

Steerable Importance Sampling

Kartic Subr*

James Arvo†

University of California Irvine

ABSTRACT

We present an algorithm for efficient stratified importance sampling of environment maps that generates samples in the positive hemisphere defined by local orientation of arbitrary surfaces while accounting for cosine weighting. The importance function is dynamically adjusted according to the surface normal using steerable basis functions. The algorithm is easy to implement and requires no user-defined parameters. As a preprocessing step, we approximate the incident illumination from an environment map as a continuous piecewise linear function on \mathcal{S}^2 and represent this as a triangulated height field. The product of this approximation and a dynamically orientable steering function, viz. the cosine lobe, serves as an importance sampling function. Our method allows the importance function to be sampled with an asymptotic cost of $O(\log n)$ per sample where n is the number of triangles. The most novel aspect of the algorithm is its ability to dynamically compute normalization factors which are integrals of the illumination over the positive hemispheres defined by the local surface normals during shading. The key to this feature is that the weight variation of each triangle due to the clamped cosine steering function can be well approximated by a small number of spherical harmonic coefficients which can be accumulated over any collection of triangles, in any orientation, without introducing higher-order terms. Consequently, the weighted integral of the entire steerable piecewise-linear approximation is no more costly to compute than that of a single triangle, which makes re-weighting and re-normalizing with respect to any surface orientation a trivial constant-time operation. The choice of spherical harmonics as the set of basis functions for our steerable importance function allows for easy rotation between coordinate systems. Another novel element of our algorithm is an analytic parametrization for generating stratified samples with linearly-varying density over a triangular support.

1 INTRODUCTION

Variance reduction strategies are crucial elements of Monte Carlo global illumination algorithms. Without them, it is generally regarded as impractical to obtain adequately converged Monte Carlo solutions, particularly for environments that incorporate challenging lighting distributions and/or surface scattering functions. Since the earliest systematic study of Monte Carlo algorithms in rendering [14, 9, 22], both importance sampling and stratification have been recognized as being particularly relevant variance reduction strategies, although it has often been a challenge to incorporate them without simultaneously introducing statistical bias [15, 16]. Both importance sampling and stratification are now commonplace in illumination computations, and often appear in several guises within a single algorithm. While improvements to both strategies continue to be an active area of research, importance sampling offers the largest potential payoff, with the total elimination of variance being theoretically achievable [21].

Importance sampling strategies appear in a wide variety of forms, from sampling incident illumination using a simple cosine distribution, to finely adapting the sampling to a particular BRDF, or to features of the environment. In recent years considerable attention has been given to importance sampling of environment maps. There are two justifications for this focus: First, environment maps frequently encode high-dynamic range light sources [10] and therefore represent a significant challenge for efficient sampling. Secondly, light from distance sources, as represented by an environment map, is spatially independent, which greatly simplifies the task of importance sampling by reducing the dependence of such distributions to direction only.

In the context of estimating reflected radiance, a variance reduction strategy must meet several inherent requirements [16], plus an additional property that should be met if at all possible:

1. Estimate the distribution of incident illumination
2. Generate samples distributed according to the estimated illumination
3. Compute the density of each sample
4. Maintain stratification (if possible)

If the incident illumination is defined by an environment map, the first requirement is partially met; the only additional aspects that should be addressed are occlusion and weighting by the cosine of the incident angle, as the incident radiance is always integrated with respect to projected solid angle. The second requirement can be met by approximating the incident illumination using piecewise constant functions, or other simple approximations [17], which admit sampling algorithms. The third requirement is that of computing the density with which a given sample was drawn which requires that the pdf be normalized. This can always be accomplished through numerical integration of the approximating function. However, such normalization is generally significantly more costly than drawing samples, as it involves the entire importance sampling function. We refer to the latter as *the renormalization problem*, as it is frequently a significant challenge to achieving unbiased importance sampling that is computationally feasible.

We present a new approach to sample a product of two functions and demonstrate that it can be used to efficiently sample high-dynamic-range environment maps to estimate reflected radiance. Several methods have been proposed to efficiently sample environment maps and some of them even sample from the product of illumination and surface reflectance functions (see Section 2). *Our main contribution is the introduction of the notion of a steerable importance function defined as the product of sharply varying incident illumination and a smooth steerable function and a scheme to draw correctly-weighted samples from this importance function.* By sampling from an importance function that is the product of illumination over the sphere of directions and the positive cosine lobe defined by the surface normal, we reduce variance in the estimate since (1) we do not generate “wasted” samples that lie below the tangent plane and (2) we down-sample directions that are close to the horizon.

While the illumination is known a priori, the importance function also depends on a dynamically oriented clamped-cosine lobe.

*e-mail: kartic@ics.uci.edu

†e-mail: arvo@uci.edu

When we account for changes in the distribution of incident illumination above the local tangent plane and/or weighting by the cosine of the incident direction, all but the first requirement become more difficult to meet: generation of samples, computation of the densities, and maintaining stratification. These difficulties stem, in part, from the problem of renormalizing the constantly-changing distribution.

We derive a method that solves the renormalization problem decisively by means of a novel hierarchical organization that encodes all possible variation very efficiently in advance using what amounts to *steerable* functions. We consider the surfaces being rendered with respect to the environment maps to steer the importance function using a cosine lobe defined by the surface normal. This lobe is *clamped* to zero at the tangent plane of the surface, which has the effect of ignoring all illumination that arrives from below the tangent plane.

2 RELATED WORK

Reflected radiance $\mathbf{L}(x, \omega_r)$ due to direct illumination from distant sources, can be expressed as the integral

$$\int_{\mathcal{S}^2} \mathbf{L}(\omega_i) \rho(x, \omega_i, \omega_r) \max(\mathbf{n} \cdot \omega_i, 0) V(x, \omega_i) d\omega_i \quad (1)$$

since radiance incident from distant sources, $\mathbf{L}(\omega_i)$, is only a function of direction. Here \mathbf{x} is a point, with normal \mathbf{n} , on a surface with ρ as its bidirectional reflectance distribution function (BRDF) and the integration is over the sphere of incoming directions ω_i . We will refer to the third and fourth terms in the integrand as the *clamped-cosine* and visibility terms respectively.

A number of sampling strategies have been proposed to efficiently estimate this integral; these methods fundamentally differ in their choice of an importance function. The method of structured importance sampling [1] defines an importance function that is a carefully chosen combination of illumination density and solid angle separating the samples. The samples are distributed using a point relaxation algorithm and the incident illumination is approximated with several point light sources. Further, as an acceleration technique, the light sources are sorted in decreasing order of power and sampled deterministically in that order. Another method that approximates the illumination with point light sources [19], extends hierarchical Penrose tiling to quickly sample the 2D environment map; the samples also satisfy certain noise properties. While both these techniques require far fewer samples than naive sampling of only the clamped-cosine term to produce images of similar visual quality, many (about half) of the samples generated are likely to lie below the tangent plane and thus be rejected. In addition, the cosine term is ignored which means that bright sources near the horizon are sampled as profusely as sources of similar power at the pole.

