Vis Comput
DOI 10.1007/s00371-013-0908-z

ORIGINAL ARTICLE

Robust and efficient adaptive direct lighting estimation

Yu-Chi Lai · Hsuan-Ting Chou · Kuo-Wei Chen · Shaohua Fan

© Springer-Verlag Berlin Heidelberg 2014

Abstract Hemispherical integrals are important for the estimation of direct lighting which has a major impact on the results of global illumination. This work proposes the population Monte Carlo hemispherical integral (PMC-HI) sampler to improve the efficiency of direct lighting estimation. The sampler is unbiased and derived from the population Monte Carlo framework which works on a population of samples and learns to be a better sampling function over iterations. Information found in one iteration can be used to guide subsequent iterations by distributing more samples to important sampling techniques to focus more efforts on the sampling sub-domains which have larger contributions to the hemispherical integrals. In addition, a cone sampling strategy is also proposed to enhance the success rate when complex occlusions exist. The images rendered with PMC-HI are compared against those rendered with multiple importance sampling (Veach and Guibas In: SIGGRAPH '95, pp 419–428, 1995), adaptive light sample distributions (Donikian et al. IEEE Trans Vis Comput Graph 12(3):353–364, 2006), and multidimensional hemispherical adaptive sampling (Hachisuka et al. ACM Trans Graph 27(3):33:1–33:10, 2008). Our PMC-HI sampler can improve rendering efficiency.

Keywords Ray-tracing · PMC · Monte Carlo · Global illumination

1 Introduction

When rendering a physical scene, we can separate the lighting effect into two components: direct lighting and indirect lighting. Direct lighting controls the main theme of the rendered image and is the main factor of the final result. Indirect lighting affects the small details of the result such as color bleeding and is important to perceptual correctness. Traditionally, direct lighting is estimated using Monte Carlo methods, but how to increase the efficiency of rendering is still the main research topics in the global illumination community. This work presents a sampling technique that significantly improves the rendering efficiency for direct lighting. The sampler is derived from the population Monte Carlo (PMC) sampling framework, which is a technique that adapts sampling distributions over iterations according to the information collected in previous iterations.

Traditionally, the primary tool for direct lighting estimation is importance sampling based on the surface bidirectional reflectance distribution functions (BRDFs) [15] or light sources [1]. And the choice of the importance functions has large influence on the estimation. It is hard to find a good important sampling function for all shading points. Although multiple important sampling (MIS) [20] can relieve this issue with a weighting scheme for samples generated from candidate functions which may be possibly good, MIS may waste precious samples on those not-ideal candidate importance functions. There are adaptive algorithms [5,10] which dynamically adapt the sample distribution function for

Funded by: NSC 99-2218-E-011-005-, Taiwan.

Y.-C. Lai · H.-T. Chou · K.-W. Chen
National Taiwan University of Science and Technology, Taipei, Taiwan, ROC
e-mail: yu-chi@mail.ntust.edu.tw

H.-T. Chou
e-mail: terry60103@gmail.com

K.-W. Chen
e-mail: chen51202@gmail.com

S. Fan (✉)
China and NYU Courant Institute of Mathematical Sciences, Suzhou University, Suzhou, China
e-mail: shaohua@cs.wisc.edu

Published online: 19 January 2014

direct lighting estimation. However, these methods may be sub-optimal for the usage of the light sampling strategy or the requirement of extra efforts for the construction and update of an auxiliary data structure. Therefore, this work proposes the population Monte Carlo hemispherical integral (PMC-HI) sampler to dynamically adjust the choice of importance functions in order to improve sampling efficiency. The sampler iterates on a population of samples which are initialized using stratified MIS sampling on each light to estimate the initial hemispheric integrals on a point. Any information available at this stage can then be used to dynamically adapt a *kernel function* that produces a new population to approach the ideal importance functions. The sampler can, for instance, avoid over-sampling a light source from a surface point within its shadow, or a BRDF specular lobe that makes no contribution. Furthermore, the sampler can also guide samples toward important illumination directions found by previous samples, without adding bias. Our PMC-HI do not require an extra data structure in the adaptation process, has low adaptation cost and can generate good results with minimum extra cost. At the end, we have compared the images rendered with PMC-HI against those rendered with MIS [20], and adaptive light sample distribution (ALSD) [5], and multidimensional hemispherical adaptive sampling (MDHI) [10] on several scenes. PMC-HI can have improvement in rendering efficiency.

The remainder of this paper is organized as follows: Sect. 2 reviews a number of work related to this algorithm. Section 3 gives a short overview of the population Monte Carlo method. Section 4 presents the concept and detail of the PMC-HI sampler. Section 5 demonstrate the applications of the PMC-HI sampler. Section 6 gives a short discussion of using the PMC-HI sampler. Finally, Sect. 7 gives the conclusion of our algorithm.

2 Related work

Here we focus on a specific area of related work: sampling for hemispheric integrals. For an overview of Monte Carlo rendering in general, see Pharr and Humphreys [15].

