Segment-based Light Transport Simulation

WENYOU WANG, University of Waterloo, Canada
REX WEST, Aoyama Gakuin University, Japan
TOSHIYA HACHISUKA, University of Waterloo, Canada

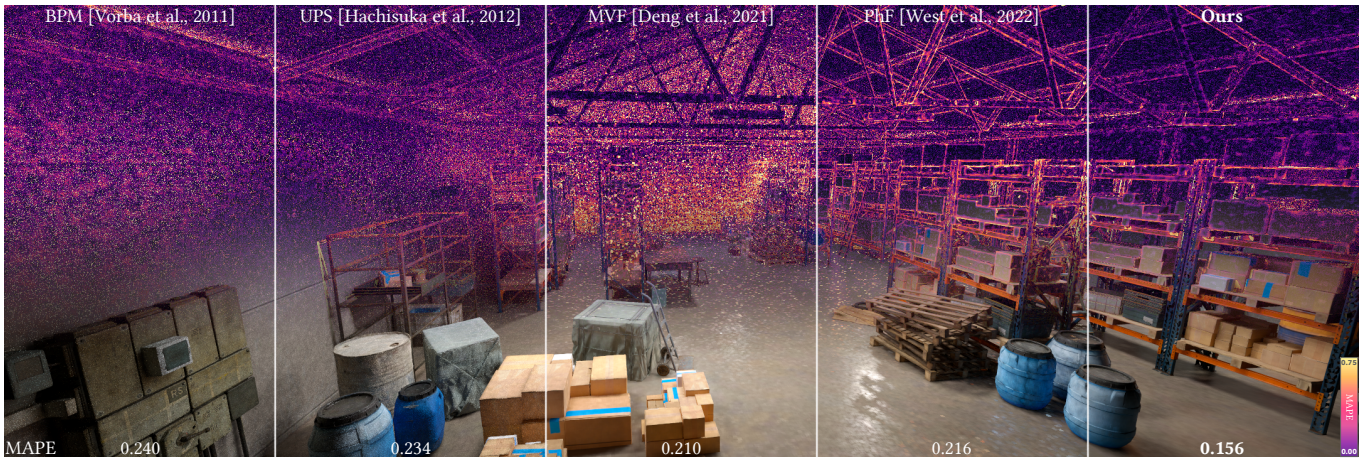


Fig. 1. Our segment-based formulation and estimation strategies open the door to new rendering methods that use segments as the basic unit of light transport. Here we show an equal-time (250s) comparison of a novel, segment-based bidirectional path filtering method (Ours) to existing state-of-the-art filtering methods, Multi-vertex Filtering (MVF)[Deng et al. 2021] and Photon Filtering (PhF)[West et al. 2022], and two robust bidirectional rendering methods, Bidirectional Photon Mapping (BPM)[Vorba 2011] and Unified Path Sampling (UPS)[Hachisuka et al. 2012]. Our segment-based approach robustly handles the predominantly indirect lighting and complex occlusion of the Warehouse scene, demonstrating significantly reduced estimation error over prior methods.

We propose a novel segment-based light transport framework that uses segments as the basic unit of light transport. Unlike vertex-based formulations, our segment-based formulation naturally accommodates the disconnected subpaths encountered in photon density estimation and path filtering methods, and opens the door to a wide range of new rendering methods that consider segments as a sampling primitive. To facilitate the development of segment-based rendering methods, we introduce several segment sampling techniques and estimation strategies, including a highly-performant recursive estimator. One of our key contributions is a general-purpose segment sampling framework based on marginal multiple importance sampling (MMIS). To demonstrate the practicality of our sampling framework, we show how it allows us to easily implement a robust bidirectional path filtering method — challenging under a vertex-based formulation — achieving superior filtering efficiency and convergence compared to state-of-the-art approaches.

Authors' addresses: Wenyou Wang, University of Waterloo, Canada, wenyouwang@outlook.com; Rex West, Aoyama Gakuin University, Japan, rexwest@gmail.com; Toshiya Hachisuka, University of Waterloo, Canada, toshiya.hachisuka@uwaterloo.ca.

0730-0301/2025/8-ART $15.00
https://doi.org/10.1145/3730847

CCS Concepts: • **Computing methodologies** → **Rendering**; **Ray tracing**.

Additional Key Words and Phrases: rendering, physically-based rendering, path sampling, multiple importance sampling

ACM Reference Format:
Wenyou Wang, Rex West, and Toshiya Hachisuka. 2025. Segment-based Light Transport Simulation. *ACM Trans. Graph.* 44, 4 (August 2025), 10 pages. https://doi.org/10.1145/3730847

1 INTRODUCTION

Monte Carlo (MC) rendering classically simulates light transport by sampling light transport paths (sequences of vertices) that connect light sources and sensors. Modern rendering methods such as photon density estimation [Jensen 1996; Hachisuka et al. 2008; Hachisuka and Jensen 2009] and path filtering [Keller et al. 2014; Binder et al. 2018, 2019; West et al. 2020; Deng et al. 2021; West et al. 2022], however, sample *disconnected* light transport paths first to *approximately* construct complete (connected) light transport paths later. Such methods often enable efficient reuse of multiple disconnected subpaths based on proximity, but a proper formulation that supports extra vertices at disconnections is necessary. To address such extra vertices, Georgiev et al. [2012] introduced the concept of vertex merging into photon density estimation that can be interpreted as rejection sampling with an associated probability of acceptance. West et al. [2020, 2022] alternatively formulated extra vertices in path filtering as marginalized variables in a continuous domain of path sampling techniques.

Another formulation to address those extra vertices is the extended path integral formulation [Hachisuka et al. 2012]. This formulation extends the space of the integration problem in the original path integral formulation [Veach 1997] to directly support disconnected subpaths as simple MC samples in the extended path space. Hachisuka et al. [2012] also pointed out that this formulation theoretically supports multiple disconnections along a path. In its extreme case, one can consider disconnection at *every single* vertex along a path, turning sampling of a path into sampling of a sequence of disconnected *segments*. Despite this segment-based view in its most generalized form, Hachisuka et al. [2012] only considered using one disconnection along a path as in photon density estimation or path filtering, losing its connection to sampling of segments. The core nature of segments as a *sampling primitive* in light transport simulation has largely been left unexplored.

We revisit this segment-based path integral formulation, introduce a first practical segment sampling framework based on marginal multiple importance sampling (MMIS) [West et al. 2022], and explore several segment-based rendering methods. Our segment-based light transport simulation opens up many unexplored options of more efficient methods. As one example of the practicality of our framework, we show how it enables a performant and consistent estimator of *bidirectional* path filtering — something that is not easily achievable without our segment-based approach. Our bidirectional path filtering method outperforms both the underlying sampling algorithm and state-of-the-art path filtering methods.

Concretely, our contributions are:

- a segment-based path integral formulation,
- sampling techniques for segments and segment paths,
- practical estimation strategies and implementable algorithms for the proposed formulation, and
- a robust bidirectional path filtering method that out performs the state-of-the-art path reuse rendering methods

2 BACKGROUND

The path integral formulation [Veach 1997] defines the intensity I_ρ of each pixel ρ as an integral over the space $\mathcal{P}$ of all possible light transport paths that can travel through a virtual scene,

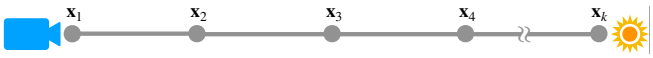
$$I_\rho = \int_{\mathcal{P}} f_\rho(\overline{\mathrm{x}})\,\mathrm{d}\overline{\mathrm{x}} = \sum_{k=1}^{\infty} \underbrace{\int_{\mathcal{P}_k} f_\rho(\overline{\mathrm{x}})\,\mathrm{d}\overline{\mathrm{x}}}_{=I_k}, \tag{1}$$

where $\mathcal{P}_k$ is the set of paths $\overline{\mathrm{x}}$ of k vertices x, $\overline{\mathrm{x}} = \{\mathrm{x}_1 \dots \mathrm{x}_k\}$, and the contribution $f_\rho(\overline{\mathrm{x}})$ is

$$f_\rho(\overline{\mathrm{x}}) = W_\rho(\mathrm{x}_1, \mathrm{x}_2) \prod_{i=1}^{k} G(\mathrm{x}_i, \mathrm{x}_{i+1}) \prod_{i=2}^{k-1} f_r(\mathrm{x}_{i-1}, \mathrm{x}_i, \mathrm{x}_{i+1})\, L_e(\mathrm{x}_{k-1}, \mathrm{x}_k), \tag{2}$$

