Appears in ACM TOG 31(6) (SIGGRAPH Asia 2012).

Light Transport Simulation with Vertex Connection and Merging

Iliyan Georgiev
Saarland University
Intel VCI, Saarbrücken

Jaroslav Křivánek
Charles University in Prague
Faculty of Mathematics and Physics

Tomáš Davidovič
Saarland University
Intel VCI, Saarbrücken

Philipp Slusallek
Saarland University
Intel VCI & DFKI, Saarbrücken

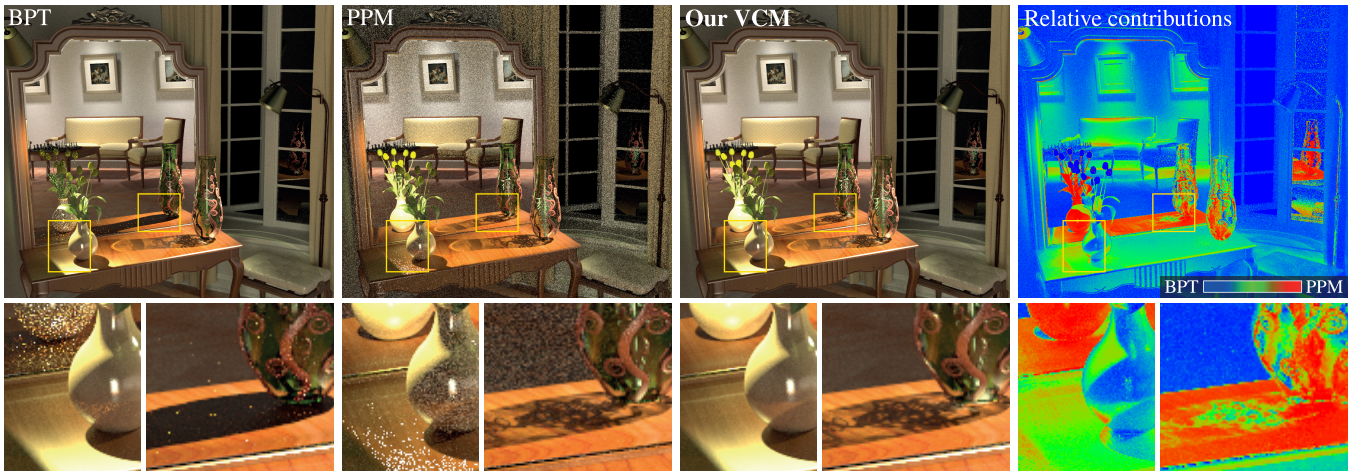


Figure 1: *A comparison of our new progressive vertex connection and merging (VCM) algorithm against bidirectional path tracing (BPT) and stochastic progressive photon mapping (PPM) after 30 minutes of rendering. BPT fails to reproduce the reflected caustics produced by the vase, while PPM has difficulties in handling the illumination coming from the room seen in the mirror. Our new VCM algorithm automatically computes a good mixture of sampling techniques from BPT and PPM to robustly capture the entire illumination in the scene. The rightmost column shows the relative contributions of the BPT and PPM techniques to the VCM image in false color.*

Abstract

Developing robust light transport simulation algorithms that are capable of dealing with arbitrary input scenes remains an elusive challenge. Although efficient global illumination algorithms exist, an acceptable approximation error in a reasonable amount of time is usually only achieved for specific types of input scenes. To address this problem, we present a reformulation of photon mapping as a bidirectional path sampling technique for Monte Carlo light transport simulation. The benefit of our new formulation is twofold. First, it makes it possible, for the first time, to explain in a formal manner the relative efficiency of photon mapping and bidirectional path tracing, which have so far been considered conceptually incompatible solutions to the light transport problem. Second, it allows for a seamless integration of the two methods into a more robust combined rendering algorithm via multiple importance sampling. A progressive version of this algorithm is consistent and efficiently handles a wide variety of lighting conditions, ranging from direct illumination, diffuse and glossy inter-reflections, to specular-diffuse-specular light transport. Our analysis shows that this algorithm inherits the high asymptotic performance from bidirectional path tracing for most light path types, while benefiting from the efficiency of photon mapping for specular-diffuse-specular lighting effects.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Raytracing

Keywords: light transport, global illumination, importance sampling, bidirectional path tracing, photon mapping, density estimation

Links: DL PDF WEB

1 Introduction

Light transport simulation is a central problem in photo-realistic image synthesis. It has been an active area of research for decades due to its utility in many applications, including architectural visualization, industrial design, as well as the entertainment industry. In the past years, considerable advances have been made with respect to the efficiency of light transport algorithms, but the improvements usually come with some sort of bias: often, some types of light interactions are disregarded, or handled inefficiently. Such approximations are sometimes acceptable, but often lead to a severe loss of image fidelity (see e.g. [Křivánek et al. 2010]). Developing truly robust light transport algorithms that can efficiently and accurately render a wide variety of scenes remains an important challenge that we address in this paper.

Bidirectional path tracing (BPT) [Lafortune and Willems 1993; Veach and Guibas 1994] is among the most versatile light transport algorithms. The true key to its robustness is the provably good combination of various path sampling techniques using multiple importance sampling (MIS) [Veach and Guibas 1995]. It has been, however, widely acknowledged that BPT is not efficient for transport paths with specular-diffuse-specular (SDS) configurations, where the notion of 'specular' also includes sharp glossy interactions. This is indeed an important practical limitation, because such paths occur in all scenes containing specular objects and their image contribution is especially important in some very common cases such as an object enclosed in glass, an interior of a car or a building, etc. The reason for this problem is that the path sampling techniques in BPT usually

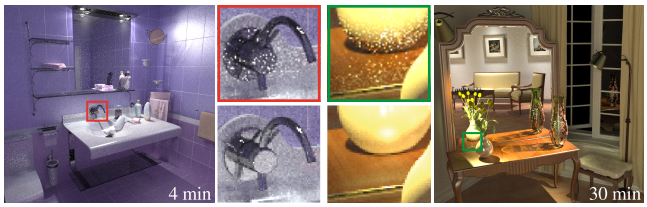


Figure 2: *Combining BPT with PM via heuristic classification of paths into caustic and non-caustic can be far from optimal (top insets). Our combination based on multiple importance sampling can be substantially more robust (bottom insets).*

sample SDS paths with low probability density, which may even go to zero if point light sources and pinhole cameras are allowed. Kollig and Keller [2000] call this the *problem of insufficient techniques*.

Efficient handling of SDS paths, on the other hand, has long been demonstrated with photon mapping (PM) [Jensen 2001]. A progressive variant has recently drawn attention with its ability to converge with a bounded memory footprint [Hachisuka et al. 2008]. However, its inefficiency under diffuse lighting and its relatively low order of convergence [Knaus and Zwicker 2011] make progressive photon mapping impractical as a general global illumination solution.

Intuitively, a combination of BPT and PM would be beneficial, as the two algorithms complement each other in terms of performance under different lighting conditions. Indeed, PM has traditionally been combined with some of BPT's path sampling techniques through heuristics such as separate direct lighting calculation, splitting of photons into global and caustic maps, and final gathering. However, as discussed by Veach and Guibas [1995] and shown in Fig. 2, such a heuristic combination can be far from optimal. Moreover, an adaptation of these heuristics to glossy reflectance is not obvious.

Judging from the success of multiple importance sampling in improving the robustness of BPT compared to its initial formulation, we can expect that a MIS-based combination of BPT and PM will yield a more robust solution than the aforementioned heuristics. However, such a principled combination has not been shown so far due to important differences in the mathematical frameworks used to formulate these two algorithms (the path integral formulation for BPT [Veach 1997], and density estimation for PM).

Contributions. In this paper, we present an integration of bidirectional path tracing and photon mapping into a framework that can efficiently handle a wide range of illumination effects. This is made possible by our novel Monte Carlo interpretation of the photon mapping radiance estimator, which circumvents the concept of density estimation and brings together these two algorithms that have so far been considered conceptually incompatible. Our new reformulation of photon mapping as a path sampling technique allows us to employ multiple importance sampling to combine the two methods in a more robust rendering algorithm that alleviates the problem of insufficient techniques. A progressive version of this algorithm is consistent, and we demonstrate its efficiently in handling a variety of lighting conditions, ranging from direct illumination and diffuse inter-reflections to SDS light transport. Our theoretical and empirical results show that this progressive algorithm retains BPT's high $O(1/N)$ mean squared error convergence rate for light paths that can be sampled by BPT, as well as the efficiency of photon mapping for SDS lighting effects. In summary, the main contributions of this paper are:

- A novel reformulation of photon mapping compatible with the path integral formulation of light transport (Section 4).
- A robust light transport simulation algorithm that combines BPT and PM via multiple importance sampling (Section 5).
- A progressive variant of the combined algorithm along with an asymptotic analysis of its error convergence (Section 6).

2 Previous Work

Path tracing. The path tracing algorithm [Kajiya 1986] follows random paths from the camera toward light sources. Dutré et al. [1993] render caustics more efficiently by tracing paths from the light sources. Since finding all high contribution paths by starting from either end is difficult (i.e. has low probability density), bidirectional path tracing (BPT) was developed to sample paths starting both from the camera and from the light sources [Lafortune and Willems 1993; Veach and Guibas 1994]. Veach [1997] formulates rendering as an integration of a pixel measurement function over all light transport paths. This *path integral framework* enables the combination of various path sampling techniques in a provably good way using multiple importance sampling [Veach and Guibas 1995], which is the real key to the robustness of BPT. Our work reformulates photon mapping as a new path sampling technique that can be combined with BPT's techniques in the same way, yielding a combined solution that is more robust than either of the two algorithms alone.

