Pacific Graphics (2016)
E. Grinspun, B. Bickel, and Y. Dobashi (Editors)

Modified Filtered Importance Sampling for Virtual Spherical Gaussian Lights

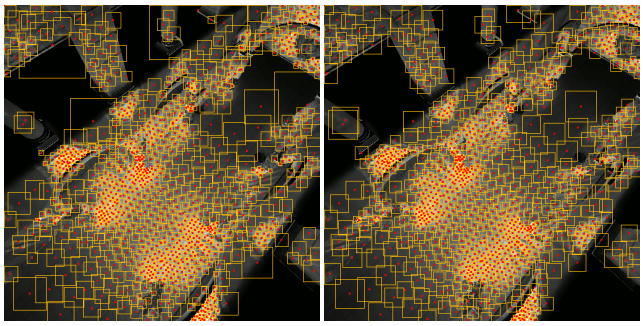
Yusuke Tokuyoshi†

Square Enix Co., Ltd., Japan

Abstract

This paper proposes a modification of the filtered importance sampling (FIS) method, and improves the quality of virtual spherical Gaussian light (VSGL) based real-time glossy indirect illumination using this modification. The original FIS method produces large overlaps of and gaps between filtering kernels for high-frequency probability density functions (PDFs). This is because the size of the filtering kernel is determined using the PDF at the sampled center of the kernel. To reduce those overlaps and gaps, this paper determines the kernel size using the integral of the PDF in the filtering kernel. Our key insight is that these integrals are approximately constant, if kernel centers are sampled using stratified sampling. Therefore, an appropriate kernel size can be obtained by solving this integral equation. Using the proposed kernel size for FIS-based VSGL generation, undesirable artifacts are significantly reduced with a negligibly small overhead.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture



(a) Previous filtering kernels (b) Our filtering kernels

Figure 1: *Sampling VPL clusters from an RSM based on FIS. Red points: kernel centers sampled according to the PDF (brightness). Orange squares: filtering kernels (i.e., VPL clusters). The previous FIS (a) produces significant overlaps of and gaps between filtering kernels. Our method (b) reduces these overlaps and gaps.*

1. Introduction

The filtered importance sampling (FIS) method [KC08] is a variance reduction technique of Monte Carlo integration often used for real-time or interactive rendering, which uses filtering kernels instead of sample points. This paper proposes a modification of FIS, and improves the quality of virtual spherical Gaussian light (VSGL) [Tok15] based real-time glossy indirect illumination using this modification. The original FIS first samples the center of each filtering kernel according to a probability density function (PDF), and then determines the size of each filtering kernel using the PDF at the sampled center. However, this kernel size determination produces large overlaps of and gaps between filtering kernels for high-frequency PDFs (Fig. 1). This is because the kernel size can be too large when the sampled center is at a local minimum of a high-frequency PDF. Therefore, this paper introduces an appropriate FIS kernel size to reduce these overlaps and gaps.

One effective application of our method is generation of VSGLs using reflective shadow maps (RSMs) [DS05]. RSM-based global illumination is well established for real-time rendering. However, stochastic sampling of virtual point lights (VPLs) [Kel97] (i.e., texels of RSMs which represent one-bounce light subpaths) produces noticeable variance especially for glossy interreflections. To reduce this variance, VSGLs were introduced recently. This method approximates a cluster of VPLs using a Gaussian-based representation. Thanks to this representation, the distribution of VPLs can be filtered with a simple summation operation (e.g., mipmaping). In addition, this representation has an analytical solution of the rendering integral for each VSGL. Therefore, if VSGLs are generated from RSMs inexpensively, we are able to render one-bounce glossy indirect illumination at real-time frame rates.

Tokuyoshi [Tok15] sampled VPL clusters as VSGLs from an

† tokuyosh@square-enix.com

RSM based on FIS to achieve real-time frame rates. However, while this approach is simpler and faster than k-means-based VPL clustering [DGR*09, PKD12], it does induce flickering structured artifacts due to the previously mentioned overlaps and gaps. This problem is noticeable when a bidirectional reflective shadow mapping (BRSM) method [REH*11] is used to build the PDF. This is because the BRSM method produces a dynamic and high-frequency PDF. Using our kernel size, we are able to reduce flickering artifacts significantly for such a high-frequency PDF.

