EUROGRAPHICS 2006 / E. Gröller and L. Szirmay-Kalos
(Guest Editors)

Volume 25 (2006), Number 3

Optimizing Control Variate Estimators for Rendering

Shaohua Fan[1], Stephen Chenney[2], Bo Hu[1], Kam-Wah Tsui[1], Yu-chi Lai[1]

[1]University of Wisconsin – Madison
[2]Emergent Game Technologies

Abstract

We present the Optimizing Control Variate (OCV) estimator, a new estimator for Monte Carlo rendering. Based upon a deterministic sampling framework, OCV allows multiple importance sampling functions to be combined in one algorithm. Its optimizing nature addresses a major problem with control variate estimators for rendering: users supply a generic correlated function which is optimized for each estimate, rather than a single highly tuned one that must work well everywhere. We demonstrate OCV with both direct lighting and irradiance-caching examples, showing improvements in image error of over 35% in some cases, for little extra computation time.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism Color, shading, shadowing, and texture G.3 [Probability and Statistics]: Probabilistic Algorithms

Keywords: *direct lighting, deterministic mixture sampling, control variates*

1. Introduction

Monte Carlo integration methods offer the most general solution to physically accurate lighting simulation: they handle near-arbitrary geometry, material properties, participatory media, etc. All Monte Carlo methods require an *estimator* that takes the information found in the samples and determines a single final value. A good estimator is unbiased and has low variance. In rendering, the unbiased property guarantees the image has on average the correct pixel values, while variance determines the noise levels in the image, or how much neighboring pixels tend to differ in value.

There are many possible estimators, each of which combines the samples in a different way to get the final answer. If we focus on unbiased estimators, then a good strategy is to choose one that minimizes variance while remaining relatively fast to compute. The most common estimator in rendering is the sample mean or an importance weighted mean. Alternatives exist, however, such as the Multiple Importance Sampling (MIS) estimator [VG95] or control variate estimators [SSSK04] (also referred to as correlated sampling).

In this paper we apply an Optimizing Control Variate (OCV) estimator to the problem of estimating irradiance integrals for direct lighting. The same basic problem is also a sub-component of many rendering algorithms, such as irradiance caching and photon-map gathering, for which we also demonstrate some results. The OCV estimator solves a small optimization problem to find a good control variate distribution given a set of samples. Unlike existing control variate methods which require a single control variate distribution for all estimates, OCV allows the distribution to vary over the scene depending on surface properties and lighting conditions. Furthermore, users are not burdened with finding an optimal correlated function; they can provide a generic parameterized function that the estimator optimizes.

OCV works with the *deterministic mixture sampling* (DMS) framework for constructing importance functions, sampling from them, and computing estimates from the samples [OZ00]. In addition to providing better estimators, DMS allows for multiple importance sampling functions to be combined in a general way. The optimizing nature of the estimator ensures that the combination of samplers performs at least as well as the best among them. In this way, OCV can be viewed as a generalization of multiple importance sampling.

2. Estimating Irradiance Integrals

In this paper we concentrate on the problem of computing integrals over hemispheric domains. The most common such integral in rendering computes the radiance, $L(\mathbf{x},\omega)$, leaving a point $\mathbf{x}$ in the direction ω:

$$L(\mathbf{x},\omega) = L_e(\mathbf{x},\omega) + \int_\Omega f(\mathbf{x},\omega,\omega')d\omega' \quad (1)$$

where $L_e(\mathbf{x},\omega)$ is light emitted at $\mathbf{x}$, Ω is the hemisphere of directions *out* of $\mathbf{x}$ and $f(\mathbf{x},\omega,\omega')$ is the light reflected at $\mathbf{x}$ from direction $-\omega'$ into direction ω:

$$f(\mathbf{x},\omega,\omega') = L_{in}(\mathbf{x},-\omega')f_r(\mathbf{x},\omega,\omega')|\cos(\theta')|$$

$L(\mathbf{x},-\omega')$ is light arriving at $\mathbf{x}$ from direction ω', $f_r(\mathbf{x},\omega,\omega')$ is the BRDF, and θ' is the angle between ω' and the normal at $\mathbf{x}$. Monte Carlo renderers use statistical sampling to estimate the integral for the reflected component of $L(\mathbf{x},\omega)$.