Photon mapping. Photon mapping (PM) [Jensen 2001] is a global illumination algorithm which approximates radiance at any point via density estimation from particles called "photons", distributed from the light sources. Its popularity stems from its simplicity and ability to render various lighting effects, including SDS paths. As pointed out by Hašan et al. [2009] and Vorba [2011], photon mapping has difficulties in scenes with many glossy objects. To alleviate this problem, Vorba [2011] uses multiple importance sampling (MIS) to combine the contributions of photons to radiance queries at different eye sub-path vertices in a manner similar to our approach. However, his expression for photon path pdfs disallows a meaningful application of MIS to combine the PM and BPT estimators, which estimate integrals w.r.t. different measures, as we detail in Section 4. Bekaert et al. [2003] apply multiple importance sampling on top of their generalized kernel density estimator in a way similar to Vorba [2011]. Tokuyushi [2009] improves caustics on glossy surfaces through a MIS-based combination of PM and final gathering estimates.

Unlike BPT, photon mapping is biased. Progressive photon mapping (PPM) modifies the original algorithm to diminish bias in the course of computation [Hachisuka et al. 2008]. PPM handles SDS paths more efficiently than BPT, but suffers from a lower asymptotic error convergence rate. Hachisuka et al. [2010] present an error estimation framework for PPM. Stochastic progressive photon mapping [Hachisuka and Jensen 2009] enables PPM to render distribution ray tracing effects. Our method generalizes stochastic PPM further, and combines it with BPT in a way that retains BPT's higher convergence rate for light paths that can be sampled by BPT.

Markov chain Monte Carlo. MCMC methods have been shown to increase the probability density of sampling high-contribution paths both in BPT [Veach and Guibas 1997] and PM/PPM [Fan et al. 2005; Hachisuka and Jensen 2011]. Jakob and Marschner [2012] improve the ability of the MCMC methods to sample SDS paths. These approaches are orthogonal to the problem addressed in this work. Indeed, enhancing the proposed algorithm through the use of a MCMC sampler is expected to bring further efficiency improvements.

Many-light methods. The efficiency of many-light rendering has recently drawn attention [Keller 1997; Walter et al. 2006; Hašan et al. 2007; Ou and Pellacini 2011]. In their basic form, such methods suffer from energy losses that degrade image fidelity [Křivánek et al. 2010]. Methods to alleviate these problems have been proposed [Kollig and Keller 2004; Hašan et al. 2009; Davidovič et al. 2010; Walter et al. 2012] but none can efficiently handle SDS paths.

This paper builds upon our previous work [Georgiev et al. 2011] that informally discusses the multiple importance sampling combination of BPT and PPM. The concurrent work of Hachisuka et al. [2012] addresses the same problem, but uses a significantly different theoretical formulation. We discuss these differences in Section 8.

3 Background

This section reviews the basis of our derivations: multiple importance sampling, the path integral formulation of light transport, path sampling techniques, and the photon mapping radiance estimate.

Multiple importance sampling. Monte Carlo methods evaluate integrals of the form $I = \int_\Omega f(x)\, d\mu(x)$, where f is a real-valued function and μ is a measure on the integration domain Ω. *Multiple importance sampling* (MIS) [Veach and Guibas 1995] constructs an unbiased estimator for I by combining estimators from m different distributions (or sampling techniques), each given by its probability density function (pdf) p_i, into a combined estimator:

$$\langle I \rangle_{\text{MIS}} = \sum_{i=1}^{m} \frac{1}{n_i} \sum_{j=1}^{n_i} w_i(X_{i,j}) \frac{f(X_{i,j})}{p_i(X_{i,j})}, \tag{1}$$

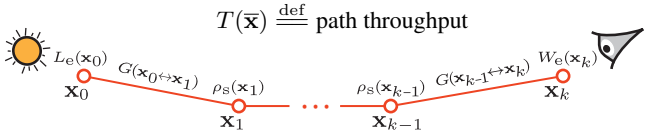
where $X_{i,j}$ are independent random variables with distribution p_i. The *power heuristic* $w_i(x) = [n_i p_i(x)]^\beta / \sum_{k=1}^{n} [n_k p_k(x)]^\beta$ is a good way to combine sampling techniques in terms of minimizing the estimator's variance. The *balance heuristic* corresponds to $\beta = 1$.

Path integral framework. In the path integral formulation of light transport [Veach 1997], the pixel measurement I for a pixel is given by an integral over the space of light paths:

$$I = \int_\Omega f(\overline{\mathbf{x}}) d\mu(\overline{\mathbf{x}}). \tag{2}$$

Here, $\overline{\mathbf{x}} = \mathbf{x}_0 \ldots \mathbf{x}_k$ is a light path with $k \geq 1$ edges, where the first vertex $\mathbf{x}_0$ is on a light source, the last vertex $\mathbf{x}_k$ is on the eye lens, and vertices $\mathbf{x}_1, \ldots, \mathbf{x}_{k-1}$ are scattering points on the scene surfaces. Ω is the space of paths of any length, $d\mu(\overline{\mathbf{x}}) = dA(\mathbf{x}_0) \ldots dA(\mathbf{x}_k)$ is the differential product area measure, and f is the *measurement contribution function* for the considered pixel:

$$f(\overline{\mathbf{x}}) = L_e(\mathbf{x}_0) \underbrace{G(\mathbf{x}_0 \leftrightarrow \mathbf{x}_1) \left[\prod_{i=1}^{k-1} \rho_s(\mathbf{x}_i) G(\mathbf{x}_i \leftrightarrow \mathbf{x}_{i+1}) \right]}_{T(\overline{\mathbf{x}}) \stackrel{\text{def}}{=} \text{path throughput}} W_e(\mathbf{x}_k). \tag{3}$$



Here $L_e(\mathbf{x}_0) = L_e(\mathbf{x}_0 \to \mathbf{x}_1)$ is the radiance emitted from $\mathbf{x}_0$ in the direction of $\mathbf{x}_1$, $W_e(\mathbf{x}_k) = W_e(\mathbf{x}_{k-1} \to \mathbf{x}_k)$ is the pixel sensitivity to light arriving at $\mathbf{x}_k$ from the direction of $\mathbf{x}_{k-1}$ (arrow is in the direction of light flow). $\rho_s(\mathbf{x}_i) = \rho_s(\mathbf{x}_{i-1} \to \mathbf{x}_i \to \mathbf{x}_{i+1})$ is the bidirectional scattering distribution function (BSDF) at $\mathbf{x}_i$, $G(\mathbf{x}_i \leftrightarrow \mathbf{x}_j) = V(\mathbf{x}_i \leftrightarrow \mathbf{x}_j) \frac{|\cos\theta_{i,j}||\cos\theta_{j,i}|}{\|\mathbf{x}_i - \mathbf{x}_j\|^2}$ is the geometry factor, and $V(\mathbf{x}_i \leftrightarrow \mathbf{x}_j)$ is the visibility term. Intuitively, f measures the contribution of the radiance emitted from $\mathbf{x}_0$ to the pixel along the path $\overline{\mathbf{x}}$, which becomes more explicit by writing $f(\overline{\mathbf{x}}) = L_e(\mathbf{x}_0) T(\overline{\mathbf{x}}) W_e(\mathbf{x}_k)$.

The path integral formulation enables the construction of an estimator for a pixel value by first sampling a random path $\overline{\mathbf{x}}$ connecting the eye with a light source and then evaluating $f(\overline{\mathbf{x}})/p(\overline{\mathbf{x}})$. Different path sampling techniques can be combined by multiple importance sampling, as is the case in bidirectional path tracing.

Path sampling techniques. The pdf $p(\overline{\mathbf{x}})$ of a light path describes the joint distribution of the individual path vertices $\mathbf{x}_0, \ldots, \mathbf{x}_k$, and is therefore given by the product of their conditional pdfs, $p(\overline{\mathbf{x}}) = p(\mathbf{x}_0, \ldots, \mathbf{x}_k) = p(\mathbf{x}_0) \ldots p(\mathbf{x}_k)$. Bidirectional path tracing (BPT) creates a path by sampling one sub-path from a light source and another one from the eye, optionally connecting them with an edge (Fig. 3a,b). In this case, the conditional vertex pdfs are:

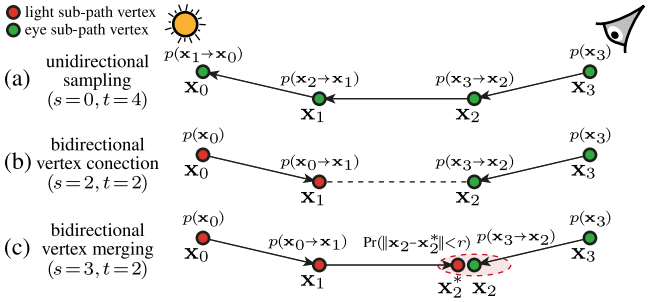


Figure 3: *Different techniques for sampling a light path, with the corresponding pdf terms associated with each vertex. For paths with k edges (here $k = 3$) bidirectional path tracing provides $k+2$ sampling techniques. Vertex merging brings $k-1$ new techniques corresponding to merging at the $k-1$ different interior path vertices.*

$$p(\mathbf{x}_j) = \begin{cases} p(\mathbf{x}_{j-1} \to \mathbf{x}_j) & \text{if } \mathbf{x}_j \text{ is on a light sub-path} \\ p(\mathbf{x}_{j+1} \to \mathbf{x}_j) & \text{if } \mathbf{x}_j \text{ is on an eye sub-path,} \end{cases} \tag{4}$$

where $p(\mathbf{x}_i \to \mathbf{x}_j)$ is the pdf for sampling the direction from $\mathbf{x}_i$ to $\mathbf{x}_j$, expressed w.r.t. the area measure. Two special cases, $j = 0$ on a light sub-path and $j = k$ on an eye sub-path, are the pdfs for the starting point of the light and eye sub-paths, respectively.

