Harmonics Virtual Lights: Fast Projection of Luminance Field on Spherical Harmonics for Efficient Rendering

Pierre Mézières, François Desrichard, David Vanderhaeghe, Mathias Paulin

Volume xx (200y), Number z, pp. 1–13

Harmonics Virtual Lights: fast projection of luminance field on spherical harmonics for efficient rendering

P. Mézières, F. Desrichard, D. Vanderhaeghe, M. Paulin

IRIT, Université de Toulouse, CNRS, INP, UT3, Toulouse, France

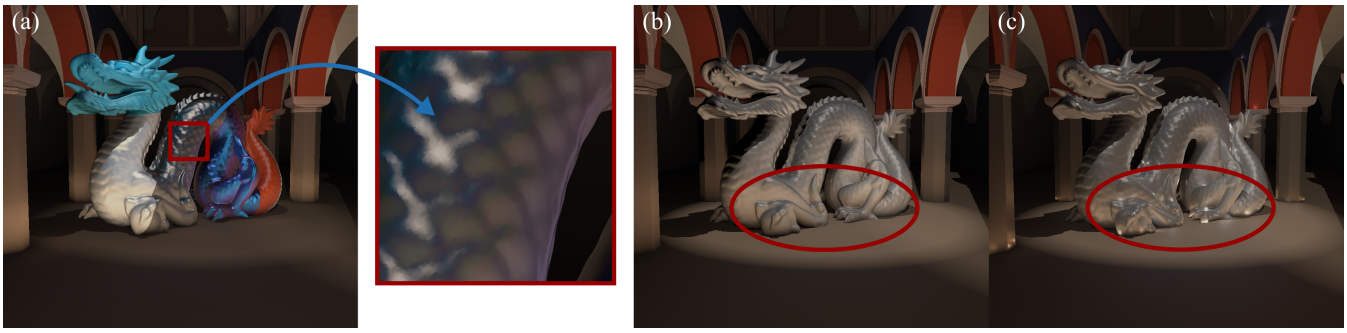

Figure 1: *(a) Harmonics Virtual Lights (HVLs) can render global illumination with arbitrary BRDF at no extra cost, while for example VPLs have an additional cost to use numerical material. (b) HVLs do not produce the classic artifacts of (c) the Virtual Point Lights methods.*

Abstract

In this paper, we introduce Harmonics Virtual Lights (HVL), to model indirect light sources for interactive global illumination of dynamic 3D scenes. Virtual Point Lights (VPL) are an efficient approach to define indirect light sources and to evaluate the resulting indirect lighting. Nonetheless, VPL suffer from disturbing artifacts, especially with high-frequency materials. Virtual Spherical Lights (VSL) avoid these artifacts by considering spheres instead of points but estimates the lighting integral using Monte-Carlo which results to noise in the final image. We define HVL as an extension of VSL in a Spherical Harmonics (SH) framework, defining a closed form of the lighting integral evaluation. We propose an efficient SH projection of spherical lights contribution faster than existing methods. Computing the outgoing luminance requires $\mathcal{O}(n)$ operations when using materials with circular symmetric lobes, and $\mathcal{O}(n^2)$ operations for the general case, where n is the number of SH bands. HVL can be used with either parametric or measured BRDF without extra cost and offers control over rendering time and image quality, by either decreasing or increasing the band limit used for SH projection. Our approach is particularly well-designed to render medium-frequency one-bounce global illumination with arbitrary BRDF at interactive frame rate.

Keywords: spherical harmonics, global illumination, real-time

CCS Concepts

• **Computing methodologies** → **Rendering; Rasterization;** *Real-time simulation;*

1. Introduction

Fast and high-quality rendering of 3D environments has numerous applications: real-time rendering allows seamless interactions in video games and virtual reality applications, while interactive rendering is used by content creators to assess the look of a scene in seconds, and progressively refine it before moving on to offline rendering. These applications require very detailed 3D geometry and displays realistic materials including high specularity and advanced effects such as anisotropy, represented by a Bidirectional Reflectance Distribution Function (BRDF).

Efficient rendering of such scenes often relies on lighting algorithms based on virtual lights, which compute the final picture in two steps. In the first step, light propagation is performed from the scene light sources, *i.e.* the primary lights, and yields Virtual Point

Lights (VPLs) distributed on the surface of the scene, *i.e.* the secondary lights. Afterwards, the estimation of global lighting at each point of the scene is reduced to the simpler problem of direct lighting from the secondary lights, which can be handled using rasterization.

Many-lights methods allow managing a big number of secondary lights in real-time thus resulting in an efficient rendering algorithm that simulates realistic light transport with inter-reflections. The key point of these families of algorithms concern the way virtual lights are distributed into the scene and how their properties are represented so that the cost of direct lighting remains very low.

VPLs provide unbiased results at interactive frame rates, but the resulting lighting presents several limitations. Millions of VPLs must be distributed inside the scene to capture detailed inter-reflections. At each pixel of the image, efficient gathering of secondary lights contributions becomes a computational challenge that must be tackled for these approaches to handle detailed 3D scene. VPLs exhibit several artifacts. The main reasons are the unbounded contributions of the VPLs, clearly visible in the vicinity of the VPLs positions, and emphasized when the VPLs count is low. These artifacts appear on glossy and specular surfaces. A common workaround is to limit the materials' glossiness of the scene, hence limiting the possible scenes. Virtual Spherical Lights (VSLs) consider spherical secondary lights to avoid VPLs singularities, and thus limits the spikes' contribution. However, VSLs evaluation relies on Monte-Carlo integration, computationally too expensive for interactive to real-time rendering.

We propose a solution compatible with interactive scenarios that addresses these issues while reducing the computational cost of virtual lights. We propose to craft a finer representation of secondary lights from spherical harmonics (SH) which we call *Harmonics Virtual Lights* (HVLs). Spherical harmonics represent spherical functions in the frequency domain, allowing a band limited approximation by taking only the first harmonics coefficients.

Our main contributions are:

- An efficient projection of spherical lights contributions to SH.
- The harmonics virtual light (HVL) model, that imposes no limitation on the materials.
- Two heuristics to efficiently estimate the HVLs radii.

As previous many-lights approaches, HVLs follow a two steps pipeline: The first step distributes secondary lights onto objects surfaces from each primary lights; the second step evaluates secondary lights and combines their contributions to obtain the final image (Sec. 3). HVL evaluation is based on Spherical Harmonics (SH) to solve the rendering equation (Sec. 4). Our formulation benefits from Zonal Harmonics (ZH) efficient projection of spherical lights on the SH basis (Sec. 4.2). Thanks to this projection, our spherical harmonics lighting framework handles hundreds to thousands of lights, with a full spherical harmonics convolution in $\mathcal{O}(n^2)$ where n is the number of SH bands.

Our approach uses reflective shadow maps [DS05] to distribute HVLs in the 3D scene while their contributions are computed using SH (Sec. 5). We propose two heuristics to manage the HVLs radii as they have a crucial role in the final image quality (Sec. 5.3). Finally, we analyze the results produced by HVLs and compare to state of the art VPL/VSL/VSGL (Sec. 6). When using circularly symmetric BRDFs, as we do when comparing to VSGL, we benefit from the HVLs projection on ZH (becoming, in this case, $\mathcal{O}(n)$) (Sec. 6.3). HVLs handle any kind of materials, measured or analytic, and exhibit fewer artifacts than VPLs while being computationally more efficient than VSLs.

2. Related Work

Spherical harmonics lighting. Ramamoorthi and Hanrahan [RH01] propose to use SH to efficiently compute and store illuminance environment maps; Sloan *et al.* [SKS02] extend this principle to store complex luminous transfer functions, and Kautz *et al.* [KSS02] further develop the method to handle arbitrary BRDFs. In order to remain computationally efficient, these methods rely on fixed scene parameters, such as the view point, the lights parameters or the geometry of the scene. To mitigate this last limitation, and to be able to render dynamic scenes in real-time, all these methods compute SH shading only on mesh vertices and linearly interpolate the resulting color at each shading point. The fixed light constraint could also be removed when light sources are modeled as SH-projected environment map. In this case, instead of computing the SH projection on the fly, it can be obtained by rotating the SH coefficients of the environment map. This enforces the need of an efficient rotation algorithm for SH coefficients. Even with effective derivations to apply the rotation as proposed by Nowrouzezahrai *et al.* [NSF12], heavy computation is necessary on high bands. In our approach, we avoid this costly step by integrating the rotation into the SH projection [SLS05].

Wang and Ramamoorthi [WR18] extend the principle of rotation proposed by Nowrouzezahrai *et al.* [NSF12] to efficiently project polygonal domains on SH. They reduce the integration of a polygonal light to the integration of its contours; in turn, contour integration fits the integration of ZH, allowing real-time performances. Wu *et al.* [WCZR20] extends this by computing the SH coefficients and their gradients, in the case of polygonal sources, on a 3D grid, thus allowing Hermite interpolation of the SH coefficients at each shaded point instead of computing them explicitly. We propose a simpler mathematical framework where integration on ZH is straightforward (Sec. 4.2) when dealing with spherical lights.

