Eurographics Symposium on Rendering - Experimental Ideas & Implementations (2017)
P. Sander and M. Zwicker (Editors)

Frequency Based Radiance Cache for Rendering Animations

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

[1]Université de Montréal, [2]Unity Technologies, [3]McGill University

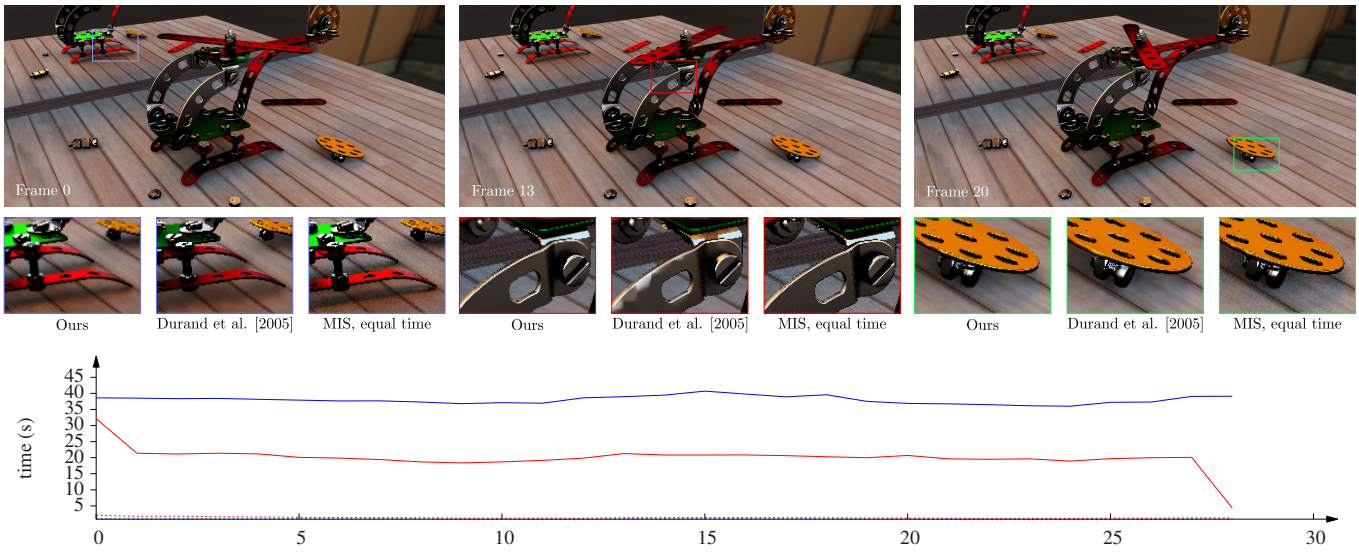


Figure 1: *For an equal quality target, our method (red curve) achieve almost two time faster rendering time for a better rendering quality than the method of Durand et al. [DHS*05] (blue curve). Insets from different images of the sequences (top) show how our method allows to reconstruct images with much less artifacts and less noise than Durand et al. [DHS*05], and an equal-time MIS integrator [VG95].*

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 finaly 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).

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 [HPJ12, GJTS12] and importance sampling [VG95]), 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 [DHS*05] or first-order [RMB07] 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 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 [DHS*05], 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 addi-

DOI: 10.2312/sre.20171193

tionally 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
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. [DHS*05] 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 [BSS*13a], distribution effects [ETH*09, SSD*09, BSS*13b, MYRD14], soft shadows from geometric sources and ambient occlusion [EDR11, EHDR11, MWR12], and diffuse indirect illumination [MWRD13]. 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. [LAC*11, LALD12] 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 [RKZ11, DSHL10, SD11, LWC12, BRM*16] 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 [WH92, KGPB05] 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 [SJJ12] interpolants. Again, here the shading integral is computed independently at each sample (without any adaptivity), and temporal coherence is not handled. Bala et al. [BDT99] devise radiance interpolants and cache re-use oracles based on error estimate bounds on (potentially reusable) shading samples, and the *render cache* system [WDP99, BWG03, VALBW06] 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 [MA06] 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 2 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 2(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 2(c));
3. using the information gathered in #1 and #2, we adaptively sample screen space (Figure 2(e)) according to the difference between the conservative sampling density determined by our frequency analysis (Figure 2(b)) and the density of radiance points reprojected from previous frames (Figure 2(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 2(f)).

We distinguish ourselves from previous techniques by:

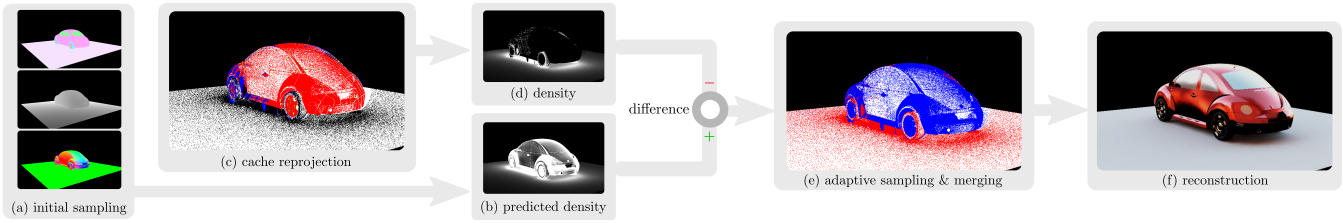


Figure 2: *From an initial sparse image-space sampling ((a), showing part of our extended G-buffer) we estimate the necessary sampling density for artifact free reconstruction (b). We then reprojection & validate samples from our temporal cache ((c), in red) and compute the sampling density from this 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.*

- 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).

