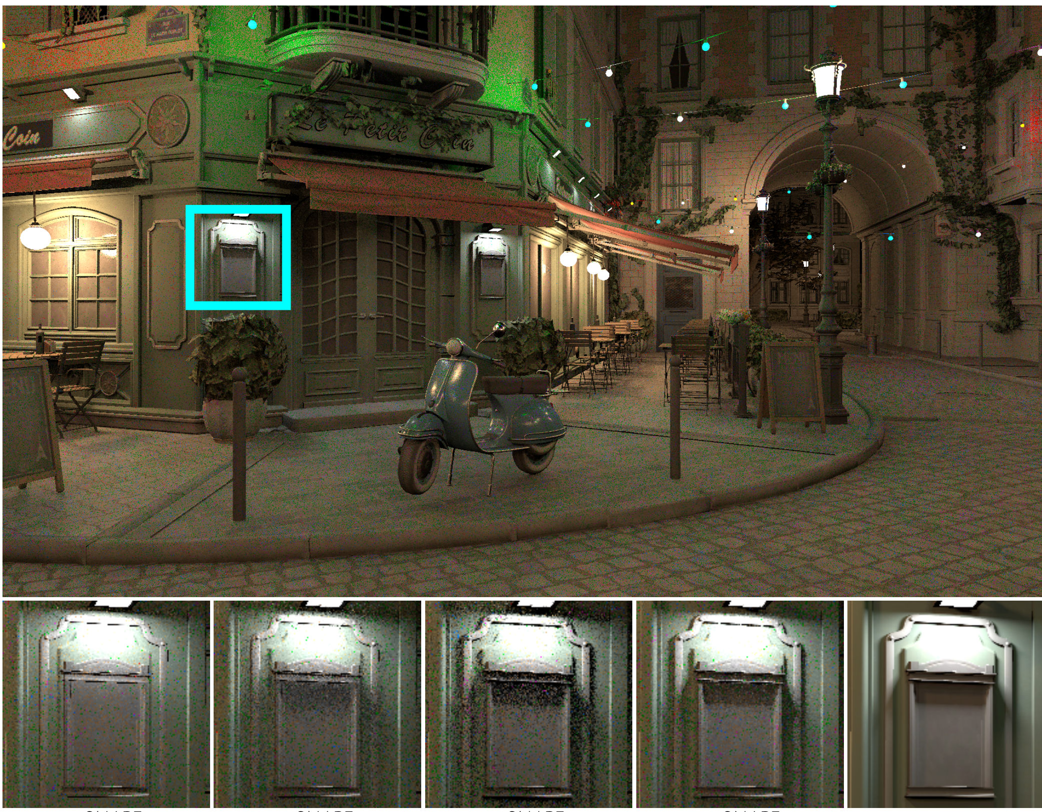
Efficient Spatial Resampling Using the PDF Similarity

YUSUKE TOKUYOSHI, Advanced Micro Devices, Inc., Japan



6.36 ms, SMAPE: 10.8% | 7.95 ms, SMAPE: 11.5% | 8.98 ms, SMAPE: 12.6% | 8.05 ms, SMAPE: 9.2%

(a) Previous, 1 rpp (b) Previous, 1.25 rpp (c) Previous, 2 rpp (d) Ours, 1.25 rpp (e) Reference

Fig. 1. Top image: A biased ReSTIR variant [Wyman and Panteleev 2021] using 1.25 shadow rays per pixel (rpp) with our rejection method for the Bistro scene (2.8 M triangles, 1 M VPLs generated on 20.6 k triangle lights, 1920×1080 pixels, AMD Radeon™ RX 6900 XT GPU). Bottom images: Close-ups of rendering results using previous rejection method with different ray counts (a–c), our rejection method (d), and the reference image (e). The biased ReSTIR variant using two rpp produces a darkening bias for shadows (c). When using less than two rays, this ReSTIR loses hard shadows (a, b) because of spatial resampling across shadow edges. With our rejection method for spatial resampling (d), we render higher-quality shadows using a smaller number of rays than the existing rejection method.

In real-time rendering, *spatiotemporal reservoir resampling* (ReSTIR) is a powerful technique to increase the number of candidate samples for *resampled importance sampling*. However, reusing spatiotemporal samples is

Author's address: Yusuke Tokuyoshi, yusuke.tokuyoshi@amd.com, Advanced Micro Devices, Inc., Japan.

not always efficient when target PDFs for the reused samples are dissimilar to the integrand. Target PDFs are often spatially different for highly detailed scenes due to geometry edges, normal maps, spatially varying materials, and shadow edges. This paper introduces a new method of rejecting spatial reuse based on the similarity of PDF shapes for single-bounce path connections (e.g., direct illumination). While existing rejection methods for ReSTIR do not support arbitrary materials and shadow edges, our PDF similarity takes them into account because target PDFs include BSDFs and shadows. In this paper, we present a rough estimation of PDF shapes using von Mises–Fisher distributions and temporal resampling. We also present a stable combination of our rejection method and the existing rejection method, considering estimation errors due to temporal disocclusions and moving light sources. This combination efficiently reduces the error around shadow edges with temporal continuities. By using our method for a ReSTIR variant that reuses shadow ray visibility for the integrand, we can reduce the number of shadow rays while preserving shadow edges.

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

Additional Key Words and Phrases: many-light rendering, ReSTIR, von Mises–Fisher distribution

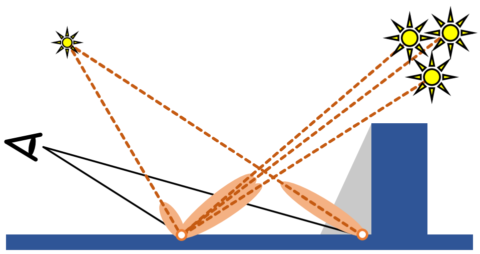
ACM Reference Format:
Yusuke Tokuyoshi. 2023. Efficient Spatial Resampling Using the PDF Similarity. *Proc. ACM Comput. Graph. Interact. Tech.* 6, 1, Article 4 (May 2023), 19 pages. https://doi.org/10.1145/3585501

1 INTRODUCTION

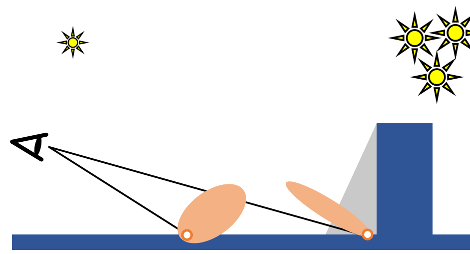
Hardware ray tracing and Monte Carlo integration are used for recent real-time applications as well as offline renderers. However, to achieve real-time frame rates, the number of rays must be limited to a few per pixel. Therefore, importance sampling is vital to render high-quality images with such a limited ray count. Recent resampling techniques based on *resampled importance sampling* (RIS) [Talbot 2005] generate samples approximately according to a target distribution by selecting samples from candidate samples. *Spatiotemporal reservoir resampling* (ReSTIR) [Bitterli et al. 2020] is one of the most powerful RIS-based techniques. It significantly increases candidates by reusing samples from past frames and neighboring pixels.

Although ReSTIR can reuse thousands of samples, spatial reuse is not always efficient for highly detailed scenes, because each pixel has a different target distribution due to geometry edges, normal maps, spatially varying materials, and shadow edges. When the target distribution for the reused pixel is significantly different from the integrand (i.e., path contribution for lighting) at the current pixel, the reuse can increase variance. This mismatch also increases a bias for biased ReSTIR variants [Bitterli et al. 2020; Wyman and Panteleev 2021] that reuse visibility to reduce the number of shadow rays for real-time applications. While existing ReSTIR rejected samples from reuse using some heuristics, such as the similarity of geometries, these heuristics did not support shadow edges and arbitrary materials. Thus, ReSTIR with visibility reuse using two rays per pixel produced a darkening bias around shadow edges (Fig. 1c). For aggressive visibility reuse using less than two rays per pixel, it lost shadow edges (Fig. 1a, Fig. 1b).

This paper introduces a new rejection method for spatial reuse in single-bounce path connections (e.g., direct illumination). Our method uses similarity in shapes of normalized target distributions (i.e., target PDFs) in the light direction domain. As shown in Fig. 2, since the target PDF includes shadows and the bidirectional scattering distribution function (BSDF), we can detect shadow edges and material boundaries by using this PDF similarity. Although it is infeasible to get the exact PDF shape for each pixel at real-time frame rates, we roughly estimate the PDF with the von Mises–Fisher (vMF) distribution [1953] using temporal resampling. Thus, our method reduces error around shadow edges with temporal continuities. Our temporal estimation for the PDF shape does not trace additional shadow rays by reusing the initial sample from lighting estimation, while it can introduce a negligibly small bias. We show that this bias is barely perceptible in our experimental



(a) Target PDFs at lit and shadowed pixels



(b) vMF approximation for target PDFs

Fig. 2. (a) Target PDFs are different between lit and shadowed pixels for direct illumination. This paper rejects undesirable sample reuse across shadow edges by using the similarity of the PDF shapes. (b) To simplify the computation, we roughly approximate the target PDFs with vMF distributions in the light direction domain. Using this vMF approximation, we estimate the PDF similarity from the vMF lobe axes and sharpness.

results. By applying our method for a biased ReSTIR with aggressive visibility reuse, we can render high-quality shadow edges with a small number of shadow rays.

Our contributions are as follows:

- We introduce a rejection method based on the similarity of PDF shapes for ReSTIR.
- To perform our method at real-time frame rates, we roughly approximate the PDF shape using a vMF. We also present a temporal estimation for this vMF approximation (Sect. 3.2).
- To handle the estimation error of the PDF shape, we combine an existing rejection heuristic and our PDF similarity based on the temporal continuity between frames (Sect. 3.3).
- We demonstrate the effectiveness of our method for several ReSTIR variants with different visibility computation methods (Sect. 4).

2 BACKGROUND

2.1 Related Work

Recent resampling techniques are built upon *sampling importance resampling* (SIR) [Rubin 1987]. SIR generates samples approximately according to a target distribution using a two-pass sampling algorithm. The first pass generates candidate samples according to a source PDF, and then the second pass selects samples from the candidates according to the ratio of the target distribution to the source PDF. For Monte Carlo integration, Talbot [2005] introduced resampled importance sampling (RIS), which unbiasedly normalizes the distribution of samples selected by SIR. We can perform SIR and RIS in a stream manner using *weighted reservoir sampling* [Chao 1982].