Some methods include the surface BRDF [6, 8, 7] in the importance function. This allows efficient sampling of a combination of high frequency illumination and glossy surfaces with a large specular component. Lawrence et al. [17] introduced a fairly general numerical method for approximating and numerically inverting cumulative distribution functions, which lends itself to both stratification and importance sampling. Ghosh et al. [12] proposed a method to account for temporal coherence in animation sequences involving environment maps. In this paper, we describe a method that uses the clamped-cosine weighted illumination as an importance function thus automatically accounting for the cosine importance given a normal direction, and also ensuring that all samples are generated above the tangent plane.

Ramamoorthi and Hanrahan first observed that a clamped cosine lobe could be very effectively approximated using a small number of spherical harmonic basis functions; indeed, nine such coefficients attains a fit that is deemed sufficiently accurate for most

graphics applications [20]. They also observed that the spherical harmonic representation of a rotated lobe is no more complex than a static one in that no higher-order terms are added as a result of any rotation. It is precisely these observations that we build upon here to obtain an importance sampling function that can dynamically account for any incident surface orientation by pre-computing its response to a steerable lobe; in this case, a clamped cosine lobe. We shall see that this solves the renormalization problem by making renormalization of an arbitrary piecewise linear importance sampling function equivalent in cost to re-weighting a single point in the environment map. We first approximate the environment map as a piecewise-linear 2D function; Section 3.3 explains how the piecewise-linear function can be re-weighted and re-normalized very efficiently by a clamped-cosine lobe, thus making the entire importance-sampling function “steerable.” Section 5 starts with the basic problem of drawing stratified samples with a linearly-varying density over a triangular support, as this will be the basis for stratified sampling of our piecewise-linear importance sampling function. Figure 9 shows such an importance sampling function, that is dynamically re-weighted and re-normalized via a steerable clamped cosine lobe along with the stratified samples drawn from it.

In the following sections we define our steerable importance function and construct procedures and a data structure to allow stratified sampling of the function and correctly computing the densities with which samples were generated. To summarize, our contributions are :

1. We define a novel importance function that dynamically accounts for (a) high frequency illumination, (b) cosine-weighting defined by the normal and (c) the local tangent plane, by only generating samples in the positive hemisphere.
2. We introduce a tree data structure whose novel traversal algorithm is guided dynamically by a parameter which is the local surface normal.
3. We derive a method for stratified sampling of triangles according to a linear density. An earlier method [3] for stratified sampling of triangles only details the scheme for sampling constant and not linear densities.
4. We draw stratified samples from the dynamically normalized importance function, construct a weighting scheme and use the generated samples to estimate reflected radiance.

3 STEERABLE IMPORTANCE SAMPLING

3.1 Steerability

Steerable functions have been used extensively as general tools in image processing and computer vision. Specifically, their properties were exploited in constructing filters of arbitrary orientations by describing them as linear combinations of basis filters [11]. Ashikmin and Shirley used Steerable Illumination Textures for lighting bumpy surfaces in a way that shadowing and interreflection were considered [5]. In this section, we briefly introduce steerable functions and our novel use of them as importance functions. A detailed literature survey and description of steerable functions can be found in Teo’s dissertation [23].

A steerable function is defined as one whose transformed version can always be expressed as a linear combination of a fixed, finite set of basis functions. The coefficients in the linear combination together form the “steering” function and are dependent on the transformation. Thus a steerable function $g(\mathbf{x})$ can be written as an inner product—

$$g_{\mathbf{n}}(\mathbf{x}) = \langle \mathbf{s}(\mathbf{n}), \mathbf{b}(\mathbf{x}) \rangle \quad (2)$$

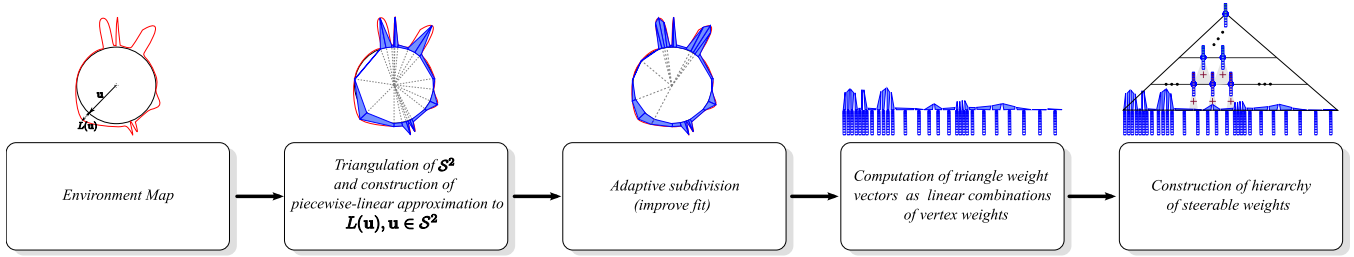


Figure 1: 2D illustration of the algorithm used to construct the tree with steerable weights. Here segments are equivalent to triangles in the 3D setting. The vectors of weights associated with the triangles are at the leaves of the tree, and are propagated up to the root; weights at internal nodes are computed simply as the sum of the weights of their children.

where \mathbf{n} is a vector of parameters for the particular transformation, $\mathbf{s}(\mathbf{n})$ is a set of steering functions which depend on the transformation and $\mathbf{b}(\mathbf{x})$ is a set of basis functions. The definition of steerable functions suggests that the transformed function must be expressible as a linear, analytic expression containing the basis functions; however, in practice, steerable functions and their basis functions are often represented in sampled form and thus are thus approximations [23].

The main idea in this paper is to use a steerable function as an importance function. By virtue of being steerable, the importance function can continuously be reoriented by the superpositioning of a fixed, finite set of basis functions. Following Ashikmin and Shirley [5], the domain of our function is the sphere of directions \mathcal{S}^2 , therefore we use the spherical harmonics bases weighted by the illumination in the corresponding direction from an environment map. As a steering function, we use the positive lobe of the cosine function and our dynamically defined importance function is the cosine weighted illumination in the positive hemisphere at each point being shaded. The decrease in variance results from (1) generating samples that lie only above the local tangent plane and (2) weighting them according to the product of illumination and the oriented cosine.

In this paper we consider only the use of a single clamped cosine as a steering function and define an importance function for reducing the variance in estimates for reflected radiance due to direct illumination from environment maps.