A standard importance sampling algorithm for $L(\mathbf{x},\omega)$ samples directions, $\omega'_1,\ldots,\omega'_N$, out of $\mathbf{x}$ according to an importance distribution, p, and computes the estimate:

$$\hat{L}(\mathbf{x},\omega) = \frac{1}{N}\sum_{i=1}^{N}\frac{f(\mathbf{x},\omega,\omega'_i)}{p(\omega'_i)} \quad (2)$$

The variance of this estimator improves as p more closely approximates f, and is zero when p differs from f by a constant scale.

In local direct lighting situations, a common choice for p is a normalized version of $f_r(\mathbf{x}',\omega,\omega')|\cos(\theta')|$ or an approximation to it. We refer to this as BRDF-based importance sampling. An alternative is light-based sampling where the integral is broken into a sum over individual light sources and points are sampled on the lights to generate directions [PH04, §16.1]. In environment map lighting situations, the wavelet product approach of Clarberg et al. [CJAMJ05] currently provides the best way to choose p.

Control variate approaches [Vea97, §2.5.3] introduce a correlated function, g, which should have the property that $f-g$ is close to a constant, and then use the estimator:

$$\hat{L}(\mathbf{x},\omega) = \int_\Omega g(\omega')d\omega' + \frac{1}{N}\sum_{i=1}^{N}\frac{\left(f(\mathbf{x},\omega,\omega'_i)-g(\omega'_i)\right)}{p(\omega'_i)} \quad (3)$$

The difficulty of applying this approach in rendering problems is in finding a function g that is sufficiently close to f in all places. We solve this problem by defining a parameterized function, $g(\omega' : \beta_1,\ldots,\beta_m)$, and optimizing the vector of parameters, $\langle\beta_1,\ldots,\beta_m\rangle$, in order to best approximate f.

The MIS estimator [VG95] uses multiple importance functions, $p_1,\ldots,p_m$, and draws a fixed number of samples from each, $n_1,\ldots,n_m$. It then computes one of several possible estimators, of which the simplest is the *balance heuristic*:

$$\hat{L}(\mathbf{x},\omega) = \frac{1}{N}\sum_{j=1}^{m}\sum_{i=1}^{n_j}\frac{f(\mathbf{x},\omega,\omega'_{i,j})}{\sum_{k=1}^{m}c_k p_k(\omega'_{i,j})} \quad (4)$$

where $c_k = n_j/N$, the proportion of samples drawn from p_j. The major advantage of MIS is that it enables importance functions to be combined in an unbiased manner. Using a slightly different estimator, the *power heuristic*, the weight of samples coming from poor importance functions can be implicitly reduced in the final estimate.

3. Related Work

The simplest effective use of control variates is in cases where the incoming illumination can be approximated by a constant ambient term – Lafortune and Willems [LW94] describe this technique – but it offers less improvement with more complex illumination. Szirmay-Kalos et al. [SKCG01] improve upon this using radiosity to obtain an estimate of the diffuse illumination which serves as the correlated function in a Monte Carlo step that accounts for other illumination. It works well for diffuse environments but not for specular surfaces.

Szécsi et al. [SSSK04] combine control variate and importance sampling estimators (Equations 2 and 3) in a linear combination with weights optimized to reduce variance, but the approach is very limited in the BRDFs that can be handled. Note that this approach combines estimates, not sampling strategies, so a single importance sampling function must still be chosen. An alternate estimator, weighted importance sampling, has been used for particle tracing algorithms by Balázs et al. [BSKG03], but a scene discretization is required and improvement is only seen under specific BRDF and lighting configurations.

The work of Lafortune and Willems [LW95] on adaptive BRDF sampling includes a control variate component. They build a 5D-tree approximation to radiance in the scene, and use it for both importance sampling and control variate estimation. In some sense this is optimizing the control variate estimator. However, large sample counts are required to adequately adapt the necessary functions, and failure to adapt correctly actually increases variance. Our algorithm uses a low-parameter function for the control variate distribution, so few samples are required to optimize.

OCV with deterministic mixture sampling offers a way to combine samples from multiple importance functions. As discussed above, Veach's [VG95] MIS is an existing approach to this problem. DMS includes the balance heuristic (Equation 4) as a special case. We improve upon MIS with a simple optimization process for selecting a better estimator at each pixel.