There is a large body of work on computing hemispheric integrals (direct lighting), mostly concerned with importance sampling functions. Veach's thesis [19] provides a description of the basic methods and analysis of variance. Importance functions are commonly based on surface BRDFs [15] or light sources [1]. Recent advances include wavelet-based importance functions for environmental lighting [4], and resampling algorithms [2,18] that avoid visibility queries for samples that are likely to be unimportant. However, the former is applicable only to environment maps, while the latter throws away samples and still requires a-priori choice of importance functions. No existing importance sampling approach for hemispheric integrals offers adaptable importance functions. Multiple important sampling (MIS) [20] is the general choice for alleviating this problem by generating a sample based on light sources and the surface BRDF and the estimated direct lighting is weighed based on its sampling probability of both methods. It generally generate good results but the sampling mechanism may have chances to waste samples on the low-contribution sampling functions.

Work on adaptive PDFs for importance sampling has focused on path tracing or irradiance caching applications. Dutré and Willems [6] used piecewise linear functions to determine shooting directions out of light sources in a particle tracing application. Dutré and Willems [7] used piecewise constant functions and Pietrek and Peter [16] used wavelets to build adaptive PDFs for sampling gather directions in path tracing. A diffuse surface and piecewise constant PDF assumption is required to reduce the number of coefficients to a manageable level, and even then very high sample counts are required. It is important to note that a bad approximation can *increase* variance. Lafortune and Willems [11] used a 5D tree to build an approximation to radiance in the scene, and then use it for importance sampling in a path tracing framework. The same problems with sample counts and approximation errors arise in their work. Our algorithm works with arbitrary BRDFs and uses a low-parameter adaptive model to minimize the sample count required to control adaption.

Adaptive algorithms have also been suggested for shadow computations. Ward [21] proposed an algorithm for scenes with many lights, where shadow tests for insignificant lights are replaced by probabilistic estimates. Ward's approach works best with many light sources (tens or hundreds) while our technique works best with few sources. Ohbuchi and Aono [14] adaptively sampled an area light source (which introduces bias). They achieved good stratification by employing quasi-Monte Carlo (QMC) techniques to place the samples. Donikian et al. [5] introduced the multi-pass adaptive light sample distribution(ALSD) algorithm to dynamically adapt the sample distribution function for each light based on its contribution to direct lighting in a many-light scene. The light sampling strategy may miss important features of BRDFs and the variance of estimation grows when the portion of glossy surface increases. Therefore, our algorithm do the adaptation in choosing the sampling strategy among the BRDFs and light sources to lower the variance. Hachisuka et al. [10] has proposed multidimensional adaptive light sampling (MDHI) to adapt sample distribution according to the variance of samples and then reconstruct a function for the estimation of hemispherical integrals. However, the cost of building a Kd-Tree and looking up a node through the Kd-Tree is large and thus, the improvement in rendering efficiency is low. Therefore, this work proposes PMC-HI to dynamically adjust the choice of importance functions in order to improve sampling efficiency. The frame-

work has low adaptation cost without the need of extra data structure and the rendered results are generally good.

A sequential Monte Carlo algorithm, similar in spirit to population Monte Carlo, has recently been applied by Ghosh et al. [9] to the problem of sampling environment maps in animated sequences. Their work exploits another property of iterated importance sampling algorithms—the ability to re-use samples from one iteration to the next—and is complementary to our approach. Population Monte Carlo is another class of iterative sampling methods and has been introduced into the rendering community for adaptively distributing image-plane samples [12] and adjusting the perturbation mechanism of a Markov chain during energy redistribution process [13]. This work adapted from the corresponding author's thesis [8] follows the same iterative adaptation concept to choose a good sampling function for hemispherical integration based on the information collected in the previous iteration.

3 Population Monte Carlo (PMC)

The population Monte Carlo algorithm [3] as shown in Fig. 1 is an iterated importance sampling scheme. Assume that we have a population of samples denoted by $\{X_1^{(t)}, \ldots, X_N^{(t)}\}$, where t is the iteration number and N is the population size, and we wish to sample according to the distribution proportional to $f(x)$. Line 1 creates the population with a known sampling algorithm to jump-start the algorithm. In each iteration of the algorithm, a *kernel function*, $K^{(t)}(x^{(t)}|x^{(t-1)})$, is determined (line 3) using information from the previous iterations. The kernel function is responsible for generating the new population, given the current one. It takes an existing sample, $X_i^{(t-1)}$, as input and produces a candidate new sample, $\hat{X}_i^{(t)}$, as output (line 5). The distinguishing feature of PMC is that the kernel functions are modified after each iteration based on information gathered from prior iterations. The kernels adapt to approximate the ideal importance function based on the samples seen so far. The weight computed for each sample, $w_i^{(t)}$, is essentially its importance weight. The resampling step in line 7 is designed to cull candidate samples with low weights and promote high-weight samples. Resampling is not always necessary, particularly if the kernel is not really a conditional distribution. In our PMC-HI algorithm, we did not use the resampling step. Fan [8] gives a detailed discussion about the analysis of unbiasedness, consistency and variance for the PMC framework and interesting readers can refer to the thesis for more details. Several steps are required to apply PMC to rendering problems:

1 generate the initial population, $t = 0$
2 **for** $t = 1, \cdots, T$
3 adapt $K^{(t)}(x^{(t)}|x^{(t-1)})$
4 **for** $i = 1, \cdots, N$
5 generate $\hat{X}_i^{(t)} \sim K^{(t)}(x|X_i^{(t-1)})$
6 $w_i^{(t)} = f(\hat{X}_i^{(t)})/K^{(t)}(\hat{X}_i^{(t)}|X_i^{(t-1)})$
7 resample according to $w_i^{(t)}$ for the new population

Fig. 1 The generic Population Monte Carlo algorithm

- Decide the sampling domain and population size. Computational concerns and stratification typically drive the choice of the domain.
- Define kernel functions and their adaption criteria. This is the most important task, and we give examples for our applications and suggest some general principles in the discussion. For rendering applications two key concerns are the degree to which the kernel supports stratification and whether it works with a small population size (as low as 4 in our hemispheric integrals sampler).
- Choose the techniques for sampling from the kernel functions and the resampling step. The deterministic sampling we use significantly reduces variance much like stratification.