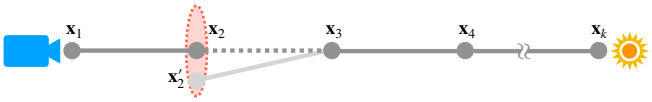
where $W_\rho(\mathrm{x}, \mathrm{y})$ is the sensor responsivity for the pixel ρ, $G(\mathrm{x}, \mathrm{y})$ is the geometry term, $f_r(\mathrm{x}, \mathrm{y}, \mathrm{z})$ is the bidirectional scattering distribution function (BSDF) (i.e., throughput), and $L_e(\mathrm{x}, \mathrm{y})$ is the light emitted from y towards x.

MC integration of the path integral (1) operates by sampling *paths* (i.e., a sequence of vertices). Each vertex x_i is connected via an edge to the following vertex x_{i+1} to construct a complete path.

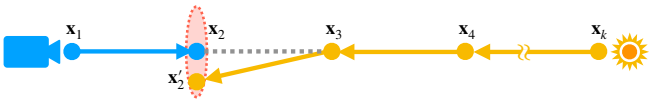


2.1 Disconnected Subpaths

Sampling a complete path that connects a light source and a sensor can become fundamentally challenging in certain cases. Bidirectional sampling of subpaths followed by connection of vertices [Veach and Guibas 1995; Lafortune and Willems 1993] is one approach to address such difficult cases, but Veach [1997] pointed out that such local sampling of subpaths is fundamentally unable to capture paths without two consecutive diffuse (D) events in the extended Heckbert notation [Heckbert 1990]. Hachisuka et al. [2008] later coined a term "SDS" (specular-diffuse-specular) paths to collectively refer to such challenging paths, and demonstrated how rendering methods that sample *disconnected* subpaths first and then approximately construct complete paths can be much more efficient at handling such paths. Under these methods, sub-paths with disconnected end points are used to form a complete path. In the following, disconnected subppaths refer to such cases, not those that can be easily handled by connections in bidirectional methods [Veach and Guibas 1995; Lafortune and Willems 1993],

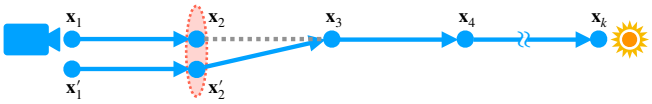


Photon density estimation. Photon density estimation methods [Jensen 1996; Hachisuka et al. 2008; Hachisuka and Jensen 2009] formulate this process as density estimation of light sub-paths over a region (i.e., the support of a kernel function).



Density estimation at disconnected sub-paths introduces bias from the solution to the path integral formulation. Hachisuka et al. [2008] and Hachisuka and Jensen [2009] proposed to progressively reduce the bandwidth of the density estimation kernel to make its bias convergent to zero in the limit. Knaus and Zwicker [2011] reformulated this progressive reduction a as probabilistic framework to simplify its implementation. Vorba [2011] showed how to combine multiple sampling techniques of a complete path at different disconnection points via MIS [Veach and Guibas 1995]. Many other approaches exist to address the problem of bias in photon density estimation [Kaplanyan and Dachsbacher 2013; Lin et al. 2020; Qin et al. 2015; Misso et al. 2022].

Path filtering. An alternative way to formulate handling of disconnected paths is path-space filtering [Keller et al. 2014; Binder et al. 2018, 2019; West et al. 2020; Deng et al. 2021; West et al. 2022], where unidirectional sub-path samples are combined over a filtering kernel to construct novel paths.



West et al. [2020] reinterpret the sub-path reuse of path filtering as conditional sampling on auxiliary variables, replacing the uniform kernel functions with a weighting function based on stochastic multiple importance sampling (SMIS). To further amortize sampling cost and improve reuse efficiency, Deng et al. [2021] introduce *path graphs* with an efficient propagation-based algorithm for filtering at multiple vertices along a path, and West et al. [2022] show that multi-vertex path filtering is equivalent to SMIS over multiple technique spaces (i.e., marginal MIS (MMIS)), and propose a general purpose path sampling framework for path reuse.

2.2 Paths to Segments

The path integral formulation (1) only considers complete and connected paths, thus proper handling of extra vertices in disconnected (sub)paths needs some extensions in its formulation. Georgiev et al. [2012] showed the extra vertices encountered in photon density estimation can be interpreted as random variables in a rejection sampling process. West et al. [2020, 2022] showed the extra vertices in path-space filtering can be interpreted as random variables that parameterize path sampling techniques as sampling from a marginal distribution. While these formulations are sound, they both consider converting disconnected sub-paths into corresponding complete and connected paths to make them compatible with the original path integral formulation. In other words, extra vertices are essentially *removed* in its formulation so that one can still use the path integral formulation.

In contrast, the extended path integral formulation [Hachisuka et al. 2012] extends the dimensionality of the path integral to explicitly capture paths constructed from disconnected sub-paths. In its most general form, paths are effectively a *sequence of (disconnected) segments* rather than vertices. Each segment is a pair of vertices, and light is transported down the potentially disconnected endpoints of adjacent segments by convolution over a kernel function. This formulation naturally supports disconnected sub-paths as its intrinsic entities without removing any vertices. Moreover, it introduces the theoretical potential to construct paths using path *segments* as the fundamental primitives, rather than *vertices*.

Despite its formulation, Hachisuka et al. [2012], or any of its follow up work, did not explicitly explore this concept of segment sampling and segment path construction. We revisit this formulation from the perspective of sampling segments *as a primitive* and highlight the significant practical potential of segment sampling.

3 SEGMENT-BASED LIGHT TRANSPORT

Our goal is to redefine light transport to use segments as *sampling primitives*. While the concept of segments are there, the extended path integral formulation [Hachisuka et al. 2012] still defines the space of vertices and integrates over vertices. To this end, we propose a *segment path integral* that considers segments as the basic unit of integration for the first time. As an interesting by-product of a segment-based formulation, the kernel and throughput terms now operate over segments, allowing for alternate interpretations of how light is transported down disconnected segments.

3.1 Formulation

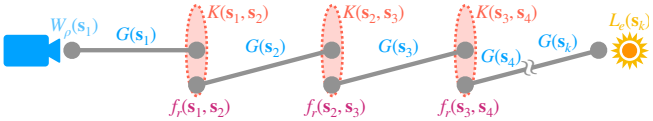
Inspired by the extended path integral formulation [Hachisuka et al. 2012], we propose a segment path integral formulation that considers segments, and segment paths, as the basic unit of integration,

$$J_\rho = \int_{\mathcal{S}} g_\rho(\bar{\mathrm{s}})\,\mathrm{d}\bar{\mathrm{s}} = \sum_{k=1}^{\infty} \underbrace{\int_{\mathcal{S}_k} g_\rho(\bar{\mathrm{s}})\,\mathrm{d}\bar{\mathrm{s}}}_{=J_{\rho k}}, \tag{3}$$

where ρ is a pixel, $\mathcal{S}_k = \mathcal{M}^{2k}$ represents the set of segments paths $\bar{\mathrm{s}}$ with k segments s, defined as $\bar{\mathrm{s}} = \{\mathrm{s}_1 \dots \mathrm{s}_k\}$. Each segment s_i consists of two vertices $\mathrm{s}_i = (\mathrm{x}_i, \mathrm{y}_i)$. The contribution function $g_\rho(\bar{\mathrm{s}})$ for a segment path $\bar{\mathrm{s}}$,