Virtual lights. We refer the reader to recent states of the art [RDGK12, DKH*14] for a general introduction on point-based global illumination and many-lights rendering. We focus here on previous work directly related to our contributions. The seminal work of Keller [Kel97] introduces the concept of point-based global illumination. The main idea is to approximate global illumination from its discretization on the surface of the scene using virtual lights. Virtual point lights (VPLs) are one of the most common ways to represent those virtual lights. Reflective shadow maps (RSM) [DS05] rely on rasterization to efficiently distribute the VPLs on the scene, placing them at the first bounce from the light. An alternative on recent hardware is to dispatch virtual lights with ray tracing, as proposed by Lin and Yuksel [LY19].

VPLs suffer from two main artifacts. First, if there are not enough virtual lights, flickering will appear in dynamic scenes. Second, and more troublesome, spike artifacts could appear as lighting

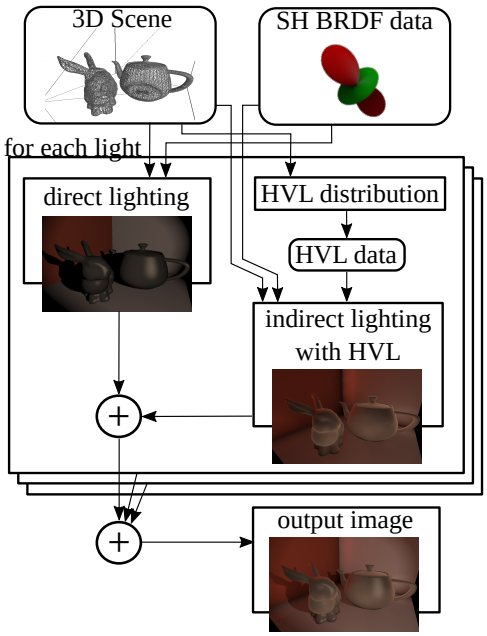

Figure 2: *Overview of the HVLs rendering pipeline. The inputs are a 3D scene with dynamic objects and lights, along with precomputed BRDF spherical harmonics coefficients. For each primary light of the scene, direct lighting is computed using SH framework. Indirect lighting starts with the distribution of HVLs from primary lights, followed by the gathering of HVLs contributions at each pixel. This pipeline allows using parametric like measured BRDF at the same cost.*

evaluation depends on the inverse of the squared distance to the VPL. When the shaded point is close to the VPL, this term tends towards infinity and cause strong light spikes in the image. This problem occurs also in the presence of glossy materials: when the view aligns with the reflection of one VPL and the VPLs density is not sufficient, the reflection of the VPL is visible. The naive way to overcome this problem is to clamp the VPL contribution, and to lessen the specular appearance of glossy materials. Hašan *et al.* [HKWB09] introduced virtual spherical lights (VSL) to robustly address this problem. To do so, they extend the integration domain from a point to a disk centered at the virtual light position. The contribution of light is averaged over the disk, drastically diminishing artifacts. Alternatively, Tokuyoshi [Tok15b] proposes virtual spherical gaussian lights (VSGL) to approximate the contribution of a set of VPLs, smoothing lighting variation. In contrast of VSGL, we consider lights individually; our idea is to distribute spherical light sources (HVL) instead of point light sources. Each HVL accounts for more information than simple VPL, in the spirit of Rich-VPL [SHD15].

3. HVL Lighting Pipeline Overview

Our approach being orthogonal to direct lighting computation, we focus on indirect lighting in the following. The HVLs rendering pipeline (Fig. 2) is built over the SH projection of the BRDFs as unified material representation. This representation is precomputed once for all for each material. Our two passes indirect lighting pipeline is similar to previous many-lights approaches. For each frame, the first pass distributes HVLs according to primary lights; it stores the SH coefficients of the reflected luminance field of each HVL in an HVLs cache. The second pass gathers the contributions of the HVLs from the cache by computing the convolution between light emitted by HVLs and the material at the shaded point location. The light emitted by HVL is the result of the reconstruction of the HVL reflected luminance field in the direction of the shaded point. Thus, HVLs are anisotropic spherical lights whose emission intensity depends on the direction ω_j, the incident direction on the receiving shaded point. Fig. 3 shows a typical one-bounce light path evaluated by our technique. The final image is then computed as the sum of direct and indirect contributions from each primary lights.

We formulate the indirect lighting $L_o(x,\omega_o)$, leaving the shaded point x in the direction ω_o as

$$L_o(x,\omega_o) = \sum_{s\in S}\Big(\sum_{j\in H_s} L_{j,o}(x,\omega_o,\omega_j)\Big) \tag{1}$$

where S is the set of primary lights, H_s the set of HVLs generated from light s, ω_j the direction from x to the HVL j and $L_{j,o}(x,\omega_o,\omega_j)$ its luminance contribution. According to the rendering equation [Kaj86], this contribution is expressed as

$$L_{j,o}(x,\omega_o,\omega_j) = \int_\Omega L_j(x,\omega_i)\,F(x,\omega_i,\omega_o)\,\mathrm{d}\omega_i \tag{2}$$

where $F = \cos(\theta_i) f_s(x,\omega_i,\omega_o)$ is the reflectance, f_s being the BRDF, and $L_j(x,\omega_i)$ is the incoming luminance at x emitted by the HVL j. As this luminance is only perceived in the solid angle subtended by the HVL, the integral domain is reduced to this solid angle Q_j. We approximate this equation by considering that the HVL emission is constant over the solid angle.

$$L_{j,o}(x,\omega_o,\omega_j) \approx L_j(x,\omega_j)\int_{Q_j} F(x,\omega_i,\omega_o)\,\mathrm{d}\omega_i \tag{3}$$

This approximation allows us to propose a computationally efficient projection of spherical lights contributions to SH (Sec. 4.2) and to quickly compute lighting contribution for each HVL (Sec. 5.1).

We rely on the number of bands for spherical harmonics evaluation as a trade-off parameter between accuracy and speed. To reach interactive performance, we typically use 0-5 bands to evaluate HVLs emission and 0-10 bands for the SH convolution on shaded points. This allows to represent low-frequency luminance field for HVLs and medium-frequency BRDF for the shaded point. Our approach scales well regarding high band limit and is able to render thousands of HVLs on 0-20 bands in a few seconds.

4. Lighting with Spherical Harmonics

The computational challenge in Eq. 2 and 3 is the evaluation of the integral term, which is a motivation for using the SH framework. Considering Eq. 2, when computing $L_{j,o}(x,\omega_o)$ and fixing both x and ω_o, we evaluate the integral of a product of two spherical functions of ω_i. Using spherical harmonics, this integral reduces to a dot product between vectors of coefficients dependent on x and ω_o. For conciseness, we omit x and ω_o in this section.

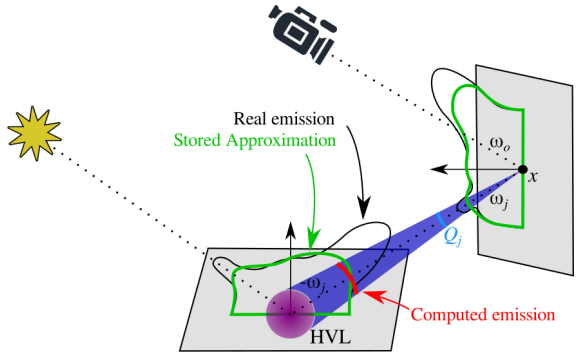

Figure 3: *Indirect lighting evaluation for one shaded point and one HVL. The HVL, in purple, uses a low-frequency approximation of the light emission corresponding to the first bounce of light emitted by the primary light. It's contribution to the shaded point at x is evaluated by projecting the HVL luminance onto SH and then convoluting with the BRDF at the shaded point's location. The HVL emission is computed as constant over its subtended solid angle.*

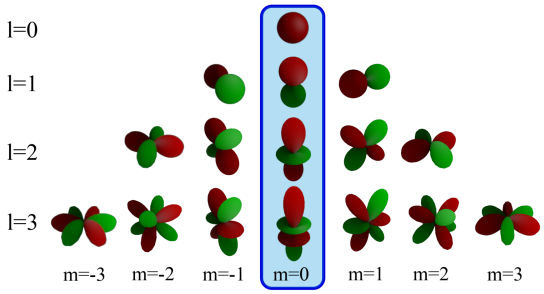

Figure 4: *Representation of the first spherical harmonics base functions. Lobes are colored according to their sign, with positive values in red and negative in green. The sub-basis of zonal harmonics for which $m = 0$ is outlined in the middle.*

4.1. Spherical Harmonics Review

Let's rewrite Eq. 2 in terms of spherical harmonics. We note $\mathbf{L}$ and $\mathbf{F}$ the projection of L and F on the real spherical harmonics basis. The dot product $\mathbf{L} \cdot \mathbf{F}$ is defined by:

$$\int_\Omega L(\omega_i)\, F(\omega_i)\, \mathrm{d}\omega_i = \mathbf{L} \cdot \mathbf{F} \tag{4}$$