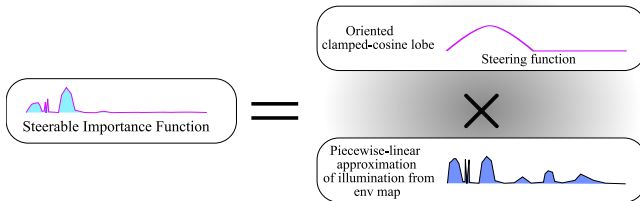


Figure 2: For the purpose of illustration we show the lower dimension analog of our steerable importance function for one orientation of the clamped cosine and a 2D environment map which is a function of $\mathbf{u} \in \mathcal{S}$.

Veach and Guibas described a technique by which to effectively sample from the sum of multiple functions provided that each can be sampled in isolation [25]. While the approach is quite general, it has several important drawbacks when applied to sampling environment maps. First, it generates samples over the entire support of each pdf, which would produce many samples in the wrong hemisphere. Second, the technique is only amenable to stratification within each pdf, not to the actual importance function that is a combination of the two pdf's.

Clarberg et al introduced a technique to sample from products of functions using wavelets as basis functions and demonstrated its

effectiveness in sampling a product of illumination and reflectance functions [7]. However, their method suffers from a fundamental limitation inherent to using wavelet multiplication, which is that there is no straightforward way of rotating wavelets [18]. In contrast, our choice of spherical harmonics bases enables easy rotation which in turn allows the two functions to easily be expressed in the same coordinate system. Consequently, their method requires the precomputation of the wavelet coefficients for a fixed set of discrete directions and interpolates the coefficients for directions that are not in this set.

3.2 The Importance Function

By partitioning the domain of integration in Equation (1), \mathcal{S}^2 , into spherical triangles S_i , $i = 1, 2, \dots, M$ we rewrite the r.h.s. of the equation as the sum of integrals over S_i

$$\sum_{i=1}^M \int_{S_i} \mathbf{L}(\omega) \rho(x, \omega, \omega_r) f(\mathbf{n}, \omega) V(x, \omega) d\omega, \quad (3)$$

where $f(\mathbf{n}, \omega) = \max(\omega \cdot \mathbf{n}, 0)$.

Consider one of the spherical triangles, S_i , and the planar triangle $\Delta(i)$ defined by the vertices of S_i . Let $\mathbf{p} = (p_0, p_1) \in [0, 1]^2$ be a point on $\Delta(i)$ defined using the parametrization $\psi: [0, 1]^2 \rightarrow \Delta$. Disregarding visibility and the BRDF for the moment, and switching to the above parameterization, we get

$$\int_0^1 \int_0^1 \mathbf{L}(\mu_i(p_0, p_1)) f(\mathbf{n}, \mu_i(p_0, p_1)) \varphi_i(p_0, p_1) |J_i(p_0, p_1)| dp_0 dp_1$$

where $\varphi_i(p_0, p_1)$ arises as a result of using a change of variables from the plane onto the sphere and $\mu_i(p_0, p_1)$ is the unit vector along $\psi_i(p_0, p_1)$. Here,

$$\mu_i(p_0, p_1) = \frac{\psi_i(p_0, p_1)}{\|\psi_i(p_0, p_1)\|}, \quad \varphi_i(p_0, p_1) = \frac{\mu_i(p_0, p_1) \cdot \mathbf{n}_{\Delta(i)}}{\|\psi_i(p_0, p_1)\|^2} \quad (4)$$

and $\mathbf{n}_{\Delta(i)}$ is the unit normal of $\Delta(i)$.

We normalize the function $|J_i(p_0, p_1)|$, to make it a pdf and obtain the Monte Carlo estimator

$$G_i \sum_{j=1}^N \mathbf{L}(\mu_i^j) f(\mathbf{n}, \mu_i^j) \varphi_i^j \quad (5)$$

where samples ψ_i^j drawn from the pdf that is proportional to $|J_i(p_0, p_1)|$ are used to obtain μ_i^j and φ_i^j .

We derive ψ_i such that the Jacobian is linear in both parameters and equal to the illumination weighted by the clamped cosine at

each vertex of $\Delta(i)$ (see Appendix). The normalization factor G_i is simply the integral of the Jacobian and is given by

$$G_i = \int_0^1 \int_0^1 |J_i(p_0, p_1)| dp_0 dp_1. \quad (6)$$

Replacing the BRDF and visibility terms and adding the estimates over all spherical triangles S_i , we arrive at our estimate of the total reflected radiance along ω_r as

$$\sum_{i=1}^M G_i \sum_{j=1}^N \mathbf{L}(\mu_i^j) \rho(x, \mu_i^j, \omega_r) f(\mathbf{n}, \mu_i^j) V(x, \mu_i^j) \varphi_i^j \quad (7)$$

Note that the piecewise linear importance function is a linear interpolation used to reduce variance need only be an approximation, and 2) no bias is introduced so long as we can correctly compute the density of the samples generated. For this task, we shall use the spherical harmonic approximation of the clamped cosine lobe first proposed by Ramamoorthi and Hanrahan [20] in the context of fast approximations of irradiance due to distant sources. Our application will differ fundamentally, but will nonetheless enjoy the benefits of concise representation and fast evaluation. First, observe that the function

3.3 Steerable Weighting

Two of the most attractive features of importance sampling are that 1) the distribution used to reduce variance need only be an approximation, and 2) no bias is introduced so long as we can correctly compute the density of the samples generated. For this task, we shall use the spherical harmonic approximation of the clamped cosine lobe first proposed by Ramamoorthi and Hanrahan [20] in the context of fast approximations of irradiance due to distant sources. Our application will differ fundamentally, but will nonetheless enjoy the benefits of concise representation and fast evaluation. First, observe that the function

$$f(\mathbf{n}, \omega) = \max(\langle \mathbf{n}, \omega \rangle, 0), \quad (8)$$

which is what we have been referring to as a clamped cosine lobe, can be approximated by a finite linear combination of spherical harmonics (SH):

$$f(\mathbf{n}, \omega) \approx \sum_{i=0}^k \mathbf{a}_i(\mathbf{n}) \mathbf{Y}_i(\omega), \quad (9)$$

where we have treated the spherical harmonics as functions defined on the sphere, \mathcal{S}^2 , rather than the more traditional function of two angles. We have also “linearized” the indexing of the basis functions, which are traditionally indexed with double subscripts denoted by ℓ and m , with $\ell = 0, 1, 2, \dots$, and $-\ell \leq m \leq \ell$. In particular, our ordering coincides with the subscripts $(0, 0)$, $(1, -1)$, $(1, 0)$, $(1, 1)$, $(2, -2)$, $(2, -1)$, $(2, 0)$, $(2, 1)$, and $(2, 2)$, etc. Here $\mathbf{a}(\mathbf{n})$ is the vector of SH coefficients after being rotated using the normal \mathbf{n} .