The contributions of our work are as follows:

- An appropriate kernel size of FIS is introduced to reduce undesirable overlaps of and gaps between filtering kernels.
- For image-based PDFs, the above kernel size is computed using a simple numerical approach with negligibly small overhead.
- Using the proposed FIS method, flickering artifacts are reduced for VSGL-based real-time glossy indirect illumination.

2. Related Work

Sampling pre-integrated values is often used for image-based lighting. Structured importance sampling [ARBJ03] stratifies samples hierarchically, and then the illumination is pre-integrated within each stratum. FIS [KC08] is introduced for glossy materials under environment maps. This technique samples pre-filtered value using a mipmap, thus it performs at real-time frame rates. For interactive indirect illumination, VPLs are often clustered and approximated using area lights such as Prutkin et al. [PKD12]. VSGLs [Tok15] were introduced to approximate a set of VPLs using a Gaussian and spherical Gaussians (SGs) [TS06] for glossy interreflections. To generate thousands of VSGLs at real-time frame rates, an FIS-based approach was used with mipmapped RSMs. However, this VSGL generation induces a flickering error for high-frequency PDFs due to inappropriate kernel sizes. In this paper, we introduce an appropriate FIS kernel size.

3. Modified Filtered Importance Sampling

FIS can be used when the integrand has a 2D image $f(\mathbf{x})$, where $\mathbf{x} \in [0,1]^2$ is the image-space position. This method first samples each kernel center $\mathbf{x}_i \in [0,1]^2$ according to a PDF $p(\mathbf{x})$, and then a filtered value of $f(\mathbf{x})$ is used as each sample value instead of $f(\mathbf{x}_i)$. This filtered value is given by a pre-filtered mipmap as follows:

$$\int_{[0,1]^2} f(\mathbf{x}) \frac{g((\mathbf{x}-\mathbf{x}_i)/s_i)}{a_i} \mathrm{d}\mathbf{x} \approx \bar{f}(\mathbf{x}_i, l_i),$$

where $g((\mathbf{x}-\mathbf{x}_i)/s_i)$ is the unnormalized filtering kernel which has a fixed maximum, s_i is the kernel size, a_i is the filtering area (i.e., normalization factor) given by $a_i = \int_{[0,1]^2} g((\mathbf{x}-\mathbf{x}_i)/s_i)\mathrm{d}\mathbf{x}$, and $\bar{f}(\mathbf{x}_i, l_i)$ is the mipmapped value of $f(\mathbf{x})$ at mip level l_i. Let M be the number of texels of $f(\mathbf{x})$, then the filtering area a_i is also written as a function of mip level l_i: $a_i = \frac{4^{l_i}}{M}$. Křivánek et al. [KC08] determined mip level l_i by representing this filtering area using the inverse of the density at sampled center $\mathbf{x}_i$ as

$$a_i = \frac{4^{l_i}}{M} = \min\left(\frac{K}{N p(\mathbf{x}_i)}, 1\right), \tag{1}$$

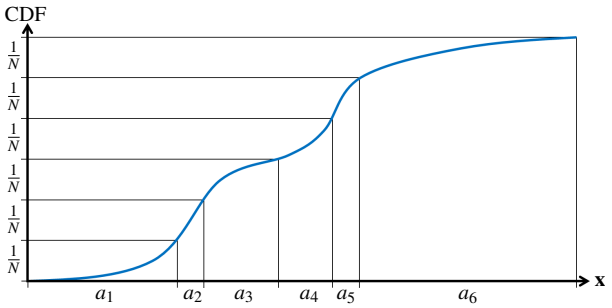


Figure 2: *CDF (blue line) and stratified sampled kernels.*

where N is the number of samples, and K is a user-specified parameter to tweak the kernel size (Křivánek et al. used $K = 4$). However, this mip level determination is sensitive to the sampled center $\mathbf{x}_i$. When $\mathbf{x}_i$ is at a local minimum of a high-frequency PDF, the filtering kernel can be too large. Conversely, the filtering kernel can also be too small when $\mathbf{x}_i$ is at a local maximum. Therefore, undesirable overlaps of and gaps between filtering kernels can be produced.