To obtain the coefficients $\mathbf{L}_l^m$ (resp. $\mathbf{F}_l^m$) of the vector $\mathbf{L}$ (resp. $\mathbf{F}$), we project the original functions on the SH basis $\{Y_l^m(\omega)\}$ which is indexed by order l and degree m. As shown in Fig. 4, successive orders $l \geq 0$ correspond to an increasing of the frequency bands, in which degrees span $-l \leq m \leq l$. Projection to and reconstruction from SH of the spherical function $S(\omega)$ (*i.e.* $\mathbf{L}$ or $\mathbf{F}$) obey the following equations:

$$\mathbf{S}_l^m = \int_\Omega S(\omega)\, Y_l^m(\omega)\, \mathrm{d}\omega \tag{5}$$

$$S(\omega) = \sum_{l=0}^{+\infty} \sum_{m=-l}^{l} \mathbf{S}_l^m\, Y_l^m(\omega) \approx \sum_{l=0}^{N} \sum_{m=-l}^{l} \mathbf{S}_l^m\, Y_l^m(\omega) \tag{6}$$

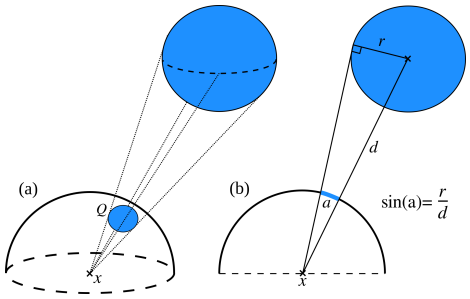

Figure 5: *Geometric setup of an HVL luminance contribution. (a) The HVL in blue projects to a spherical cap Q on the unit sphere centered at the shading point x. (b) 2D cut showing the HVL radius r, the distance d with the HVL, and the half-angle a of the cap.*

In practice, SH coefficients are bands limited to $l = N$ and the reconstruction becomes an approximation with fixed maximum frequency. In this case, the integral of Eq. 2 reduces to a sum of $(N+1)^2$ terms, allowing for an efficient evaluation of the equation.

Using spherical coordinates, real SH Y_l^m are expressed as

$$Y_l^m(\theta,\phi) = \begin{cases} \sqrt{2} K_l^{|m|}\, P_l^{|m|}(\cos\theta)\, \sin(|m|\phi) & m < 0 \\ K_l^0\, P_l^0(\cos\theta) & m = 0 \\ \sqrt{2} K_l^m\, P_l^m(\cos\theta)\, \cos(m\phi) & m > 0 \end{cases} \tag{7}$$

where P_l^m denotes the associated Legendre polynomial, and K_l^m is a normalizing factor:

$$K_l^m = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} \tag{8}$$

4.2. Computing Luminance and Material Coefficients

To compute the vector $\mathbf{L}$ at a shaded point x, we need to project the incoming luminance field $L(\omega_i)$ on the SH basis. As stated by Eq. 3 we consider the luminance emitted constant over the solid angle subtended by the light source.

SH projection of simple light primitive The luminance contribution of a simple light source, such as point light, spot, or directional light, corresponds to a Dirac distribution. Thus, the computation of the projection amounts to evaluate the SH in the Dirac direction and scaling by the light source's intensity.

Efficient SH projection of spherical light The projection of a spherical light is more expensive. Our framework aims at handling thousands of them, since HVL secondary lights correspond to spherical lights. Zonal Harmonics offers an efficient projection and reconstruction algorithm for spherical lights.

For the hemisphere around a shaded point, the solid angle Q subtended by the light source is a spherical cap (Fig. 5). Considering the incident luminance as constant, and equal to 1, over Q and 0 elsewhere, the projection on SH of the incident luminance function simplifies to:

$$\mathbf{L}_l^m = \int_Q Y_l^m(\omega)\, \mathrm{d}\omega \tag{9}$$

Let ω_{light} be the direction from the shaded point to the center of the spherical light. Q induces a circularly symmetric luminance field around ω_{light}. Considering the shading frame with ω_{light} as z-axis, the computation of Zonal Harmonics boils down to the computation of $\tilde{\mathbf{L}}_l$ coefficients, since the other coefficients are null.

$$\tilde{\mathbf{L}}_l = K_l \int_Q P_l(\omega_{light} \cdot \omega)\, d\omega \tag{10}$$

where $K_l = K_l^0$ and $P_l = P_l^0$. Using spherical coordinates, this equation becomes

$$\tilde{\mathbf{L}}_l = K_l \int_{\theta=0}^{a} \int_{\phi=0}^{2\pi} P_l(\cos(\theta)) \sin(\theta)\, d\phi\, d\theta \tag{11}$$

where a is the half-angle subtended by the spherical light source (Fig. 5). Integrating on ϕ, we obtain

$$\tilde{\mathbf{L}}_l = 2\pi K_l \int_{\theta=0}^{a} P_l(\cos(\theta)) \sin(\theta)\, d\theta \tag{12}$$

As detailed in Appendix A, the value of this integral is

$$\int_{\theta=0}^{a} P_l(\cos(\theta)) \sin(\theta)\, d\theta = \frac{-P_{l+1}(\alpha) + P_{l-1}(\alpha)}{2l+1} \tag{13}$$

where $\alpha = \cos(a)$. By re-injecting this into Eq. 12, the computation of the zonal harmonics coefficients for a spherical light in the coordinates frame aligned with the emission direction only requires evaluating Legendre polynomials and by simplifying, we obtain

$$\tilde{\mathbf{L}}_l = \begin{cases} \sqrt{\frac{\pi}{2l+1}} (P_{l-1}(\alpha) - P_{l+1}(\alpha)) & \text{if } l \neq 0 \\ \sqrt{\pi}(1-\alpha) & \text{otherwise} \end{cases} \tag{14}$$

The SH coefficients are then obtained by rotating the ZH coefficients as proposed by Sloan *et al.* [SLS05]:

$$\mathbf{L}_l^m = \sqrt{\frac{4\pi}{2l+1}} Y_l^m(\omega_{light}) \tilde{\mathbf{L}}_l \tag{15}$$

As shown Fig. 6, beyond 5 SH bands, computing the ZH coefficients $\tilde{\mathbf{L}}_l$ using the closed form derived by Sloan [Slo08] exhibits more arithmetic operations than our general recurrence (Eq. 14). To evaluate the Y_l^m basis function, we use the work of Sloan [Slo13].

BRDF coefficients To compute BRDF coefficients, we use the approach proposed by Kautz *et al.* [KSS02]. A BRDF is a function of the two directions ω_i and ω_o, yielding a 4-dimensional function. When evaluating F, the direction ω_o is fixed; hence the BRDF evaluation corresponds to a 2-dimensional function projected on SH.

In a pre-processing step, we tabulate SH coefficients according to discrete samples of the direction $\omega_o = (\theta_o, \phi_o)$.

$$\text{Tab}[\theta_o, \phi_o] = \mathbf{F}_{\omega_o} \tag{16}$$

where $\mathbf{F}_{\omega_o}$ is the SH coefficient vector of $\mathbf{F}$ computed for the output direction ω_o. We created a database of materials projected on spherical harmonics from the MERL database [MPBM03] and the database produced by Dupuy and Jakob [DJ18] with a sampling step of one degree for θ and ϕ. With such uniform sampling, a material has 90×360 samples of $\mathbf{F}_{\omega_o}$ and accessing the coefficients requires a constant time array lookup. The one degree step allows capturing subtle material variations. Other sampling and reconstruction strategies might be used but are out of scope of this paper. Appendix C lists the materials we used for the results shown in this paper.

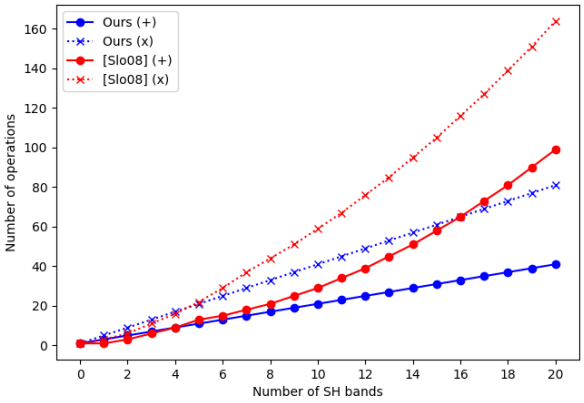

Figure 6: *Number of arithmetic operations needed to compute the vector of ZH coefficients up to a certain band. Our method (Eq. 14) is compared with the analytic Sloan's method [Slo08]. Number of operations calculated assumes that the recurrence loop is unrolled to avoid operations corresponding to constants for each ZH bands (*e.g. $\sqrt{\pi/(2l+1)}$ *counts as one multiplication for each band.)*

As an optimization compared to the approach of Kautz *et al.* [KSS02], we store isotropic material in a one-dimensional array. Isotropic materials depend on the $\Delta\phi = |\phi_i - \phi_o|$ rather than the actual values of ϕ_i and ϕ_o. Hence, by using the isotropic parametrization of F

$$F_{iso}(\theta_o, \theta_i, \Delta\phi) = F(\theta_o, 0, \theta_i, \Delta\phi) \tag{17}$$

we drop the explicit dependence on ϕ_o and only discretize θ_o. This way, the storage of isotropic material drops down to 90 samples.

$$\text{Tab}_{iso}[\theta_o] = \mathbf{F}_{\omega_o} \text{ with } \omega_o = (\theta_o, 0) \tag{18}$$

$\mathbf{F}$ is precomputed relative to its own frame, in order to apply the convolution, $\mathbf{L}$ is first expressed in the same frame by rotating ω_{light} (Eq. 15). This frame change avoids specific SH rotation, such as the *zxzxz* rotation [KSS02].