$$g_\rho(\bar{\mathrm{s}}) = W_\rho(\mathrm{s}_1) \prod_{i=1}^{k} G(\mathrm{s}_i) \prod_{i=1}^{k-1} K(\mathrm{s}_i, \mathrm{s}_{i+1}) f_r(\mathrm{s}_i, \mathrm{s}_{i+1})\, L_e(\mathrm{s}_k)\,, \tag{4}$$

is composed of a sensor responsivity term $W_\rho(\mathrm{s}) = W_\rho(x_1, y_1)$ for the first segment, a geometry term $G(\mathrm{s}_i)$ for each segment, a kernel function $K(\mathrm{s}_i, \mathrm{s}_{i+1})$, and an emission term $L_e(\mathrm{s}_k) = L_e(x_k, y_k)$ along the final segment s_k. The geometry term expands to $G(\mathrm{s}_i) = \frac{|\mathrm{n}_{\mathrm{x}_i} \cdot \vec{\mathrm{s}_i}||\mathrm{n}_{\mathrm{y}_i} \cdot \vec{\mathrm{s}_i}|}{|\mathrm{x}_i - \mathrm{y}_i|^2} V(\mathrm{x}_i, \mathrm{y}_i)$, where $\mathrm{n}_{\mathrm{x}_i}$ is the surface normal at x_i, $\vec{\mathrm{s}_i}$ is the direction along s_i from x_i towards y_i, and $V(\mathrm{x}_i, \mathrm{y}_i)$ is the visibility indicator function between x_i and y_i.



While the sensor responsivity term $W_\rho(\mathrm{s}_1)$, geometry term $G(\mathrm{s}_i)$, and emission term $L_e(\mathrm{s}_k)$ are effectively unchanged from the extended path integral formulation, the kernel function $K(\mathrm{s}_i, \mathrm{s}_{i+1})$ and throughput $f_r(\mathrm{s}_i, \mathrm{s}_{i+1})$ terms now operate over pairs of segments, rather than vertices, as follows.

Kernel function. In contrast to the formulation by Hachisuka et al. [2012] where the kernel function $K(\mathrm{y}, \mathrm{w})$ operates over a pair of endpoint vertices, the kernel function in our formulation $K(\mathrm{s}_i, \mathrm{s}_{i+1})$ operates over a pair of segments, s_i and s_{i+1}. This formulation, for example, now allows the kernel function to depend on the *directions* onto y and w, which is important for formulating path filtering. The kernel function can be almost arbitrary given it satisfies the constraint,

$$\int_{\mathcal{M}} K(\mathrm{s}_i, \mathrm{s}_{i+1})\,\mathrm{d}\mathrm{x}_{i+1} = 1\,, \tag{5}$$

such that convolution over the kernel function K preserves the total system energy. A practical choice, demonstrated in the results of this paper, is the uniform kernel over a finite region $\mathcal{K}(\mathrm{y}_i)$ centered at the vertex y_i, $K(\mathrm{s}_i, \mathrm{s}_{i+1}) = (\int_{\mathcal{K}(y_i)} \mathrm{d}\mathrm{x}_{i+1})^{-1}$. When the kernel function is a Dirac delta, our segment-based formulation reduces to the vertex-based path integral formulation. Similarly to advancements in photon density estimation, exploration of alternative kernels for segment-based light transport is an interesting avenue of future work.

Throughput function. Another notable difference from existing formulations is the throughput function $f_r(\mathrm{s}_i, \mathrm{s}_{i+1})$, which introduces an additional vertex not present in the common vertex triplet

form. To be consistent with the path integral formulation in the limit of the kernel becoming a Dirac delta [Hachisuka et al. 2012], we define the segment-based throughput function $f_r(\mathrm{s}_i, \mathrm{s}_{i+1})$ to reduce to the triplet form $f_r(\mathrm{x}_i, \mathrm{y}_i, \mathrm{y}_{i+1})$ when s_i and s_{i+1} share one of the endpoints at y_i. While there are many such possible definitions, we propose a *shift invariant* throughput where we translate the segments such that x_{i+1} lies on y_i,

$$f_r(\mathrm{s}_i, \mathrm{s}_{i+1}) = f_r(\mathrm{x}_i, \mathrm{y}_i, \mathrm{y}_{i+1} - (\mathrm{x}_{i+1} - \mathrm{y}_i)) \,. \tag{6}$$

The shift-invariant form aligns closely with the actual implementation of merging-based methods (e.g., path filtering [Deng et al. 2021; West et al. 2022]), where incoming radiance along each segment is reused without correcting the direction. We use the shift-invariant form in our results and leave the exploration of other definitions for future work.

Recursive Segment Path Integral. Given those redefinitions of K and f_r in terms of segments, one can rewrite the segment path integral into a recursive form,

$$L(\mathrm{s}) = L_e(\mathrm{s}) + \int_{\mathcal{S}_1} K(\mathrm{s}, \mathrm{s}') f_r(\mathrm{s}, \mathrm{s}') G(\mathrm{s}') L(\mathrm{s}') \, \mathrm{ds}' \,, \tag{7}$$

where $L(\mathrm{s})$ is the radiance transported down the segment s, and pixel values J_ρ are computed by the pixel-forming equation,

$$J_\rho = \int_{\mathcal{S}_1} W_\rho(\mathrm{s}) G(\mathrm{s}) L(\mathrm{s}) \, \mathrm{ds} \,. \tag{8}$$

Rewriting as a recursive integral formulation was not previously possible in the extended path integral formulation because not all the terms were written as functions of segments. One can think of this recursive form as a segment-based extension of the rendering equation [Kajiya 1986]. The recursive form, though equivalent in solution to the segment path integral (3), provides a foundation for implementing propagation-based algorithms that locally compute the throughput of individual segments as we will explain later.

3.2 Basic Monte Carlo Estimation

Similarly to the path integral (1), we can approximate the solution to the segment integral using MC integration,

$$\langle J_{\rho k} \rangle = \frac{g_\rho(\bar{\mathrm{s}})}{p(\bar{\mathrm{s}})} \,, \tag{9}$$

where segment path samples $\bar{\mathrm{s}}$ are distributed according to the PDF $p(\bar{\mathrm{s}})$ (i.e., a joint PDF of sampling all the endpoints). We can similarly have a MC estimator for the pixel value (8),

$$\langle J_\rho \rangle = \frac{W_\rho(\mathrm{s}) G(\mathrm{s}) \langle L(\mathrm{s}) \rangle}{p(\mathrm{s})} \,, \tag{10}$$

where the estimate $\langle L(\mathrm{s}) \rangle$ is a nested MC estimator for Eq. (7):

$$\langle L(\mathrm{s}) \rangle = L_e(\mathrm{s}) + \frac{K(\mathrm{s}, \mathrm{s}') f_r(\mathrm{s}, \mathrm{s}') G(\mathrm{s}') \langle L(\mathrm{s}') \rangle}{p(\mathrm{s}'|\mathrm{s})} \,. \tag{11}$$

In later sections we will explore more advanced estimation strategies that can account for a wide range of sampling scenarios.

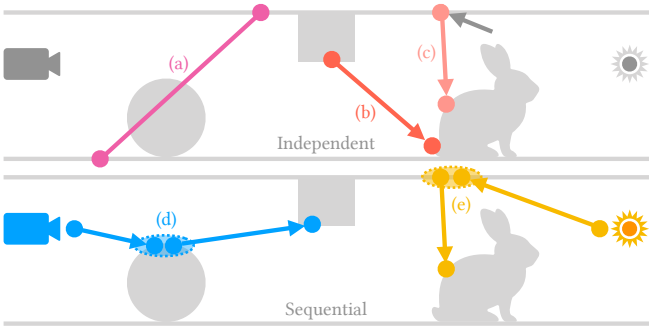


Fig. 2. While theoretically interesting, the independent segment samplers: uniform (a), ray-casted (b), BSDF importance-sampled (c) are not good importance samplers for the contribution function (4). The sequential segment samplers starting from the camera (d) and light sources (e) greatly improve importance sampling of the segment path contribution (4).