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.

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)

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 [Kaj86]:

$$L(\mathbf{p},\boldsymbol{\omega}_v) = \int_{\mathcal{H}^2} L_\infty(\boldsymbol{\omega}_l) V(\mathbf{p},\boldsymbol{\omega}_l) \rho(\mathbf{p},\boldsymbol{\omega}_v,\boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n}) d\boldsymbol{\omega}_l, \quad (1)$$

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†. We remark that the integral in Equation 1 can be interpreted as a windowing of the incident lighting by a filter comprised of the product of the BRDF and cosine term [RH01]. 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 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 3). 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 [BS87, TS67]. 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 :

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

In practice, we approximate the gamma functions in Equation 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

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

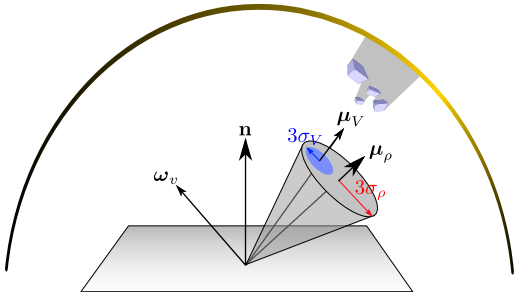


Figure 3: *We only consider the frequency content of the (distant) incident lighting and visibility within the view-evaluated BRDF footprint (in yellow), defined as a cone centered about the mean reflection direction of the BRDF slice $\boldsymbol{\mu}_\rho$ with radius equal to three standard deviation 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.*

time. For microfacet BRDFs, we first find the Beckmann-equivalent shininess such as $e_{MF} = \sqrt{2/(2+e_{Phong})}$ [WMLT07].

It is reasonable to treat any BRDF variation within the cone as negligible compared to that of the incident light (see Figure 3), 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 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 $\boldsymbol{\mu}_\rho$ [BSS*13a]: this data is stored in a mip-like hierarchy of spherical textures (see Figure 4). 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 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 [RH01, RH02, KC07, BSS*13a] 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.

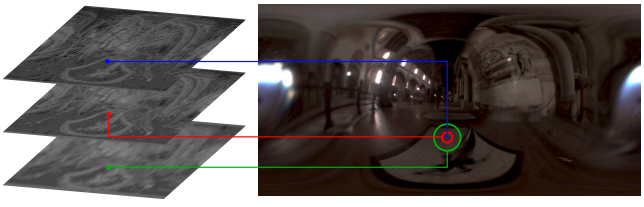


Figure 4: *We precompute the distant environment light's bandwidth for multiple footprint sizes (left), at discrete footprint directions. During rendering, we query this structure to obtain the lighting's bandwidth within the BRDF's footprint (right).*

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:

$$N_\Omega = f_V + 4B^2_{L_\infty}. \tag{3}$$

To determine the amount of occlusion, we accumulate a statistical representation of visibility during a first Monte Carlo estimation of Equation 1, which we compute using multiple importance sampling (MIS) [Vea97] and $N_L = 4B^2_{L_\infty}$ 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,

$$\begin{aligned} \boldsymbol{\mu}_o &= \frac{1}{A_o}\sum_{i=1}^{N_L} \boldsymbol{p}_i \rho_i (1-v_i), \\ \Sigma_o &= \frac{N}{(N_L-1){A_o}^2}\sum_{i=1}^{N_L} (\rho_i(1-v_i))^2 (\boldsymbol{p}_i - \boldsymbol{\mu}_o)(\boldsymbol{p}_i - \boldsymbol{\mu}_o)^T, \end{aligned} \tag{4}$$

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 $\boldsymbol{\mu}_\rho$) 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

Figure 5: *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.*

$\rho_i = p_\rho(\boldsymbol{\omega}_i)$ 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 3) and is derived as

$$\sigma_V = max(\sqrt{(\widehat{\boldsymbol{\mu}_{\rho_t}})^T \Sigma_o \widehat{\boldsymbol{\mu}_{\rho_t}}}, \sqrt{(\widehat{\boldsymbol{\mu}_{\rho_b}})^T \Sigma_o \widehat{\boldsymbol{\mu}_{\rho_b}}}), \tag{5}$$

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, $f(x) = (N^\Omega_{\max} - N_\Omega)\, e^{-1/2(x-1/2)^2}$ yields good results, where $N^\Omega_{\max}$ is our maximum spherical sampling budget (Figure 5). We use the remaining integration samples to improve our Monte Carlo estimate of Equation 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 a 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.

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 [DHS*05, BSS*13a] as illustrated in Figure 6: beginning with the bandwidth emitted within the solid angle of the (distant) light subtended by the BRDF's footprint B_{L_∞} (Figure 6(a)), we analyze the reflected bandwidth (Figure 6(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 6(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 deviation (corresponding to the 99$^{\text{th}}$ percentile of a Gaussian) as the