The product of the incident distant illumination along ω and the clamped cosine at ω for a given normal \mathbf{n} can be expressed as

$$\begin{aligned} \mathbf{L}(\omega) f(\mathbf{n}, \omega) &\approx \mathbf{L}(\omega) \sum_{i=0}^k \mathbf{a}_i(\mathbf{n}) \mathbf{Y}_i(\omega) \\ &= \sum_{i=0}^k \mathbf{L}(\omega) \mathbf{a}_i(\mathbf{n}) \mathbf{Y}_i(\omega) \\ &= \langle \mathbf{a}(\mathbf{n}), \mathbf{w}(\omega) \rangle \end{aligned} \quad (10)$$

where $\mathbf{w}(\omega) = \mathbf{L}(\omega) \mathbf{Y}(\omega)$ is a vector containing the SH bases associated with a direction ω , weighted by the illumination along that direction. Observe that this expression is similar to the r.h.s. of Equation (2), suggesting that the product $\mathbf{L}(\omega) f(\mathbf{n}, \omega)$ is steerable.

We define the function $|J_j(p_0, p_1)|$ in each triangle $\Delta(j)$ with vertices A, B and C as a linear combination of products of illumination and the clamped cosines at vertices. The integral of the Jacobian (see Equation (6)) within each triangle is simply the volume

of the truncated triangular prism defined by the triangle. Given that the Jacobian varies linearly within each triangle $\Delta(j)$, we can write G_j as $\langle \mathbf{a}(\mathbf{n}), \mathbf{W}_j \rangle$ where

$$\mathbf{W}_j = \frac{\text{Area}(\Delta(j))}{3} (\mathbf{w}_j^a + \mathbf{w}_j^b + \mathbf{w}_j^c). \quad (11)$$

We precompute and store $\mathbf{w}(\omega)$ at each vertex in the partition of the sphere of directions and a weight W_j associated with each triangle $\Delta(j)$. Given a normal \mathbf{n} we first compute $\mathbf{a}(\mathbf{n})$ and then G_j in constant time for each triangle $\Delta(j)$ with just one dot product.

The next observation is the crucial one for the approach we present in this paper. If Q is any set of triangle indices, then

$$\begin{aligned} \sum_{j \in Q} \mathbf{W}_j &= \sum_{j \in Q} \langle \mathbf{a}, \mathbf{W}_j \rangle \\ &= \left\langle \mathbf{a}, \sum_{j \in Q} \mathbf{W}_j \right\rangle \\ &= \langle \mathbf{a}, \mathbf{W}_Q \rangle, \end{aligned}$$

where \mathbf{W}_Q is a new collection of nine coefficients. Thus, the total weight of all the triangles combined is, once again, represented by a collection of the *same* number of spherical harmonic coefficients; *summing the contributions of any number of triangles in any orientation does not introduce higher-order terms.*

To fully exploit this property, we organize the triangles in the partition of the sphere of directions then organize hierarchically as a binary tree. Each triangle is assigned a vector of nine values, which is then propagated up the tree, adding the weights of the children at each internal node, until the root is reached (see Figure 1). The algorithm for generating samples from the resulting piecewise-linear function is illustrated in Figure 5, and the algorithms for generating the rotated cosine lobe coefficients and for traversing the three structure are shown in Figure 4, respectively. Further details on the generation of the triangular mesh are provided in Section 4.

4 IMPLEMENTATION AND RESULTS

4.1 Preprocess

The preprocess step is composed of two main stages—triangulation of the environment map and construction of a reasonably balanced binary tree. While a balanced tree is not required for correctness of the algorithm, balance ensures an $O(S \log N_\Delta)$ asymptotic bound on the execution time if S stratified samples are required to be drawn for any given normal vector and the triangulation consists of N_Δ triangles. The domain is triangulated by uniform subdivision of an icosahedron followed by a step of adaptive subdivision. During adaptive subdivision, triangles are subdivided if the deviation of the linear approximation within them from the actual illumination is found to be greater than a threshold. After subdivision, vertex and triangle weights are computed and stored.

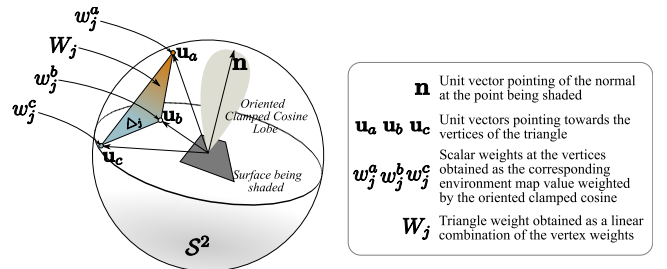


Figure 3: The weight of each triangle vertex is determined by the environment map and its position relative to the surface under consideration. The weight of the entire triangle is a linear combination of its vertex weights.

```

function Sample( $\mathbf{n}, \xi_1, \xi_2$ )
1  $\mathbf{a} \leftarrow \text{RotateLobeCoeffs}(\mathbf{n})$ 
2  $\mathbf{w} \leftarrow$  weight coefficients of tree
3  $v \leftarrow$  root of tree
4 while  $v$  is not a leaf do
5    $w_l \leftarrow \langle \mathbf{a}, \text{LeftWeightCoeffs}(v) \rangle$ 
6    $w_r \leftarrow \langle \mathbf{a}, \text{RightWeightCoeffs}(v) \rangle$ 
7    $w \leftarrow \frac{w_l}{w_l + w_r}$ 
8   if  $\xi_1 < w$  then
9      $\xi_1 \leftarrow \frac{\xi_1}{w}$ 
10     $v \leftarrow \text{LeftChild}(v)$ 
11  else
12     $\xi_1 \leftarrow \frac{\xi_1 - w}{1 - w}$ 
13     $v \leftarrow \text{RightChild}(v)$ 
14  endif
15 endwhile
16  $(s_\Delta, \rho_\Delta) \leftarrow \text{SampleTriangle}(\text{Triangle}(v), \xi_1, \xi_2)$ 
17 if  $\langle \mathbf{n}, s_\Delta \rangle < 0$  then  $s_\Delta \leftarrow -s_\Delta$ 
18 return  $(s_\Delta, \frac{\rho_\Delta}{\langle \mathbf{a}, \mathbf{w} \rangle})$ 