Our filtering kernels. This paper introduces an appropriate kernel size to reduce overlaps of and gaps between filtering kernels for FIS. Sampling according to a PDF is done by computing the inverse cumulative distribution function (CDF) of the PDF. As shown in Fig. 2, a sampling interval of the vertical axis is the integral of the PDF in each filtering kernel. Therefore, if kernel centers are sampled using stratified sampling, this integral is almost $\frac{1}{N}$. Hence, an appropriate kernel size s_i is obtained by solving the following integral equation:

$$\int_{[0,1]^2} p(\mathbf{x}) g((\mathbf{x}-\mathbf{x}_i)/s_i)\mathrm{d}\mathbf{x} = \frac{1}{N}. \tag{2}$$

Since the left side is monotonically increasing with respect to the kernel size, we can obtain the kernel size using a bisection method. When PDF $p(\mathbf{x})$ is given by a 2D image, we can use the mipmap of the PDF, which is also used for sampling $\mathbf{x}_i$ via hierarchical sample warping [CJAMJ05]. Using this mipmap, Eq. 2 is rewritten as

$$\boxed{\frac{4^{l_i}}{M}\bar{p}(\mathbf{x}_i, l_i) = \frac{1}{N}}, \tag{3}$$

where $\bar{p}(\mathbf{x}_i, l_i) \approx \int_{[0,1]^2} p(\mathbf{x}) \frac{g((\mathbf{x}-\mathbf{x}_i)/s_i)}{a_i}\mathrm{d}\mathbf{x}$ is the mipmapped value of $p(\mathbf{x})$. In this paper, l_i is calculated using the bisection method with an iteration count of 12.

4. Application to Virtual Spherical Gaussian Lights (VSGLs)

In this paper, we demonstrate generation of VSGLs as an effective application of our filtered sampling. A VSGL represents the positional distribution and total radiant intensity of VPLs using a Gaussian and SGs, respectively. Since SGs have closed-form solutions to evaluate rendering integrals, all-frequency illumination is computed analytically for each VSGL. The VSGL algorithm is composed of the following five phases: RSM rendering, PDF building, VSGL generation, shadow map rendering, and shading. This paper improves only on the VSGL generation phase.

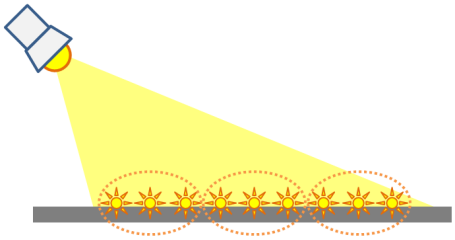


Figure 3: *Clustered VPLs. Each cluster is approximated with a VSGL by computing the total VPL power and averaged VPL distributions within the cluster. These operations are done by filtered sampling on the RSM.*

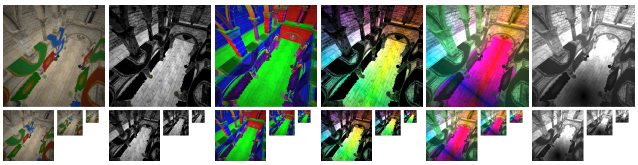


Figure 4: *Mipmapped RSM for VSGL generation. Average emission directions (c)(d) and positions (e)(f) are weighted by VPL powers (a)(b). VSGLs are sampled from this RSM based on FIS.*