In BPT, a length-k path with $k+1$ vertices can be generated in $k+2$ ways, using different path sampling techniques identified by a pair (s, t), corresponding to the number of vertices on the light and eye sub-paths, respectively. *Unidirectional sampling* corresponds to the $(s = 0, t = k+1)$ and $(s = k+1, t = 0)$ techniques, (Fig. 3a shows the former case). We refer to the remaining sampling techniques as *vertex connection* because they involve an explicit connection of the light and eye sub-path endpoints (Fig. 3b).

The key to the improved robustness of BPT, compared to its initial implementation [Lafortune and Willems 1993; Veach and Guibas 1994], is the combination of the different path sampling techniques into a MIS estimator using the power heuristic [Veach and Guibas 1995]. The reason is that MIS automatically diminishes the weight of a sampling technique that is inappropriate (i.e. has a low pdf value) for a given path. This is, however, only possible if alternative techniques exist for sampling that path. But specular-diffuse-specular (SDS) paths can only be found via unidirectional sampling, which usually has low pdf, resulting in a high variance estimator. We address this problem by adding new path sampling techniques, corresponding to photon mapping, into the combined estimator.

Photon mapping radiance estimate. Photon mapping (PM) computes the scattered radiance L_s at a point $\mathbf{x}$ in direction ω via density estimation from photons distributed throughout the scene. The well-known photon map radiance estimate [Jensen 2001] reads:

$$L_s(\mathbf{x}, \omega) \approx \sum_j K_r(\|\mathbf{x} - \mathbf{x}_j\|) \rho_s(\omega_j, \mathbf{x}, \omega) \Phi_j, \tag{5}$$

where K_r is a 2D filtering kernel with support radius r, and j iterates over photons found within distance r from $\mathbf{x}$, so $\mathbf{x}_j$ and ω_j are the photon's position and incident direction, respectively, and Φ_j is the photon's flux. The main difficulty that arises when trying to combine PM with BPT into a robust estimator using MIS is the difference in the mathematical frameworks in which the two algorithms are defined. In particular, the above PM radiance estimate (5) does not include the notion of path pdf necessary for MIS. We address the problem in the next section by reformulating the radiance estimate as a path sampling technique.

4 Vertex Merging

Our goal is to combine photon mapping (PM) with bidirectional path tracing (BPT) into a more robust algorithm via multiple-importance sampling (MIS). The first step is to formulate the two algorithms in a common mathematical language, for which we choose the path integral framework, reviewed above. Since BPT is already naturally defined in this framework, we only need to reformulate the PM algorithm. This involves defining the light transport paths sampled by PM with their associated pdfs that we could then plug into the power heuristic. We refer to the path sampling techniques derived from PM as *vertex merging* (VM), as one can intuitively think of the radiance estimate (5) as merging the path vertices corresponding to a photon and the radiance estimate location within a small neighborhood (Fig. 3c). However, we underline that our derivation of the path pdf departs from this intuitive notion.

Our discussion considers light paths of a fixed length k. The PM radiance estimate is performed on the s-th vertex from the light, which yields a single path sampling technique. An entire family of techniques is then obtained by considering different locations of the radiance estimate $s \in [1, \dots, k-1]$.

PM as a sampling technique for extended paths. A natural approach to define a path sampled by PM would consider a single photon located at $\mathbf{x}_s^*$ and its tracing history as a sub-path $\mathbf{x}_0\mathbf{x}_1 \dots \mathbf{x}_s^*$ sampled form the light source. In a similar way, the point $\mathbf{x}_s$ where we perform the PM radiance estimate would be considered the end point of a sub-path $\mathbf{x}_s\mathbf{x}_{s+1} \dots \mathbf{x}_k$ traced from the eye. The PM radiance estimate (5) would then then complete a full length-k *extended path*, which we define as $\overline{\mathbf{x}}^* = (\mathbf{x}_0 \dots \mathbf{x}_s^*, \mathbf{x}_s \dots \mathbf{x}_k)$ and illustrate in Fig. 4 left. (To make the distinction clear, in this section we refer to the usual light paths $\overline{\mathbf{x}} = \mathbf{x}_0 \dots \mathbf{x}_k$ as *regular*.) The pdf of an extended path constructed this way would simply be the joint pdf of all its vertices (including the photon at $\mathbf{x}_s^*$):

$$\begin{aligned} p(\overline{\mathbf{x}}^*) &= p(\mathbf{x}_0 \dots \mathbf{x}_s^*)\, p(\mathbf{x}_s \dots \mathbf{x}_k) \\ &= [p(\mathbf{x}_0) \dots p(\mathbf{x}_{s-1} \to \mathbf{x}_s^*)]\, [p(\mathbf{x}_k) \dots p(\mathbf{x}_{s+1} \to \mathbf{x}_s)]. \end{aligned} \tag{6}$$

Discussion. We have now defined PM as path sampling technique for *extended* paths, with an associated pdf. We could directly use these definitions to apply MIS and combine the results of PM radiance estimates at different vertices of an eye sub-path. Indeed, that would correspond to what was done in the previous work of Vorba [2011] (with the difference that he uses pdfs w.r.t. the solid angle measure). Note, however, that we can *not* use the above definitions for a MIS-based combination BPT and PM. The reason is that for a given path length (i.e. number of edges), the extended paths sampled by PM have one extra vertex, the photon at $\mathbf{x}_s^*$, compared to the regular paths sampled by BPT. The pdf of an extended path is consequently expressed w.r.t. a different, higher-dimensional product area measure than the pdf of a regular path of the same length. However, the power heuristic in MIS expects the pdfs to be expressed w.r.t. the same measure to yield a meaningful weight.

To combine BPT and PM using MIS, we must express the pdfs of same-length paths sampled by BPT and PM w.r.t. the same measure, i.e. both algorithms should conceptually operate either in the space of the extended or the regular paths. Both options are possible, and we choose the space of regular paths, because doing so preserves the original path integral formulation we reviewed in Section 3.

PM as a sampling technique for regular paths. We now consider that PM samples *regular*, not extended, paths by *excluding the photon at $\mathbf{x}_s^*$ from the generated path*. That is, a PM radiance estimate at vertex $\mathbf{x}_s$ creates regular paths $\overline{\mathbf{x}} = \mathbf{x}_0 \dots \mathbf{x}_k$ (one such path for each photon). The radiance contribution of such a path now contains an extra area integration that corresponds to blurring by

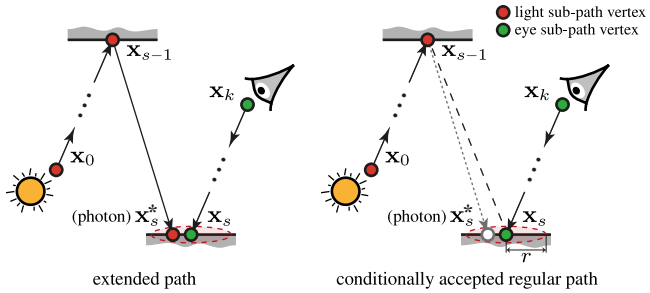


Figure 4: ***Left:*** *Photon mapping can be considered to sample an extended path $\overline{\mathbf{x}}^*$ of length k that has $k+2$ vertices.* ***Right:*** *To remain compatible with the path integral framework for BPT, we interpret this process as sampling a regular path $\overline{\mathbf{x}}$ of the same length that has only $k+1$ vertices. The path is accepted only if the photon at $\mathbf{x}_s^*$ lies within distance r to $\mathbf{x}_s$.*

the K_r kernel to which the path integral is oblivious, as we detail in Appendix A. We interpret the photon as a Monte Carlo sample used to estimate that integral. More importantly, the photon serves as a Russian roulette random variable that conditions the acceptance of the proposed path. The path $\overline{\mathbf{x}}$ is accepted if and only if the photon location $\mathbf{x}_s^*$ is within distance r from the radiance estimate location $\mathbf{x}_s$. This interpretation is in line with the traditional view and implementation of photon mapping, where photons outside of the search radius are not considered in the radiance estimate.

Path pdf. The pdf of a regular path $\overline{\mathbf{x}}$ created with the above procedure is $p_{\mathrm{VM}}(\overline{\mathbf{x}}) = P_{\mathrm{acc}}(\overline{\mathbf{x}})\, p_{\mathrm{VC}}(\overline{\mathbf{x}})$, where $p_{\mathrm{VC}}(\overline{\mathbf{x}})$ is the pdf for the BPT vertex connection (VC) technique that samples $\overline{\mathbf{x}}$ by connecting sub-paths with endpoints $\mathbf{x}_{s-1}$ and $\mathbf{x}_s$, as described in Section 3 (Fig. 4 right). The acceptance probability of the path is given by:

$$\begin{aligned} P_{\mathrm{acc}}(\overline{\mathbf{x}}) = \Pr(\|\mathbf{x}_s - \mathbf{x}_s^*\| < r) &= \int_{\mathcal{A}_\mathcal{M}} p(\mathbf{x}_{s-1} \to \mathbf{x}) d\mathbf{x} \\ &\approx |\mathcal{A}_\mathcal{M}|\, p(\mathbf{x}_{s-1} \to \mathbf{x}_s^*) \approx \pi r^2\, p(\mathbf{x}_{s-1} \to \mathbf{x}_s^*), \end{aligned} \tag{7}$$

where $\mathcal{A}_\mathcal{M} = \{\mathbf{x} \in \mathcal{M} \mid \|\mathbf{x}_s - \mathbf{x}\| \le r\}$ is the set of the surface points within distance r of $\mathbf{x}_s$. Exact analytic evaluation of the integral in (7) is generally impossible. To make the calculation feasible, we first assume constant pdf p inside $\mathcal{A}_\mathcal{M}$. This is a common assumption made by the progressive radiance estimation [Hachisuka et al. 2008] and its asymptotic analysis [Knaus and Zwicker 2011]. This allows us to take $p(\mathbf{x}_{s-1} \to \mathbf{x})$ out of the integral and replace it by $p(\mathbf{x}_{s-1} \to \mathbf{x}_s^*)$, which is known. Second, we make the common photon mapping assumption that $\mathcal{A}_\mathcal{M}$ is a disk with radius r, and area πr^2, centered around $\mathbf{x}_i$. Note that the accuracy of the resulting P_{acc} approximation reduces in areas of geometric variation and when p is far from constant, e.g. when $\mathbf{x}_{s-1}$ has a very sharp glossy BSDF. However, note that the approximation converges to the true value as the radius r goes to zero. Also, for $\mathbf{x}_s^*$ inside $\mathcal{A}_\mathcal{M}$ coming from a specular vertex $\mathbf{x}_{s-1}$ the acceptance probability is simply $P_{\mathrm{acc}} = 1$.