4. Deterministic Mixture Sampling

The optimizing control variate estimator begins with a deterministic mixture sampling process to generate the samples. This is practically equivalent to MIS's step of generating a fixed number of samples from each of multiple importance functions, but motivated differently.

A mixture probability density function (PDF) is one composed of a weighted sum of component PDFs:

$$p(x:\alpha) = \sum_{j=1}^{m} \alpha_j p_j(x) \tag{5}$$

where m is the number of components and α is a vector of *mixture weights*, $\langle \alpha_1, \ldots, \alpha_m \rangle$, with $\alpha_j > 0$ and $\sum_{j=1}^{m} \alpha_j = 1$. The simplest way to draw a sample from a mixture density is to first select a component, j, with probability $p(j) \propto \alpha_j$, and then sample from $p_j(x)$.

For rendering, the mixture can include any importance function that is typically used alone. Hence, we include a component for sampling according to the BRDF and one for each light source. In environment lighting conditions, a component for sampling the environment map should be included. We could break the BRDF into sub-components (diffuse, glossy, etc.) but we did not experiment with this. Also note that the environment map sampling of Carlberg et al. [CJAMJ05] can be viewed as a mixture where each wavelet basis function is a component.

Deterministic mixture sampling chooses a fixed number of samples from each component: $n_j = N\alpha_j$ samples are drawn from component $p_j(x)$ where N is the total sample size. We can view this as a form of stratification over the mixture components, and Hesterberg [Hes95] shows that this reduces variance. Note that this is exactly what MIS does, and Equation 4 can be re-written in terms of $p(\omega' : \alpha)$:

$$\hat{L}(\mathbf{x},\omega) = \frac{1}{N} \sum_{i=1}^{N} \frac{f(\mathbf{x},\omega,\omega'_i)}{p(\omega'_i : \alpha)} \tag{6}$$

We can also construct a control variate estimate using a mixture of functions as the correlated distribution in addition to the importance distribution [OZ00]:

$$\hat{L}(\mathbf{x},\omega) = \sum_{j=1}^{m} \beta_j + \frac{1}{N} \sum_{i=1}^{N} \frac{f(\mathbf{x},\omega,\omega'_i) - p(\omega'_i : \beta)}{p(\omega'_i : \alpha)} \tag{7}$$

where the β_j are a vector of real valued variables. This estimator is unbiased, as can be seen by writing

$$\begin{aligned} E[\hat{L}_{\alpha,\beta}] &= \int \frac{f(x) - \sum_{j=1}^{m} \beta_j p_j(x)}{p(x:\alpha)} p(x:\alpha)dx + \sum_{j=1}^{m} \beta_j \\ &= \int f(x)dx - \sum_{j=1}^{m} \beta_j \int p_j(x)dx + \sum_{j=1}^{m} \beta_j \\ &= \int f(x)dx \end{aligned}$$

Note that $p_j(x)$ is a PDF so integrates to 1. The variance of the estimator in Equation 7 is

$$\sigma^2_{\alpha,\beta} = \int \left(\frac{f(x) - \sum_{j=1}^{m} \beta_j p_j(x)}{p(x:\alpha)} - I + \sum_{j=1}^{m} \beta_j \right)^2 p(x:\alpha)dx \tag{8}$$

where I is the true value of the integral being estimated.

There is no improvement over importance sampling if we set $\beta_j = \alpha_j$ for all j; it is the same estimator as Equation 6. However, we are free to choose the β_j in a variety of ways – they need not even sum to 1. In particular, we can solve an optimization problem, which gives us an OCV estimator.

