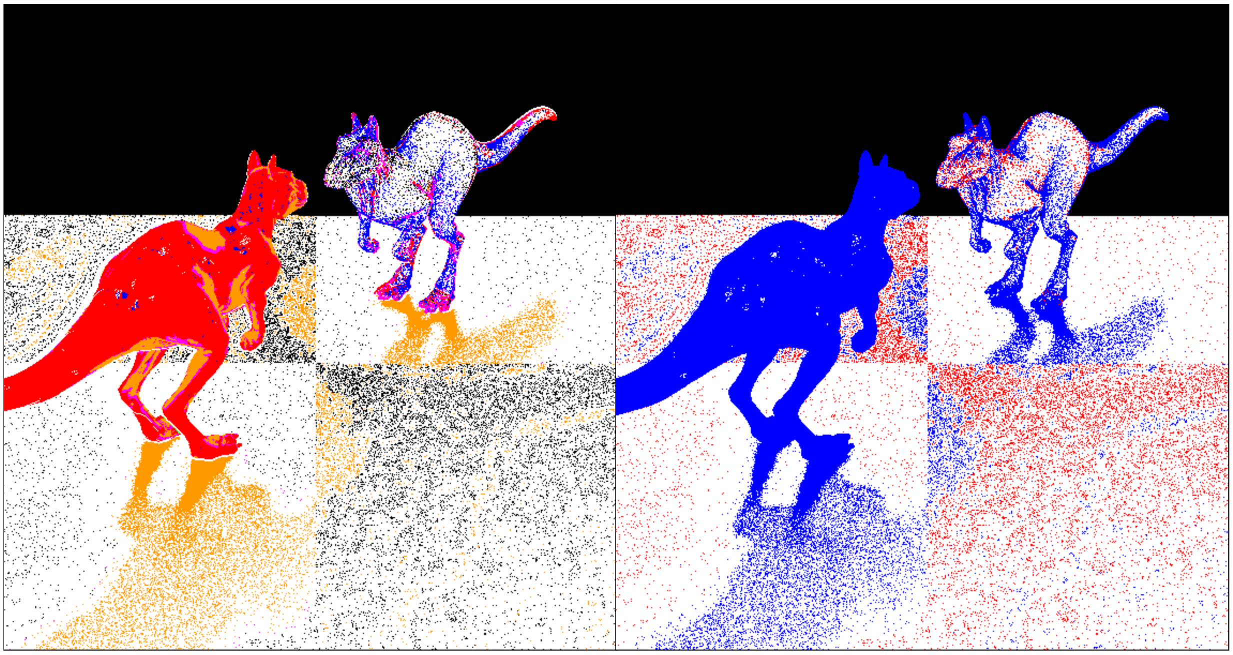
(1) camera and object motion can lead to failed z-buffer (i.e., camera visibility) tests during reprojection,

(2) object motion can influence the validity of cached radiance points that lie *on the surface* of the moving object,

(3) temporal changes of distant lighting affects the product of the light and BRDF during integration, and

(4) object motion can influence the accuracy of radiance points due to changes in the spherical visibility *on all other objects*.

We describe the latter case (visibility changes) in Section 5.3 and discuss here the remaining points.



(a) Cache discard & reuse (b) Resampling

Fig. 12. Left, we visualize the previous frame's reused (in black) and discarded radiance points (in different colors). Invalid samples due to B_x are blue, to B_θ red, to B_t orange, and to occlusion ratio are purple; Right, we show the number of reused and resampled radiance points (in red and blue resp.) used for reconstruction.

Validating Reprojected Cache Points. We use the spatial, and angular bandwidths to define a circular surface patch and directional cone with radii $r_x = {}^{2\pi\epsilon_x}/_{B_x}$ and $r_\theta = {}^{2\pi\epsilon_\theta}/_{B_\theta}$, where the directional cone is centered along $\boldsymbol{\omega}_v$ (at the time of the radiance point's insertion into the cache), $\epsilon_{\{x,\theta,t\}} = \text{acos}(1 - q_{\epsilon\{x,\theta,t\}})$ is based on quality thresholds $q_{\epsilon\{x,\theta,t\}}$ corresponding to the maximum relative error a cached point can cause during reprojetion. Validating a reprojected radiance point corresponds to z-buffer testing the pixel onto which it projects (for camera occlusion) and ensuring that the view vector is inside its directional cone (Figure 11).

To treat temporal changes in the radiance value due to lighting and occlusion motion, we add a temporal validity period to each cache point based on temporal bandwidth $\Delta t = {}^{2\pi\epsilon}/_{B_t}$. We apply Egan et al.'s [**13**] bandwidth estimate to treat variations in lighting rotation, and we derive temporal bandwidths for changes in radiance due to occlusion motion in Section 5.3. The final temporal bandwidth B_t is the maximum of these two bandwidths.

Image Resampling and Cache Updates. Once reprojected, cache samples contribute in decreasing the per-pixel density in their screen-space reprojection neighbourhood. Each reprojected sample is splatted on screen in a manner similar to the final reconstruction (Section 6) and their derived screen-space bandwidth is subtracted from B_S. This difference density is used to distribute new samples with the same rejection sampling rules as the empty cache case (see Figure 12(b)). Given the increased start-up density due to cache reprojection, we are more likely to introduce samples in areas that are not covered by the reprojected sampling.

Cache Point Discard and Merging. Each newly generated radiance point is added to its corresponding cache and uses the spherical sampling scheme of Section 4. If two cached radiance points reproject onto the same pixel they may be merged if their object IDs are identical and if their spatial and angular bandwidths agree with the angular and spatial distances between them (Figure 13). If a radiance point reprojects outside the view frustum, it is also flagged for discard. Once per frame, each cache will remove in a batch all the flagged-as-discard entries and set their slot as open again for further entries to be added on future frames. The cache memory size increases if no open slot is available for this object, which happens implicitly for the first frame, and less regularly during the rest of an animation. Cache size is grown by sample batches and (re)allocations can thus occur a maximum of once per object per frame. Each cache is grown at least once (for the first frame) by the amount of slots the sampler gave each object for the initial frame.

5.3. Handling Temporal Occlusion Changes

Modeling the impact of object motion on the scene visibility is challenging. Starting from a time-dependent visibility formulation of the reflection equation (Equation 4.1), we will perform a frequency analysis of changes in the outgoing radiance due to changes in visibility. We will show that the temporal bandwidth B_t of the change *in the final shading* can be modeled using the projected angular velocity of occluders and the distant light's bandwidth. We store this bandwidth B_t at each cached radiance point.

We consider the *relative* motion of occluders at a shade point in our time-dependent reflection equation (below; Equation 5.4), and assume that only the spherical visibility changes over time, but not the lighting nor reflectance:

$$L(\mathbf{p}, \boldsymbol{\omega}_v, t) = \int_{\mathcal{H}^2} L_\infty(\boldsymbol{\omega}_l) V(\mathbf{p}, \boldsymbol{\omega}_l, t) \rho(\mathbf{p}, \boldsymbol{\omega}_v, \boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n}) d\boldsymbol{\omega}_l. \tag{5.4}$$

Concurrently treating temporal variations in lighting and reflectance is a challenging problem that we leave to future work.

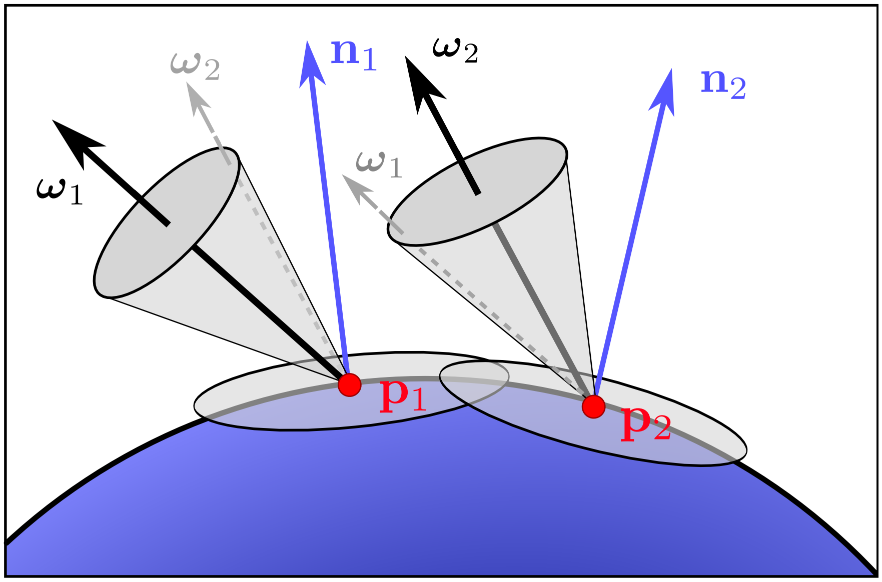


Fig. 13. We merge two cache points that reproject onto the same pixel only if both their spatial and angular bandwidths overlap enough. In this example, while the two cache point's angular cones ($\boldsymbol{\omega}_1$, $\boldsymbol{\omega}_2$) align (gray), their spatial bandwidths at $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ do not overlap enough. Consequently, the two points will not be merged.

Next, we model the *change* in outgoing radiance ΔL due to the motion $V(t)$ of a small moving occluder patch[2]; we will later generalize this model to large occluders. The outgoing radiance "removed" due to the (potentially moving) occluder at time t is

$$\Delta L(t) = \int_{\mathcal{V}_t} L_\infty(\boldsymbol{\omega}_l)\rho(\boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n})d\boldsymbol{\omega}_l\,, \tag{5.5}$$

where we denote the set of all occluded directions at time t as $\mathcal{V}_t = \{\boldsymbol{\omega}_l \mid V(\mathbf{p}, \boldsymbol{\omega}_l, t) = 0\}$. We can now rewrite the outgoing radiance at any moment of time as the sum of a temporally constant (unoccluded) component, and the change in outgoing radiance that does vary with time:

$$L(\mathbf{p}, \boldsymbol{\omega}_v, t) = \underbrace{[L(\mathbf{p}, \boldsymbol{\omega}_v, 0) - \Delta L(0)]}_{\text{constant w.r.t. time } t} + \Delta L(t)\ . \tag{5.6}$$

Here, $\Delta L(t)$ can be thought of as time-dependent antiradiance [**8**]. We perform a frequency analysis of this formulation in order to reason about the temporal bandwidth of changes to the outgoing radiance at a radiance point and, given this segmentation, the Fourier transform of the outgoing radiance is equal to the Fourier transform of the *change* in the outgoing radiance, $\widehat{L}(\Omega_t) = \widehat{\Delta L}(\Omega_t)$, as:

$$\begin{aligned}\widehat{L}(\Omega_t) = \mathcal{F}[L(\mathbf{p}, \boldsymbol{\omega}_v, t)] &= \mathcal{F}\Big[\,[L(\mathbf{p}, \boldsymbol{\omega}_v, 0) - \Delta L(0)] + \Delta L(t)\Big] \\ &= A\delta(\Omega_t) + \mathcal{F}[\Delta L(t)],\end{aligned} \tag{5.7}$$

where A is a constant offset we can ignore since it has a bandwidth equal to zero; as such, we only need consider the last term in our bandwidth derivation:

$$\mathcal{F}[\Delta L(t)] = \widehat{\Delta L}(\Omega_t) = \mathcal{F}\left[\int_{\mathcal{V}_t} L_\infty(\boldsymbol{\omega}_l)\rho(\boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n})d\boldsymbol{\omega}_l\right].$$

We will directly relate the Fourier transform of the (change in) outgoing radiance to the Fourier transform of the distant lighting, by assuming that the projected (spherical) area of the moving occluder $V(t)$ does not change w.r.t. time t and that the cosine-weighted BRDF is constant over the projected occluder patch; this is reasonable given our small occluder and small motion assumptions, and similar to reflectance-constancy assumptions used in previous work [**13, 11, 26**]. We additionally model the spherical angular motion of the occluder as a motion relative to the distant illumination, allowing us to rewrite the Fourier transform in Equation 5.7 as an integration of the occluding patch in its *original configuration* at $V(0)$,

---

[2] We occasionally omit location $\mathbf{p}$ and view $\boldsymbol{\omega}_v$ parameters for brevity.

but now lit under the distant lighting rotated according to the appropriate inverse (angular) rotation of the occluder motion:

$$\widehat{\Delta L}(\Omega_t) \approx \mathcal{F}\Bigg[\rho_n \int_{\mathcal{V}_0} L_\infty(\boldsymbol{\omega}_l + t\dot{\mathbf{v}}) d\boldsymbol{\omega}_l\Bigg], \tag{5.8}$$

where $\dot{\mathbf{v}}$ is the angular velocity of the occluder patch, we introduce a notational abuse of the $+$ operator to denote angular rotation of the lighting direction $\boldsymbol{\omega}_l$ for simplicity, and

$$\rho_n = \frac{1}{|\mathcal{V}_0|} \int_{\mathcal{V}_0} \rho(\mathbf{p}, \boldsymbol{\omega}_v, \boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n}) d\boldsymbol{\omega}_l \tag{5.9}$$

is the average cosine-weighted BRDF value over the occluder.

We can interchange the order of the integral and Fourier transform (due to linearity) in Equation 5.8, allowing us to express the Fourier transform of the (change in) outgoing radiance as the product of the Fourier transform of the lighting, the ratio of the average cosine-weighted BRDF and the angular occluder motion magnitude, and a phase term:

$$\begin{aligned}\widehat{\Delta L}(\Omega_t) &\approx \rho_n \int_{\mathcal{V}_0} \frac{e^{i\Omega_t(\boldsymbol{\omega}_l[\theta]+\boldsymbol{\omega}_l[\phi])}}{|\dot{\mathbf{v}}|} \widehat{L}_\infty(|\dot{\mathbf{v}}|\Omega_t) d\boldsymbol{\omega}_l \\ &= \frac{\rho_n}{|\dot{\mathbf{v}}|} \widehat{L}_\infty(|\dot{\mathbf{v}}|\Omega_t) \left[\int_{\mathcal{V}_0} e^{i\Omega_t(\boldsymbol{\omega}_l[\theta]+\boldsymbol{\omega}_l[\phi])} d\boldsymbol{\omega}_l\right], \end{aligned} \tag{5.10}$$

where the bracketed term is the phase component of the Fourier transform which we conservatively bound below.

We can bound the **bandwidth** of $\widehat{\Delta L}(\Omega_t)$ by analyzing the spectrum's amplitude, where the amplitude is $|\cdot| = \sqrt{\mathrm{Re}(\cdot)^2 + \mathrm{Im}(\cdot)^2}$; we first bound the amplitude of the phase term as the projected solid angle of the occluder V_Ω,

$$\left|\int_{\mathcal{V}_0} e^{i\Omega_t(\boldsymbol{\omega}_l[\theta]+\boldsymbol{\omega}_l[\phi])} d\boldsymbol{\omega}_l\right| \leq \underbrace{\int_{\mathcal{V}_0} d\boldsymbol{\omega}_l}_{V_\Omega}, \tag{5.11}$$

since $|e^{ix}| \leq 1$, and we can then bound the amplitude of $\widehat{\Delta L}(\Omega_t)$ in Equation 5.10 as

$$\left|\widehat{\Delta L}(\Omega_t)\right| \leq \frac{V_\Omega\, \rho_n}{|\dot{\mathbf{v}}|}\left|\widehat{L}_\infty(|\dot{\mathbf{v}}|\Omega_t)\right|. \tag{5.12}$$

Given this bound on the *amplitude* we can conservatively bound the *bandwidth* $B_{t,\text{single}}$ of the (change in) outgoing radiance due to a small occluding patch: since the temporal rotation (i.e., shift) in the primal domain of L_∞ in Equation 5.8 results in an scaling of the spectrum $\widehat{L}_\infty$ by a factor of $|\dot{\mathbf{v}}|$ in Equation 5.10, the lighting's original bandwidth B_{L_∞} is similarly

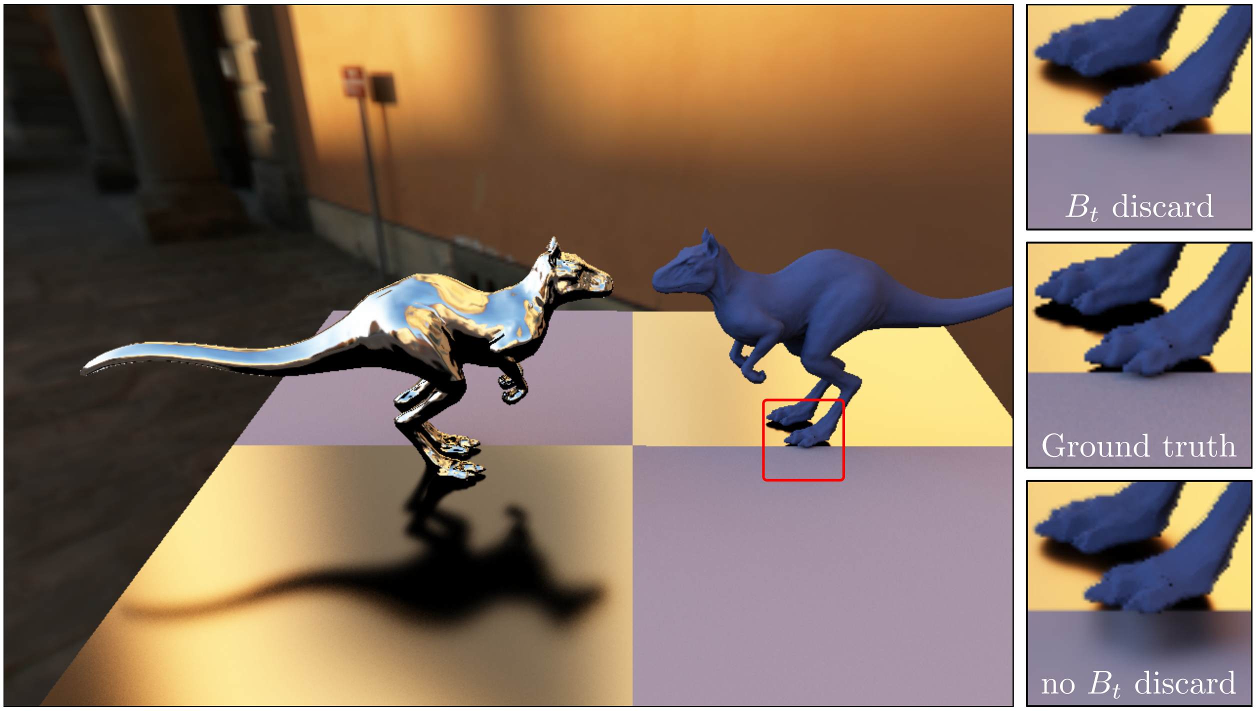


Fig. 14. We illustrate the need for our moving occluders bandwidth to discard samples during animations in the KILLEROO scene after 10 frames. Without this bandwidth, the cache incorrectly reproject shadows (bottom inset) onto the screen which result in a dark region in at the feet of the Killeroo.