4 SEGMENT SAMPLERS AND ESTIMATORS

As in the vertex-based counterparts [Georgiev et al. 2012; Hachisuka et al. 2012], one can look at rendering methods based on segments from the perspective of: 1) sampling a set of segments, and then 2) constructing estimators based on the sampled segments. Variations in these two steps provide a general process for exploring different rendering methods based on segments.

4.1 Samplers

Let us first look at several techniques for sampling segments. Some of these sampling techniques are mostly of theoretical interest — often lacking good importance sampling properties. These sampling techniques will provide a baseline for more practical sampling techniques introduced later in the section.

Uniform segment sampler. The simplest way to sample a segment s is to sample its two vertices (x, y) independently and uniformly across scene geometry, such that,

$$p(\mathrm{s}) = p(\mathrm{x}) p(\mathrm{y}) = \frac{1}{|\mathcal{M}|^2} \,, \tag{12}$$

where $|\mathcal{M}|$ is the area of scene surfaces (see Fig. 2a). While this sampling technique is straightforward, it importance samples none of the terms of the contribution function. Note that uniform sampling of vertices is also possible in the path integral formulation and equally impractical because of its lack of importance sampling.

Ray tracing segment sampler. A simple improvement on the uniform segment sampler is importance sampling the geometry term $G(\mathrm{s})$. The first vertex x is sampled uniformly within the scene, and the second vertex y is sampled by first sampling a direction using cosine-weighted hemisphere sampling at x, and then projecting the directional sample into area domain using ray casting, such that,

$$p(\mathrm{s}) = p(\mathrm{x}) p(\mathrm{y}|\mathrm{x}) = \frac{1}{|\mathcal{M}|} \cdot \frac{|\mathrm{n}_{\mathrm{x}_i} \cdot \vec{\mathrm{s}}_i| |\mathrm{n}_{\mathrm{y}_i} \cdot \vec{\mathrm{s}}|}{\pi |\mathrm{x} - \mathrm{y}|^2} = \frac{G(\mathrm{s})}{\pi |\mathcal{M}|} \,, \tag{13}$$

exactly importance samples the geometry term $G(\mathrm{s})$ of the segment path contribution function (4) (see Fig. 2b).

We can further improve the above strategy by additionally considering the segment throughput function $f_r(\mathrm{s}, \mathrm{s}')$. After sampling

the first vertex x, we then sample an auxiliary incoming direction $\vec{s}'$, BSDF importance-sample an outgoing direction $\vec{s}$ conditionally on $\vec{s}'$, and project the directional sample $\vec{s}$ using ray casting to find the second vertex y, such that,

$$p(\mathrm{s}|\vec{s}') = p(\mathrm{x})p(\mathrm{y}|\mathrm{x},\vec{s}') = \frac{1}{|\mathcal{M}|}\cdot\frac{p(\vec{s}|\mathrm{x},\vec{s}')|\mathrm{n}_{\mathrm{y}_i}\cdot\vec{s}|}{|\mathrm{x}-\mathrm{y}|^2}, \tag{14}$$

where $p(\vec{s}|\mathrm{x},\vec{s}')$ is a directional PDF proportional to the BSDF term at x (see Fig. 2(c)). Note that this PDF is still conditional on $\vec{s}'$. For this PDF to make sense, there has to be a preceding segment to a sampled segment s with the (approximately) matching direction $\vec{s}'$ which may or may not exist. This exemplifies one of the key issues with the naive segment samplers thus far — segments are sampled independently.

Sequential segment sampler. Independent sampling of segments is ineffective at importance sampling terms of the contribution function (4) that usually consider *pairs* of segments. We thus consider sampling a *sequence of segments*, rather than sampling them independently (see Fig. 2d,e). This sequential segment sampler considers the terms of the contribution function $g(\bar{s})$ one-by-one, forming a natural counterpart to sequential vertex samplers in vertex-based formulations (e.g., path tracing and light tracing).

When sampling a segment sequence, a good choice for importance sampling the first segment s_1 is according to either sensor responsivity $p_W(\mathrm{s}_1) \sim W_\rho(\mathrm{s}_1)G(\mathrm{s}_1)$ or light emission $p_L(\mathrm{s}_1) \sim L_e(\mathrm{s}_1)G(\mathrm{s}_1)$. Each subsequent segment s_{i+1} can then be conditionally sampled on the previous segment s_i,

$$\begin{aligned} p(\mathrm{s}_{i+1}|\mathrm{s}_i) &= p(\mathrm{x}_{i+1}|\mathrm{s}_i)p(\mathrm{y}_{i+1}|\mathrm{x}_{i+1},\mathrm{s}_i) \\ &= p_K(\mathrm{x}_{i+1}|\mathrm{s}_i)\cdot\frac{p_\omega(\vec{\mathrm{s}_{i+1}}|\mathrm{x}_{i+1},\mathrm{s}_i)|\mathrm{n}_{\mathrm{y}_{i+1}}\cdot\vec{\mathrm{s}_{i+1}}|}{|\mathrm{x}_{i+1}-\mathrm{y}_{i+1}|^2}, \end{aligned} \tag{15}$$

where $p_K(\mathrm{x}_{i+1}|\mathrm{s}_i) \sim K(\mathrm{s}_i,\mathrm{s}_{i+1})$ importance samples the vertex x_{i+1} according to the kernel at s_i, $\vec{\mathrm{s}_{i+1}}$ is the direction down the segment s_{i+1} from x_{i+1} to y_{i+1}, and $p_\omega(\vec{\mathrm{s}_{i+1}}|\mathrm{x}_{i+1},\mathrm{s}_i) \sim f_r(\mathrm{s}_i,\mathrm{s}_{i+1})$ importance samples $\vec{\mathrm{s}_{i+1}}$ according to the BSDF term at y_i.

Just like *virtual perturbation* in the UPS algorithm [Hachisuka et al. 2012], we can approximately sample $p_K(\mathrm{x}_{i+1}|\mathrm{s}_i)$ by duplicating $\mathrm{y}_i = \mathrm{x}_{i+1}$ while keeping its PDF, avoiding explicit sampling of x_{i+1} in practice. This approximation with $\mathrm{y}_i = \mathrm{x}_{i+1}$ effectively turns this sequential segment sampler into an equivalent sequential vertex samplers, but with duplicated vertices and an extended definition of the PDF $p_K(\mathrm{x}_{i+1}|\mathrm{s}_i)$.

Given segment sequences starting from the camera and light sources, a natural extension is to introduce a segment equivalent of connections as in vertex-based samplers [Veach and Guibas 1994] to combine different segment sequences. We refer to such segments as bridge segments to distinguish them from the classical definition of deterministic connections. While theoretically interesting, preliminary results (see Fig. 7) show no tangible benefit for bridge segments in practice. We include additional details about bridge segments in the supplemental document.

4.2 Estimators

Given these segment sampling techniques, we now look at how we can use them to construct practical estimators. Similarly to vertex-based methods, we can *reuse* sampled segments to define multiple estimators. However, unlike vertices that are essentially tied to a sampling direction and path position (e.g., second vertex on a light path), segments are agnostic to how they were sampled. This extra flexibility vastly expands the number of possible ways to construct a complete segment path given a set of sampled segments.

We explore several baseline estimation strategies, as well as a practical recursive estimator based on marginal multiple importance sampling (MMIS) that considers an *exponential* number of techniques given sampled segments, as opposed to a linear number of techniques in vertex-based rendering [Georgiev et al. 2012; Hachisuka et al. 2012]. We note that our goal here is not to exhaustively list all possible estimators based on segments, and further study of even more efficient estimators is left for future work.