The final path pdf reads:

$$p_{\mathrm{VM}}(\overline{\mathbf{x}}) = P_{\mathrm{acc}}(\overline{\mathbf{x}})\, p_{\mathrm{VC}}(\overline{\mathbf{x}}) \approx [\pi r^2\, p(\mathbf{x}_{s-1} \to \mathbf{x}_s^*)] p_{\mathrm{VC}}(\overline{\mathbf{x}}). \tag{8}$$

This path pdf formula is not arbitrary; indeed, it describes the actual random events that occur during the path construction in PM. Note that the pdf is expressed w.r.t. the same measure as any length-k regular path sampled by BPT. The pdf of vertex $\mathbf{x}_s^*$ only appears as a part of the approximation of the acceptance probability P_{acc}, which is 'unitless'[1], and therefore has no impact on the measure. Note that this approximation makes the path pdf expression (8) symmetric, in the sense that it includes the densities of all sampled vertices.

[1] Considering that a vertex pdf has the units of $[\mathrm{m}^{-2}]$ and πr^2 is in $[\mathrm{m}^2]$.

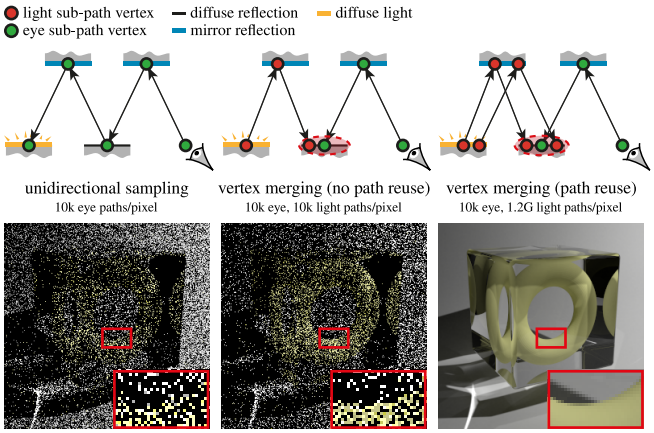


Figure 5: *Vertex merging (VM) explains why photon mapping can be more robust in handling SDS light transport than BPT. It is not because it can find these paths more easily, but because it can efficiently reuse light sub-paths across pixels. Without reuse, with equal light source and merging disk areas, photon mapping performs similarly to unidirectional path sampling.*

The acceptance probability term $\pi r^2\, p(\mathbf{x}_{s-1} \to \mathbf{x}^*_s)$ in the path pdf is useful for understanding the efficiency of photon mapping, as we show in Section 4.1. The dependence of this term on the radius r will also prove crucial for the good asymptotic performance of our progressive combined algorithm in Section 6.

Summary and discussion. We have reformulated photon mapping as a path sampling technique, which we call vertex merging. This formulation provides us with a clear definition of paths sampled by performing a PM radiance estimate, with their associated pdfs, which are expressed w.r.t. the same measure as in BPT. We use the path pdf in Section 5 for a MIS-based combination of PM and BPT. In Appendix A we derive the measurement contribution function, as well as the path integral actually estimated by photon mapping.

4.1 Efficiency of Different Path Sampling Techniques

Our vertex merging (VM) formulation allows us to analyze the relative efficiency of different path sampling techniques in different settings by comparing their pdfs for a given path. This tool has been long available to BPT, as the power heuristic is based on the observation that a higher pdf most often results in a lower-variance estimate [Veach and Guibas 1995]. Vertex merging allows us to include photon mapping in the comparison and to reason about its efficiency as a Monte Carlo sampling technique.

Sampling densities. Note in equation (8) that for any path, the VM pdf is at most equal to that of vertex connection (VC). This is due to the acceptance probability $P_{\text{acc}} \leq 1$, given by equation (7). Consider the path in Fig. 4 as an example. For practical values of r, the merging disk often spans a small solid angle as seen from $\mathbf{x}_{s-1}$, depending on its distance to that vertex. If the vertex $\mathbf{x}_{s-1}$ is diffuse, P_{acc} is low, as it is equal to the probability of sampling a ray inside that solid angle. The resulting VM path pdf p_{VM} can then be six or more orders of magnitude lower than the corresponding VC pdf. It can be also shown that if the merging disk area equals the light source area, then unidirectional sampling (US) and VM can have almost equal pdfs. The intuition is that the probability of hitting the light would be roughly the same as for hitting the merging disk.

Path reuse efficiency. Based on the above observations, we can deduce that VM, and thus photon mapping (PM), is not an intrinsically more efficient sampling technique than the BPT techniques. However, the power of VM is its *computational efficiency*. It performs conditional path concatenation, which is as cheap as neighborhood checking. This enables the reuse of a large number of light sub-paths at the cost of a single range search. Therefore, in cases where the pdfs for the other techniques are not much higher than for VM, the latter can result in a much lower-error estimate, due to its *efficient brute-force variance reduction* capabilities. The most prominent example for such cases are the SDS paths. We demonstrate this in Fig. 5 by comparing the quality achieved by VM without and with path reuse against US in the same time. Without reuse, VM performs like path tracing without explicit direct lighting computation.

5 A Combined Light Transport Algorithm

In this section we take advantage of our vertex merging formulation to combine PM and BPT into a more robust light transport algorithm. Because vertex merging formulates PM as a path sampling technique with an associated pdf, we can now directly apply multiple importance sampling (MIS) to combine the different path sampling techniques from PM and BPT, illustrated in Fig. 3, in a way that minimizes the variance of the resulting estimator. For a path of length k, BPT offers $k+2$ techniques, corresponding to different lengths of the light and eye sub-paths. Vertex merging adds $k-1$ more techniques, corresponding to "merging" at different interior vertices on the path.[2] We call the combined rendering algorithm, described below, *vertex connection and merging* (VCM).

This section presents a formulation of VCM that assumes a fixed radius r. Since VCM includes biased estimators, its combined estimator will also be biased for any merging radius $r > 0$. However, in Section 6 we present a progressive version of the algorithm that converges to the true value as r approaches zero.

5.1 Mathematical Formulation

A pixel estimator of (2), combining weighted contributions C_{VC} from vertex connection[3] estimators $\langle I \rangle_{\text{VC}}$ and weighted contributions C_{VM} from vertex merging estimators $\langle I \rangle_{\text{VM}}$ using MIS, reads:

$$\begin{aligned}\langle I \rangle_{\text{VCM}} &= C_{\text{VC}} + C_{\text{VM}} \\ &= \frac{1}{n_{\text{VC}}} \sum_{l=1}^{n_{\text{VC}}} \sum_{s\geq 0, t\geq 0} w_{\text{VC},s,t}(\overline{\mathbf{x}}_l)\, \langle I \rangle_{\text{VC}}(\overline{\mathbf{x}}_l) + \\ &\quad \frac{1}{n_{\text{VM}}} \sum_{l=1}^{n_{\text{VM}}} \sum_{s\geq 2, t\geq 2} w_{\text{VM},s,t}(\overline{\mathbf{x}}_l)\, \langle I \rangle_{\text{VM}}(\overline{\mathbf{x}}_l),\end{aligned} \tag{9}$$

which is an extended version of the BPT estimator [Veach 1997, p. 300]. It considers an eye sub-path through the pixel, whose vertices are connected to the vertices of n_{VC} light sub-paths and potentially merged with the vertices of n_{VM} light sub-paths. Term evaluations, dependent on whether vertex connection or merging is used, are denoted by subscripts VC and VM, respectively. The subscript s,t corresponds to a path constructed from a light sub-path with s vertices and an eye sub-path with t vertices. The power heuristic weight for technique v,s,t has the usual form

$$w_{v,s,t}(\overline{\mathbf{x}}) = \frac{n_v^\beta\, p_{v,s,t}^\beta(\overline{\mathbf{x}})}{n_{\text{VC}}^\beta \sum_{s'\geq 0,t'\geq 0} p_{\text{VC},s',t'}^\beta(\overline{\mathbf{x}}) + n_{\text{VM}}^\beta \sum_{s'\geq 2,t'\geq 2} p_{\text{VM},s',t'}^\beta(\overline{\mathbf{x}})}, \tag{10}$$

where v can be VC or VM. Note that the weight of a technique is amplified by the total number of samples, i.e. light paths, it uses.

[2] In practice, we do not consider merging at the path end-vertices since directly evaluating light emission or sensor sensitivity is usually more efficient.

