Efficient Schemes for Monte Carlo Markov Chain Algorithms in Global Illumination

Yu-Chi Lai, Feng Liu, Li Zhang, and Charles Dyer

Computer Science, University of Wisconsin – Madison,
1210 W. Dayton St., Madison, WI 53706-1685, USA

Abstract. Current MCMC algorithms are limited from achieving high rendering efficiency due to possibly high failure rates in caustics perturbations and stratified exploration of the image plane. In this paper we improve the MCMC approach significantly by introducing new lens perturbation and new path-generation methods. The new lens perturbation method simplifies the computation and control of caustics perturbation and can increase the perturbation success rate. The new path-generation methods aim to concentrate more computation on "high perceptual variance" regions and "hard-to-find-but-important" paths. We implement these schemes in the Population Monte Carlo Energy Redistribution framework to demonstrate the effectiveness of these improvements. In addition., we discuss how to add these new schemes into the Energy Redistribution Path Tracing and Metropolis Light Transport algorithms. Our results show that rendering efficiency is improved with these new schemes.

1 Introduction

Generating a physically-correct image involves the estimation of a large number of highly correlated integrals of path contributions falling on the image plane. Markov Chain Monte Carlo (MCMC) algorithms such as Metropolis Light Transport (MLT) [1], Energy Redistribution Path Tracing (ERPT) [2], and Population Monte Carlo Energy Redistribution (PMC-ER) [3] exploit the correlation among integrals. They all reduce the variance and improve the efficiency during rendering images. However, MCMC algorithms are limited from achieving higher rendering efficiency due to the possibly high failure rate in caustics perturbation and the stratified exploration of the image plane.

The predicted range of the perturbation angle for caustics perturbation depends on the path and scene properties. If the predicted range is too large, the failure rate of the caustics perturbation will be high and cause extra high energy to accumulate at some specific spots on the image plane. As a result, the large predicted range decreases the rendering efficiency. Additionally, the MCMC algorithms need to implement the lens and caustics perturbation separately because it is impossible for the original lens perturbation to generate a new mutated path for caustics paths with the form of $EDS^*D^+(D|L)$ which is a notation of light tranport paths introduced by [4, 1]. MCMC algorithms have issues in the

extra cost needed for computing the perturbation angles for each path and the burden in predicting the perturbation change on the image plane.

Stratified exploration of the image plane is another limitation because the importance of regions in the image are not perceptually the same. To achieve unbaisedness, the ERPT and PMC-ER algorithms carefully generate new paths by evenly distributing the path samples on the image plane. This choice is suboptimal because some areas on the image plane contain higher perceptual variance than others. To contribute more computational effort in reducing the variance in these high perceptual variance regions would increase the perceptual quality. In addition, some types of paths such as caustics paths are visually important but hard to find with a general path tracing algorithm. Concentrating more computational effort on these paths can further improve the rendering efficiency. However, evenly exploring the image plane prevents MCMCs from spending more computation effort in exploring those "hard-to-find-but-important" paths and limits the improvement in rendering efficiency.

To address these two limitations, we augment lens perturbation to include caustics perturbation. This new perturbation allows us to control the mutation by a single and simple lens perturbation radius and increases the success rate of caustics perturbations to improve the rendering efficiency. We propose two methods to generate paths in order to spend more computational effort on exploration of noisy regions and "hard-to-find" paths without introducing bias. We present a variance-generation method that generates paths passing through high perceptual variance regions on the image plane to enhance the perceptual quality of the visually important regions in the rendered image. We also present a caustics-generation method generates a set of caustics paths with the goal of exploring the caustics path space more thoroughly. We weigh the energy deposited by each perturbation according to the type of the population path to prevent new generation methods from introducing bias.

2 Related Work

Currently, most global illumination algorithms are based on ray tracing and Monte Carlo integration. Two categories exist: unbiased methods such as [5–7]; and biased methods such as [8, 4, 9]. Interested readers can refer to Pharr and Humphreys [10] for an overview of Monte Carlo rendering algorithms.