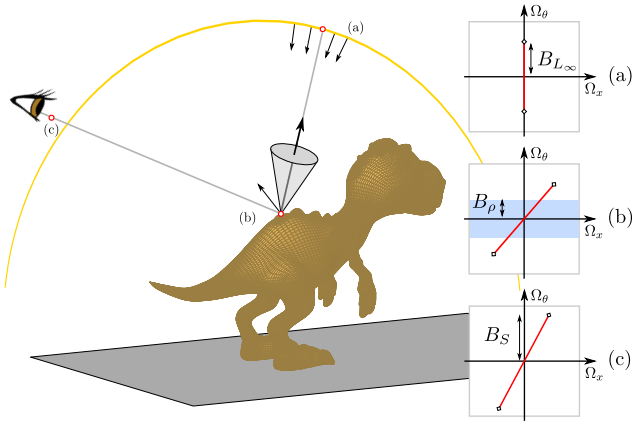


Figure 6: *To estimate the screen-space bandwidth B_S, we start with the emitted (angular) bandwidth of the environment light B_{L_∞} reduced to the BRDF's footprint (a), we then apply the BRDF bandlimit B_ρ (b) and modulated according to the curvature, foreshortening, and spatial transport (c). The angular bandwidth serves directly as the screen-space bandwidth estimate.*

bandwidth of the spectrum from its variance: $B \approx 3\sqrt{\sigma^2}$. We use a compact 2D variance representation [BSS*13b] 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. 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

$$t_{min} = \mu_t - 3 \times \sigma_t. \tag{6}$$

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

$$\Sigma_{S,\theta} = \mathbf{T}_{\mathbf{x}\to\mathbf{v}}\left(\mathbf{C}_v \circ \mathbf{B}_{\rho_s} \circ \mathbf{C}_L\right)\left(\sigma^2_{L_\infty} + \mathbf{T}_{V\to\mathbf{x}}(t_{min})\right), \tag{7}$$

where each of the five operators above are simply 2×2 matrices [BSS*13a] (see Appendix A 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; $\sigma^2_{L_\infty}$ 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 6 [DHS*05, EDR11].

From the covariance matrix $\Sigma_{S,\theta}$ we retrieve the angular part σ^2_θ defining our final scalar image-space variance, and derive its band-

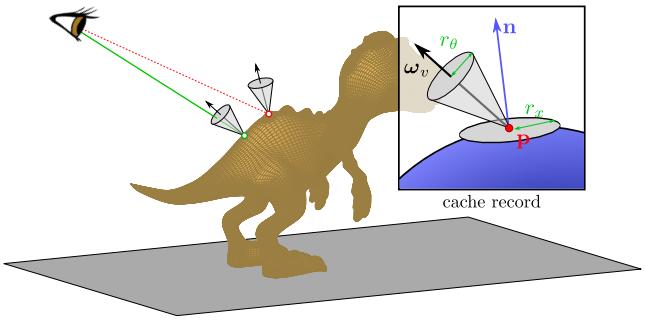


Figure 7: *During reprojection, we test whether the new viewing direction remains inside the BRDF's footprint and, if so, we project the sample onto the screen (plain green line); if not, we (optionally) discard the sample from the cache (dashed red line).*

width as stated above as

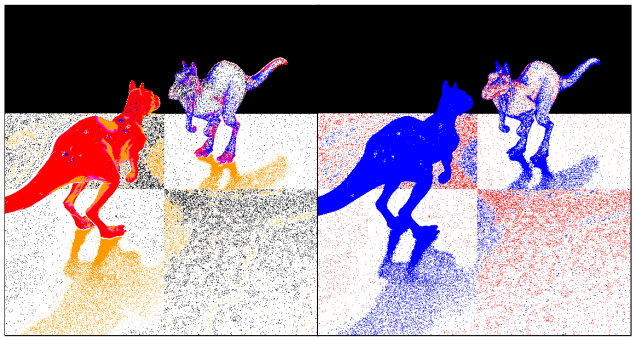
$$B_S \approx 3\sqrt{\sigma_\theta^2}. \tag{8}$$

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 [SGNS07] with filter sizes proportional to the pixel's unoccluded angular variance (see Appendices A, A) 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 7 and 8, 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



(a) Cache discard & reuse (b) Resampling

Figure 8: *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.*

used to define f_V. The spatial, angular and temporal bandwidths (B_x, B_θ, B_t) (see derivation in Appendix A) are used to define a circular surface patch, a cone of directions (see Figure 7, 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 $q_{\varepsilon_{\{x,\theta,t\}}}$ 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 7). 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 8(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.

Validating Reprojected Cache Points We use the spatial, and angular bandwidths to define a circular surface patch and directional cone with radii $r_x = 2\pi\varepsilon_x/B_x$ and $r_\theta = 2\pi\varepsilon_\theta/B_\theta$, where the directional

cone is centered along $\boldsymbol{\omega}_v$ (at the time of the radiance point's insertion into the cache), $\varepsilon_{\{x,\theta,t\}} = \text{acos}(1 - q_{\varepsilon\{x,\theta,t\}})$ is based on quality thresholds $q_{\varepsilon\{x,\theta,t\}}$ 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 7).

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 $\Delta t = {}^{2\pi\varepsilon}/_{B_t}$. We apply Egan et al.'s [ETH*09] 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 8(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 9). 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.