[3] We consider here unidirectional sampling as a special case of vertex connection with zero vertices on the light or eye path, respectively (Section 3).

```
1:  function RENDER(r)
2:      ▶ Stage 1: Light path sampling
3:      lightPaths = TRACELIGHTPATHS(pixelCount)
4:      CONNECTTOEYE(lightVertices)
5:      BUILDRANGESEARCHSTRUCT(lightVertices)
6:      ▶ Stage 2: Eye path sampling and pixel estimator construction
7:      for i = 1 to pixelCount do
8:          eyeVertex = TRACERAY(SAMPLEPIXEL(i))
9:          while eyeVertex is valid do
10:             ▶ Unidirectional sampling (US)
11:             if eyeVertex is emissive then
12:                 ACCUM(eyeVertex, US, r, i)
13:             end if
14:             ▶ Vertex connection (VC)
15:             for lightVertex in lightPaths[i] ∪ SAMPLELIGHTPOINT() do
16:                 ACCUM(CONNECT(eyeVertex, lightVertex), VC, r, i)
17:             end for
18:             ▶ Vertex merging (VM)
19:             for lightVertex in RANGESEARCH(eyeVertex, r) do
20:                 ACCUM(MERGE(eyeVertex, lightVertex), VM, r, i)
21:             end for
22:             eyeVertex = CONTINUERANDOMWALK(eyeVertex)
23:         end while
24:     end for
25: end function
26: ▶ Accumulates the pixel measurement estimate due to a given path
27: function ACCUM(path, technique, r, i)
28:     contrib = MEASUREMENTCONTRIBUTION(path, technique, r)
29:     pdf = PDF(path, technique, r)
30:     weight = POWERHEURISTIC(path, technique, pdf)
31:     image[i] += weight * contrib / pdf
32: end function
```

Figure 6: *Pseudocode for rendering an image using our combined VCM algorithm, given a maximum vertex merging distance r.*

The computational efficiency of vertex merging allows in practice n_{VM} to be much higher than n_{VC}. Usually, n_{VM} would be the total number of light sub-paths for the image and $n_{\mathrm{VC}} = 1$. As discussed in Section 4.1, path reuse and the consequent brute-force variance reduction is the key to the efficiency of the vertex merging technique.

5.2 Implementation

Since path sampling is expensive, it is desirable to amortize this effort. The BPT implementation according to Veach [1997] reuses sub-paths by connecting every eye sub-path vertex to every vertex on one light sub-path. Vertex merging lends itself to a substantially more efficient path reuse scheme: thanks to the low cost of range query, an eye sub-path vertex can be potentially merged with vertices of a large number of pre-generated light sub-paths. To maximize path reuse, our algorithm runs in two stages, separating the sampling of the light and eye sub-paths, just like in photon mapping.

An outline of the algorithm is given in Fig. 6. In the *first stage*, we trace a number of sub-paths from the light sources, connect their vertices to the eye, and build a range search data structure over them (lines 3-5). In the *second stage*, we trace eye sub-paths by performing a random walk for each pixel. Upon sampling an eye vertex, we check if it lies on a light source and possibly accumulate the emitted radiance (lines 11-13). Then, we connect it to the vertices of one of the pre-generated light sub-paths, similarly to BPT (lines 15-17). To reduce correlation, we follow Veach [1997] and do not store the first vertex of a light sub-path, instead connecting every eye vertex to a new, randomly sampled point on a light source. Finally, we perform a range search to merge all light sub-path vertices that lie within the r-neighborhood of the current eye vertex (lines 19-21), similarly to photon mapping. We construct the estimate for each generated full path, evaluate its MIS weight, and accumulate the weighted estimate into the running pixel estimate (lines 27-32). Most of the terms required to evaluate path contributions and pdfs are stored with the sub-path vertices for improved efficiency. Please check our open-source reference implementation for more details.

6 Achieving Consistency

Apart form the usual variance in the form of noise, the images generated by our combined algorithm described above contain systematic error (bias) in the form of blur, inherited from vertex merging. In this section, we show that the combined algorithm can be made consistent by progressively reducing the merging radius r and accumulating the resulting images, such that the variance and bias vanish in the limit, much like in the progressive photon mapping (PPM) algorithm proposed by Knaus and Zwicker [2011].

The progressive variant of the pixel value estimator (9) corresponds to averaging the results of N independent rendering iterations:

$$\langle I \rangle_{\mathrm{VCM}} = \frac{1}{N} \sum_{i=1}^{N} \left(C_{\mathrm{VC},i} + C_{\mathrm{VM},i} \right), \tag{11}$$

where $C_{\mathrm{VC},i}$ and $C_{\mathrm{VM},i}$ are as in (9), but use a new set of light and eye sub-paths for each iteration i, and a reduced merging radius $r_i = r_1 \sqrt{i^{\alpha-1}}$, where $\alpha \in (0; 1)$ is a user parameter. This simple radius reduction scheme, derived in the supplemental document, is asymptotically equivalent to that of Knaus and Zwicker [2011], but is easier to compute for each iteration independently.

6.1 Asymptotic Error Analysis

As discussed earlier, an important advantage of BPT over PPM is its higher error convergence rate. This rate measures how fast the estimate approaches the true value with the growing number of samples N. BPT has a mean squared error (MSE) convergence rate of $O(1/N)$ [Veach 1997]. In Appendix B we show that the optimal MSE rate of PPM is $O(1/N^{2/3})$, reached for $\alpha = 2/3$. Since our progressive combined estimator (11) is a weighted average of vertex connection (VC) and vertex merging (VM) estimators, its MSE convergence rate must necessarily lie between the rates for BPT and PPM. We show next that, in fact, the MSE of the combined estimator converges asymptotically as fast as that of BPT for light paths that can be sampled by BPT.

We first perform asymptotic simplifications w.r.t. the iteration counter i, using $r_i = O(\sqrt{i^{\alpha-1}})$ from our radius reduction scheme. Knaus and Zwicker [2011] have shown that PPM, and thus VM, estimators have $\mathrm{Var}[\langle I \rangle_{\mathrm{VM}}] = O(1/r_i^2)$ and $\mathrm{Bias}[\langle I \rangle_{\mathrm{VM}}] = O(r_i^2)$. In contrast, the unbiased VC estimators are independent of r_i:

$$\begin{aligned} \mathrm{Var}[\langle I \rangle_{\mathrm{VC}}] &= O(1) & \mathrm{Bias}[\langle I \rangle_{\mathrm{VC}}] &= 0 \\ \mathrm{Var}[\langle I \rangle_{\mathrm{VM}}] &= O(i^{1-\alpha}) & \mathrm{Bias}[\langle I \rangle_{\mathrm{VM}}] &= O(i^{\alpha-1}). \end{aligned} \tag{12}$$

For any path $\overline{\mathbf{x}}$ and any s and t, we have $p_{\mathrm{VC},s,t}(\overline{\mathbf{x}}) = O(1)$, as the path pdf for VC is independent of r_i. From equation (8) it follows that $p_{\mathrm{VM},s,t}(\overline{\mathbf{x}}) = O(i^{\alpha-1})$. Substituting in (10), and using $\alpha - 1 < 0$:

$$\begin{aligned} w_{\mathrm{VC},s,t}(\overline{\mathbf{x}}) &= \frac{O(1)}{O(1) + O(i^{\beta(\alpha-1)})} = O(1) \\ w_{\mathrm{VM},s,t}(\overline{\mathbf{x}}) &= \frac{O(i^{\beta(\alpha-1)})}{O(1) + O(i^{\beta(\alpha-1)})} = O(i^{\beta(\alpha-1)}). \end{aligned} \tag{13}$$

Recall that β is the power heuristic parameter. Note that while the weights of VC techniques are asymptotically constant, the VM weights decrease as the iteration index i increases.

Variance. The variance of the progressive estimator (11) is

$$\mathrm{Var}[\langle I \rangle_{\mathrm{VCM}}] = \frac{1}{N^2} \sum_{i=1}^{N} \left(\mathrm{Var}[C_{\mathrm{VC},i}] + \mathrm{Var}[C_{\mathrm{VM},i}] \right). \tag{14}$$

The sums over sub-paths in $C_{\mathrm{VC},i}$ and $C_{\mathrm{VM},i}$ (see (9)) can be simplified away, as they are independent of i. $\mathrm{Var}[C_{\mathrm{VC},i}]$ and $\mathrm{Var}[C_{\mathrm{VM},i}]$

now reduce to the variances (12) weighted by the weights (13):

$$\begin{aligned}\mathrm{Var}[\langle I\rangle_{\mathrm{VCM}}] &= \frac{1}{N^2}\sum_{i=1}^{N}\Big(O(1)O(1)+O(i^{2\beta(\alpha-1)})O(i^{1-\alpha})\Big)\\ &= \frac{1}{N^2}\sum_{i=1}^{N}O(1)+\frac{1}{N^2}\sum_{i=1}^{N}O(i^{(2\beta-1)(\alpha-1)})\\ &= \frac{1}{N^2}NO(1)+\frac{1}{N^2}NO(N^{(2\beta-1)(\alpha-1)})\\ &= O(N^{-1})+O(N^{2\beta(\alpha-1)-\alpha}) = O(N^{-1}),\end{aligned} \tag{15}$$

where in the last step we assume $2\beta(\alpha-1)-\alpha<-1$. For practical values of $\beta\geq 1$ this inequality holds, since $\alpha\in(0;1)$. This means that the variance of the combined estimator is independent of α, and in fact has as high order as the variance of BPT.

Bias. For the bias of the estimator (11) we analogously obtain:

$$\begin{aligned}\mathrm{Bias}[\langle I\rangle_{\mathrm{VCM}}] &= \frac{1}{N}\sum_{i=1}^{N}\left(\mathrm{Bias}[C_{\mathrm{VC},i}]+\mathrm{Bias}[C_{\mathrm{VM},i}]\right)\\ &= \frac{1}{N}\sum_{i=1}^{N}\Big(0+O(i^{\beta(\alpha-1)})O(i^{\alpha-1})\Big) = O(N^{(\beta+1)(\alpha-1)}).\end{aligned} \tag{16}$$

This means that the bias of the combined estimator diminishes faster than that of PPM for the same α value, since $\beta>0$ and $(\alpha-1)<0$.