Sample reuse is an important technique to reduce the variance by exploiting the correlation among integrals. Metropolis Light Transport (MLT) [1] and Energy Redistribution Path Tracing (ERPT) [2] mutate existing light transport paths into new ones to make use of the correlated information among paths. However, finding a good mutation strategies is important but non-trivial for rendering efficiency. PMC-ERs [3] adapt the Population Monte Carlo framework into energy redistribution. Their algorithms can concentrate the computation on the important light paths and automatically adapt the extent of energy redistribution according to each path's properties. This eases the problem of choosing the non-trivial mutation strategies existing in MLT and ERPT algorithms. How-

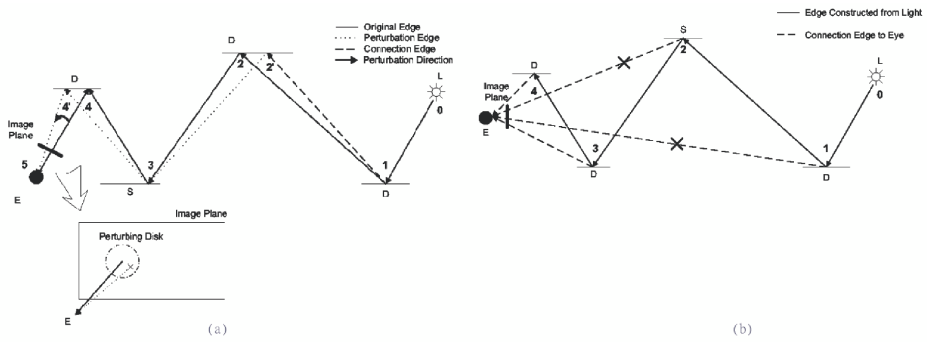

Fig. 1. (a). This is a path of the form LDDSDE and used to demonstrate the replacement of caustics perturbation with the new lens perturbation. We would like to replace the caustics sub-path $\mathbf{y}_5\mathbf{y}_4\mathbf{y}_3\mathbf{y}_2\mathbf{y}_1$ of the form of EDSDD. We first perturb the pixel position of the original path at $\mathbf{y}_5$ by uniformly choosing a point from the perturbing disk and then cast a view ray to pass through the new pixel position as shown in the bottom to get $\mathbf{y}'_4$. We link $\mathbf{y}'_4$ and $\mathbf{y}_3$ to form the link path. Then, we extend the sub-path through the same specular bounce at $\mathbf{y}'_3$ as the corresponding $\mathbf{y}_3$ to get $\mathbf{y}'_2$. Then, $\mathbf{y}'_2$ and $\mathbf{y}_1$ are linked to form a new lens-perturbed path with the same form of LDDSDE as the original one. (b). A caustics path is generated by tracing the ray from a light source. Each vertex in the path is linked to the camera vertex. The algorithm then checks whether the new linked path is a caustics path and if it is, it keeps it in the candidate pool. After finishing the whole process, we then randomly choose a path from the candidate pool and put it into the caustics path pool.

ever, there exist several limitations in MLT, ERPT, and PMC-ERs that prevent them from achieving higher rendering efficiency. In this paper we propose several new modifications to MCMC algorithms and implement them in PMC-ERs to demonstrate their effectiveness in improving rendering efficiency.

3 New Schemes to MCMCs

In this section we present new lens mutation and path-generation methods in PMC-ER-E. Interested readers can refer to [3] for the details of the original PMC-ER-E algorithm. In this work, a path, $\tilde{\mathbf{Y}}$, is referred to as a light transport path defined as [4, 11] and denoted as as $L(S|D)^*E$. Figure 1.(a) shows an example of such paths.

3.1 New Lens Perturbation

Details related to the kernel function, the choice of mutation strategies, and the computation of acceptance probability for the selected mutation are discussed in [3]. Here we only focus on how to use the perturbation method to replace the original caustics perturbation.