The following sections describe our sampler in detail, and a general discussion on PMC for rendering problems before we conclude with results.

4 PMC-HI: adaptive hemispheric integral sampling

Hemispheric integral samplers generate incoming directions, ω', at a surface point, $\mathbf{x}$. One application is in direct lighting, which assumes that the light leaving a surface point, $L(\mathbf{x}, \omega)$, can be evaluated by the following integral, composed of terms for light emitted from and reflected at $\mathbf{x}$:

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

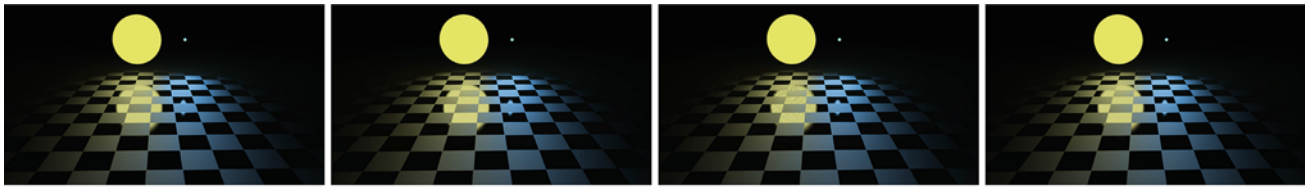
where $L_e(\mathbf{x}, \omega)$ is light emitted at $\mathbf{x}$, Ω is the hemisphere of directions *out* of $\mathbf{x}$ and $f(\mathbf{x}, \omega, \omega')$ is the light reflected at $\mathbf{x}$ from direction $-\omega'$ into direction ω:

$$f(\mathbf{x}, \omega, \omega') = L_{\text{in}}(\mathbf{x}, -\omega') f_r(\mathbf{x}, \omega, \omega') |\cos(\theta')| \tag{2}$$

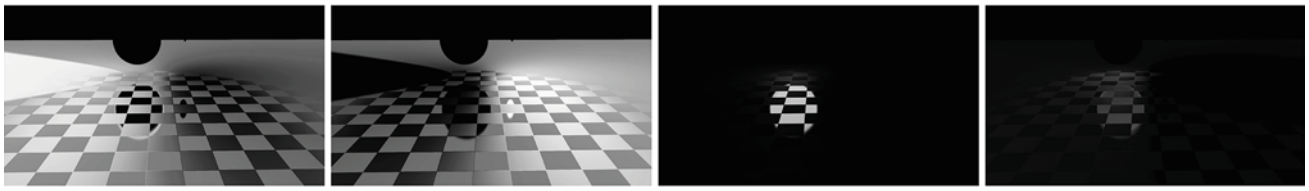
where $L(\mathbf{x}, -\omega')$ is the light arriving at $\mathbf{x}$ from direction ω', $f_r(\mathbf{x}, \omega, \omega')$ is the BRDF, and θ' is the angle between ω' and the normal at $\mathbf{x}$.

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 function, p, and computes the estimate:

$$\hat{L}(\mathbf{x}, \omega) = \frac{1}{n} \sum_{i=1}^{n} \frac{f(\mathbf{x}, \omega, \omega'_i)}{p(\omega'_i)} \tag{3}$$



(a) Cheker rendered results



(b) Component sampling weights

Fig. 2 **a** The rendered results of the Checkerboard scene using PMC-HI, ALDS, MDHI and MIS. This is a scene constructed to demonstrate how the optimal sampling strategy varies over an image. The checkers contains diffuse and glossy squares, with near-pure specular toward the back and rougher toward the front. There are two light sources. **b** Are maps which show how the mixture component weights for PMC-HI vary over the image, after two iterations. Bright means high weight. From *left* to *right*: $\alpha^{(2)}_{L1}$, the left light's weight; $\alpha^{(2)}_{L2}$, the right light's weight; $\alpha^{(2)}_{\text{BRDF}}$; and $\alpha^{(2)}_{\text{cone}}$, which can show its usage for generating more samples for glossy surfaces. The large light dominates in regions where no light is seen in a glossy reflection, while the right light is favored in nearby diffuse squares. The BRDF component is favored only when the large light is specularly reflected at a pixel. The images are quite noise-free for such small sample counts (32 total samples per estimate), indicating that the adaption mechanism converges to a consistent result

The variance of this estimator improves as p more closely approximates f, and is zero when p is proportional to f.

In the local direct lighting situation, one common choice for p is proportional to $L_{\text{in}}(\mathbf{x}, -\omega')\, f_{\text{r}}(\mathbf{x}, \omega, \omega')|\cos(\theta')|$ or a normalized approximation to it. An alternative is to break the integral into a sum over individual light sources and sample points on the lights to generate directions [15, §16.1]. In an environment map lighting situation, the wavelet product approach of Clarberg et al. [4] currently provides the best way to choose p. However, none of these individual importance functions behaves well in all cases.