```

```

function RotateLobeCoeffs( $\mathbf{n}$ )
1  $\mathbf{a}_0 \leftarrow c_4$ ;  $\mathbf{a}_1 \leftarrow 2c_2\mathbf{n}_y$ ;  $\mathbf{a}_2 \leftarrow 2c_2\mathbf{n}_z$ 
2  $\mathbf{a}_3 \leftarrow 2c_2\mathbf{n}_x$ ;  $\mathbf{a}_4 \leftarrow 2c_1\mathbf{n}_x\mathbf{n}_y$ ;  $\mathbf{a}_5 \leftarrow 2c_1\mathbf{n}_y\mathbf{n}_z$ 
3  $\mathbf{a}_6 \leftarrow c_3\mathbf{n}_z^2 - c_5$ ;  $\mathbf{a}_7 \leftarrow 2c_1\mathbf{n}_x\mathbf{n}_z$ ;  $\mathbf{a}_8 \leftarrow c_1(\mathbf{n}_x^2 - \mathbf{n}_y^2)$ 
4 return  $\mathbf{a}$ 

```

Figure 4: The basic algorithm for stratified sampling of the dynamically re-weighted piecewise-linear importance function. The variables ξ_1 and ξ_2 are assumed to be stratified random variables in $[0, 1] \times [0, 1]$. Note that step 16 introduces a bias which can be eliminated by a slight increase in computational cost (Figure 7). $c_1 = 0.429043, c_2 = 0.511644, c_3 = 0.743125, c_4 = 0.886227, c_5 = 0.247708$ according to Ramamoorthi and Hanrahan [20].

We build a binary tree that has the triangles randomly distributed as its leaf nodes. Each triangle is associated with a weight, which is the volume of the truncated prism formed by raising its vertices by the appropriate heights. We approximate this volume with one third the area of the triangle times the average height at its vertices. Although this is an approximation and makes the importance function deviate slightly from the actual function on the sphere, it does not introduce a bias so long as the weights computed are in accordance with the densities that samples are drawn from. This approximation converges to the correct volume as the triangulation is refined.

The internal nodes of the tree represent clusters of triangles and their volumes can each be written as a sum of the volumes of their respective child nodes. Thus we sum up the individual basis vectors of the children to compute the basis vector at each internal node. The actual volume, including the cosine weighting, is computed by a dot product of this weighted basis vector with the coefficient vector of the clamped cosine which is provided during query. Thus we build the tree in a bottom-up fashion, at each node summing up and storing the basis vectors of child nodes (see Figure 1).

It is interesting to note that the volume of the root, which represents the volume under the importance function, is computed for a given normal direction by just one dot product which trivializes the cost of renormalization.

4.2 Sample Generation

Given a normal direction and two random variables chosen uniformly in $[0, 1]$ we draw a single sample from our importance function in three steps—picking the triangle to sample from, drawing a sample from that triangle according to the weights defined at the vertices and actually computing the density with which the sample

was chosen.

Starting with the root we evaluate the volume at each internal node (one dot product each) and use the information to guide the path down to the leaf. At each level the path favors the child with a higher volume (see Figure 5). Thus using one of the random variables, and $O(\log N_\Delta)$ inner products (each of 9 coefficient vectors), we pick a triangle proportional to the integral of the linearly-varying densities (See Figure 4). Once we pick a triangle we re-scale the random value used to traverse the tree to $[0, 1]$ and sample from the triangle using the two random variables as shown in Section 5.

Computing the density with which the given sample was chosen is trivially obtained by the ratio of the actual height at that sample (which is obtained by interpolating between the heights of the vertices) and the total volume associated with all the triangles (which is the volume of the root). Both are evaluated in constant time.

4.3 Results

Figure 9 shows the importance function as the product of the environment map values and the oriented clamped cosine lobe for two different environment maps, each with a differently oriented lobe; the resulting samples drawn are also shown.

We compare our method against *standard stratified importance sampling* by obtaining irradiance estimates for a set of normal directions by varying the polar angle and comparing variances in the estimates. The standard method involves treating the environment map image as a discrete 2D function from which stratified samples are drawn using numerical inversion [1]. Since our method is steerable, and accounts for the normal while generating samples, we achieve significantly lower variance especially when the number of samples is few or the normal is facing away from bright illumination. Figure 8 shows the variances in irradiance estimates using stratified importance sampling and steerable stratified importance sampling for 16 and 64 samples.

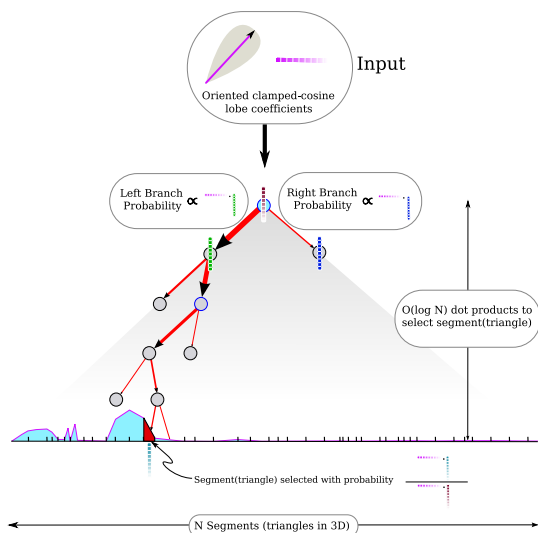


Figure 5: Once a nearly-balanced binary tree for the triangles has been built, and the spherical harmonic representation of the weight at each vertex has been propagated up to the root, stratified sampling from with respect to any surface orientation is very straightforward. As the cosine lobe is changed, the branching probabilities along every path are altered. To reach a leaf triangle with the correct probability, only 9-element dot products along the path to that triangle are computed. Thus, the cost of generating a sample and computing its correct density is $O(\log n)$, where n is the number of triangles. The details of path traversal are shown in Figure 4.

```

function SampleTriangle( $T, \xi_1, \xi_2$ )
1  $s \leftarrow f(\xi_1, T.w^a, T.w^b, T.w^c)$ 
2  $t \leftarrow g(\xi_2, s, T.w^a, T.w^b, T.w^c)$ 
3  $w \leftarrow (1-s)T.w^a + s(1-t)T.w^b + stT.w^c$ 
4  $\mathbf{p} \leftarrow (1-s)T.\mathbf{A} + s(1-t)T.\mathbf{B} + stT.\mathbf{C}$ 
5 return ( $w, \mathbf{p}$ )

```

```

function  $f(\xi, w^a, w^b, w^c)$ 
1  $X \leftarrow (w^b - w^a)/3 + (w^c - w^b)/6$ 
2  $Y \leftarrow w^a/2$ 
3  $\alpha \leftarrow X/(X+Y)$ 
4  $\beta \leftarrow Y/(X+Y)$ 
5 return RootOf( $\alpha x^3 + \beta x^2 - \xi$ )

```

```

function  $g(s, \xi, w^a, w^b, w^c)$ 
1  $t \leftarrow s(w^c - w^b) + 2(1-s)w^a + sw^b$ 
2  $\gamma \leftarrow s(w^c - w^b)/t$ 
3  $\rho \leftarrow 2((1-s)w^a + sw^b)/t$ 
4 return  $2\xi/(\rho + \sqrt{\rho^2 + 4\gamma\xi})$ 

```

Figure 6: The complete algorithm for stratified sampling of triangles with vertex weights w^a , w^b , and w^c defining linear probability density functions. The RootOf function can be either closed form, or iterative, using Newton’s method.