As state Eq. 2, $\mathbf{F}$ accounts for the cosine term, *i.e.* $\cos(\theta_i) f_s(x, \omega_i, \omega_o)$, HVL evaluation needs the projection of the BRDF only (Sec. 5.1). Hence, SH BRDF projection without cosine term, denoted $\mathbf{F}'$, is also precomputed and stored analogously.

Practical SH evaluation drops high frequencies due to band limit. Band limiting typically produces ringing artifacts. To mitigate these artifacts, SH coefficients are multiplied with a sinc windowing function [Slo08].

5. Harmonics Virtual Light (HVL)

We define a HVL as a spherical light. Light's emitted luminance is stored as SH coefficients whose contribution is efficiently evaluated using the projection defined Sec. 4.2. Our real-time experimentation handles only simple light primitives as primary lights. Then, their incident luminance field at HVL location is a dirac and HVL only need to store a reference to the projected BRDF, rather

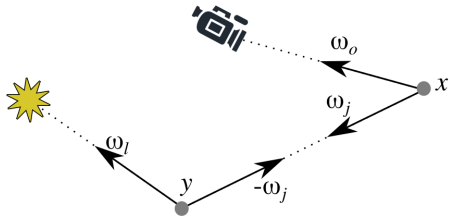

Figure 7: *Notations for a light path, y is an HVL and x a shaded point.*

than the SH coefficients of the luminance field. The HVL emission according to the outgoing direction corresponds to the referenced BRDF, reconstructed and scaled on the fly.

5.1. HVL Luminance Contribution

The luminance contribution of each HVL corresponds to the computation of Eq. 3. This equation is expressed as

$$L_{j,o}(x,\omega_o,\omega_j) \approx L_j(x,\omega_j)\mathbf{L}\cdot\mathbf{F}_{\omega_o,x} \tag{19}$$

where $\mathbf{L}$ corresponds to the coefficients vector of Eq. 15. The incoming luminance at x emitted by the j-th HVL is defined as

$$L_j(x,\omega_j) = \Phi_j(\mathbf{F}'_{\omega_l,y}\cdot Y(-\omega_j))G_j(-\omega_j) \tag{20}$$

where y and ω_l are respectively defined as the position of the HVL and its direction to primary light (Fig. 7). Φ_j is the power of the HVL, corresponding to the power of the primary light divided by the number of HVL. The dot product corresponds to the reconstruction of the emission function towards the shaded point (Eq. 6). Isotropic HVL reconstruction only use band 0 coefficient and skip SH evaluation.

The geometric factor $G_j(-\omega_j)$ corrects the energy taking into account the HVL light source spherical shape:

$$G_j(-\omega_j) = \frac{1}{\pi r_j^2}\max(0, n_j\cdot -\omega_j)H(n_x,\omega_j) \tag{21}$$

where n_j is the normal of the j-th HVL, r_j its radius, πr_j^2 is an approximation of the surface area. The dot product corrects the energy according to the surface orientation. The factor H represents the proportion of the HVL that lies in the shading hemisphere, it corresponds to the intersection of two spherical caps where an approximation is given by Oat [OS07]. We derive the equation to our case, where the spherical cap is the hemisphere oriented by n_x, the normal at point x:

$$H(n_x,\omega_j) = S\left(\frac{\bar{a}-\min(\max(\cos^{-1}(n_x\cdot\omega_j),\underline{a}),\bar{a})}{2a}\right) \tag{22}$$

where a is the half-angle subtended by the spherical light source, $\bar{a} = a+\frac{\pi}{2}$, $\underline{a} = a-\frac{\pi}{2}$ and $S(x) = 3x^2-2x^3$ is the Smoothstep function. H could also be computed with the exact equation given in [OS07]. Since Monte-Carlo integration explicitly evaluates the clamped cosines, the VSL approach does not need to use such a corrective term.

When evaluating the HVL emission (Eq. 20), we use the BRDF projection $\mathbf{F}'$, without the cosine term $\cos(\theta_i)$ which is integrated into the geometric factor (Eq. 21). This ensures that, when using low-frequency reconstruction (*i.e.* a few SH bands), the cosine is not over-smoothed and then, the energy is preserved.

The visibility term, although not present in the equations for the sake of clarity, is approximated by the visibility of the center of the HVL, as the VSL approach does.

5.2. Distribution and Caching

We use a cache to store each HVL data: emission function, position, normal, BRDF index. To fill the cache, our implementation follow the principle of Reflective Shadow Maps [DS05]: the scene is rendered from each primary light and each rendered fragment stores HVL data in a set of 2D GPU textures. The BDRF index corresponds to the index in the precomputed material SH coefficients set. This set is stored as 2D textures, one per material.

The memory requirement of the HVL cache spans from 16KB for 400 HVLs to 400KB for 10 thousands HVLs. We thus focus the discussion on the storage cost for materials. For an isotropic material, 90 directions are sampled; using 10 harmonic bands and 32-bit float RGB coefficients, one material represents 108kB. For instance, the required memory to use 200 different materials is 21MB. On the other hand, in the presence of anisotropic materials, the storage for one material is 360 times larger and raise to 388MB for one material.

5.3. Radius Heuristics

The radius of an HVL is an important parameter that impacts the rendered image quality. Intuitively, if HVLs do not sufficiently cover the scene, the rendered image exhibits the same kind of artifacts as VPLs. But if HVL radius is too large, high frequency shading details will be lost as the HVLs contributions will overlap. Hence, we have to adapt HVLs radius so that there is the right amount of overlapping.

The HVL overlapping is directly linked to the HVL radius with regard to HVL density. As done in VSL [HKWB09], we define each HVL radius based on an estimation of local HVL density. Instead of searching for the closest neighbors, as proposed for VSL, we propose two dynamic heuristics relying on HVL cache sampling to gather the HVL locations. Our first heuristic estimates the radius of an HVL from an approximation of the HVLs local density by sampling the HVL's 8-connected neighbors from the HVL cache:

$$r_1 = \frac{\sum_n w_n\|p-p_n\|}{\sum_n w_n}, \quad w_n = \frac{1}{|d-d_n|+\varepsilon} \tag{23}$$

where r_1 is the radius of the HVL, d the HVL distance to its related primary light, p denotes the position in world space coordinates and $\sum_n$ sums over the 8-connected neighbors of the HVL, p_n and d_n being neighbor's position and distance. This heuristic averages the distance of a HVL to its neighbors so that it overlap them and thus avoid spike artifacts. We weight the contribution of neighbors by the difference of distances from HVLs to light to avoid too large overlapping because of the scene geometry, such as the hole formed by the rabbit's ears in Fig. 8. A small ε value is added to avoid a division by 0, should two HVLs distances to light be the same.

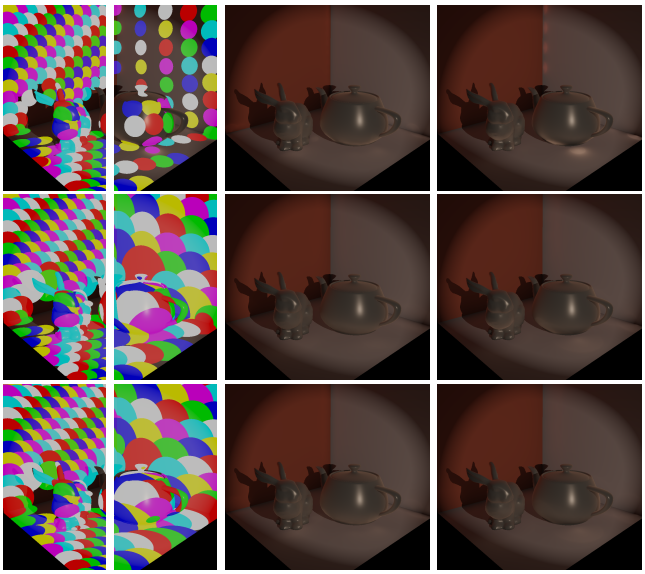

Figure 8: *HVLs coverage. Each colored circle corresponds to one HVL. From left to right: HVLs coverage with 400 and 100 HVLs; resulting image with 400 HVLs and then 100 HVLs. From top to bottom: fixed radius of 0.25, heuristic 1, heuristic 2 respectively.*