scaled to $|\dot{\mathbf{v}}|B_{L_\infty}$ under the occluder's relative motion. Thus, the bandwidth $B_{t,\text{single}}$ of $\widehat{\Delta L}$ is bounded as

$$B_{t,\text{single}} \leq |\dot{\mathbf{v}}|B_{L_\infty} \,. \tag{5.13}$$

Our analysis above considers the motion of only a single small occluding patch, and we conceptually estimate the bandwidth due to the motion of *all* the occluders by taking the maximum bandwidth across all the small occluding patches: during spherical radiance point integration (Section 4), we compute (and cache; Section 5.1) the mean angular velocity of occluding samples times the light bandwidth (estimated using our hierarchichal structure; Section 4) $\mu[|\dot{\mathbf{v}}|B_{L_\infty}]$ and its variance $\sigma[|\dot{\mathbf{v}}|B_{L_\infty}]$, in a manner similar to the accumulation of the visibility statistics in Equation 4.4. We then approximate the final temporal bandwidth, and thus the *maximum* bandwidth across all occluding patches, as the mean plus 3 times the

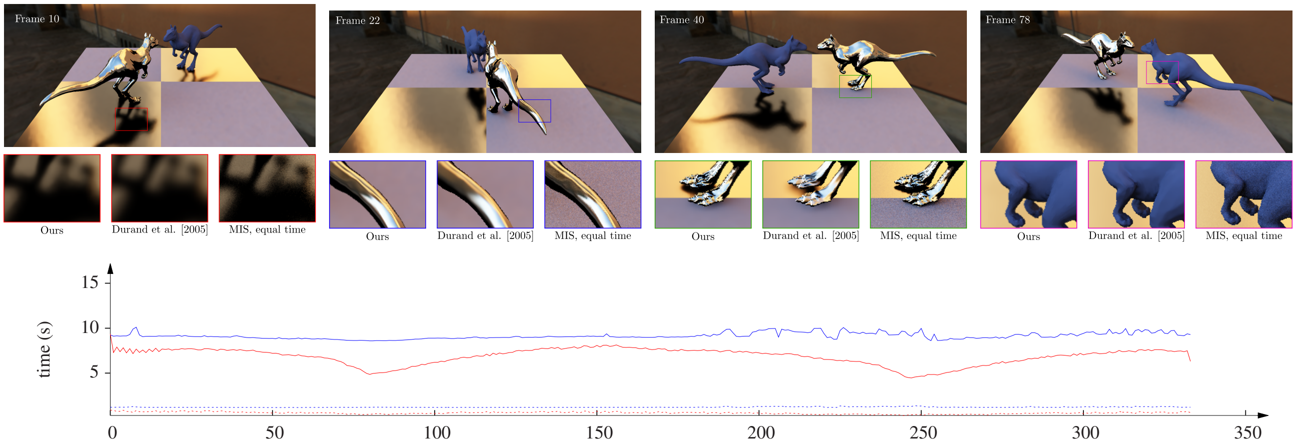


Fig. 15. Stills from the KILLEROO animation with one glossy and one diffuse Killeroo rotating above a diffuse and glossy checkerboard, lit by the PISA environment. Rendering times per frame for our method and Durand et al. [**10**] are in red and blue, respectively. Reconstruction times are provided using dashed lines of the same color. Insets compare feature reconstruction our approach, equal-time Durand et al. [**10**], and an equal-time MIS integrator.

standard deviation,

$$B_t = \mu[|\dot{\mathbf{v}}|B_{L_\infty}] + 3\sigma[|\dot{\mathbf{v}}|B_{L_\infty}]\,. \tag{5.14}$$

This conservative bandwidth estimate models the change in outgoing radiance that a radiance point will undergo in time due to occluder motion; while it uses a conservative bound on the amplitude of the integrated phase term, we do not need to explicitly account for the phase term during its calculation. This bandwidth can be interpreted as associating a *lifespan* to radiance points in the cache, and we simply extend our cache validity checks to accept reprojected cache samples only if they are "young" enough; if not, we discard them. We also discard samples which occlusion ratio is different from more than 0.125 with the occlusion ratio computed at the pixel the reproject to (occlusion ratio are ranging from 0 to 1). This further incorporate a change of viewpoint for the cache sample.

While our bandwidth estimate is far from accurate, it improves the quality of all-frequency shadowed regions and behaves consistently: without occluder motion, cached samples will have infinite lifespan and only be discarded if their spatial or angular bandwidth tests are not satisfied resulting in blurred shadows (see Figure 14).

6. Image Reconstruction

We generate the final image(s) with a reconstruction using all our (sparse) screen-space samples. We apply texture maps after shading reconstruction to avoid having to account for their potentially high-frequency content in our frequency analysis.

The reconstructed shading L_i at a pixel i is a weighted sum of the radiance points' values $L_{p'}$ (where p' denotes a projected pixel position) in its local neighborhood $\mathcal{N}_i$, where we only consider radiance points that reside on the same object as i:

$$L_i = T_i \sum_{p' \in \mathcal{N}_i} W_{p' \to i} \, L_{p'} \,, \tag{6.1}$$

where T_i is the texture value for pixel i. We use a standard cross-bilateral filter, as in previous shading reconstruction works [**10**], tailored to our predicted bandwidth estimates:

$$W_{p' \to i} = \exp\left[-\sum_{\mathbf{f}} \frac{1}{2\sigma_{\mathbf{f}}^2} ||\mathbf{f}_i - \mathbf{f}_{p'}||^2 \right] , \tag{6.2}$$

where $\mathbf{f}_{i|p'} = \{\mathbf{p}, \mathbf{n}, \mu_v, \sigma_v\}$ is our feature vector, so that the reconstruction will not combine values that differ much along any of the axes in this vector, and $\sigma_{\mathbf{f}}$ is a user defined standard deviation. In the case of the pixel feature vector, $\mathbf{f}_i$, we use the center of the pixel. Since our feature vector contains the frequency estimate, is capable of reconstructing both hard and soft transitions in image-space caused by shadows and reflections.

7. Results and Implementation

Our implementation is built directly atop Intel's Embree raytracing engine and operates completely on the CPU, although it is readily parallelizable on the GPU. Our implementation runs in parallel using OpenMP and a screen buffer separated into tiles of 16 by 16 pixels. We used the a screen-space curvature estimator depending on object-space normals and positions when computing an object's curvature at eye-ray hitpoints. Our MIS estimator of Equation 4.1 (Section 4) uses the power heuristic [**37**] and evenly distributes samples between the light and BRDF *pdf*s.

To construct the environment map's bandwidth hierarchy (Section 4), we use a 2D fast-Fourier transform [**7**] and, for footprints smaller than 16 × 16 pixels, we return the maximum possible frequency (one sample per texel). The bandwidth is computed by taking the 95$^{\text{th}}$

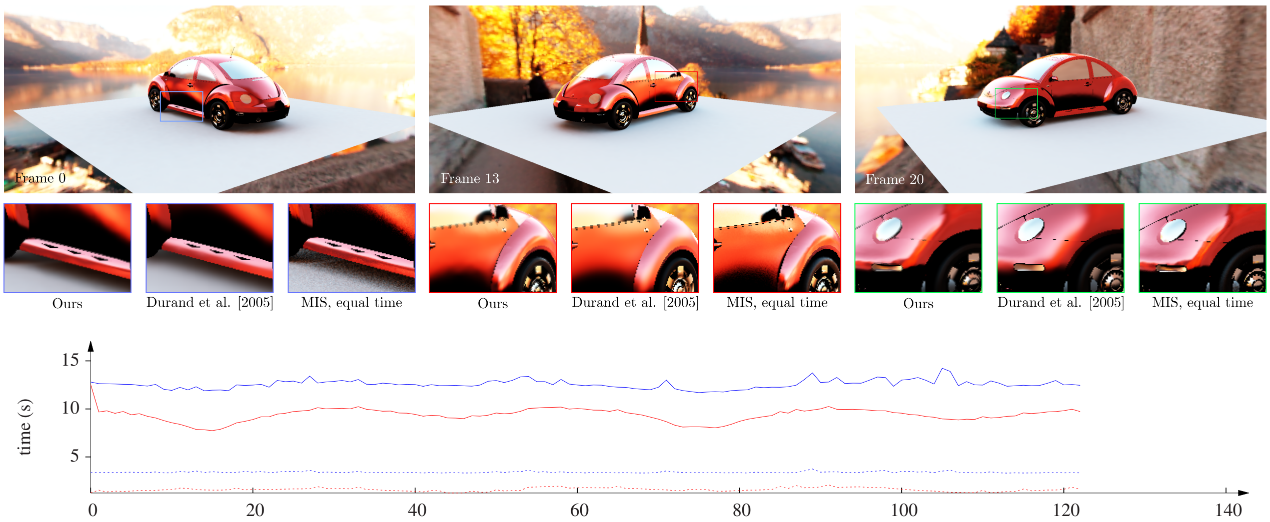
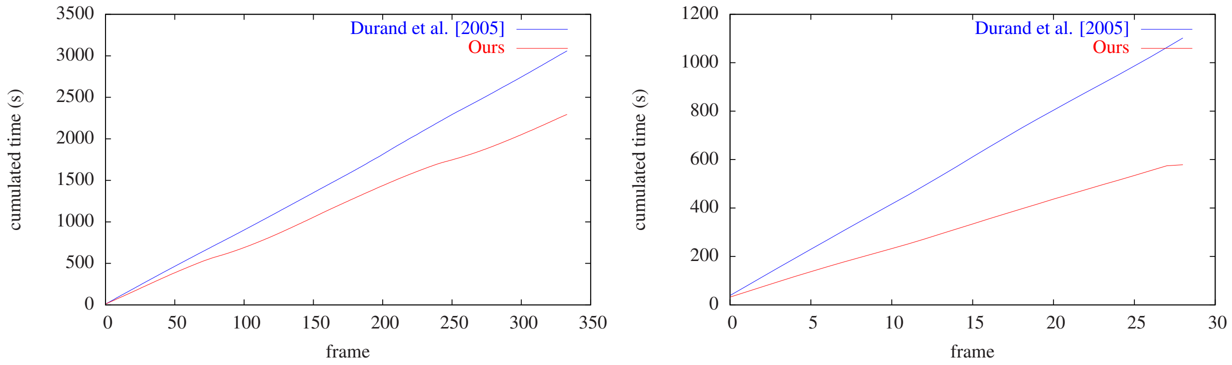


Fig. 16. Stills from the CAR animation with static geometry and a camera rotating around, lit by the HALLSTAT environment. Rendering times per frame for our method and Durand et al. [**10**] are in red and blue, respectively. Reconstruction times are provided using dashed lines. Insets compare feature reconstruction our approach, equal-time Durand et al. [**10**], and an equal-time MIS integrator.

percentile of the 2D spectrum. If the footprint spans several pixels at the target resolution, we conservatively take the maximum value over the resulting pixels.

To prepare the screen-space bandwidth of Equation 5.3 for sampling by rejection, we first modulate it according to the screen's aspect ratio, and then bound it between zero and one. So that the sampling density conforms to the aspect ratio, we multiply it by the maximum of the horizontal and vertical modifiers, so that $B_S = B_S\, max\left[\frac{f_x}{W}, \frac{f_y}{H}\right]$, in pixel^{-1}, where f_x and f_y are the horizontal and vertical fields of view, for a $W \times H$ sized image. We expose the maximum filter radius f_{max} for reconstruction as a user parameter, from which we can derive the minimum bandwidth we allow according to Shannon's theorem as $B_{min} = \frac{1}{2 \times f_{max}}$. B_{min} is fixed at 1 so that we never sample a pixel more than once. This higher bound could be extended to add support for sub-pixel supersampling.

If an object's associated cache needs to accommodate more samples than its current capacity during an animation, we resize it to be 1.5 time its current capacity so that we avoid the constant reallocations that could theoretically occur.



Killeroo cumulative time　　Helicopter cumulative time

Fig. 17. When comparing cumulative rendering times, our cache (in red) allows for smaller rendering time compared to the non-caching method [**10**] (in blue) where radiance points are recomputed every frame. In the case of Helicopter we achieve almost half the rendering time since we reuse most of the cache samples.

We add anti-aliasing as a post-process operation for the results of all the compared techniques and which processing time is not included in the resulting times since it isn't relevant to the measure. For that reason, we use sub-pixel information for it about the normals, depths and object IDs for a more precise detection of edges.

Results. We have tested our method on the following scenes: Helico (Figure 19) shows a rotating rotor blade in a static scene with simple Whitted-style indirect specular effects and texture mapping, Car (Figure 16) is an example of glossy, diffuse and mirror objects under camera rotation, Killeroo (Figure 15) combines diffuse and glossy BRDFs with meshes that exhibit high curvature, and so are particularly challenging for resolving reflections. All results are generated on an Intel i7 CPU 930 2.80 Ghz at a resolution of 1280 × 720, with the following parameter settings $f_{max} = 8$, $q_{\epsilon_x} = q_{\epsilon_\theta} = 1e^{-4}$ and $q_{\epsilon_t} = 1e^{-7}$.

Our method consistently reconstructs image sequences using only a fraction of the pixels on a screen, and our results are temporally coherent (see video): we typically reuse between 10 and 25% of the pixels to render any given frame of an animation. Our cache size ranges on tested scenes from 45k to 500k radiance points, but could potentially go further than that (1M) for long, complex animations over convoluted scenes; corresponding to a maximum size of 6, 68 and 136 Mb respectively.

We perform an equal-quality comparison of an animation sequence generated with our method to that of Durand et al.'s [**10**] prototype (Figure 17): we also exhibit sublinear scaling of the rendering time with respect to the number of rendered frames.

Finally, we perform an equal-time comparison of a still frame to the SURE-based optimization for Adaptive Sampling and Reconstruction from Li et al. [**21**] (Figure 18). The results were rendered with the parameter set put forth by the authors, that is $\sigma_{fk} = 0.4$, 0.125 and 0.3 for normal, texture color and depth respectively. We adapted the number of filter bank iterations for the final pass as $\sigma_s = \{0, 1, 2, 4, 8\}$ for performance purposes to reduce the execution time of their technique to something closer to ours.

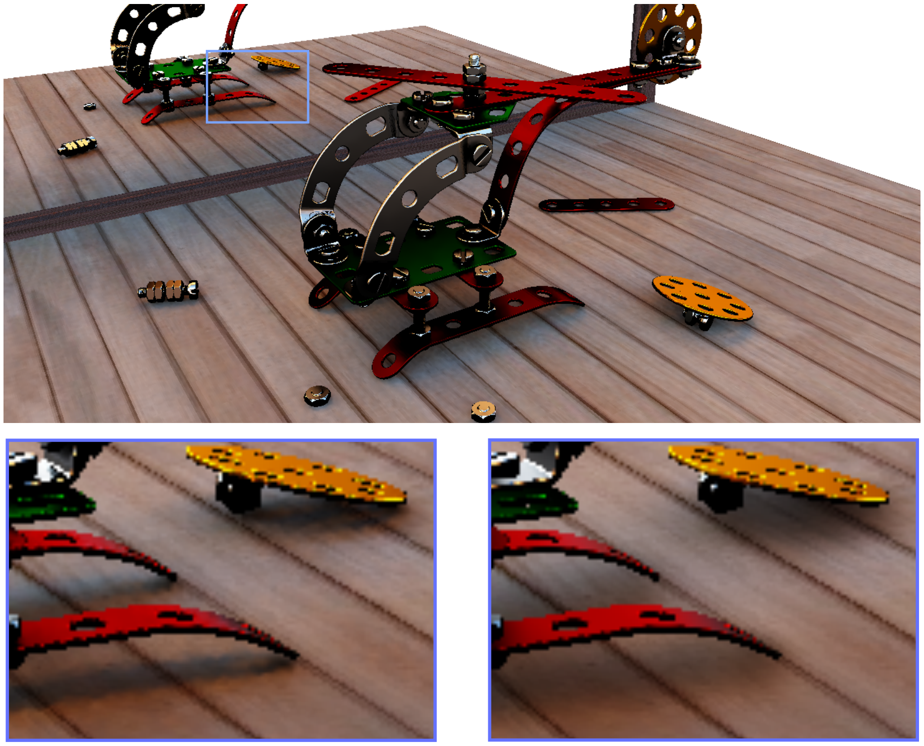
The SURE-based adaptive sampling method performs a lot better than raw MIS perceptually, especially in low-frequency regions where their algorithm makes good use of larger filters. While it performs quantitatively better than our method here, Li et al. technique appear noisier, especially when handling all-frequency shadows, as present in the HELICO scene.