Figure 10 shows the effectiveness of our sampling technique by rendering images using different numbers of samples of a scene with diffuse, glossy and specular materials.

5 LINEAR STRATIFIED SAMPLING OF TRIANGLES

In this section we present a simple and compact algorithm that allows generation of stratified samples according to a linearly-varying density function over a triangle with vertex weights that we shall denote by w^a , w^b and w^c . Our approach is similar to the method that can be used to derive area-preserving maps onto 2-manifolds [3] such as planar and spherical triangles [24, 2] with the fundamental difference that the general approach had been used to derive mappings with only a constancy constraint on the determinant of the Jacobian matrix. For sampling a linear density, we need to derive a mapping such that this determinant is proportional to the linear function we wish to sample from. We show that this problem—of deriving a continuous bijection from the unit square onto the triangle such that the determinant of the jacobian of this bijection is proportional to the linear function defined by the vertex weights—can be reduced to finding the roots of a cubic and a quadratic in the interval $[0, 1]$ with coefficients being functions of the vertex weights (see appendix for the derivation). It can be easily verified that the cubic function cannot attain a maxima in the interval $(0, 1)$ and is therefore guaranteed to have exactly one real root in that interval.

The algorithm for sampling from the triangle with vertex weights is summarized in Figure 6, and sample output from the algorithm is shown in Figure 11. Using this algorithm to sample from triangles in conjunction with the tree-traversal algorithm (Section 4.2) allows stratified sampling of the steerable importance function.

6 BIAS: SOURCES AND REMEDIES

Ring in the lobe approximation: Because the clamped cosine lobe is approximated by its projection onto a finite set of basis functions, there is a small amount of ringing near the derivative discontinuity. The ringing causes the approximation to become slightly negative where the lobe is clamped to zero. This is easily fixed by adding an offset (approximately 0.09) to the coefficient corresponding to the constant basis function. This will increase the values uniformly, thus somewhat reducing the effectiveness of the importance sampling by decreasing the overall variation. This approach completely eliminates the possibility of negatively weighted triangles and negative densities, and introduces no additional bias. However,

```

15 ( $s_\Delta, \rho_\Delta$ )  $\leftarrow$  SampleTriangle( Triangle( $v$ ),  $\xi_1, \xi_2$ )
16  $\rho^- \leftarrow$  GetDensity( $\mathbf{a}, -s_\Delta$ )
17 if ( $\mathbf{n}, s_\Delta$ ) < 0 then  $s_\Delta \leftarrow -s_\Delta$ 
18 return ( $s_\Delta, \frac{\rho_\Delta + \rho^-}{\langle \mathbf{a}, \mathbf{w} \rangle}$ )

```

Figure 7: This is a modification to the algorithm shown in Figure 4. When samples are generated in the wrong hemisphere, they can be reflected through the origin and used. This introduces no bias, provided the density of the sample takes into account both antipodal samples.

uniformly raising the value of the function causes some stray samples, with low probability, to be generated in triangles that should not have been sampled and hence marginally increases the variance of the estimator.

Samples in the wrong hemisphere: A second minor source of bias is due to samples that are occasionally generated below the horizon. This results from an approximation to the clamped cosine lobe that is not exactly zero in the hemisphere below the horizon. Consequently, there is a small probability that it will be sampled. This problem is exacerbated by the global offset that guarantees the function is non-negative. One solution is to simply ignore such samples which amounts to rejection. Another solution is to ignore the bias due to reflecting them into the positive hemisphere; as they occur infrequently, there is little error in any case. However to remove this bias, we increase the density of all samples generated to account for the density of those that arrived there through reflection. Thus, whether a sample falls in the correct hemisphere or not, we add the densities of the two antipodal directions, as shown in Figure 7. This policy will generate a very low-probability “ghost” of the opposite hemisphere, and is therefore likely to produce a small number of samples that are not very useful, but the resulting estimator will be unbiased.

7 CONCLUSION AND FUTURE WORK

In this paper, we defined a novel piecewise linear steerable importance function that accounts for the cosine weighting for any surface orientation and presented an efficient algorithm for stratified importance sampling of environment maps by using this importance function. We achieved this by constructing a binary tree whose structure remains unchanged during the entire sample generation process. Given a surface orientation (normal vector), renormalization of densities associated with a triangles was done in constant time and our sampling cost was logarithmic in the number of triangles used to partition the environment map.

Another novel contribution of this paper is an algorithm to generate stratified samples on a triangle according to a linearly-varying density. The algorithm is simple, compact and does not require any numerical inversion techniques.

Including reflectance functions in the importance function has been shown to be an important step towards variance reduction. Our method to sample the environment map may be used in conjunction with sophisticated BRDF sampling using multiple importance sampling [25]. As future work, we would like to increase the degrees of freedom in the steering function so that specular lobes of reflectance functions can be taken into consideration, allowing us to efficiently sample from a product of the illumination and reflectance functions. We expect that this will be a significant step in reducing variance further. However, the increased dimensionality of the steering function suggests that the compressed storage of the steerable basis could prove challenging.

Using stripification[13] to benefit from the stratification (by improving locality) or low-discrepancy sequences in place of stratified sampling are other directions to explore for further variance reduction.

Acknowledgements: The authors wish to thank Cyril Soler, Fredo Durand and the anonymous reviewers for their helpful comments.

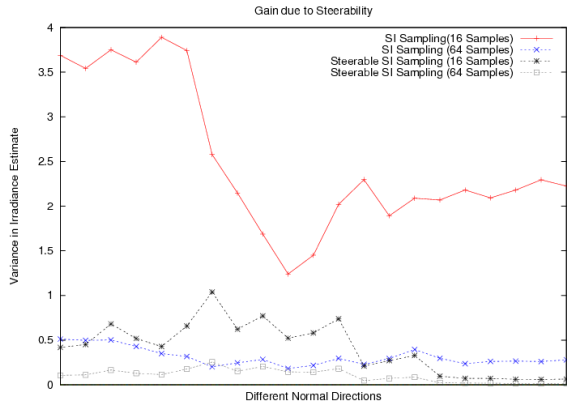


Figure 8: Comparison plots of variance in irradiance estimates of our algorithm against standard stratified importance sampling which uses the 2D-density of the illumination in the environment map as an importance function. Tests were run using the St.Peter’s Basilica environment map, on a set of normal directions by varying the polar angle in the interval $(0, \pi)$. There is a significant increase in variance for normals facing away from the illumination using the standard method as a result of not considering local surface orientation in the importance function.