Sequential. A simple and straightforward estimation strategy is to use a Monte Carlo estimator (9) with a segment path sampling technique comprised of a sequence of k segment sampling techniques (15). From this sequence of sampling techniques we draw a single segment from each, and construct a single segment path sampled from the segments in the order they were sampled. The PDF of a segment path sampled in this way is the joint PDF as in

$$\langle J_{\rho k}\rangle = \frac{g_\rho(\bar{s})}{p(\bar{s})}, \quad p(\bar{s}) = p(\mathrm{s}_1,\ldots,\mathrm{s}_k), \tag{16}$$

and any auxiliary variables that conditioned sampling, but are not a part of the segment path sample (e.g. incident direction in triplet segment sampling), now parameterize the segment path sampling technique. Constructing a single segment path sample $\bar{s}$ from the set of sampled segments $\mathrm{s}_1,\ldots,\mathrm{s}_k$ still under-utilizes the segment path sampling potential.

Multiple sampling techniques. In the spirit of bidirectional path tracing (BDPT), we can reuse subsequences of sampled segment paths to construct new segment paths. Each different way of constructing a segment path sample is a different segment path sampling technique, and we can combine them into an estimator using MIS [Veach 1997]. For T segment path sampling techniques $p_i(\bar{s})$, with n_i segment path samples $\bar{s}_i$ from each, the corresponding MIS estimator is

$$\langle J_{\rho k}\rangle_{\mathrm{MIS}} = \sum_{i=1}^{T}\sum_{j=1}^{n_i}\frac{w_i(\bar{s}_{i,j})g_\rho(\bar{s}_{i,j})}{n_i p_i(\bar{s}_{i,j})}, \tag{17}$$

where for a segment path $\bar{s}$ the weight function $w_i(\bar{s})$ sums to 1 over the T segment path sampling techniques, $\sum_{i=1}^{T} w_i(\bar{s}) = 1$. While this increases the number of segment path samples for the same segment sampling cost, we can further increase segment reuse by considering a recursive estimator.

Recursive estimation. An alternative way to construct segment paths from sampled segments is recursively — enabled by our recursive formulation (7). Starting at a known contributive segment for the pixel-forming equation (8) (e.g., a segment that connects to the camera), we can recursively determine the set of continuation

segments within the non-zero support of the kernel function of the recursive term (7). At each recursive expansion, the set of continuation segment samples may have been sampled by different segment sampling techniques (e.g., sequential sampling conditioned on different previous segments).

To handle these situations West et al. [2022] introduce marginal multiple importance sampling (MMIS), which supports combining both unconditional sampling techniques and sampling techniques conditioned on auxiliary random variables $\bar{\mathrm{t}}$. We can apply their MMIS theory to derive a general purpose estimator for the recursive form of the segment path integral (see supplemental document),

$$\langle L(\mathrm{s})\rangle_{\mathrm{BH}} = L_e(\mathrm{s}) + \sum_{i=1}^{T}\sum_{j=1}^{n_i} \frac{K(\mathrm{s},\mathrm{s}'_{i,j}) f_r(\mathrm{s},\mathrm{s}'_{i,j}) G(\mathrm{s}'_{i,j}) \langle L(\mathrm{s}'_{i,j})\rangle_{\mathrm{BH}}}{\sum_{i'=1}^{T}\sum_{j'=1}^{n_{i'}} p_{i'}(\mathrm{s}'_{i,j}|\mathrm{t}_{i',j'})}, \tag{18}$$

where we use the approximated balance heuristic weights, and, similarly, an MMIS estimator for the pixel forming equation (8),

$$\langle J_\rho\rangle_{\mathrm{BH}} = \sum_{i=1}^{T}\sum_{j=1}^{n_i} \frac{W_\rho(\mathrm{s}_{i,j}) G(\mathrm{s}_{i,j}) \langle L(\mathrm{s}_{i,j})\rangle_{\mathrm{BH}}}{\sum_{i'=1}^{T}\sum_{j'=1}^{n_{i'}} p_{i'}(\mathrm{s}_{i,j}|\mathrm{t}_{i',j'})}. \tag{19}$$

In practice, while the nested estimators of Eq. (19) can be computed recursively as we expand outwards from the camera, the resulting computational complexity is exponential in the number of segments — essentially intractable. Similarly to Deng et al. [2021]'s implementation of multi-vertex path filtering, we can use an iterative propagation-based algorithm to efficiently compute the recursive MMIS estimator (18), reducing the computational complexity from exponential to linear. Further details on propagation are provided in the supplemental document.

Marginal segment path sampling. Not all sets of segment path sampling techniques correspond well to the recursive MMIS estimator (18). For these sets of sampling techniques we can use an MMIS estimator for the segment path integral (3) that operates over complete segment paths $\bar{\mathrm{s}}$,

$$\langle J_{\rho k}\rangle_{\mathrm{BH}} = \sum_{i=1}^{\bar{T}}\sum_{j=1}^{n_i} \frac{g_\rho(\bar{\mathrm{s}}_{i,j})}{\sum_{i'=1}^{\bar{T}}\sum_{j'=1}^{n_{i'}} p_{i'}(\bar{\mathrm{s}}_{i,j}|\bar{\mathrm{t}}_{i',j'})}. \tag{20}$$

where we use the approximate balance heuristic weights (see the supplemental document for a complete derivation). Much like the marginal path sampling (MPS) framework of West et al. [2022], this general-purpose estimator provides a flexible framework for implementing various segment-based light transport algorithms.

5 PRACTICAL ALGORITHMS

Based on samplers and estimators in the previous section, both existing and novel light transport methods can be devised by combining segment samplers with different estimation strategies.

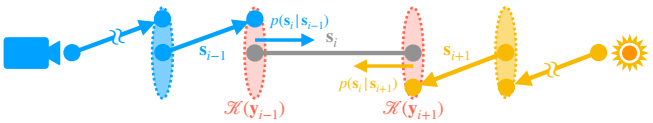


Fig. 3. In the proposed bidirectional filtering, the underlying distribution of segments comes from sequential sampling from the camera or light sources. Segments can then only have been sampled sequentially within the support of the kernels, $\mathcal{K}(\mathrm{y}_{i-1})$ and $\mathcal{K}(\mathrm{y}_{i+1})$, at each endpoint. The MMIS weight of the segment s_i only needs to consider conditioning on any segment s_{i-1} and s_{i+1} within kernel support, greatly bounding computational cost. Note that the segment s_i can be sampled in three different ways, as a continuation of either a light or camera segment path or by directly sampling two vertices within the kernel supports.

Segment-based path tracing. A straightforward example of segment-based light transport methods combines a sequential segment sampler (15) with the sequential estimation strategy (16), effectively creating segment-based equivalents of path tracing and light tracing. To overcome the limitations of unidirectional approaches, two sequential segment samplers can be combined with a bridge segment sampler using a MIS estimator (17), yielding a segment-based version of bidirectional path tracing (BDPT). While theoretically intriguing, these simple rendering methods do not fully leverage the advantages of a segment-based formulation, offering no tangible benefit over their vertex-based counterparts.

Segment-based multi-vertex path filtering. We can implement a segment-based variant of undirectional path filtering methods [Deng et al. 2021; West et al. 2022] by combining a recursive MMIS estimator to sets of segments sampled from different segment sampling techniques. An underlying distribution of segments that come from a sequential segment sampler (15) starting at the camera corresponds to multi-vertex filtering (MVF) [Deng et al. 2021]. Similarly, segments sampled sequentially starting at light sources corresponds to photon filtering (PhF) [West et al. 2022].

Bidirectional path filtering. The vertex-based formulation underlying existing multi-vertex path filtering methods makes it challenging to implement a bidirectional filtering method. In contrast, our segment-based formulation and estimators naturally handle segment samples from almost arbitrary distributions. By applying our recursive MMIS estimator (18) to a set of segments drawn from sequential samplers starting at both the camera and light sources we can implement a bidirectional filtering method (see Fig. 3). Interestingly, the resulting segment path sampling techniques of this bidirectional filtering method is a superset of those in bidirectional photon mapping [Vorba 2011]. As we demonstrate in the results, bidirectional path filtering is robust — inheriting the importance sampling of prior unidirectional filtering methods, but with none of their failure cases — and significantly outperforms competing methods in scenes with complex light transport. We include a full algorithm and implementation details for our bidirectional path filtering method in the supplemental document.