Figure 2a demonstrates the various difficult cases for importance sampling. The floor consists of a checker pattern with diffuse and glossy squares (with two types of gloss settings). There are two lights, one large and one small. In diffuse pixels, an importance function based on the lights is best. In highly glossy pixels that reflect the large light, BRDF sampling is best. For glossy pixels that do not reflect light, light sampling is best, and rough glossy pixels benefit from both BRDF and light sampling, but we have no way of knowing this a-priori, and most practitioners would use BRDF sampling. In rough glossy regions that reflect only one light, sampling from the other light is wasteful, but again most algorithms would sample equally or according to total emitted power.

Multiple importance sampling (MIS) and bidirectional importance sampling address many of these problems, by trying several importance functions and combining their results. While this does very well at reducing variance, it is wasteful in cases where one of the importance functions is much better than the others and could be used alone. Other techniques assume knowledge of which strategy will dominate where.

PMC-HI is a sampler that generates directions out of a point by adapting a kernel function to match the integrand of interest—$L_{\text{in}}(\mathbf{x}, -\omega')\, f_{\text{r}}(\mathbf{x}, \omega, \omega')|\cos(\theta')|$ in the direct lighting case. For example, the lower images in the bottom in Fig. 2 indicate the relative usefulness of different importance functions at each pixel. Furthermore, the PMC framework enables important samples from one iteration to guide sampling in subsequent iterations.

4.1 The PMC-HI Kernel function

Each direct lighting estimate takes place at a single surface point and is only one small step in a larger computation. The same surface point, and hence the same target function, f_{r}, essentially never re-appears. We choose to adapt on a per-estimate basis, which avoids the need to store information about the adaptation state at the surface points and interpolate to find information at new points. Hence, the number of samples on which to base adaption is low and the number of samples on which to consistent results is high. At one shading point, the mixture is

$$\begin{aligned} K^{(t)}_{\text{HI}}(\omega^{(t)}|\mathbf{d}^{(t)}, \beta^{(t)}) = {} & \alpha^{(t)}_{\text{BRDF}} h_{\text{BRDF}}(\omega^{(t)}) \\ & + \sum_{l} \alpha^{(t)}_{\text{light}_l} h_{\text{light}_l}(\omega^{(t)}) \\ & + \alpha^{(t)}_{\text{cone}} h_{\text{cone}}(\omega^{(t)}|\mathbf{d}^{(t)}, \beta^{(t)}) \end{aligned} \tag{4}$$

where h_{BRDF} is the BRDF-based importance function which should be a good approximation to f_{r}, h_{light} is the light-source-based importance function for each light and h_{cone} is the importance function which samples based on important sample directions from the previous iteration and used to capture good sampling directions. The cone function samples a direction uniformly within a cone of directions with axis $\mathbf{d}^{(t)}$ and half-angle $\beta^{(t)}$, which is set based on the population in the previous iteration. The motivation for this term is because when complex occlusions exist, the estimation may still have a high failure rate for those samples which choose the right importance sampling functions. Therefore, it is helpful to generate samples based on those successful generated samples to increase the success rate. It is particularly useful for situations like partial shadowing where previous samples that found visible portions of the light generate more samples that also reach the light.

The population in PMC-HI is a set of sample directions out of the surface point we are estimating. The population size must be large enough to obtain reasonable estimates for the $\alpha_k^{(t)}$ values at each iteration but not so large as to increase computation time unnecessarily. We typically use $N = 2m$, where m is the number of mixture components. This is a sufficient size to see the benefits of adaption.

4.2 Adapting for PMC-HI

An initial population of N samples, $\{\omega_1'^{(0)}, \ldots, \omega_{n_0}'^{(0)}\}$, is generated using $\alpha_{\text{cone}}^{(0)} = 0$ and the other $\alpha_k^{(0)}$ equal and summing to one. A deterministic mixture sampling is used to select the number of samples from each component. Each sample is tagged with the mixture component that was used to generate it, and their importance weights are computed:

$$w_i^{(0)} = \frac{f\left(\mathbf{x}, \omega, \omega_i'^{(0)}\right)}{K_{\text{HI}}^{(0)}\left(\omega_i'^{(0)}\right)} \tag{5}$$

There is no resampling step for direct lighting. The sample size is so small that resampling tends to unduly favor high-weight directions at the expense of others, thus reducing the degree to which sampling explores the domain. Instead, the cone mixture component is used to incorporate the information from previous samples.

The new component weights, $\alpha_k^{(1)}$, can now be determined along with the $\mathbf{d}^{(1)}$ and $\beta^{(1)}$ parameters for h_{cone}. The cone direction $\mathbf{d}^{(1)}$ is found by taking a weighted average of the $t = 0$ population samples, with weights $w_i^{(0)}$. The cone size is set to the standard deviation of those samples. The component weights are set based on the sample importance weights:

$$\alpha_k^{(t)} = \frac{\sum_{i \in \mathscr{S}_k} w_i^{(t-1)}}{\sum_{j=1}^{n} w_j^{(t-1)}} \tag{6}$$

where $\mathscr{S}_k$ is the set of samples that were generated using component k. In the first iteration there is no sample from the cone perturbation, so we set $\alpha_{\text{cone}}^{(1)} = 0.2$ and adjust the other α's by a factor of 0.8 to make them all sum to one.