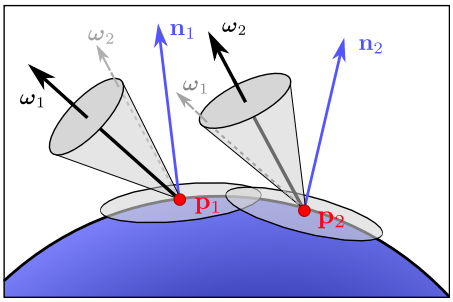


Figure 9: *We merge two cache points that reproject onto the same pixel only if both their spatial and angular bandwidths overlap enough. In this example, while the two cache point's angular cones ($\boldsymbol{\omega}_1$, $\boldsymbol{\omega}_2$) align (gray), their spatial bandwidths at $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ do not overlap enough. Consequently, the two points will not be merged.*

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 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 9), and assume that only the spherical visibility changes over time, but not the lighting nor reflectance:

$$L(\mathbf{p},\boldsymbol{\omega}_v,t) = \int_{\mathcal{H}^2} L_\infty(\boldsymbol{\omega}_l) V(\mathbf{p},\boldsymbol{\omega}_l,t) \rho(\mathbf{p},\boldsymbol{\omega}_v,\boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n}) d\boldsymbol{\omega}_l . \quad (9)$$

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

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

$$\Delta L(t) = \int_{\mathcal{V}_t} L_\infty(\boldsymbol{\omega}_l) \rho(\boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n}) d\boldsymbol{\omega}_l , \quad (10)$$

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:

$$L(\mathbf{p},\boldsymbol{\omega}_v,t) = \underbrace{[L(\mathbf{p},\boldsymbol{\omega}_v,0) - \Delta L(0)]}_{\text{constant w.r.t. time } t} + \Delta L(t) . \quad (11)$$

Here, $\Delta L(t)$ can be thought of as time-dependent antiradiance [DSDD07]. 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:

$$\begin{aligned}\widehat{L}(\Omega_t) &= \mathcal{F}[L(\mathbf{p},\boldsymbol{\omega}_v,t)] = \mathcal{F}\Big[[L(\mathbf{p},\boldsymbol{\omega}_v,0) - \Delta L(0)] + \Delta L(t)\Big] \\ &= A\delta(\Omega_t) + \mathcal{F}[\Delta L(t)], \end{aligned} \quad (12)$$

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

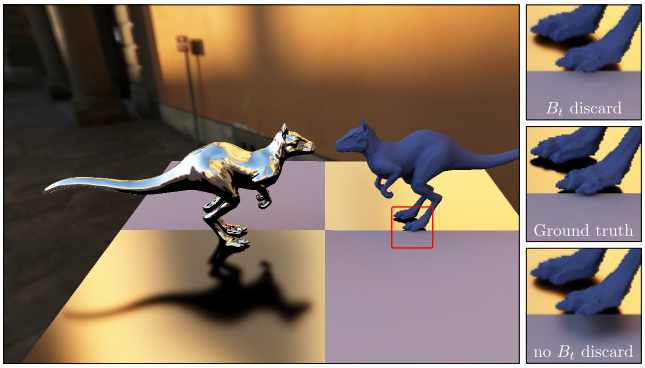


Figure 10: *We illustrate the need for our moving occluders bandwidth to discard samples during animations in the* KILLEROO *scene after 10 frames. Without this bandwidth, the cache incorrectly reproject shadows (bottom inset) onto the screen which result in a dark region in at the feet of the Killeroo.*

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 [ETH*09, EDR11, RAMN12]. 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 12 as an integration of the occluding patch in its *original configuration* at $V(0)$, but now lit under the distant lighting rotated according to the appropriate inverse (angular) rotation of the occluder motion:

$$\widehat{\Delta L}(\Omega_t) \approx \mathcal{F}\left[\rho_n \int_{\mathcal{V}_0} L_\infty(\boldsymbol{\omega}_l + t\dot{\mathbf{v}}) d\boldsymbol{\omega}_l\right], \tag{13}$$

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

$$\rho_n = \frac{1}{|\mathcal{V}_0|} \int_{\mathcal{V}_0} \rho(\mathbf{p}, \boldsymbol{\omega}_v, \boldsymbol{\omega}_l)(\boldsymbol{\omega}_l \cdot \boldsymbol{n}) d\boldsymbol{\omega}_l \tag{14}$$

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 13, 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:

$$\begin{aligned}\widehat{\Delta L}(\Omega_t) &\approx \rho_n \int_{\mathcal{V}_0} \frac{e^{i\Omega_t(\boldsymbol{\omega}_l[\theta]+\boldsymbol{\omega}_l[\phi])}}{|\dot{\mathbf{v}}|} \widehat{L}_\infty(|\dot{\mathbf{v}}|\Omega_t) d\boldsymbol{\omega}_l \\ &= \frac{\rho_n}{|\dot{\mathbf{v}}|} \widehat{L}_\infty(|\dot{\mathbf{v}}|\Omega_t) \left[\int_{\mathcal{V}_0} e^{i\Omega_t(\boldsymbol{\omega}_l[\theta]+\boldsymbol{\omega}_l[\phi])} d\boldsymbol{\omega}_l\right], \end{aligned} \tag{15}$$

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,

$$\left|\int_{\mathcal{V}_0} e^{i\Omega_t(\boldsymbol{\omega}_l[\theta]+\boldsymbol{\omega}_l[\phi])} d\boldsymbol{\omega}_l\right| \leq \underbrace{\int_{\mathcal{V}_0} d\boldsymbol{\omega}_l}_{V_\Omega}, \tag{16}$$

since $|e^{ix}| \leq 1$, and we can then bound the amplitude of $\widehat{\Delta L}(\Omega_t)$ in Equation 15 as

$$\left|\widehat{\Delta L}(\Omega_t)\right| \leq \frac{V_\Omega\, \rho_n}{|\dot{\mathbf{v}}|} \left|\widehat{L}_\infty(|\dot{\mathbf{v}}|\Omega_t)\right|. \tag{17}$$

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 13 results in an scaling of the spectrum $\widehat{L}_\infty$ by a factor of $|\dot{\mathbf{v}}|$ in Equation 15, the lighting's original bandwidth B_{L_∞} is similarly 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

$$B_{t,\text{single}} \leq |\dot{\mathbf{v}}| B_{L_\infty}. \tag{18}$$

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. We then approximate the final temporal bandwidth, and thus the *maximum* bandwidth across all occluding patches, as the mean plus 3 times the standard deviation,

$$B_t = \mu[|\dot{\mathbf{v}}|B_{L_\infty}] + 3\sigma[|\dot{\mathbf{v}}|B_{L_\infty}]. \tag{19}$$

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 10).