6 RESULTS AND DISCUSSIONS

We implemented our method and prior work on a CPU-based RGB renderer with an AMD Ryzen 5950X CPU and 32 GB of memory. Images were rendered with a resolution of 1200×800, with segment path depth limited to 8. Results were generated progressively, with one segment path per pixel per iteration (960K camera segment paths per iteration). The ratio of light to camera segment paths, denoted by β, was set to 0.25 (see Fig. 6). The initial support of the kernel function in all methods was set to 4 times the average pixel footprint, and we progressively shrink the kernel support with $\alpha = 0.67$ following Knaus and Zwicker [2011]. The reference implementation is publicly available on GitHub. For further implementation details see the supplemental document.

6.1 Bidirectional Filtering

Fig. 4 compares our bidirectional filtering method to its unidirectional counterparts, including multi-vertex filtering (MVF) [Deng et al. 2021] and photon filtering (PhF) [West et al. 2022], as well as two baseline methods bidirectional photon mapping (BPM) [Vorba 2011] and unified path sampling (UPS) [Hachisuka et al. 2012]. MVF extends path tracing (PT) by incorporating efficient sample reuse over generated path samples. As it consistently outperformed PT in our tests, we chose not to include PT results. A similar rationale applies to omitting a separate comparison of BDPT versus UPS.

The Bathroom scene has a single spherical light source, posing a challenge due to its glossy materials and the occlusion between objects. The Saloon Bar scene contains predominantly diffuse materials but is lit by six directional ceiling lights, a challenging scenario for methods relying solely on path tracing for sample generation (e.g., MVF). The Kitchen scene is primarily illuminated by a large area light source, making path tracing-based methods like MVF effective, and conversely, light tracing-based methods like PhF ineffective. The Disco Box scene features two nearly specular disco balls reflecting light from two spot lights, resulting in chromatic caustics that can only be handled robustly by merging light paths (e.g. UPS, BPM, and PhF). For this challenging caustics scenario, we increase the value of β to 1.0 for all methods. The Warehouse scene provides an ideal case for bidirectional filtering, where it is lit predominantly by indirect light from several small area lights with surrounding enclosures and contains complex occlusion.

In the Bathroom, Saloon Bar, and Warehouse scenes, our method outperforms prior work both visually and numerically, achieving smooth global illumination without any visible artifacts. In the Kitchen and Disco Ball scenes, our method performs no worse than the unidirectional filtering methods suitable for these specific scenarios (e.g., MVF for the Kitchen scene and PhF for the Disco Box scene). While the unidirectional path sampling of MVF and PhF shows strong failure cases, the bidirectional path sampling of BPM and UPS demonstrates consistent performance and overall robustness. BPM and UPS, however, fail to fully exploit the reuse potential of path samples, showing slower convergence than our bidirectional filtering method. Overall our bidirectional path filtering robustly estimates the light transport and achieves state-of-the-art image quality across all scenes.

6.2 Filtering Efficiency and Kernel Size

In Fig. 5, we compare the performance of our method with varying filtering kernel sizes. We use our primary ray footprint heuristic (4 times the average pixel footprint) as the baseline 1.0x kernel size. Similar to previous methods, larger filtering kernels show faster convergence but suffer from excessive bias, while smaller filtering kernels trade bias for increased variance in the form of noise. By progressively reducing the kernel size we can achieve a balance between bias and variance reduction.

6.3 Path Reuse Efficiency

In Fig. 6, we compare the converge rates of BPM and our bidirectional filtering for varying ratios β of path samples. Our bidirectional filtering uses a super set of the sampling techniques considered in BPM. These additional sampling techniques correspond to the increased granularity is sample reuse, allowing our method to produce lower error estimates for the same sampling cost. However, due to quadratic overhead of filtering, performance for our method begins to decline as the number of path samples grows large.

6.4 Adding Bridge Segments

In Fig. 7, we present preliminary experimental results that include bridge segments (explained in detail in the supplemental) in bidirectional path filtering. By including bridge segments into the set of segments we use in propagation, we can effectively capture a super set of the path sampling techniques used in UPS [Hachisuka et al. 2012]. In the equal-sample comparison, the bidirectional filtering variant with bridge segments shows faster convergence than the one without. However, as shown in the equal-time results, the error reduction of including bridge segments is negated by the increased computational overhead. This highlights a performance bottleneck of our method — filtering is quadratic $O(n^2)$ in the number of segments in a cluster, so merely adding more segments can be detrimental. Improving the efficiency and practicality of bridge segments is left for future work.

7 LIMITATIONS AND FUTURE WORK

Segment sampling. Our work on segment sampling introduces a few basic strategies, but only begins to address the broader potential of segment sampling. A key open question is identifying the ideal target distribution for importance sampling segments to estimate the segment path integral. Assuming the target distribution is known or can be approximated, one promising avenue would be developing a segment sampler guided by a learned distribution, inspired by ideas from path guiding in path sampling [Müller et al. 2017]. Another potential approach involves exploring MCMC-based segment sampling. Given that our segment framework is compatible with path reuse, this could potentially enable the unification of path reuse with MCMC sampling techniques. Segment sampling can also be potentially much more suitable for quasi-Monte Carlo (QMC) integration since segments are low dimensional properties than paths where QMC integration tends to perform very well. Our segment-based formulation subsumes vertex-based formulation as a special case of having Dirac-delta kernels

(as in the UPS algorithm [Hachisuka et al. 2012]), so further combinations of segments and vertices samplers is another interesting avenue to explore.

In our current implementation, bridge segments are sampled around the endpoints of a camera segment and an arbitrary light segment, showing negligible benefit in practice. A natural extension would involve developing a bridge segment sampler inspired by the concept of probabilistic connections [Popov et al. 2015; Nabata et al. 2020], improving the importance sampling of bridge segments. More generally, the overhead of segment sampling is not a bottleneck of our method, leaving the door open for exploration of more advanced segment sampling techniques.

Segment paths construction. In our segment-based formulation, the direction in which a segment is sampled does not need to be preserved when constructing segment paths. This flexibility enables segments to be reused in the opposite direction than how it was sampled. While directionality influences the computation of probability density functions (PDFs), light transport itself can be evaluated in either direction using the segment path contribution function. Our current segment path construction could potentially be generalized to a directionless variant, offering further opportunities for sample reuse and improved efficiency.

Path filtering. Similarly to prior implementations [Deng et al. 2021; West et al. 2022] of path filtering, the computational cost of filtering grows quadratically with cluster size, limiting the feasible size of filtering kernels. Exploring the optimal kernel and cluster sizes, as well as the optimal distribution of camera and light paths, could lead to further improvements in filtering performance. Additionally, the exploration of filtering-aware importance sampling of segments could potentially lead to further filtering efficiency.

Lastly, we have primarily focused on spatial filtering over a single set of segments. One potential direction for future research is the exploration of filtering over additional dimensions (e.g. temporally) as well as across multiple sets of segments, as in the streaming-style resampling importance sampling (RIS) as explored in ReSTIR [Bitterli et al. 2020; Lin et al. 2022].

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable feedback and suggestions. We gratefully acknowledge the creators of the scene resources used in our experiments: the *Bathroom* scene created by Quantum State Studio, the *Kitchen* scene originally created by Jay-Artist and adapted from the PBRT-v4 version by Benedikt Bitterli, as well as the *Saloon Bar* and *Warehouse* scenes from Quixel Megascans. This research was supported by NSERC under grant number RGPIN-2020-03918.