This first heuristic is based on the neighborhood of the HVL, making it expensive to compute. We propose a second heuristic, more suitable for real-time scenarios, independent of the HVL neighborhood by considering the FOV angle λ used to generate a square RSM instead. First, we approximate the angle γ formed by two diagonally adjacent HVLs by considering that the two HVLs are at the same distance of the light.

$$\gamma = \sqrt{2}\frac{\lambda}{\sqrt{M}} \tag{24}$$

where M is the total number of HVLs and $\sqrt{2}$ is the length of the diagonal. Starting from this, we approximate the distance between those HVLs, and hence the radius r_2 of an HVL, by considering a right triangle between the primary light and the two HVLs, where the right angle is at the HVL whose radius is sought.

$$r_2 = d\tan(\gamma) \tag{25}$$

where d is the distance of the light of the HVL and k is the same user-specified constant as for r_1. To avoid any trigonometric function evaluation, as γ is close from 0, we approximate the tan function by its Taylor series $\tan(x) \approx x + x^3/3$. The heuristic then becomes

$$r_2 \approx d\left(\gamma + \frac{\gamma^3}{3}\right) \tag{26}$$

The approximation of Eq. 25 by Eq. 26 is accurate. Indeed, in order to have an error of less than 0.01 for the evaluation of the function tan from its Taylor series, calculus hints that we must have $\gamma \lesssim 0.58$ and thus $\lambda/\sqrt{M} \lesssim 0.2$. So, for a spotlight of half-angle 45°, $\sqrt{M} \gtrsim 3.92$, means that we need at least 16 HVLs for precise evaluation of the tan function. For a spotlight of half-angle 90°, we need at least 64 HVLs. These requirements are far below the number of HVLs required in practice (*i.e.* at least a few hundreds). As shown in Fig. 8, these heuristics allow having HVLs that overlap well, and thus covers the directly illuminated scene geometry, without having one HVL that swallow another one, which would reduce the impact of the number of HVLs. However, the balance remains fragile and artifacts may still appear, especially when the depth distribution of HVLs is heterogeneous because holes may appear in the coverage. To this end, and as in VSL, a user specified factor should be used to scale the computed radii. In our experiments, we obtained good results by setting this factor between 1 and 2.

Figure 9: *Global illumination using HVLs with different measured materials. In our framework, materials can be changed dynamically without affecting rendering time. In the bottom row, only the materials applied to the chairs and the table are modified between the three pictures.*

Scene	HVL number	Convolution band limit			
		3	5	7	9
Teaser Fig. 1(a)	64	6.6	7.8	10.3	18.6
	196	19.8	23.5	30.4	56.1
	400	41.0	48.6	62.7	115.3

Table 1: *Timings (ms) for indirect lighting gathering from various HVLs numbers and SH bands for convolution on shaded points. The emission function for each HVL is computed with 3 bands.*

6. Results

We implemented our prototype using OpenGL / GLSL 4.5 on an Intel Xeon 2.10 GHz processor and a RTX 2080 graphics card. We did not take into account the HVLs visibility in our renderings.

6.1. Experimental results

Fig. 9 illustrates how materials change the rendered image through the use of HVLs. Our method applies a low-pass filter on the light-

ing due to the use of SH, thus providing smooth and artifact-free results (Fig. 10).

The HVL count has a linear impact on the computational cost, each HVL contribution being independent of the others. Increasing the SH band limit has a quadratic cost with respect to the band limit as the number of SH coefficients is equal to the square of the bands number. Table 1 confirms these trends. Thus, the computation time for HVLs is predictable, but is greatly impacted by the number of SH bands. Furthermore, if low-frequency BRDF are used, increasing the band limit will have no impact on the final picture quality. Using an adaptive band limit might keep the computational cost minimal but might also increase code divergence in the shader evaluation of the SH projection.

6.2. Comparison with VPL, VSL and path-traced reference

To fairly compare HVL, VPL and VSL, we only use microfacet based BRDF (Fig. 10 and 11). Indeed, if VPL/VSL computation time might be greatly impacted by the use of measured and SH encoded BRDF, our HVL method is agnostic to the used BRDF model and performs similarly on parametric or measured BRDF. In our comparison, we use the GGX microfacet distribution and evaluate the Fresnel reflectance using the exact term. We used material from MERL database, fitted on GGX by the work of Ribardière*et al.* [RBMS17]. The three methods are compared to path-traced references, rendered using pbrt-v3 [PJH16], using three metrics: RMSE, PSNR and SSIM. Table 2 shows timings and error metric for the three methods.

Fig. 10a, 10b and 10d are computed using the same budget of virtual lights. To avoid a visual comparison bias, virtual lights are placed at the exact same locations for all three methods. Our method avoids the classical VPLs artifacts but is more computationally expensive. Using 25 samples per virtual light, VSL is even more expensive while still exhibiting noise. Aiming at interactive rendering, VSLs must use low sample count and results on noisy pictures (Fig. 10e). To remove this noise, particularly visible on the bunny's feet, the number of samples must be at least tripled, with a similar increase on the computation time (Fig. 10d).

With the same computation time budget, VPLs still exhibit artifacts despite using more virtual lights, whereas HVLs produce a smoother but artifact-free result (Fig. 10a and 10c). VSLs produce a noisy result as only a low number of samples might be used (Fig. 10e). Thus, HVLs are an interesting alternative to produce a fast low-frequency global lighting and gradually raise the final image in frequency. Fig. 11a and 11b are computed given a time budget of 500ms. VPLs still exhibit artifacts despite the big number of virtual lights, while HVLs give a smooth result.

Light leaks artifacts may appear in some situations due to a combination of several factors (orange and red circles in Fig. 10 and 11). Inaccurate visibility estimation is one of these factors when using virtual lights. The red circles show a light leak due to this visibility problem for VPLs due to sources located on the background. These artifacts are not related to the VPLs themselves but to the limitations of our implementation. For HVL and VSL (orange circles), the spherical nature of the sources is the main cause of artifacts, due here to sources located near Suzanne. The spherical nature of the sources generate an overestimation of the actual lit area. This is especially problematic when using a few sources, since smaller is the number of sources, larger is their radius. Finally, the frequency band limit resulting from the SH order for HVL smooths out these artifacts.

As shown by these experiments, HVLs always give a smooth but biased result, whatever the number of virtual lights or the time budget given. This robustness of our method is its biggest advantage for interactive lighting design. Note that while VPLs are less appropriate for interactive usages due to the high-variance estimation of indirect lighting with small number of VPLs, they define an unbiased estimator of indirect lighting. Hence, running simulation using VPL for a long enough time will always give a better result than HVL.

6.3. Comparison with VSGL

We use our own implementation of a simplified version of VSGL [Tok15b], based on publicly available faster VSGL code [Tok15a]. This allows us to produce a fair comparison between both methods, but without importance or interleaved sampling. We setup a scene similar to the one used in the VSGL paper (Fig. 12) thanks to the data provided by the authors.Our method benefits from faster computation time thanks to the ZH expansion, thereby limiting convolution to materials with a circularly symmetric specular lobe which is fair compared to VSGL, being itself limited to BRDF with a single lobe. To summarize our implementation:

- The ZH projection for the light is done using our proposal (Eq. 14).
- The BRDF convolution uses the PRT framework proposed by Sloan *et al.* [SKS02] and adapted to our use case, *i.e.* where the luminance and specular lobe of the BRDF are both circularly symmetric functions:
  - The diffuse contribution is computed using the work of

Figures		Parameters			Timings		Error metrics		
		lights	VSL samp.	SH bands	distrib.*	indirect light	rmse	psnr	ssim
VPL	10b	400	-	-	0.59	13	0.054	25.20	0.977
	10c	1928	-	-	0.59	61	0.088	21.10	0.981
	11b	10k	-	-	**2.44**	500	0.040	27.91	0.984
VSL	10d	400	25	-	0.59	254	0.043	27.24	0.974
	10e	256	9	-	0.59	63	0.043	27.26	0.972
HVL	10a	400	-	5	0.38	61	0.039	28.01	0.980
	11a	1.5k	-	9	**1.6**	500	0.039	28.24	0.983

* cache size is 1024^2 pixels (**bold=2048^2**)

Table 2: *Timings (ms) and error metrics (path-tracing used as reference) detail for Fig. 10 and 11. RSM for HVLs are cheaper because fewer data are manipulated (see Sec. 5.2 for details).*