Figure 1(a) shows an example of our new lens perturbation method for a caustics path. The lens perturbation replaces a sub-path $\mathbf{y}_{n-1} \cdots \mathbf{y}_k$ of the form $EDS^*(L|D)$. In the original implementation of lens perturbation, the lens fails to replace this kind of path because it is impossible to find exactly the same outgoing direction at the first specular bounce from the eye vertex when we perturb the pixel position at the eye vertex. Thus, we need to use caustics perturbation. However, in our new lens mutation, we look to replace the sub-path chain with $E(D|S)^+[S(D|S)]^*$ sub-paths, which can directly replace the lens and caustics perturbation. First, the perturbation takes the existing path and moves the point on the image plane through which it passes. In our case, the new pixel location is uniformly sampled within a disk of radius d, a parameter of the kernel component. The path is reconstructed to pass through the new image point. If y'_{n-2} is a specular vertex, we choose a specular bounce to find the next vertex and then extend the sub-path through additional specular bounces to be the same length as the original path. If y'_{n-2} is a diffuse vertex, we link y'_{n-2} to y'_{n-3} to form the link path and then extend the sub-path through additional specular bounces to be the same length as the original path. The transition probability for the new lens perturbation for a caustics path can be computed as $T_{d,lens}(\tilde{\mathbf{Y}}'|\tilde{\mathbf{Y}}) = \frac{G(\mathbf{y}'_{n-1},\mathbf{y}'_{n-2})}{A_d} \prod_{j=n-3}^{n-k-2} \left(\frac{G(\mathbf{y}'_j,\mathbf{y}'_{j+1})}{|\cos\theta_{j',in}|} : 1?\mathbf{y}'_j \subset S \right)$ where $G(\mathbf{y}'_j, \mathbf{y}'_{j+1})$ is the geometric term between $\mathbf{y}'_j$ and $\mathbf{y}'_{j+1}$, A_d is the area of the perturbation, and $\theta_{j',in}$ is the angle between the normal of the surface and the direction of the incoming light ray at $\mathbf{y}'_j$. This relieves us from the need in the original caustics perturbation to estimate the perturbation angle, θ, for each path. The computation for θ is difficult and hard to predict the movement caused by the caustics perturbation on the image plane. When using our new lens perturbation, we can use a single pixel perturbation radius to control the movement of the radius on the image plane. The results show that control is easier and movement on the image plane is more predictable.

3.2 Resampling

The resampling process consists of three steps: **elimination**, which eliminates well-explored and low-contribution samples and deposits the remaining energy of the eliminated path into the image; **regeneration**, which maintains a constant number of paths in the population and designs an exploration pattern in the path space; and **adaptation of α values**, which adjusts the energy distribution area. In this section, we only focus on the process of regeneration. For details of elimination and adaptation refer to [3]. To generate a new replacement path, we use three types of regeneration paths: paths passing through a set of stratified pixel positions, paths passing through a set of pixel positions generated according to the perceptual variance, and a set of caustics paths tracing from the light sources. To achieve this, we need to have two modifications in the original algorithm. First, we split the resampling loop into two loops (s, t). At the beginning of each s loop, we need to modify the step of the generation of a pool of stratified pixel positions to a pool of pixel positions and a set of caustics

paths in the original energy redistribution algorithm. At the t loop, we apply the resampling process to the entire population. Second, we need to modify the energy deposition from $E_d = e_d$ to $E_d = R * e_d$ where R is related to the properties of the path discussed later in this section. The following are the implementation details.

Pixel Positions from Stratification Criterion It is important to evenly distribute the starting pixel positions in order to reduce variance and guarantee the unbiasedness of energy redistribution algorithms. Thus, in each s loop, we assign the $N_{uniform}$ samples as initial paths for each pixel.

Pixel Positions from Perceptual Variance Criterion In order to generate new sample paths in regions with possibly higher perceptual variance, we have to keep track of the radiance of traced paths similar to a path tracing algorithm by adding an extra image variable, I. In the process of estimating the average path energy, we keep track of the radiance of energy-estimated paths in I. In the following steps we also keep track of the radiance of the newly generated initial and replacement population paths in I. Then we can compute the value $\beta_{i,j}^{(s)} = \frac{\overline{\sigma}_{i,j}^2}{tvi(I_{i,j})}$ where $I_{i,j}$ is the average radiance that falls on pixel (i,j); $\sigma_{i,j}$ is the variance among all radiance samples falling in pixel (i,j), and $tvi(I)$ is threshold-versus-intensity function introduced by Ferweda et al. [12] for perceptually weighting the variance. $\beta_{i,j}^{(s)}$ is used to indicate the degree of requirement for more samples at pixel (i,j). At the beginning of each s loop, we first choose $N_{variance}$ for $(i,j)'s$ pixels, according to the weight $\beta_{i,j}^{(s)}$. After choosing pixels, we can compute the total number of samples falling on a pixel, $N_{uniform} + N_{variance}(i,j)$, and then we evenly distribute the starting pixel positions inside the pixel. This forms a pool of pixel positions. During the regeneration process, we ask for a pixel sample from this pool or ask for a new path from the pool of caustics paths. If we get a pixel sample, we then use the path tracing algorithm to generate a path passing through the new pixel positions. The unweighed energy of the path is calculated as described in [3]. Later, we will describe how to weigh the deposited energy without introducing bias.