5. Optimizing Control Variates

A natural strategy for choosing the β_j is to minimize the variance in Equation 8. We can't do this because we don't know I, the value we are trying to estimate. Instead, we form a linear problem that minimizes the following objective function with respect to the β_j:

$$\sum_{i=1}^{N} \left(\frac{f(X_i) - \sum_{j=1}^{m} \beta_j p_j(X_i)}{p(X_i:\alpha)} \right)^2 \tag{9}$$

This is a standard linear least squares problem, but we modify it in three ways. First, we include an intercept term, β_0 [OZ00], which after optimization evaluates to

$$\frac{1}{N} \sum_{i=1}^{N} \frac{f(X_i) - \sum_{j=1}^{m} \beta_j p_j(X_i)}{p(X_i:\alpha)}$$

Putting β_0 into Equation 7 and simplifying, we get a simpler form of the OCV estimator:

$$\hat{L}(\mathbf{x},\omega) = \beta_0 + \sum_{j=1}^{m} \beta_j \tag{10}$$

The second problem is that the condition $\sum_{j=1}^{m} \alpha_j = 1$ required to make $p(x:\alpha)$ a distribution function means that the $p_j(x)/p(x:\alpha)$ terms are linearly dependent. This can be solved by dropping p_m from the optimization and setting $\beta_m = 0$. This leaves us minimizing $\|\mathbf{y} - \mathrm{A}\beta\|^2$ with

$$\mathbf{y} = \begin{bmatrix} \frac{f(X_1)}{p(X_1:\alpha)} \\ \vdots \\ \frac{f(X_N)}{p(X_N:\alpha)} \end{bmatrix}$$

$$\mathrm{A}\beta = \begin{bmatrix} 1 & \frac{p_1(X_1)}{p(X_1:\alpha)} & \cdots & \frac{p_{m-1}(X_1)}{p(X_1:\alpha)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \frac{p_1(X_N)}{p(X_N:\alpha)} & \cdots & \frac{p_{m-1}(X_N)}{p(X_N:\alpha)} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{m-1} \end{bmatrix}$$

A further problem occurs when all of the samples from some component are zero. In rendering this is quite likely due to occlusion or some other factor that gives zero radiance from some directions. To deal with this we use *penalized least squares* with a penalty term pushing the β_i toward zero. The resulting objective function is $\|\mathbf{y} - \mathrm{A}\beta\|^2 + \lambda\|\beta\|^2$. The solution to this problem is

$$\hat{\beta} = \left(\mathrm{A}'\mathrm{A} + \lambda \mathrm{I}\right)^{-1} \mathrm{A}'\mathbf{y} \tag{11}$$

where A' is the transpose of A and I is the identity matrix. We found $\lambda = 1$ to be good in practice.

5.1. OCV for Rendering

Optimizing control variate estimation is useful in rendering when evaluating integrals over a single domain, with the same PDF used for each sample, and a choice of importance functions. While Veach [VG95] showed a bi-directional path tracing application, in practice the conditions are met in *gather* integrals where we integrate incoming irradiance at a point by sampling over the hemisphere. Such integrals arise in direct lighting, irradiance caching, photon-mapping, and radiosity. We show examples from the first two applications.

Apart from choosing components for the mixture, we must also set their weights, α_i. In all our experiments we used a single BRDF-based component and one component for each light (we did not use environmental lighting). We made a conservative choice: half of the samples came from the BRDF, $\alpha_{BRDF} = 0.5$, while the remainder were divided equally among the lights. If for some reason a user thought some sampling function was more likely to succeed then the weight for that component could be increased.

To summarize, each time we require an estimate of the integral in Equation 1, we draw a fixed number of direction samples, n_j, from each importance function in the mixture, p_j. We trace rays for each sample to determine the incoming radiance, $L_{in}(\mathbf{x}, -\omega_i')$. With each sample direction evaluated, we form the matrices and vectors and solve Equation 11 for the β_j. Finally, Equation 10 is evaluated to compute the estimate of outgoing radiance.

In direct lighting, an irradiance integral estimate is obtained for every surface point hit with a pixel sample. For irradiance caching, another application we have implemented, the incoming irradiance must be estimated at diffuse surface points when a nearby cached estimate is not available. The irradiance integral is broken into two terms:

$$Ir(\mathbf{x}) = \int_{\Omega} L_{sources}(\mathbf{x}, -\omega')d\omega' + \int_{\Omega} L_{ind}(\mathbf{x}, -\omega')d\omega'$$

where $Ir(\mathbf{x})$ is the irradiance at point $\mathbf{x}$, $L_{sources}$ is incoming radiance due to light or environmental sources, and L_{ind} is radiance due to indirect lighting. In our implementation [PH04], $L_{ind}(\mathbf{x}, -\omega')$ is computed using path tracing, but each point along the path also evaluates the direct lighting integral.