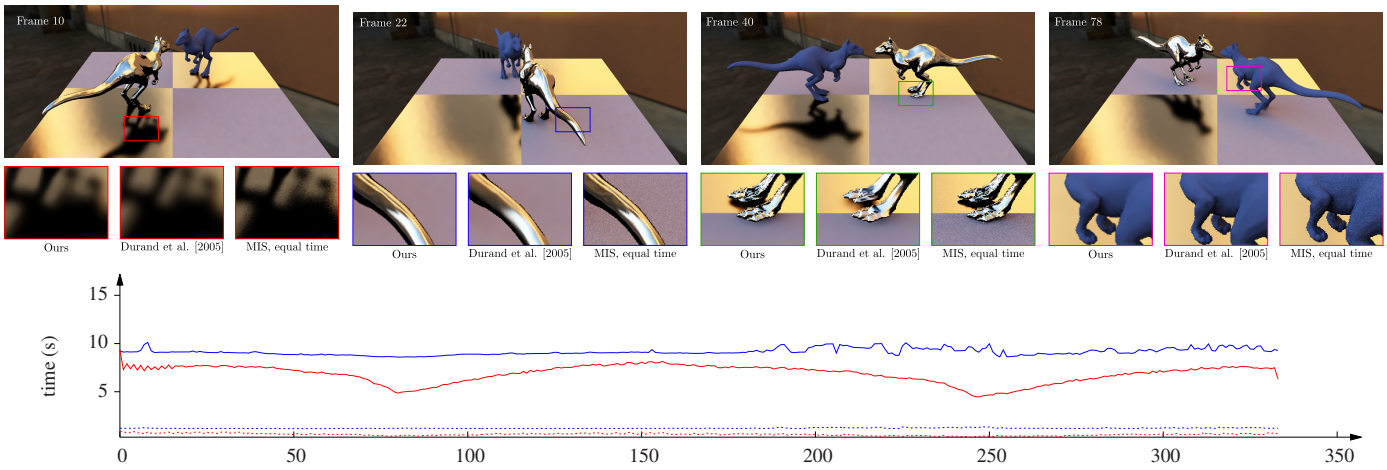


Figure 11: *Stills from the* KILLEROO *animation with one glossy and one diffuse Killeroo rotating above a diffuse and glossy checkerboard, lit by the* PISA *environment. Rendering times per frame for our method and Durand et al. [DHS*05] are in red and blue, respectively. Reconstruction times are provided using dashed lines of the same color. Insets compare feature reconstruction our approach, equal-time Durand et al. [DHS*05], and an equal-time MIS integrator.*

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:

$$L_i = T_i \sum_{p' \in \mathcal{N}_i} W_{p' \to i} L_{p'} \,, \tag{20}$$

where T_i is the texture value for pixel i. We use a standard cross-bilateral filter, as in previous shading reconstruction works [DHS*05], tailored to our predicted bandwidth estimates:

$$W_{p' \to i} = \exp\left[-\sum_{\mathbf{f}} \frac{1}{2\sigma_{\mathbf{f}}^2} ||\mathbf{f}_i - \mathbf{f}_{p'}||^2 \right], \tag{21}$$

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 1 (Section 4) uses the power heuristic [VG95] 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 [CT65] 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^{th} 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 8 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.

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 1) shows a rotating rotor blade in a static scene with simple Whitted-style indirect specular effects and texture mapping,

