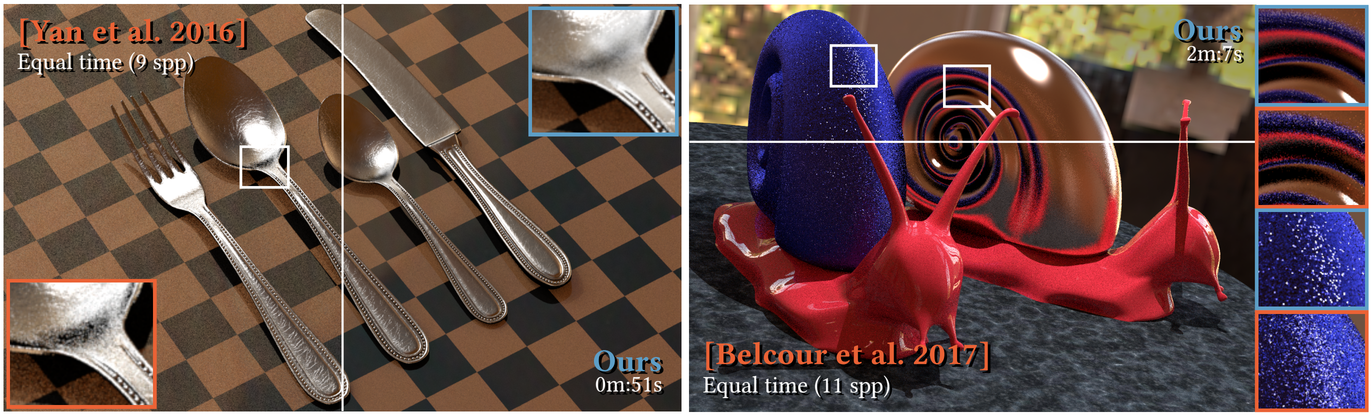
Scalable Appearance Filtering for Complex Lighting Effects

LUIS E. GAMBOA, Université de Montréal
JEAN-PHILIPPE GUERTIN, Université de Montréal
DEREK NOWROUZEZAHRAI, McGill University and Université de Montréal



(a) Direct illumination (b) Global illumination

Fig. 1. We filter direct (a) and global illumination (b) with high-frequency appearance and complex emitters ~**30**× faster than the state-of-the-art and ~**10**% the memory footprint. Unless otherwise stated, our results have converged and residual noise is due to high-frequency appearance.

Realistic rendering with materials that exhibit high-frequency spatial variation remains a challenge, as eliminating spatial and temporal aliasing requires prohibitively high sampling rates. Recent work has made the problem more tractable, however existing methods remain prohibitively expensive when using large environmental lights and/or (correctly filtered) global illumination. We present an appearance model with explicit high-frequency micro-normal variation, and a filtering approach that scales to multi-dimensional shading integrals. By combining a novel and compact half-vector histogram scheme with a directional basis expansion, we accurately compute the integral of filtered high-frequency reflectance over large lights with angularly varying emission. Our approach is scalable, rendering images indistinguishable from ground truth at over **10**× the speed of the state-of-the-art and with only **15**% the memory footprint. When filtering appearance with global illumination, we outperform the state-of-the-art by ~**30**×.

CCS Concepts: • **Computing methodologies** → **Rendering**;

Additional Key Words and Phrases: normal maps, glints, spherical harmonics, summed area tables

ACM Reference Format:
Luis E. Gamboa, Jean-Philippe Guertin, and Derek Nowrouzezahrai. 2018. Scalable Appearance Filtering for Complex Lighting Effects. *ACM Trans. Graph.* 37, 6, Article 277 (November 2018), 13 pages. https://doi.org/10.1145/3272127.3275058

1 INTRODUCTION

Microfacet reflectance models are a powerful tool for expressing the behavior of real-world appearance. Traditionally, these models relied on aggregate statistical formulations of the normal distribution function (NDF), however many real-world objects exhibit features at scales visible to a viewer. Meso- and macro-scale scratches, flakes and bumps all produce visually rich "*sparkle*" effects.

One challenge in simulating these effects lies in resolving aliasing in the *effective* reflectance inside a pixel's projected footprint (or a path vertex's footprint, for indirect bounces), requiring prohibitively large sampling rates. Previous approaches (e.g., [Belcour et al. 2017; Jakob et al. 2014; Yan et al. 2016]) resolve this issue by hierarchically culling normals and positions when *evaluating* an appropriately filtered effective reflectance towards a *fixed lighting direction*.

These methods are tailored to sharp directional or point lighting, where sparkle effects can be quite pronounced; however, scenes with larger area and environmental sources can also exhibit sparkly behavior; here, resolving final antialiased images additionally requires *integrating* the *evaluation* of these previous models over the domain of the extended light source (Figure 2). Even with their efficient importance sampling schemes this integration over extended lights becomes prohibitively expensive, and the problem is compounded if we seek to simulate additional bounces of global illumination.

We present a filtered appearance model that admits an efficient numerical integration of incident radiance over a shading footprint from, e.g., all-frequency environmental light sources. We allow an explicit specification of the underlying normal variation and present a simple, efficient *double filtering* algorithm that adapts to *both* the

Authors' addresses: Luis E. Gamboa, Université de Montréal, gamboagl@iro.umontreal.ca; Jean-Philippe Guertin, Université de Montréal, guertijp@iro.umontreal.ca; Derek Nowrouzezahrai, McGill University, Université de Montréal, derek@cim.mcgill.ca.

0730-0301/2018/11-ART277 $15.00
https://doi.org/10.1145/3272127.3275058

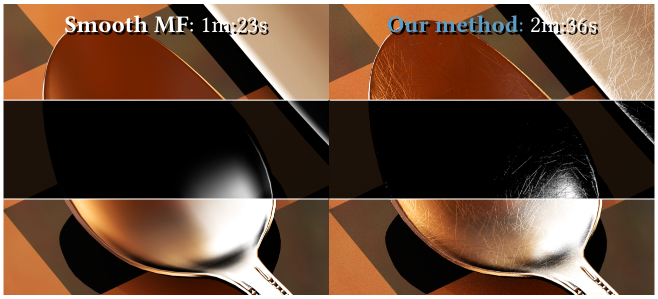


Fig. 2. While high-frequency sparkle-like appearance is highlighted by strong directional or point lighting (middle row), environmental lights (top row) still contribute significantly to sparkly appearance. Our method (right column) generates converged images with these effects in less than twice the time needed to generate converged results with a smooth microfacet model (left column), and ~10× faster than [Yan et al. 2016] (not shown).

frequency-content of the underlying lighting *and* the effective NDF within an arbitrary filter footprint. Our representation has modest memory needs, we easily incorporate it into standard offline and real-time rendering engines, and we demonstrate its ability to scale to scenes with complex lighting and global illumination.

Specifically, we present the following technical contributions:

- a novel spherical histogram to query scale-dependent NDFs in time independent of the normal map or footprint size,
- an efficient basis-space half-vector integrator that adapts to the frequency of *both* the incident radiance *and* multi-scale NDF, and
- applications to direct lighting and correctly filtered secondary bounces in global illumination, both with complex lighting.

We generate alias-free animations in a fraction of the time (~2–10%) of the state-of-the-art ([Belcour et al. 2017; Yan et al. 2016]).

2 PREVIOUS WORK

We aim to efficiently render alias-free ground truth-quality images of scenes with microfacet BRDFs, high-frequency normal variation, all-frequency lighting, both with and without global illumination. While no current method can efficiently handle these scenarios, we discuss the most relevant prior works: specifically, we draw upon work in both the interactive and offline rendering communities.

2.1 Microfacet models

A long history of work on appearance models for reflection from rough surfaces, using microfacet theory [Beckmann and Spizzichino 1963; Cook and Torrance 1982; Walter et al. 2007], lies at our foundation. As with most appearance filtering techniques, our model focusses on the representation and treatment of *continuous multi-scale* NDFs in the microfacet model, remaining agnostic to the choice for the remaining Fresnel and shadowing-masking terms in the model . We will outline the specific instance of these terms we employ in our model (Section 3), acknowledging the large body of work on both parametric (e.g., [Bagher et al. 2012; Dupuy et al. 2015; Ngan et al. 2005]) and non-parametric microfacet models (e.g., [Bagher et al. 2016]), we use the same framework but with a discrete set of normals rather than a continuous distribution.

2.2 Procedural texture antialiasing

Prior to most work on NDF-based appearance filtering, high-frequency and/or procedural textures were employed extensively as a means of introducing high-fidelity spatial variation in simple shading models [Perlin 2002]. Approaches for antialiasing these procedural textures include directly filtering out high spatial frequency content, or imposing simple geometric models atop the textures in order to facilitate more flexible filtering schemes [Cook and DeRose 2005; Heitz et al. 2013; Lagae et al. 2009]. While suitable for albedo filtering, these approaches cannot be applied to filtering multi-scale NDFs.

2.3 Accurate appearance filtering

Brute force numerical integration is a simple and prohibitively costly method to accurately render sparkle effects that arise from high-frequency normal map variation [Jakob et al. 2014; Yan et al. 2014]. This problem is compounded if, in addition to having to resolve spatial aliasing within a pixel footprint, variation from incident radiance need also be integrated numerically. Neither efficient (multiple) importance sampling schemes [Jakob et al. 2014; Veach and Guibas 1995; Yan et al. 2014] nor more efficient pruning strategies [Atanasov and Koylazov 2016; Yan et al. 2016] help, due to the nature of the full integrand, which includes a *product* of the filtered NDF with the incident lighting. Our filtered appearance model explicitly treats the fact that both a spatial *and* angular *integral* must be computed (Figure 4), as opposed to an *evaluation* of a spatial integral (Figure 3).

Most recently, several method approach the appearance filtering problem with solutions to efficiently prune only the normals in an NDF that will contribute non-negligibly to the final shading, for a given view and lighting direction pair [Atanasov and Koylazov 2016; Jakob et al. 2014; Yan et al. 2014, 2016]. These works provide significant improvements over brute force integration in the presence of strong point or directional lights, but they become prohibitively costly when large lights or global illumination are considered.

Two notable exceptions are the works of Raymond et al. [2016] and Belcour et al. [2017]. Raymond and colleagues present a multi-scale appearance model tailored to scratch microstructures, and they demonstrate its flexibility in the context of complex lighting. We instead treat arbitrary user- and procedurally-generated high-resolution normal variations, and explicitly compute both the spatial and angular integrations with a single method. Belcour et al. [2017] integrate an existing appearance filtering model ([Yan et al. 2014]) atop a covariance tracing-based global illumination framework in order to correctly filter indirect bounces off of, and on to, sparkly materials. This approach demonstrated orders of magnitude performance improvements over brute force Monte Carlo (at the time, this was the state-of-the-art for accurately simulating *indirect bounces* of sparkly materials). We similarly integrate our appearance model into a filter-aware global illumination algorithm, and demonstrate a 30× performance improvement over the approach of Belcour et al.

Recently, appearance models that treat wave optics for fine-scale microstructures have demonstrated the ability of simulating subtle iridescence effects [Werner et al. 2017; Yan et al. 2018]. We rely on geometric optics and instead focus on accelerating lighting integration for multi-scale appearance in direct and indirect illumination.

Incorporating wave optics in our model is an interesting, complementary direction of future work.