We use OCV only for the irradiance due to sources. OCV is less readily applied to the integration of indirect illumination because BRDF-based sampling is the only viable distribution and hence there is no simple way to form a mixture. Note, however, that direct lighting is evaluated as part of the path tracing procedure, so OCV does still contribute to indirect illumination at the pixel. It may be worthwhile to apply OCV for indirect illumination using a mixture of BRDF components (diffuse, glossy, etc.) or a mixture of basis functions over the hemisphere, but we have not experimented with this.

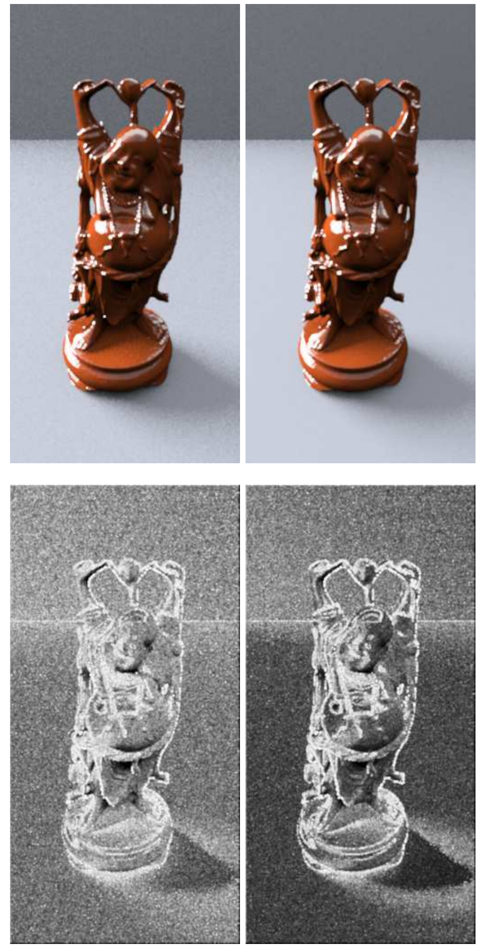

Figure 1: *Results for MIS and OCV for the Buddha model. MIS, left, has noticeably higher variance in the soft shadow boundary and the base of the Buddha. The variance images, below, reveal significant reduction in variance with OCV over the entire image.*

6. Results

We first experimented with a scene (Figure 2) that demonstrates the importance of including multiple sampling functions for direct lighting (following [PH04]). This example contains two lights, so half of all the samples come from sampling a BRDF-based component, while one quarter come from sampling the area of the yellow light and a quarter from the blue light. Table 1 presents timing and error results, where error is a perceptually weighted error metric:

$$E = \left[\frac{1}{n} \sum_{pixels} \left(\frac{L - L_{true}}{tvi(L_{true})} \right)^2 \right]^{\frac{1}{2}} \tag{12}$$

where n is number of pixels, L is the luminance of the result, L_{true} is the true luminance, and $tvi(x)$ is the perceptual threshold-vs-intensity function introduced by Ferwerda