Mipmap-based VSGL generation. To generate VSGLs, VPLs are first clustered. Then VPL powers are summed and VPL distributions are averaged for each cluster (Fig. 3). To represent VPL distributions with a Gaussian and SGs, weighted averages of emission directions, VPL positions, and squared VPL positions weighted by each VPL power are required (for the detail, please refer to the supplemental material). Therefore, an RSM to store the above VPL power and weighted distribution parameters is generated, and then they are mipmapped to approximately obtain total texel values (Fig. 4). Let $f(\mathbf{x})$ be the RSM, then the total texel value in ith VPL cluster centered at $\mathbf{x}_i$ is approximated using the mipmap $\bar{f}(\mathbf{x}_i, l_i)$ as follows:

$$\int_{[0,1]^2} f(\mathbf{x}) g((\mathbf{x}-\mathbf{x}_i)/s_i)\mathrm{d}\mathbf{x} \approx \frac{4^{l_i}}{M}\bar{f}(\mathbf{x}_i, l_i),$$

where filtering kernel $g((\mathbf{x}-\mathbf{x}_i)/s_i)$ represents the VPL cluster. To sample the kernel center $\mathbf{x}_i$ and mip level l_i, an FIS-based approach can be used. The kernel center $\mathbf{x}_i$ is sampled according to a dynamic and high-frequency view-dependent PDF $p(\mathbf{x})$ given by the BRSM method. Tokuyoshi [Tok15] determined l_i using Eq. 1 with $K = 1$ according to the previous FIS. Using this mip level determination, the total RSM texel value in each cluster is given by

$$\int_{[0,1]^2} f(\mathbf{x}) g((\mathbf{x}-\mathbf{x}_i)/s_i)\mathrm{d}\mathbf{x} \approx \frac{\bar{f}(\mathbf{x}_i, l_i)}{\max\left(Np(\mathbf{x}_i), 1\right)}. \tag{4}$$

However, since the numerator is filtered while the denominator is not, this sampling method can induce an intensive error due to overlaps of and gaps between filtering kernels.

VSGL generation using our filtering kernels. To obtain an appropriate mip level l_i, this paper employs Eq. 3 instead of Eq. 1 for FIS-based VSGL generation. Using this mip level l_i, the total texel value in each cluster is approximated as follows:

$$\boxed{\int_{[0,1]^2} f(\mathbf{x}) g((\mathbf{x}-\mathbf{x}_i)/s_i)\mathrm{d}\mathbf{x} \approx \frac{\bar{f}(\mathbf{x}_i, l_i)}{N\bar{p}(\mathbf{x}_i, l_i)}}. \tag{5}$$

Unlike Eq. 4, both the numerator and denominator are filtered using the same kernel. Hence, temporal coherence is improved for a dynamic high-frequency PDF. Furthermore, the approximation error can be reduced if PDF $p(\mathbf{x})$ is approximately proportional to $f(\mathbf{x})$, similar to standard importance sampling.