2.4 Fast appearance filtering

Initial work on normal antialiasing was led by the interactive graphics community. Here, smooth and compact normal distributions were used to leverage graphics MIP hardware to perform fast multi-scale filtering [Dupuy et al. 2013; Olano and Baker 2010; Toksvig 2005]. These single-lobe approximations of the NDF are suitable for interactive applications, but they cannot capture the details and anisotropies of NDFs (across scales) with high spatial resolution [Yan et al. 2014].

Han et al. [2007] notably use spherical harmonics (SH) and spherical Gaussian mixture models of the multi-scale NDF, allowing for multimodal NDFs. We too rely on SH, however only when performing our final shading (i.e., directional integration) and *not* when computing the form of the multi-scale NDF (i.e., spatial normal map filtering). Combined with an efficient half vector-space shading formulation (Section 4.2), this allows us to correctly integrate environmental lighting with both all-frequency underlying BRDFs *and* all-frequency environmental lighting, generating spatially- and temporally-antialiased image sequences.

Finally, real-time approximations of existing accurate appearance filtering approaches for glints [Zirr and Kaplanyan 2016] and iridescent scratches [Velinov et al. 2018] have shown promise towards pushing these important visual effects into an interactive context. The approach of Zirr and Kaplanyan [2016] is effective but limited to simple lighting, whereas Velinov et al. [2018] derive analytic approximations for spherical and polygonal area lights. Their model, however, only treats scratch-like microstructures, whereas we consider both arbitrary normal maps *and* arbitrary, i.e., environmental incident illumination. While we primarily target offline, fully-accurate simulations, we also discuss a proof-of-concept interactive renderer implementation in Section 6.

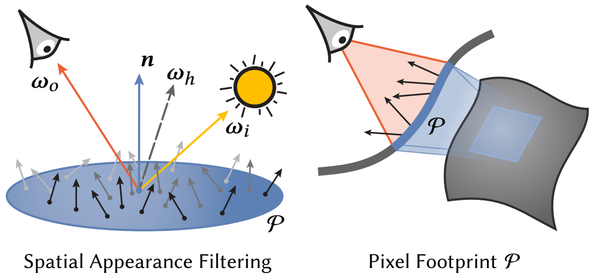
3 PRELIMINARIES & BASELINE APPEARANCE MODEL

As with recent works on filtered appearance we build atop a standard microfacet BRDF model of reflection from smooth surfaces

$$f_r(\boldsymbol{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \rho_d + \rho_s \left(\frac{D(\boldsymbol{x}, \boldsymbol{\omega}_h)\, F(\boldsymbol{\omega}_i \cdot \boldsymbol{\omega}_h)\, G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)}{4(\boldsymbol{n} \cdot \boldsymbol{\omega}_i)(\boldsymbol{n} \cdot \boldsymbol{\omega}_o)} \right), \quad (1)$$

but with a critical deviation that the *normal distribution function* (NDF) $D(\boldsymbol{x}, \boldsymbol{\omega}_h)$ is a scale-dependent, potentially high-frequency and anisotropic spherical distribution in the space of half-vectors $\boldsymbol{\omega}_h = (\omega_i+\omega_o)/|\omega_i+\omega_o|$. Here, $\boldsymbol{\omega}_i$ and $\boldsymbol{\omega}_o$ are the incident and outgoing lighting direction at the shading point $\boldsymbol{x}$ (with geometric normal $\boldsymbol{n}$), ρ_d and ρ_s are diffuse and specular reflection coefficients, $F(\boldsymbol{\omega}_i \cdot \boldsymbol{\omega}_h)$ is the Fresnel term and a geometry term $G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)$ captures micro-scale shadowing, masking and inter-reflection.

Techniques that target ground truth-quality renderings of microfacet models with high-frequency normal variation can be categorized by whether they allow **implicit** or **explicit** specification of the underlying normal variation. Explicit approaches (e.g., the "glints" methods of Yan et al. [2014; 2016]) are flexible in that they allow an arbitrary high-resolution normal map texture as input, but generally have large memory requirements (see Section 7). Implicit approaches (e.g., the "discrete microfacet" method of Jakob et al. [2014]) instead rely on statistical processes to describe the underlying high-frequency normal variation (typically using lazy, on-the-fly evaluation), leading to more compact run-time algorithms but at the cost of reduced control over appearance variations.



Spatial Appearance Filtering

Pixel Footprint $\mathcal{P}$

Fig. 3. Previous methods only integrate over a *spatial* footprint, before evaluating filtered appearance in the direction of a point/directional source.

We will efficiently compute filtered appearance integrals *in the presence of* complex incident lighting, and we propose our solution in Section 4. First, however, we establish a *baseline* filtered appearance model with which we can generate ground truth results and compare to in the context of prior work. As with our final model (Section 4), our baseline model is an explicit model and, so, allows for user-controllability through arbitrary normal map inputs. Our baseline (and final) models will require low memory footprints, in order to facilitate comparisons on complex scenes. In the context of prior work, our baseline model is a cross between *glint* [Yan et al. 2014, 2016] and *discrete microfacet* models [Jakob et al. 2014], and so we refer to it as the G×D model.

3.1 Baseline Filtered Microfacet Model (G×D)

The $D(\boldsymbol{x}, \boldsymbol{\omega}_h)$ term in Equation 1 can be expressed as a hemispherical probability distribution (over half-vectors) of the angle θ_h formed by every microfacet normal with the underlying macro-scale/geometric normal $\boldsymbol{n}$ at a shade point $\boldsymbol{x}$. Unlike smooth microfacet models, the NDF within a filtering footprint $\mathcal{P}$ (e.g., subtended by the projection of a pixel onto a shading surface; see Figure 3) can be modelled according to the individual normals that lie inside it.

We borrow and combine previous notation [Jakob et al. 2014; Yan et al. 2014] and define the normalized NDF as the set of normals $\boldsymbol{\omega}_x$ at shading locations $\boldsymbol{x}$ inside the footprint $\mathcal{P}$, each of which represents the mean direction of a spherical Gaussian G_s that models the roughness of an equivalently-smooth underlying microsurface (Figure 3),

$$\widehat{D}(\mathcal{P}, \boldsymbol{\omega}_h) = \int_{\mathcal{P}} G_s(\boldsymbol{\omega}_h; \boldsymbol{\omega}_x, \Sigma_s) d\boldsymbol{x}\,, \quad (2)$$

where the Gaussian roughness centered about each micro-normal is parameterized by its covariance matrix Σ_s. In practice, we apply isotropic roughness, and so a diagonal Σ_s with elements σ_s.

While Equation 2 assumes an arbitrary underlying spatial distribution of micronormals, in practice we use *discrete* micronormal distributions defined in normal map textures. That is, our normalized NDF is the finite set of *explicitly specified* normals $\boldsymbol{\omega}_x$ located at shading locations $\boldsymbol{x}$ inside the footprint $\mathcal{P}$. For this reason, it can be convenient to rewrite Equation 2 in terms of normal map texels

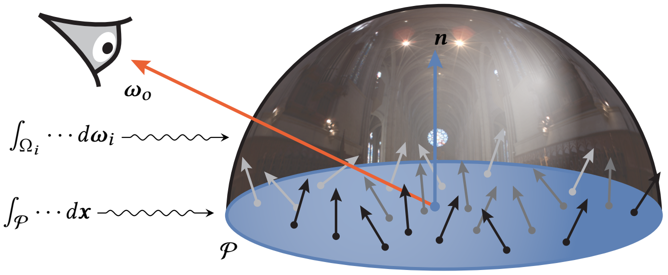

ω_o
n
$\int_{\Omega_i} \cdots d\omega_i$
$\int_{\mathcal{P}} \cdots d\boldsymbol{x}$
$\mathcal{P}$

Fig. 4. Our method integrates high-frequency appearance variation over both spatial footprint **and** incident lighting directions. To integrate over all light directions $\boldsymbol{\omega}_i$, as is needed for environment lighting, previous work requires many Monte Carlo samples, each of which involves an expensive integral over $\mathcal{P}$. Our method solves both integrals with a single evaluation.

(dropping the dependence on σ_s, for brevity), as

$$\widehat{D}(\mathcal{P}, \boldsymbol{\omega}_h) = \frac{1}{N_{\mathcal{P}}} \sum_{\boldsymbol{\omega}_x \in \mathcal{P}} G_s(\boldsymbol{\omega}_h; \boldsymbol{\omega}_x) \ , \tag{3}$$

where $N_{\mathcal{P}}$ are the number of texels that the footprint $\mathcal{P}$ projects onto in texture-space (Figure 3, right), and we assume equally weighted (i.e., box-filtered) averaging of these micronormals.

Note that as $\sigma_s \longrightarrow 0$ the roughness approaches mirror-like (delta) reflection and $G_s \longrightarrow \delta_\Omega$, our baseline approaches the model of Jakob et al. [2014], but with the important difference that micronormals are defined *explicitly.*

Incorporating the multi-scale NDF (i.e. for arbitrary pixel footprint size) described in Equation 2 into Equation 1, involves changing the notion of a point-wise BRDF f_r (dependent on intersection point $\boldsymbol{x}$) into a "multi-scale" BRDF $\widehat{f_r}$ that depends on pixel footprint $\mathcal{P}$. Having done this, we can evaluate for a *single* view and light direction, arriving at the formal definition of our baseline G×D filtered appearance model:

$$\begin{aligned}\widehat{f_r}(\mathcal{P}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) &= \rho_s \int_{\mathcal{P}} \frac{D(\boldsymbol{x}, \boldsymbol{\omega}_h) F(\boldsymbol{\omega}_i \cdot \boldsymbol{\omega}_h) G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)}{4(\boldsymbol{n} \cdot \boldsymbol{\omega}_i)(\boldsymbol{n} \cdot \boldsymbol{\omega}_o)} d\boldsymbol{x} \\ &= \rho_s \frac{\overbrace{\int_{\mathcal{P}} G_s(\boldsymbol{\omega}_h; \boldsymbol{\omega}_x) d\boldsymbol{x}}^{\widehat{D}(\mathcal{P}, \boldsymbol{\omega}_h)} \ F(\boldsymbol{\omega}_i \cdot \boldsymbol{\omega}_h) G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)}{4(\boldsymbol{n} \cdot \boldsymbol{\omega}_i)(\boldsymbol{n} \cdot \boldsymbol{\omega}_o)} \ ,\end{aligned} \tag{4}$$

where we solve the spatial integral by means of Equation 3 and we employ a factorization of the G term as a product of rational approximations of the Smith shadowing term, $G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \approx G_1(\theta_i) \, G_1(\theta_o)$ with $G(\theta)$ [Walter et al. 2007], and we use the exact Fresnel term for conductors.

Note that shading with Equation 4 for lighting configurations that require anything other than a **simple evaluation** of (single) $\boldsymbol{\omega}_i$, and hence $\boldsymbol{\omega}_h$, directions would have to rely on brute force numerical integration. This quickly becomes prohibitively expensive (Figures 4 and 5). In Section 4 we present a mathematical representation, and detail an efficient and scalable algorithm, for solving integrals of Equation 4 against (non-delta) lighting distributions.

Our method inherits the following assumptions from [Yan et al. 2014, 2016] and [Jakob et al. 2014]: locally flat macro-geometry and no view/light variation within the footprint. Our results are consistent with prior work, as can be observed even under highly curved surfaces, e.g. the snail (Figure 1).