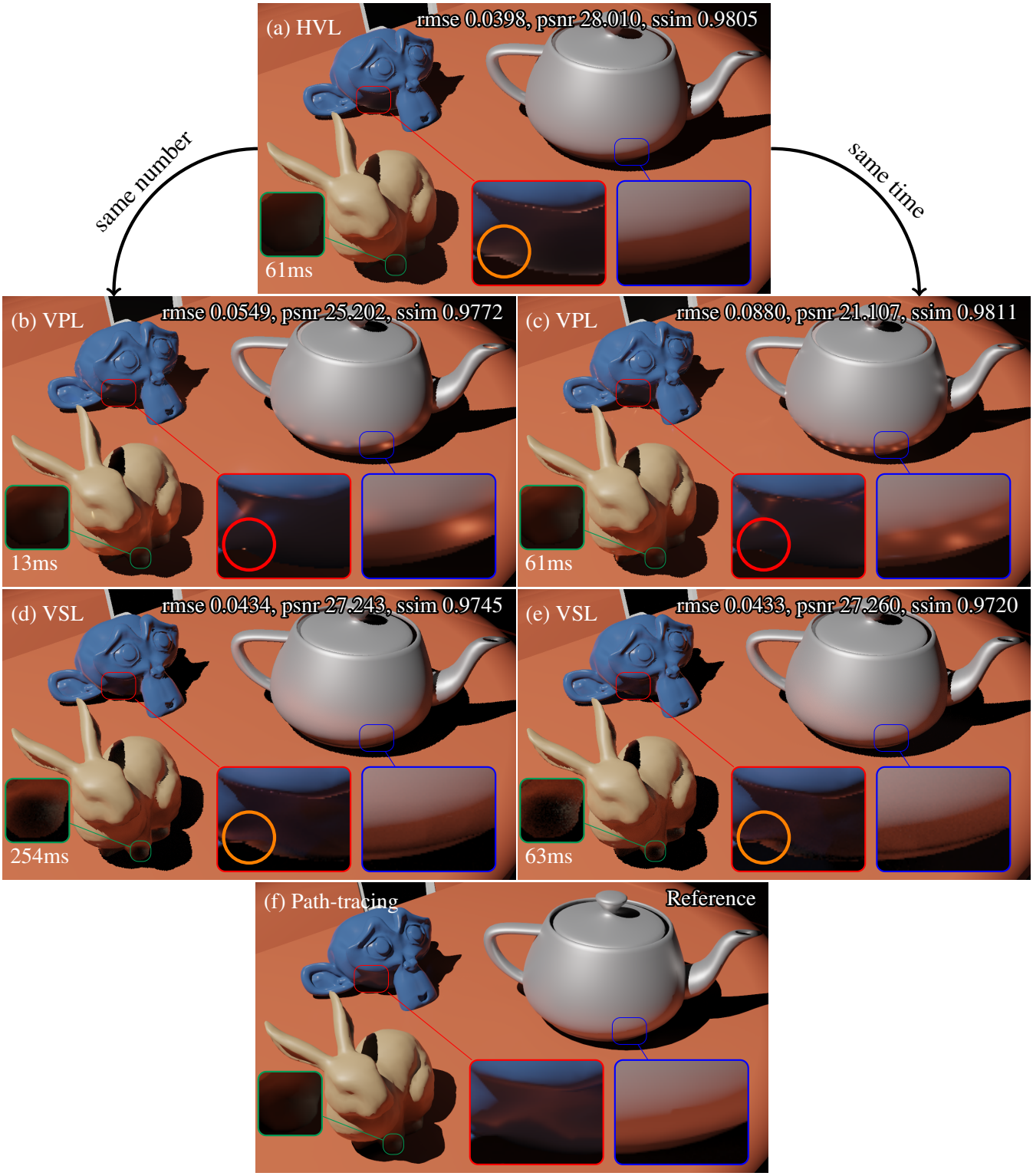

Figure 10: *Comparison between HVL, VPL, VSL and path-tracing. (a) uses 5 SH bands for convolution on shaded point, 3 bands for the emission function and 400 HVLs, (b) and (c) are resp. render with 400 and 1928 VPLs. (d) and (e) are render resp. with 400 and 256 VSLs and 25 and 9 importance samples of the shaded point's BRDF. HVLs and VSLs use heuristic r_2. (f) is a path-traced reference. Images are rendered at 1280x1024. The red circles show an artifact due to VPLs on the background and resulting from the lack of visibility management for secondary lights in our implementation. The orange circles highlight an overestimation of the indirect lighting due to the spherical nature of the HVL and VSL.*

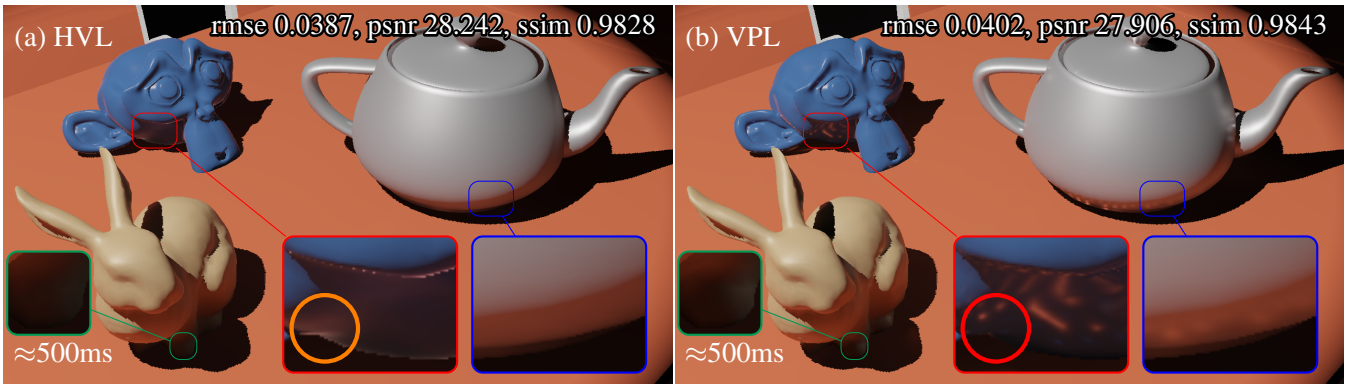

Figure 11: *Comparison of HVL and VPL at same time using same resolution and same heuristic for HVL as the figure above. (a) is rendered with 1500 HVLs and uses 9 SH bands for convolution and 3 SH bands for the emission of HVLs. (b) is rendered with 10000 VPLs.*

Ramamoorthi and Hanrahan adapted when using only ZH [RH01]. (Appendix B)
  - The specular contribution is computed using the convolution of two circularly symmetric function on ZH [DSJN19]. So, convolution complexity becomes $\mathcal{O}(n)$ instead of $\mathcal{O}(n^2)$, where n is the number of SH bands. The weakening factor $\cos\theta_i$ is taken out of the integral (Eq. 2) and calculated with the direction of the HVL center as input direction.
- The HVLs directional emission is computed by evaluating a GGX BRDF.

VSGL method approximates a set of nearby virtual light relying on RSM, as it is the most efficient method to quickly access a VPL and its neighbors, and taking advantage of RSM mipmaping. Our HVL method extends VSL and only relies on RSM to sample the position of the virtual lights. HVL are then less dependent on RSM, thus more flexible than VSGL. As shown in Fig. 12, HVL and VSGL exhibit similar results without any spikes.

For a fair comparison, both methods are restricted to single-lobe BRDF only. Our VSGL implementation defines one VSGL per pixel group; HVL are restricted to single-lobe BRDF when using the ZH acceleration method. Moreover, this ZH acceleration makes the HVL method much more competitive than the full SH method (40ms for 15 SH bands).

For the scene shown Fig. 12, to catch the strong specular reflection on the ground, we use 400 HVLs on 15 ZH bands. For the same time budget, we can have 529 VSGLs. On scenes with less specular effects, HVLs will require fewer bands, and therefore become cheaper to evaluate. This is shown in the supplemental materials, where the comparison between both methods is pushed a little further.

7. Limitations and Future Work

Primary light As shown in Sec. 5.1, our HVLs framework only considers primary light primitives from which incident direction is a Dirac. In order to manage area lights, the process of computing HVLs luminance contribution should be adapted. The SH coefficients of the incident luminance field due to the area light are to be stored on each HVL. Once the receiving shaded point is known, the SH coefficients for the emission of each HVL are computed according to the shaded point. The convolution of this emission function with the HVL incident luminance field will then result in the incident luminance field at the shaded point. Due to this additional convolution, the computation time of the method will increase.

HVLs visiblity The HVLs visibility could be approximated by taking into account only their center and using shadow maps (or imperfect shadow map [RGK*08]), as done in VSL or VSGL methods [HKWB09, Tok15b]. This approximation is only valid when HVLs radii are small. For large HVLs, visibility management is more complex. It might be interesting to project the visibility on SH, as done by Ren *et al.* [RWS*06], but this will require to evaluate an SH triple product ($\mathbf{V}\cdot\mathbf{L}\cdot\mathbf{F}$) instead of only a double product (Eq. 4). The extra cost of visibility projection and SH triple product would prevent to reach interactive rendering time. Indeed, unlike like the SH double product, the triple product does not reduce to a dot product of vectors.