Bitterli et al. [2020] introduced spatiotemporal reservoir resampling (ReSTIR) to reuse samples across pixels and frames based on RIS. ReSTIR puts a sample into a reservoir for each pixel and then applies weighted reservoir sampling to spatiotemporal neighboring pixels. They also showed both biased and unbiased variants of their algorithm for direct illumination. In their biased variant, they reused the visibility of the initial sample for target distributions but did not reuse the visibility for the integrand. The reason is that their visibility reuse ignores high-frequency shadow edges in spatial reuse. Thus, the shadow edges in reused visibility are blurred and disappear during resampling. Wyman and Panteleev [2021] rearchitected the biased ReSTIR variant for production use. In this improved method, they reused the visibility for the integrand separately from Bitterli et al. [2020]'s visibility reuse. For their visibility reuse, they also proposed an adaptive shadow ray tracing based on the distance of current and reused pixels. Their adaptive approach allowed us to control the tradeoff between the performance and detailed shadows. Recently, ReSTIR has been extended to world-space reservoirs [Boissé 2021; Boksansky et al. 2021] and multi-bounce path samples for global illumination [Lin et al. 2021; Ouyang et al. 2021]. Lin et al. [2022] generalized RIS and ReSTIR. They also improved ReSTIR for path tracing by resampling similar paths using different domains.

To reduce variance and bias for ReSTIR and its biased variants, it is desirable to reuse only similar pixels for real-time rendering. Bitterli et al. [2020] heuristically rejected dissimilar pixels based on the similarity of geometries (i.e., depth and normal) as in an edge-stopping function for bilateral image denoising [Eisemann and Durand 2004; Petschnigg et al. 2004]. This rejection heuristic prevents the propagation of samples and the blurring of reused visibility across geometry edges. Lin et al. [2022] used roughness parameters of surfaces and edge length for their connectivity of paths. Unlike these approaches, our rejection method uses the similarity of target PDF shapes to take shadow edges and arbitrary materials into account for single-bounce path connections.

2.2 Algorithm of ReSTIR

Spatiotemporal reservoir resampling (ReSTIR) builds upon resampled importance sampling (RIS) [Talbot 2005]. RIS first generates candidate samples x_i according to a source PDF $p_i(x_i)$ and then randomly selects a sample X from the candidates according to the weight of each candidate w_i. For one-sample case, this RIS estimator is written as

$$\int_{\Omega_s} f(x)\mathrm{d}x \approx f(X)W_X, \tag{1}$$

where W_X is an unbiased contribution weight which is an estimated reciprocal PDF given by

$$W_X = \frac{1}{\hat{p}_s(X)} \sum_i w_i, \tag{2}$$

where $\hat{p}_s(x) \approx f(x)$ is the target distribution and its shape is more similar to the integrand $f(x)$ than the source PDF. For notations used in this paper, please see Table 1. In generalized RIS [Lin et al. 2022], the candidate weight is given by

$$w_i = m_i(T_i(x_i))\hat{p}_s\left(T_i(x_i)\right) W_i \left|\frac{\partial T_i}{\partial x_i}\right|, \tag{3}$$

where $m_i(\cdot)$ is the weight of *multiple importance sampling* (MIS) [Veach and Guibas 1995] that satisfies $\sum_i m_i(x) = 1$, T_i is a bijective shift mapping from the candidate's domain Ω_i to the integral domain Ω_s, and W_i is the contribution weight for the candidate i (e.g., $W_i = 1/p_i(x_i)$ for classic RIS [Talbot 2005]). One high-quality MIS weight for RIS is Talbot MIS [2005], and it is generalized by Lin et al. [2022] as follows:

$$m_i(x) = \frac{\hat{p}_{s\to i}(x)}{\sum_j \hat{p}_{s\to j}(x)}, \quad \text{where } \hat{p}_{s\to i}(x) = \begin{cases} \hat{p}_i\left(T_i^{-1}(x)\right)\left|\partial T_i^{-1}/\partial x\right| & \text{if } x \in T_i\left(\mathrm{supp}\left(\hat{p}_i\right)\right) \\ 0 & \text{otherwise} \end{cases}.$$

For other MIS weights, please refer to Lin et al. [2022]. In this generalized RIS, a sample X is selected from shifted candidates $T_i(x_i)$. For an infinite number of candidate samples, the resulting sample X follows the normalized target PDF $\bar{p}_s(X) = \hat{p}_s(X)/\|\hat{p}_s\|$ and W_X converges to $1/\bar{p}_s(X)$.

ReSTIR is a chained form of the generalized RIS, and it increases the number of candidates by reusing spatiotemporal neighboring samples stored in each pixel. This algorithm (shown in Algorithm 1) first performs classic RIS [Talbot 2005] for each pixel using a target distribution without shadow visibility. Then, a sample X is resampled from spatiotemporal neighboring pixels with visibility test (or visibility reuse [Bitterli et al. 2020]). This spatiotemporal resampling performs using the candidate weight given by Eq. 3 where i is a reused pixel. The contribution weight W_X

Table 1. Notations used in this paper

Symbol		Description
s		Current pixel to compute the integral
i		Candidate index, or reused pixel
$f(\cdot)$	$\in [0, \infty)$	Integrand, path contribution for lighting
Ω_s		Domain of integration for $f(\cdot)$
Ω_i		Domain of ith candidate sample
$p_i(\cdot)$	$\in [0, \infty)$	Source PDF for ith candidate sample
$\hat{p}_s(\cdot)$	$\in [0, \infty)$	Unnormalized target distribution: $\hat{p}_s(x) \approx f(x)$
$\bar{p}_s(\cdot)$	$\in [0, \infty)$	Normalized target PDF: $\bar{p}_s(x) = \hat{p}_s(x)/\|\hat{p}_s\|$, where $\|\hat{p}_s\| = \int_{\Omega_s} \hat{p}_s(x')\mathrm{d}x'$
X	$\in \Omega_s$	Path sample selected via RIS
x_i	$\in \Omega_i$	Path of candidate sample
w_i	$\in [0, \infty)$	Weight of candidate sample
W_i	$\in [0, \infty)$	Contribution weight, estimate reciprocal PDF
M_i	$\in [0, \infty)$	Accumulated count of candidate samples
$T_i(\cdot)$	$\in \Omega_s$	Shift mapping from Ω_i to Ω_s
$\mathbf{y}_{s,i}$	$\in \mathbb{R}^3$	Light vertex of the shifted candidate path $T_i(x_i)$
$\mathbf{z}_i$	$\in \mathbb{R}^3$	Shading point at pixel i
$\boldsymbol{\omega}$	$\in \mathrm{S}^2$	Unit vector
$\boldsymbol{\omega}_{s,i}$	$\in \mathrm{S}^2$	Candidate light direction from $\mathbf{z}_s$ to $\mathbf{y}_{s,i}$
$\mathbf{v}_i$	$\in [-1, 1]^3$	Temporal average of candidate light directions
$g(\cdot)$	$\in [0, \infty]$	von Mises–Fisher (vMF) distribution
$\boldsymbol{\mu}_s$	$\in \mathrm{S}^2$	Lobe axis of vMF
κ_s	$\in [0, \infty]$	Lobe sharpness of vMF

for each pixel is also updated using Eq. 2. In this chained RIS, we use an accumulated candidate count M_i for the MIS weight as follows:

$$m_i(x) = \frac{M_i \hat{p}_{s\to i}(x)}{\sum_j M_j \hat{p}_{s\to j}(x)}. \tag{4}$$

Sawhney et al. [2022] used a similar MIS weight for their temporal resampling. For the explicit form of the MIS weight used in this paper, please refer to Appendix A.

To improve the efficiency, ReSTIR rejects dissimilar pixels from reuse by using some heuristics (e.g., geometry similarity [Bitterli et al. 2020]). This rejection can be implemented by zeroing, reducing, or limiting the accumulated candidate count M_i for the reused pixel if the MIS weight takes the candidate count into account (as shown in Eq. 4). Bitterli et al. [2020] clamped the candidate count for past frames which may be different from the current frame for dynamic scenes. This clamping is also one of rejection heuristics. Such a reduction of candidate counts does not introduce a bias if the reduction rate is determined independently from samples. The rejection of dissimilar pixels is important to reduce error, especially for spatial reuse, because neighboring pixels can have detailed geometry, different materials, and shadow edges. The difference between pixels is often more significant than the difference between frames when lighting changes continuously. Therefore, we introduce a new rejection heuristic for spatial reuse.

ALGORITHM 1: ReSTIR algorithm that reuses one temporal neighbor and one spatial neighbor. This paper proposes a new rejection heuristic for spatial reuse (written in red).

```
function ReSTIR(s)
    [x_s, W_s, M_s] ← RIS(s);
    if Shadowed(x_s) then W_s ← 0 ;                              // Visibility reuse
    [x_s, W_s, M_s] ← Temporal Resampling(s, [x_s, W_s, M_s]);
    [x_s, W_s, M_s] ← SpatialResampling(s, [x_s, W_s, M_s]);
    StoreReservoir(s, [x_s, W_s, M_s]);
    return f(x_s)W_s;                                            // Eq. 1
end
function Temporal Resampling(s, [x_s, W_s, M_s])
    i ← PickTemporalNeighbor(s);
    [x_i, W_i, M_i] ← GetReservoir(i);
    h ← TemporalRejectionHeuristic(s, i);
    M_i ← min(M_i, M_max)h;         // Reduce the candidate count based on heuristics
    return Resampling(s, [x_s, W_s, M_s], i, [x_i, W_i, M_i]);
end
function SpatialResampling(s, [x_s, W_s, M_s])
    i ← PickSpatialNeighbor(s);
    [x_i, W_i, M_i] ← GetReservoir(i);
    h ← SpatialRejectionHeuristic(s, i);
    M_i ← M_i h;                    // Reduce the candidate count based on heuristics
    return Resampling(s, [x_s, W_s, M_s], i, [x_i, W_i, M_i]);
end
function Resampling(s, [x_s, W_s, M_s], i, [x_i, W_i, M_i])
    [m_s, m_i] ← MISWeights(s, x_s, M_s, i, x_i, M_i);           // Eqs. 17 and 18
    w_s ← m_s p̂_s(x_s)W_s; w_i ← m_i p̂_s(T_i(x_i))W_i|∂T_i/∂x_i|;   // Eq. 3
    w_sum ← w_s + w_i;
    ξ ← GenerateRandomNumber();
    if ξ < w_i/w_sum then X ← x_i else X ← x_s ;
    W_X ← w_sum/p̂_s(X);                                          // Eq. 2
    M_X ← M_s + M_i;
    return [X, W_X, M_X];
end
```

3 OUR REJECTION METHOD FOR RESTIR

As mentioned in Sect. 2.2, the reuse of similar pixels is desirable to reduce error for ReSTIR. While Lin et al. [2022] described such similar pixels as similar path contributions: $f(x) \approx f(T_i(x))$ and $|\partial T_i/\partial x| \approx 1$, we propose to reuse pixels with similar normalized target PDFs instead of unnormalized target distributions or path contributions (Sect. 3.1). Then, we introduce a similarity of normalized target PDFs between pixels for our rejection heuristic. This approach takes shadows and arbitrary materials into account since target PDFs include them. In this section, we present an efficient method to compute the PDF similarity for spatial reuse in single-bounce path connections. The pseudo code of our method is shown in Algorithm 2.