3.2 Evaluating G×D

We adapt the pruning algorithm of Jakob et al. [2014] to treat explicit normal maps and Gaussian roughness, when evaluating Equation 4 for ground truth baseline renderings. Specifically, we build a min-max MIP-hierarchy in texture space. Spatial footprint pruning is trivial, as we project a conservative bounding box for pixel (or indirect path vertex) footprints onto the UV space. Normals are encoded as 2D (s, t)-coordinates on a projected disc parameterization [Yan et al. 2014] and, at each MIP-level, a min and max (s, t) are computed and used to cull $\boldsymbol{\omega}_x$ directions that fall outside of $\mathcal{P}$ and Ω_o. The hierarchy is constructed once at start-up at a cost negligible to the total render time.

Of note, this G×D approach requires less memory and is consistently faster than the earliest approach of Yan et al. [2014], however Yan et al.'s latter approach [2016] converges between 5 – 25× faster (albeit requiring typically 100× more memory with higher per-evaluation cost) than G×D. For this reason, we often treat G×D as our "brute force" solution for ground truth image generation (even though it is roughly 500× faster than naïve importance sampled Monte Carlo), and we treat the latest Yan et al. technique [2016] as our "high-performance" benchmark. All results generated with Yan et al.'s [2016] techniques rely on implementations provided to us by the authors.

Figure 5 demonstrates the performance gap introduced when even the state-of-the-art [Yan et al. 2016] has to rely on numerical integration to resolve variation in incident radiance. This gap is compounded by global illumination (e.g., Figures 1 and 14).

In Section 4, we will incorporate incident lighting variation as a component of a new filtered appearance model and detail an approach for computing integrals that filter over the spatial, outgoing view *and* incident lighting dimensions. Our method relies on a simple multi-scale spherical histogram (Section 4.1) and basis-space integration scheme (Section 4.2), it is easy to implement in existing rendering engines (Section 6), and it scales favorably in performance and memory (outperforming the state-of-the-art by 10 – 30×).

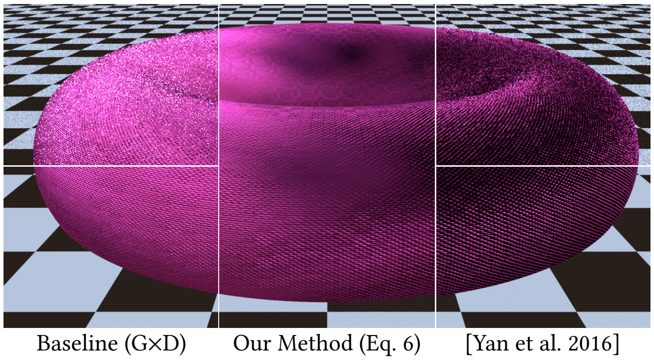


Fig. 5. **Middle:** Our method generates a converged filtered result in 4.89s (1spp) in a scene with a high-resolution normal map and enviromental lighting. **Top halves:** equal time for our baseline (2spp, left) and [Yan et al. 2016] (1spp, right). **Bottom halves**: Equal quality rendering requires 16,384spp and 13m:8s with our baseline and 512spp and 2m:25s with [Yan et al. 2016].

In Section 5 we will apply our model to filtering direct illumination and multi-bounce global illumination transport, both in scenes with complex environmental (and point) lighting.

4 FILTERING APPEARANCE IN SPACE AND DIRECTION

To synthesize spatially- and temporally-antialiased image sequences with sparkly materials, complex lighting and global illumination, we devise a compact and efficient approach to compute integrals of Equation 4 over arbitrary illumination signals $L_{\mathrm{i}}(\boldsymbol{\omega})$, namely

$$\begin{aligned} L_{\mathrm{o}}(\mathcal{P},\boldsymbol{\omega}_o) &= \int_{\Omega_i} L_{\mathrm{i}}(\boldsymbol{x},\boldsymbol{\omega}_i)\,\widehat{f_r}(\mathcal{P},\boldsymbol{\omega}_i,\boldsymbol{\omega}_o)(\boldsymbol{n}\cdot\boldsymbol{\omega}_i)d\boldsymbol{\omega}_i \\ &= \int_{\Omega_i}\int_{\mathcal{P}} \frac{L_{\mathrm{i}}(\boldsymbol{x},\boldsymbol{\omega}_i)\,D(\boldsymbol{x},\boldsymbol{\omega}_h)\,F(\boldsymbol{\omega}_i\cdot\boldsymbol{\omega}_h)\,G(\boldsymbol{\omega}_i,\boldsymbol{\omega}_o)}{4(\boldsymbol{n}\cdot\boldsymbol{\omega}_o)}\,d\boldsymbol{x}\,d\boldsymbol{\omega}_i\,, \qquad (5)\end{aligned}$$

where Ω_i are hemispherical directions about the macro-scale normal $\boldsymbol{n}$ and we drop the specular coefficient ρ_s for brevity. In this context, $D(\boldsymbol{x},\boldsymbol{\omega}_h)$ is the point-wise NDF $G_s(\boldsymbol{\omega}_h;\boldsymbol{\omega}_x)$. State-of-the-art techniques would have to rely on Monte Carlo integration to resolve Equation 5 and, even with efficient (spatially-filtered) BRDF importance sampling [Jakob et al. 2014; Yan et al. 2014, 2016], light importance sampling and MIS [Veach and Guibas 1995], this is prohibitively costly (Figure 5).

We simplify Equation 5 by factoring and approximating the Fresnel, geometric terms and microfacet normalization terms from the integral with a transfer function $\overline{FG}(\boldsymbol{\omega}_o)$ (see Appendix A for details), leading to our *lighting-aware* filtered appearance model,

$$\boxed{L_{\mathrm{o}}(\mathcal{P},\boldsymbol{\omega}_o) \approx \overline{FG}(\boldsymbol{\omega}_o)\int_{\Omega_i}\int_{\mathcal{P}} L_{\mathrm{i}}(\boldsymbol{x},\boldsymbol{\omega}_i)D(\boldsymbol{x},\boldsymbol{\omega}_h)\,d\boldsymbol{x}\,d\boldsymbol{\omega}_i}\,. \qquad (6)$$

An immediate issue arises when addressing solutions to Equation 6: the discrepancy in the natural spherical parameterizations of L_{i} and D. We choose to perform integration in the space of half-vectors $\boldsymbol{\omega}_h$ and reparameterize Equation 6 appropriately by introducing an incident illumination function $\widehat{L_{\mathrm{i}}}$ that is dependent on both outgoing and half-vector directions, reducing the problem to solving equations of the form

$$\begin{aligned} &\int_{\Omega_i}\int_{\mathcal{P}} L_{\mathrm{i}}(\boldsymbol{x},\boldsymbol{\omega}_i)\,D(\boldsymbol{x},\boldsymbol{\omega}_h)\,d\boldsymbol{x}\,d\boldsymbol{\omega}_i \\ &= \int_{\Omega_h} L_{\mathrm{i}}(\bar{\boldsymbol{x}},r(\boldsymbol{\omega}_o,\boldsymbol{\omega}_h))\,(4(\boldsymbol{\omega}_h\cdot\boldsymbol{\omega}_o))\underbrace{\int_{\mathcal{P}} D(\boldsymbol{x},\boldsymbol{\omega}_h)\,d\boldsymbol{x}}_{\widehat{D}(\mathcal{P},\boldsymbol{\omega}_h)}\;d\boldsymbol{\omega}_h \\ &= \int_{\Omega_h} \widehat{L_{\mathrm{i}}}(\bar{\boldsymbol{x}},\boldsymbol{\omega}_o,\boldsymbol{\omega}_h)\,(4(\boldsymbol{\omega}_h\cdot\boldsymbol{\omega}_o))\int_{\mathcal{P}} G_s(\boldsymbol{\omega}_h;\boldsymbol{\omega}_x)\,d\boldsymbol{x}\,d\boldsymbol{\omega}_h\,, \qquad (7)\end{aligned}$$

where the $4(\boldsymbol{\omega}_h\cdot\boldsymbol{\omega}_o)$ factor accounts for the change of parameterization between differential incident and half-vectors [Torrance and Sparrow 1967; Walter 2005], Ω_h are the hemispherical directions about $\boldsymbol{\omega}_o$, $r(\boldsymbol{\omega}_o,\boldsymbol{\omega}_h) = \boldsymbol{\omega}_i$ reflects $\boldsymbol{\omega}_o$ w.r.t. $\boldsymbol{\omega}_h$, and we define our *filtered multi-scale NDF*, $\widehat{D}(\mathcal{P},\boldsymbol{\omega}_h)$, which depends on the filtering footprint $\mathcal{P}$. Note that above, as with previous work, we assume the incident illumination $\widehat{L_{\mathrm{i}}}$ does not vary significantly within the spatial footprint and use a "central" shading point $\bar{\boldsymbol{x}}$ to distinguish between its spatial variation at scales larger than a single filtering footprint. Sections 5 and 6 will detail our representation of the incident illumination $\widehat{L_{\mathrm{i}}}$, which allows us to handle physically-based microfacet models in the presence of environmental lighting, addressing certain limitations that we discuss regarding previous work (e.g., [Han et al. 2007]; see Section 6.3). For the purposes of our exposition here, however, it is safe to assume that we have arrived at an expression $\widehat{L_{\mathrm{i}}}$ that can be treated as a simple (hemi-)spherical function of $\boldsymbol{\omega}_h$.

Now, to efficiently solve integrals of the form in Equation 7 (and, so, Equation 6), we combine two ideas that lead to orders of magnitude faster performance than the state-of-the-art with a modest memory requirement: first, a representation for computing multi-scale spherical NDFs over arbitrary filtering footprints in constant time, using *spherical histograms*; and, second, a basis-space representation for computing efficient spherical integrals that can *adapt* to the frequency-content of *both* the incident lighting $\widehat{L_{\mathrm{i}}}$ and the multi-scale NDF D inside a footprint.

4.1 Multi-scale NDFs using Histogram Accumulation

For explicit high-frequency normal variation represented in, i.e., a normal map, the multi-scale NDF $\widehat{D}(\mathcal{P},\boldsymbol{\omega}_h)$ is a normalized spherical distribution of all the normals that lie inside the projected filtering footprint $\mathcal{P}$, convolved with the isotropic roughness kernel G_s. One can naïvely arrive at a discretized (e.g., a spherical texture map) approximation of $\widehat{D}(\mathcal{P},\boldsymbol{\omega}_h)$ by convolving and summing each normal map texel that lies inside the projection of $\mathcal{P}$ onto $(u,v,)$-space. This naïve approach is not favorable for two key reasons:

(1) the cost of computing $\widehat{D}$ would scale linearly in the size of $\mathcal{P}$, and
(2) the cost of spherical Gaussian convolution would scale with the underlying discretization resolution for $\widehat{D}$.

Motivated by these scalability issues, we propose a novel representation and an efficient data-structure, based on *spherical histogram accumulation*, to compute a discrete representation of $\widehat{D}$ in **constant time** for footprints $\mathcal{P}$ of arbitrary size (item #1, above). We detail our histogram formulation below, before discussing how basis-space representation can help us avoid the scalability cost of spherical Gaussian convolution (item #2, above; see Section 4.2).

We observe that, instead of individually representing every possible discrete micro-normal direction, we can collect histogram statistics of the normals that fall within a footprint. This has two benefits: first, we show below how histogram statistics for an *arbitrary footprint* can be computed using histogram statistics collected for each *individual* micro-normal in the normal map, and this all in constant time independent of the footprint size; second, we can control the NDF accuracy by adjusting the histogram bin resolution.