8. Conclusion and Future Work

We presented an adaptive sampling, signal-tailored integration and reconstruction technique for all-frequency direct illumination, rendering images and animations using a fraction of the cost of standard techniques. We develop new frequency bandwidth estimates to appropriately sample occlusion, reflectance, and lighting variations. In the case of animation sequences, we further amortize rendering cost with a lightweight caching scheme that also exploits our frequency analysis.

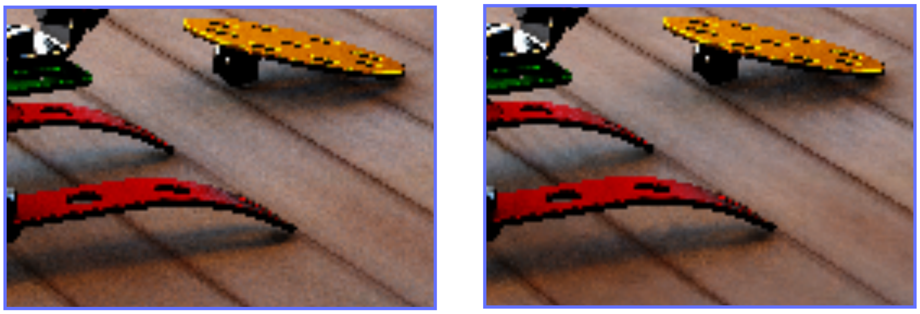
Discussion. Our temporal bandwidth derivation for moving occluders assumes that the projected size of a small occluding patch does not change over time; this only holds for purely rotational occluder motion (from the shading point's perspective), but still provides a good approximation in the case of small motions. Alleviating this constraint could further improve our visibility bandwidth, and we would like to investigate coupling the effects of temporal changes in the reflectance and lighting together with the occlusion. Similarly, we only consider linear object motion, as is common in many rendering approaches, so modeling shading variations due to *rotational* motion could be an interesting avenue of future work.

We do not model the temporally varying depth complexity of occluders, which could lead to shading variations when occluders subtend the same spherical region at certain moments in time during there motion; in the case where one moving occluder completely blocks another, this may lead temporal bandwidth underestimation in our current approach, however we have not noticed a case where this results in any artifacts during our investigation.



Ours
RMSE : 0.01001

Durand et al. [2005]
RMSE : 0.01967



MIS
RMSE : 0.01174

Li et al. [2012]
RMSE : 0.00913

Fig. 18. We present an equal-time still frame comparison between our technique and the work of Li et al. [**21**], Durand et al.[**10**] and MIS as reference.

Another more subtle issue arises when trying to simply multiply the reconstructed shading by a texture value after the fact (Equation 6.1): this only holds when the texture value at a pixel does not vary over time; this is clearly not the case for objects under relative motion w.r.t. the camera. The correct solution would be to project the texture, masked by the pixel's spatial footprint, across time and onto our other filters in order to model its maximum bandwidth using e.g. a mean-plus-3-standard-deviation approximation. We can then weight the spatially-varying texture by the shading variation over the same time period.

While our caching scheme and bandwidth estimates can be readily integrated into other frequency-based shading approaches to account for distribution effects such as defocus [**34, 5**] or motion blur [**13, 5**], our algorithm and analysis are currently restricted to single-bounce direct illumination (and Whitted-style recursive effects, e.g. mirror reflections). Extensions to global illumination are not trivial, but one direction would be to formulate a progressively-accumulated bandwidth estimate to deal with multiple bounces in e.g. a path-tracing estimator.

Lastly, while one of the benefits of our cache system is that it is very lightweight, we do not currently share or reuse integration samples across radiance points; previous approaches [**11, 12**] have treated similar problems using a heavyweight ray-space cache. We find our proposed solution to be a reasonable middle-ground solution, using higher-order statistics to *compare* and reason about nearby radiance points, but we never *share* this information *across* radiance points to influence the integration process. Doing so is left to future work and could further improve the efficiency of a technique like ours.

9. Appendix A: Image-space spatio-angular variance

We derive the 2D spatial and angular variances of the incident light field in object and image space to direct our sampling and reconstruction algorithms. We do this by analysing how the light field function changes along the light path from the light source to the eye, and work with 2×2 matrix operators to facilitate derivation. Our analysis is based on previous works [**10, 11, 5, 1**].

We do not consider the convolution by the surface texture's spectrum in our analysis and apply texturing by simply doing the per-pixel product of the final radiance estimation and the texture value at that (u,v) point.

9.1. Covariance Analysis

We first derive the light field variance in the unoccluded case, where the incident light field is not obstructed from its origin to the considered surface point $\mathbf{x}$, so that $\mathbf{T}_{V\to\mathbf{x}}$ has no effect on the purely angular variance at emission. We then study the variance in the occluded case, where part of the light field can be obstructed by blockers between the light and the surface.

Unoccluded Case. We only consider light from distant illumination modelled by environment maps, which has no spatial variation and only non-zero angular variance, as

$$\Sigma = \begin{bmatrix} 0 & 0 \\ 0 & \sigma^2_{L_\infty} \end{bmatrix}. \tag{9.1}$$

Light travels from the source to the surface point $\mathbf{x}$ in a straight line (since we do not tackle participating media). This results in an angular shear with a magnitude equal to the distance travelled d_1. Since we use infinitely distant light with only angular variation, this operation has no effect on the signal and isn't represented in Equation 5.2. The operator is defined as

$$\mathbf{T}_{d_1} = \begin{bmatrix} 1 & -d_1 \\ 0 & 1 \end{bmatrix}. \tag{9.2}$$

Reflection of the incident light field at the surface point $\mathbf{x}$ in the direction of the eye is composed of a serie of transformations [**10**]:

(1) The re-parametrization of the light field in the surface's local frame, where the light field is first scaled by $c_i = cos\,\theta_i$ to account for the foreshortening of the incident ray, and then sheared by the effect of the curvature k of the surface. The scale due to the incident angle is spatial and has no effect on our purely angular light signal here. In matrix form, Σ is modified by

$$\mathbf{C}_L = \begin{bmatrix} c_i & 0 \\ k & 1 \end{bmatrix}. \tag{9.3}$$

(2) The angular convolution of the re-parametrized light field by the surface's BRDF, which band-limits the signal by an amount inversely-proportional to the BRDF's shininess. We only consider time- and space-invariant isotropic BRDFs and our analysis is based on the Phong BRDF, for which we obtain the covariance $cov_{\theta,\theta}(\rho_s) = s/4\pi^2$

for a Phong exponent (or shininess) of s [**5**], which can be rewritten in matrix form, with $b = 1/cov_{\theta,\theta}(\rho_s)$, as

$$\mathbf{B}_{\rho_s} = \begin{bmatrix} 0 & 0 \\ 0 & b \end{bmatrix}. \tag{9.4}$$

(3) The re-parametrization in the outgoing direction to the eye, similar to the first step, is first a mirror reflection in the spatial domain, followed by the inverse curvature shear by $-k$ and the scale by $c_v = 1/cos\,\theta_v$, as

$$\mathbf{C}_v = \begin{bmatrix} c_v & 0 \\ c_v\,k & -1 \end{bmatrix}. \tag{9.5}$$

Finally, light travels from $\mathbf{x}$ to the eye by a distance d, and the signal is modified one last time by the operator

$$\mathbf{T}_{\mathbf{x}\to\mathbf{v}} = \begin{bmatrix} 1 & -d \\ 0 & 1 \end{bmatrix}. \tag{9.6}$$

We put everything together to get the spatial and angular variances in object and image space. Let Σ be the covariance of the incident light field, we get the object space covariance $\Sigma^o_{S,\theta}$ by the succession of operations involving the matrix operators defined above such as

$$\Sigma_c = \mathbf{C}_L^T\,\Sigma\,\mathbf{C}_L \quad \Sigma_\rho = \Sigma_c - \frac{\Sigma_c^T\,\mathbf{B}_{\rho_s}\,\Sigma_c}{1 + \mathrm{Tr}\left[\Sigma_c^T\,\mathbf{B}_{\rho_s}\right]} \quad \Sigma^o_{S,\theta} = \mathbf{C}_v^T\,\Sigma_\rho\,\mathbf{C}_v, \tag{9.7}$$

and the image space covariance $\Sigma_{S,\theta}$ by applying the travel operator

$$\Sigma_{S,\theta} = \mathbf{T}^T_{\mathbf{x}\to\mathbf{v}}\,\Sigma^o_{S,\theta}\,\mathbf{T}_{\mathbf{x}\to\mathbf{v}}. \tag{9.8}$$

Occluded Case. When accounting for occlusion, an additional transport operator $\mathbf{T}_{V\to\mathbf{x}}$ warps the spatio-angular occlusion bandwidth according to the *minimum occluder distance* t_{min} between the shade point and the light [**10, 11**]

$$\mathbf{T}_{V\to\mathbf{x}} = \begin{bmatrix} 1 & -t_{min} \\ 0 & 1 \end{bmatrix}. \tag{9.9}$$

Since we employ distant environmental illumination, this operation wouldn't affect the covariance as it only has non-zero angular variance $\sigma^2_{L_\infty}$ at emission. We add an undefined spatial frequency content a in the matrix at emission and apply the transport shear before the interaction with the surface, the BRDF and the re-projection to the eye, so that our covariance matrix at emission is defined as

$$\Sigma = \begin{bmatrix} a & 0 \\ 0 & \sigma^2_{L\infty} \end{bmatrix}. \tag{9.10}$$

The rest of the derivation of the object and image space covariances is done in the same manner as in the unoccluded case above, except that

$$\Sigma_c = \mathbf{C}_L^T \, \Sigma_t \, \mathbf{C}_L, \text{ with } \Sigma_t = \mathbf{T}_{V\to\mathbf{x}}^T \, \Sigma \, \mathbf{T}_{V\to\mathbf{x}}. \tag{9.11}$$

To get defined results, we compute the covariance in the limit where the spatial content a of our emitted light tends to infinity.

9.2. Scalar variance & Bandwidth

The matrix operations above are used to determine the object space spatial and angular bandwidths (B_x, B_θ) to control the validity of re-projected cache samples in Section 5.2; and the image space angular variance σ^2_θ in Equation 5.3 of Section 5.1.

In practice, the occluded derivation is always used to get σ^2_θ, B_x and B_θ. The only case where the unoccluded derivation is used is when determining filter sizes of the bilateral filtering pass used to smooth the visibility statistics $\{\mu_o, \Sigma_o\}$ in Section 5.1.

When computing the occluded variables, in the frequent case where no actual occlusion occur between the surface and the light, t_{min} is undefined and we should optimally switch to the unoccluded variables. However, we found that by using a large enough distance as t_{min} when there is no occlusion gives us smoother transitions between occluded and unoccluded areas and avoid the use of a hard switch. This maximum distance is scene dependent and is defined proportionally to the scene's bounds. By considering every direction as occluded, we first bias our sampling scheme and reconstruction filter sizes to distribute more samples and be smaller, respectively, than with the hard switch; and second make our cache re-projection policy stricter. We allow those biases since they can only improve quality, and because the maximum distance used is large enough as to make any of those effect imperceptible.

Below are the scalar variables used in practice, evaluated from the derivations above.

Unoccluded Case.

$$\begin{aligned}
(\sigma_S^o)^2 &= \Sigma^o_{S,\theta}[1,1] = (4\,c_v^2\,k^2\,\sigma^2_{L_\infty})\Big/(1 + b\,\sigma^2_{L_\infty}),\\
(\sigma_\theta^o)^2 &= \Sigma^o_{S,\theta}[2,2] = (\sigma^2_{L_\infty})\Big/(1 + b\,\sigma^2_{L_\infty}),\ \text{and}\\
\sigma_\theta^2 &= \Sigma_{S,\theta}[2,2] = (1 - 2\,c_v\,d\,k)^2\,\sigma^2_{L_\infty}\Big/(1 + b\,\sigma^2_{L_\infty}).
\end{aligned}$$

Occluded Case.

$$\begin{aligned}
(\sigma_S^o)^2 &= \lim_{a\to\infty} \Sigma^o_{S,\theta}[1,1] = \Big((c_i + 2\,t_{lim}\,k)^2 + b\,c_i^2\,\sigma^2_{L_\infty}\Big)\Big/ b\,t^2_{lim}\ ,\\
(\sigma_\theta^o)^2 &= \lim_{a\to\infty} \Sigma^o_{S,\theta}[2,2] = b^{-1},\ \text{and}\\
\sigma_\theta^2 &= \lim_{a\to\infty} \Sigma_{S,\theta}[2,2] = \frac{(t_{min} - c_v\,d\,(c_i + 2\,t_{min}\,k))^2 + b\,c_v^2\,c_i^2\,d^2\,\sigma^2_{L_\infty}}{b\,t^2_{min}}\ .
\end{aligned}$$

We derive bandwidths B_x and B_θ in the same manner (see Section 5.1) as

$$B_x = 3\sqrt{{\sigma_S^o}^2} \text{ and } B_\theta = 3\sqrt{{\sigma_\theta^o}^2}.$$

References

[1] Mahdi Mohammad Bagher, Cyril Soler, Kartic Subr, Laurent Belcour, and Nicolas Holzschuch. Interactive rendering of acquired materials on dynamic geometry using frequency analysis. *IEEE TVCG*, 19(5), 2013.

[2] Kavita Bala, Julie Dorsey, and Seth Teller. Radiance Interpolants for Accelerated Bounded-Error Ray Tracing. *ACM Trans. Graph.*, 18(3), 1999.

[3] Kavita Bala, Bruce Walter, and Donald Greenberg. Combining edges and points for interactive high-quality rendering. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2003)*, 22(3):631–640, 2003.

[4] Petr Beckmann and Andre Spizzichino. *Scattering of Electromagnetic Waves from Rough Surfaces.* 1987.

[5] Laurent Belcour, Cyril Soler, Kartic Subr, Nicolas Holzschuch, and Frédo Durand. 5D Covariance tracing for efficient defocus and motion blur. *ACM Transactions on Graphics*, 32(3):31:1–31:18, 2013.

[6] Benedikt Bitterli, Fabrice Rousselle, Bochang Moon, José A Iglesias-Guitián, David Adler, Kenny Mitchell, Wojciech Jarosz, and Jan Novák. Nonlinearly weighted first-order regression for denoising monte carlo renderings. In *Computer Graphics Forum*, volume 35, pages 107–117, 2016.

[7] James W. Cooley and John W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*, 19(90):297–297, May 1965.

[8] Carsten Dachsbacher, Marc Stamminger, George Drettakis, and Frédo Durand. Implicit visibility and antiradiance for interactive global illumination. *ACM Trans. Graph.*, 26(3), 2007.

[9] Holger Dammertz, Daniel Sewtz, Johannes Hanika, and Hendrik P. A. Lensch. Edge-avoiding À-Trous wavelet transform for fast global illumination filtering. In *Proceedings of High Performance Graphics*. Eurographics, June 2010.

[10] Frédo Durand, Nicolas Holzschuch, Cyril Soler, Eric Chan, and François X. Sillion. A frequency analysis of light transport. *ACM Transactions on Graphics*, 24(3):1115–1126, 2005.

[11] Kevin Egan, Frédo Durand, and Ravi Ramamoorthi. Practical filtering for efficient ray-traced directional occlusion. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2011)*, 30(6), December 2011.

[12] Kevin Egan, Florian Hecht, Frédo Durand, and Ravi Ramamoorthi. Frequency analysis and sheared filtering for shadow light fields of complex occluders. *ACM Transactions on Graphics*, 30(2):1–13, April 2011.

[13] Kevin Egan, Yu-Ting Tseng, Nicolas Holzschuch, Frédo Durand, and Ravi Ramamoorthi. Frequency analysis and sheared reconstruction for rendering motion blur. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)*, 28(3):93:1–93:13, 2009.

[14] Iliyan Georgiev, Krivanek Jaroslav, Davidovic Tomas, and Philipp Slusallek. Bidirectional Light Transport with Vertex Connection and Merging. In *Proceedings of SIGGRAPH Asia 2012*, 2012.

[15] Toshiya Hachisuka, Jacopo Pantaleoni, and Henrik Wann Jensen. A Path Space Extension for Robust Light Transport Simulation. Technical report, 2012.

[16] James T. Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques - SIGGRAPH '86*, volume 20, New York, New York, USA, August 1986. ACM Press.

[17] Jaroslav Křivánek and Mark Colbert. Real-time Shading with Filtered Importance Sampling. *Computer Graphics Forum*, 27(4):71, June 2007.

[18] Jaroslav Krivánek, Pascal Gautron, Sumanta Pattanaik, and Kadi Bouatouch. Radiance caching for efficient global illumination computation. *IEEE transactions on visualization and computer graphics*, 11(5), January 2005.

[19] Jaakko Lehtinen, Timo Aila, Jiawen Chen, Samuli Laine, and Frédo Durand. Temporal light field reconstruction for rendering distribution effects. *ACM Transactions on Graphics*, 30(4):1, July 2011.

[20] Jaakko Lehtinen, Timo Aila, Samuli Laine, and Frédo Durand. Reconstructing the indirect light field for global illumination. *ACM Transactions on Graphics*, 31(4):1–10, July 2012.

[21] Tzu-Mao Li, Yu-Ting Wu, and Yung-Yu Chuang. SURE-based optimization for adaptive sampling and reconstruction. *ACM Transactions on Graphics*, 31(6):1, November 2012.