Caustics Paths A path tracing algorithm traces paths starting from the eye. However some types of paths are easier to trace when starting from a light source, e.g. caustics paths. The photonmapping algorithm uses caustics photons to improve the rendering efficiency. The rendering results in Figure 2.(c) show that caustics paths are hard to find by the path tracing algorithm but they are very important to generate the smooth caustic regions on the floor near the dragon. Thus, these two observations motivate us to have specific types of light paths to enable the exploration of caustics path space. At the beginning of each outer iteration i.e the s loop, we generate a pool of $N_{caustics}$ caustics paths in the following way. First, we choose a light source and then choose a position on that light source as the start vertex. From the light vertex, we trace a path in the

scene as described in [11, 7]. Then, we connect each vertex in the light path to the camera vertex. If the complete path formed is a valid caustics path, we keep the path in the candidate pool. Finally, we can construct one valid caustics path by randomly choosing a valid one from the candidate pool. Figure 1(b) shows an example. The criterion for a caustics path is: first, the length of the path must be over 4 vertices; second, the path must contain at least one specular vertex; third, the first connection vertex from the eye vertex must be a diffuse surface. Without weighting the path energy, these extra "hard-to-find" paths will introduce bias. Next, we describe how to weigh the deposited energy without introducing bias.

Weighting the Energy of Newly Regenerated Paths In the original energy redistribution algorithm, we evenly distribute the pixel positions. The energy distribution ratio R should be 1. However, if we apply extra samples on each pixel and extra caustics paths without weighting the energy of each path, the extra samples and paths introduce biased energy into the image. In this section, we describe how to weigh the energy to ensure that the result is still unbiased. For the perceptual-variance-type regeneration, each pixel originally has $N_{uniform}$ samples dropped in the effective area and this guarantees that the expected energy deposited from paths initialized from each pixel is the same. To keep the energy deposit in the region statistically equal, we should weigh the deposited energy of the path by $R_{variance} = \frac{N_{uniform}}{N_{variance}(i,j)+N_{uniform}}$ where $N_{uniform}$ is the assigned uniform samples per pixel and $N_{variance}(i, j)$ is the sample assigned to pixel (i, j) according to perceptual variance. By weighing the energy by a ratio of $R_{variance}$ we make sure that the total energy expected to be distributed starting from that pixel is the same.

The caustics paths are global because they are light paths that can pass through any pixel position on the image plane. Thus, we need to handle them a little differently. In a scene we should expect the ratio of caustics paths and general paths generated from path tracing algorithm to be fixed. We can use this ratio to weigh the energy of all caustics paths in order to avoid bias. The ratio can be calculated as $R_{caustics} = \frac{N_{expect}}{N_{add}+N_{expect}}$ where N_{expect} is the expected total number of caustics paths generated by the stratified regeneration method. In the initial process, when we estimate the average energy of a path, we also estimate R_{G2C} which is the ratio of the total number of caustics paths to the total number of general paths. Then, during the regeneration process, we compute the $N_{expect} = R_{G2C} * N_{pixels} * N_{uniform}$ and $R_{caustics}$. By weighing the energy of each caustics path by a ratio of $R_{caustics}$, we can guarantee the unbiasedness of the final result.

However, the real ratio, R, of the path deposit energy is separated into the following three situations: first, if a path is from the pool of pixel positions and is a caustics path, the ratio should be $R_{variance}(i, j) \times R_{caustics}$; second, if a path is from the pool of pixel position but is not a caustics path, the ratio should be $R_{variance}(i, j)$; third, if a path is from the pool of caustics paths, the ratio should be $R_{caustics}$. By using the appropriate ratio, we guarantee the unbiasedness.