Mean squared error (MSE). Finally, for the MSE of (11), which measures the total expected error of the estimate, we get:

$$\begin{aligned}\mathrm{MSE}[\langle I\rangle_{\mathrm{VCM}}] &= \mathrm{Var}[\langle I\rangle_{\mathrm{VCM}}]+\mathrm{Bias}[\langle I\rangle_{\mathrm{VCM}}]^2\\ &= O(N^{-1})+O(N^{2(\beta+1)(\alpha-1)}),\end{aligned} \tag{17}$$

which, for $\alpha\leq\frac{2\beta+1}{2\beta+2}$, has an optimal convergence rate of $O(N^{-1})$, equal to that of unbiased estimators. When using the balance heuristic, i.e. $\beta=1$, this optimal error convergence rate is achieved for *any* $\alpha\in(0;0.75]$. This means that our combined algorithm inherits the higher convergence rate from bidirectional path tracing, and is thus *asymptotically faster than progressive photon mapping*, whose maximum MSE convergence rate is only $O(1/N^{2/3})$. Moreover, this result is achieved for a wide range of values for the parameter α. Note that this result only holds for paths that can be sampled by BPT; the contribution of paths without two subsequent non-specular vertices, starting on a point source and ending on a pinhole camera [Veach 1997], still converge at the slower rate of PPM.

Discussion. The intuition behind the high convergence rate of our progressive VCM algorithm is that the contribution of vertex merging (VM) in the progressive estimate diminishes as the number of iterations grows, as illustrated in Fig. 7. As we progressively shrink the radius r_i, we increase variance of VM (equation (12)). The power heuristic automatically compensates for this by assigning a reciprocal weight (equation (13)). The resulting algorithm is thus asymptotically equivalent to BPT. However, VM brings efficient initial variance reduction, which helps to faster achieve acceptable image quality with a finite number of samples, as we show next.

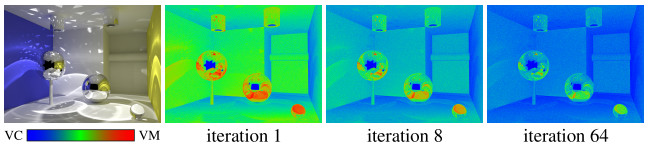


Figure 7: *Relative contributions of vertex connection (VC) and vertex merging (VM) at different rendering iterations, indicating that VM's contribution to the average estimate diminishes over time.*

7 Results

This section presents an empirical evaluation of our method. We compare three algorithms in the paper: (1) bidirectional path tracing (BPT), (2) progressive photon mapping (PPM), and (3) our combined progressive vertex connection and merging (VCM). The supplemental material provides additional results for (4) path tracing, (5) Metropolis light transport, (6) an algorithm that combines BPT and PPM based on a classification of paths as caustic and non-caustic, as well as images with longer rendering time (30 minutes), relative reference error images, and perceptual difference measurements.

Setup. Rendering is performed progressively with one eye sub-path per pixel per iteration, at resolution 1024×768. Each iteration starts by sampling the same number of light sub-paths as there are image pixels (i.e. $\approx$ 786k). All light sub-paths are reused by the PPM and vertex merging estimators for each pixel, while for vertex connections in BPT and in our VCM algorithm we associate one light sub-path with each pixel (i.e. $n_{\mathrm{VM}}\approx$ 786k and $n_{\mathrm{VC}}=1$ in the combined estimator (9)). Our PPM implementation closely follows Hachisuka and Jensen [2009]. We use a radius reduction parameter $\alpha=2/3$ for both PPM and our VCM. All measurements have been obtained on a 4-core Intel Core i7-860 2.8GHz processor machine using a simple CPU ray tracer. We also found our algorithm easy to add on top of an existing GPU BPT implementation, and achieved a preliminarily speed-up of 6-10$\times$ over the CPU version.

Scenes. The **Living room** scene in Fig. 1 shows an environment with different "illumination scales". Most of the diffuse illumination comes from far away, resulting in excessive noise in PPM. The objects on the desk are lit by two local area light sources, which, in combination with the mirror and the vases, produce caustic paths that are difficult for BPT, but well handled by PPM. Our combined VCM algorithm handles all light paths robustly, automatically finding a balance between the many sampling techniques it has at its disposal.

Fig. 8 compares images of three scenes rendered by the three algorithms in 4 minutes. We compare unconverged images so that the differences are easily perceived. The reference images were obtained using PPM for the Mirror balls scene, while our VCM algorithm was used for the other scenes, as none of the other algorithms was able to provide noise-free results in a feasible amount of time.

The **Bathroom** scene is a compact environment with moderately glossy tiles and highly glossy metal elements, illuminated by two small area lights, each almost fully enclosed in a metal-glass shell. This configuration poses a challenge to BPT and PPM. Our VCM handles all light interactions robustly, as it not only adaptively combines the two methods, but also includes more sampling techniques – unlike PPM, it merges light vertices at *every* eye sub-path vertex. The benefit is clearly visible in the inset of the water tap in Fig. 8.

The highly glossy **Car** with specular chrome elements is placed in a semi-open studio environment, illuminated by two area lights from above. The interior is additionally lit by a small area light enclosed in a transparent shell. This illumination, seen through the windows, and the reflections of the caustics on ground in the exterior are difficult for BPT. On the other hand, PPM performs poorly on the diffuse illumination and on the glossy inter-reflections, e.g. between the tire and the fender. Our VCM renders a smooth low-noise image.

The **Mirror balls** scene is lit by concentrated light coming from the metal-enclosed area lights on the ceiling, focused by the lampshade lenses. The entirely caustic illumination, which is additionally reflected in the balls, makes this scene well suited for PPM. BPT still handles well the directly visible caustics via vertex connections to the camera, but relies on unidirectional sampling to find reflections of these caustics. Even though VCM is more than 2$\times$ slower per iteration than PPM, and thus slightly noisier on the reflected caustics, it once again delivers the highest overall image quality.

Convergence. To verify that our VCM algorithm converges to the correct solution, we measure the root mean squared (RMS) difference between the images produced by VCM and PPM on the Mirror balls scene. Fig. 9 left shows a log-log plot of the steadily decreasing difference over time. The difference image at the end of the measurement indicates that any remaining variations between the two solutions are only due to random noise. The log-log reference error plots for all three algorithms in Fig. 9 right clearly show that our combined VCM algorithm converges at a higher rate than PPM.

8 Discussion

Parameter choice. Two parameters that our progressive VCM algorithm inherits from PPM are the initial merging radius r_1 and its reduction rate α. For both PPM and our VCM, we globally set r_1 to $0.01\% - 0.07\%$ of the scene's bounding box. For a fair comparison against PPM, we use $\alpha = 2/3$ for all measurements. In general, for VCM we recommend setting r_1 smaller than for PPM and $\alpha = 0.75$. Such settings introduce less initial bias, and maximize the variance convergence rate of vertex merging and thus its efficiency too.

Limitations. Our VCM algorithm inherits from PM the ability to approximately capture SDS paths due to point light sources, which BPT cannot sample. Therefore, the convergence rate of the contribution of such paths in our progressive VCM is as low as in PPM. This limitation does not apply to our scenes, as we use only physically plausible light sources with finite area. Slightly counter-intuitively, this means that, in the presence of specular objects, area light sources are asymptotically cheaper to handle than point lights.

While our combined VCM algorithm is more robust than each of its ingredients alone, it does not perform better on transport paths that are poorly handled by both BPT and PM. Such paths are, for example, caustics falling on a highly glossy surface. Efficient handling of such cases is a challenging avenue for future work. An interesting possibility would be to employ Markov Chain Monte Carlo techniques on top of VCM, e.g. [Jakob and Marschner 2012], which would also improve its efficiency in highly occluded scenes.

Concurrent work. The concurrent work of Hachisuka et al. [2012] also combines BPT and PM using multiple importance sampling (MIS). Their resulting combination is equivalent to ours, but as a major difference to our approach, they express BPT path pdfs in an extended, higher-dimensional space, by considering random path vertex perturbations. In addition, they analyze MIS weighting in the presence of biased sampling techniques. Our derivations focus on the optimal asymptotic behavior of the progressive estimator (11).

9 Conclusion

Our work addresses efficient light transport simulation under a wide variety of lighting configurations, including specular-diffuse-specular light paths. The key idea is to reformulate photon mapping in the path integral framework, which enables the use of multiple importance sampling for its efficient combination with bidirectional path tracing in a single, unified rendering algorithm. We show that this algorithm is more robust than bidirectional path tracing and photon mapping alone, while achieving a higher order of convergence than progressive photon mapping. We believe that our solution has an immediate practical utility in a wide range of applications, especially in predictive rendering systems, both interactive and offline, where robustness under different lighting setups is critical.

Acknowledgements. We would like to thank the reviewers for their insightful comments, Miloš Hašan and Wenzel Jakob for the fruitful discussions, and Ondřej Karlík and Jiří Vorba for proofreading the paper. The Mirror balls scene has been modeled by Toshiya Hachisuka. Chaos Group kindly provided the Bathroom scene.

References

BEKAERT, P., SLUSSALEK, P., COOLS, R., HAVRAN, V., AND SEIDEL, H.-P. 2003. A custom designed density estimator for light transport. Tech. rep., Max-Planck-Institut für Informatik.

DAVIDOVIČ, T., KŘIVÁNEK, J., HAŠAN, M., SLUSALLEK, P., AND BALA, K. 2010. Combining global and local virtual lights for detailed glossy illumination. *ACM Trans. Graph. 29*, 6 (Dec.).

DUTRÉ, P., LAFORTUNE, E. P., AND WILLEMS, Y. 1993. Monte Carlo light tracing with direct computation of pixel intensities. In *Compugraphics '93*, 128–137.

FAN, S., CHENNEY, S., AND CHI LAI, Y. 2005. Metropolis photon sampling with optional user guidance. In *Eurographics Symposium on Rendering*, Eurographics Association, 127–138.

GEORGIEV, I., KŘIVÁNEK, J., AND SLUSALLEK, P. 2011. Bidirectional light transport with vertex merging. In *SIGGRAPH Asia 2011 Sketches*, ACM, New York, NY, USA, 27:1–27:2.

HACHISUKA, T., AND JENSEN, H. W. 2009. Stochastic progressive photon mapping. *ACM Trans. Graph. 28*, 5 (Dec.), 141:1–141:8.