[22] Soham Mehta, Jiaxian Yao, Ravi Ramamoorthi, and Frédo Durand. Factored Axis-Aligned Filtering for Rendering Multiple Distribution Effects. *Proceedings of ACM SIGGRAPH*, 33(4), 2014.

[23] Soham Uday Mehta, Brandon Wang, and Ravi Ramamoorthi. Axis-aligned filtering for interactive sampled soft shadows. *Proceedings of ACM SIGGRAPH Asia*, 31(6), November 2012.

[24] Soham Uday Mehta, Brandon Wang, Ravi Ramamoorthi, and Frédo Durand. Axis-aligned filtering for interactive physically-based diffuse indirect lighting. *Proceedings of ACM SIGGRAPH*, 31(4), July 2013.

[25] Mark Meyer and John Anderson. Statistical acceleration for animated global illumination. *ACM Transactions on Graphics*, 25(3), July 2006.

[26] Ravi Ramamoorthi, John Anderson, Mark Meyer, and Derek Nowrouzezahrai. A theory of monte carlo visibility sampling. *ACM Transactions on Graphics*, 2012.

[27] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *SIGGRAPH*, NY, 2001. ACM.

[28] Ravi Ramamoorthi and Pat Hanrahan. Frequency space environment map rendering. *ACM Transactions on Graphics*, 21(3), July 2002.

[29] Ravi Ramamoorthi, Dhruv Mahajan, and Peter Belhumeur. A first-order analysis of lighting, shading, and shadows. *ACM Transactions on Graphics*, 26(1), 2007.

[30] Fabrice Rousselle, Claude Knaus, and Matthias Zwicker. Adaptive Sampling and Reconstruction using Greedy Error Minimization. *ACM Transactions on Graphics*, 30(6):1—-12, 2011.

[31] Jorge Schwarzhaupt, Henrik Wann Jensen, and Wojciech Jarosz. Practical Hessian-based error control for irradiance caching. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2012)*, 31(6):1–10, 2012.

[32] Pradeep Sen and Soheil Darabi. On Filtering the Noise from the Random Parameters in Monte Carlo Rendering. *ACM Transactions on Graphics (TOG)*, 31(3):1–15, May 2011.

[33] Peter-Pike Sloan, Naga K. Govindaraju, Derek Nowrouzezahrai, and John Snyder. Image-based proxy accumulation for real-time soft global illumination. *Pacific Conference on Computer Graphics and Applications*, January 2007.

[34] Cyril Soler, Kartic Subr, Frédo Durand, Nicolas Holzschuch, and François X. Sillion. Fourier depth of field. *ACM Transactions on Graphics*, 28(2):18:1–18:12, 2009.

[35] Kenneth Torrance and Emett Sparrow. Theory for off-specular reflection from roughened surfaces. *Journal of the Optical Society of America*, 1967.

[36] Eric Veach. *Robust Monte Carlo Methods for Light Transport Simulation.* PhD thesis, Stanford University, 1997.

[37] Eric Veach and Leonidas J. Guibas. Optimally combining sampling techniques for Monte Carlo rendering. In *SIGGRAPH 1995.* ACM Press, September 1995.

[38] Edgar Velázquez-Armendáriz, Eugene Lee, Kavita Bala, and Bruce Walter. Implementing the render cache and the edge-and-point image on graphics hardware. In *Proceedings of Graphics Interface*, June 2006.

[39] Bruce Walter, George Drettakis, and Steven Parker. Interactive rendering using the render cache. pages 19–30, June 1999.

[40] Bruce Walter, Stephen Marschner, Hongsong Li, and Kenneth Torrance. Microfacet Models for Refraction through Rough Surfaces. *Rendering Techniques (Proceedings of Eurographics Symposium on Rendering 2007)*, pages 195–206, 2007.

[41] Gregory J. Ward and Paul S. Heckbert. Irradiance gradients. In *Eurographics Workshop on Rendering*, pages 17–20, 1992.

Deuxième article.

Impulse Responses for Precomputing Light from Volumetric Media

par

Renaud Adrien Dubouchet[1], Peter-Pike Sloan[2],
Wojciech Jarosz[3], and Derek Nowrouzezahrai[4]

([1]) Université de Montréal
([2]) Activision Publishing Inc.
([3]) Dartmouth College
([4]) McGill University

Cet article a été soumis à la revue Eurographics Symposium on Rendering 2019, en avril 2019. Il a été reformaté afin de suivre convenablement le format de cette thèse.

Renaud Adrien Dubouchet est l'auteur principal de cet article. Il a été principalement impliqué dans les dérivations mathématiques du nouveau modèle analytique basé sur les réponses impulsives pour représenter le transport de lumière à travers les médias volumétriques participatifs, ainsi que dans le design et l'implémentation de la technique dans un outil de précalcul professionnel chez Activision Blizzard, Inc. Il a été l'auteur principal de la rédaction de cet article et son présentateur à la conférence associée.

Ce second article présente une méthode novatrice permettant d'ajouter à faible coût les chemins de lumière à travers les médias volumétriques participatifs sur des surfaces. Ces transports ne sont habituellement pas pris en compte pour les applications intéractives du au coût élevé nécessaire pour simuler les multiples évenements de dispersions que la lumière effectue dans les médias participatifs.

Notre méthode se base sur une nouvelle vision du transport à travers les médias, jamais encore étudiée, factorisant ces multiples intéractions sous forme d'harmoniques zonales; délivrant ainsi une représentation compacte et simple d'utilisation. La technique est étendue afin de pouvoir gérer le cas des médias hétérogènes–ajoutant une dimension spatiale à la densité volumétrique– ainsi que les fonctions de phase anisotropiques–guidant angulairement la distribution de lumière à chaque évènement de dispersion.

RÉSUMÉ. Les méthodes modernes de rendu intéractif s'appuient souvent sur le précalcul du rendu d'illumination surfacique et volumétrique. Cependant, le transport de lumière *à travers* des médias participatifs *sur* des surfaces est typiquement ignoré pour ces précalculs. Nous proposons une méthode efficace et compacte pour simuler le transport de lumière de volume à surface pour le précalcul d'illumination. Nous prenons avantage d'un nouveau modèle se basant sur la réponse impulsive de la lumière dispersée et attenuée par les médias volumétriques afin de simuler son transport sur des surfaces par le biais de simples tables de correspondance. Nos tables modèlent cette réponse impulsive en fonction de la distance et de l'angle entre lumières et surfaces, que nous remappons ensuite à des médias possèdant différents paramètres de dispersion (possiblement hétérogènes) et/ou fonctions de phase, et gèrant par composition la possibilité d'avoir multiples évènements de dispersion dans le volume. Nous appliquons notre méthode pour précalculer le transport de lumière de volume à surface dans des scènes complexes et générons des résultats indiscernables de simulations vérité terrain. Nos tables permettent ces précalculs de transport de lumière à des ordres de grandeur plus rapide que le pourraient des solutions optimisées de tracé de chemins.
Mots clés : Harmoniques Sphériques, Illumination précalculée, Rendu de Volumes

ABSTRACT. Modern interactive rendering can rely heavily on precomputed static lighting on surfaces and in volumes. Scattering from volumetric media can be similarly treated using precomputation, but transport *from* volumes *onto* surfaces is typically ignored here. We propose a compact, efficient method to simulate volume-to-surface transport during lighting precomputation . We leverage a novel model of the spherical impulse response of light scattered (and attenuated) in volumetric media to simulate light transport from volumes onto surfaces with simple precomputed lookup tables. These tables model the impulse response as a function of distance and angle to the light and surfaces. We then remap the impulse responses to media with arbitrary, potentially heterogeneous scattering parameters and various phase functions. Moreover, we can compose our impulse response model to treat multiple scattering events in the volume (arriving at surfaces). We apply our method to precomputed volume-to-surface light transport in complex scenes, generating results indistinguishable from ground truth simulations. Our tables allow us to precompute volume-to-surface transport orders of magnitude faster than even an optimized path tracing-based solution would.
Keywords: Spherical Harmonics, Precomputed Lighting, Volume Rendering

Fig. 19. In-game screenshot (a) with complex surface and volume shading. Standard lightmaps (b) ignore volume-to-surface transport, which our method compensates for (c) with media-attenuated surface global illumination (d) and volumetric single- and multiple-scattering (e).

1. Introduction

Volumetric participating media can significantly impact the realism of virtual scenes, due to both the subtle interplay between surface- and volume-transport effects and the presence of entities like smoke and clouds. Modern video games increasingly seek to include such effects to improve realism and artistic flexibility. While many works propose efficient simulation methods for volumetric single- and multiple-scattering (e.g., [**49, 12**]), no interactive methods accurately model the effects of scattering *from* a volume *onto* other surfaces/locations in a scene, let alone several such bounces. *Volume-to-surface* transport has been studied in offline image synthesis but these methods are too costly for interactive content pipelines.

Some interactive applications rely heavily on precomputed lighting [**33, 6, 16, 38**] where the tradeoffs lie between what to precompute and what to evaluate at run-time. Indirect light and lighting from complex area/sky sources is more often precomputed, whereas run-time light source influence radii are often restricted to reduce shading; here, their artificially clamped shading may also be precomputed [**4**], leading to shading that can be interpolated between run-time evaluated and precomputed sources. Precomputed lighting data is cached on surfaces and in free-space: lightmaps, per-vertex data and light probes. Precomputed light probes are used to relight dynamic objects with the effects of the (static) lighting in the scene. Relying on even the most efficient offline methods in order to include the effects of volumetric scattering in these precomputed datasets would require hours to days to converge for even moderately-sized in-game assets.

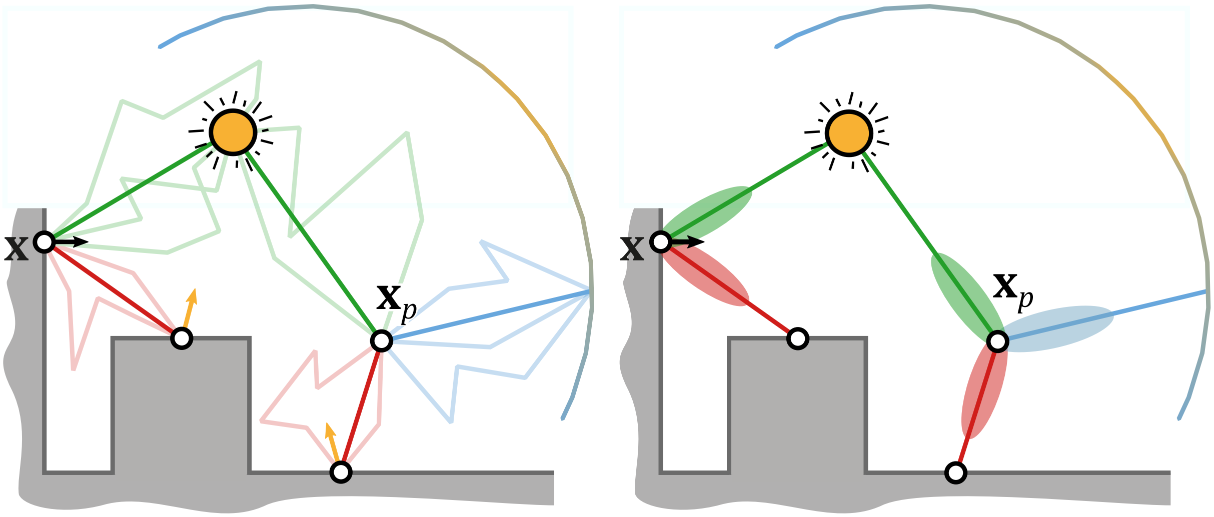


Fig. 20. Consider surface $\mathbf{x}$ and probe $\mathbf{x}_p$ locations lit by point (green), environment (blue) and indirect (red) light. Offline rendering traces light paths to solve the radiative transport equation (left; transparent lines). We encode the impulse response of volumetric transport (right), to model media scattering in precomputed lighting.

As such, even in scenes with participating media, games usually only precompute surface-to-surface transport. We propose a very efficient method to include the neglected volume-to-surface interactions during precomputation, resulting in more accurate lighting and only with modest performance overhead. Instead of relying on costly volumetric light transport simulation, we model the *spherical impulse response* of volumetric light in a form that allows us to quickly precompute volume-to-surface transport. Specifically, our contributions are:

- a formulation and analysis of the impulse response of volumetric in-scattered light for point, directional and differential emitters,
- a modular extension from single- to multiple-scattering regimes,
- a treatment of *heterogeneous*, *anisotropic* volume scattering, and
- a compact, pretabulated *zonal harmonic* parameterization of the impulse response applied to a constant-time algorithm for adding accurate volume-to-surface transport to precomputed light maps.

2. Related Work

Interactive rendering and games rely on faithful simulations of physically based light transport effects. While many effects remain outside the performance constraints imposed in these applications, precomputation-based solutions provide an avenue to include complex lighting with low runtime cost. Building on light mapping [**1**] where precomputed lighting is cached at texels, separate from higher frequency albedo, recent methods cache other forms of light transport data: e.g., by representing variations due to normal mapped surfaces [**33**] in a spherical/hemispherical basis, non-Lambertian view-dependent reflection can also be cached [**6, 36, 15, 38**]. This added angular resolution can, in turn, allow baked data to be represented at coarser spatial resolutions, such as per-vertex [**28**] or per-probe [**16**], improving storage and performance. For non-static geometry, lighting is also precomputed in volumetric data structures, variants of irradiance volumes [**10**]. Precomputation time can take between minutes and hours, and even with new GPU advances it is costly [**13**].

Precomputed Radiance Transfer (PRT) cache transport operators (e.g., per vertex) to model how incoming light (typically represented in a basis space) is locally transformed by occlusion and scattering. The basis representation allows for precomputed transport to be adapted, at runtime, to dynamic lighting and/or viewing scenarios [**45, 44, 30, 31**]. Several approaches explore various discretizations of the caching domain, including methods that optimize the placement and interpolation of per-vertex lighting data [**28**] and lighting "probe" placed in the open space of a scene in order to relight dynamic objects [**10, 42**]. These precomputation-based methods allow for costly effects, like interreflections or ambient occlusion, to be incorporated at a fraction of the cost required for their dynamic evaluation.

Many basis choices have been used for PRT. Spherical and Zonal Harmonics (SH, ZH) benefit from closed-form rotation and convolution operations [**40, 45, 46, 43**]. They are unable, however, to compactly represent higher frequency angular variation. On the other hand, wavelet bases are capable of representing angular variation across frequencies, but at the cost of more complicated runtime operations [**11**]. Volumetric transport tends towards angularly smooth radiance distributions, and so our work relies on compact ZH formulations to encode the impulse response of emitted light scattered within a volumetric media.

In volumetric shading, the underlying transport theory and effective Monte Carlo-based solutions are well understood [**47, 29, 7, 37**], and a tremendous amount of work has found

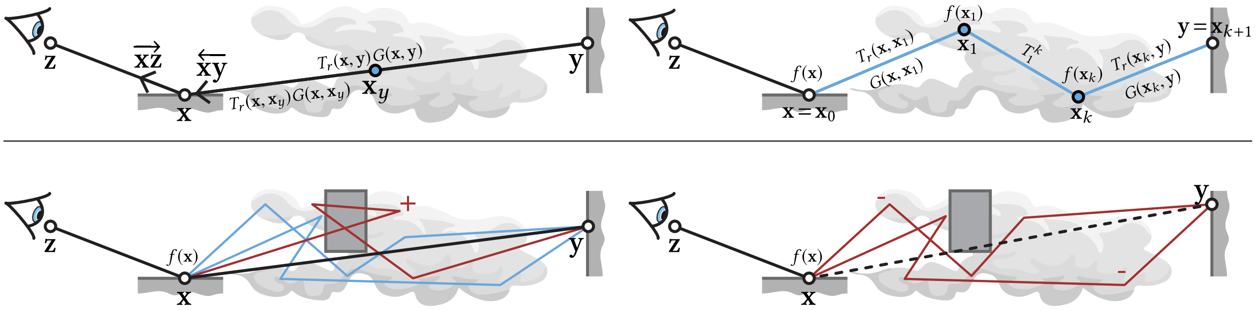


Fig. 21. Top left: 3-point geometry for radiance reflecting at $\mathbf{x}$ towards $\mathbf{z}$ as the sum of radiance arriving from surface point $\mathbf{y}$ and all media points $\mathbf{x}_y$. Top right: expanding the recursion results in path throughput of a media subpath between $\mathbf{x}$ and $\mathbf{y}$. Bottom left: we decompose the incident radiance at $\mathbf{x}$ as the sum of the attenuated surface radiance from $\mathbf{y}$ (**black**) and radiance arriving at $\mathbf{x}$ indirectly from $\mathbf{y}$ with any number of media bounces in between (**blue**). This can include false contributions from light paths which should be blocked (red +). Bottom right: We ignore contributions from surfaces that are directly occluded, even though radiance could still arrive at $\mathbf{x}$ indirectly from $\mathbf{y}$ (red −).

success in applications to offline visual effects [**48, 9**]. Path tracing has dominated film rendering [**39, 8**] due to its ability to scale and its speed to initial (albeit noisy) image result. Two dominant forms of caching have furthered the adoption of these methods: radiance caching [**32, 41, 23, 19**] and photon point and beam mapping [**3, 21, 20, 24, 22**]. In contrast, the performance constraints of interactive graphics necessitate the use of simplified volume shading methods.

Early work on subsurface scattering [**25**] modeled variation of scattering parameters by assuming local homogeneity. Our method approximate heterogeneous media similarly, relying on scattering parameters along a ray to make a *directional*-homogeneity simplification. Tabulated BSSRDFs, and their applications to discrete random media [**35**] and shell transport functions [**34**] are also tangentially related to our impulse response models for volume-to-point transport.