Efficient shading Our prototype implementation only uses a simple light gathering to compute HVLs contributions at each pixel, in which each pixel gets the contribution of all HVLs. This has a strong impact on the computation time when using numerous HVLs. For a better efficiency, a shaded point must ideally consider only HVLs that will have an influence on itself. As our proposal is orthogonal to any virtual lights gathering method, Lightcuts [WFA*05], ManyLoDs [HREB11], or clustered shading method, could be implemented over the HVLs set. As shown for the VSGL method [Tok15b], the shading time can be reduced by using interleaved sampling [KH01]. HVLs are band limited by construction, thus, they do not generate arbitrary high frequencies. This property is particulary useful in interleaved sampling method to define a proper sampling strategy.

Flickering Techniques dealing with efficient shading such as RSM clustering [PKD12] allow reducing the flickering problem that might appear on dynamic scenes. In our implementation, we do not apply normal filtering, temporal coherency management nor any filtering method, thus flickering becomes visible. As HVLs inherit properties of VPLs, any method resolving flickering for VPLs

Figure 12: *Comparison with 900 VPL, 529 VSGL and 400 HVL, generated at equal time. Images are rendered at 1600×900. The HVL cache resolution is* 32^2 *pixels, and* 14^2 *pixels for one VSGL. The contribution of HVLs is on 15 SH bands. In this rendering, HVL are limited to circularly symmetric specular lobe to compute lighting only on ZH. Insets show VPL spike artifacts and VSGL or HVL robustness to those artifacts. See additional material for more comparisons and implementation details.*

could be adapted to HVLs, such as the temporally coherent sampling proposed by Barák *et al.* [BBH13].

Hemispherical bases Although spherical in shape, HVLs only emit light in a hemisphere. While the current implementation accounts for this using the geometric term, we thought about using the hemispherical basis proposed by Gautron *et al.* [GKPB07]. Nevertheless, this requires to correctly adapt several computations to this basis, such as the integration of Legendre polynomials or the efficient evaluation of harmonics. Unfortunately, the tricks proposed by Sloan [Slo13] to efficiently evaluate spherical harmonics do not apply directly to hemispherical harmonics. Alternatively, the derivations could be adapted to a more recent basis, such as the $\mathcal{H}$-basis of Habel *et al.* [HW10].

Soft angular kernels In our proposal, as for VSLs, we consider that the emission is constant on the solid angle subtended by the spherical light. If VSLs rely on this hypothesis for efficient uniform sampling for Monte-Carlo integration, we use this to derive an analytical solution for HVL (Eq. 9). However, the solid angle boundary corresponds to an infinite frequency in the emission function that produces ringing artifacts in the SH approximation. To limit these ringing artifacts, one can add a soft angular kernel in Eq. 12. However, finding an analytical and efficient solution with an arbitrary kernel is challenging but should improve the final result.

Anisotropic materials As presented in Sec. 4.2, we use an equiangular parametrization of the input direction to project materials on SH. If this memory efficient parametrization scales well for isotropic materials, the memory cost for anisotropic ones quickly become prohibitive (Sec. 5.2). Investigating the parabolic parametrization proposed by Kautz *et al.* [KSS02] should help to reduce this memory cost. Indeed, using their 128×128 parabolic map would reduce the cost of one anisotropic material on 10 SH bands from 388MB to 20MB. Doing so, it must be first demonstrated that the resolution will be sufficient to capture the full behavior of the material according to the number of SH bands.

8. Conclusion

We have presented harmonics virtual lights, a new model to capture and render indirect lighting in dynamic scenes with arbitrary materials. Our HVLs are spherical, secondary lights distributed on the surfaces of the scene after one bounce from the primary lights. When viewed from a shaded point, their circularly symmetric profile allows efficient projection of incoming luminance on the basis of zonal harmonics. Using a fast rotation formula, we align the incoming luminance field with any reflectance function encoded on spherical harmonics, and perform shading with realistic materials as a simple dot product. Thanks to our radius heuristics, each HVL radius is determined to cover the scene as well as possible, thus avoiding VPLs artifacts.

Our results show that, in comparison to virtual point lights, a much lower HVLs count can be used to obtain images of similar quality. Moreover, the rendered images do not suffer artifacts such as unbounded contributions around the virtual lights, or undesirable reflections on glossy materials. Our method is able to render one-bounce indirect illumination at interactive frame rates in fully dynamic scenes; the two hyperparameters, HVLs count and band limit, can be further tuned to achieve real-time rendering.

Acknowledgements

This project was partially funded by the CaLiTrOp project from French National Research Agency (ANR-16-CE33-0026).

Bunny and Dragon taken from the Stanford 3D Scanning Repository. Breakfast Room, Conference Room, Crytek Sponza, Dabrovic Sponza and Teapot taken from the McGuire Computer Graphics Archive. Suzanne taken from blender foundation.

References

[BBH13] BARÁK T., BITTNER J., HAVRAN V.: Temporally coherent adaptive sampling for imperfect shadow maps. *Comput. Graph. Forum 32*, 4 (2013), 87–96. doi:10.1111/cgf.12154. 11

[DJ18] DUPUY J., JAKOB W.: An adaptive parameterization for efficient material acquisition and rendering. In *SIGGRAPH Asia 2018 Tech. Pap. SIGGRAPH Asia 2018* (2018), vol. 37, ACM New York, NY, USA, pp. 1–14. doi:10.1145/3272127.3275059. 5, 13

[DKH*14] DACHSBACHER C., KŘIVÁNEK J., HAŠAN M., ARBREE A., WALTER B., NOVÁK J.: Scalable realistic rendering with many-light methods. *Comput. Graph. Forum 33*, 1 (2014), 88–104. doi:10.1111/cgf.12256. 2

[DS05] DACHSBACHER C., STAMMINGER M.: Reflective shadow maps. In *Proc. Symp. Interact. 3D Graph.* (2005), pp. 203–208. doi:10.1145/1053427.1053460. 2, 6

[DSJN19] DUBOUCHET A., SLOAN P.-P., JAROSZ W., NOWROUZEZAHRAI D.: Impulse Responses for Precomputing Light from Volumetric Media. doi:10/gf6rx8. 10

[GKPB07] GAUTRON P., KRIVANEK J., PATTANAIK S., BOUATOUCH K.: A novel hemispherical basis for accurate and efficient rendering. In *ACM SIGGRAPH 2007 Pap. - Int. Conf. Comput. Graph. Interact. Tech.* (2007), vol. 2004, pp. 321–330. doi:10.2312/EGWR/EGSR04/321-330. 11

[HKWB09] HAŠAN M., KŘIVÁNEK J., WALTER B., BALA K.: Virtual Spherical Lights for Many-Light Rendering of Glossy Scenes. *ACM Trans. Graph. 28*, 5 (2009), 1–6. doi:10.1145/1618452.1618489. 3, 6, 10

[HREB11] HOLLÄNDER M., RITSCHEL T., EISEMANN E., BOUBEKEUR T.: ManyLoDs: Parallel many-view level-of-detail selection for real-time global illumination. *Comput. Graph. Forum 30*, 4 (2011), 1233–1240. doi:10.1111/j.1467-8659.2011.01982.x. 10

[HW10] HABEL R., WIMMER M.: Efficient irradiance normal mapping. In *Proc. I3D 2010 ACM SIGGRAPH Symp. Interact. 3D Graph. Games* (2010), pp. 189–195. doi:10.1145/1730804.1730835. 11

[Kaj86] KAJIYA J. T.: The Rendering Equation. *Comput. Graph. 20*, 4 (aug 1986), 143–150. doi:10.1145/15886.15902. 3

[Kel97] KELLER A.: Instant radiosity. In *Proc. 24th Annu. Conf. Comput. Graph. Interact. Tech. SIGGRAPH 1997* (1997), pp. 49–56. doi:10.1145/258734.258769. 2

[KH01] KELLER A., HEIDRICH W.: Interleaved Sampling. In *Eurographics Work. Render. Tech.* 2001, pp. 269–276. doi:10.1007/978-3-7091-6242-2_25. 10

[KSS02] KAUTZ J., SLOAN P.-P. P., SNYDER J.: Fast, arbitrary BRDF shading for low-frequency lighting using spherical harmonics. In *Eurographics Work. Render.* (2002), vol. 2, pp. 291–296. 2, 5, 11

[LY19] LIN D., YUKSEL C.: Real-Time Rendering with Lighting Grid Hierarchy. *Proc. ACM Comput. Graph. Interact. Tech. 2*, 1 (2019), 1–17. doi:10.1145/3321361. 2

[MPBM03] MATUSIK W., PFISTER H., BRAND M., MCMILLAN L.: A data-driven reflectance model. In *ACM Trans. Graph.* (2003), vol. 22, pp. 759–769. doi:10.1145/882262.882343. 5, 13

[NSF12] NOWROUZEZAHRAI D., SIMARI P., FIUME E.: Sparse zonal harmonic factorization for efficient SH rotation. *ACM Trans. Graph. 31*, 3 (2012), 1–9. doi:10.1145/2167076.2167081. 2

[OS07] OAT C., SANDER P. V.: Ambient aperture lighting. In *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games* (New York, NY, USA, 2007), I3D '07, Association for Computing Machinery, p. 61–64. doi:10.1145/1230100.1230111. 6