HACHISUKA, T., AND JENSEN, H. W. 2011. Robust adaptive photon tracing using photon path visibility. *ACM Trans. Graph. 30* (October), 114:1–114:11.

HACHISUKA, T., OGAKI, S., AND JENSEN, H. W. 2008. Progressive photon mapping. *ACM Trans. Graph. 27*, 5 (Dec.).

HACHISUKA, T., JAROSZ, W., AND JENSEN, H. W. 2010. A progressive error estimation framework for photon density estimation. *ACM Trans. Graph. 29*, 6 (Dec.), 144:1–144:12.

HACHISUKA, T., PANTALEONI, J., AND JENSEN, H. W. 2012. A path space extension for robust light transport simulation. *ACM Trans. Graph. 31* (December).

HAŠAN, M., PELLACINI, F., AND BALA, K. 2007. Matrix row-column sampling for the many-light problem. *ACM Trans. Graph. 26*, 3 (July).

HAŠAN, M., KŘIVÁNEK, J., WALTER, B., AND BALA, K. 2009. Virtual spherical lights for many-light rendering of glossy scenes. *ACM Trans. Graph. 28*, 5 (Dec.), 143:1–143:6.

JAKOB, W., AND MARSCHNER, S. 2012. Manifold exploration: A Markov chain Monte Carlo technique for rendering scenes with difficult specular transport. *ACM Trans. Graph. 31*, 4.

JENSEN, H. W. 2001. *Realistic Image Synthesis Using Photon Mapping*. A. K. Peters, Ltd., Natick, MA, USA.

KAJIYA, J. 1986. The rendering equation. *SIGGRAPH '86*.

KELLER, A. 1997. Instant radiosity. In *SIGGRAPH '97*, 49–56.

KNAUS, C., AND ZWICKER, M. 2011. Progressive photon mapping: A probabilistic approach. *ACM Trans. Graph. 30* (May).

KOLLIG, T., AND KELLER, A. 2000. Efficient bidirectional path tracing by randomized quasi-monte carlo integration. In *MCQMC Methods 2000*, 290–305.

KOLLIG, T., AND KELLER, A. 2004. Illumination in the presence of weak singularities. In *MCQMC Methods 2004*.

KŘIVÁNEK, J., FERWERDA, J. A., AND BALA, K. 2010. Effects of global illumination approximations on material appearance. *ACM Trans. Graph. 29*, 4 (July), 112:1–112:10.

LAFORTUNE, E., AND WILLEMS, Y. D. 1993. Bi-directional path tracing. In *Compugraphics '93*, 145–153.

OU, J., AND PELLACINI, F. 2011. Lightslice: matrix slice sampling for the many-lights problem. *ACM Trans. Graph. 30*, 6 (Dec.).

TOKUYOSHI, Y. 2009. Photon density estimation using multiple importance sampling. In *SIGGRAPH Asia 2009 Posters*.

VEACH, E., AND GUIBAS, L. 1994. Bidirectional estimators for light transport. In *Eurographics Workshop on Rendering*.

VEACH, E., AND GUIBAS, L. 1995. Optimally combining sampling techniques for Monte Carlo rendering. In *SIGGRAPH '95*.

VEACH, E., AND GUIBAS, L. J. 1997. Metropolis light transport. In *SIGGRAPH '97*.

VEACH, E. 1997. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Standford Univeristy.

VORBA, J. 2011. Bidirectional photon mapping. In *Proc. of the Central European Seminar on Computer Graphics (CESCG '11)*.

WALTER, B., ARBREE, A., BALA, K., AND GREENBERG, D. P. 2006. Multidimensional lightcuts. *ACM Trans. Graph. 25*, 3.

WALTER, B., KHUNGURN, P., AND BALA, K. 2012. Bidirectional lightcuts. *ACM Trans. Graph. 31*, 4 (July), 59:1–59:11.

A Additional Vertex Merging Derivations

This appendix derives the measurement contribution function corresponding to the photon mapping (PM) path sampling techniques discussed in Section 4, as well as the path integral estimated by PM.

A.1 Contribution Function for Extended Paths

We first write the pixel measurement estimator corresponding to the PM radiance estimate (5). Its expected value is the path integral estimated. Finally, the path integral's integrand is defined to be the desired contribution function. As in Section 4, we fix the path length k and the vertex index s where the radiance estimate is performed.

PM pixel measurement estimator. The contribution of a photon at $\mathbf{x}_s^*$ to a pixel via a radiance estimate (5) performed at $\mathbf{x}_s$ is:

$$\langle I_{k,s,r}\rangle = \Phi(\mathbf{x}_s^*) \underbrace{\left[K_r(\|\mathbf{x}_s^*-\mathbf{x}_s\|)\rho_{\mathrm{s}}(\mathbf{x}_s^*,\mathbf{x}_s)\right]}_{Q_r(\mathbf{x}_s^*,\mathbf{x}_s)\ \stackrel{\mathrm{def}}{=}\ \text{photon weight}} W(\mathbf{x}_s). \tag{18}$$

$\rho_{\mathrm{s}}(\mathbf{x}_s^*,\mathbf{x}_s) = \rho_{\mathrm{s}}(\mathbf{x}_{s-1}{\to}\mathbf{x}_s^*,\mathbf{x}_s,\mathbf{x}_s{\to}\mathbf{x}_{s+1})$ denotes the BSDF at $\mathbf{x}_s$ evaluated for the incoming direction $\mathbf{x}_{s-1}{\to}\mathbf{x}_s^*$ and outgoing direction $\mathbf{x}_s{\to}\mathbf{x}_{s+1}$, just like in the PM radiance estimate. The flux $\Phi(\mathbf{x}_s^*)$ carried by the photon, and the cumulative importance $W(\mathbf{x}_s)$, expressed using the notation defined in Section 3, are [Jensen 2001]:

$$\Phi(\mathbf{x}_s^*)=\frac{L_{\mathrm{e}}(\mathbf{x}_0)T(\mathbf{x}_0\dots\mathbf{x}_s^*)}{p(\mathbf{x}_0)\dots p(\mathbf{x}_{s-1}{\to}\mathbf{x}_s^*)}\quad W(\mathbf{x}_s)=\frac{T(\mathbf{x}_s\dots\mathbf{x}_k)W_{\mathrm{e}}(\mathbf{x}_k)}{p(\mathbf{x}_k)\dots p(\mathbf{x}_{s+1}{\to}\mathbf{x}_s)}.$$

Path integral and contribution function. The path integral actually estimated by the estimator (18) is given by its expected value:

$$I_{k,s,r} = E[\langle I_{k,s,r}\rangle] = \int_{\mathcal{M}^{k+2}} f_{k,s,r}(\overline{\mathbf{x}}^*)\,d\mu_{k+2}(\overline{\mathbf{x}}^*) \tag{19}$$

where $\mathcal{M}$ is the scene surface and $d\mu_i$ denotes the differential area product measure on the space $\mathcal{M}^i$. We define $f_{k,s,r}$ to be the *measurement contribution of extended paths*:

$$f_{k,s,r}(\overline{\mathbf{x}}^*)=L_{\mathrm{e}}(\mathbf{x}_0)T(\mathbf{x}_0\dots\mathbf{x}_s^*)Q_r(\mathbf{x}_s^*,\mathbf{x}_s)T(\mathbf{x}_s\dots\mathbf{x}_k)W_{\mathrm{e}}(\mathbf{x}_k).$$

Note the extra area integration in (19) compared to the regular path integral (2), for a given path length k. This integration over the possible positions of vertex $\mathbf{x}_s^*$, i.e. the photon, corresponds to blurring by the kernel K_r, a well known effect of kernel density estimation.

A.2 Reducing the Path Integral Dimension

As discussed in Section 4, we need to express the PM path integral (19) as an integral over regular, not extended paths. We achieve this by considering the extra area integral in (19) as a nested integration problem to which the path integral is oblivious:

$$\begin{aligned} I_{k,s,r} &= \int_{\mathcal{M}^{k+1}} \left[\int_{\mathcal{M}} f_{k,s,r}(\overline{\mathbf{x}}^*)\,dA(\mathbf{x}_s^*)\right] d\mu_{k+1}(\overline{\mathbf{x}}) \\ &= \int_{\mathcal{M}^{k+1}} F_{k,s,r}(\overline{\mathbf{x}})\,d\mu_{k+1}(\overline{\mathbf{x}}), \end{aligned} \tag{20}$$

Here $\overline{\mathbf{x}}$ is a regular light path created from an extended path $\overline{\mathbf{x}}^*$ by leaving out the $\mathbf{x}_s^*$ vertex and concatenating all the remaining vertices. The function $F_{k,s,r}$ evaluates the contribution of a regular path $\overline{\mathbf{x}}$ by blurring in an r-neighborhood around $\mathbf{x}_s$, described by an area integral of $f_{k,s,r}$, i.e. $F_{k,s,r}(\overline{\mathbf{x}}) = \int_{\mathcal{M}} f_{k,s,r}(\overline{\mathbf{x}}^*)\,dA(\mathbf{x}_s^*)$. (Note that it is the small support of the kernel K_r inside $f_{k,s,r}$ that effectively limits the integration to the neighborhood of $\mathbf{x}_s$.)

Formally, we define *vertex merging* as a path sampling technique for Monte Carlo estimation of (20). That is, VM samples *regular* paths, and uses $F_{k,s,r}(\overline{\mathbf{x}})$ as the contribution function. The path pdf is derived in Section 4. We now focus on the evaluation of $F_{k,s,r}(\overline{\mathbf{x}})$.