3.1 Resampling with Similar PDFs

ReSTIR reduces error by converging the contribution weight W_X to $1/\bar{p}_s(X)$ for many candidate samples. For this case, by substituting $W_X \approx 1/\bar{p}_s(X)$ and Eq. 3 in Eq. 2, we yield

$$\sum_i m_i(T_i(x_i))\bar{p}_s\left(T_i(x_i)\right) W_i \left|\frac{\partial T_i}{\partial x_i}\right| \approx 1. \tag{5}$$

When ReSTIR converges, we also obtain $W_i \approx 1/\bar{p}_i(x_i)$. Therefore, we can rewrite Eq. 5 into

$$\sum_i m_i(T_i(x_i))\frac{\bar{p}_s\left(T_i(x_i)\right)}{\bar{p}_i\left(x_i\right)} \left|\frac{\partial T_i}{\partial x_i}\right| \approx 1. \tag{6}$$

ALGORITHM 2: ReSTIR with our rejection method (our contribution is written in red). *DirectionTemporalResampling* is the same as *TemporalResampling* except for the rejection of moving lights and the calculation of the average-light direction $\mathbf{v}_X$. Using the average directions, *OurRejectionHeuristic* approximately computes the PDF similarity between pixels.

function ReSTIR(s)
    $[x_s, W_s, M_s] \leftarrow$ RIS(s);
    if Shadowed(x_s) **then** $W_s \leftarrow 0$; // Visibility reuse
    $\omega_{s,s} \leftarrow$ GetLightDirection(x_s);
    $[x_s^{\mathrm{v}}, W_s^{\mathrm{v}}, M_s^{\mathrm{v}}, \mathbf{v}_s] \leftarrow [x_s, W_s, M_s, \omega_{s,s}]$; // Reuse the initial sample
    $[x_s^{\mathrm{v}}, W_s^{\mathrm{v}}, M_s^{\mathrm{v}}, \mathbf{v}_s] \leftarrow$ DirectionTemporalResampling(s, $[x_s^{\mathrm{v}}, W_s^{\mathrm{v}}, M_s^{\mathrm{v}}, \mathbf{v}_s]$);
    StoreDirectionReservoir(s, $[x_s^{\mathrm{v}}, W_s^{\mathrm{v}}, M_s^{\mathrm{v}}, \mathbf{v}_s]$);
    $[x_s, W_s, M_s] \leftarrow$ TemporalResampling(s, $[x_s, W_s, M_s]$);
    $[x_s, W_s, M_s] \leftarrow$ SpatialResampling(s, $[x_s, W_s, M_s]$, $[W_s^{\mathrm{v}}, M_s^{\mathrm{v}}, \mathbf{v}_s]$);
    StoreReservoir(s, $[x_s, W_s, M_s]$);
    return $f(x_s)W_s$; // Eq. 1
end
function DirectionTemporalResampling(s, $[x_s, W_s, M_s, \mathbf{v}_s]$)
    $i \leftarrow$ PickTemporalNeighbor(s);
    $[x_i, W_i, M_i, \mathbf{v}_i] \leftarrow$ GetDirectionReservoir(i);
    $h \leftarrow$ DirectionRejectionHeuristic(s, i, x_i); // Rejection heuristic for moving lights (Eq. 12)
    $M_i \leftarrow \min(M_i, M_{\max})h$; // Reduce the candidate count based on heuristics
    $[m_s, m_i] \leftarrow$ MISWeights(s, x_s, M_s, i, x_i, M_i); // Eqs. 17 and 18
    $w_s \leftarrow m_s \hat{p}_s(x_s) W_s$; $w_i \leftarrow m_i \hat{p}_s(T_i(x_i)) W_i |\partial T_i / \partial x_i|$; // Eq. 3
    $w_{\mathrm{sum}} \leftarrow w_s + w_i$;
    $\xi \leftarrow$ GenerateRandomNumber();
    if $\xi < w_i / w_{\mathrm{sum}}$ **then** $X \leftarrow x_i$ **else** $X \leftarrow x_s$;
    $W_X \leftarrow w_{\mathrm{sum}} / \hat{p}_s(X)$; // Eq. 2
    $M_X \leftarrow M_s + M_i$;
    $\mathbf{v}_X \leftarrow (\mathbf{v}_s w_s + \mathbf{v}_i w_i) / w_{\mathrm{sum}}$; // Average the light directions (Eq. 10)
    return $[X, W_X, M_X, \mathbf{v}_X]$;
end
function SpatialResampling(s, $[x_s, W_s, M_s]$, $[W_s^{\mathrm{v}}, M_s^{\mathrm{v}}, \mathbf{v}_s]$)
    $i \leftarrow$ PickSpatialNeighbor(s);
    $[x_i, W_i, M_i] \leftarrow$ GetReservoir(i);
    $h_{\mathrm{prev}} \leftarrow$ ExistingRejectionHeuristic(s, i);
    $[x_i^{\mathrm{v}}, W_i^{\mathrm{v}}, M_i^{\mathrm{v}}, \mathbf{v}_i] \leftarrow$ GetDirectionReservoir(i);
    $h_{\mathrm{our}} \leftarrow$ OurRejectionHeuristic($\mathbf{v}_s, \mathbf{v}_i$);
    $t \leftarrow \max((\min(M_s^{\mathrm{v}}, M_i^{\mathrm{v}}) - M)/M_{\max}, 0)$; // Eq. 16
    if $W_s^{\mathrm{v}} > 0 \wedge W_i^{\mathrm{v}} > 0$ **then** $h \leftarrow \mathrm{lerp}(h_{\mathrm{prev}}, h_{\mathrm{our}}, t)$ **else** $h \leftarrow h_{\mathrm{prev}}$; // Combine heuristics (Eq. 15)
    $M_i \leftarrow M_i h$; // Reduce the candidate count based on heuristics
    return Resampling(s, $[x_s, W_s, M_s]$, i, $[x_i, W_i, M_i]$);
end
function OurRejectionHeuristic($\mathbf{v}_s, \mathbf{v}_i$)
    $[\mu_s, \kappa_s] \leftarrow$ ConvertVMF($\mathbf{v}_s$); $[\mu_i, \kappa_i] \leftarrow$ ConvertVMF($\mathbf{v}_i$); // Get vMF axis and sharpness (Eq. 9)
    $\tilde{\kappa}_s \leftarrow \kappa_s \alpha / (\kappa_s + \alpha)$; $\tilde{\kappa}_i \leftarrow \kappa_i \alpha / (\kappa_i + \alpha)$; // Smooth vMFs (Eq. 13)
    return $\left(2\sqrt{\tilde{\kappa}_s \tilde{\kappa}_i} / (\tilde{\kappa}_s + \tilde{\kappa}_i)\right)^{\beta} \exp\left(\frac{\beta \tilde{\kappa}_s \tilde{\kappa}_i}{\tilde{\kappa}_s + \tilde{\kappa}_i}\left((\mu_s \cdot \mu_i) - 1\right)\right)$; // Similarity of the smoothed vMFs (Eq. 14)
end

Since $\sum_i m_i(T_i(x_i)) = 1$, ReSTIR can have a small error in the following case:

$$\boxed{\frac{\bar{p}_s\left(T_i(x_i)\right)}{\bar{p}_i\left(x_i\right)} \left|\frac{\partial T_i}{\partial x_i}\right| \approx 1}. \tag{7}$$

Since this is the ratio of normalized target PDFs instead of unnormalized target distributions, we use the similarity of normalized PDFs for our rejection heuristic. Although Eq. 6 is a necessary condition for convergence and not a sufficient condition, we show that our method reduces error in our experimental results (Sect. 4).

PDFs for Single-Bounce Path Connections. Although we can use vertex parametrization [Veach 1998] for a path sample x_i, this paper expresses our method using the light direction domain for single-bounce path connections as follows:

$$\frac{\bar{p}_s\left(T_i(x_i)\right)}{\bar{p}_i\left(x_i\right)}\left|\frac{\partial T_i}{\partial x_i}\right| = \frac{\bar{p}_s\left(\omega_{s,i}\right)}{\bar{p}_i\left(\omega_{i,i}\right)}\frac{|\mathbf{n}(\mathbf{y}_{s,i})\cdot\omega_{s,i}|/\|\mathbf{y}_{s,i}-\mathbf{z}_s\|^2}{|\mathbf{n}(\mathbf{y}_{i,i})\cdot\omega_{i,i}|/\|\mathbf{y}_{i,i}-\mathbf{z}_i\|^2}, \tag{8}$$

where $x_i = [\mathbf{c}_i, \mathbf{z}_i, \mathbf{y}_{i,i}]$ is the path from the camera vertex $\mathbf{c}_i$ to the light vertex $\mathbf{y}_{i,i}$ at the reused pixel i, $\mathbf{z}_i$ is the shading point at the reused pixel i, $\mathbf{y}_{s,i}$ is the reused light vertex shifted to current frame, $\mathbf{n}(\mathbf{y}_{s,i})$ is the surface normal at $\mathbf{y}_{s,i}$, and $\omega_{s,i} = (\mathbf{y}_{s,i} - \mathbf{z}_s)/\|\mathbf{y}_{s,i} - \mathbf{z}_s\|$ is the light direction from the shading point $\mathbf{z}_s$. The spherical PDF $\bar{p}_s\left(\omega\right)$ represents the product of incoming radiance and the cosine-weighted BSDF at $\mathbf{z}_s$ as follows:

$$\bar{p}_s\left(\omega\right) = \frac{L\left(\mathbf{z}_s,\omega\right)\rho\left(\mathbf{z}_s, \frac{\mathbf{c}_s-\mathbf{z}_s}{\|\mathbf{c}_s-\mathbf{z}_s\|},\omega\right)|\mathbf{n}(\mathbf{z}_s)\cdot\omega|}{\int_{\mathrm{S}^2} L(\mathbf{z}_s,\omega')\rho\left(\mathbf{z}_s, \frac{\mathbf{c}_s-\mathbf{z}_s}{\|\mathbf{c}_s-\mathbf{z}_s\|},\omega'\right)|\mathbf{n}(\mathbf{z}_s)\cdot\omega'|\mathrm{d}\omega'},$$

where $\rho(\cdot)$ is the BSDF, and $L\left(\mathbf{z}_s,\omega\right)$ is the incoming radiance given by the product of the visibility of the light source and its emissive radiance viewed from the shading point $\mathbf{z}_s$. When shading points are close and lighting is static (i.e., $\mathbf{z}_i \approx \mathbf{z}_s$ and $\mathbf{y}_{i,i} = \mathbf{y}_{s,i}$), we get $\omega_{i,i} \approx \omega_{s,i}$ and $|\mathbf{n}(\mathbf{y}_{i,i})\cdot\omega_{i,i}|/\|\mathbf{y}_{i,i}-\mathbf{z}_i\|^2 \approx |\mathbf{n}(\mathbf{y}_{s,i})\cdot\omega_{s,i}|/\|\mathbf{y}_{s,i}-\mathbf{z}_s\|^2$. However, even for such close shading points, the shapes of spherical PDFs $\bar{p}_i\left(\omega\right)$ and $\bar{p}_s\left(\omega\right)$ may be dissimilar due to shadow edges and material boundaries (Fig. 2a). Therefore, we evaluate a shape similarity between PDFs $\bar{p}_i\left(\omega\right)$ and $\bar{p}_s\left(\omega\right)$ in the light direction domain.