We now begin the next iteration. A new set of samples is generated using deterministic mixture sampling from the kernel $K_{\text{HI}}^{(t)}(\omega^{(t)}|\mathbf{d}^{(t)}, \beta^{(t)})$, weights are computed, and the kernel function is updated based on the weights. To form the estimate, we use Eq. 3, with each sample, $\omega_i'^{(t)}$, weighted by $w_i^{(t)}$ from Eq. 5.

5 Results and comparisons

All statistics provided in this section are done with a personal laptop whose CPU is Intel i7-3517U 2.4 GHz with 4 GB DDR3 memory. This section will present the results of using our PMC-HI to estimate the direct lighting of several scenes. These results are compared with multiple importance sampling (MIS) [20], adaptive light sample distribution (ALSD) [5], and multidimensional hemispherical adaptive sampling (MDHI). We choose to have the same number of samples per pixel for all operators and manipulate the number of shadow rays per light to let the running time of all operators be almost the same.

We choose to test our algorithm in four test scenes: Checkerboard (CK), Budda (BU), Jack-o-lattern (JA) and Yearright (YR). CK, BU, JA and YR are rendered with a resolution of 1,920 × 1,080, 1,080 × 1,920, 1,920 × 1,080 and 1,920 × 1,080. The timing and error comparisons with MIS, ALDS and MDHI appear in Table 1. The results show that PMC-HI gains an improvement in rendering efficiency over all algorithms. The performance of MDHI is generally not good due to the requirement of extra efforts for the construction and update of a Kd-tree. Figure 2 shows the rendered result of the CK scene using PMC-HI, ALSD, MDHI and MIS. ALSD generates direct lighting samples based on the light sampling strategy and thus, it does not perform well in the glossy regions of the Checkerboard scene. Figure 3 shows the cropped results of the area marked with the red rectangle using PMC-HI, ALSD, MDHI and MIS. The PMC-HI operator can get much better results in the glossy regions than ALSD do because the sampling strategy for each shading point can be adjusted according to the lighting configuration and surface material properties. Our sampler can also improve rendering efficiency over MIS because less samples are distributed to less important sampling strategies. Figure 4 shows the rendered results of the BU scene using PMC-HI, ALSD, MDHI and MIS. ALSD generates direct lighting samples based on the light sampling strategy and thus, it does not perform well in the glossy regions such as the head of the budda. The PMC-HI operator can get better results for regions in partial shadowing as shown in the cropped images

Table 1 This shows the statistics of measurements when rendering Checkerboard (CK), Budda (BU), Jack-o-lattern (JO) and Yearright (YR) with MIS [20], ALSD [5] and MDHI and the PMC-HI sampler

Image	Method	# SPP	# SRL	T(s)	Err	Eff
CK	MIS	8	18	873.67	2.93e−3	0.391
	ALSD	4–4	12	906.59	1.56e−2	0.071
	MDHI	8	2–2	1,021.98	8.53e−3	0.115
	PMC-HI	8	4–12	846.12	1.37e−3	0.863
BU	MIS	8	10	585.29	1.43e−2	0.119
	ALSD	4–4	4	525.72	7.17e−2	0.026
	MDHI	8	8–8	565.08	1.04e−1	0.017
	PMC-HI	8	3–5	516.29	2.06e−3	0.94
JA	MIS	8	18	1,239.09	6.84e−3	0.117
	ALSD	4–4	16	1,271.58	4.79e−3	0.164
	MDHI	8	5–5	1,335.91	1.17e−1	0.006
	PMC-HI	8	4–12	1,211.12	3.79e−3	0.217
YR	MIS	16	18	1,359.58	2.81e−4	2.62
	ALSD	8–8	11	1,493.34	1.46e−3	0.459
	MDHI	16	4–4	1,436.50	5.15e−3	0.135
	PMC-HI	16	4–12	1,325.98	1.23e−4	6.13

When rendering, we use the same number of samples per pixel (SPP) for all direct lighting operators and manage the number of shadow rays per light (SRL) to let the running time be close. For the ALSD, $N_1 - N_2$ in the SPP column denotes the number of samples per pixel in the first and second iterations and ALSD must have two image-plane iteration in order for light sampling adaptation. For the MDHI and PMC-HI algorithms, $N_1 - N_2$ in the SRL column denotes the number of rays in the initial and adaptive iterations. T denote the time to finish rendering the entire image, Err denote the average mean square error and Eff denotes the efficiency of the estimator which is defined as $\frac{1}{T \times \text{Err}}$ in [19]. Generally, we would like to have an estimator whose computational time and estimated error are both small and thus, when given a fixed computational time, a more efficient estimator should get a lower error

in the bottom of Fig. 4. Figures 5 and 6 shows the rendered result of the JO and YR scenes using PMC-HI, ALSD, MDHI and MIS. The PMC-HI, ALSD and MIS operators can get better results in the bright-light-shading regions from the light shedding out of the holes of the Jack-o-lantern. The PMC-HI operator can improve the result of regions in partial shadowing in the Yearright scene.