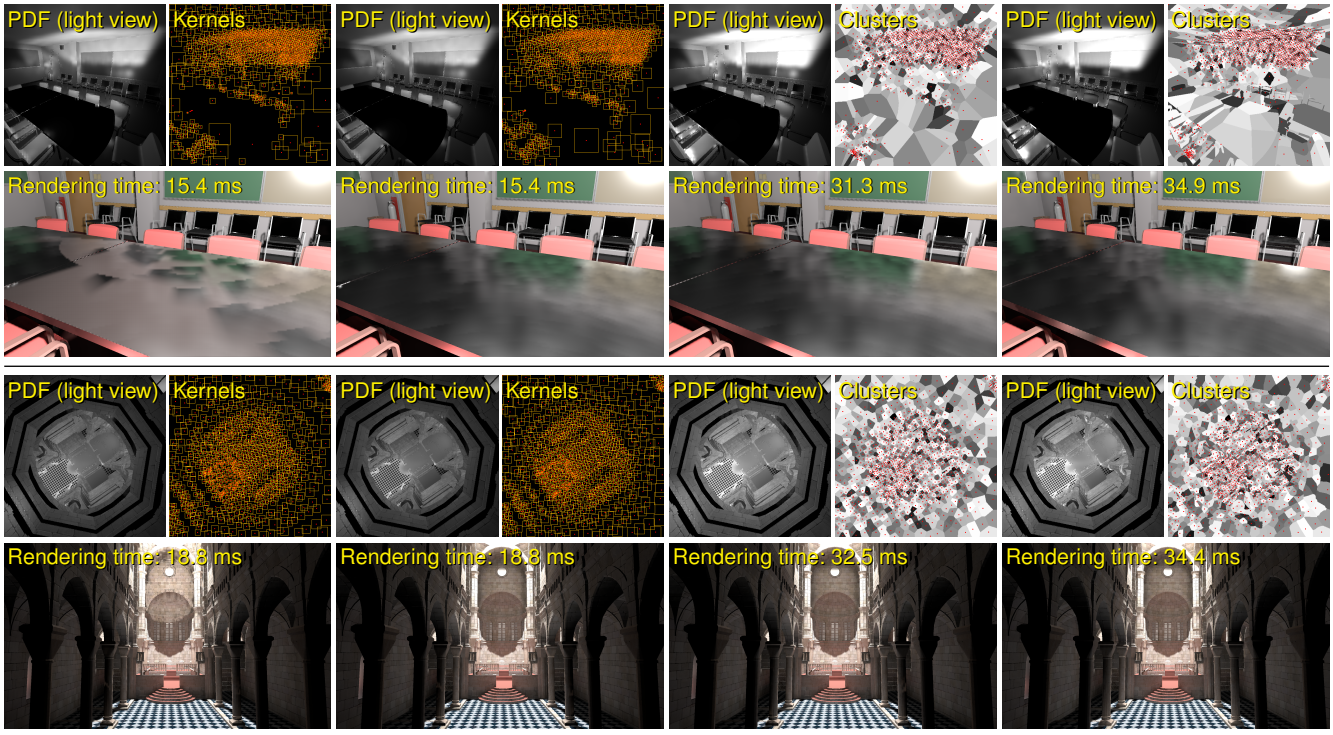
Controlling the kernel size. For Eq. 5, the mip level l_i affects only the filter bandwidth. Therefore, the user-specified parameter K can also be used for calculating l_i in our case. This is implemented using $\frac{K}{N}$ instead of $\frac{1}{N}$ in Eq. 3. Using $K > 1$, the temporal coherence is improved, though an overblurring error is induced. This overblurring error is reduced by increasing the number of samples N, similar to the original FIS.

5. Experimental Results

Here we present rendering results using 1024 VSGLs generated using our FIS with $K = 1$ on an NVIDIA® GeForce® GTX™ 970 GPU. The frame buffer and RSM resolutions are 1920×1088 and 512^2, respectively. A tessellation-based imperfect shadow map [BBH13] of resolution 64^2 is employed to evaluate the visibility of each VSGL. To estimate a view-dependent PDF on the RSM using the BRSM method, 2048 VSGLs without shadow maps are generated on the G-buffer. For the PDF on the G-buffer, reflectance is used. To perform stratified sampling, the Fibonacci lattice point set using a golden ratio approximation [SJ94] is employed as a quasi-random number. For comparison, this paper uses k-means clustering using 2D image space and 3D world space. In these k-means-based approaches, once clusters are assigned to all the texels, those texels are sorted by cluster ID. Then, to compute the total value of clustered texels, a thread is dispatched for each cluster similar to Prutkin et al. [PKD12]. For implementation details, please refer to the supplemental material.

Quality. Fig. 5 shows rendered images using different VSGL generation methods. Using the previous kernel size (a), intense artifacts can be produced with low probability, though this sampling method is faster than the k-means-based approaches (c)(d). This is because too large of a filtering kernel is produced when the sampled kernel center is at a local minimum of the PDF. On the other hand, our kernel size (b) does not produce these undesirable filtering kernels nor does it noticeably sacrifice performance.

Performance. Table 1 shows the computation times of VSGL generation both for BRSM (upper row) and final shading (lower row). Our contribution is written in red. Although our method is a numerical approach, its overhead is a total of about five microseconds more compared to the previous FIS-based generation. In addition, our method is about 7-9 times faster than the k-means-based approaches. The difference is significant especially for the BRSM method, which uses a higher-resolution G-buffer (1920×1088) than the RSM (512^2). Table 2 shows the computation times using different PDFs. For these PDFs, the performance of k-means-based approaches is more expensive than Table 1. This is because



(a) [Tok15] (b) Ours (c) k-means (2D image space) (d) k-means (3D world space)