3.2 Similarity Computation for Spherical PDFs

3.2.1 vMF Approximation. It is difficult to obtain the exact shape of the target PDF $\bar{p}_s(\omega)$ in practice. Therefore, as shown in Fig. 2b, we roughly approximate the PDF with the von Mises–Fisher (vMF) distribution in S^2 (a.k.a. normalized spherical Gaussian [Tsai and Shih 2006; Wang et al. 2009]):

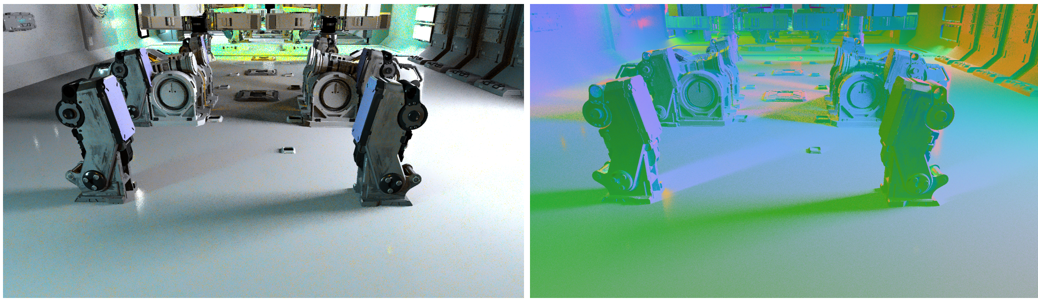
$$\bar{p}_s(\omega) \approx g(\omega; \mu_s, \kappa_s) = \frac{\kappa_s}{4\pi\sinh\kappa_s}\exp\left(\kappa_s\left(\omega\cdot\mu_s\right)\right),$$

where μ_s and κ_s are the lobe axis and sharpness to represent $\bar{p}_s(\omega)$. This vMF distribution is obtained by the average direction of the PDF $\acute{\mathbf{v}}_s = \int_{\mathrm{S}^2}\omega\bar{p}_s(\omega)\mathrm{d}\omega$ using Banerjee et al.'s conversion [2005]:

$$\mu_s = \frac{\acute{\mathbf{v}}_s}{\|\acute{\mathbf{v}}_s\|}, \quad \kappa_s = \frac{3\|\acute{\mathbf{v}}_s\| - \|\acute{\mathbf{v}}_s\|^3}{1-\|\acute{\mathbf{v}}_s\|^2}. \tag{9}$$

For this vMF approximation, we roughly estimate the average direction $\acute{\mathbf{v}}_s$ at real-time frame rates.

3.2.2 Temporal Estimation of the Average Direction. For single-bounce path connections, we can rewrite the spherical integral $\acute{\mathbf{v}}_s = \int_{\mathrm{S}^2}\omega\bar{p}_s(\omega)\mathrm{d}\omega$ into a path-space integral: $\acute{\mathbf{v}}_s = \int_{\Omega_s}\omega(x)\bar{p}_s(x)\mathrm{d}x$. Therefore, to estimate the average direction $\acute{\mathbf{v}}_s$, this paper uses a biased variant of ReSTIR which resample a light direction ω_X according to the target distribution $\hat{p}_s(X)$. In our average-direction estimation, we reuse the visibility over time similar to existing biased ReSTIR methods [Bitterli et al. 2020; Wyman and Panteleev 2021]. Unlike regular ReSTIR, we reuse only temporally neighboring pixels and do not reuse spatially neighboring pixels to preserve shadow edges and material boundaries. In addition, we reuse the initial sample from the lighting estimation (see Algorithm 2). Thus, our average-direction estimation does not trace additional shadow rays. In ReSTIR, a selected sample direction ω_X is used to estimate the integral as follows: $\acute{\mathbf{v}}_s \approx \omega_X\bar{p}_s(X)W_X$, but one sample



(a) Rendered image (b) Average light directions

Fig. 3. Visualization of the estimated average light direction for each pixel. Average light directions are different between lit and shadowed pixels. By using these directions, we detect shadow edges for our rejection method.

direction is insufficient to estimate the average direction. Therefore, instead of selecting one direction $\boldsymbol{\omega}_X$ according to the candidate weight, we temporally accumulate candidate directions as follows:

$$\mathbf{v}_X = \frac{\mathbf{v}_s w_s + \mathbf{v}_i w_i}{w_s + w_i}, \tag{10}$$

where $\mathbf{v}_s = \boldsymbol{\omega}_{s,s}$ is the initial candidate direction, $\mathbf{v}_i$ is the weighted average direction at the previous frame, w_s and w_i are candidate weights for initial and reused samples, and $\mathbf{v}_X$ will be reused for the next frame as $\mathbf{v}_i$. Since the accumulated candidate count is clamped for temporal resampling (as mentioned in Sect. 2.2), $\mathbf{v}_X$ can be an exponential moving average of sample directions. If the scene is not animated and thus $T_i(x_i) = x_i$ between frames, we can rewrite the Monte Carlo estimator with an exponential MIS weight m'_j into the following temporal estimator:

$$\acute{\mathbf{v}}_s \approx \sum_j \frac{\boldsymbol{\omega}_{j,j} \bar{p}_s(x_j) m'_j(x_j)}{p_j(x_j)} = \frac{\mathbf{v}_s w_s + \mathbf{v}_i w_i}{\|\hat{p}_s\|} = \mathbf{v}_X \bar{p}_s(X) W_X, \tag{11}$$

where x_j and $\boldsymbol{\omega}_{j,j}$ are the initial sample and its direction for each frame. However, it is infeasible to evaluate the normalized target PDF $\bar{p}_s(X) = \hat{p}_s(X)/\|\hat{p}_s\|$ analytically. By substituting $W_X \approx 1/\bar{p}_s(X)$ in Eq. 11, we obtain the following simple approximation:

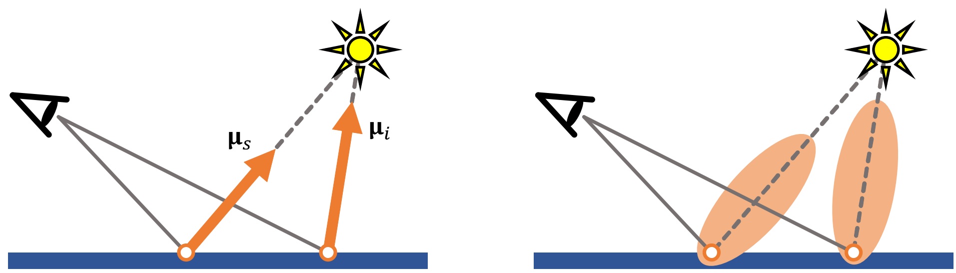
$$\boxed{\acute{\mathbf{v}}_s \approx \mathbf{v}_X}.$$

Since $\mathbf{v}_X$ is a weighted average (Eq. 10), this estimator is a variant of weighted importance sampling [Bekaert et al. 2000] (or ratio estimator [Heitz et al. 2018]). Thus, it has a bias due to normalization, but the bias reduces quickly. In addition, this approximation satisfies $\|\mathbf{v}_X\| \leq 1$ which is required for Eq. 9. Fig. 3 shows visualization of estimated average direction $\mathbf{v}_X$ for each pixel. In this scene, average directions are different between lit and shadowed pixels.

Although our approximation is efficient, it can produce a temporal delay especially for moving shadows. To reduce the delay for shadows, we reject rapidly moving lights from temporal reuse by using the following light-direction-based heuristic:

$$h_{\text{dir}} = h_{\text{prev}} \exp\left(\lambda \left(\left(\boldsymbol{\omega}_{i,i} \cdot \boldsymbol{\omega}_{s,i}\right) - 1\right)\right), \tag{12}$$

where $h_{\text{prev}} \in [0, 1]$ is an existing rejection heuristic (e.g., geometry similarity [Bitterli et al. 2020]), $\boldsymbol{\omega}_{i,i}$ and $\boldsymbol{\omega}_{s,i}$ are light directions at previous and current frames, and $\lambda \in (0, \infty)$ is a user-specified parameter to control the sensitivity for moving lights ($\lambda = 1000$ is used in this paper). We multiply



(a) Delta PDFs for a point light (b) Smoothed PDFs

Fig. 4. PDFs are delta functions for a point light source (a). Although the PDF similarity between pixels can be obtained using the product integral of PDFs, it is always zero for delta PDFs since we ignore the shift of light directions. For such shifted high-frequency PDFs, we use the similarity of smoothed PDFs (b).

the accumulated candidate count by h_{dir} to reject past candidate directions. When lighting changes, although the rejection of moving lights introduces a bias and variance for our average-direction estimation, we obtain the temporal continuity of lighting from the reduced candidate count. We use this temporal continuity in Sect. 3.3 to handle the estimation error of the average direction.