REFERENCES

Nikolaus Binder, Sascha Fricke, and Alexander Keller. 2018. Fast Path Space Filtering by Jittered Spatial Hashing. In *ACM SIGGRAPH 2018 Talks* (Vancouver, British Columbia, Canada) *(SIGGRAPH '18)*. ACM, New York, NY, USA, Article 71, 2 pages. https://doi.org/10.1145/3214745.3214806

Nikolaus Binder, Sascha Fricke, and Alexander Keller. 2019. Massively Parallel Path Space Filtering. *CoRR* abs/1902.05942 (2019). arXiv:1902.05942 http://arxiv.org/abs/1902.05942

Benedikt Bitterli, Chris Wyman, Matt Pharr, Peter Shirley, Aaron Lefohn, and Wojciech Jarosz. 2020. Spatiotemporal reservoir resampling for real-time ray tracing with dynamic direct lighting. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 148–1.

Xi Deng, Miloš Hašan, Nathan Carr, Zexiang Xu, and Steve Marschner. 2021. Path Graphs: Iterative Path Space Filtering. *ACM Trans. Graph.* 40, 6, Article 276 (dec 2021), 15 pages. https://doi.org/10.1145/3478513.3480547

Iliyan Georgiev, Jaroslav Křivánek, Tomáš Davidovič, and Philipp Slusallek. 2012. Light Transport Simulation with Vertex Connection and Merging. 31, 6 (Nov. 2012), 192:1–192:10. https://doi.org/10/gbb6q7

Toshiya Hachisuka and Henrik Wann Jensen. 2009. Stochastic Progressive Photon Mapping. 28, 5 (Dec. 2009), 130:1–130:8. https://doi.org/10/d8xxn3

Toshiya Hachisuka, Shinji Ogaki, and Henrik Wann Jensen. 2008. Progressive Photon Mapping. 27, 5 (Dec. 2008), 130:1–130:8. https://doi.org/10/cn8h39

Toshiya Hachisuka, Jacopo Pantaleoni, and Henrik Wann Jensen. 2012. A Path Space Extension for Robust Light Transport Simulation. 31, 6 (Jan. 2012), 191:1–191:10. https://doi.org/10/gbb6n3

Paul S. Heckbert. 1990. Adaptive Radiosity Textures for Bidirectional Ray Tracing. 24, 4 (Aug. 1990), 145–154. https://doi.org/10/bsxgp4

Henrik Wann Jensen. 1996. *The Photon Map in Global Illumination.* Ph.D. Thesis. Technical University of Denmark.

James T. Kajiya. 1986. The Rendering Equation. 20, 4 (Aug. 1986), 143–150. https://doi.org/10/cvf53j

Anton S. Kaplanyan and Carsten Dachsbacher. 2013. Adaptive Progressive Photon Mapping. 32, 2 (April 2013), 16:1–16:13. https://doi.org/10/gbc2fq

Alexander Keller, Ken Dahm, and Nikolaus Binder. 2014. Path Space Filtering *(SIGGRAPH '14)*. ACM, 68:1–68:1. https://doi.org/10/gfz6mr

Claude Knaus and Matthias Zwicker. 2011. Progressive Photon Mapping: A Probabilistic Approach. 30, 3 (May 2011), 25:1–25:13. https://doi.org/10/bcw2ph

Eric P. Lafortune and Yves D. Willems. 1993. Bi-Directional Path Tracing, H. P. Santo (Ed.), Vol. 93. Alvor, Portugal, 145–153.

Daqi Lin, Markus Kettunen, Benedikt Bitterli, Jacopo Pantaleoni, Cem Yuksel, and Chris Wyman. 2022. Generalized resampled importance sampling: Foundations of restir. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–23.

Zehui Lin, Sheng Li, Xinlu Zeng, Congyi Zhang, Jinzhu Jia, Guoping Wang, and Dinesh Manocha. 2020. CPPM: chi-squared progressive photon mapping. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–12.

Zackary Misso, Benedikt Bitterli, Iliyan Georgiev, and Wojciech Jarosz. 2022. Unbiased and consistent rendering using biased estimators. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 41, 4 (2022). https://doi.org/10.1145/3528223.3530160

Thomas Müller, Markus Gross, and Jan Novák. 2017. Practical Path Guiding for Efficient Light-Transport Simulation. 36, 4 (June 2017), 91–100. https://doi.org/10/gbnvrs

Kosuke Nabata, Kei Iwasaki, and Yoshinori Dobashi. 2020. Resampling-aware weighting functions for bidirectional path tracing using multiple light sub-paths. *ACM Transactions on Graphics (TOG)* 39, 2 (2020), 1–11.

Stefan Popov, Ravi Ramamoorthi, Fredo Durand, and George Drettakis. 2015. Probabilistic Connections for Bidirectional Path Tracing. *Comput. Graph. Forum* 34, 4 (July 2015), 75–86.

Hao Qin, Xin Sun, Qiming Hou, Baining Guo, and Kun Zhou. 2015. Unbiased Photon Gathering for Light Transport Simulation. *ACM Trans. Graph.* 34, 6, Article 208 (Oct. 2015), 14 pages. https://doi.org/10.1145/2816795.2818119

Eric Veach. 1997. *Robust Monte Carlo Methods for Light Transport Simulation.* Ph.D. Thesis. Stanford University, United States – California.

Eric Veach and Leonidas J. Guibas. 1994. Bidirectional Estimators for Light Transport, Georgios Sakas, Peter Shirley, and Stefan Müller (Eds.). 145–167. https://doi.org/10/gfznbh

Eric Veach and Leonidas J. Guibas. 1995. Optimally Combining Sampling Techniques for Monte Carlo Rendering, Vol. 29. 419–428. https://doi.org/10/d7b6n4

Jiri Vorba. 2011. Bidirectional photon mapping. *Proceedings of CESCG* (2011), 25–32.

Rex West, Iliyan Georgiev, Adrien Gruson, and Toshiya Hachisuka. 2020. Continuous Multiple Importance Sampling. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)* 39, 4 (July 2020). https://doi.org/10.1145/3386569.3392436

Rex West, Iliyan Georgiev, and Toshiya Hachisuka. 2022. Marginal Multiple Importance Sampling. In *SIGGRAPH Asia 2022 Conference Papers* (Daegu, Republic of Korea) *(SA '22)*. Association for Computing Machinery, New York, NY, USA, Article 42, 8 pages. https://doi.org/10.1145/3550469.3555388