Contribution evaluation. The contribution function $F_{k,s,r}(\overline{\mathbf{x}})$ is itself defined as an integral (see equation (20)). We can obtain a one-sample MC estimate of $F_{k,s,r}(\overline{\mathbf{x}})$ using $\mathbf{x}_s^*$, i.e. the photon:

$$\langle F_{k,s,r}(\overline{\mathbf{x}})\rangle = f_{k,s,r}(\overline{\mathbf{x}}^*)\Big/\left[\frac{p(\mathbf{x}_{s-1}{\to}\mathbf{x}_s^*)}{\int_{\mathcal{A}_{\mathcal{M}}} p(\mathbf{x}_{s-1}{\to}\mathbf{x})d\mathbf{x}}\right], \tag{21}$$

where we normalize $p(\mathbf{x}_{s-1}{\to}\mathbf{x}_s^*)$ to obtain a valid pdf over $\mathcal{A}_{\mathcal{M}}$, since only points $\mathbf{x}_s^*$ inside this set survive the Russian roulette used to decide if the path should be accepted, as described in Section 4.

Vertex merging estimator. Let us now construct the estimator $\langle I_{k,s,r}\rangle_{\mathrm{VM}} = \langle F_{k,s,r}(\overline{\mathbf{x}})\rangle / p_{\mathrm{VM}}(\overline{\mathbf{x}})$, using the path pdf $p_{\mathrm{VM}}(\overline{\mathbf{x}})$:

$$\langle I_{k,s,r}\rangle_{\mathrm{VM}}=\frac{f_{k,s,r}(\overline{\mathbf{x}}^*)}{\left[\frac{p(\mathbf{x}_{s-1}{\to}\mathbf{x}_s^*)}{\int_{\mathcal{A}_{\mathcal{M}}} p(\mathbf{x}_{s-1}{\to}\mathbf{x})d\mathbf{x}}\right]p_{\mathrm{VM}}(\overline{\mathbf{x}})}=\frac{f_{k,s,r}(\overline{\mathbf{x}}^*)}{p(\mathbf{x}_{s-1}{\to}\mathbf{x}_s^*)p_{\mathrm{VC}}(\overline{\mathbf{x}})}.$$

We have arrived at an expression for the pixel estimator (18). This result shows that our reformulation of PM as a sampling technique for regular paths is compatible with the classic view of PM, because the final estimators for both views are the same. The importance of our view is that it clearly separates the path pdf from the contribution function, thereby allowing PM to be combined with BPT via MIS.

B Optimal Error Convergence Rate of PPM

The rates of the variance and bias of a progressive photon mapping estimator $\langle I\rangle_N$ after N iterations are $\mathrm{Var}[\langle I\rangle_N] = O(1/N^{\alpha})$ and $\mathrm{Bias}[\langle I\rangle_N] = O(1/N^{1-\alpha})$, where α is the user-specified parameter that trades off variance and bias [Knaus and Zwicker 2011, Appendices C, D, E, F]. Thus, the mean squared error (MSE) is:

$$\begin{aligned} \mathrm{MSE}[\langle I\rangle_N] &= \mathrm{Var}[\langle I\rangle_N] + \mathrm{Bias}[\langle I\rangle_N]^2 \\ &= O(1/N^{\alpha}) + O(1/N^{2(1-\alpha)}). \end{aligned} \tag{22}$$

The maximum MSE rate is then $O(1/N^{2/3})$, reached for $\alpha = 2/3$.

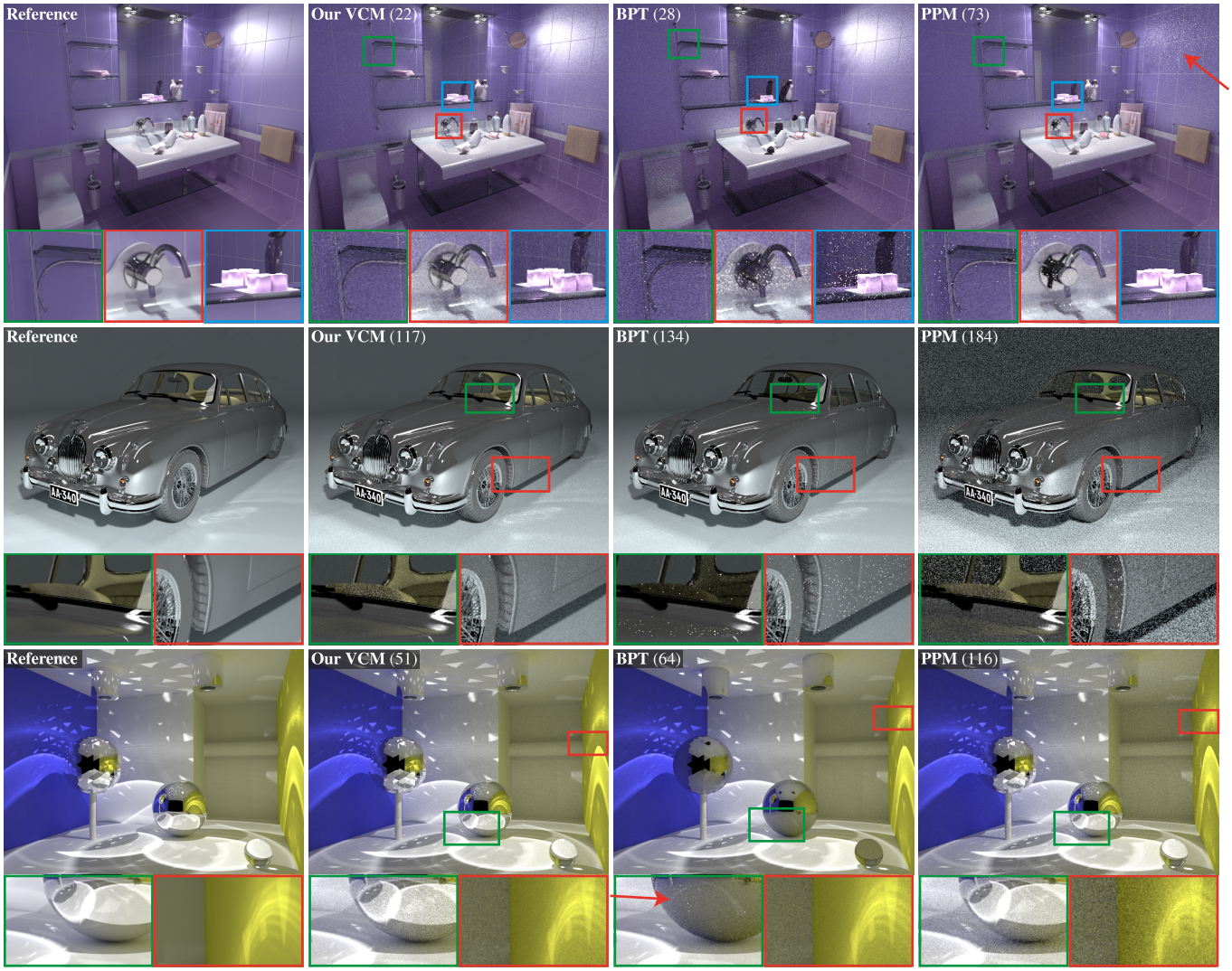


Figure 8: *Three scenes rendered using bidirectional path tracing (BPT), progressive photon mapping (PPM), and our combined algorithm (VCM) from Section 6. The reference images on the left have been rendered in 24 hours; the rest of the shots have been taken after 4 minutes of rendering. The numbers in parentheses denote the number of rendering iterations, proportional to the total number of samples, taken by each algorithm in the given time. The difficult lighting in these scenes, due to the complex light interactions between diffuse, glossy and specular objects, is challenging for both BPT and PPM. Our VCM algorithm employs more sampling techniques than BPT and PPM together, and handles all light interactions robustly by finding a good mixture of them for each individual light path. The achieved visual robustness is clearly visible in the insets. (Readers of the electronic version are encouraged to zoom in the document for closer inspection.)*

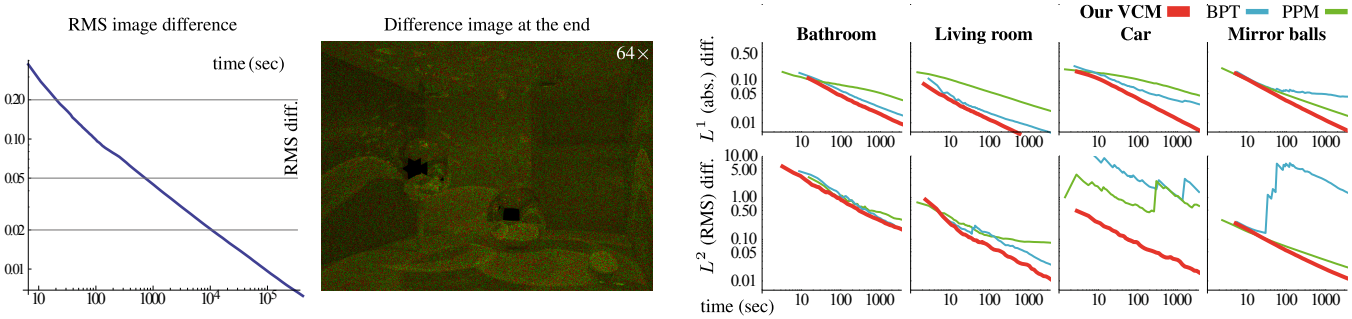


Figure 9: ***Left****: A comparison between our progressive VCM algorithm against PPM on the Mirror balls scene. The RMS image difference plot shows that our algorithm and PPM converge to the same solution. The 64× amplified colorcoded difference image, taken at the end of the measurement, indicates that any remaining differences are due to random noise.* ***Right****: Mean absolute (L^1) and root mean squared (L^2) log-log reference difference plots for the three algorithms compared on the four scenes shown in Figures 1 and 8. The plots show that our combined VCM algorithm converges at a higher rate than PPM. The oscillations in the plots are due to the 'fireflies' caused by low sampling probability of high contribution paths which increase the error of the produced image when found occasionally. For BPT and PPM, these occur often on glossy surfaces; for our VCM algorithm, these are highly glossy and perfectly specular (i.e. LS^+E) unidirectional paths.*