Image	Method	Total Iter(S, T)	$N_{variance}$	$N_{caustics}$	Time (s)	Err	Eff
Box1	E*	1, 1225	0	0	4769.1	0.0267	7.85e-3
	E+Lens**	1, 1225	0	0	4683.1	0.0207	1.03e-2
	E+Reg***	1, 1225	199200	108000	5366.3	0.0135	1.38e-2
	E+Lens+Reg	1, 1225	199200	108000	5266.4	0.0113	1.68e-2
Dragon	E	1, 2430	0	0	13081.3	3.09	2.47e-5
	E+Lens	1, 2430	0	0	12640.4	1	7.91e-5
	E+Reg	1, 2430	779700	30300	14296.7	0.985	7.10e-5
	E+Lens+Reg	1, 2430	779700	30300	14097.7	0.164	4.33e-4
Room	E	8, 2350	0	0	96575.1	0.0274	3.78e-4
	E+Lens	8, 2350	0	0	95812.1	0.0208	6.91e-4
	E+Reg	8, 2350	170400	121200	98158.9	0.0105	9.70e-4
	E+Lens+Reg	8, 2350	170400	121200	98032.5	0.00569	1.52e-3

Table 1. Measurements comparing the PMC-ER with original lens and caustics mutation and the stratified regeneration with PMC-ER with the new lens mutation and the stratified regeneration, PMC-ER with original lens and caustics perturbation with all regeneration methods, and PMC-ER-E using all the new schemes.
* E represents the original PMC-ER-E algorithm using the lens and caustics mutation with stratified regeneration.
** +Lens represents that we implement the new lens mutation into PMC-ERs.
*** +Reg represents implementations of the new regeneration methods in PMC-ERs.

4 Results

To evaluate the performance of our improvements, we compared our methods against the original PMC-ER equal deposition algorithm on a Cornell Box (CB) scene, a bunny scene, and a complex room scene using the criterion of starting with a similar number of initial PT paths and the same perturbation, and the same regeneration algorithm. In all three cases, we used a population size of 5000 and three perturbation radii: 5, 10, and 50 pixels. In each step in the inner loop, each member generates 20 mutations, and 40% of the population is eliminated based on its remaining energy and regenerated. We used 16 samples per pixel (SPPs) for estimating $\tilde{E}$ and R_{G2C}.

When applying the new schemes to the PMC-ER algorithms, we used N_{SPP}, the number of SPPs, to compute N_{total}, the number of initial paths, and $N_{iteration}$, the number of total iterations, for the PMC-ER algorithms. We then chose (S, T) so that $N_{iteration} = S \times T$ to indicate the total iterations used in PMC-ERs. If we implement new regenerations into PMC-ERs, we chose the pool size of the variance regeneration, $N_{variance}$ and the pool size of caustics regeneration, $N_{caustics}$ for each S. Thus, (S, T), ($N_{variance}$, and $N_{caustics}$) are the main parameters used. Table 1 presents the improvement statistics when applying each new scheme separately and together with PMC-ER-E. We used the perceptually-based mean squared efficiency (P-Eff) metric defined in [13] for comparing algorithms.