3. Background and Theoretical Model

We review volumetric light transport (Section 3.1) and present a factorization we will rely on (Section 3.2), before providing important definitions and properties of the SH basis

(Section 3.3). We focus on efficient simulation of costly volume-to-point transport effects during surface lightmap and lightprobe precomputation.

3.1. Light Transport in Volumetric Media

Light transport in volumetric participating media is governed by the *radiative transfer equation* (RTE) [**5**], a differential equation that describes the change in radiance L along a ray due to scattering and absorption. Integrating this equation and treating the surface *rendering equation* [**26, 14**] as a boundary condition yields the *volume rendering equation* (VRE) for the radiance $L(\mathbf{x}, \omega)$ arriving at point $\mathbf{x}$ from direction ω:

$$L(\mathbf{x}, \omega) = \underbrace{\int_0^s \sigma_t(\mathbf{x}_t)\, T_r(\mathbf{x}, \mathbf{x}_t)\, L_o(\mathbf{x}_t,\omega)\, \mathrm{d}t}_{L_m(\mathbf{x},\omega)} + \underbrace{T_r(\mathbf{x}, \mathbf{x}_s)\, L_o(\mathbf{x}_s,\omega)}_{L_s(\mathbf{x},\omega)}, \tag{3.1}$$

where L_m is the radiance arriving from all points $\mathbf{x}_t = \mathbf{x} - t\omega$ in the medium, L_s is the radiance arriving from a surface at $\mathbf{x}_s = \mathbf{x} - s\omega$, and σ_t is the media's extinction coefficient. Transmittance T_r models the attenuation due to extinction between two points:

$$T_r(\mathbf{x}, \mathbf{x}_t) = \mathrm{e}^{-\int_0^t \sigma_t(\mathbf{x}+t'\omega)\, \mathrm{d}t'} = \mathrm{e}^{-\sigma_t t}. \tag{3.2}$$

The outgoing radiance L_o is the sum of emitted radiance L_e and the angular integral of the incident radiance:

$$L_o(\mathbf{x},\omega) = L_e(\mathbf{x}, \omega) + \int_{\mathcal{S}^2} f(\mathbf{x}, \omega, \omega_i)\, L(\mathbf{x},\omega_i)\, \mathrm{d}_{\perp}\omega_i, \tag{3.3}$$

where ω_i and ω are incident and outgoing spherical directions about $\mathbf{x}$, $\mathrm{d}_{\perp}\omega_i$ is the differential projected solid angle, and f is either the volumetric phase function $f_p(\mathbf{x}, \omega, \omega_i)$ or surface BSDF $f_r(\mathbf{x}, \omega, \omega_i)$, depending on whether $\mathbf{x}$ is in the medium $\mathcal{V}$ or on a surface $\mathcal{A}$:

$$f(\mathbf{x}, \omega, \omega_i) = \begin{cases} \alpha(\mathbf{x}) f_p(\mathbf{x}, \omega, \omega_i) & \text{if } \mathbf{x} \in \mathcal{V}, \\ f_r(\mathbf{x}, \omega, \omega_i) & \text{if } \mathbf{x} \in \mathcal{A}, \end{cases} \tag{3.4}$$

where $\alpha = {}^{\sigma_s}/_{\sigma_t}$ is the albedo and σ_s the scattering coefficient. We precompute and tabulate response parameterized by the scattering coefficient σ_s; we devise an approach to optionally incorporate absorption coefficient during precomputation (Section 4.3).

3.2. Radiance Decomposition

We can reparameterize Eq. (3.3) in a three-point configuration as

$$L_o(\overrightarrow{\mathbf{xz}}) = L_e(\overrightarrow{\mathbf{xz}}) + \int_{\mathcal{A}} f(\mathbf{x})\, L(\overleftarrow{\mathbf{xy}})\, \mathrm{d}A(\mathbf{y}) \tag{3.5}$$

where we let $\overrightarrow{\mathbf{xz}}$ (and $\overleftarrow{\mathbf{xy}}$) denote the unit direction from $\mathbf{x}$ to $\mathbf{z}$ (and from $\mathbf{z}$ to $\mathbf{x}$), $L(\overleftarrow{\mathbf{xy}}) = L_s(\overleftarrow{\mathbf{xy}}) + L_m(\overleftarrow{\mathbf{xy}})$ is the radiance arriving at $\mathbf{x}$ from surface points $\mathbf{y} \in \mathcal{A}$, with

$$L_s(\overleftarrow{\mathbf{xy}}) = T_r(\mathbf{x},\mathbf{y})\, G(\mathbf{x},\mathbf{y})\, L_o(\overrightarrow{\mathbf{yx}}), \quad \text{and} \tag{3.6}$$

$$L_m(\overleftarrow{\mathbf{xy}}) = \int_0^y T_r(\mathbf{x},\mathbf{x}_y)\, G(\mathbf{x},\mathbf{x}_y)\, L_o(\overrightarrow{\mathbf{x}_y\mathbf{x}})\, \mathrm{d}y, \tag{3.7}$$

where $\mathbf{x}_y = \mathbf{x} + y \cdot \overrightarrow{\mathbf{xy}} \in \mathcal{V}$ are points in the volume between $\mathbf{x}$ and $\mathbf{y}$, and G and f are generalized geometry and scattering terms:

$$G(\mathbf{x},\mathbf{y}) = \frac{D_{\mathbf{x}}(\mathbf{y})\, V(\mathbf{x},\mathbf{y})\, D_{\mathbf{y}}(\mathbf{x})}{\|\mathbf{x}-\mathbf{y}\|^2},\ D_{\mathbf{x}}(\mathbf{y}) = \begin{cases} |\vec{n}_{\mathbf{x}} \cdot \overrightarrow{\mathbf{xy}}| & \text{if } \mathbf{x}{\in}\mathcal{A}, \\ 1 & \text{if } \mathbf{x}{\in}\mathcal{V}, \end{cases} \tag{3.8}$$

$$f(\mathbf{x}_i) = \begin{cases} \sigma_t(\mathbf{x}_i)\, \alpha(\mathbf{x}_i)\, f_p(\mathbf{x}_{i+1}, \mathbf{x}_i, \mathbf{x}_{i-1}) & \text{if } \mathbf{x} \in \mathcal{V}, \\ f_r(\mathbf{x}_{i+1}, \mathbf{x}_i, \mathbf{x}_{i-1}) & \text{if } \mathbf{x} \in \mathcal{A}. \end{cases} \tag{3.9}$$

Here, V is binary visibility and $\vec{n}_{\mathbf{x}}$ the normal at $\mathbf{x}$ (Fig. 21, top left).

Assuming non-emissive media and expanding the recursion at media scattering events in Eq. (3.7), we can rewrite $L(\overleftarrow{\mathbf{xy}})$ as a sum of the radiance arriving at $\mathbf{x}$ from points $\mathbf{y}$ with ≥ 0 medium scattering events in between:

$$L(\overleftarrow{\mathbf{xy}}) = \sum_{k=0}^{\infty} \underbrace{\int \ldots \int T_0^k\, L_o(\overrightarrow{\mathbf{y}\mathbf{x}_k})\, \mathrm{d}V(\mathbf{x}_1) \ldots \mathrm{d}V(\mathbf{x}_k)}_{L^k(\mathbf{x},\mathbf{y})} \tag{3.10}$$

where $L^k(\mathbf{x},\mathbf{y})$ is incident radiance at $\mathbf{x}$ that has scattered **exactly** k **times** in the medium after leaving $\mathbf{y}$, and T_0^k is the throughput for subpath $\mathbf{x}_0 \equiv \mathbf{x}, \ldots, \mathbf{x}_{k+1} \equiv \mathbf{y}$ (see Fig. 21; top right, bottom left):

$$T_0^k = \left[\prod_{i=1}^{k} f(\mathbf{x}_i)\right] \left[\prod_{i=0}^{k} T_r(\mathbf{x}_i,\mathbf{x}_{i+1})\, G(\mathbf{x}_i,\mathbf{x}_{i+1})\right]. \tag{3.11}$$

Note, for $k = 0$, Eq. (3.10) reduces to the attenuated surface radiance in Eq. (3.6), whereas $k > 0$ accounts for recursively expanded media radiance in Eq. (3.7). Moreover, while $L^k(\mathbf{x},\mathbf{y})$ represents radiance arriving (potentially indirectly) from $\mathbf{y}$, the incident direction at $\mathbf{x}$ is defined by $\overleftarrow{\mathbf{x}\mathbf{x}_1}$, which can be any spherical direction about $\mathbf{x}$ when $k \neq 0$. This means that when Eq. (3.10) is inserted into Eq. (3.5), the scattering function should be evaluated as $f(\mathbf{x}_1, \mathbf{x}, \mathbf{z})$. When $k = 0$, $\mathbf{x}_1 = \mathbf{x}_{k+1} = \mathbf{y}$, so the scattering is evaluated as $f(\mathbf{y}, \mathbf{x}, \mathbf{z})$.

3.3. Spherical Harmonics – Definitions and Properties

Preliminaries. Consider a scalar-valued function $f(\omega)$ over the unit sphere $\mathcal{S}^2$, with spherical directions $\omega = (x,y,z) = (\theta,\phi) \in S^2$. We obtain a vector $\boldsymbol{f}$ of SH coefficients by projecting $f(\omega)$ onto the real SH basis as $\boldsymbol{f} = \int_{S^2} f(\omega)\, \boldsymbol{y}(\omega) \mathrm{d}\omega$, where $\boldsymbol{f} = \{f_0^0, f_1^{-1}, f_1^0, f_1^1, \ldots\}$ is a vector of scalar projection coefficients f_l^m and $\boldsymbol{y}(\omega) = \{y_0^0, y_1^{-1}, y_1^0, y_1^1, \ldots\}$ a vector of the SH basis functions:

$$y_l^m(\theta,\phi) = \begin{cases} \sqrt{2}\ K_l^m\ \sin(|m|\phi)\ P_l^{|m|}(\cos\theta),\ m \leq 0 \\ \sqrt{2}\ K_l^m\ \cos(m\phi)\ P_l^m(\cos\theta)\quad,\ m > 0 \end{cases} . \tag{3.12}$$

Where P_l^m are associated Legendre polynomials and K_l^m are normalization factors. An order-N expansion of f onto the SH basis includes all functions for bands $0 \leq l \leq N-1$. Each band comprises $2l+1$ basis functions, indexed by m. For a fixed band l, each of the basis functions is a degree l polynomials in the Cartesian coordinates (x,y,z) of the (unit) direction ω, and we often rely on a single indexing scheme, with $i = l(l+1)+m$, for brevity and convenience.

Zonal Harmonics. Zonal $m = 0$ subset of SH functions, $y_l^0(\omega) = y_l^0(\theta)$, are circularly symmetric about $\cos\theta = z$ and referred to as *zonal harmonics* (ZH). Sloan et al. [**46**] introduced a fast rotation convolutional formulation for ZHs in order to align any weighted combination of the y_l^0 functions about an *arbitrary* axis $\bar{\omega} \neq x$. Doing so yield a function that can no longer be reconstructed as using solely a weighted combination of ZH functions, however the SH coefficients h_l^m of this arbitrarily-aligned (circularly symmetric) function can be directly obtained from the *ZH projection coefficients* f_l of the function in its original orientation (i.e.,about z), as

$$h_l^m = n_l^* \, f_l \, y_l^m(\bar{\omega}) = f_l^* \, y_l^m(\bar{\omega}) \ , \tag{3.13}$$

where $n_l^* = \sqrt{4\pi/(2l+1)}$ are convolutional normalization factors arising from the fact that Eq. (3.13) is the Funke-Hecke theorem applied to the original zonal function and a delta at $\bar{\omega}$.

Double-product Integration. Given two spherical functions $a(\omega)$ and $b(\omega)$ with (effective) bandlimits N_a and N_b, the integral of their product is $\int_{\mathcal{S}^2} a(\omega) b(\omega) d\omega = \sum_{i=0}^{n^2-1} a_i \, b_i$, where

$n = \min(N_a, N_b)$ and we arrive at the RHS by substituting the SH expansions of a and b into the LHS and then applying the orthonormality property of SH: $\int_{\mathcal{S}^2} y_i(\omega) y_k(\omega) \mathrm{d}\omega = \delta_{i,k}$, where $\delta_{i,k}$ is the Kroenecker delta.

We derive a double-product formulation for the special-case where a and b are (arbitrarily rotated) circularly symmetric zonal functions (see Appendix 8), and we leverage this more efficiently integration formulation in our shading algorithm (Section 4).

4. Impulse Response Models & Extensions

We present our theoretical contributions below, leaving discussions of practical and technical considerations for Section 5. Specifically,

- we factorize homogeneous inscattered radiance (Section 4.1) to motivate a novel *spherical impulse response* formulation for volume-to-point transport with three emission schemes (point-, surface- and indirectly-reflected cosine-profiles; Section 4.2),
- we extend these canonical impulse responses to support media with heterogeneous scattering coefficients (Section 4.4) and anisotropic phase functions (Section 4.5), and
- we derive an efficient double-product integration scheme (Appendix 8) to decouple incident and outgoing radiance, leading to important run-time flexibility (Section 5).

We aim to accelerate the precomputation of volume-to-point transport in the digital content creation pipeline for interactive graphics applications. We will leverage a novel factorized formulation that admits an efficient basis-space integration scheme. Our scheme will model singly- and multiply-scattered volumetric transport effects, as well as the effects of extinction through the media (Section 4.1). We avoid costly numerical path-based Monte Carlo simulations typical to existing pipelines used, i.e., in feature film production.

We precompute small impulse response tables once in a scene-agnostic manner (Sections 4.2 to 4.5), and parameterize these tables to permit fast queries during *surface transport-only* precomputation to incorporate volume-to-point transport effects, and all at a minimal performance overhead[~4-15%](Section 5). Our results closely track ground truth obtained with volumetric path-tracing, despite operating several orders of magnitudes faster (Section 6). Indeed, existing volume-to-point transport simulation incurs a cost so large that it is normally omitted from most interactive graphics pipeline due to the impact on art direction and design iteration times.

4.1. Caching Lightmap Radiance in the Presence of Media

Consider the spherical radiance distribution $L(\mathbf{x}, \omega)$ *incident* at location $\mathbf{x} \in \{\mathcal{V} \cup \mathcal{A}\}$ either a lightmap texel or a location in free-space.

Instead of accumulating the contribution of many paths scattering at surfaces and in the volume, and this for many cached receiver locations $\mathbf{x}$, we will consider a bounce- and emitter-based decomposition of the problem. Specifically, by carefully parameterizing the *single-scattered* volumetric radiance arriving at a point from different emitter and reflector configurations, we will first show:

- how to compactly express the incident radiance distribution $L(\mathbf{x},\omega)$ for an *arbitrary* receiver-emitter configuration, for the three emission profiles (see Fig. 22), and
- how to shade lightmaps with this compact representation, composing the single-scattered response to analytically account for approximate multiple-scattering effects.

We will show that zonal harmonics are an effective basis for representing, shading, and sampling from the incident volume-to-point radiance distributions at cache locations in Section 4.2.

4.2. Spherical Impulse Response for Airlight Integrals

Consider the directly attenuated and single-scattered light arriving at $\mathbf{x}$ from a point emitter at $\mathbf{x}_s$ in a homogeneous, isotropic medium. The spherical incident radiance at $\mathbf{x}$ due to in-scattering and extinction $L^{(\leq 1)}(\mathbf{x},\omega)$, i.e., from light that has scattered zero and one times before arriving at $\mathbf{x}$, is a special case of Eqs. (3.1) and (3.3) with $L_o = L_e$ where the only source radiance is from the emitter, $L_e(\mathbf{x},\omega) \neq 0$ iff $\mathbf{x} = \mathbf{x}_s$.

In the case of point emitters, this spherical incident radiance exhibits circular symmetric about the axis from $\mathbf{x}$ to $\mathbf{x}_S$, and so we can represent it as $L^{(\leq 1)}(\mathbf{x},\omega) \approx \sum_l L_l^{(\leq 1)}\, y_l^0(\theta)$ with ZH coefficients

$$\begin{aligned} L_l^{(\leq 1)} = \frac{\sigma_t}{4\pi} \int_{\mathcal{S}^2} \bigg[\int_0^s & T_r(\mathbf{x}, \mathbf{x}_t)\, T_r(\mathbf{x}_t, \mathbf{x}_s) L_e(\mathbf{x}_s,\omega)\, \mathrm{d}t \\ & + T_r(\mathbf{x}, \mathbf{x}_s)\, L_e(\mathbf{x}_s,\omega) \bigg]\, y_l^0(\omega)\, \mathrm{d}\omega\,, \end{aligned} \tag{4.1}$$

where $L_e(\mathbf{x}_s,\omega) = I_0/(|\mathbf{x} - \mathbf{x}_S|^2)$ and I_0 is the point source's intensity. For a fixed receiver $\mathbf{x}$, emitter $\mathbf{x}_s$ and media $\{\sigma_t, \sigma_s\}$ configuration we compute the coefficients with Monte Carlo integration, importance sampling points $\mathbf{x}_t$ on the ray in the inner integrand with equi-angular

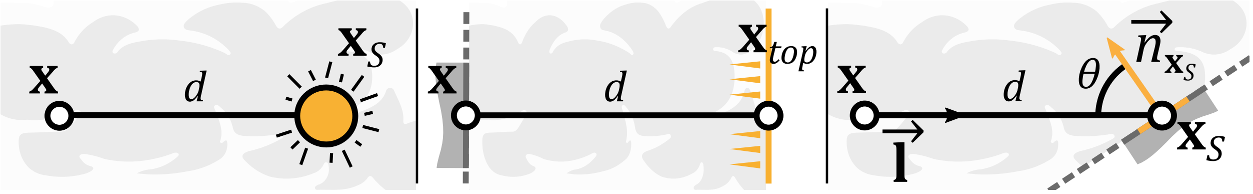


Fig. 22. The three canonical emission types we support: point, directional and differential surface area profiles.

sampling [**29**], and uniformly sample the outer spherical integrand with low-discrepancy quasi-Monte Carlo samples [**39**]. More advanced strategies exist, taking advantage of special parameterizations of SH [**18**] or semi-analytic integration schemes [**2**], but we found this yields acceptable results.