3.2.3 Similarity of vMFs. Once the vMF distribution (i.e., approximate target PDF) is estimated for each pixel, we compute the similarity of them. In this paper, we use a product integral-based similarity to evaluate the overlaps of PDFs. However, if a scene has only one point light source, the PDFs are delta functions and there is no overlaps since we ignore the shift of the lobe axis as shown in Fig. 4a. We cannot evaluate the similarity for this case. Therefore, to obtain the similarity based on the distance between shifted lobe axes for such high-frequency PDFs, we smooth each PDF (Fig. 4b) using a smoothing kernel $g(\boldsymbol{\omega}'; \boldsymbol{\omega}, \alpha)$ as follows:

$$\tilde{p}_s(\boldsymbol{\omega}) = \int_{\mathbb{S}^2} \bar{p}_s(\boldsymbol{\omega}')\, g(\boldsymbol{\omega}'; \boldsymbol{\omega}, \alpha) \mathrm{d}\boldsymbol{\omega}' \approx \int_{\mathbb{S}^2} g(\boldsymbol{\omega}'; \boldsymbol{\mu}_s, \kappa_s) g(\boldsymbol{\omega}'; \boldsymbol{\omega}, \alpha) \mathrm{d}\boldsymbol{\omega}' \approx g\left(\boldsymbol{\omega}; \boldsymbol{\mu}_s, \tilde{\kappa}_s\right), \quad (13)$$

where $\alpha \in (0, \infty)$ is a user-specified kernel sharpness to control the sensitivity for the shift ($\alpha = 100$ is used in this paper), and $\tilde{\kappa}_s = \kappa_s \alpha / (\kappa_s + \alpha)$ is derived in Iwasaki et al. [2012]. Then, we compute the similarity of the smoothed vMFs between pixels. In this paper, we use an analytical product integral-based similarity derived in Tokuyoshi [2015] as follows:

$$h_{\mathrm{our}} = \left(\frac{\int_{\mathbb{S}^2} \tilde{p}_s(\boldsymbol{\omega}) \tilde{p}_i(\boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega}}{\sqrt{\int_{\mathbb{S}^2} \left(\tilde{p}_s(\boldsymbol{\omega})\right)^2 \mathrm{d}\boldsymbol{\omega} \int_{\mathbb{S}^2} \left(\tilde{p}_i(\boldsymbol{\omega})\right)^2 \mathrm{d}\boldsymbol{\omega}}} \right)^{\beta} \approx \left(\frac{\int_{\mathbb{S}^2} g\left(\boldsymbol{\omega}; \boldsymbol{\mu}_s, \tilde{\kappa}_s\right) g\left(\boldsymbol{\omega}; \boldsymbol{\mu}_i, \tilde{\kappa}_i\right) \mathrm{d}\boldsymbol{\omega}}{\sqrt{\int_{\mathbb{S}^2} \left(g\left(\boldsymbol{\omega}; \boldsymbol{\mu}_s, \tilde{\kappa}_s\right)\right)^2 \mathrm{d}\boldsymbol{\omega} \int_{\mathbb{S}^2} \left(g\left(\boldsymbol{\omega}; \boldsymbol{\mu}_i, \tilde{\kappa}_i\right)\right)^2 \mathrm{d}\boldsymbol{\omega}}} \right)^{\beta}$$
$$\approx \boxed{\left(\frac{2\sqrt{\tilde{\kappa}_s \tilde{\kappa}_i}}{\tilde{\kappa}_s + \tilde{\kappa}_i} \right)^{\beta} \exp\left(\frac{\beta \tilde{\kappa}_s \tilde{\kappa}_i}{\tilde{\kappa}_s + \tilde{\kappa}_i} \left(\left(\boldsymbol{\mu}_s \cdot \boldsymbol{\mu}_i\right) - 1 \right) \right)}, \quad (14)$$

where $\beta \in (0, \infty)$ is a user-specified parameter to control the sensitivity for our rejection heuristic ($\beta = 10$ is used in this paper). Using this PDF similarity $h_{\mathrm{our}} \in [0, 1]$, we can prevent spatial reuse across shadow edges and material boundaries if estimated vMFs have small errors.

3.3 Combination with Existing Heuristics

Although our PDF similarity can detect shadow edges and material boundaries, it has a variance caused by the temporal estimation of the average direction. In addition, since our average-direction estimation shares the initial sample with lighting estimation, the variance of our PDF similarity can correlate to the variance of lighting. This correlation results in a bias in the rejection of spatial reuse. Although the variance is decorrelated by using different random numbers in every resampling routine, the variance and its correlation may be noticeable when the number of candidate samples is small due to temporal disocclusions and rapidly moving lights. Therefore, we use our rejection heuristic only when the candidate count is sufficient. In this paper, we interpolate our heuristic h_{our} and existing heuristic h_{prev} using the temporally accumulated candidate count as follows:

$$\boxed{h = \begin{cases} th_{\text{our}} + (1-t)h_{\text{prev}} & \text{if } W_s^{\mathbf{v}} > 0 \text{ and } W_i^{\mathbf{v}} > 0 \\ h_{\text{prev}} & \text{otherwise} \end{cases}}, \tag{15}$$

$$t = \max\left(\frac{\min(M_s^{\mathbf{v}}, M_i^{\mathbf{v}}) - M}{M_{\max}}, 0\right), \tag{16}$$

where $M_s^{\mathbf{v}}$ and $M_i^{\mathbf{v}}$ are accumulated candidate counts for our average-direction estimation, M is the initial candidate count, and $M_{\max}$ is the maximum candidate count for temporal reuse (we set $M_{\max} = 20M$ as in Bitterli et al. [2020]). If the contribution weight $W_s^{\mathbf{v}}$ or $W_i^{\mathbf{v}}$ for the average-direction estimation is zero, either light direction is indefinite. Thus, we use only the existing heuristic for this case. Using this combination, our heuristic is effective only for temporally continuous pixels and static or slowly moving lights.

4 EXPERIMENTAL RESULTS

Here we show results using our rejection method and the previous geometry-based rejection method [Bitterli et al. 2020] for several ReSTIR variants. We implement these methods on Microsoft MiniEngine using DirectX Raytracing. All images are rendered with 1920×1080 screen resolution on an AMD Radeon™ RX 6900 XT GPU. The image quality is evaluated with the symmetric mean absolute percentage error (SMAPE) metric. For direct illumination, we generate 1048576 virtual point lights (1 M VPLs) [Keller 1997] on area light sources, and then we sample one VPL using ReSTIR from them. For the first RIS pass in the ReSTIR algorithm (Algorithm 2), we use an unbiased tile-based light culling [Tokuyoshi 2022] to improve the efficiency. For spatial reuse, we sample neighboring pixels according to the Gaussian distribution of variance 64. In our experimental implementation, we use 16 bytes per pixel to store reservoirs for our average-direction estimation (32-bit integer for a VPL index, 32-bit floating point for $W_s^{\mathbf{v}}$, 16-bit floating point for $M_s^{\mathbf{v}}$, and 16-bit floating point for each dimension of $\mathbf{v}_s$).

ReSTIR with two rays per pixel. Fig. 5 shows ReSTIR using two shadow rays per pixel for visibility reuse (i.e., one shadow ray is reused for the target distribution $\hat{p}_i$, and the other shadow ray is reused for the integrand f [Wyman and Panteleev 2021]). Although this visibility reuse is efficient for real-time applications, it duplicately casts shadows on shadow edges and thus produces a darkening bias. Using our rejection method, we reduce both bias and variance on shadow edges for temporally continuous pixels. Our method also reduces variance around temporally continuous glossy highlights, since the target PDF includes the BSDF. While the computational complexity of our method is constant for each pixel, it samples more visible lights than the previous method. Thus, our method can affect shadow ray tracing cost which depends on the complexity of the scene geometry. For scenes in Fig. 5, the total overhead for our method is about 0.2 milliseconds.

ReSTIR with one ray per pixel. Fig. 6 shows ReSTIR that reuses the visibility of the initial sample for the integrand f as well as the target distribution $\hat{p}_i$. When using the previous rejection heuristic for this case, shadow edges blur and disappear due to spatial reuse. By using our rejection method with

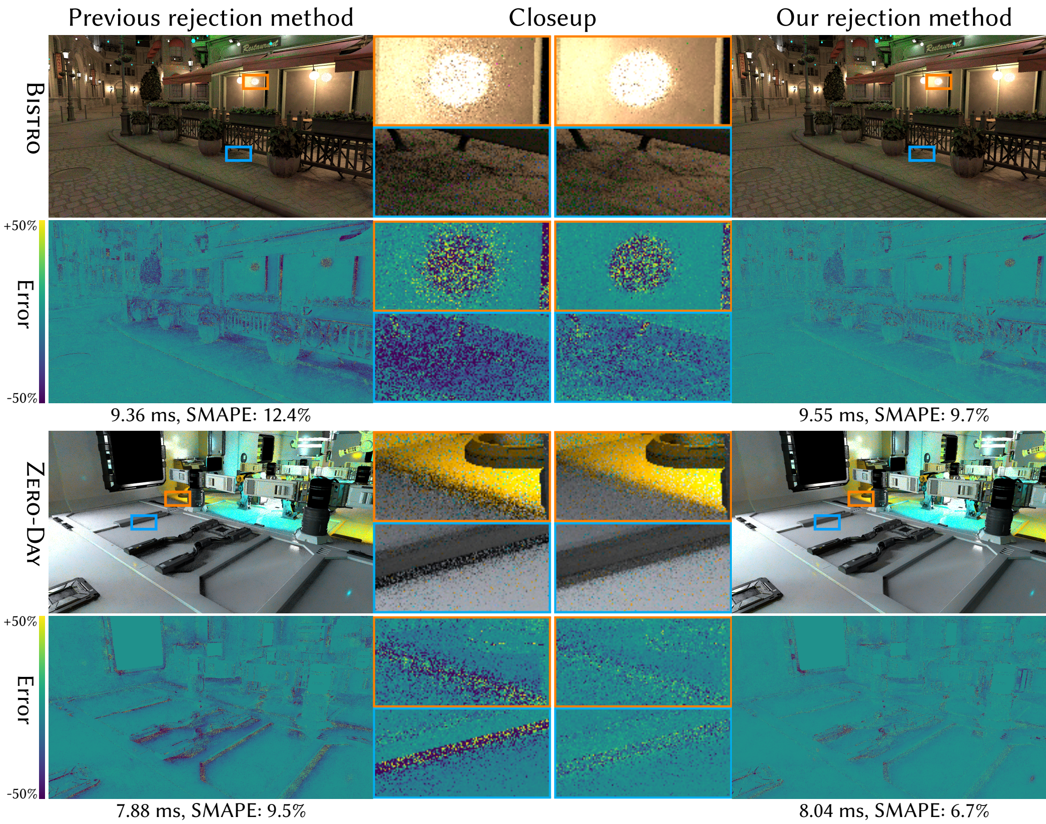


Fig. 5. Visibility-reuse ReSTIR with two rays per pixel (2 rpp) for the Bistro scene (2.8 M triangles, 20.6 k triangle lights) and the Zero-Day scene (5.2 M triangles, 10.3 k triangle lights). Our rejection method reduces a darkening bias as well as variance on shadow edges. Our method also reduces variance around glossy highlights with temporal continuities.

an overhead of about 0.3 milliseconds, we obtain hard contact shadows for temporally continuous pixels without tracing two rays per pixel. On the other hand, our method produces almost the same results as the previous heuristic for temporal disocclusions (Fig. 7).