[PJH16] PHARR M., JAKOB W., HUMPHREYS G.: *Physically Based Rendering: From Theory to Implementation (3rd ed.)*, 3rd ed. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, Nov. 2016. 8

[PKD12] PRUTKIN R., KAPLANYAN A., DACHSBACHER C.: Reflective Shadow Map Clustering for Real-Time Global Illumination. *Eurographics* (2012). doi:10.2312/conf/EG2012/short/009-012. 10

[RBMS17] RIBARDIÈRE M., BRINGIER B., MENEVEAUX D., SIMONOT L.: STD: Student's t-Distribution of Slopes for Microfacet Based BSDFs. *Comput. Graph. Forum 36*, 2 (2017), 421–429. doi:10.1111/cgf.13137. 8

[RDGK12] RITSCHEL T., DACHSBACHER C., GROSCH T., KAUTZ J.: The state of the art in interactive global illumination. *Comput. Graph. Forum 31*, 1 (2012), 160–188. doi:10.1111/j.1467-8659.2012.02093.x. 2

[RGK*08] RITSCHEL T., GROSCH T., KIM M. H., SEIDEL H. P., DACHSBACHER C., KAUTZ J.: Imperfect shadow maps for efficient computation of indirect illumination. In *ACM SIGGRAPH Asia 2008 Pap. SIGGRAPH Asia'08* (2008), vol. 27, ACM New York, NY, USA, pp. 1–8. doi:10.1145/1457515.1409082. 10

[RH01] RAMAMOORTHI R., HANRAHAN P.: An efficient representation for irradiance environment maps. In *Proc. 28th Annu. Conf. Comput. Graph. Interact. Tech. SIGGRAPH 2001* (2001), pp. 497–500. doi:10.1145/383259.383317. 2, 10, 13

[RWS*06] REN Z., WANG R., SNYDER J., ZHOU K., LIU X., SUN B., SLOAN P. P., BAO H., PENG Q., GUO B.: Real-time soft shadows in dynamic scenes using spherical harmonic exponentiation. In *ACM SIGGRAPH 2006 Pap. SIGGRAPH '06* (aug 2006), ACM, pp. 977–986. doi:10.1145/1179352.1141982. 10

[SHD15] SIMON F., HANIKA J., DACHSBACHER C.: Rich-VPLs for Improving the Versatility of Many-Light Methods. In *Comput. Graph. Forum* (2015), vol. 34, pp. 575–584. doi:10.1111/cgf.12585. 3

[SKS02] SLOAN P. P., KAUTZ J., SNYDER J.: Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. In *SIGGRAPH '02* (2002), pp. 527–536. doi:10.1145/566570.566612. 2, 8

[Slo08] SLOAN P.-P.: Stupid Spherical Harmonics (SH) Tricks. *Game Dev. Conf. 9* (2008), 320–321. 5

[Slo13] SLOAN P.-P.: Efficient Spherical Harmonic Evaluation. *J. Comput. Graph. Tech. 2*, 2 (sep 2013), 84–90. 5, 11

[SLS05] SLOAN P. P., LUNA B., SNYDER J.: Local, deformable precomputed radiance transfer. In *ACM Trans. Graph.* (2005), vol. 24, ACM New York, NY, USA, pp. 1216–1224. doi:10.1145/1073204.1073335. 2, 5

[Tok15a] TOKUYOSHI Y.: Fast Indirect Illumination Using Two Virtual Spherical Gaussian Lights. In *SIGGRAPH Asia 2015 Posters* (New York, NY, USA, 2015), SA '15, Association for Computing Machinery. doi:10.1145/2820926.2820929. 8

[Tok15b] TOKUYOSHI Y.: Virtual Spherical Gaussian Lights for Real-time Glossy Indirect Illumination. In *Comput. Graph. Forum* (2015), vol. 34, pp. 89–98. doi:10.1111/cgf.12748. 3, 8, 10

[WCZR20] WU L., CAI G., ZHAO S., RAMAMOORTHI R.: Analytic spherical harmonic gradients for real-time rendering with many polygonal area lights. *ACM Trans. Graph. 39*, 4 (2020). doi:10.1145/3386569.3392373. 2

[WFA*05] WALTER B., FERNANDEZ S., ARBREE A., BALA K., DONIKIAN M., GREENBERG D. P.: Lightcuts: A scalable approach to illumination. In *ACM Trans. Graph.* (2005), vol. 24, pp. 1098–1107. doi:10.1145/1073204.1073318. 10

[WR18] WANG J., RAMAMOORTHI R.: Analytic spherical harmonic coefficients for polygonal area lights. *ACM Trans. Graph. 37*, 4 (2018). doi:10.1145/3197517.3201291. 2

Appendix A: Legendre Polynomials Integration

We search for a closed form to the integral of Legendre polynomials over a spherical cap of angle a:

$$\int_{\theta=0}^{a} P_l(\cos(\theta))\sin(\theta)\,\mathrm{d}\theta$$

Substituting the variable $u=\cos(\theta)$ yields

$$\int_{\theta=0}^{a} P_l(\cos(\theta))\sin(\theta)\,\mathrm{d}\theta = \int_{u=\cos(a)}^{1} P_l(u)\,\mathrm{d}u$$

We get the primitive from the recursive formula

$$\int P_l(x)\,\mathrm{d}x = \frac{P_{l+1}(x)-P_{l-1}(x)}{2l+1}$$

and as $\forall l \geq 0,\ P_l(1)=1$ we obtain

$$\begin{aligned}\int_{\cos(a)}^{1} P_l(u)\,\mathrm{d}u &= \left[\frac{P_{l+1}(u)-P_{l-1}(u)}{2l+1}\right]_{\cos(a)}^{1} \\ &= \frac{-P_{l+1}(\cos(a))+P_{l-1}(\cos(a))}{2l+1}\end{aligned}$$

In summary, we have established that

$$\boxed{\int_{\theta=0}^{a} P_l(\cos(\theta))\sin(\theta)\,\mathrm{d}\theta = \frac{-P_{l+1}(\cos(a))+P_{l-1}(\cos(a))}{2l+1}}$$

Appendix B: Illuminance from circularly symmetric emission function

Ramamoorthi and Hanrahan use a 4×4 matrix to compute illuminance [RH01]. This matrix can be replaced by a 2×2 matrix when using a circularly symmetric emission function which can be projected on ZH, instead of SH, by orienting the projection according to the direction of emission function.

$$M = \begin{pmatrix} c_3\mathbf{L}_2^0 & c_2\mathbf{L}_1^0 \\ c_2\mathbf{L}_1^0 & c_4\mathbf{L}_0^0 - c_5\mathbf{L}_2^0 \end{pmatrix}$$

$$c_2 = 0.511664 \quad c_3 = 0.743125$$
$$c_4 = 0.886227 \quad c_5 = 0.247708$$

Illuminance can be evaluated according to the direction of the emission lobe ω_l. If we note $z=\omega_l\cdot\mathrm{n}$, illuminance is computed as

$$E(\mathrm{o}) = \mathrm{o}^t M \mathrm{o} \quad \text{with} \quad \mathrm{o}^t = \begin{pmatrix} z & 1 \end{pmatrix}$$

Hence, we can write

$$E(z) = c_3\mathbf{L}_2^0 z^2 + 2c_2\mathbf{L}_1^0 z + c_4\mathbf{L}_0^0 - c_5\mathbf{L}_2^0$$

Appendix C: Materials used from MERL/RGL database

MERL [MPBM03] (1) alum-bronze, (2) alumina-oxide, (3) aluminium, (4) beige-fabric, (5) blue-acrylic, (6) blue-fabric, (7) blue-rubber, (8) gold-metallic-paint, (9) gold-paint, (10) green-plastic, (11) red-fabric2, (12) red-phenolic, (13) red-specular-plastic, (14) silver-metallic-paint, (15) silver-metallic-paint2, (16) teflon, (17) two-layer-gold, (18) white-fabric, (19) white-marble, (20) white-paint, (21) yellow-phenolic.

RGL [DJ18] (22) acrylic-felt-green, (23) acrylic-felt-orange, (24) acrylic-felt-pink, (25) acrylic-felt-purple, (26) acrylic-felt-white, (27) cardboard, (28) cc-iris-purple-gem, (29) ilm-l3-37-metallic, (30) ilm-solo-m-68, (31) ilm-solo-illenium-falcon, (32) irid-flake-paint2, (33) paper-blue, (34) paper-red, (35) paper-yellow, (36) sating-gold, (37) vch-frozen-amethyst.

Fig.	Object	materials
Fig. 1	Dragon	12, 18, 32, 33, 36
	Atrium	1, 2, 6, 11, 18
Fig. 8	-	1, 4, 11
Fig. 9 (Sponza)	Sponza	10, 11, 18, 19, 20, 22, 23 24, 25, 26, 29, 34, 35, 37
Fig. 9 (Breakfast)	Table	7, 16, 34
	Chairs	16, 23, 30
	Other	3, 8, 15, 17, 18, 21, 22, 27, 31
Fig. 10, 11	-	5, 9, 13, 14