6 Discussion

The PMC-HI sampler can be viewed as generalizations of MIS which is a special case of deterministic mixture sampling. It corresponds to fixing the α_k weights ahead of time, which fixes the number of samples from each function. The MIS balance heuristic results in the same estimator that we use. We improve upon MIS by adapting the weights over time, which avoids wasting samples on unimportant component functions.

Many PMC kernels in the literature are mixture models. Mixtures are typically formed by combining several components that are each expected to be useful in some cases but not others. The adaption step then determines which are useful for a given input. Mixtures allow otherwise unrelated functions to be combined, such as the light source and BRDF importance functions in Eq. 4. If an environment map was present, we could even include the wavelet importance functions of Clarberg et al. [4] in the mixture. Typically, the common rule of choosing importance functions applies here also: when f is a product of several unrelated functions, then a good choice of mixture components is something proportional to each factor.

The most notable limitation of PMC is the high sample counts required when the kernel has many adaptable

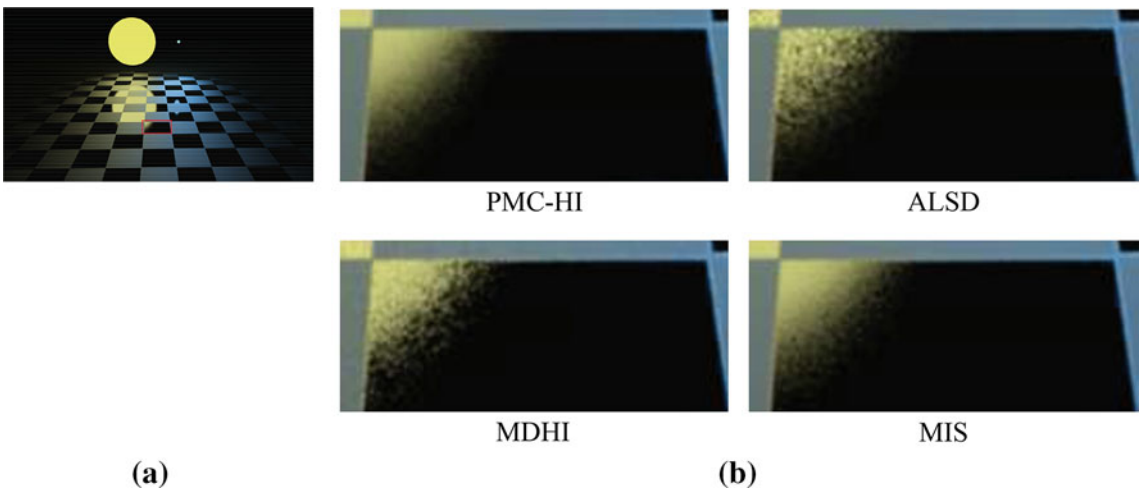


Fig. 3 This shows four images the cropped results of the CK scene which focuses on the glossy square and is marked with the red rectangle in **a** when using PMC-HI, MDHI, ALSD and MIS