4.1.1 Histogram initialization. Specifically, we define a new 2D histogram texture with the same resolution as the normal map. Each texel stores an *accumulated spherical histogram of normals*, encoded as a 2D bin of (θ,ϕ)-space normal buckets. Bins have equal elevation and azimuthal angle extents. Given the SAT-like structure, a value at position (u',v') corresponds to the histogram of normals in the rectangular area (of the normal map) with corners $(0,0)$, $(0,v')$, $(u',0)$ and (u',v')

At initialization, beginning from the top-left histogram texel, we accumulate normals into bins from left-to-right and top-to-bottom (see Figure 6): each histogram texel accumulates additional normals

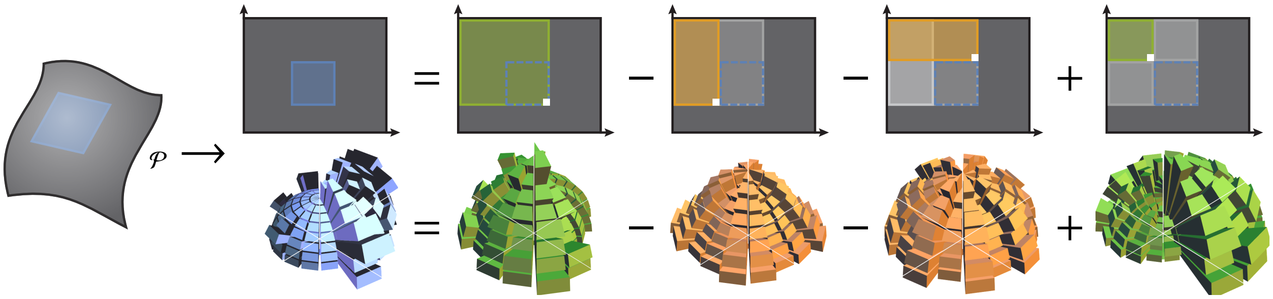


Fig. 6. We use texture-space accumulated NDF histograms to efficiently query the multi-scale NDF histogram for an *arbitrary filtering footprint* $\mathcal{P}$ using four lookups into this SAT-like data structure (histogram radial sizes not to scale, above). Figure 11 illustrates the visual impact of different histogram resolutions.

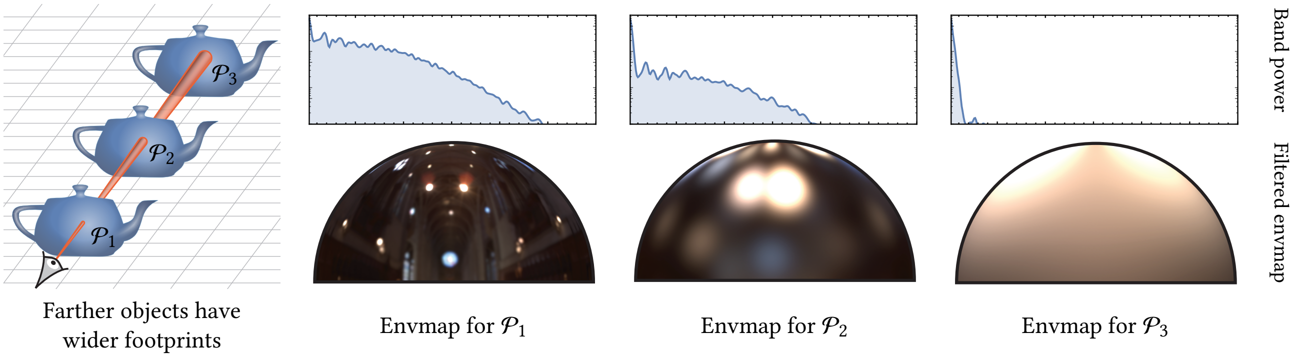

$\mathcal{P}_3$
$\mathcal{P}_2$
$\mathcal{P}_1$
Farther objects have wider footprints
Envmap for $\mathcal{P}_1$
Envmap for $\mathcal{P}_2$
Envmap for $\mathcal{P}_3$
Band power
Filtered envmap

Fig. 7. We adapt the shading to the minimum between the max frequency bandwidth of the multi-scale NDF in a footprint (top row) and the max bandwidth of the lighting environment. We visualize enviroment maps bandlimited to this *effective shading bandwidth* (bottom).

from the normal map to cover the same area up to the current (u, v)-coordinate. Normals only get added (not removed) to histogram bins as we traverse across texels, and so a histogram at texel (u', v') is a strict superset of the histograms at texel, e.g., $(u' - 1, v' - 1)$. In this case, the histogram at (u', v') is equal to the sum of histogram entries at $(u', v' - 1)$ and $(u' - 1, v')$ (correcting for overlaps) plus the corresponding frequency increase for the normal at (u', v') in the normal map. This structure allows for a fast sweeping accumulation, which adds a negligible overhead at startup (see Section 6.1).

4.1.2 *Summed area table histogram queries.*

At run-time we compute the NDF histogram of an *arbitrary* axis-aligned (u, v) footprint by treating the histogram as a summed area table, using only four constant cost queries (Figure 6). As with Yan et al. [Yan et al. 2014, 2016], we utilize axis-aligned (u, v) footprints and a Gaussian image reconstruction kernel to approximate Gaussian footprint weighing.

Given the ability to compute multi-scale NDF histograms (efficiently, in constant time w.r.t. footprint size), we require an approach to integrate the *product* of these spherical histograms with arbitrary incident illumination distributions (i.e., Equations 6 and 7).

4.2 Adaptive Basis-space Integration

A naïve integration solution is to apply a spherical quadrature at central axes of each spherical histogram bin, effectively a bin as a weighted delta function. Alternatively, a quadrature with pre-filtered environmental summed area tables can be used, treating each bin as a subtended spherical rectangle. These integration approaches scale linearly with the histogram resolution, whereas we propose an *adaptive* method with several advantages:

- we adapt to the *frequency content* of the entire integrand in Equation 7, treating the bandlimits of *both* the NDF *and* the incident illumination and shading at the minimum of these bandlimits,
- our shading scales *indepedent* of the histogram resolution, and
- we do not need any specialized spherical SAT parameterizations, which would incur a resampling cost at shade-time.

To apply a basis-space shading method to our problem, we require a basis capable of efficiently:

(1) determining the bandlimit of lighting, NDF and their product,
(2) computing bandlimited integrals of this product (i.e., adaptivity) ,
(3) capturing all-frequency signal variation (i.e., accuracy), and
(4) having only modest memory requirements (i.e., compactness),

To these ends, we choose to employ spherical harmonics (SH). As with previous work [Han et al. 2007], we detail several properties of SH that we leverage to meet our performance, accuracy and memory requirements; we contrast our choice and overall representations to that of prior work below and in Section 6.3. Basis selection involves trade-offs and, while other choices may have met our requirements (e.g., spherical radial basis functions), our SH method demonstrates significant performance and memory improvements over the state-of-the-art, and so we leave the application/exploration of other bases to this problem to future work.

Before detailing the interplay of our SH integration scheme with our spherical histogram representation, we provide a quick primer to SH and the specific properties we leverage.

4.2.1 *Spherical harmonic preliminaries.*

We can represent a spherical function $f(\boldsymbol{\omega})$, with $\boldsymbol{\omega} = (x, y, z) = (\theta, \phi) \in S^2$, with SH projection coefficients obtained by projecting f onto the real SH basis as $\boldsymbol{f} = \int_{S^2} f(\boldsymbol{\omega})\, \boldsymbol{y}(\boldsymbol{\omega}) d\boldsymbol{\omega}$, where $\boldsymbol{f}$ is a vector of these coefficients and

$\boldsymbol{y}(\boldsymbol{\omega})$ is a vector of the individual SH basis functions:

$$y_l^m(\theta,\phi) = \begin{cases} \sqrt{2}\,K_l^m\ \cos(m\phi)\,P_l^m(\cos\theta) & ,\ m > 0 \\ \sqrt{2}\,K_l^m\ \sin(|m|\phi)\,P_l^{|m|}(\cos\theta), & m \le 0 \end{cases}, \tag{8}$$

where an "order N" SH representation comprises bands $0 \le l \le N-1$ m, indexes the $(2l+1)$ basis functions in each band l, K_l^m is a normalization term, P_l^m are the associated Legendre polynomials, and each band-l basis function is a degree l polynomials in (x,y,z). We often use a single index $i = l(l+1)+m$ for basis functions and coefficients.

The signal f has bandlimit M if $f_l^m = 0,\ \forall l \ge M-1$. We treat the concept of an *effective bandlimit* M when $|f_l^m| \le \epsilon,\ \forall l \ge M-1$ for sufficiently small ϵ (we use $\epsilon = 10^{-6}$). We can reconstruct f by weighting the SH basis functions by its SH projection coefficients

$$f(\boldsymbol{\omega}) = \boldsymbol{f}\cdot\boldsymbol{y}(\boldsymbol{\omega}) = \textstyle\sum_{l=0}^{N-1}\sum_{m=-l}^{m=l} f_l^m\ y_l^m(\boldsymbol{\omega}) \tag{9}$$

and, unless f has bandlimit $M < N$, the reconstruction is a bandlimiting approximation of f (to a bandlimit of N.)

The $m = 0$ subset of SH functions, called *zonal harmonics* (ZH), are circularly symmetric functions of $cos\theta = z$. Sloan et al. [2005] introduced a fast rotation to compute the SH coefficients g_l^m of a circularly symmetric function aligned about z (represented by ZH coefficients f_l) rotated to an *arbitrary* direction $\bar{\boldsymbol{\omega}}$, by simply scaling the SH basis functions evaluated at $\bar{\boldsymbol{\omega}}$ as

$$g_l^m = n_l^*\,f_l\,y_l^m(\bar{\boldsymbol{\omega}}) = f_l^*\,y_l^m(\bar{\boldsymbol{\omega}})\,, \tag{10}$$

with $n_l^* = \sqrt{4\pi/(2l+1)}$. This amounts to an application of the Funke-Hecke convolution theorem that allows us to obtain the SH coefficients of the function that results from a convolution of a circularly-symmetric function (with ZH coefficients f_l) and a spherical function (with SH coefficients h_l^m). Above, this amounts to setting $g(\boldsymbol{\omega}) = f(\theta) \circledast h(\boldsymbol{\omega})$, where $h(\boldsymbol{\omega}) = \delta(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})$ is simply a delta function in the direction of the rotation axis and h has SH projection coefficients $h_l^m = y_l^m(\bar{\boldsymbol{\omega}})$.

The final SH property of interest is the *fast double-product integral* formulation: given two spherical functions $a(\boldsymbol{\omega})$ and $b(\boldsymbol{\omega})$ with (effective) band-limits M_a and M_b, the integral of their product is

$$\textstyle\int_{\mathcal{S}^2} a(\boldsymbol{\omega})b(\boldsymbol{\omega})d\boldsymbol{\omega} = \sum_{i=0}^{M_\star^2-1} a_i\,b_i\,, \tag{11}$$

where $M_\star = \min(M_a, M_b)$ and we arrive at the right hand side by substituting the SH expansions of a and b (Equation 9) into the left hand side and then applying the orthonormality property of SH basis functions: $\int_{\mathcal{S}^2} y_i(\boldsymbol{\omega})y_k(\boldsymbol{\omega})d\boldsymbol{\omega} = \sigma_{i,k}$, where $\sigma_{i,k}$ is the Kroenecker delta.