Figure 5: *Rendered images using different VSGL generation methods for 331k triangles scene (upper row) and 75k triangles scene (lower row). When a sampled kernel center is at a local minimum of the PDF, the previous method (a) produces too bright of a VSGL with low probability. On the other hand, our method (b) does not produce such an error similar to k-means-based approaches (c)(d). Aliasing artifacts on the glossy table in the upper row are the shadow acne of imperfect shadow maps.*

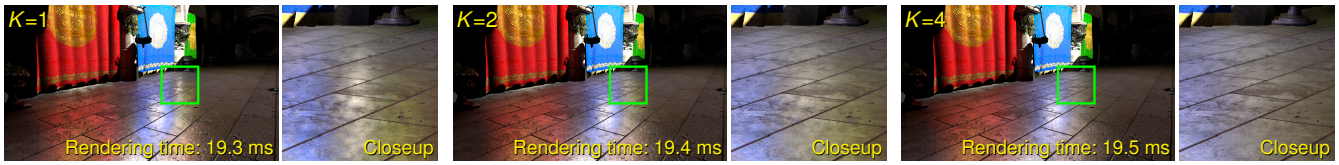
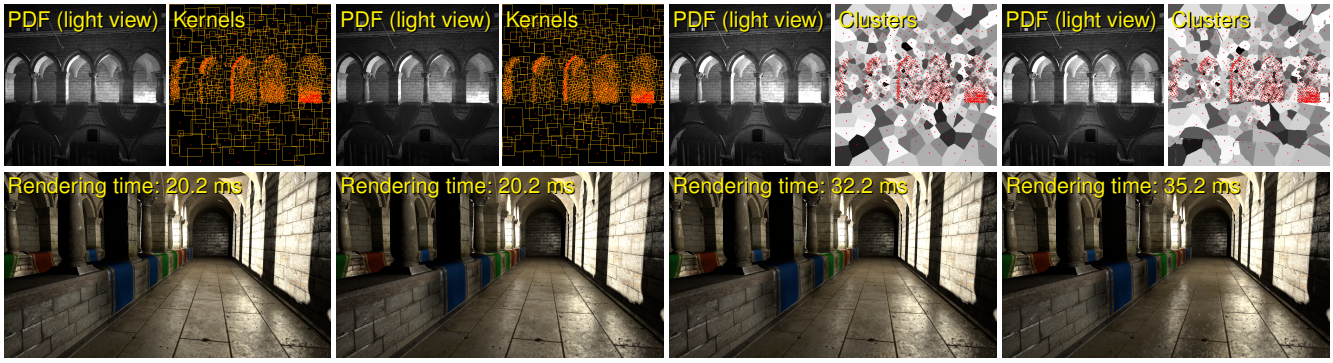


Figure 6: *Unlike k-means-based approaches, our approach can control the kernel size using the user-specified parameter K. By increasing K, the temporal coherence is improved, while some illumination appearance is overblurred (262k triangles scene).*



(a) [Tok15] (b) Ours (c) k-means (2D image space) (d) k-means (3D world space)

Figure 7: *Light occluded by columns (262k triangles scene). Since FIS-based approaches (a)(b) ignore the difference of world space positions, they blur some indirect illumination for this scene similar to the k-means-based approach using image space (c). These low-frequency errors are visually acceptable compared to high-frequency artifacts.*