REFERENCES

- [1] S. Agarwal, R. Ramamoorthi, S. Belongie, and H. W. Jensen. Structured importance sampling of environment maps. In *Computer Graphics Proceedings*, ACM SIGGRAPH, pages 605–612, 2003.
- [2] J. Arvo. Stratified sampling of spherical triangles. In *Computer Graphics Proceedings*, ACM SIGGRAPH, pages 437–438, Aug. 1995.
- [3] J. Arvo. Stratified sampling of 2-manifolds. In *State of the Art in Monte Carlo Ray Tracing for Realistic Image Synthesis, SIGGRAPH 2001 Course Notes*, volume 29, Aug. 2001.
- [4] M. Ashikhmin and P. Shirley. An anisotropic Phong BRDF model. *Journal of Graphics Tools*, 5(2):25–32, 2000.
- [5] M. Ashikhmin and P. Shirley. Steerable illumination textures. *Transactions on Graphics*, 21(1):1–19, Jan. 2002.
- [6] D. Burke, A. Ghosh, and W. Heidrich. Bidirectional importance sampling for illumination from environment maps. In *ACM SIGGRAPH 2004 Sketches*. ACM Press, 2004.
- [7] P. Clarberg, W. Jarosz, T.-A. Moller, and H. W. Jensen. Wavelet importance sampling: Efficiently evaluating products of complex functions. In *ACM TOG*, pages 1166–1175, Aug. 2005.
- [8] D. Cline, P. K. Egbert1, J. F. Talbot, and D. L. Cardon. Two stage importance sampling for direct lighting. *Proc. Eurographics Symposium on Rendering (EGSR’06)*, 2006.
- [9] R. L. Cook. Stochastic sampling in computer graphics. *ACM TOG*, 5(1):51–72, 1986.
- [10] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *Computer Graphics Proceedings*, ACM SIGGRAPH, pages 369–378, Aug. 1997.
- [11] W. Freeman. Steerable filters and the local analysis of image structure, 1992.
- [12] A. Ghosh, A. Doucet, and W. Heidrich. Sequential sampling for dynamic environment map illumination. *Proc. Eurographics Symposium on Rendering (EGSR’06)*, 2006.
- [13] M. Gopi and D. Eppstein. Single strip triangulation of manifolds with arbitrary topology. *Computer Graphics Forum (EUROGRAPHICS)*, 23(3):371–379, 2004.
- [14] J. T. Kajiya. The rendering equation. *CG*, 20(4):143–150, Aug. 1986.
- [15] D. Kirk and J. Arvo. Unbiased sampling techniques for image synthesis. *CG*, 25(4):153–156, July 1991.
- [16] D. Kirk and J. Arvo. Unbiased variance reduction for global illumination. In *Proceedings of the Second Eurographics Workshop on*

Rendering, Barcelona, May 1991.

- [17] J. Lawrence, S. Rusinkiewica, and R. Ramamoorthi. Adaptive numerical cumulative distribution functions for efficient importance sampling. In *EGSR 2005*, pages 11–20, 2005.
- [18] R. Ng, R. Ramamoorthi, and P. Hanrahan. All-frequency shadows using non-linear wavelet lighting approximation. In *SIGGRAPH ’03: ACM SIGGRAPH 2003 Papers*, pages 376–381, New York, NY, USA, 2003. ACM Press.
- [19] V. Ostromoukhov, C. Donohue, and P.-M. Jodoin. Fast hierarchical importance sampling with blue noise properties. *ACM TOG*, 23(3):488–495, 2004. Proc. SIGGRAPH 2004.
- [20] R. Ramamoorthi and P. Hanrahan. An efficient representation for irradiance environment maps. In *Computer Graphics Proceedings*, ACM SIGGRAPH, pages 497–500, 2001.
- [21] R. Y. Rubinstein. *Simulation and the Monte Carlo Method*. John Wiley & Sons, NY, 1981.
- [22] G. Shirley, C. Wang, and K. Zimmerman. Monte Carlo methods for direct lighting calculations. *ACM TOG*, 15(1):1–36, Jan. 1996.
- [23] P. C.-S. Teo. *Theory and applications of steerable functions*. PhD thesis, 1998. Adviser-David J. Heeger.
- [24] G. Turk. Generating random points in triangles. In A. S. Glassner, editor, *Graphics Gems*, pages 24–28. Acad. Press, NY, 1990.
- [25] E. Veach and L. J. Guibas. Optimally combining sampling techniques for monte carlo rendering. In *SIGGRAPH ’95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 419–428, New York, NY, USA, 1995. ACM Press.

APPENDIX: DERIVATION OF PARAMETERIZATION

We start with a parametrization

$$\phi : [0, 1]^2 \rightarrow \Delta \quad (12)$$

of the triangle \mathbf{ABC} , based on barycentric coordinates,

$$\phi(s, t) = (1-s)\mathbf{A} + s(1-t)\mathbf{B} + st\mathbf{C} \quad (13)$$

which defines a smooth bijection from the unit square to the triangle. From this mapping, we derive two functions f and g such that the composition

$$\psi(\xi_1, \xi_2) \equiv \phi(f(\xi_1), g(f(\xi_1), \xi_2)) \quad (14)$$

is a parametrization whose Jacobian matrix is proportional to the linearly-varying density. Note that the functions are cascaded; that is, the result of f is used as a parameter to the function g . If the parameters ξ_1 and ξ_2 are two uniformly distributed random variables in $[0, 1]$, then the resulting points will have the desired pdf. Moreover, if the samples are stratified, the generated samples will retain the stratification.

The functions f and g are inverses of the normalized cumulative distributions $F(s)$ and $G_s(t)$ over the first and second and second parameters of the mapping in (13) respectively.