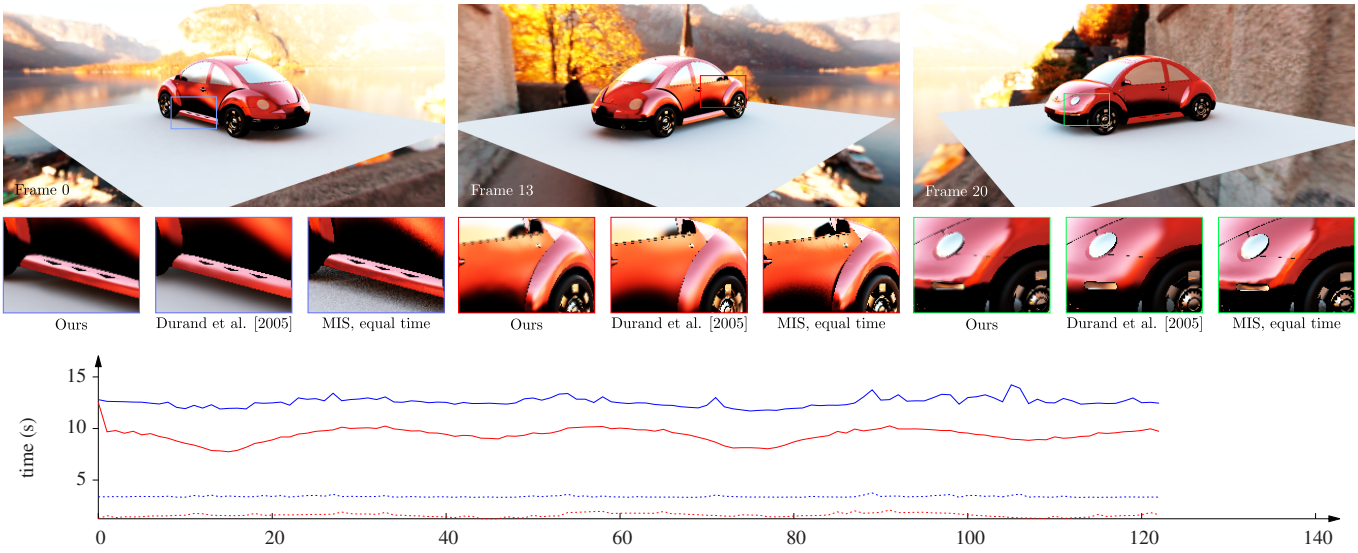


Figure 12: *Stills from the* CAR *animation with static geometry and a camera rotating around, lit by the* HALLSTAT *environment. Rendering times per frame for our method and Durand et al. [DHS*05] are in red and blue, respectively. Reconstruction times are provided using dashed lines. Insets compare feature reconstruction our approach, equal-time Durand et al. [DHS*05], and an equal-time MIS integrator.*

CAR (Figure 12) is an example of glossy, diffuse and mirror objects under camera rotation, KILLEROO (Figure 11) 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 $f_{max} = 8$, $q_{\varepsilon_x} = q_{\varepsilon_\theta} = 1e^{-4}$ and $q_{\varepsilon_x} = 1e^{-7}$.

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 [DHS*05] prototype (Figure 14): 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. [LWC12] (Figure 13). 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.

Another more subtle issue arises when trying to simply multiply the reconstructed shading by a texture value after the fact (Equation 20): 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

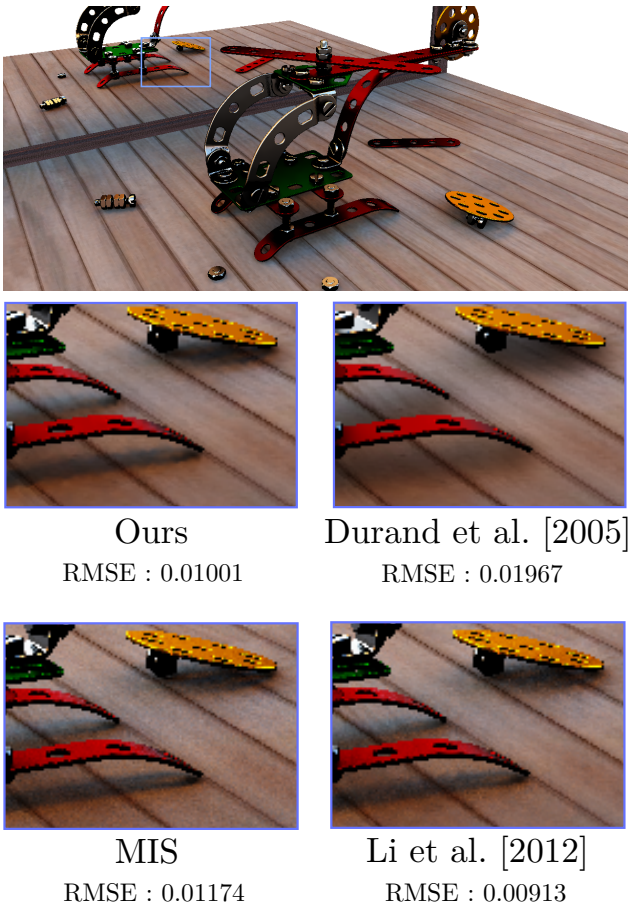


Figure 13: *We present an equal-time still frame comparison between our technique and the work of Li et al. [LWC12], Durand et al. [DHS*05] and MIS as reference.*

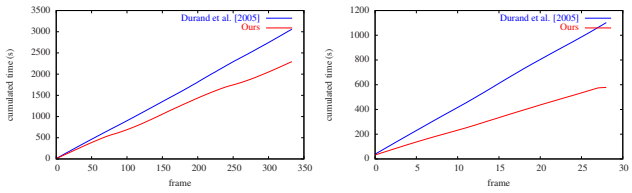


Figure 14: *When comparing cumulative rendering times, our cache (in red) allows for smaller rendering time compared to the non-caching method [DHS*05] (in blue) where radiance points are recomputed every frame. In the case of* HELICOPTER *we achieve almost half the rendering time since we reuse most of the cache samples.*

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 [SSD*09, BSS*13b] or motion blur [ETH*09, BSS*13b], 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 [EDR11, EHDR11] 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.

References

[BDT99] BALA K., DORSEY J., TELLER S.: Radiance Interpolants for Accelerated Bounded-Error Ray Tracing. 2

[BRM*16] BITTERLI B., ROUSSELLE F., MOON B., IGLESIAS-GUITIÁN J. A., ADLER D., MITCHELL K., JAROSZ W., NOVÁK J.: Nonlinearly weighted first-order regression for denoising monte carlo renderings. In *Computer Graphics Forum* (2016), vol. 35, 2

[BS87] BECKMANN P., SPIZZICHINO A.: *Scattering of Electromagnetic Waves from Rough Surfaces*. 3

[BSS*13a] BAGHER M. M., SOLER C., SUBR K., BELCOUR L., HOLZSCHUCH N.: Interactive rendering of acquired materials on dynamic geometry using frequency analysis. 2, 4, 5, 12