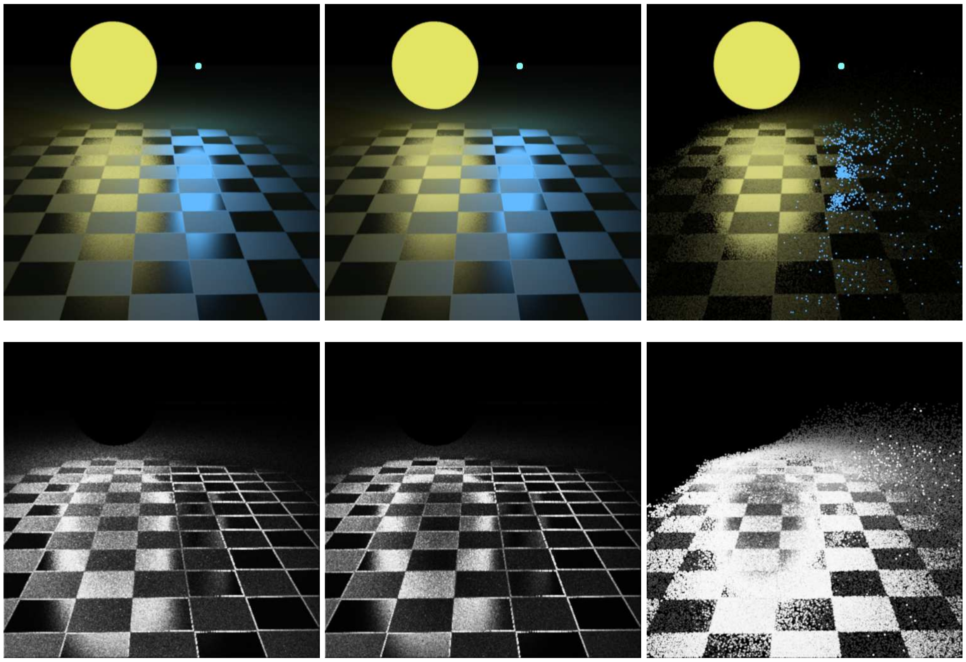

Figure 2: *Images for the checkers scene. Left is MIS, center is OCV and right is correlated sampling. Correlated sampling performs poorly because it must choose only one importance function before rendering begins (typically BRDF-based, as we have here) and the best choice is not always obvious. Bottom are perceptually-based variance images, which show the variance of the direct illumination estimates obtained at each pixel. The most significant improvement of OCV over MIS is apparent within the left glossy reflection of the large light source. Note that variance is expected to be large at material property boundaries because different pixel samples are hitting different materials.*

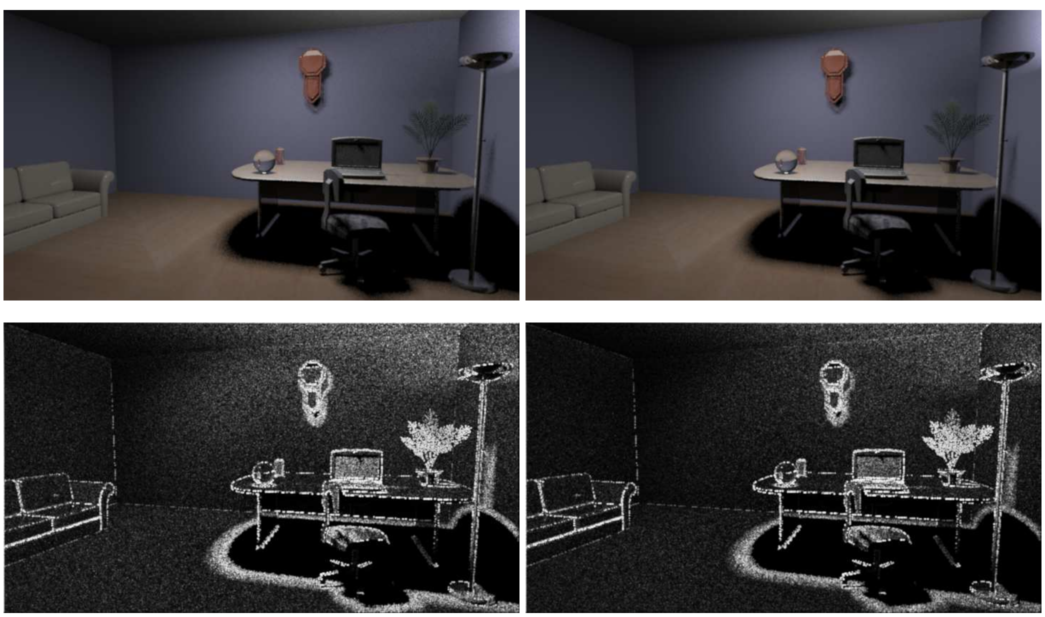

Figure 3: *Results for MIS (left) and OCV (right) for the room scene. The images are very similar, but the variance images below reveal an overall improvement with OCV over MIS.*

Image	Method	SPE	SPP	Time (s)	Err
Checks	MIS	64	4	172.8	0.60
	OCV	64	4	180.8	0.48
Buddha	MIS	64	4	98.3	0.72
	OCV	64	4	105.6	0.46
Room	MIS	18	2	37.4	0.75
	OCV	18	2	43.2	0.68
Box	MIS	18	9	196.5	4.9
	OCV	18	9	207.2	4.0

Table 1: *Measurements comparing MIS to OCV for direct lighting computations. SPE is the sample count per estimate, with SPP estimates per pixel. Err is the error computed using Equation 12.*

et al. [FPSG96]. We use perceptual weighting to avoid giving too much weight to very bright or very dark areas of the image. The ground truth image is computed using MIS running for several hours.