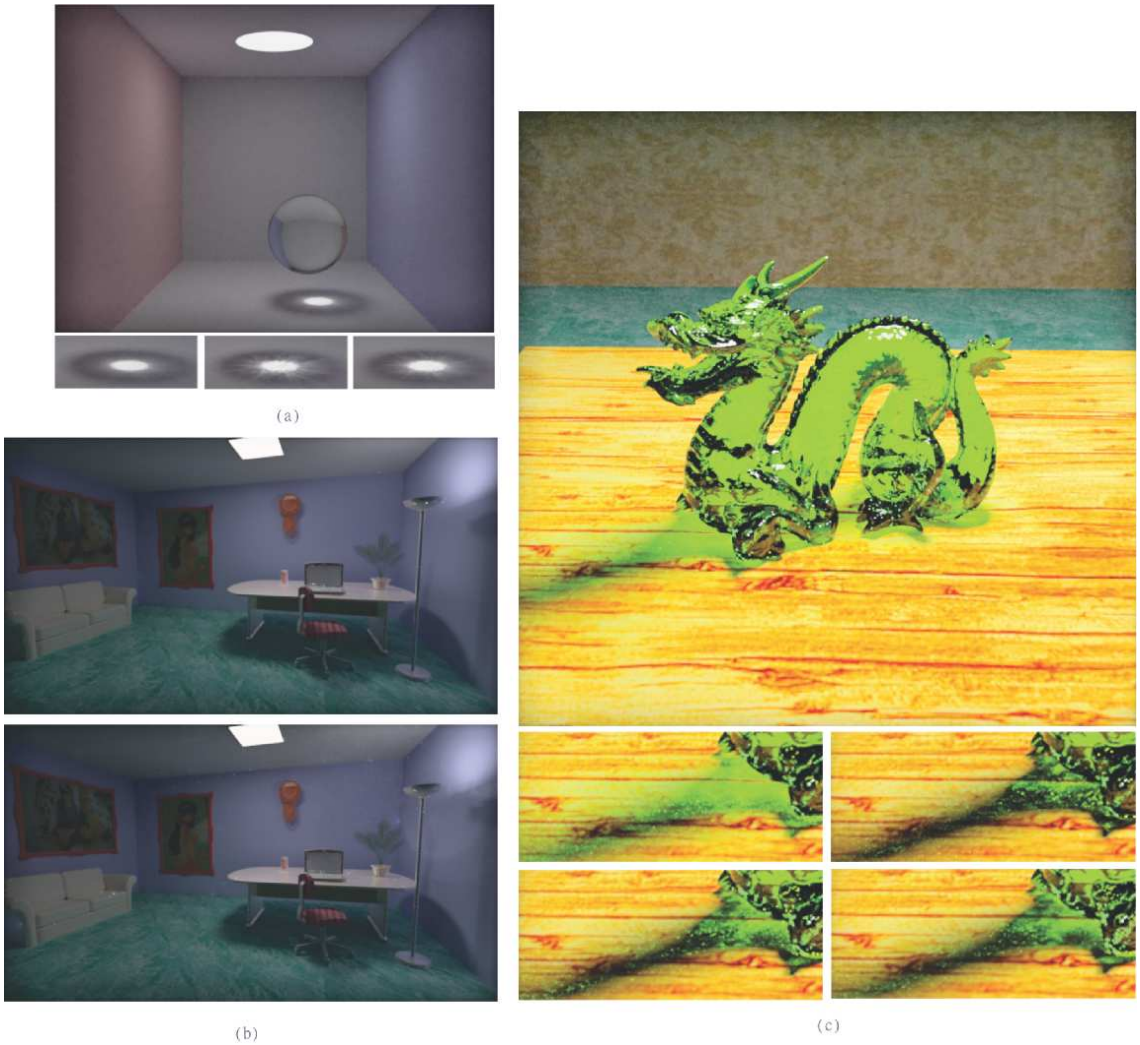

Fig. 2. (a). The top image is a Cornell Box image computed using PMC-ER-E with all new schemes with ($S = 1, T = 1225, N_{variance} = 199200, N_{caustics} = 108000$); the left in the bottom is the cropped image of the caustics region for the Cornell Box scene computed using PMC-ER-E with ($S = 1, T = 1225$), the middle the cropped image computed by the PMC-ER-E algorithm with ($S = 1, T = 1225$), and the right is the cropped image computed by the PMC-ER-E algorithm with ($S = 2, T = 1225$). (b). The top image is a room scene computed using PMC-ER-E with all new schemes with ($S = 8, T = 2350$); the bottom is computed using PMC-ER-E in ($S = 8, T = 2350, N_{variance} = 170400, N_{caustics} = 121200$). (c). The top is the rendering result of a dragon scene computed using PMC-ER-E with all new schemes with ($S = 1, T = 2430, N_{variance} = 779700, N_{caustics} = 30300$); the left in the middle row is the cropped image of the caustics region below the dragon head computed using PMC-ER-E, the right in the middle row is the cropped image computed by PMC-ER-E with ($S = 1, T = 2430$), the left in the bottom row is the cropped image computed by PMC-ER-E with ($S = 3, T = 2430$), and the right in the bottom row is the cropped image computed by PMC-ER-E in ($S = 8, T = 1620$) iterations.

The comparison between PMC-ER-E with the original perturbations and PMC-ER-E with the new lens perturbation in rendering the three scenes shows that we gain improvement in rendering efficiency by a factor of 1.31 for the CB scene, 3.2 for the dragon scene, and 1.82 for the room scene. The comparison between the original PMC-ER-E algorithm with stratified regeneration and PMC-ER-E with the new regeneration methods in rendering the three scenes shows an improvement of a factor of 1.76 for the CB scene, 2.87 for the dragon scene, and 2.56 for the room scene. The comparison between the original PMC-ER-E algorithm with stratified regeneration and PMC-ER-E with all new schemes shows an improvement of a factor of 2.14 for the CB scene, 17.53 for the dragon scene, and 4.02 for the room scene.