Table 1: *Computation times of VSGL generation (ms).*

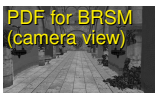
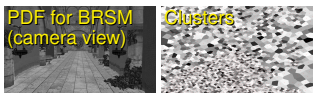
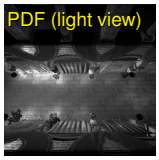
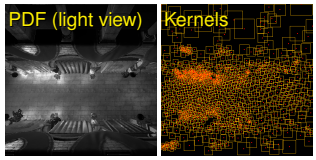
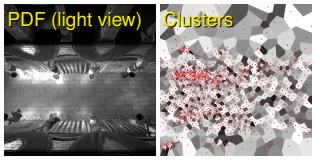
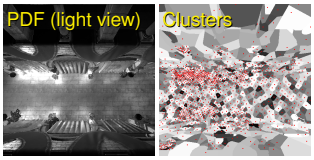
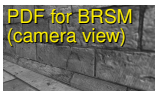
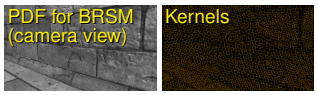
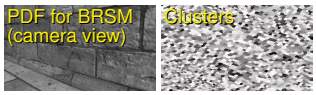
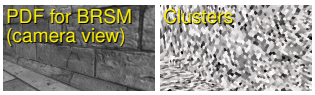
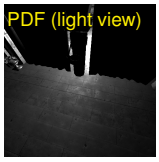
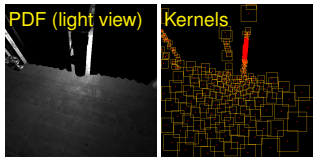
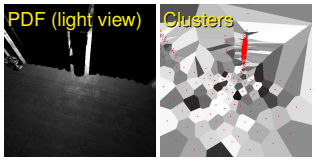
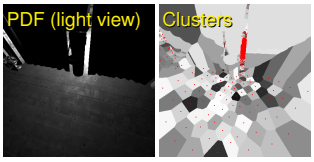
[Tok15]		Ours		k-means (2D image space)		k-means (3D world space)	
							
Additional G-buffer:	0.816	Additional G-buffer:	0.816	Cluster assignment:	3.004	Cluster assignment:	5.009
Mipmapping:	0.968	Mipmapping:	0.968	Sort:	1.541	Sort:	1.517
Filtered sampling:	0.090	Filtered sampling:	0.092	Sum:	9.096	Sum:	9.856
							
Additional RSM:	0.165	Additional RSM:	0.165	Cluster assignment:	0.619	Cluster assignment:	0.860
Mipmapping:	0.201	Mipmapping:	0.201	Sort:	0.306	Sort:	0.299
Filtered sampling:	0.084	Filtered sampling:	0.088	Sum:	1.995	Sum:	4.231
Total:	2.324	Total:	2.330	Total:	16.561	Total:	21.772

Table 2: *Computation times of VSGL generation (ms).*

[Tok15]		Ours		k-means (2D image space)		k-means (3D world space)	
							
Additional G-buffer:	0.752	Additional G-buffer:	0.752	Cluster assignment:	3.262	Cluster assignment:	5.756
Mipmapping:	0.985	Mipmapping:	0.985	Sort:	1.523	Sort:	1.499
Filtered sampling:	0.087	Filtered sampling:	0.089	Sum:	8.850	Sum:	8.735
							
Additional RSM:	0.154	Additional RSM:	0.154	Cluster assignment:	1.172	Cluster assignment:	0.899
Mipmapping:	0.199	Mipmapping:	0.199	Sort:	0.306	Sort:	0.305
Filtered sampling:	0.065	Filtered sampling:	0.067	Sum:	5.275	Sum:	20.489
Total:	2.242	Total:	2.246	Total:	20.388	Total:	37.683

the last pass "Sum" (which is the summation of texel values based on Prutkin et al.'s implementation) has a linear complexity with respect to the number of texels in a cluster. Comparatively, the performance of our approach is almost independent of the PDFs, because it uses pre-filtered mipmaps. Hence, the proposed method is suitable for applications which require stable performance.

Code size. Table 3 shows the code size of VSGL generation in our implementation. FIS-based approaches use only two compute shaders. One is for the calculation of the additional RSM (or G-buffer), and the other is for the filtered sampling of VSGLs. The difference of our method from the previous method [Tok15] is only the mip level determination (10 lines of code). On the other hand, the k-means-based approaches require more compute shaders than ours. In addition, some of them are dispatched iteratively for the GPU sort. Our method is about five times fewer lines of code than the k-means-based approaches.

Table 3: *Code size of VSGL generation (C++ and HLSL)*

	[Tok15]	Ours	k-means (2D)	k-means (3D)
# of shaders	2	2	9	9
# of dispatch calls	2	2	39	39
Lines of code	222	232	1143	1172

Kernel size controlling. As shown in Fig. 6, the kernel size of our method is controllable by using the user-specified parameter K unlike k-means-based approaches. Although some illumination appearance is overblurred by using $K > 1$, the temporal coherence is improved. The parameter K can be used to balance illumination details and temporal coherence according to the liking of a user.