We compare the algorithms based on the same number of samples instead of the same computational time, because pre-setting the number of samples allows us to use comparable stratification.

Figure 2 shows comparison between MIS, OCV and the correlated sampling approach of Szécsi et al. [SSSK04]. These images were rendered at 500×500 resolution. They highlight primarily the value in using multiple importance functions, which correlated sampling cannot do. OCV performs better than MIS on this scene with little additional computation time. Improvement in the form of lower variance is most apparent in the glossy region reflected in the yellow light. In this scene the OCV estimator results in a 18% improvement in image quality with about 5% more computation time.

The Buddha images (Figure 1) show a more marked improvement with OCV over MIS. These images were rendered at 256×512 resolution, and the OCV estimator results in a 37% improvement for 7% more time. This scene has a greater variety of lighting conditions, ranging from tight specularities to occluded regions. Our final direct lighting test used a room scene (Figure 3), for which the OCV estimator produced lower error compared to MIS, but the additional computation cost resulted in comparable rendering efficiency. The scene requires relatively few samples to obtain a good estimate because the light sources are small and there is limited occlusion. Our method performs best when occlusion is complex and with larger light sources. Still, due to the optimization in OCV the results are unlikely to be worse than alternate methods.

The Cornell Box scene (Figure 4) demonstrates OCV estimates in irradiance caching. The perceptual RMS error (Equation 12) for the standard implementation is 4.9, which OCV reduces to 4.0 with about 5% more computational time.

6.1. Limitations

The primary limitation with the OCV estimator comes from the relationship between the number of components in the mixture and the number of samples required. A larger mixture requires more samples to obtain reliable values for optimized β – at least as many samples as components. Furthermore, more mixture components and samples increases the cost of the optimization, to the extent that MIS would perform better for the same computation time. Hence, very small sample counts (less than about 10) cannot be used and situations with many light sources cause problems, at least as we have constructed the mixture. In a many-light situation, nearby lights could be grouped into one component or an environmental lighting approach could be used.

We do not use OCV for the indirect lighting component of the irradiance caching integral because our techniques for forming a mixture result in a single component. We could form a mixture by sub-dividing the hemisphere and using one component for each sub-region. This would allow things such as occluded paths to be accounted for in the estimator.

As stated above, an OCV estimator is only useful in situations when all the samples come from the same mixture distribution. In bi-directional path tracing, this means we can only use it on a per-path basis with a mixture component for each method of forming the path. Path tracing is ruled out because each path has a different length and hits a different set of material properties, and hence has a different PDF. Integrals of the form in Equation 1 are very common, however, so OCV does cover a large set of practical cases.

7. Conclusion

We have presented a new estimator for use in computing irradiance gather integrals. The OCV estimator maximizes the benefits of control variate sampling by optimizing the correlated function at each estimate. This also reduces the user's burden of finding correlated functions. In addition, OCV allows multiple importance functions to be combined, which is particularly useful when no one function works well across an entire image.

In importance sampling applications, one use of mixtures is in *defensive* sampling [Hes95], where one component of the mixture is certain to have "heavier tails" than the integrand to ensure finite variance of the estimate. In rendering, situations where a defensive component is useful are rare: one example is a glossy surface under environmental lighting where the dominant reflectance lobe is blocked by an occluder, and wavelet product sampling is in use. A cosine-weighted mixture component could be used as a defensive choice in such situations.