ReSTIR with adaptive ray tracing. Whether or not to trace a shadow ray for the integrand can be determined based on the distance between current and reused pixels [Wyman and Panteleev 2021]. For this adaptive ray tracing (Fig. 8), we can control the tradeoff between the performance and detailed shadows by using a threshold for the distance. Even with a small number of rays per pixel, our rejection method preserves hard contact shadows more than the previous method for temporally continuous pixels. To obtain hard shadow edges for temporal disocclusions, we should trace a shadow ray for such pixels. In Fig. 9, we stochastically trace a ray according to a probability $1 - t$ (where t is the temporal continuity given by Eq. 16) in addition to the distance-based approach. When the camera moves, although this approach traces more shadow rays than using only the pixel distance and can increase a darkening bias, it produces more highly detailed and temporally coherent shadows.

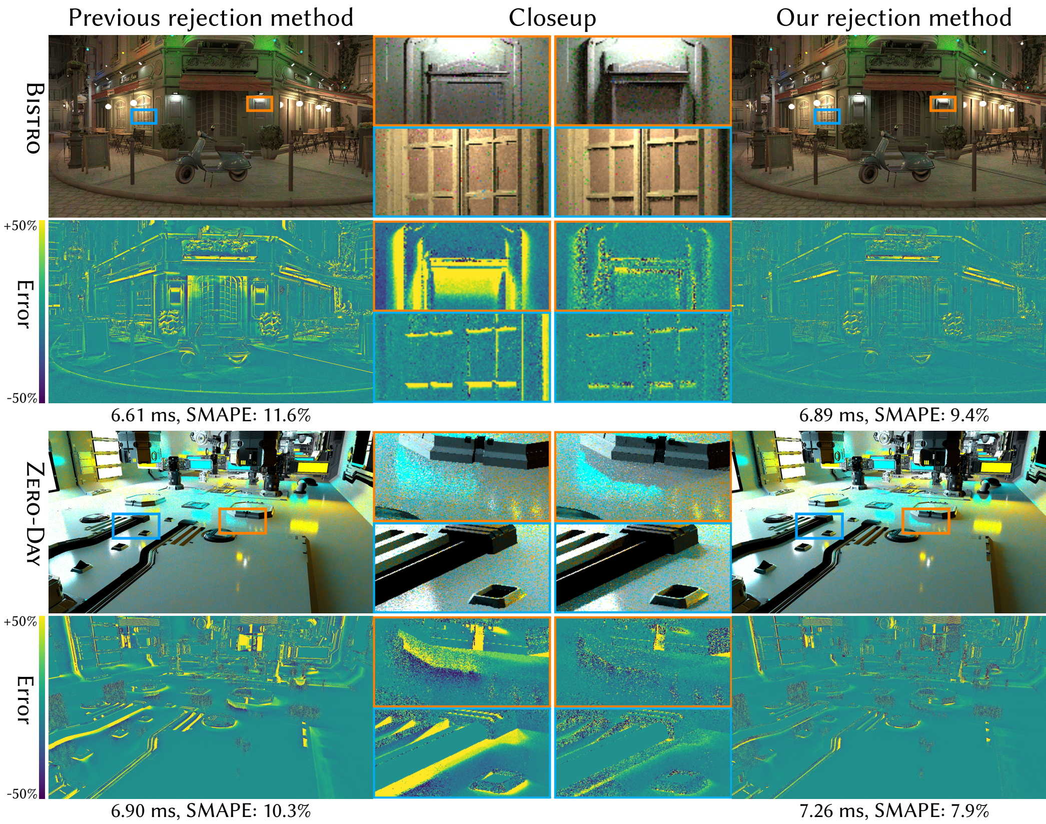


Fig. 6. Visibility-reuse ReSTIR with one ray per pixel (1 rpp) for the Bistro scene (2.8 M triangles, 20.6 k triangle lights) and the Zero-Day scene (5.2 M triangles, 10.3 k triangle lights). While the previous rejection method loses hard contact shadows, our method reduces the loss of these high-frequency shadows.

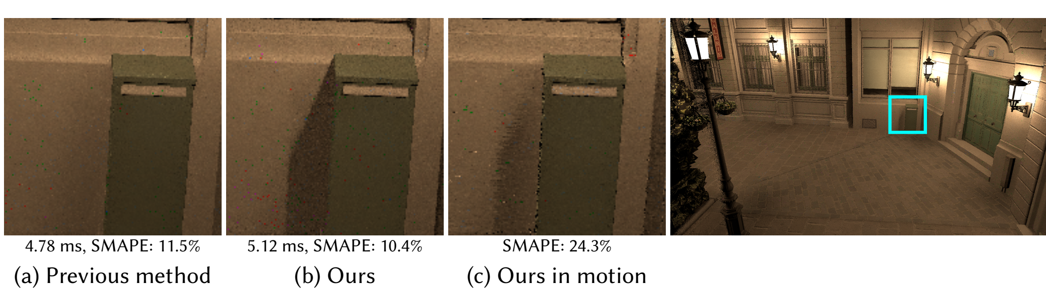


Fig. 7. Visibility-reuse ReSTIR with 1 rpp for a static camera (a, b) and moving camera (c) in the Bistro scene. Although our heuristic (b) helps to render high-frequency shadows unlike the previous heuristic (a), it works only for temporal continuities. (c) For temporal discontinuities, our method uses the previous heuristic, and thus it can lose shadows in motion.

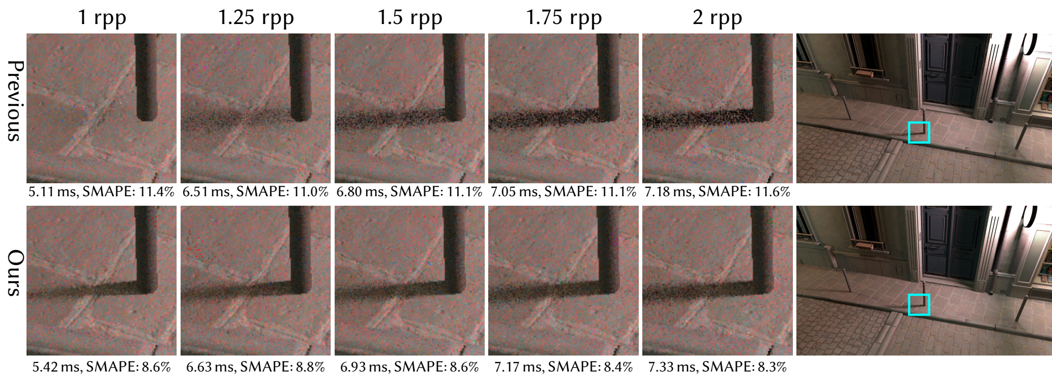
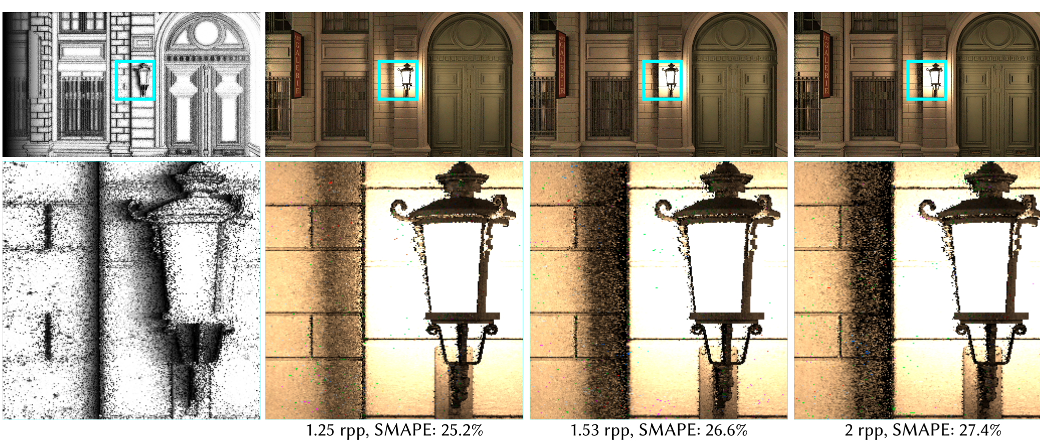


Fig. 8. Pixel-distance-based adaptive shadow ray tracing [Wyman and Panteleev 2021] with the previous rejection method (upper row) and our rejection method (lower row) for the Bistro scene. Our method preserves more contact shadows than the previous method for a small number of rays per pixel (rpp).



(a) Temporal continuity t (b) Spatially adaptive (c) Spatiotemporally adaptive (d) Nonadaptive

Fig. 9. Combination of our rejection method and adaptive shadow ray tracing for the Bistro scene with a moving camera. Since our heuristic works only for temporally continuous pixels, (b) pixel-distance-based adaptive tracing [Wyman and Panteleev 2021] can lose shadows for temporal disocclusions. (c) We can compensate the lost shadows by tracing more shadow rays based on the temporal continuity, while this approach can increase a darkening bias as in 2-rpp nonadaptive tracing (d).