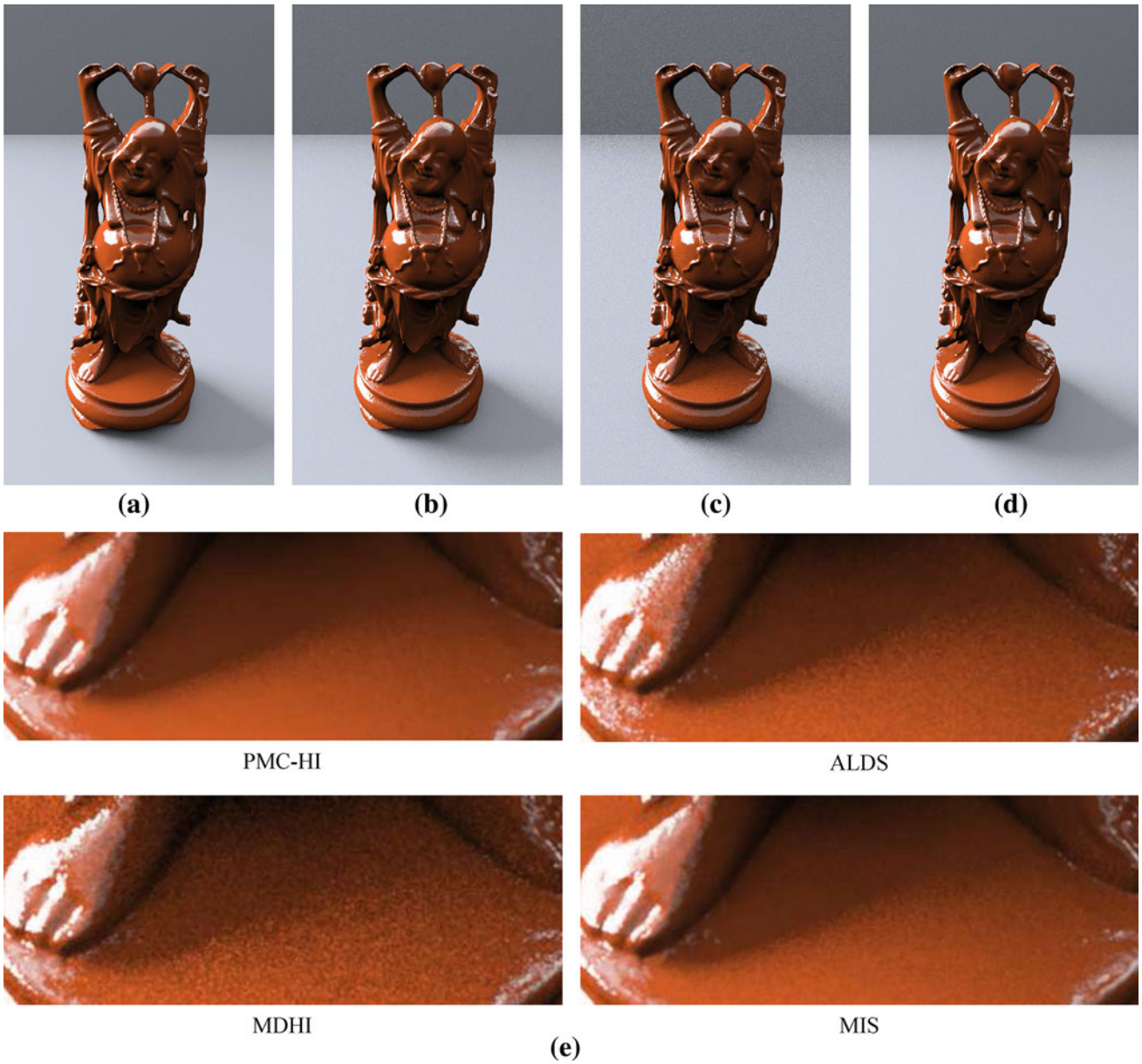


Fig. 4 **a–d** The rendered results of the BU scene rendered using PMC-HI, MDHI, ALSD and MIS respectively. **e** Shows the cropped results of the bottom part of the budda

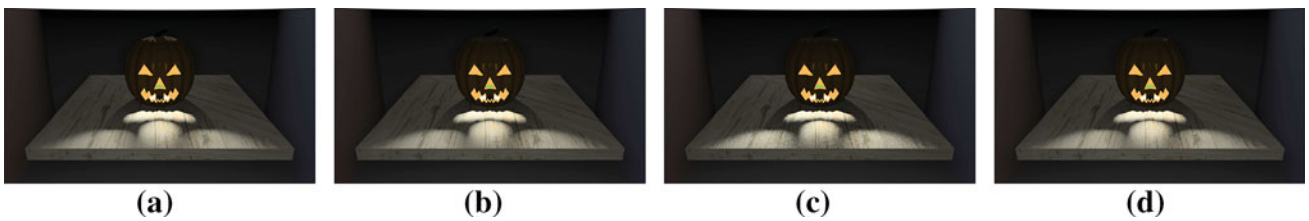


Fig. 5 **a–d** The rendered results of the JO scene when using PMC-HI, ALSD, MDHI and MIS

parameters. This precludes, for instance, using one component per light when there are many lights. Such a strategy would be appealing for efficiently sampling in complex shadow situations (some components would see the lights, others would not), but the sample count required to adequately determine the mixture component weights would be

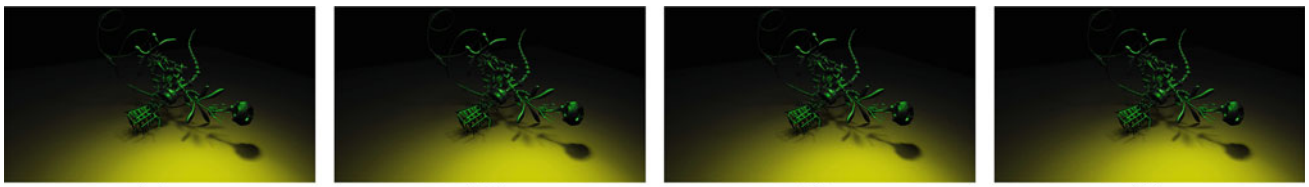


Fig. 6 **a–d** The rendered results of the YR scene when using PMC-HI, ALSD, MDHI and MIS

too large. Instead we use a single mixture component for all the lights and rely on the cone perturbation component to favor visible lights. This does not work well if the illumination sources are widely spaced.

To achieve the consistency criterion of PMC discussed in [8], the numbers of sample for the convergence iteration must approach infinity. In our algorithm, we can separate the sampler into two stages: base adaptation and convergence. The base adaptation stage aims at determining a proper weight for each kernel function and then the kernel functions are used to generate a consistent result in the convergence stage. Therefore, the numbers of sample for the base stage is small and the numbers of sample for the convergence iteration is large for generating consistent results.

7 Conclusion

We have shown how the hemispheric integral estimation can be derived from the PMC framework. The algorithm learns to be an effective sampler based on the results generated in early iterations. This alleviates one of the greatest problems in Monte Carlo rendering: the choice of importance functions and other parameters. Although this work currently only applies the PMC-HI sampler for direct lighting estimation, the sampler could be used in any situation where estimates of an integral over the hemisphere are required. For example, irradiance caching would possibly benefit greatly from the sampler in the computation of each cached value. PMC is just one approach from the family of iterated importance sampling algorithms [17]. The Kalman filter is another well-known example. Common to these techniques is the idea of sample reuse through resampling and the adaption of sampling parameters over iterations. Computer graphics certainly offers further opportunities to exploit these properties.