We can express the ZH-projected incident radiance at $\mathbf{x}$ due to directly attenuated and single-scattered light arriving from a *directional emitter*. It is important to note that the effects of transmission due to such an infinitely-distant source depends on the projection of the emitter's direction onto the boundary of the medium's bounding shape, which we denote $\mathbf{x}_{top}$ (Fig. 22, middle). For infinite media, the transmittance from *any point* to the "location" of the directional source is 0. The projection coefficients are,

$$L_l^{(\leq 1)} = \frac{\sigma_t}{4\pi}\int_{\mathcal{S}^2}\Bigg[\int_0^s T_r(\mathbf{x},\mathbf{x}_t)\, T_r(\mathbf{x}_t,\mathbf{x}_{top}) L_e(\mathbf{x}_{top},\omega)\,\mathrm{d}t + T_r(\mathbf{x},\mathbf{x}_{top})\, L_e(\mathbf{x}_{top},\omega)\Bigg]\, y_l^0(\omega)\,\mathrm{d}\omega \tag{4.2}$$

and we can compute Eq. (4.2) with the same MC scheme as before, with $L_e(\mathbf{x}_{top},\omega) = I_0\,\delta(\omega - \overrightarrow{\mathbf{x}\mathbf{x}_{top}})$ for distant directional sources.

The last emitter profile we consider is a differential area source centered at $\mathbf{x}_s$ with normal $\vec{n}_{\mathbf{x}_s}$ (Fig. 22, right). Unlike the point and directional emitters, we will use this source to not only model the effects of area lights but also the effects due to indirectly-reflected surface reflection from global illumination. Furthermore, the directly-attenuated and single-scattered incident radiance at $\mathbf{x}$ due to this source is not generally a circularly-symmetric function: as the source's normal deviates from the direction towards the receiver $\overleftarrow{\mathbf{x}\mathbf{x}_s}$, the radiance's spherical anisotropy increases, necessitating a full SH representation as opposed to a much more compact ZH one.

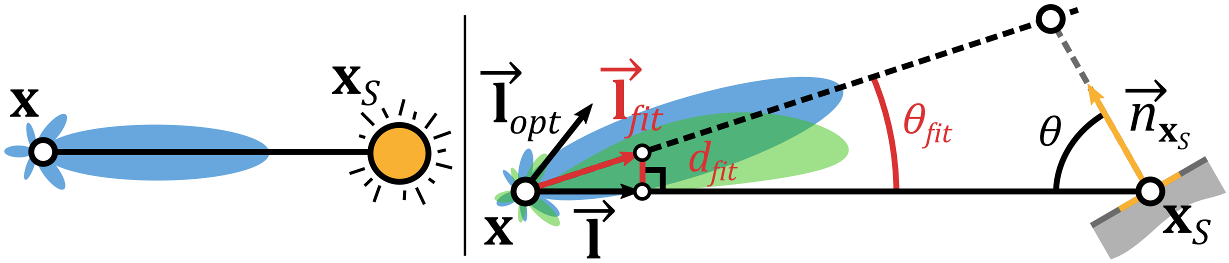


Fig. 23. Light scatterd from point (and directional) sources result in circularly symmetric radiance distributions at $\mathbf{x}$ (left; in blue). For differential area sources the distribution skews with the emitters normal and is no-longer circularly symmetric; we determine a best-fit circularly symmetric approximation (green).

Fortunately, we observe strongly unimodal (albeit off-axis) directionality in this distribution, even as the angle between $\vec{n}_{\mathbf{x}_s}$ and $\overleftarrow{\mathbf{x}\mathbf{x}_s}$ approaches $\pi/2$ (Fig. 23, right), motivating the following circularly-symmetric approximation: beginning from the full SH projection vector of the incident radiance at $\mathbf{x}$,

$$\mathbf{L}^{(\leq 1)} = \frac{\sigma_t}{4\pi}\int_{\mathcal{S}^2}\Bigg[\int_0^s T_r(\mathbf{x},\mathbf{x}_t)\,T_r(\mathbf{x}_t,\mathbf{x}_s) L_e(\mathbf{x}_s,\omega)\,\lfloor\vec{n}_{\mathbf{x}_s}\cdot\omega\rfloor\,\mathrm{d}t \\ + T_r(\mathbf{x},\mathbf{x}_s\,L_e(\mathbf{x}_s,\omega)\,\lfloor\vec{n}_{\mathbf{x}_s}\cdot\omega\rfloor\Bigg]\,\mathbf{y}(\omega)\,\mathrm{d}\omega\,,$$

one choice for the ZH approximation of $\mathbf{L}^{(\leq 1)}$ uses the *optimal linear direction* $\overrightarrow{\mathbf{l}_{opt}} = (-(L^{(\leq 1)})_1^1, -(L^{(\leq 1)})_1^{-1}, (L^{(\leq 1)})_1^0)$ as the axis of symmetry [**46**]. We found that the lowest-error ZH approximation's axis depends on the angle θ between $\vec{n}_{\mathbf{x}_s}$ and $\overleftarrow{\mathbf{x}\mathbf{x}_s}$, lying between the direction to the light $\vec{\mathbf{l}} \equiv -\overleftarrow{\mathbf{x}\mathbf{x}_s}$ and the optimal linear direction $\overrightarrow{\mathbf{l}_{opt}}$. We apply Brent's parabolic interpolation method to search for the best-fit ZH axis $\overrightarrow{\mathbf{l}_{fit}}$, repeating the search for many discretized incident angles. We store the angle θ_{fit} between $\overrightarrow{\mathbf{l}_{fit}}$ and $-\overleftarrow{\mathbf{x}\mathbf{x}_s}$, the optimized lobe axis $\overrightarrow{\mathbf{l}_{opt}}$ itself, and its ZH coefficients $L_{l,\theta}^{(\leq 1)}$ for every incident angle θ (Fig. 23, right),

$$L_{l,\theta}^{(\leq 1)} = \frac{\sigma_t}{4\pi}\int_{\mathcal{S}^2}\Bigg[\int_0^s T_r(\mathbf{x},\mathbf{x}_t)\,T_r(\mathbf{x}_t,\mathbf{x}_s)\,\lfloor\vec{n}_{\mathbf{x}_s}\cdot\omega\rfloor L_e(\mathbf{x}_s,\omega)\,\mathrm{d}t \\ + T_r(\mathbf{x},\mathbf{x}_s)\,L_e(\mathbf{x}_s,\omega)\Bigg]\,y_l^0\left((\overrightarrow{\mathbf{l}_{fit}}(\theta))\cdot\omega\right)\,\mathrm{d}\omega\,, \tag{4.3}$$

and $L_e(\mathbf{x}_s,\omega) = I_0/(|\mathbf{x}-\mathbf{x}_S|^2)$. We re-parameterize according to a *shift distance* $d_{fit} = \sin\theta_{fit}$ instead of the angle θ_{fit} to optimize the runtime. These impulse responses only hold for a single receiver-emitter and media configuration. They also only treat homogeneous media

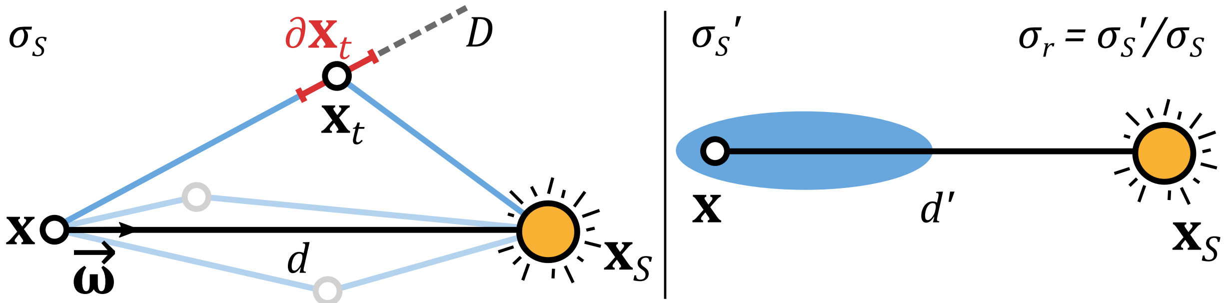


Fig. 24. Left: we compute and store the ZH impulse response at $\mathbf{x}$ for many distances d and a canonical σ_s. Right: when querying the tables in a scene with a different σ_s', we adjust the lookup distance $d_q = \sigma_r\, d'$ and scale the ZH coeffs. according to our σ-ratio σ_r.

with isotropic scattering. We show how to resolve each of these limitations in Sections 4.3 to 4.5 to compactly represent spherical impulse responses for *arbitrary* configurations, before detailing how to apply these responses when computing full, multi-bounce volume-to-point transport for precomputed lighting in Section 5.

4.3. Compact Impulse Response Parameterizations

We reduce from the 6D spatial configuration (3D for each of $\mathbf{x}$ and $\mathbf{x}_s$) to 1D tables for the point and directional sources using the relative distance $d_{\mathbf{x},\mathbf{x}_S} \equiv |\mathbf{x} - \mathbf{x}_S|$ between a receiver/shading point $\mathbf{x}$ and the point emitter location, and the latter by the distance $d_{\mathbf{x},\mathbf{x}_{top}} \equiv |\mathbf{x} - \mathbf{x}_{top}|$ to the nearest position on the boundary of the media $\mathbf{x}_{top}$ towards the directional source. The 2D ZH tables for differential area sources are parameterized by the shift distance d_{shift} **and** the (cosine of the) angle θ formed between the source area patch's normal $\vec{n}_{\mathbf{x}_s}$ and the direction $\overrightarrow{-\mathbf{l}}$ from the receiver to the patch (see Fig. 22). This works for fixed media parameters σ_t and σ_s, 3D and 4D tables would still be unwieldy.

Relative Media Coefficient Parameterization. Next, we eliminate the dependence on media parameter configurations by showing how to analytically map between impulse response coefficients tabulated from a canonical $\{\sigma_t,\sigma_s\}$ to those needed for an arbitrary medium's $\{\sigma_t',\sigma_s'\}$. To do so, we consider the manner in which solutions to the RTE behave when these parameters change. Begin by considering the impact of changing σ_s. Here, both the transmittance T_r and inscattering probability in Eq. (3.7) are affected. For $T_r(\mathbf{x},\mathbf{y}) = T_r(d)$ for $d \equiv \|\mathbf{x} - \mathbf{y}\|$, the optical thickness $-\sigma_s\, d$ behaves predictably as σ_s changes: modifying the optical thickness for varying σ_s it amounts to inversely modifying d. Specifically, $\exp -\sigma_s d =$

$\exp -\sigma_s' d'$ when $d = (\sigma_s' \big/ \sigma_s) d'$. From this relation, we can precompute the inscattering impulse response for some canonical σ_s account for changes in the actual media parameters in two simple steps: first, we adjust the table distance query as $d_q = (\sigma_s' \big/ \sigma_s) d'$; second, as inscattered events have new scattering probability σ_s', we account for this discrepancy by scaling the queried ZH coefficients by $\sigma_r \equiv \sigma_s' \big/ \sigma_s$, which we call the σ-ratio.

With point and differential area sources, we need to also consider the effect on emission intensity $I_0 \big/ d^2$, which fall-off with an inverse square profile. When querying the tables with an adjusted d_q, the intensity fall-off must be similarly adjusted by the σ-ratio, as per $I_0 \big/ [\sigma_r\, d]^2$. This adds an additional factor of σ_r^2 for these types of sources that needs to be modeled when computing their inscattered contribution over a single-scattered ray in Eqs. (3.1) and (3.3). When integrating over the inscattered ray, and over a distance adjusted by σ_r, a factor of σ_r arises from the integration and cancels one of the additiona $1/\sigma_r$ intensity adjustment terms, above.

As such, point and differential area entries are ultimately adjusted by a factor of σ_r^2, and directional source entries remain unaffected.

Accounting for Absorption. While less often used in game assets, we can also support media absorption with $\sigma_t = \sigma_a + \sigma_s'$ is used. The adjusted query distance and coefficient weighing scheme need only rely on the appropriately modified σ-ratio of $\sigma_r = \sigma_t \big/ \sigma_s$.

We note that including absorption will preclude an efficient table composition method we propose for multiple-scattering in Section 5.1. We will show that, since only the scattering coefficient drives inscattering probability, in the absence of absorption we can arrive at a single table that encodes the ZH response for *all of the media interactions*: direct attenuation, single-scattering and an arbitrary number of multiple-scattering bounces. To do so, we will employ a multi-bounce generalization of the σ-ratio. Figure 24 summarizes the adjustments necessary for arbitrary media parameters.

Next we discuss how to extend the impulse response formulations to heterogeneous (Section 4.4), anisotropic (Section 4.5) media.

4.4. Impulse Response in Heterogeneous Media

In games, homogeneous media are most often used to model the effects atmospheric scattering, but in closed environments the predominant form of participating media used in

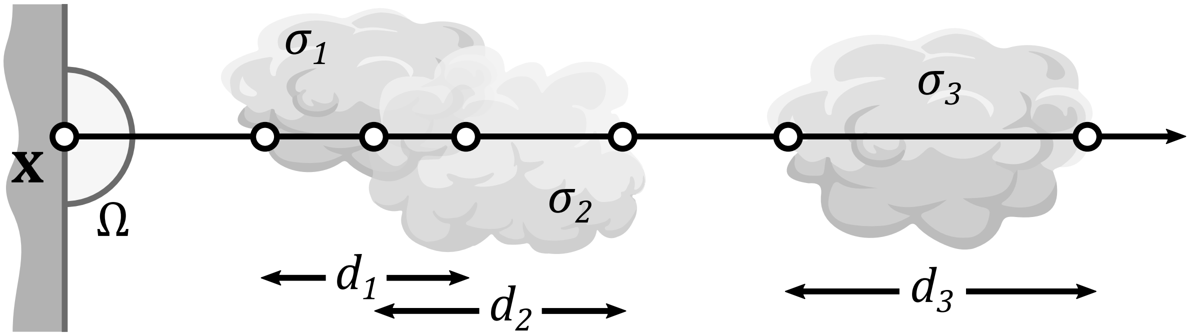


Fig. 25. Heterogeneous media is approximated by spherically-homogeneous media from the point of view of $\mathbf{x}$. Here, the ray's optical thickness is $d_1\sigma_1 + d_2\sigma_2 + d_3\sigma_3$. The ray's scattering coefficient is obtained by normalizing optical thickness by ray length.

interactive media are heterogeneous – with spatially-varying density and scattering parameters. Artists author this media with many strategies, including layering procedural patterns (e.g., Perlin noise), storing and combining volumetric "brushes" with hierarchical data structures (e.g., oriented bounding boxes, octrees), or by "sweeping" 2D image stencils (e.g., gradients, exponential ramps, binary masks).

One of the largest complications due to heterogenous media is the added complexity of evaluating the transmittance T_r, as this necessitates some ray marching-based solution in the setting of general heterogeneity. We make the following simplification to enable another remapping of our (homogeneous) impulse response coefficients to heterogeneous media: we assume that the media acts homogeneously for any fixed ray in space, but that the parameters of the per-ray "effective homogeneity" can vary per ray. This can be interpreted as replacing heterogeneous media with a spatio-directionally-varying homogeneous media: every ray in a scene "observes" a different homogeneous medium. The effective scattering parameter can be efficiently computed: $\sigma_s^{\text{effective}} = \left(\sum_i \sigma_{s,i}\, d_i\right)\Big/ D$, where D is the total ray length (Fig. 25).

We can now apply σ-ratio remapping to our impulse response tables (Section 4.3) with $\sigma_s^{\text{effective}}$ to query the table entries. Note that our assumption for heterogeneous media does not introduce any approximation for the directly-attenuated volumetric light, since the product of transmittance (for discrete changes in density) along the ray is equal to the transmittance of the average density over the ray: $\exp\left(-\sum_i \sigma_{s,i}\, d_i\right)\Big/ D = \Pi_i \exp\left(-\sigma_{s,i}\, d_i\right)$. The single-scattered impulse contribution will, however, incur approximation error since it integrates rays that fall off-axis from the central ray used to estimate the effective homogeneous scattering coefficient.