ReSTIR with exact visibility test. Fig. 10 shows ReSTIR using exact visibility test for target distributions. In our implementation, it requires five rays per pixel. For this case, the MIS weight using target distributions already takes the shadow edges into account. Even using this MIS, our rejection method reduces variance with an overhead of 0.5 milliseconds in our experiment. This is because the above MIS weight ignores the normalization of target distributions, while our PDF similarity takes this normalization into account. The normalization factor for the PDF $\|\hat{p}_s\| \approx \int_{\Omega_s} f(x)\mathrm{d}x$ is approximately equal to the expected value of the pixel luminance. Therefore, the normalization factors are significantly different between a lit pixel and shadowed pixel. We take this difference into account for our PDF similarity. Our method can have a bias due to the correlation of variance

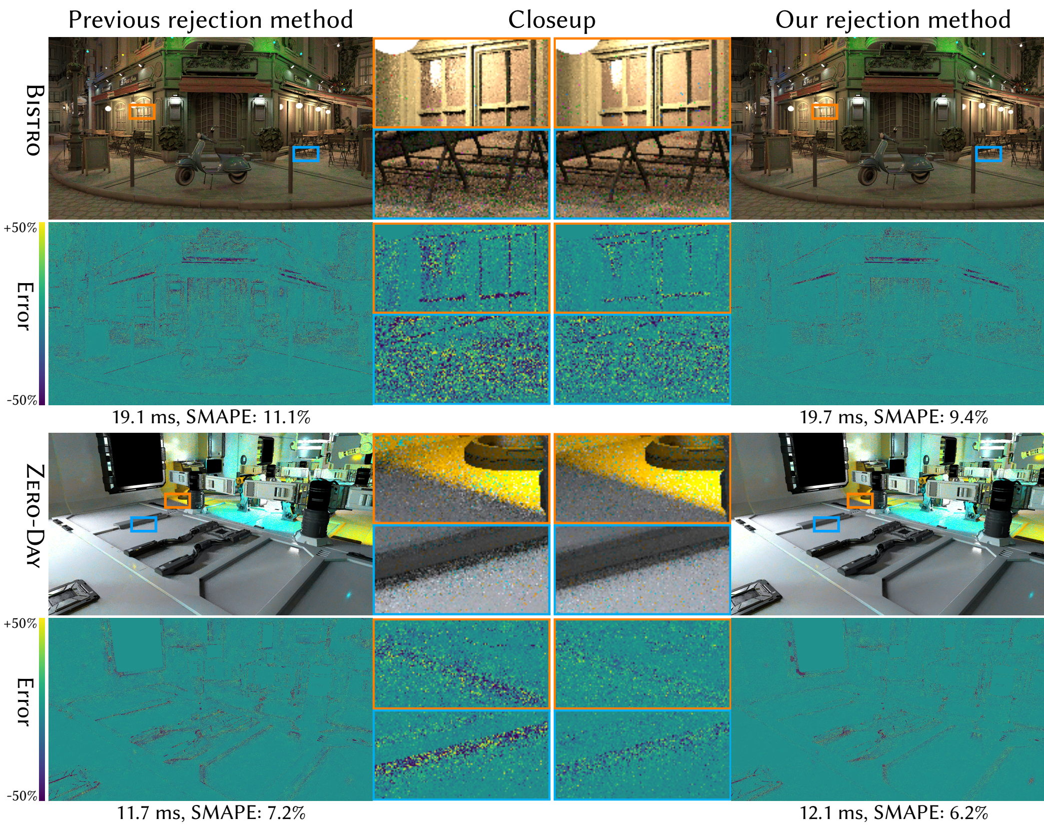


Fig. 10. ReSTIR with exact visibility test (5 rpp) for the Bistro scene (2.8 M triangles, 20.6 k triangle lights) and the Zero-Day scene (5.2 M triangles, 10.3 k triangle lights). Although our rejection method introduces a small bias, it significantly reduces variance around shadow edges and material boundaries.

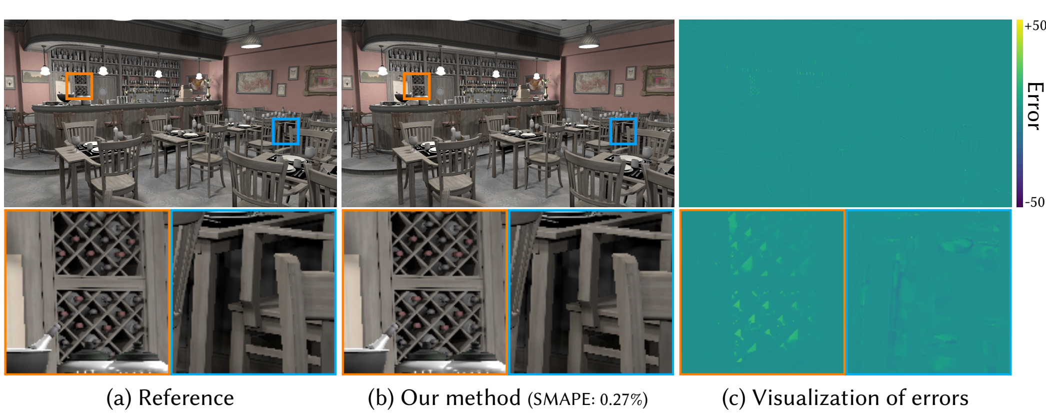


Fig. 11. ReSTIR with exact visibility test and our rejection method using 300000 samples per pixel for the Bistro Interior scene (1.3 M triangles, 44.0 k triangle lights). Our method produces a small bias in dark shadows as shown in the error visualization (c), but it is barely perceptible in the rendered image (b).

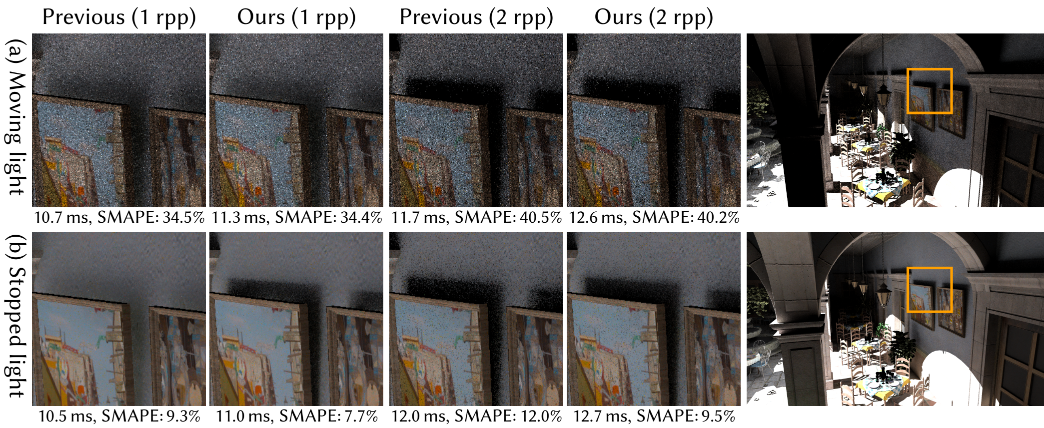


Fig. 12. Dynamic one-bounce indirect illumination from a point light source for the San Miguel scene (10 M triangles). (a) Our method produces comparable image quality to the previous method for a rapidly moving light source. (b) When the light source stops, our method renders higher-quality indirect shadows than the previous method.

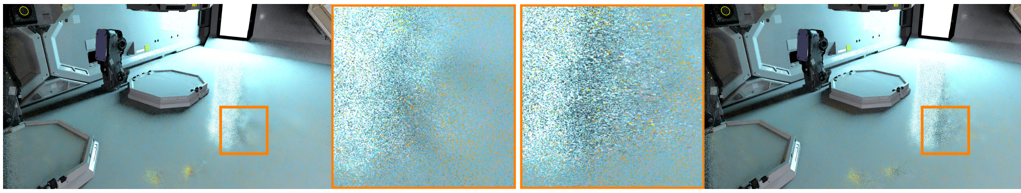
between our similarity computation and lighting estimation. However, the bias is imperceptibly small as shown in Fig. 11, since spatiotemporal resampling decorrelates the variance every frame.

Dynamic indirect illumination. Our method is applicable to indirect illumination as well as direct illumination by using VPLs. Fig. 12 shows dynamic one-bounce indirect illumination from a point light source. For this scene, we generate 1 M VPLs via a reflective shadow map [Dachsbacher and Stamminger 2005] of 1024×1024 resolution and use the same ReSTIR algorithm for illumination from these VPLs. When the light source moves rapidly, our method provides image quality comparable to the previous method's. On the other hand, when the light source stops, our method renders indirect shadows of higher quality than the previous method. The reason is that our rejection heuristic is effective only for static or slowly moving lights.

5 LIMITATIONS

Bias. Since the average-direction estimation for our PDF similarity shares the initial sample with lighting estimation, our rejection heuristic based on the PDF similarity can introduce a bias due to the correlation of samples. Although spatiotemporal resampling decorrelates samples in every frame, the proposed method is still an inconsistent estimator. This is because the accumulated candidate count is limited and initial samples are combined every frame in ReSTIR. However, the bias is negligibly small for temporally continuous pixels. Enabling our heuristic only for non-zero contribution weights also introduces a sample correlation if the expected value of the lighting integral is not zero. However, such zero weights are rare for temporally continuous pixels. We can decorrelate samples by using different initial samples, though this approach increases the computational overhead.

Temporal discontinuities. Our rejection heuristic works only for temporal continuities. Therefore, our rejection heuristic is not always effective in motion. Although our method renders high-frequency shadow edges using only one ray per pixel for static scenes, we have to trace an additional shadow ray to render shadows for animated scenes as in previous work. To preserve



(a) Previous method (7.52 ms, SMAPE: 17.3%) (b) Ours (7.72 ms, SMAPE: 16.9%)

Fig. 13. Previous rejection method (a) and our rejection method (b) for glossy surfaces viewed from a moving camera (2 rpp). While our method reduces error on shadow edges, it can increase variance around glossy highlights as the camera moves. This is because our PDF shape estimation has a temporal delay when the view direction changes for glossy surfaces. Reduction of this delay is left for future work.

shadow edges while using less than two rays per pixel, we use spatiotemporal adaptive ray tracing for animated scenes.

False positives. Our method can approximate different PDFs into an identical vMF lobes. In this case, our method cannot reject samples that should be rejected. However, this case does not occur often enough to be a problem in our experimental results.

Multiple bounces. Since our method uses the PDF similarity in a spherical domain, it does not support multiple bounces whose PDF is the product of spherical PDF sequences. Extension for multi-bounce illumination such as glossy-to-glossy interreflections is left for future work.

Memory overhead. Our average-direction estimation stores reservoirs in memory. Thus, it has a memory transfer cost. In our experimental implementation, we use 16 bytes per pixel for these reservoirs. We consider reduction of the reservoir data size as future work.

Highly glossy surfaces. Our PDF similarity estimation has a temporal delay for highly glossy surfaces when the view direction changes. Thus, this delay can produce variance around glossy highlights with a moving camera (Fig. 13). To reduce this delay, we can add an analytic glossy lobe similarity [Tokuyoshi 2015] between frames to the rejection heuristic (Eq. 12) in our average-direction estimation. Another approach to avoid the view-dependent delay is to decouple incoming radiance and the BSDF from the PDF. This decoupling approach separately approximates incoming radiance and the cosine-weighted BSDF using two vMFs, and then computes a vMF representing the PDF by using the product of the two vMFs as in spherical Gaussian lighting [Wang et al. 2009]. Since we can obtain the vMF for the BSDF analytically or using lookup tables, we can avoid the view-dependent delay for highly glossy surfaces while increasing the vMF approximation error. We would like to investigate the efficiency of these approaches in the future.

6 CONCLUSION

This paper has presented a new rejection method based on the PDF shape similarity between pixels for single-bounce ReSTIR (e.g., direct illumination). Using this PDF similarity, we alleviated undesirable spatial resampling across shadow edges and material boundaries. To perform at real-time frame rates, our method roughly approximates the PDF with a vMF by using the temporal average of sample light directions for each pixel. We have also presented a stable combination of an existing rejection heuristic and our PDF similarity considering the estimation error of the temporal average direction. Using our method, we improved the image quality for temporally continuous lighting while using a smaller number of rays than the previous method. On the other hand, our method is comparable quality to the existing heuristic for temporal disocclusions and rapidly moving lights.