[BSS*13b] BELCOUR L., SOLER C., SUBR K., HOLZSCHUCH N., DURAND F.: 5D Covariance tracing for efficient defocus and motion blur. *ACM Transactions on Graphics 32*, 3 (2013), 2, 5, 11, 12, 13

[BWG03] BALA K., WALTER B., GREENBERG D.: Combining edges and points for interactive high-quality rendering. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2003) 22*, 3 (2003), 2

[CT65] COOLEY J. W., TUKEY J. W.: An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation 19*, 90 (May 1965), 9

[DHS*05] DURAND F., HOLZSCHUCH N., SOLER C., CHAN E., SILLION F. X.: A frequency analysis of light transport. *ACM Transactions on Graphics 24*, 3 (2005), 1, 2, 5, 9, 10, 11, 12, 13

[DSDD07] DACHSBACHER C., STAMMINGER M., DRETTAKIS G., DURAND F.: Implicit visibility and antiradiance for interactive global illumination. 7

[DSHL10] DAMMERTZ H., SEWTZ D., HANIKA J., LENSCH H. P. A.: Edge-avoiding À-Trous wavelet transform for fast global illumination filtering. In *Proceedings of High Performance Graphics* (June 2010), 2

[EDR11] EGAN K., DURAND F., RAMAMOORTHI R.: Practical filtering for efficient ray-traced directional occlusion. 2, 5, 8, 11, 12, 13

[EHDR11] EGAN K., HECHT F., DURAND F., RAMAMOORTHI R.: Frequency analysis and sheared filtering for shadow light fields of complex occluders. *ACM Transactions on Graphics 30*, 2 (Apr. 2011), 2, 11

[ETH*09] EGAN K., TSENG Y.-T., HOLZSCHUCH N., DURAND F., RAMAMOORTHI R.: Frequency analysis and sheared reconstruction for rendering motion blur. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009) 28*, 3 (2009), 2, 7, 8, 11

[GJTS12] GEORGIEV I., JAROSLAV K., TOMAS D., SLUSALLEK P.: Bidirectional Light Transport with Vertex Connection and Merging. 1

[HPJ12] HACHISUKA T., PANTALEONI J., JENSEN H. W.: *A Path Space Extension for Robust Light Transport Simulation.* Tech. rep., 1

[Kaj86] KAJIYA J. T.: The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques - SIGGRAPH '86* (New York, New York, USA, Aug. 1986), vol. 20, 3

[KC07] KÅŹIVÁNEK J., COLBERT M.: Real-time Shading with Filtered Importance Sampling. *Computer Graphics Forum 27*, 4 (June 2007), 4

[KGPB05] KRIVÁNEK J., GAUTRON P., PATTANAIK S., BOUATOUCH K.: Radiance caching for efficient global illumination computation. 2

[LAC*11] LEHTINEN J., AILA T., CHEN J., LAINE S., DURAND F.: Temporal light field reconstruction for rendering distribution effects. *ACM Transactions on Graphics 30*, 4 (July 2011), 2

[LALD12] LEHTINEN J., AILA T., LAINE S., DURAND F.: Reconstructing the indirect light field for global illumination. *ACM Transactions on Graphics 31*, 4 (July 2012), 2

[LWC12] LI T.-M., WU Y.-T., CHUANG Y.-Y.: SURE-based optimization for adaptive sampling and reconstruction. *ACM Transactions on Graphics 31*, 6 (Nov. 2012), 2, 10, 11

[MA06] MEYER M., ANDERSON J.: Statistical acceleration for animated global illumination. 2

[MWR12] MEHTA S. U., WANG B., RAMAMOORTHI R.: Axis-aligned filtering for interactive sampled soft shadows. 2

[MWRD13] MEHTA S. U., WANG B., RAMAMOORTHI R., DURAND F.: Axis-aligned filtering for interactive physically-based diffuse indirect lighting. 2

[MYRD14] MEHTA S., YAO J., RAMAMOORTHI R., DURAND F.: Factored Axis-Aligned Filtering for Rendering Multiple Distribution Effects. 2

[RAMN12] RAMAMOORTHI R., ANDERSON J., MEYER M., NOWROUZEZAHRAI D.: A theory of monte carlo visibility sampling. 8

[RH01] RAMAMOORTHI R., HANRAHAN P.: An efficient representation for irradiance environment maps. In *SIGGRAPH* (NY, 2001), 3, 4

[RH02] RAMAMOORTHI R., HANRAHAN P.: Frequency space environment map rendering. 4

[RKZ11] ROUSSELLE F., KNAUS C., ZWICKER M.: Adaptive Sampling and Reconstruction using Greedy Error Minimization. *ACM Transactions on Graphics 30*, 6 (2011), 2

[RMB07] RAMAMOORTHI R., MAHAJAN D., BELHUMEUR P.: A first-order analysis of lighting, shading, and shadows. 1

[SD11] SEN P., DARABI S.: On Filtering the Noise from the Random Parameters in Monte Carlo Rendering. *ACM Transactions on Graphics (TOG) 31*, 3 (May 2011), 2

[SGNS07] SLOAN P.-P., GOVINDARAJU N. K., NOWROUZEZAHRAI D., SNYDER J.: Image-based proxy accumulation for real-time soft global illumination. 6

[SJJ12] SCHWARZHAUPT J., JENSEN H. W., JAROSZ W.: Practical Hessian-based error control for irradiance caching. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2012) 31*, 6 (2012), 2