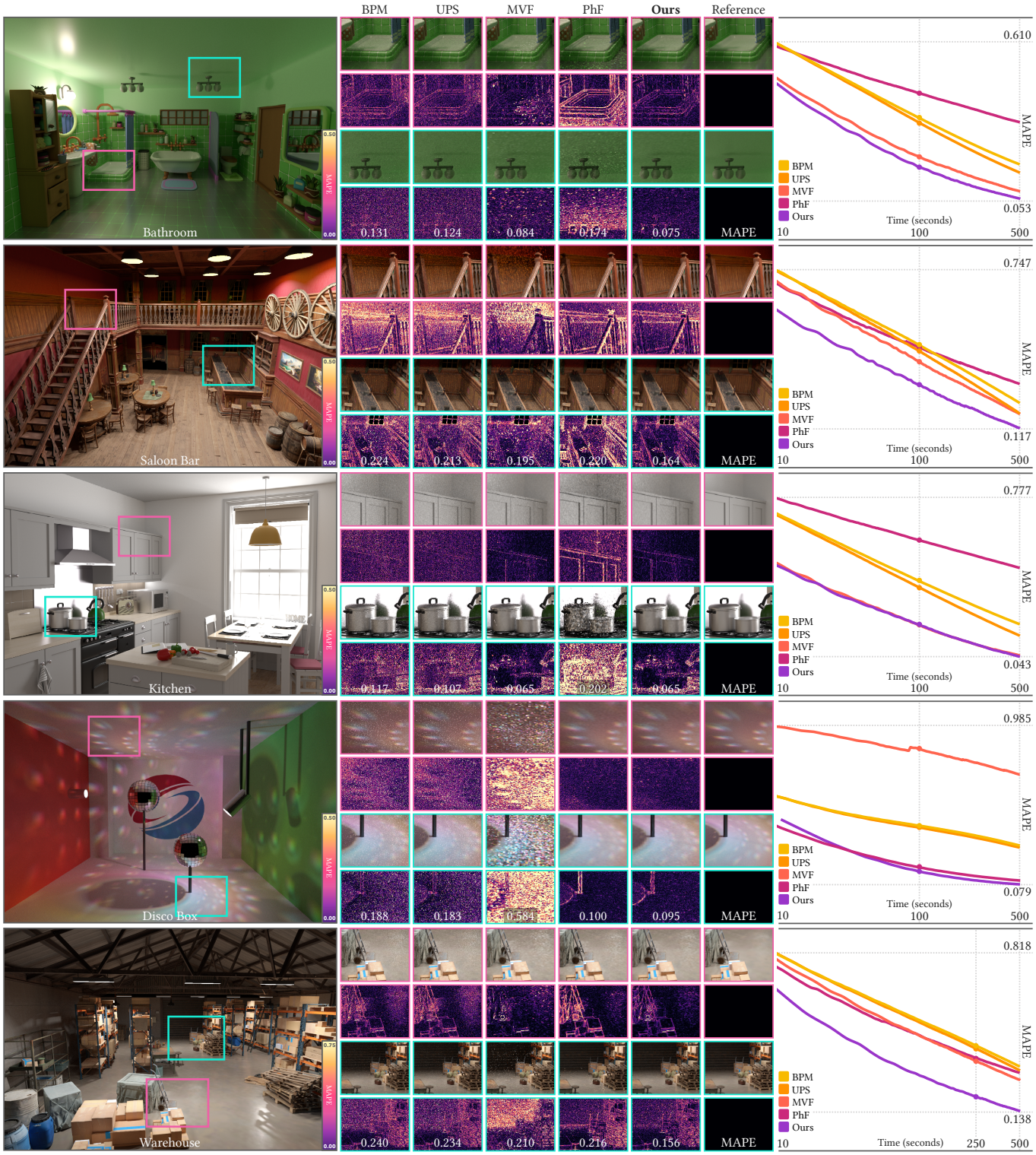


Fig. 4. Equal-time comparison (100s) of our method with unidirectional filtering methods (MVF, PhF) and bidirectional merging methods (BPM, UPS) across five scenes. Our method demonstrates robustness under a wide-range of light transport scenarios, either outperforming competing methods or achieving performance comparable to the best unidirectional filtering variant in their ideal rendering scenario (e.g. Kitchen for MVF and Disco Box for PhF).

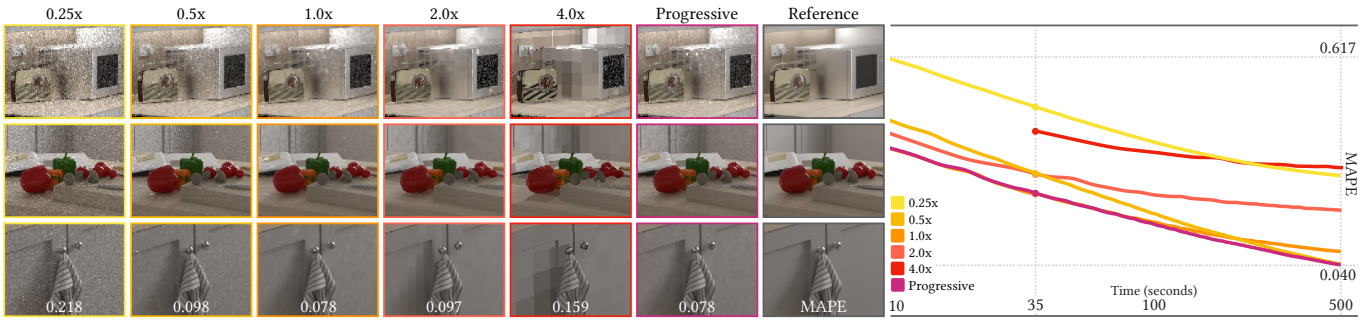


Fig. 5. Equal-time comparison (35s) of our method with varying kernel sizes, ranging from 0.25x to 4.0x. Larger filtering kernels (e.g., 2.0x, 4.0x) achieve faster convergence but suffer from higher bias, while smaller filtering kernels (e.g., 0.25x, 0.5x) produce less biased results but require more time to converge. Additionally, even with smaller kernel sizes, the propagation overhead remains, making our method less efficient with small kernels. We recommend using our progressive variant, which starts at 1.0x and gradually reduces the kernel size, enabling fast initial convergence while progressively reducing bias as more samples accumulate.

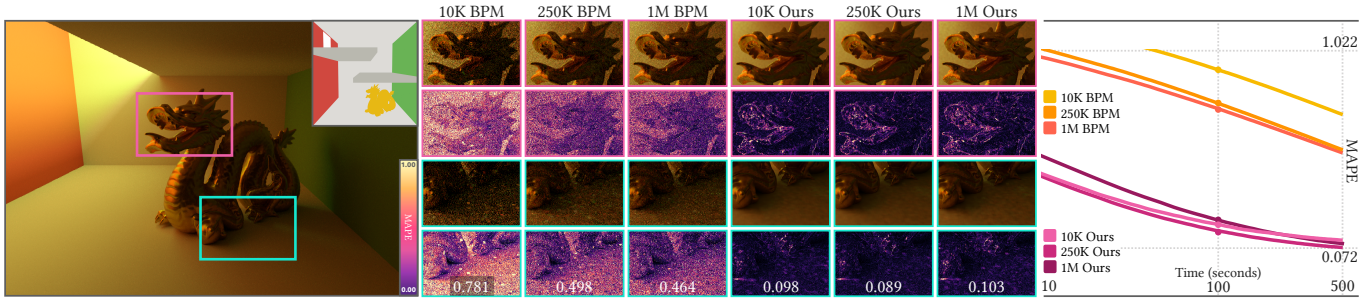


Fig. 6. Equal time comparison (100s) of our method and BPM with different numbers of light segment paths (10k, 250k, and 1M paths). The results highlight the efficiency of our method in path reuse, producing highly-converged results with only 10k light paths, while BPM struggles with noise. This difference in performance is due to our methods ability to more effectively reuse disconnected path samples. However, the performance of our method declines when using 1M light paths due to the quadratic increase in filtering cost.

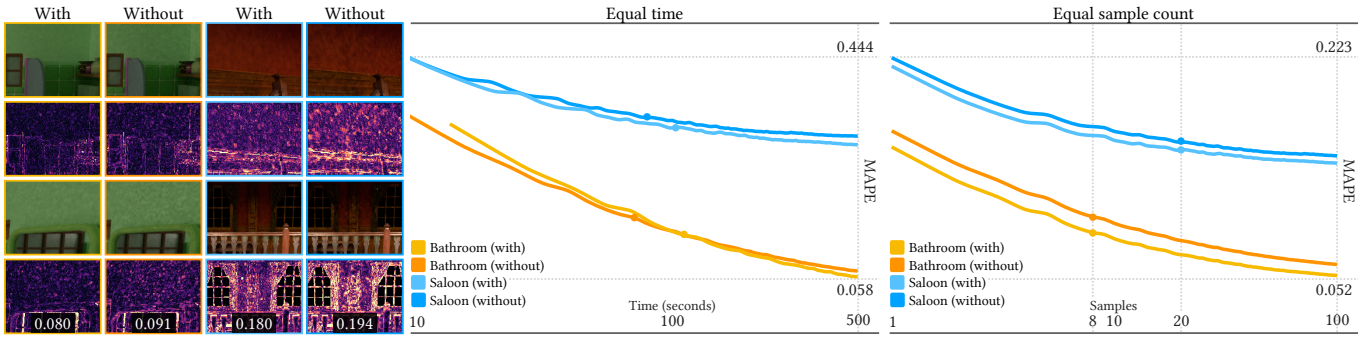


Fig. 7. Comparison of our method with and without bridge segments in the Bathroom and the Saloon Bar scenes using a fixed kernel size. The results highlight the advantages of introducing bridge segments in an equal sample comparison. However, these advantages are less pronounced in an equal time comparison due to the additional filtering overhead introduced by bridge segments.