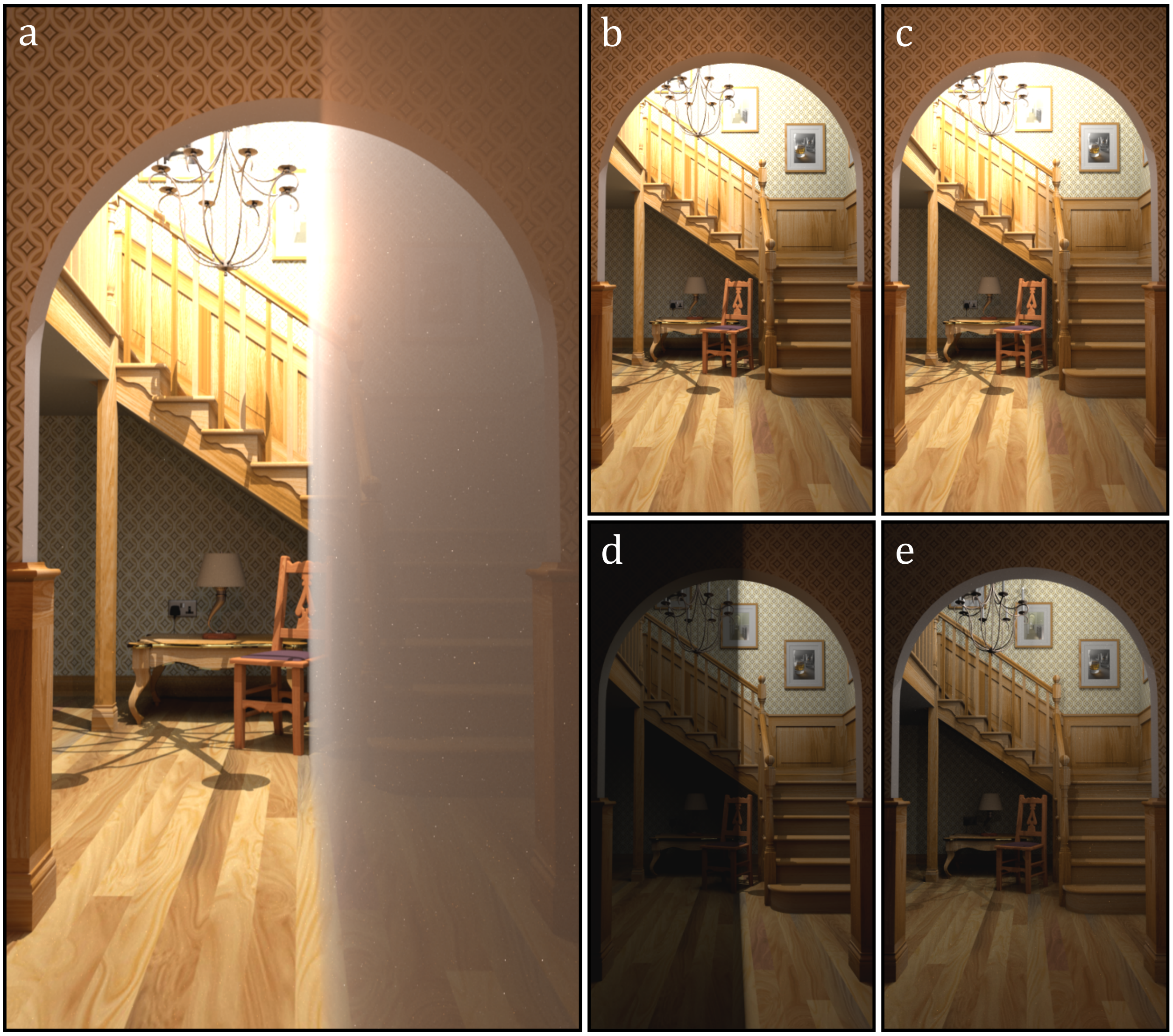


Fig. 26. Comparing our method to ground truth generated in Mitsuba [**17**] in an extreme scenario. The final rendering (a) includes eye-ray attenuation atop the final precomputed surface lightmaps. Our lightmap (b) and the ground truth lightmap (c) include volume-to-surface transport from directly attenuated (but unoccluded, in our case) lights, and single- and multiple-scattering. Artifacts are most evident when only visualizing inscattering (d) versus ground truth (e): these are due to the extreme heterogeneous discontinuity, the positioning of light sources outside the medium, and surface-to-light connections that do not receive inscattering when not intersecting the volume.

We only noticed visible artifacts in degenerate cases, e.g., binary media; see Fig. 26 for an failure case under extreme conditions.

4.5. Impulse Response with Anisotropic Phase Functions

We consider the Henyey-Greenstein phase function (HG), parameterized by a single parameter $g \in [-1{,}1]$, where $g > 0$ favors forward scattering, $g < 0$ favors backward scattering, and $g = 0$ is isotropic scattering. When tabulating the canonical impulse response coefficients (Section 4.2) we compute three separate tables, one for each of $g = \{-g_{range}, 0, g_{range}\}$, where g_{range} is an artist-chosen parameter depending on the degree of desired anisotropy. All our results use $g_{range} = 0.5$. We do this only for the directly-attenuated and single-scattered tables, as we only treat isotropic scattering for multiple-scattering (see Section 5.1).

We track both an effective scattering coefficient $\sigma_s^{\text{effective}}$ **and** an effective HG anisotropy $g^{\text{effective}}$ per ray during lighting precomputation (Section 5). Given $g^{\text{effective}}$, we linearly interpolate the ZH coefficients from the three g-dependent impulse response tables and, for multiple-scatter (Section 5.1) we adjust our query σ_s to be $(1 - g)\sigma_s$ according to similarity theory [**50**].

This is equivalent to building a discrete function space $b_i(\theta)$ to parameterize the space of phase functions, such that any phase function can be expressed as $f_p(\theta) = \mathbf{w} \cdot \mathbf{b}(\theta) = \sum_i w_i \, b_i(\theta)$, and then computing single-scattered impulse response *matrices* $\mathbf{M}$ with ZH response coefficient columns due to each basis phase function b_i. The matrix-vector product $\mathbf{M} \cdot \mathbf{w}$ would result in the appropriate ZH impulse response function for the arbitrary phase function. This does not scale to multiple scattering events: K-scattering events requires a $(K + 1)$-dimensional *tensor*, that has to be collapsed to a response vector for the given per-bounce scattering parameters along the ray. With N basis phase functions, you would require $N^{(K+1)}$ coefficients for the tensor, which is impractical.

5. Using Impulse Responses to Precompute Lighting

Given pretabulated impulse response coefficients (Section 4.2) and approaches to query them for arbitrary receiver, emitter and media configurations (Sections 4.3 to 4.5), we first detail the process of the incorporating volume-to-point effects from volume attenuated and single-scattered contributions due to direct- and indirect-illumination from surfaces and emitters, below. We then discuss our treatment of multiple-scattering and the effects of occlusion on our approximation (Sections 5.1 and 5.2).

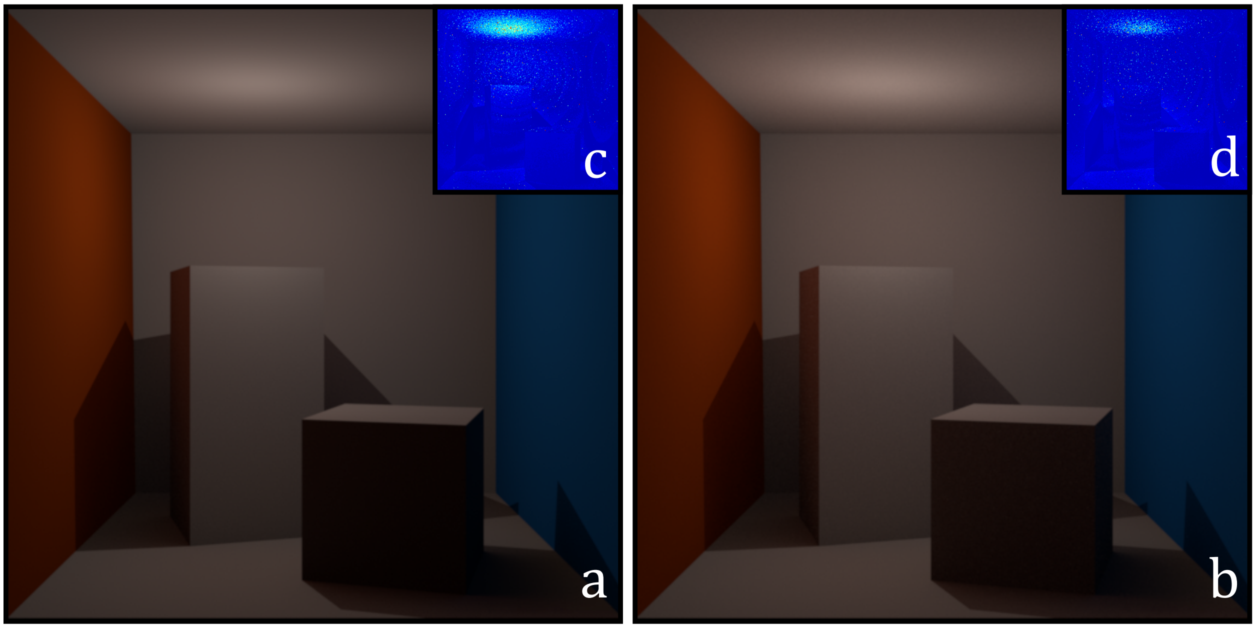


Fig. 27. Cornell box with homogeneous media, 10-bounce inscattering-only without (a) and with (b) back-scattering. (c) and (d) are false-color ground truth difference images at 3EV.

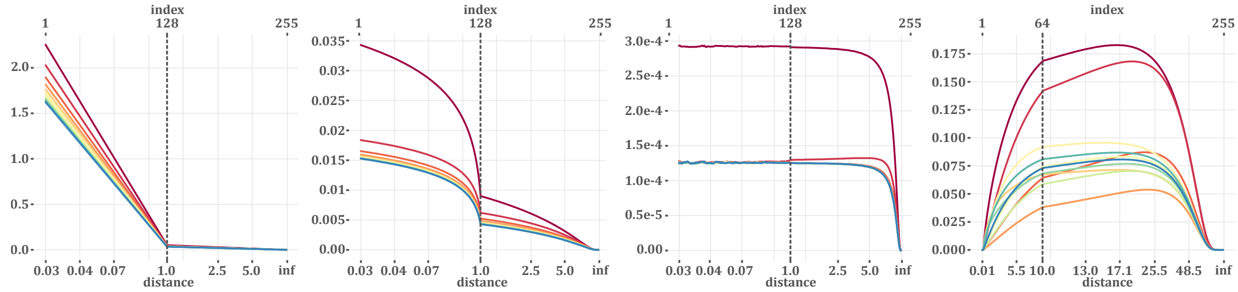


Fig. 28. Order-11 ZH response coefficients (dark red to blue) vs. canonical distances $d_{\mathbf{x},\mathbf{x}_s}$ or $d_{\mathbf{x},\mathbf{x}_{top}}$: 1-, 2- and 10-bounce response for point emitters, and 1-bounce for directional emitters. Table bin indices are on the top x-axis. Radiance tends to isotropy with higher bounces, increasing the ZH DC component (dark red) as expected. Differential area emitter plots (not shown) behave similarly to the point emitter.

From Surface-only Baking to Volumetric Transport. Integrating volume-to-point interactions atop standard path-tracers using in surface-transport light baking is straightforward. Moreover, we effectively avoid the cost and complexity of tracing additional volumetric paths inside media and between the media and surfaces.

We traces surface-transport paths, but we keep track of when rays enter and exit media. For any path vertex $\mathbf{x}_i$ with a ray entering or exiting media, we weigh the radiance $L(\overleftarrow{\mathbf{x}_i\mathbf{x}_{i+1}})$ to $\mathbf{x}$ from the direction towards the next path vertex $\overleftarrow{\mathbf{x}_i\mathbf{x}_{i+1}}$ by the appropriately queried ZH impulse response vector at $\mathbf{x}$: determined by tracking the (potentially heterogeneous) σ_s along the ray for the path segment $\mathbf{x}_i\mathbf{x}_{i+1}$ (Section 4.4) and computing the necessary σ-ratio and g (Sections 4.3 and 4.5). Weighing this ZH vector amounts to replacing I_0 in the L_e terms of Eqs. (4.1) to (4.3) with $L(\overleftarrow{\mathbf{x}_i\mathbf{x}_{i+1}})$.

Note that when $\mathbf{x}_{i+1}$ lies on an emitter (i.e., for explicit direct lighting connections in path tracing), the weighted ZH vector models the incident radiance due to directly attenuated light emission and single scattering from the light source; when $\mathbf{x}_{i+1}$ lies on another surface (i.e., for implicit indirect lighting connections), it models the incident radiance due to directly attenuated indirect surface radiance and single-scattered events that arise from a bounce of light from indirectly lit surfaces into the volume and then towards the receiver $\mathbf{x}$. For $\mathbf{x}_{i+1}$ on emitters, we use the point or directional lookup tables for these types of emitters, and the differential area table for area sources. For $\mathbf{x}_{i+1}$ on (indirectly-scattering) surfaces, we query the differential area table. When querying the differential area table, we need to also compute the angle between $\overrightarrow{\mathbf{x}_i\mathbf{x}_{i+1}}$ and $\vec{n}_{\mathbf{x}_{i+1}}$.

Given the queried ZH incident radiance vector $\mathbf{l}_{in}$ at $\mathbf{x}$, the final step in precomputed shading is to compute the outgoing radiance contribution at $\mathbf{x}$, i.e., in the lightmap. For lightprobes, we typically directly store the incident radiance $\mathbf{l}_{in}$ and convert to outgoing radiance when shading dynamic objects, using the same process we detail next for the lightmap shading case. Given a cosine-weighted BRDF f_r and ZH incident radiance $L_{in}(\mathbf{x}, \omega_i) = \mathbf{l}_{in} \cdot \mathbf{y}(\omega_i)$, we wish to compute the double-product integral

$$\int_{\mathcal{S}^2} f_r(\mathbf{x}, \omega, \omega_i)\, L_{in}(\mathbf{x}, \omega_i)\, \mathrm{d}\omega_i = \mathbf{f_r}(\omega) \cdot \mathbf{l}_{in} \text{ from Section 3.3,} \tag{5.1}$$

where are the SH coefficients of the view-evaluated cosine-weighted BRDF $\mathbf{f_r}(\omega)$ at $\mathbf{x}$, rotated to align in the local coordinate frame of the incident radiance's axis of alignment. In general, evaluating Eq. (5.1) can be costly for arbitrary BRDFs since their projection coefficients would have to either be pre-tabulated for many outgoing directions ω or computed on-the-fly. Moreover, the SH rotation incurs an additional cost; alternatively, we can rotate the ZH incident radiance more efficiently (Section 3.3) to the local frame at $\mathbf{x}$.

With special-case circularly symmetric BRDFs, like Lambertian or Phong, their (potentially view-evaluated) projection coefficients can be expressed in ZH along an axis of symmetry, then rotated into an appropriate frame for shading. This still requires expanding at least one of the two terms in Eq. (5.1) into full SH. When both terms in the double-product integrand exhibit circular symmetry, but along different axes, we develop a new fast-ZH shading formulation that avoids expanding any of the two terms into full SH (Section 8). We use this formulation in these special-case scenarios, typically on diffuse surfaces in the lightmap where the cosine-weighted BRDF is a circularly-symmetric function about the surface $\vec{n}_{\mathbf{x}}$ normal at $\mathbf{x}$, with analytically obtainable ZH coefficients [**40**].

5.1. Single- and Multiple-scattering Tables

Without absorption, we use our single-scattering tables to accelerate the computation of multiple-scattered ZH incident radiance impulse vectors: much like how indirect bounces are precomputed in surface-based PRT, where indirect ray intersections accumulate transport vector contributions from previous bounce's PRT simulation [**45**], we can recompute the MC integral estimates of the directly-attenuated and single-scattered responses in Eqs. (4.1) to (4.3), but each time now replacing the L_e terms with the previous bounce's ZH inscattered response evaluated in the appropriate direction using the SH/ZH expansion equation in Section 3.3. Here, the same σ-ratio, heterogeneous and HG phase function remappings (Sections 4.3 to 4.5) can be applied at multiple-scattered bounces, and we can combine scattering impulse response vectors across each of these bounces into a **single lookup table**.

When $\sigma_a \neq 0$ we need to individually tabulate the multiple-scattering and, in the case of anisotropic media, we adjust isotropic scattering according to a similarity theory mapping. Specifically, we weight the ZH response of each bounce-b of multiple scattering by a factor of $\left(\sigma'_s \middle/ \sigma_t\right)^b$, as per the discussion concerning the σ-ratio adjustments in the presence of absorption in Sections 4.2 and 4.5.

We always tabulate order-11 ZH response, with 256 bins for discretized distances $d_{\mathbf{x},\mathbf{x}_{top}}$ or $d_{\mathbf{x},\mathbf{x}_s}$, and eight discrete angle (really, *shift distance*) bins. Fig. 28 illustrates the form of the 11-vector ZH response, and the tabulation ranges and steps, for our tables. As the number of scattering bounces increases the signal tends towards lower directionality (i.e., high-order coefficients lose magnitude relative to low-order ones).

Fig. 29. In-game screenshot with heterogeneous media (a). Lightmaps without (b) and with (c) our volume-to-surface method: we capture media-attenuated direct and indirect light from surfaces (d) and volumetric single- and multi-scattering from emitters, surfaces and media (e).

5.2. Occlusion and Back-scattered Contributions

In general, light may scatter around occluders in participating media (Fig. 21) and our impulse responses do not model this effect. After analyzing the solid angle subtended by the impulse response lobes, however, we found that single-scattered response is limited to angular extents between 5-to-6°, across all configurations. This reduces the impact of ignoring occluders during pre-tabulation, and we further mitigate this by testing for occlusion with direct emission rays and only computing impulse contributions for unoccluded rays.

Another important factor to take into consideration is light that scatters volumetrically off of surfaces that are *behind* (or co-planar with) $\mathbf{x}$, before arriving back at $\mathbf{x}$. We observe empirically that this *back-scattered contribution* can be significant, and so we compensate for it during impulse response MC pre-tabulation, as follows: we slightly shift $\mathbf{x} \to \mathbf{x} + \epsilon\,\vec{n}_{\mathbf{x}}$ and sample incident rays over the *entire sphere of directions* $\mathcal{S}^2$, instead of just about the hemisphere Ω at $\vec{n}_{\mathbf{x}}$, when accumulating ZH impulse response. We always set $\epsilon = 0.125$. We artificially set the distances of these back-scattered ray intersections to 0, as if $\mathbf{x}$ had not been shifted at all. Fig. 27 illustrates the impact of the back-scattered contribution.