[SSD*09] SOLER C., SUBR K., DURAND F., HOLZSCHUCH N., SILLION F. X.: Fourier depth of field. *ACM Transactions on Graphics 28*, 2 (2009), 2, 11

[TS67] TORRANCE K., SPARROW E.: Theory for off-specular reflection from roughened surfaces. 3

[VALBW06] VELÁZQUEZ-ARMENDÁRIZ E., LEE E., BALA K., WALTER B.: Implementing the render cache and the edge-and-point image on graphics hardware. 2

[Vea97] VEACH E.: *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University, 4

[VG95] VEACH E., GUIBAS L. J.: Optimally combining sampling techniques for Monte Carlo rendering. In *SIGGRAPH 1995* (Sept. 1995), 1, 9

[WDP99] WALTER B., DRETTAKIS G., PARKER S.: Interactive rendering using the render cache. 2

[WH92] WARD G. J., HECKBERT P. S.: Irradiance gradients. In *Eurographics Workshop on Rendering* (1992), 2

[WMLT07] WALTER B., MARSCHNER S., LI H., TORRANCE K.: Microfacet Models for Refraction through Rough Surfaces. *Rendering Techniques (Proceedings of Eurographics Symposium on Rendering 2007)* (2007), 4

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 [DHS*05, EDR11, BSS*13b, BSS*13a].

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.

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

$$\Sigma = \begin{bmatrix} 0 & 0 \\ 0 & \sigma^2_{L_\infty} \end{bmatrix}. \tag{22}$$

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 7. The operator is defined as

$$\mathbf{T}_{d_1} = \begin{bmatrix} 1 & -d_1 \\ 0 & 1 \end{bmatrix}. \tag{23}$$

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 [DHS*05]:

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

$$\mathbf{C}_L = \begin{bmatrix} c_i & 0 \\ k & 1 \end{bmatrix}. \tag{24}$$

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 [BSS*13b], which can be rewritten in matrix form, with $b = 1/cov_{\theta,\theta}(\rho_s)$, as

$$\mathbf{B}_{\rho_s} = \begin{bmatrix} 0 & 0 \\ 0 & b \end{bmatrix}. \tag{25}$$

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

$$\mathbf{C}_v = \begin{bmatrix} c_v & 0 \\ c_v k & -1 \end{bmatrix}. \tag{26}$$

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

$$\mathbf{T}_{\mathbf{x}\to\mathbf{v}} = \begin{bmatrix} 1 & -d \\ 0 & 1 \end{bmatrix}. \tag{27}$$

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

$$\Sigma_c = \mathbf{C}_L^T\,\Sigma\,\mathbf{C}_L \qquad \Sigma_\rho = \Sigma_c - \frac{\Sigma_c^T\,\mathbf{B}_{\rho_s}\,\Sigma_c}{1+\mathrm{Tr}\left[\Sigma_c^T\,\mathbf{B}_{\rho_s}\right]} \qquad \Sigma^o_{S,\theta} = \mathbf{C}_v^T\,\Sigma_\rho\,\mathbf{C}_v, \tag{28}$$

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

$$\Sigma_{S,\theta} = \mathbf{T}^T_{\mathbf{x}\to\mathbf{v}}\,\Sigma^o_{S,\theta}\,\mathbf{T}_{\mathbf{x}\to\mathbf{v}}. \tag{29}$$

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 [DHS*05, EDR11]

$$\mathbf{T}_{V\to\mathbf{x}} = \begin{bmatrix} 1 & -t_{min} \\ 0 & 1 \end{bmatrix}. \tag{30}$$

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

$$\Sigma = \begin{bmatrix} a & 0 \\ 0 & \sigma^2_{L_\infty} \end{bmatrix}. \tag{31}$$

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

$$\Sigma_c = \mathbf{C}_L^T\,\Sigma_t\,\mathbf{C}_L, \quad \text{with} \quad \Sigma_t = \mathbf{T}^T_{V\to\mathbf{x}}\,\Sigma\,\mathbf{T}_{V\to\mathbf{x}}. \tag{32}$$

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

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 8 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

$$(\sigma^o_S)^2 = \Sigma^o_{S,\theta}[1,1] = (4c_v^2k^2\sigma^2_{L_\infty})/(1+b\sigma^2_{L_\infty}),$$
$$(\sigma^o_\theta)^2 = \Sigma^o_{S,\theta}[2,2] = (\sigma^2_{L_\infty})/(1+b\sigma^2_{L_\infty}), \text{ and}$$
$$\sigma^2_\theta = \Sigma_{S,\theta}[2,2] = (1-2c_v d k)^2\sigma^2_{L_\infty}/(1+b\sigma^2_{L_\infty}).$$

Occluded Case

$$(\sigma^o_S)^2 = \lim_{a\to\infty}\Sigma^o_{S,\theta}[1,1] = \left((c_i+2t_{lim}k)^2 + bc_i^2\sigma^2_{L_\infty}\right)/bt^2_{lim},$$
$$(\sigma^o_\theta)^2 = \lim_{a\to\infty}\Sigma^o_{S,\theta}[2,2] = b^{-1}, \text{ and}$$
$$\sigma^2_\theta = \lim_{a\to\infty}\Sigma_{S,\theta}[2,2] = \frac{(t_{min}-c_v d\,(c_i+2t_{min}k))^2 + bc_v^2c_i^2d^2\sigma^2_{L_\infty}}{bt^2_{min}}.$$

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

$$B_x = 3\sqrt{\sigma^{o2}_S} \text{ and } B_\theta = 3\sqrt{\sigma^{o2}_\theta}.$$