There are several alternate importance functions that could be used as components. One particularly interesting possibility is using the low-frequency wavelets from

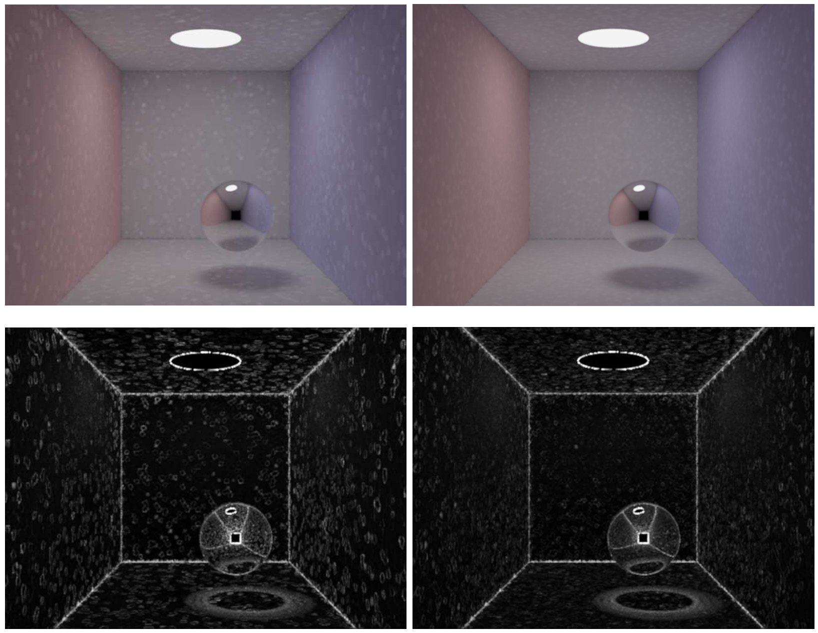

Figure 4: *Results for MIS and OCV for irradiance caching computations on a box scene. Standard irradiance caching, which uses MIS for its estimates, is on the left, while a version using OCV estimators is on the right.*

Carlberg et al. [CJAMJ05]. The potential advantage is that wavelets representing occluded directions could have their weight in the estimate reduced. Even more advantage could come from an approach that adapts the mixture weights, and hence avoids any sampling in occluded directions.

References

[BSKG03] BALÁZS B., SZIRMAY-KALOS L., GYÖRGY A.: Weighted importance sampling in shooting algorithms. In *Proceedings of the Spring Conference on Computer Graphics* (2003), pp. 177–184.

[CJAMJ05] CLARBERG P., JAROSZ W., AKENINE-MÖLLER T., JENSEN H. W.: Wavelet importance sampling: efficiently evaluating products of complex functions. In *SIGGRAPH '05* (2005), pp. 1166–1175.

[FPSG96] FERWERDA J. A., PATTANAIK S. N., SHIRLEY P., GREENBERG D. P.: A model of visual masking for computer graphics. In *SIGGRAPH '96* (1996), pp. 249–258.

[Hes95] HESTERBERG T.: Weighted average importance sampling and defensive mixture distributions. *Technometrics 37* (1995), 185–194.

[LW94] LAFORTUNE E. P., WILLEMS Y. D.: The ambient term as a variance reducing technique for Monte Carlo ray tracing. In *Photorealistic Rendering Techniques (Proceedings of the Fifth Eurographics Workshop on Rendering)* (1994), pp. 168–176.

[LW95] LAFORTUNE E. P., WILLEMS Y. D.: A 5D tree to reduce the variance of Monte Carlo ray tracing. In *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)* (1995), pp. 11–20.

[OZ00] OWEN A., ZHOU Y.: Safe and effective importance sampling. *Journal of the American Statistical Association 95* (2000), 135–143.

[PH04] PHARR M., HUMPHREYS G.: *Physically Based Rendering from Theory to Implementation*. Morgan Kaufmann, 2004.

[SKCG01] SZIRMAY-KALOS L., CSONKA F., GYÖRGY A.: Global illumination as a combination of continuous random walk and finite-element based iteration. *Computer Graphics Forum (Proceedings of Eurographics 2001) 20*, 3 (2001), 288–298.

[SSSK04] SZÉCSI L., SBERT M., SZIRMAY-KALOS L.: Combined correlated and importance sampling in direct light source computation and environment mapping. *Computer Graphics Forum (Proceedings of Eurographics 2004) 23*, 3 (2004), 585–593.

[Vea97] VEACH E.: *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University, 1997.

[VG95] VEACH E., GUIBAS L. J.: Optimally combining sampling techniques for Monte Carlo rendering. In *SIGGRAPH '95* (1995), pp. 419–428.