6. Results & Discussion

We implement our method in a game studio's baking pipeline, supporting volume-to-surface transport for light maps, light grids and lightprobe-based meshes. Our implementation uses Intel® Embree ray tracing kernels and runs on the CPU. The runtime engine models single scattering to the eye using a modified implementation of existing work [**49**], ignoring the effects of media between the light source and froxel grid and only treating media back to the eye.

For large-scale production scenes with both homogeneous air density and dozens of heterogeneous media volumes, adding volume-to-surface transport with our method incurs a modest ~4 to 15% overhead atop *surface-only baking.* We benchmark using high- and lower-quality tracing settings to differentiate between final bake and iterative workflow modes. For the same scene (shown from viewpoints in Figs. 19, 29 and 30), baking takes **10.5 / 3.7 minutes** without and **12 / 3.8 minutes** with our volume-to-surface transport, for **high-/low-**quality: corresponding to a **14.6% / 4.7%** overhead. Runtime performance remains unchanged.

Limitations & Future Work. We discuss the approximation in single- and multiple-scattering (but not directly attenuated radiance) due to our treatment of heterogeneous media (Section 4.4), and addressing this limitation can further improve our accuracy (see Fig. 26).

Ignoring effects of light scattering around occluders is reasonable for single-scattering, due to the limited angular extent of this impulse response, the impact on multiple-scattering can be larger (i.e.,around a corners or small occluders; Fig. 21, bottom right). The amount of energy in multiple-scattered events diminishes quickly for thin media [**27**]. Better modeling local occlusion to compensate for this lost energy is an avenue we leave for future work. Handling glossy materials might require investigating representations that allow for higher angular frequencies.

7. Conclusion

We present a method for incorporating volumetric transport during lighting precomputation for interactive graphics applications. Our results agree closely to ground truth path tracing, at a fraction of the cost: we incorporate volumetrically-attenuated emission, single- and multiple-scattering from volumetric media with a modest overhead of 4-15% compared

Fig. 30. Homogeneous media lit by point and directional emitters. Final renderings (a, b) include scattering towards the eye, lightmap visualizations (c, d) only include indirect light. Only treating volumetric extinction (a, c) fails to capture important volume-to-surface effects. Our method (b, d) accounts for these with minimal overhead, avoiding costly multi-bounce media path tracing.

to *existing surface-only precomputation times.* It is straightforward to implement in existing pipelines, and has been implemented at a large gaming studio. Our impulse response tables are compact and need to be precomputed only once.

Our spherical impulse response analysis of directly-attenuated and single-scattered airlight integrals validates the utility of ZH representations for this problem. We generalize our canonical (i.e., unit homogeneous, isotropic) impulse response analysis to account for media with arbitrary scattering, extinction and absorption coefficients, arbitrary phase functions, and heterogeneity. Our algorithm is parallelizable and, as we target the static lighting pipeline, our method requires no added runtime cost. Despite this, the subtle but important lighting cues that we account for with volume-to-surface/point transport allows digital content artists to better employ media when developing complex and realistic virtual environments.

8. Appendix A: Efficient Cosine Double-product Integration

Computing Eq. (5.1) with a diffuse BRDF is required to convert the ZH incident radiance from our impulse response into outgoing radiance. We derive an efficient $O(N)$ integration scheme that avoids the $O(N^2)$ computation of SH coefficients for either of these (independently symmetric) factors.

We use the *SH addition theorem*, that expresses ZH rotated along any axis $\bar{\omega}$ as a product of its canonical z-axis oriented coefficients n_l^* and the value of SH basis functions evaluated at $\bar{\omega}$: $z_l^0(\omega \to \bar{\omega}) = \sum_l n_l^* \sum_m y_l^m(\omega) y_l^m(\bar{\omega})$.

The symmetry of the two ZH expansions allows a coordinate system where the first function $g(\theta)$ (with coefficients g_l) is aligned about a canonical axis ω_c and the second $h(\theta)$ (with h_l) is related to g by the angular difference in their axes. Reconstructing g along this axis is non-zero only for the $m = 0$ ZH, and h need only be evaluated at these coefficients, simplifying its reconstruction to $h(\omega) = \sum_l g_l\, y_l^0(\omega_c)\, h_l\, y_l^0(\omega \cdot \bar{\omega})$, where ω is ω_c in the transformed space and $(\omega \cdot \bar{\omega})$ is the cosine of the angle between the axes.

SH reconstruction is

$$h(\omega) = \sum_l \frac{g_l}{y_l^0(\omega_c)}\, y_l^0(\omega_c) \frac{h_l}{y_l^0(\omega_c)}\, y_l^0(\omega \cdot \bar{\omega}) = \sum_l g_l\, h_l\, y_l^0(\omega \cdot \bar{\omega}) \Big/ y_l^0(\omega_c)\,.$$

References

[1] Michael Abrash. Quake's lighting model: Surface caching, 2000.

[2] Laurent Belcour, Guofu Xie, Christophe Hery, Mark Meyer, Wojciech Jarosz, and Derek Nowrouzezahrai. Integrating clipped spherical harmonics expansions. *ACM Trans. on Graphics*, March 2018.

[3] Benedikt Bitterli and Wojciech Jarosz. Beyond points and beams: Higher-dimensional photon samples for volumetric light transport. *ACM Trans. on Graphics*, 36(4), jul 2017.

[4] Danny Chan. Material advances in Call of Duty: WWII. In *Advances in Real-Time Rendering in Games*, SIGGRAPH Courses, 2018.

[5] Subrahmanyan Chandrasekhar. *Radiative Transfer.* Dover, 1960.

[6] Hao Chen. Lighting and material of Halo 3. In *Game Developers Conference*, 2008.

[7] Eugene d'Eon. *A Hitchhiker's Guide to Multiple Scattering.* 2016.

[8] Luca Fascione, Johannes Hanika, Marcos Fajardo, Per Christensen, Brent Burley, Brian Green, Rob Pieké, Christopher Kulla, Christophe Hery, Ryusuke Villemin, Daniel Heckenberg, and André Mazzone. Path tracing in production. In *SIGGRAPH Courses*, 2017.

[9] Julian Fong, Magnus Wrenninge, Christopher Kulla, and Ralf Habel. Production volume rendering. In *SIGGRAPH Courses*, pages 2:1–2:79, 2017.

[10] Gene Greger, Peter Shirley, Philip M. Hubbard, and Donald P. Greenberg. The irradiance volume. *IEEE Computer Graphics and Applications*, pages 32–43, March 1998.

[11] Miloš Hašan, Fabio Pellacini, and Kavita Bala. Direct-to-indirect transfer for cinematic relighting. In *ACM Trans. on Graphics*, pages 1089–1097, 2006.

[12] Sébastien Hillaire. Towards unified and physically-based volumetric lighting in Frostbite. In *Advances in Real-Time Rendering in 3D Graphics and Games*, SIGGRAPH Courses, 2015.

[13] Sébastien Hillaire. Real-time raytracing for interactive global illumination workflows in Frostbite. In *Game Developers Conference*, 2018.

[14] David S. Immel, Michael F. Cohen, and Donald P. Greenberg. A radiosity method for non-diffuse environments. In *Computer Graphics (Proc. of SIGGRAPH)*, pages 133–142, 1986.

[15] Michał Iwanicki and Peter-Pike Sloan. Ambient dice. In *Proc. of EGSR - EI&I Track*, 2017.

[16] Michał Iwanicki and Peter-Pike Sloan. Precomputed lighting in Call of Duty: Infinite Warfare. In *Advances in Real-Time Rendering in Games*, SIGGRAPH Courses, 2017.

[17] Wenzel Jakob. Mitsuba renderer, 2010. http://www.mitsuba-renderer.org.

[18] Wojciech Jarosz, Nathan A. Carr, and Henrik Wann Jensen. Importance sampling spherical harmonics. *Computer Graphics Forum*, page 577–586, apr 2009.

[19] Wojciech Jarosz, Craig Donner, Matthias Zwicker, and Henrik Wann Jensen. Radiance caching for participating media. *ACM Trans. on Graphics*, 27(1):7:1–7:11, mar 2008.

[20] Wojciech Jarosz, Derek Nowrouzezahrai, Iman Sadeghi, and Henrik Wann Jensen. A comprehensive theory of volumetric radiance estimation using photon points and beams. *ACM Trans. on Graphics*, 30(1):5:1–5:19, jan 2011.

[21] Wojciech Jarosz, Derek Nowrouzezahrai, Robert Thomas, Peter-Pike Sloan, and Matthias Zwicker. Progressive photon beams. *ACM Trans. on Graphics*, 30(6), dec 2011.

[22] Wojciech Jarosz, Matthias Zwicker, and Henrik Wann Jensen. The beam radiance estimate for volumetric photon mapping. *Computer Graphics Forum*, page 557–566, apr 2008.

[23] Wojciech Jarosz, Matthias Zwicker, and Henrik Wann Jensen. Irradiance gradients in the presence of participating media and occlusions. *Computer Graphics Forum*, page 1087–1096, jun 2008.

[24] Henrik Wann Jensen and Per H. Christensen. Efficient simulation of light transport in scenes with participating media using photon maps. In *Annual Conference Series (Proc. of SIGGRAPH)*, pages 311–320, 1998.

[25] Henrik Wann Jensen, Stephen R. Marschner, Marc Levoy, and Pat Hanrahan. A practical model for subsurface light transport. In *Annual Conference Series (Proc. of SIGGRAPH)*, 2001.

[26] James T. Kajiya. The rendering equation. In *Computer Graphics (Proc. of SIGGRAPH)*, pages 143–150, 1986.

[27] Simon Kallweit, Thomas Müller, Brian McWilliams, Markus Gross, and Jan Novák. Deep scattering: Rendering atmospheric clouds with radiance-predicting neural networks. *ACM Trans. on Graphics*, 36(6), November 2017.

[28] Ladislav Kavan, Adam W. Bargteil, and Peter-Pike Sloan. Least squares vertex baking. In *Computer Graphics Forum*, pages 1319–1326, 2011.

[29] Christopher Kulla and Marcos Fajardo. Importance sampling techniques for path tracing in participating media. *Computer Graphics Forum*, pages 1519–1528, June 2012.

[30] Jurgen Laurijssen, Rui Wang, Philip Dutré, and B. J. Brown. Fast estimation and rendering of indirect highlights. In *Computer Graphics Forum*, pages 1305–1313, 2010.

[31] Jaakko Lehtinen. A framework for precomputed and captured light transport. *ACM Trans. on Graphics*, 26(4), October 2007.

[32] Julio Marco, Adrian Jarabo, Wojciech Jarosz, and Diego Gutierrez. Second-order occlusion-aware volumetric radiance caching. *ACM Trans. on Graphics*, 37(2), apr 2018.

[33] Gary McTaggart. Half-Life 2 source shading. In *Game Developers Conference*, 2004.

[34] Thomas Müller, Marios Papas, Markus Gross, Wojciech Jarosz, and Jan Novák. Efficient rendering of heterogeneous polydisperse granular media. *ACM Trans. on Graphics*, 35(6):168:1–168:14, dec 2016.

[35] Jonathan T Moon, Bruce Walter, and Stephen R Marschner. Rendering discrete random media using precomputed scattering solutions. In *Computer Graphics Forum*, pages 231–242, july 2007.

[36] David Neubelt and Matt Pettineo. Advanced lighting R&D at Ready at Dawn Studios. In *Physically Based Shading in Theory and Practice*, SIGGRAPH Courses, pages 22:1–22:8, 2015.

[37] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte Carlo methods for volumetric light transport simulation. *Computer Graphics Forum*, May 2018.

[38] Yuriy O'Donnell. Precomputed global illumination in Frostbite. In *Game Developers Conference*, 2018.

[39] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically Based Rendering: From Theory To Implementation.* 3rd edition, 2016.

[40] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *Annual Conference Series (Proc. of SIGGRAPH)*, pages 497–500, 2001.

[41] Jorge Schwarzhaupt, Henrik Wann Jensen, and Wojciech Jarosz. Practical hessian-based error control for irradiance caching. *ACM Trans. on Graphics*, 31(6), nov 2012.

[42] Ari Silvennoinen and Jaakko Lehtinen. Real-time global illumination by precomputed local reconstruction from sparse radiance probes. *ACM Trans. on Graphics*, 36(6):230:1–230:13, November 2017.

[43] Peter-Pike Sloan. Stupid spherical harmonics (SH) tricks. In *Game Developers Conference*, 2008.

[44] Peter-Pike Sloan, Jesse Hall, John Hart, and John Snyder. Clustered principal components for precomputed radiance transfer. *ACM Trans. on Graphics*, 22(3):382–391, July 2003.

[45] Peter-Pike Sloan, Jan Kautz, and John Snyder. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Trans. on Graphics*, 21(3):527–536, July 2002.

[46] Peter-Pike Sloan, Ben Luna, and John Snyder. Local, deformable precomputed radiance transfer. *ACM Trans. on Graphics*, 24(3):1216–1224, July 2005.

[47] Bo Sun, Ravi Ramamoorthi, Srinivasa G. Narasimhan, and Shree K. Nayar. A practical analytic single scattering model for real time rendering. *ACM Trans. on Graphics*, 24(3):1040–1049, July 2005.

[48] Magnus Wrenninge and Nafees Bin Zafar. Fundamentals. In *Production Volume Rendering*, SIGGRAPH Courses, 2011.

[49] Bartlomiej Wronski. Volumetric fog: Unified compute shader-based solution to atmospheric scattering. In *Advances in Real-Time Rendering in 3D Graphics and Games*, SIGGRAPH Courses, 2014.

[50] Shuang Zhao, Ravi Ramamoorthi, and Kavita Bala. High-order similarity relations in radiative transfer. *ACM Trans. on Graphics*, 33(4):104:1–104:12, July 2014.

Conclusion

In this dissertation we have introduced two novel methods for caching and reusing prior lighting computation to accelerate path-traced rendering for interactive and precomputed real-time applications.

Speeding up light transport computations is an active field of research as they are the standard for production renderers, whether it is in the gaming or movie industry. Physically-based renderers have become ubiquitous and have completely changed the authoring pipelines present in the industry. The tools that artists and designers use to model worlds and effects need to showcase what the final output of on-going work looks like, and fast workflow and iteration time have become targets when conceiving new rendering techniques for authoring tools, i.e., the time-to-first-frame is now a common goal in and of itself. Adaptive and progressive rendering have by this standard become essential, as is the use of memory caches to avoid recomputing what needn't.

At the same time, the quality of the final frame has increased phenomenally in games, especially considering the frame time restrictions. This has in part been due to an increased use of global illumination solutions, bringing scenes to life through indirect lighting. Still, trade-offs are made and increased quality is often possible only by setting constraints on the dynamicality of the system. Thus, precomputation methods for global illumination still dominate, adding load to the authoring pipeline. By its nature, indirect lighting is hard to decouple from itself and modifying the environment implies having to recompute light transport, making the process fastidious if too slow.

Our first paper presented an adaptive sampling and reconstruction technique for direct illumination, allowing us to shade only a fraction of the pixels and amortizing rendering cost. To this end we developped new frequency bandwidth estimates to drive both sampling and reuse across animation frames. We adequately sample and compute lighting contribution when necessary as our frequency criterion considers light emission, travel, occlusion and shading bandwidths. Finally, our lightweight cache handles the life time of these contributions following the same criterion. The principal obstacle during the development of this technique has been the question of occlusion. As the bandwidth estimate is exact for the unnocludded case, rendering correct shadows was the principal source of error. Visibility being such a high frequency signal, investigating a non-frequency-driven solution for visibility that would pair with the unnoccluded bandwidth estimate would be interesting. Another intriguing extension would be to implement this frequency-driven method to recent GPU ray-tracing frameworks as they are particularly optimized for coherent primary ray tracing. Extending the method to indirect transport isn't trivial, potentially through a progressively estimated indirect bandwidth. Doing this would mean a change in the caching and reuse scheme, as previously cached sample points would need updating. A solution based on sample connections that would keep computed points longer in the cache as secondary elements is also interesting to think about.

The second paper presents a novel analysis of in-scattered light in participating media which we formulate as impulse responses. This operator allows a complete decoupling from any particular medium and is tabulated one time only. In a specific scene-dependent setting, this zonal harmonics transport parameterization is efficiently queried for varying potentially-heterogeneous media densities and handles multiple scattering and anisotropic phase functions. As a result, the usual multi-dimensional extension of scattered light transport in media can be folded back to a simple table lookup, and we implemented our method in a modern AAA video game light baking tool. There are limitations to this method which have mainly to do with occlusion. We only model in-scattering if there is a direct connection between the surface and the light source and we left the investigation of the effect of occluding geometry between the two for future work. It would be interesting to extend the model to arbitrary orientations of the receiving surface, i.e., with normal in the same hemisphere as the light direction, though such a solution would also necessitate a different handling of the lighting bake using our tables afterwards, as direct visibility is not a predicate of a potential contribution anymore. Our method works well with heterogeneous media, but modern renderers make heavy use of animated media such as rolling fog or flowing smoke. Due to the low frequency of indirect lighting, approximating dynamic media with its time-averaged density yields a decent estimate but tailored solutions have to be designed for more extreme cases. This joins the category of solutions that have to be thought through in the near future as precomputation algorithms and precomputed data become more dynamic.