Equations 9 and 11 satisfy requirements #1 and #2 for our basis (listed earlier in this subsection), and we address the final two basis requirements. Regarding all-frequency signal variation (item #3), while they are not often leveraged for all-frequency shading applications, SH basis functions are *capable* of capturing all-frequency signal variation if a sufficiently high bandlimit is chosen. We will discuss below exactly the manner in which we apply SH to our problem, as it scales independently to the normal/histogram map resolution and allows us to use extremely high bandlimits (order 300 in some cases), which are more than sufficient to capture the frequency content of NDFs (across scales) and realistic incident illumination from, e.g., environment maps (see Figure 7). While the number of SH coefficients we need to store grows quadratically in the max bandlimit order, we still maintain a memory footprint significantly smaller than the state-of-the-art [Yan et al. 2016], satisfying item #4, above (see Section 6.1).

4.2.2 Multi-scale SH NDFs and adaptive integration. We compute the SH projection of our filtered multi-scale NDF (Equation 7) using the multi-scale NDF histogram we query for our filtering footprint $\mathcal{P}$ and, after determining the effective bandlimit of this projection (and of the incident lighting; see Section 6), we set the shading bandlimit adaptively per-pixel as $M_\star$ and apply Equation 11 to solve the integral (Equation 7) of Equation 6.

Concretely, we arrive at the SH coefficients of the filtered multi-scale NDF $D_i(\mathcal{P}) = \int_{\mathcal{S}^2} \widehat{D}(\mathcal{P}, \boldsymbol{\omega})\,y_i(\boldsymbol{\omega})\,d\boldsymbol{\omega}$ by summing the SH projection coefficients of its individual Gaussian roughness kernels, each aligned about the $N_{\mathcal{P}}$ normals in the filtering footprint $\mathcal{P}$,

$$D_i(\mathcal{P}) = \frac{1}{N_{\mathcal{P}}}\sum_{\boldsymbol{\omega}_x\in\mathcal{P}}\int_{\mathcal{S}^2} G_s(\boldsymbol{\omega};\boldsymbol{\omega}_x)\,y_i(\boldsymbol{\omega})\ d\boldsymbol{\omega}\,. \tag{12}$$

We query our NDF histogram for $\mathcal{P}$ (Figure 6) and iterate over the $N_{\mathcal{P}}$ micronormals in the histogram (outer sum in Equation 12). SH projection coefficients for each of these individual Gaussians are precomputed once at initialization, as they align with the histogram bin central axes. In fact, the histogram nature of our intermediate NDF representation allows us to further reduce the number of elements from $N_{\mathcal{P}}$ to M (with $M \ll N_{\mathcal{P}}$ as $\mathcal{P}$ increases in size), where M is simply the resolution of the histogram.

Recalling that spatial integration of an individual Gaussian roughness kernel projects that Gaussian onto the sphere about the shading frame, so we take advantage of the fact that these spherical Gaussians are circularly symmetric and that their SH coefficients can be computed by applying the fast ZH rotation formulation (Equation 10) to the ZH coefficients of a canonically-oriented spherical Gaussian (which we compute and store once, numerically). Computing the SH coefficients of $\widehat{D}(\mathcal{P}, \boldsymbol{\omega})$ amount to a weighted sum of M ZH-rotated (precomputed) spherical Gaussian SH coefficients.

One could imagine avoiding the histogram abstraction and directly storing SH coefficients for incrementally accumulated NDFs, in a similar SAT-like structure as our histogram, but this approach has several limitations that we discuss in more detail in Section 6: of note, the storage costs of such an SAT would scale (quadratically) with the maximum SH order (e.g., **351 GB** for order-300 and a 1024 × 1024 normal map). This alternative also bears some similarities to the multi-scale NDF representation in Han et al.'s work [2007], which we also discuss in Section 6.3.

5 APPLICATIONS

After computing the SH coefficients of the filtered NDF $\widehat{D}$ we can solve Equation 6 with an efficient SH double product integral (Equation 11). We do so in the context of two applications, detailing their specifics, below: filtered direct illumination (e.g., [Jakob et al. 2014; Yan et al. 2014, 2016]) and filtered global illumination [Belcour et al. 2017]. In all cases, our rendering algorithms use an order-of-magnitude less time and less memory to match converged ground truth compared to the state-of-the-art.

5.1 Appearance Filtered Direct illumination

We compute filtered direct illumination $L_d(\mathcal{P}, \boldsymbol{\omega}_o)$ in a scene with complex environment and area lighting with Equations 6 and 7 and an appropriate substitution of $\widehat{L}_i(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h) \equiv \widehat{L}_e(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h)\widehat{V}(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h)$ that accounts for the direct shadowing visibility term $V(\boldsymbol{x}, \boldsymbol{\omega}_i)$ and the incident radiance due to emission from light sources $L_e(\boldsymbol{x}, \boldsymbol{\omega}_i)$. Note that L_e combines lighting from area (e.g., polygonal), environmental and point/directional light sources.

Here, we note that we can *immediately* obtain a converged *unshadowed* direct illumination result $L_u(\mathcal{P}, \boldsymbol{\omega}_o)$ with a single shading "sample" (i.e., a single evaluation of our basis-space double-product integral), if we ignore the visibility term and use $\widehat{L}_i(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h) \equiv \widehat{L}_e(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h)$. Here, we precompute the SH projection of $\widehat{L}_e(\boldsymbol{x}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h)(4\boldsymbol{\omega}_h \cdot \boldsymbol{\omega}_o)$ for many $\boldsymbol{\omega}_o$ at initialization, and query these coefficients during fast double-product integration for the unshadowed term L_u. We discuss technical details in Section 6.

A common numerical solution in scenarios such as this is to devise a Monte Carlo estimator that uses the unshadowed direct illumination as a *control variate*, and then relies on numerical sampling to resolve the shadowing:

$$L_d(\mathcal{P}, \boldsymbol{\omega}_o) = L_u(\mathcal{P}, \boldsymbol{\omega}_o) - \sum_{s=0}^{N_s} \frac{L_e(\boldsymbol{x}, \boldsymbol{\omega}_s)\,(1 - V(\boldsymbol{x}, \boldsymbol{\omega}_s))\widehat{f_r}(\mathcal{P}, \boldsymbol{\omega}_s, \boldsymbol{\omega}_o)}{N_s\ \mathrm{pdf}(\boldsymbol{x}, \boldsymbol{\omega}_s)} \tag{13}$$

where we rely on our baseline G×D technique to resolve the Monte Carlo integral, with a spherical pdf that applies multiple importance sampling of the light/environment and our filtered BRDF sampling. Spatial sampling (i.e., to resolve the implicit spatial integral over the footprint of $\widehat{f_r}$) is absorbed into standard sub-pixel anti-aliasing.

One side effect of this control variate estimator is that noise in the shadowed regions can become objectionable, since individual estimator samples are unbounded. We rely instead on the recent *ratio estimator* approach of Heitz et al. [2018], that leverages a factored *multiplicative* decomposition of the unshadowed and shadowed direct radiance, instead of the difference-based estimator above. Unlike their approach, we do not employ any spatial denoising to our ratio estimate, to avoid adding bias; doing so would significantly improve the visual appearance of our results at low sampling rates, but our goal is to match ground truth results.

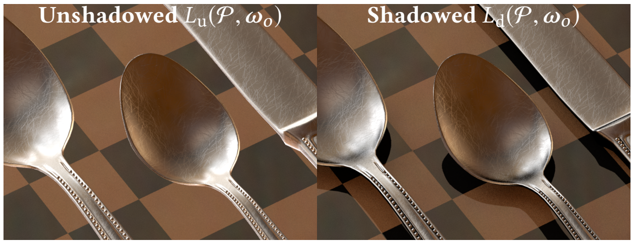

Unshadowed $L_u(\mathcal{P}, \omega_o)$
Shadowed $L_d(\mathcal{P}, \omega_o)$

Fig. 8. We can generate converged unshadowed direct illumination with a single shading sample (left; 0m:42s) and then apply the technique of Heitz et al. [2018] to compute shadowing numerically (with 4spp in 1m:08s).

5.2 Filtered Global Illumination

We adapt the method of Belcour et al. [2017] to filter high-frequency appearance in the presence of global illumination effects, but now also in scenes with complex environmental and area light sources. Their approach propagates a lightweight covariance-based lightfield representation along paths (i.e., in a standard uni- or bi-directional path tracing framework). The covariance at any path vertex can be used to define a local filtering footprint which, in turn can be applied to filter an underlying high-frequency appearance model.

Even in scenes with strong point and directional sources, their indirect filtered appearance method provided several orders-of-magnitude improvements over a naïve path traced solution (the only alternative, at the time). The added complexity of filtering indirect reflections of high-frequency appearance *in the presence of* large angularly-varying light sources stresses their method as, just with filtered appearance in direct illumination, their underlying appearance filtering model does not take variation due to incident illumination into account.

By applying our solution to Equation 6 in the next-event estimator of secondary path vertices in a standard uni-directional path tracer, augmented with Belcour et al.'s secondary footprint computation, we are effectively leveraging our method's ability to compute a *converged* value that integrates over all light directions (instead of just one). This way, we are able to outperform their approach in scenes with complex lighting by about a factor of **10 – 50**× (see Figures 1 and 14).

6 IMPLEMENTATION AND DISCUSSION

We discuss implementation details, as well as discussing design decisions in the context of previous work, below.

At a high-level, our approach can build atop any rendering engine capable of generating ray differentials [Igehy 1999] (for direct illumination) and, for global illumination, secondary path filtering differentials (we employ Belcour et al.'s [2017] approach, here).

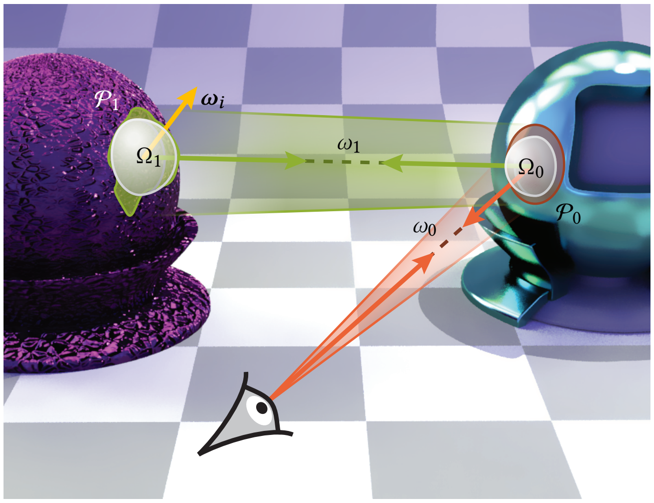
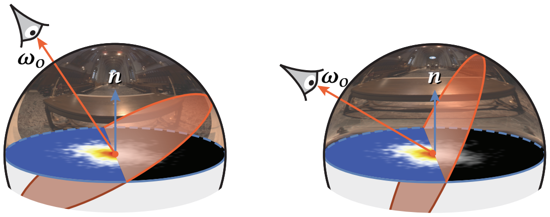


Fig. 9. We adapt the technique of Belcour et al. [2017] to compute secondary path vertex footprints for filtered appearance with global illumination.



(a) θ_o = 30°

(b) θ_o = 60°

Fig. 10. When integrating with respect to ω_h our basis-space approach automatically discards NDF normals outside the shading frame (grayscale) and uses an appropriately warped incident illumination distribution.

As with previous work [Yan et al. 2014, 2016], we assume axis-aligned footprints in texture-space, computed with conservative (u, v) bounding boxes.

Below, we provide technical implementation details of our data-structures and algorithm, and discuss relationships (and differences) to certain aspects of previous work. Our rendering framework involves the construction of our histogram SAT (Section 6.1), the precomputation of rotated Gaussian SH lobe coefficients (Section 6.2) and the precomputation of weighted SH lighting coefficients (Section 6.4).