When viewing an image, the attention of the viewer is drawn towards the caustic regions in the image because caustic regions are usually brighter than the regions next to them. Thus, improving the quality of the rendered caustic regions has a large impact on the perception of a rendered image. The caustics regeneration concentrates more computation in the caustics path space. In addition, the new lens mutation increases the perturbation success rate to increase the exploration of the caustics path for each population path. As a result, our algorithm can generate smoother caustics regions for the dragon and CB scene.

In the room scene, we observe that the variance-regeneration puts more samples around the regions of the light in the right of the image. There is no obvious caustics region in the scene but the bright spots generated during the rendering process mostly come from the caustics paths. Thus, concentrating more computation in exploration of caustics path space reduces the variance of the result image. In addition, the failure rate of the caustics perturbation is high for this scene. With the new lens mutation method, the success rate can increase significantly. As a result, the rendered image is much smoother.

5 Discussion and Conclusion

In this section we present a short discussion of how to apply these schemes in the MLT and ERPT frameworks. The original lens and caustics pertubation methods in MLT can be directly replaced by our new lens perturbation method. To apply the new generation methods to MLT, we can first use these methods to generate a pool of paths passing through high-variance regions and caustics paths. Then, during the mutation process, we can replace the current seed path with one of the paths from the pool. We can compute the acceptability probability accordingly and decide whether the seed path transfers to the new generated path. This should achieve a similar result as presented in our demonstration. Since ERPTs contain a preprocessing phase to estimate the average energy of paths, we can implement a similar algorithm as stated in Section 3.2 to estimate the perceptual variance in each pixel and R_{G2C} in the preprocessing phase. After deciding on the number of caustics paths and variance-generated samples, we distribute the variance-generated samples according to the perceptual vari-

ance and generate caustics paths. The energy-deposited ratio, R, is computed as described in Section 3.2.

Except for the factors listed in the original PMC-ER algorithm [3], there is another important factor which is the ratio between the total number of the stratified regeneration paths and the special regeneration paths. If the ratio is high, the image space exploration rate will be too low. As a result, this reduces the variance of those highly explored regions but we will have a higher variance in other regions. If the ratio is too low, our algorithm reverts to the original PMC-ER algorithm. In the current implementation a proper value is set by trial and error. In the future, we would like to implement some automatic mechanisms.

In this paper we proposed two new path regeneration mechanisms by tracing paths through high perceptual variance regions and generating "hard-to-find" paths with a proper weighting scheme for concentration sampling without introducing bias. In addition, the new mutation method eases the control and computation of the caustics perturbation. Both schemes improve rendering efficiency.

References

1. Veach, E., Guibas, L.J.: Metropolis light transport. In: SIGGRAPH '97. (1997) 65–76
2. Cline, D., Talbot, J., Egbert, P.: Energy redistribution path tracing. In: SIGGRAPH '05. (2005) 1186–1195
3. Lai, Y., Fan, S., Chenney, S., Dyer, C.: Photorealistic image rendering with population monte carlo energy redistribution. In: Eurographics Symposium on Rendering. (2007) 287–296
4. Heckbert, P.S.: Adaptive radiosity textures for bidirectional ray tracing. In: SIGGRAPH '90. (1990) 145–154
5. Kajiya, J.T.: The rendering equation. In: SIGGRAPH '86. (1986) 143–150
6. Veach, E., Guibas, L.J.: Bidirectional estimators for light transport. In: Proc. of the 5th Eurographics Workshop on Rendering, Eurographics Association (1994) 147–162
7. Lafortune, E.P., Willems, Y.D.: Bi-directional path tracing. In: Proceedings of Compugraphics. (1993) 145–153
8. Ward, G.J., Rubinstein, F.M., Clear, R.D.: A ray tracing solution for diffuse interreflection. In: SIGGRAPH '88. (1988) 85–92
9. Jensen, H.W.: Realistic image synthesis using photon mapping, AK Peters (2001)
10. Pharr, M., Humphreys, G.: Physically Based Rendering from Theory to Implementation. Morgan Kaufmann (2004)
11. Veach, E.: Robust Monte Carlo Methods for Light Transport Simulation. PhD thesis, Stanford University (1997)
12. Ferwerda, J.A., Pattanaik, S.N., Shirley, P., Greenberg, D.P.: A model of visual adaptation for realistic image synthesis. In: SIGGRAPH '96. (1996) 249–258
13. Fan, S.: Sequential Monte Carlo Methods for Physically Based Rendering. PhD thesis, University of Wisconsin-Madison (2006)