References

1. Agarwal, S., Ramamoorthi, R., Belongie, S., Jensen, H.W.: Structured importance sampling of environment maps. In: SIGGRAPH '03, pp. 605–612 (2003)
2. Burke, D., Ghosh, A., Heidrich, W.: (2005) Bidirectional importance sampling for direct illumination. In: Proceedings of the 16th Eurographics Symposium on Rendering, pp. 147–156
3. Cappé, O., Guillin, A., Marin, J.M., Robert, C.: Population Monte Carlo. J. Comput. Graph. Stat. **13**(4), 907–929 (2004)
4. Clarberg, P., Jarosz, W., Akenine-Möller, T., Jensen, H.W.: Wavelet importance sampling: efficiently evaluating products of complex functions. In: SIGGRAPH '05, pp. 1166–1175 (2005)
5. Donikian, M., Walter, B., Bala, K., Fernandez, S., Greenberg, D.P.: Accurate direct illumination using iterative adaptive sampling. IEEE Trans. Vis. Comput. Graph. **12**(3), 353–364 (2006)
6. Dutré, P., Willems, Y.D.: Importance-driven Monte Carlo light tracing. In: Proceedings of the 5th Eurographics Workshop on Rendering, pp. 185–194 (1994)
7. Dutré, P., Willems, Y.D.: Potential-driven Monte Carlo particle tracing for diffuse environments with adaptive probability functions. In: Proceedings of the 6th Eurographics Workshop on Rendering, pp. 306–315 (1995)
8. Fan, S.: (2006) Sequential monte carlo methods for physically based rendering. PhD thesis, University of Wisconsin at Madison
9. Ghosh, A., Doucet, A., Heidrich, W. Sequential sampling for dynamic environment map illumination. In: Proceedings of Eurographics Symposium on Rendering, pp. 115–126 (2006)
10. Hachisuka, T., Jarosz, W., Weistroffer, R.P., Dale, K., Humphreys, G., Zwicker, M., Jensen, H.W.: Multidimensional adaptive sampling and reconstruction for ray tracing. ACM Trans. Graph. **27**(3), 33:1–33:10 (2008)
11. Lafortune, E.P., Willems, Y.D.: A 5D tree to reduce the variance of Monte Carlo ray tracing. In: Proceedings of the 6th Eurographics Workshop on Rendering, pp. 11–20 (1995)
12. Lai, Y., Chenney, S., Liu, F., Niu, Y., Fan, S.: Animation rendering with population monte carlo image-plane sampler. Vis. Comput. **26**, 543–553 (2010)
13. Lai, Y.-C., Fan, S., Dyer, C.: Population Monte Carlo path tracing. Technical Report TR-1614. University of Wisconsin, Madison, USA (2010)
14. Ohbuchi, R., Aono, M.: Quasi-Monte Carlo rendering with adaptive sampling. Technical Report RT0167. IBM Tokyo Research Laboratory, Yamato, Japan (1996)
15. Pharr, M., Humphreys, G.: Physically based rendering from theory to implementation. Morgan Kaufmann, Burlington, MA (2004)
16. Pietrek, G., Peter, I.: Adaptive wavelet densities for Monte Carlo ray tracing. In: Skala V (ed) WSCG'99 Conference Proceedings, pp. 217–224 (1999)
17. Robert, C.P., Casella, G.: Monte Carlo statistical methods, 2nd edn. Springer, New York (2004)
18. Talbot, J., Cline, D., Egbert, P. Importance resampling for global illumination. In: Proceedings of the 16th Eurographics Symposium on Rendering, pp. 139–146 (2005)
19. Veach, E.: Robust Monte Carlo methods for light transport simulation. PhD thesis, Stanford University (1997)
20. Veach, E., Guibas, L.J.: Optimally combining sampling techniques for Monte Carlo rendering. In: SIGGRAPH '95, pp. 419–428 (1995)
21. Ward, G. Adaptive shadow testing for ray tracing. In: Proceedings of the 2nd Eurographics Workshop on Rendering, pp. 11–20 (1991)



Yu-Chi Lai received his B.S. from the National Taiwan University, Taipei, R.O.C., in 1996 in the Electrical Engineering Department. He received his M.S. and Ph.D. degrees from the University of Wisconsin-Madison in 2003 and 2009, respectively, in electrical and computer engineering and his M.S. and Ph.D. degrees in 2004 and 2010, respectively, in computer science. He is currently an assistant professor in NTUST and his research interests are in the area of graphics, vision, and multimedia.



Hsuan-Ting Chou received his B.S. from the National Taiwan University of Science and Technology, Taipei, R.O.C., in 2012 in the Department of Electronic and Computer Engineering. He is currently a student in NTUST in the Department of Computer Science and Information Engineering and his research interests are in the area of graphics, vision, and multimedia.



Kuo-Wei Chen received his B.S. from the National Taiwan University of Science and Technology, Taipei, R.O.C., in 2013 in the Department of Electronic and Computer Engineering. He is currently a graduate student of the Department of Computer Science and Information Engineering in NTUST and his research interests are in the area of graphics, vision, and multimedia.



Shaohua Fan received his B.S degree in mathematics from Beijing Normal University, M.S. degree in mathematical finance from Courant Institute at NYU, and Ph.D. in computer science from the University of Wisconsin-Madison. He is currently a professor at financial engineering research center at the Suzhou University, China and a quantitative researcher at Ernst & Young in New York. His research interests are statistical arbitrage, high-frequency trading, and Monte Carlo methods for financial derivatives pricing.