6.1 Histogram Resolution

We store a discrete spherical histogram in our SAT-like multi-scale NDF lookup texture (Section 4.1). The texture itself has the same resolution as the underlying normal map, and each texel stores a discretized histogram with (θ, ϕ)-parameterized bins. We experimented with different histogram resolutions, and settled on a resolution of 9 × 32 for every result in the paper: this value was chosen such that renderings visually match the reference ground truth, all while maintaining a modest memory footprint (e.g., 576MB for 1024×1024 normal maps and 1.1GB for 2048 × 2048). Figures 11 and 15 illustrates visual artifacts that arise from reducing the histogram bin resolution.

6.2 Gaussian Roughness Lobe Discretization

We precompute SH coefficients for the rotated ZH Gaussian roughness lobes ($\int G_s(\omega; \omega_x) y_i(\omega) d\omega$ in Equation 12), for a fixed base roughness σ_s (and each high-frequency material in the scene), at start-up. We store these rotated lobes at a 4× *higher resolution* (i.e., 65 × 128) than the histogram due to the difference in their spherical parameterizations: the NDF histograms are expressed in the *local-shading frame* (with $\boldsymbol{n}$ up), whereas shading with the Gaussian lobes is conducted in the *half-vector frame* (with $\boldsymbol{\omega_o}$ up). This means that we need to rotate the $\boldsymbol{n}$-oriented Gaussian lobes to the $\boldsymbol{\omega_o}$-frame for shading. At run-time, we rotate each (non-zero) NDF bin's central direction into the $\boldsymbol{\omega_o}$-frame and bilinearly interpolate between the four nearest pre-rotated Gaussian lobe coefficient vectors. Alternatively, we could perform a *perfect* rotation of a canonically-oriented Gaussian ZH lobe to the rotated bin direction (Equation 10), but the precomputed solution allows us to avoid evaluating SH basis functions at run-time (whose number grow quadratically w.r.t. order). We illustrate the effects of different lobe direction sampling rates (relative to the NDF histogram resolution) on a simple smooth microfacet sphere, rendered in the half-vector space, in Figure 12. The effect of different rates can be observed in Figure 16.

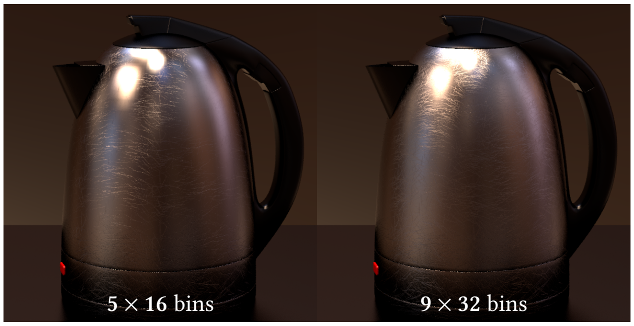


Fig. 11. Lower resolutions for our accumulated NDF histogram result in angular blurring of details (left), albeit a much faster rendering (2.5×, here).

The precomputed lobe coefficients still require very little storage: 29MB for order-30 SH, 114MB for order-60 and 714MB for order-150.

6.3 Naïve NDF SAT and Relationship to Han et al. [2007]

We combine a SAT-like NDF histogram representation with rotated Gaussian lobes in order to compute the SH coefficients of a multi-scale NDF, for a certain filtering footprint. An alternative approach would be to *directly* store SH coefficients of the accumulated NDF, in an SAT-like data structure; then, when querying for a specific footprint's NDF's SH coefficients, we can similarly apply an SAT-like 4-query sampling (as in Figure 6). At a high-level, this is similar to the data structure proposed by Han et al. [2007], with the main differences being that they leverage MIP-hardware to compute a hierarchy of footprints, and then use a single texture sample to query a specific footprint's NDF's coefficients.

We implemented both of these strategies and, in both cases, there are significant performance and scalability constraints that preclude their use for high, ground truth-quality renderings with high-resolution appearance maps. Firstly, memory-wise, storing SH coefficients per texel quickly becomes prohibitive as the normal map resolution increases, and the maximum SH order $\mathcal{F}_{\max}$ increases: a 2048 × 2048 normal map and $\mathcal{F}_{\max} = 100$ would require **more than 160GB** of storage for single-precision floats. And if a MIP-hierarchy is employed, instead of an SAT lookup approach, then 213GB would be needed.

Apart from storage, another downside is that the NDF coefficients would be expressed in the $\boldsymbol{n}$ local shading frame (the same as the NDF histogram parameterization) and, in order to shade with realistic microfacet models in the half-vector $\boldsymbol{\omega_h}$ frame, these coefficients need to be rotated (for each pixel sample) at run-time, which is costly in SH [Nowrouzezahrai et al. 2012].

We instead decouple memory costs due to increasing SH order from the asymptotic memory costs, as increases in the SH order only affect the size of our pre-rotated Gaussian lobe coefficients; and, as discussed earlier, we *could* completely avoid this precomputation by performing the fast ZH rotation on-the-fly. This decoupling also allows us to avoid costly SH rotations when changing between the local shading frame and the half-vector rendering frame.

Table 1. Memory and performance comparisons for our examples.

Scene	Normal Map	Max SH Order	Total Memory	Render Times		Speed-up Factor
				Our Method	[Yan et al. 2016] or [Belcour et al. 2017]	
Cutlery	2048 × 2048	60	2.7GB	50.7s	7.1m	8×
Snails	2048 × 2048	20	2.3GB	3.53m	121.4m	34×
Torus	512 × 512	15	172MB	4.89s	2.35m	28.8×
Kettle (3×)	2048 × 2048	60	2.7GB	1.22m	-	-
Kettle	1024 × 1024	60	1GB	2.43m	-	-

6.4 Light Source Coefficients

When computing Equation 6 as a double product integral, we require the SH coefficients of the filtered multi-scale NDF (i.e., the rotated Gaussian NDF lobes) and the incident illumination (both expressed in half-vector space). Rewriting Equation 7 to make explicitly the double product integral decomposition, we have

$$\int_{\Omega_h} \left[\widehat{L}_{\mathrm{i}}(\bar{\boldsymbol{x}}, \boldsymbol{\omega}_o, \boldsymbol{\omega}_h)\,(4(\boldsymbol{\omega}_h \cdot \boldsymbol{\omega}_o))\right]\left[\widehat{D}(\mathcal{P}, \boldsymbol{\omega}_h)\right] d\boldsymbol{\omega}_h$$
$$= \int \Big[\textstyle\sum_j \underbrace{\int \widehat{L}_{\mathrm{i}}(\bar{\boldsymbol{x}}, \boldsymbol{\omega}_o, \boldsymbol{\omega})(4(\boldsymbol{\omega}\cdot\boldsymbol{\omega}_o))y_j(\boldsymbol{\omega})d\boldsymbol{\omega}}_{L_j^{\mathrm{SH}}(\boldsymbol{\omega}_o)}\Big]\Big[\textstyle\sum_k \underbrace{D_k(\mathcal{P})}_{\text{Eq. 12}}\Big] d\boldsymbol{\omega}_h\,,$$

where the lighting SH coefficients L_j^{SH}, when expressed in the half-vector space, depend explicitly on the viewing direction $\boldsymbol{\omega}_o$. These coefficients also include the change-of-parameterization Jacobian.

Similarly to our discretization of the Gaussian roughness lobe coefficients, we pre-tabulate SH coefficients for the lighting, discretized at the same resolution as the Gaussian lobes (and also bilinearly interpolated at run-time; see Figure 17). This table requires three-times (or as many spectral components used) the storage as the Gaussian lobe coefficient table, due to separate RGB coefficients for lighting (so, e.g., just under 342MB for order-60).

7 RESULTS

We report render times from a dual Intel Xeon E5-2683 with 32 cores and 128 GB of RAM, and normal maps have a resolution of 2048 × 2048, unless stated otherwise. In all cases, render times do not include renderer initialization, a process that typically takes roughly 4 seconds for our scenes. In addition to this initialization cost, our method requires no more than 9 seconds to construct all of its internal data structures. As discussed in Section 6, the memory consumption of our approach depends on storing the spherical NDF histograms, the rotated Gaussian roughness lobe coefficients, and the rotated lighting coefficients. The latter of these two can be shared across materials, require negligible memory compared to the spherical NDF histogram texture, and require a short precomputation time dependent on projection quality, roughly 7 minutes for all 65 × 128 order 60 projections, each computed with half a million numerical integration samples. Figures 15-18 illustrate the effect of our various simplifications and parameter settings. Table 1 summarizes our memory usage, and we discuss each scene below.

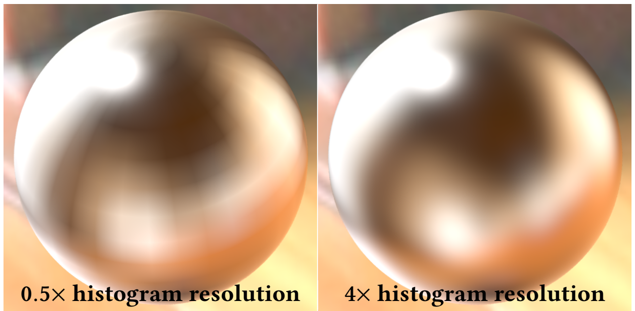


Fig. 12. When storing rotated Gaussian roughness SH coefficients below the spherical histogram resolution (left), visible re-sampling artifacts can appear. We use a conservative discretization of 4× the histogram's resolution.

The **snails** scene uses the flakes normal map [Yan et al. 2014, 2016], an environment map and a single point light, simulating global illumination with a single indirect bounce. Our method obtains a converged result (see supplemental video) in 2m:15s, and Belcour et al.'s approach [2017] generates 11spp in equal time.

The **cutlery** scene uses a scratched metal normal map [Yan et al. 2014, 2016] with environmental and point lighting, and our method converges in 0m:51s, whereas the method of Yan et al. [2016] can only generate 9 spp in equal time (far from enough to convergence).

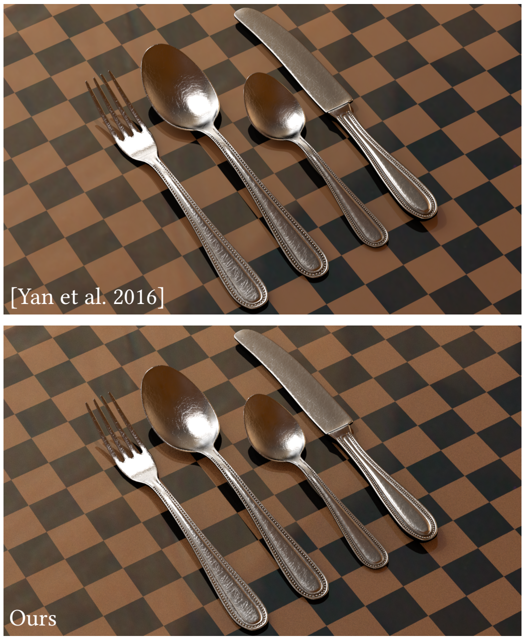


Fig. 13. Cutlery scene using the same environment map and point light, shadowed direct lighting only. Our method approached the ground truth in 50.7s while [Yan et al. 2016] required 7m:6s to reach the same quality at 2116 samples per pixel.

The **torus** scene uses a 512 × 512 scales normal map that is tiled repeatedly in the scene, with lighting from a lower-frequency environment map. This allows us to use significantly lower effective shading bandwidths (see Section 4.2) and we obtain a converged rendering in less than 5 seconds, whereas the state-of-the-art [Yan et al. 2016] rendered a single sample in 6 seconds (and our baseline G×D fired 2spp in equal time). Fully converged renderings for these two alternative baselines took 2m:25s and 13m:8s, respectively.