To find F and G we first define a function $\sigma_w : [0, 1]^2 \rightarrow \Delta$, which corresponds to the weighted area of the triangle. That is,

$$\int_{\Delta} \sigma_w = \int_{\Delta} w \sigma, \quad (15)$$

where w is the linear function

$$w(u, v) = (1-s)w^a + s(1-t)w^b + stw^c \quad (16)$$

and σ is the surface area 2-form,

$$\int_{\Delta} \sigma = \text{Area}(\phi(\Delta)). \quad (17)$$

The cumulative distributions are expressed analogous to their uniform counterparts [3], with the difference being that the integrand is replaced by σ_w . That is,

$$F(s) = \frac{\int_0^1 \int_0^s \sigma_w(u, v) dudv}{\int_0^1 \int_0^1 \sigma_w(u, v) dudv} \quad (18)$$

$$G_s(t) = \frac{\int_0^1 \sigma_w(s, v) dv}{\int_0^1 \sigma_w(s, v) dv} \quad (19)$$

Substituting the appropriate expressions and integrating, we obtain $\alpha x^3 + \beta x^2 - \xi_1$ and $\gamma x^2 + \rho x - \xi_2$ as expressions for F and G_s respectively. These cumulative distributions

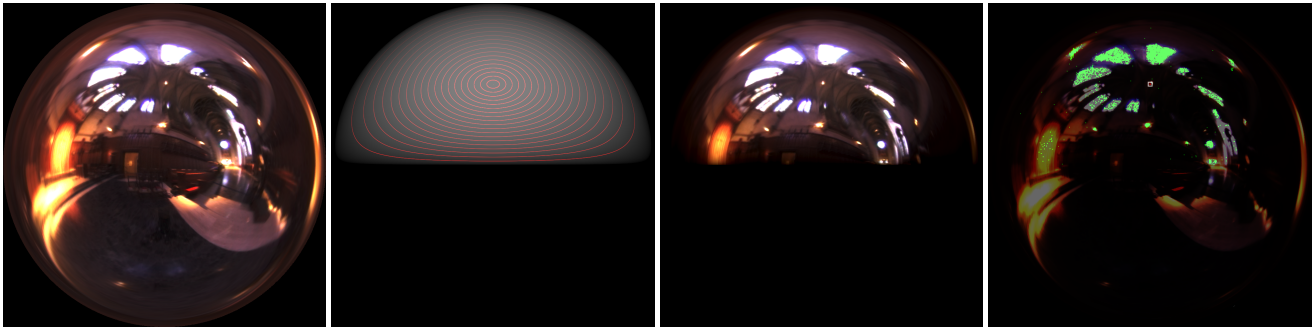


Figure 9: From left to right: Input map of Grace Cathedral; clamped-cosine function (with iso-polar lines in red); our importance function (clamped-cosine weighted input); Samples (green) drawn from the importance function (juxtaposed on dimmed input). Very few of the samples lie in the low-intensity regions of the map and none in the hemisphere below the tangent plane. A large number (100,000) of samples is shown to highlight the effectiveness of the method.

may be inverted analytically or numerically to obtain functions f and g . Here α , β , γ and ρ_s are

$$\alpha = \frac{w^b + w^c - 2w^a}{w^a + w^b + w^c} \quad (20)$$

$$\beta = \frac{3w^a}{w^a + w^b + w^c} \quad (21)$$

$$\gamma_s = \frac{s(w^c - w^b)}{s(w^c - w^b) + 2(1-s)w^a + sw^b} \quad (22)$$

$$\rho_s = \frac{2(1-s)w^a + sw^b}{s(w^c - w^b) + 2(1-s)w^a + sw^b}, \quad (23)$$

and $s = f(\xi_1)$

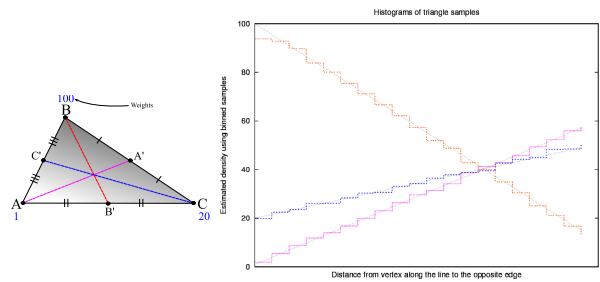


Figure 11: Triangle ABC (left) was sampled using our linear stratified sampling algorithm. Samples along AA' (magenta), BB' (red) and CC' (blue) were collected in 20 bins (for each line) and estimates of the density in each bin were computed and plotted. Also shown are the analytically computed expected densities along each line (black).

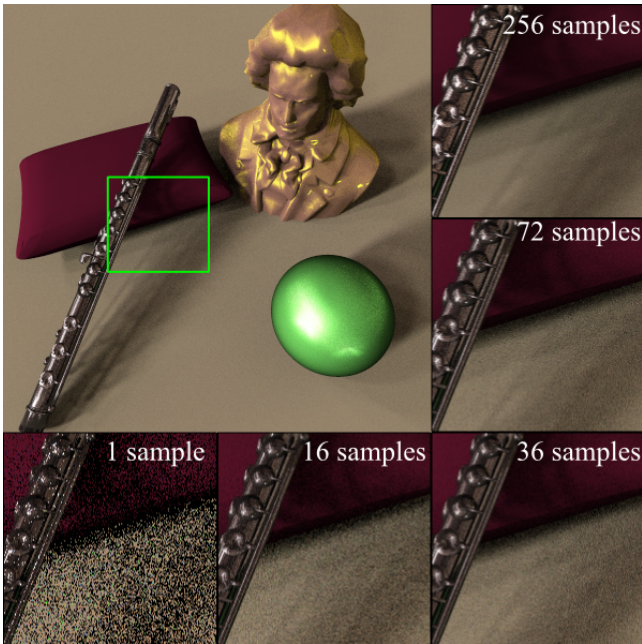


Figure 10: Figure shows images rendered using our sampling algorithm within the "Galileo's Tomb" environment map. Insets show that the variance is tolerable even with few samples and quickly converges as the number of samples is increased.

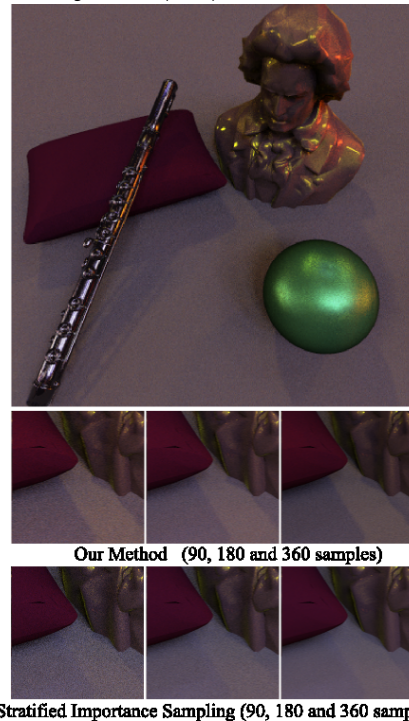


Figure 12: Rendered image of objects of different materials ranging from purely diffuse to purely specular in the Grace Cathedral HDR environment. Inset compares our results with Stratified Importance sampling which does not take the cosine term into account while generating samples.