Although our method takes into account the estimation error when lighting changes, it ignores error due to the temporal changes of view directions for highly glossy surfaces. To accurately handle such view-direction changes, we are currently considering the integration of a glossy lobe similarity [Tokuyoshi 2015] or decoupling of incoming radiance and BSDFs [Wang et al. 2009] into our method. We would like to investigate the efficiency of these techniques in the future. Our PDF similarity is limited to single bounce, but it is applicable to indirect illumination by using VPLs. We would also like to investigate the efficiency of our method for VPL-based ReSTIR algorithms such as ReSTIR GI [Ouyang et al. 2021].

ACKNOWLEDGMENTS

We would like to thank the Amazon Lumberyard team for the Bistro scene [2017], M. Winkelmann and K. Anderson for the Zero-Day scene [2019], and G. M. Leal Llaguno for the San Miguel scene. The San Miguel scene is distributed by McGuire [2017]. We would also like to the anonymous reviewers for their valuable comments and constructive suggestions.

REFERENCES

Arindam Banerjee, Inderjit S. Dhillon, Joydeep Ghosh, and Suvrit Sra. 2005. Clustering on the Unit Hypersphere Using von Mises-Fisher Distributions. *J. Mach. Learn. Res.* 6 (2005), 1345–1382.

Philippe Bekaert, Mateu Sbert, and Yves D. Willems. 2000. Weighted Importance Sampling Techniques for Monte Carlo Radiosity. In *EGWR '00*. 35–46.

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 Trans. Graph.* 39, 4, Article 148 (2020), 17 pages. https://doi.org/10.1145/3386569.3392481

Guillaume Boissé. 2021. World-Space Spatiotemporal Reservoir Reuse for Ray-Traced Global Illumination. In *SIGGRAPH Asia '21 Tech. Commun.* Article 22, 4 pages. https://doi.org/10.1145/3478512.3488613

Jakub Boksansky, Paula Jukarainen, and Chris Wyman. 2021. Rendering Many Lights with Grid-Based Reservoirs. In *Ray Tracing Gems II: Next Generation Real-Time Rendering with DXR, Vulkan, and OptiX*. Apress, 351–365. https://doi.org/10.1007/978-1-4842-7185-8_23

Min-Te Chao. 1982. A General Purpose Unequal Probability Sampling Plan. *Biometrika* 69, 3 (1982), 653–656. https://doi.org/10.1093/biomet/69.3.653

Carsten Dachsbacher and Marc Stamminger. 2005. Reflective Shadow Maps. In *I3D '05*. 203–231.

Elmar Eisemann and Frédo Durand. 2004. Flash photography enhancement via intrinsic relighting. *ACM Trans. Graph.* 23, 3 (2004), 673–678.

Ronald Aylmer Fisher. 1953. Dispersion on a sphere. *Proc. R. Soc. Lond. Ser. A* 217, 1130 (1953), 295–305. https://doi.org/10.1098/rspa.1953.0064

Eric Heitz, Stephen Hill, and Morgan McGuire. 2018. Combining Analytic Direct Illumination and Stochastic Shadows. In *I3D '18*. Article 2, 11 pages. https://doi.org/10.1145/3190834.3190852

Kei Iwasaki, Yoshinori Dobashi, and Tomoyuki Nishita. 2012. Interactive Bi-Scale Editing of Highly Glossy Materials. *ACM Trans. Graph.* 31, 6, Article 144 (2012), 7 pages. https://doi.org/10.1145/2366145.2366163

Alexander Keller. 1997. Instant Radiosity. In *SIGGRAPH '97*. 49–56. https://doi.org/10.1145/258734.258769

Daqi Lin, Markus Kettunen, Benedikt Bitterli, Jacopo Pantaleoni, Cem Yuksel, and Chris Wyman. 2022. Generalized Resampled Importance Sampling: Foundations of ReSTIR. *ACM Trans. Graph.* 41, 4, Article 75 (2022), 23 pages. https://doi.org/10.1145/3528223.3530158

Daqi Lin, Chris Wyman, and Cem Yuksel. 2021. Fast Volume Rendering with Spatiotemporal Reservoir Resampling. *ACM Trans. Graph.* 40, 6, Article 279 (2021), 18 pages. https://doi.org/10.1145/3478513.3480499

Amazon Lumberyard. 2017. Amazon Lumberyard Bistro, Open Research Content Archive (ORCA). http://developer.nvidia.com/orca/amazon-lumberyard-bistro

Morgan McGuire. 2017. Computer Graphics Archive. https://casual-effects.com/data

Yaobin Ouyang, Shiqiu Liu, Markus Kettunen, Matt Pharr, and Jacopo Pantaleoni. 2021. ReSTIR GI: Path Resampling for Real-Time Path Tracing. *Comput. Graph. Forum* 40, 8 (2021), 17–29. https://doi.org/10.1111/cgf.14378

Georg Petschnigg, Richard Szeliski, Maneesh Agrawala, Michael Cohen, Hugues Hoppe, and Kentaro Toyama. 2004. Digital photography with flash and no-flash image pairs. *ACM Trans. Graph.* 23, 3 (2004), 664–672.

Donald B. Rubin. 1987. A Noniterative Sampling/Importance Resampling Alternative to the Data Augmentation Algorithm for Creating a Few Imputations When Fractions of Missing Information Are Modest: The SIR Algorithm. *J. Amer. Statist. Assoc.* 82, 398 (1987), 543–546.

Rohan Sawhney, Daqi Lin, Markus Kettunen, Benedikt Bitterli, Ravi Ramamoorthi, Chris Wyman, and Matt Pharr. 2022. Decorrelating ReSTIR Samplers via MCMC Mutations. https://doi.org/10.48550/ARXIV.2211.00166

Justin F. Talbot. 2005. *Importance Resampling for Global Illumination.* Master's thesis. Brigham Young U.

Yusuke Tokuyoshi. 2015. Specular Lobe-Aware Filtering and Upsampling for Interactive Indirect Illumination. *Comput. Graph. Forum* 34, 6 (2015), 135–147. https://doi.org/10.1111/cgf.12525

Yusuke Tokuyoshi. 2022. *Tiled Reservoir Sampling for Many-Light Rendering.* Technical Report No. 21-11-ecdc. Advanced Micro Devices, Inc.

Yu-Ting Tsai and Zen-Chung Shih. 2006. All-Frequency Precomputed Radiance Transfer Using Spherical Radial Basis Functions and Clustered Tensor Approximation. *ACM Trans. Graph.* 25, 3 (2006), 967–976. https://doi.org/10.1145/1141911.1141981

Eric Veach. 1998. *Robust Monte Carlo Methods for Light Transport Simulation.* Ph. D. Dissertation. Stanford U.

Eric Veach and Leonidas J. Guibas. 1995. Optimally Combining Sampling Techniques for Monte Carlo Rendering. In *SIGGRAPH '95.* 419–428. https://doi.org/10.1145/218380.218498

Jiaping Wang, Peiran Ren, Minmin Gong, John Snyder, and Baining Guo. 2009. All-Frequency Rendering of Dynamic, Spatially-Varying Reflectance. *ACM Trans. Graph.* 28, 5 (2009), 133:1–133:10. https://doi.org/10.1145/1618452.1618479

Mike Winkelmann and Kate Anderson. 2019. Zero-Day, Open Research Content Archive (ORCA). https://developer.nvidia.com/orca/beeple-zero-day

Chris Wyman and Alexey Panteleev. 2021. Rearchitecting Spatiotemporal Resampling for Production. In *HPG '21.* 19 pages. https://doi.org/10.2312/hpg.20211281

A MIS WEIGHTS USED IN THIS PAPER

In this paper, we reuse one sample at a neighboring pixel i and combine it into the sample at the current pixel s in each resampling routine as in an existing practical implementation [Wyman and Panteleev 2021]. For this case, we weight target distributions by using accumulated candidate counts M_s and M_i. In our implementation for area light sources, we use the following equation:

$$\begin{aligned} m_s(x_s) &= \frac{M_s \hat{p}_{s\to s}(x_s)}{M_s \hat{p}_{s\to s}(x_s) + M_i \hat{p}_{s\to i}(x_s)} \\ &= \frac{M_s \hat{p}_s(\omega_{s,s}) \frac{|\mathbf{n}(\mathbf{y}_{s,s})\cdot\omega_{s,s}|}{\|\mathbf{y}_{s,s}-\mathbf{z}_s\|^2}}{M_s \hat{p}_s(\omega_{s,s}) \frac{|\mathbf{n}(\mathbf{y}_{s,s})\cdot\omega_{s,s}|}{\|\mathbf{y}_{s,s}-\mathbf{z}_s\|^2} + M_i \hat{p}_i(\omega_{i,s}) \frac{|\mathbf{n}(\mathbf{y}_{i,s})\cdot\omega_{i,s}|}{\|\mathbf{y}_{i,s}-\mathbf{z}_i\|^2}}, \end{aligned} \tag{17}$$

$$\begin{aligned} m_i(T_i(x_i)) &= \frac{M_i \hat{p}_{s\to i}(T_i(x_i))}{M_s \hat{p}_{s\to s}(T_i(x_i)) + M_i \hat{p}_{s\to i}(T_i(x_i))} \\ &= \frac{M_i \hat{p}_i(\omega_{i,i}) \frac{|\mathbf{n}(\mathbf{y}_{i,i})\cdot\omega_{i,i}|}{\|\mathbf{y}_{i,i}-\mathbf{z}_i\|^2}}{M_s \hat{p}_s(\omega_{s,i}) \frac{|\mathbf{n}(\mathbf{y}_{s,i})\cdot\omega_{s,i}|}{\|\mathbf{y}_{s,i}-\mathbf{z}_s\|^2} + M_i \hat{p}_i(\omega_{i,i}) \frac{|\mathbf{n}(\mathbf{y}_{i,i})\cdot\omega_{i,i}|}{\|\mathbf{y}_{i,i}-\mathbf{z}_i\|^2}}, \end{aligned} \tag{18}$$

where the spherical target distribution $\hat{p}_s(\cdot)$ is the product of the incoming radiance and the cosine-weighted BSDF at $\mathbf{z}_s$ as follows:

$$\hat{p}_s(\omega) = L(\mathbf{z}_s, \omega)\, \rho\left(\mathbf{z}_s, \frac{\mathbf{c}_s - \mathbf{z}_s}{\|\mathbf{c}_s - \mathbf{z}_s\|}, \omega\right) |\mathbf{n}(\mathbf{z}_s)\cdot\omega|\,.$$