For the **kettle**, all lighting in this scene is due to an environmental source and and we use a metal panel 1024 × 1024 normal map. We render this scene to convergence in 2m:27s.

8 CONCLUSION AND FUTURE WORK

We present a scalable appearance model for filtered reflections from high-frequency microfacet models under complex illumination. We demonstrate significant speed-ups, up to 10× for direct illumination and 50× for global illumination, compared to the state-of-the-art in these two scenarios [Belcour et al. 2017; Yan et al. 2016]. Our method includes limitations due to our assumptions, which we discuss shortly.

Axis-aligned footprints are inherent to an SAT-like data structure, and as such, susceptible of being extended to arbitrary convex footprints [Piponi 2013].

Light source coefficients are precomputed once, per environment map. Area lights require on the fly SH coefficient calculation [Belcour et al. 2018; Wang and Ramamoorthi 2018]. An interesting direction for future work is to obtain analytical expressions (e.g. for spherical sources) that are suitable for half-vector space shading.

For **F** and **G**, an alternative to our decoupling term $\overline{FG}$ is to compute a triple product integral using tripling coefficients. This would increase the complexity of our method with negligible benefit, depending on the scenario.

An interesting avenue of future work is to explore interactive approximations of our method. Initial experiments show that a naïve port of our direct integrator to a shader-based GPU renderer yields performances on the order of 1Hz at a resolution of 1600 × 900 on an NVIDIA GTX 1080 (with 8GB of vRAM) for the cutlery scene from Figure 1. More aggressive data compression, a better trade-off between data accesses and compute (e.g., rotating the canonical ZH Gaussian roughness lobes by directly computing fast ZH rotations), and perhaps an alternative spherical basis (e.g., spherical RBFs) could increase performance into the realm of real-time applications.

ACKNOWLEDGMENTS

We would like to thank Wenzel Jakob, Ling-Qi Yan and Laurent Belcour for source code of their implementations. Computing resources provided by Calcul Québec and the National Systems of Compute Canada. Funding for this research was made possible by Consejo Nacional de Ciencia y Tecnología (CONACYT) of Mexico and the Natural Sciences and Engineering Council of Canada.

REFERENCES

Asen Atanasov and Vladimir Koylazov. 2016. A Practical Stochastic Algorithm for Rendering Mirror-like Flakes. In *ACM SIGGRAPH 2016 Talks (SIGGRAPH '16)*. ACM, New York, NY, USA, Article 67, 2 pages.

Mahdi Bagher, Cyril Soler, and Nicolas Holzschuch. 2012. Accurate fitting of measured reflectances using a Shifted Gamma micro-facet distribution. *Computer Graphics Forum* 31, 4 (June 2012).

Mahdi M. Bagher, John Snyder, and Derek Nowrouzezahrai. 2016. A Non-Parametric Factor Microfacet Model for Isotropic BRDFs. *ACM Transactions on Graphics* 36, 6 (Aug. 2016).

P. Beckmann and A. Spizzichino. 1963. The Scattering of Electromagnetic Waves from Rough Surfaces. *New York: Pergamon* (1963).

Laurent Belcour, Guofu Xie, Christophe Hery, Mark Meyer, Wojciech Jarosz, and Derek Nowrouzezahrai. 2018. Integrating Clipped Spherical Harmonics Expansions. *ACM Trans. Graph.* 37, 2, Article 19 (March 2018), 12 pages.

Laurent Belcour, Ling-Qi Yan, Ravi Ramamoorthi, and Derek Nowrouzezahrai. 2017. Antialiasing Complex Global Illumination Effects in Path-Space. *ACM Trans. Graph.* 36, 1, Article 75b (Jan. 2017).

Robert L. Cook and Tony DeRose. 2005. Wavelet noise. *Transactions on Graphics* 24, 3 (July 2005), 803–811.

R. L. Cook and K. E. Torrance. 1982. A Reflectance Model for Computer Graphics. *ACM Trans. Graph.* 1, 1 (Jan. 1982), 7–24.

Jonathan Dupuy, Eric Heitz, Jean-Claude Iehl, Pierre Poulin, Fabrice Neyret, and Victor Ostromoukhov. 2013. Linear efficient antialiased displacement and reflectance mapping. *ACM Transactions on Graphics* 32, 6 (Nov. 2013), 1–11.

Jonathan Dupuy, Eric Heitz, Jean-Claude Iehl, Pierre Poulin, and Victor Ostromoukhov. 2015. Extracting Microfacet-based BRDF Parameters from Arbitrary Materials with Power Iterations. *Computer Graphics Forum* (2015), 10.

Charles Han, Bo Sun, Ravi Ramamoorthi, and Eitan Grinspun. 2007. Frequency Domain Normal Map Filtering. *ACM Trans. Graph.* 26, 3, Article 28 (July 2007).

Eric Heitz, Stephen Hill, and Morgan McGuire. 2018. Combining Analytic Direct Illumination and Stochastic Shadows. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D '18)*. ACM, New York, NY, USA, Article 2, 11 pages.

Eric Heitz, Derek Nowrouzezahrai, Pierre Poulin, and Fabrice Neyret. 2013. Filtering Color Mapped Textures and Surfaces. In *ACM Siggraph Symposium on Interactive 3D Graphics and Games*. ACM, New York, NY, USA.

Homan Igehy. 1999. Tracing Ray Differentials. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 179–186.

Wenzel Jakob, Miloš Hašan, Ling-Qi Yan, Jason Lawrence, Ravi Ramamoorthi, and Steve Marschner. 2014. Discrete Stochastic Microfacet Models. *ACM Trans. Graph.* 33, 4, Article 115 (July 2014), 10 pages.

Ares Lagae, Sylvain Lefebvre, George Drettakis, and Philip Dutré. 2009. Procedural noise using sparse Gabor convolution. *j-TOG* 28, 3, Article 54 (Aug. 2009).

Addy Ngan, Frédo Durand, and Wojciech Matusik. 2005. Experimental Analysis of BRDF Models. In *Eurographics Workshop on Rendering*.

Derek Nowrouzezahrai, Patricio Simari, and Eugene Fiume. 2012. Sparse Zonal Harmonic Factorization for Efficient SH Rotation. *ACM Transactions on Graphics* (2012).

Marc Olano and Dan Baker. 2010. LEAN Mapping. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D '10)*. ACM, New York, NY, USA, 181–188.

Ken Perlin. 2002. Improving noise. *Transactions on Graphics* 21, 3 (July 2002), 681–682.

Dan Piponi. 2013. Polygon kernels for image processing. US Patent 8,400,461 B1.

Boris Raymond, Gaël Guennebaud, and Pascal Barla. 2016. Multi-scale Rendering of Scratched Materials Using a Structured SV-BRDF Model. *ACM Trans. Graph.* 35, 4, Article 57 (July 2016), 11 pages.

Peter-Pike Sloan, Ben Luna, and John Snyder. 2005. Local, deformable precomputed radiance transfer. In *SIGGRAPH*. ACM, NY, USA.

Michael Toksvig. 2005. Mipmapping normal maps. *Journal of Graphics Tools 10* 3 (2005), 65–71.

K. E. Torrance and E. M. Sparrow. 1967. Theory for Off-Specular Reflection From Roughened Surfaces∗. *J. Opt. Soc. Am.* 57, 9 (Sep 1967), 1105–1114.

Eric Veach and Leonidas J. Guibas. 1995. Optimally Combining Sampling Techniques for Monte Carlo Rendering. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*. ACM, New York, NY, USA, 419–428.

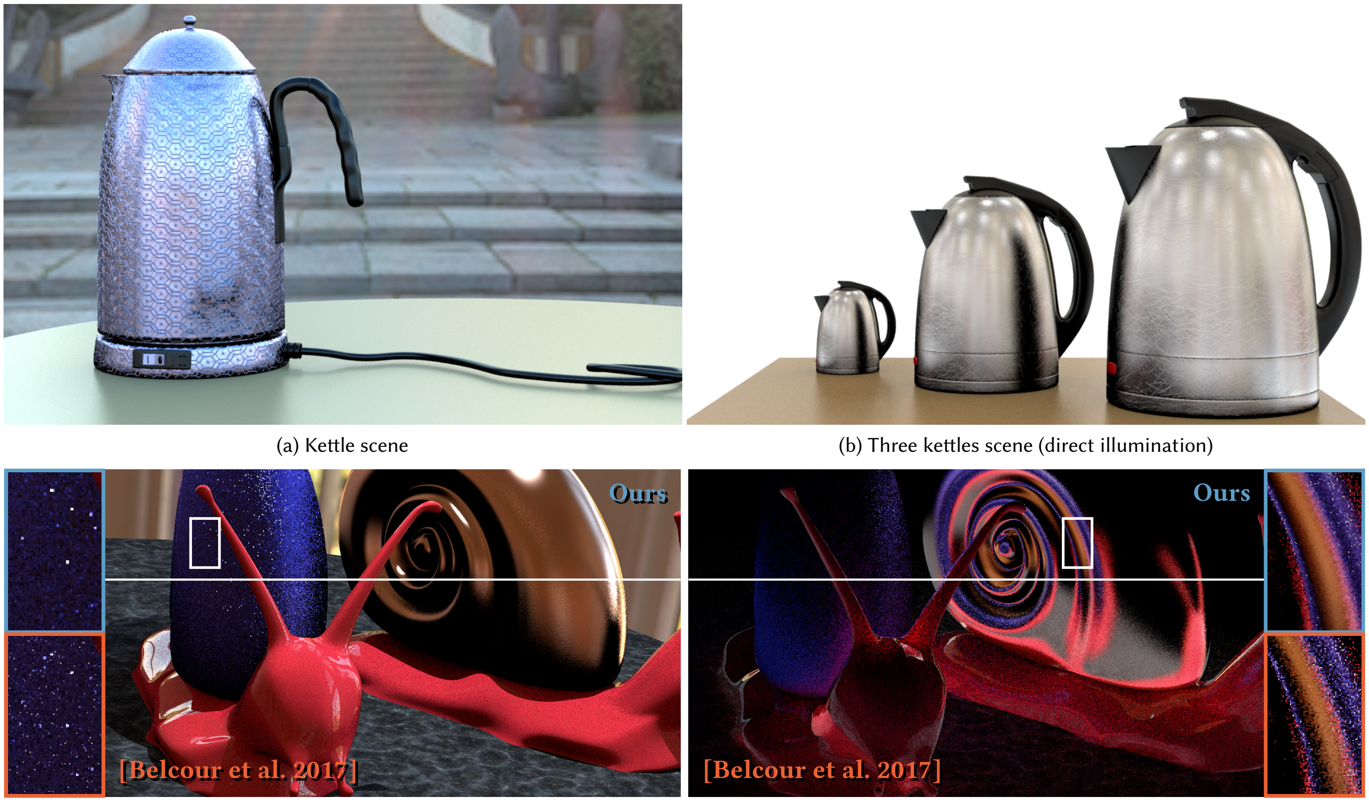
Zdravko Velinov, Sebastian Werner, and Matthias B. Hullin. 2018. Real-Time Rendering of Wave-Optical Effects on Scratched Surfaces. *Computer Graphics Forum* (2018).

Bruce Walter. 2005. *Notes on the Ward BRDF.* Technical Report. Cornell Program of Computer Graphics.

Bruce Walter, Stephen Marschner, Hongsong Li, and Keneth Torrance. 2007. Microfacet Models for Refraction through Rough Surfaces. *Eurographics Symposium on Rendering* (2007).

Jingwen Wang and Ravi Ramamoorthi. 2018. Analytic Spherical Harmonic Coefficients for Polygonal Area Lights. *ACM Trans. Graph.* 37, 4, Article 54 (July 2018), 11 pages.

Sebastian Werner, Zdravko Velinov, Wenzel Jakob, and Matthias B. Hullin. 2017. Scratch Iridescence: Wave-optical Rendering of Diffractive Surface Structure. *ACM Trans. Graph.* 36, 6, Article 207 (Nov. 2017), 14 pages.



(a) Kettle scene

(b) Three kettles scene (direct illumination)

(c) Snails scene, direct only

(d) Snails scene, indirect only

Fig. 14. Additional results for both direct and global illumination. Both (a) and (b) showcase additional models and environment maps with very high fidelity lighting while remaining temporally stable and fast to compute, 2m:27s and 1m:13s, respectively. Figures (c) and (d) are a breakdown of a frame from the Snails scene, showcasing the performance of our method across both direct and first-bounce indirect illumination in a global illumination context. In both cases, our results handily beat [Belcour et al. 2017] at equal time (11 spp).

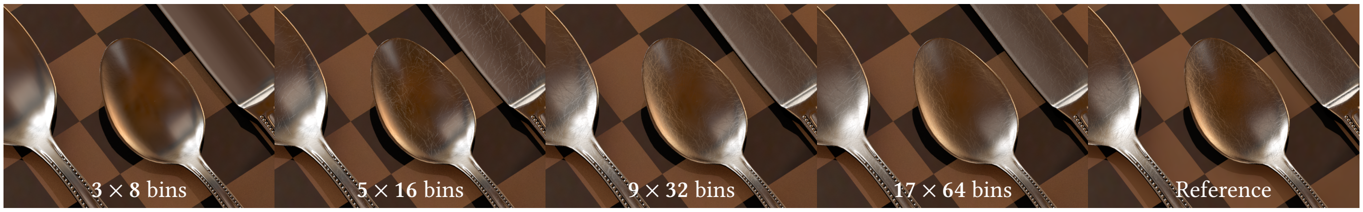


Fig. 15. We rely on a spherical histogram that discretizes the upper hemisphere into bins (see Section 4.1). The resolution of this histogram directly impacts the minimally reproducible detail size: lower resolutions progressively lose detail in the normal map, effectively smoothing it out. Our selected setting (9×32) is adequate for all scenes we have tested, when compared to the ground truth.

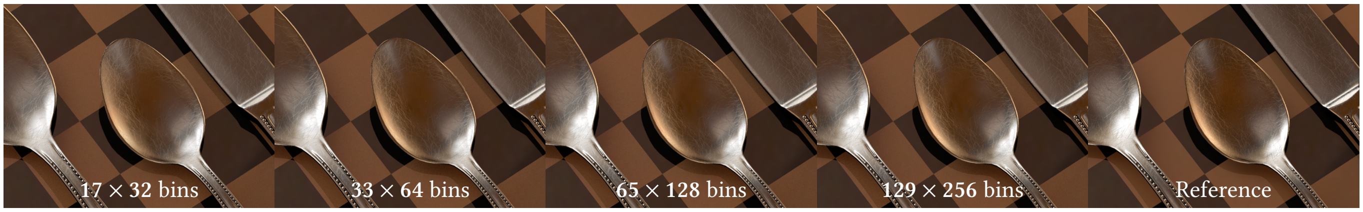


Fig. 16. Impact of the resolution of the rotated ZH Gaussian roughness lobes (see Section 6.2). As is apparent, the resolution does not have a significant effect on image quality for the resolutions we tested. We use a conservative setting of 65×128.

Ling-Qi Yan, Miloš Hašan, Wenzel Jakob, Jason Lawrence, Steve Marschner, and Ravi Ramamoorthi. 2014. Rendering Glints on High-resolution Normal-mapped Specular Surfaces. *ACM Trans. Graph.* 33, 4, Article 116 (July 2014), 9 pages.

Ling-Qi Yan, Miloš Hašan, Steve Marschner, and Ravi Ramamoorthi. 2016. Position-normal Distributions for Efficient Rendering of Specular Microstructure. *ACM Trans. Graph.* 35, 4, Article 56 (July 2016), 9 pages.

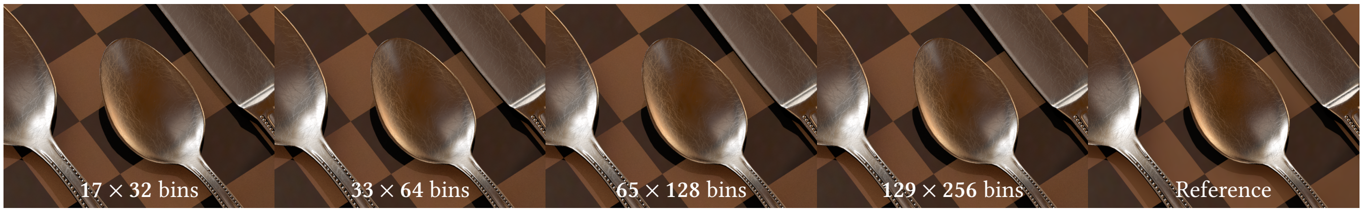


Fig. 17. Impact of the resolution of the environment light's SH coefficients table (see Section 6.4). As with Figure 16, the effect of a lower resolution is minor. We again conservatively use 65 × 128.

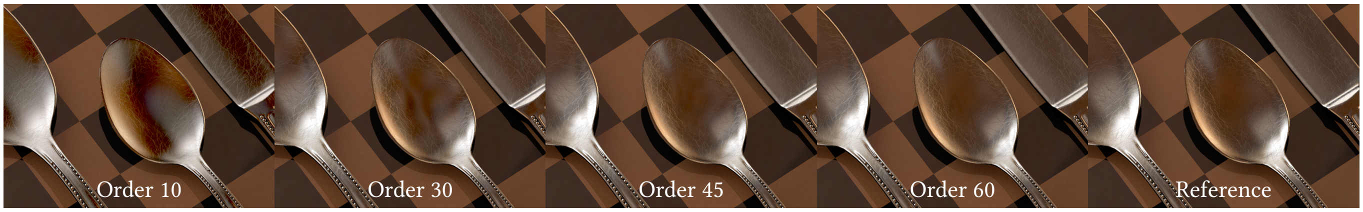


Fig. 18. Impact of the chosen band limit for the NDF or light spherical harmonics. Band limiting either is equivalent. Artifacts start to appear when either band limit is chosen below the true band limit of the respective signal (i.e. the maximum of the band limit of the NDF and the environment map). For this scene, Order 60 was sufficient.

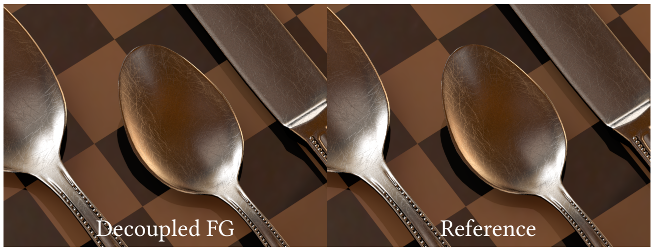


Fig. 19. Effect of the simplification made in Equation 6 and the proposed alternative in Equation 14. Both images were generated using our reference G×D implementation with sufficient sampling to match ground truth.

Ling-Qi Yan, Miloš Hašan, Bruce Walter, Steve Marschner, and Ravi Ramamoorthi. 2018. Rendering Specular Microgeometry with Wave Optics. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2018)* 37, 4 (2018).

Tobias Zirr and Anton S. Kaplanyan. 2016. Real-time Rendering of Procedural Multiscale Materials. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D '16)*. ACM, New York, NY, USA, 139–148.

A FRESNEL AND GEOMETRIC TERM APPROXIMATION

As is common in existing interactive and offline rendering models, we apply a factorization of the Fresnel and Geometry terms of our filtered shading model in Equation 6: $\overline{FG}(\boldsymbol{\omega}_o)$. One possible approximation would be to precompute and apply an "ideal" two-factor *integral-of-products equals product-of-integral* as $\overline{FG}(\boldsymbol{\omega}_o) = {}^1\!/_{(4(\boldsymbol{n}\cdot\boldsymbol{\omega}_o))} \int_{\Omega_i} F(\boldsymbol{\omega}_i \cdot \boldsymbol{\omega}_h)\, G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) d\boldsymbol{\omega}_i$, but we opt for a simpler alternative.

We instead evaluate the product of the Fresnel and Geometry terms in the view direction and the mirror reflection of the view, as

$$\overline{FG}(\boldsymbol{\omega}_o) = \frac{F(\boldsymbol{\omega}_r \cdot \boldsymbol{n})\, G(\boldsymbol{\omega}_r, \boldsymbol{\omega}_o)}{4(\boldsymbol{n} \cdot \boldsymbol{\omega}_o)}, \qquad (14)$$

where $\boldsymbol{\omega}_r = 2(\boldsymbol{n} \cdot \boldsymbol{\omega}_o)\boldsymbol{n} - \boldsymbol{\omega}_o$ is the mirrored view vector. The figure below illustrates the nature of this approximation, compared to the more costly integrated factorization, above.

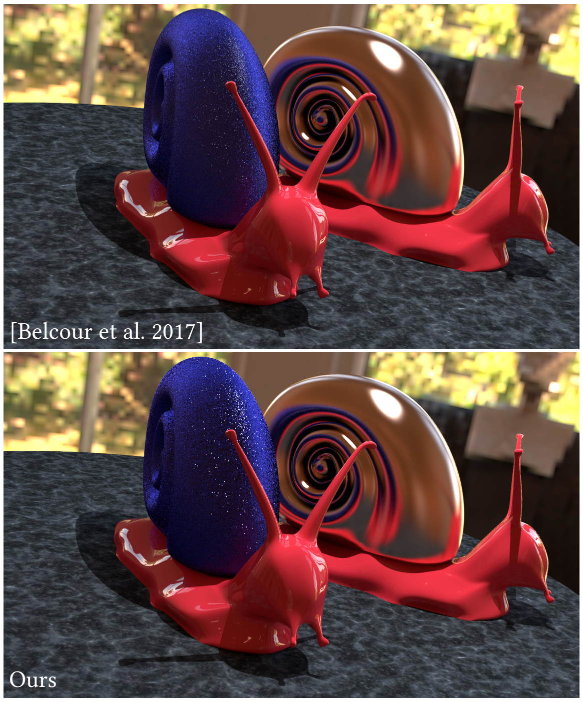
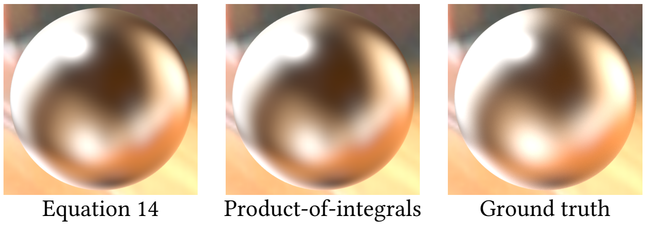


Fig. 20. Equal quality comparison for global illumination in the snails scene. [Belcour et al. 2017] took 121m:24s to render at 2048 samples per pixel while our method finished in 3m:32s.


Equation 14
Product-of-integrals
Ground truth