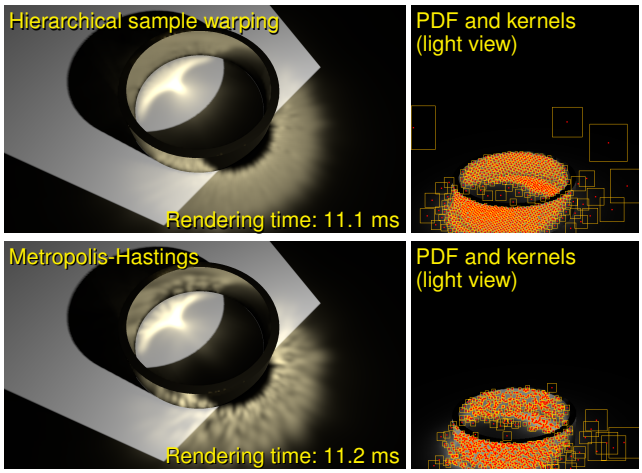


Figure 8: *Caustics rendered using our method with the same PDF (514 triangles scene). Kernel centers of the upper row and lower row are generated using hierarchical sample warping [CJAMJ05] and the Metropolis-Hastings-based temporally coherent sampling [BBH13], respectively. Due to a lack of stratification, Metropolis-Hastings produces noticeable artifacts.*

6. Limitations

Feature space. As shown in Fig. 7, FIS-based approaches (a)(b) ignore the difference of world space positions similar to the k-means-based-approach using image space (c). If VPLs are clustered ignoring such high-dimensional features, some indirect illumination is blurred when using VSGLs. These low-frequency errors are a limitation of FIS-based VSGL generation to achieve real-time frame rates, but they are more visually acceptable than high-frequency artifacts (e.g., flickering and spiky artifacts).

Overlaps and gaps. Although our method reduces overlaps of and gaps between filtering kernels, they cannot be removed completely for inhomogeneous sample distributions. This problem is alleviated by using stratified sampling.

PDF. Since our method requires a given PDF, it cannot be applied to sampling strategies without the PDF (e.g., sequential Monte Carlo instant radiosity [HKL16]).

Temporal coherence. Since our method improves only the kernel size, kernel centers can still be temporally incoherent for dynamic PDFs. Although the Metropolis-Hastings algorithm can be used for temporally coherent sampling [BBH13], it is limited to static light sources and has a lack of stratification. This problem induces noticeable artifacts especially for caustics (Fig. 8). Therefore, this paper employs hierarchical sample warping for stratified sampling. If the temporal coherence is more important than detailed illumination, $K > 1$ can be used for our method to improve the temporal coherence.

7. Conclusions

This paper improved the kernel size of FIS to reduce overlaps of and gaps between filtering kernels. Using this modification for VSGL generation, we are able to render glossy indirect illumination with fewer artifacts than the previous VSGL generation. The overhead of our method is about 5 microseconds for thousands of VSGLs on a commodity GPU. Although the FIS-based approach cannot take into account the difference of higher-dimensional features (e.g., world position) unlike k-means-based approaches, it is simple, fast, and has stable performance. This paper has demonstrated VSGL-based dynamic glossy indirect illumination, but our method is also usable for SG light generation for dynamic environment maps. Since environment maps are 2D light distribution, it might be more suitable than VSGL generation. We would like to investigate its efficiency in the future.

Acknowledgements

The polygon models are courtesy of M. Dabrovic, F. Meinl, A. Grynberg and G. Ward. The authors would like to thank the anonymous reviewers for valuable comments and helpful suggestions.

References

[ARBJ03] AGARWAL S., RAMAMOORTHI R., BELONGIE S., JENSEN H. W.: Structured importance sampling of environment maps. *ACM Trans. Graph. 22*, 3 (2003), 605–612. 2

[BBH13] BARÁK T., BITTNER J., HAVRAN V.: Temporally coherent adaptive sampling for imperfect shadow maps. *Comput. Graph. Forum 32*, 4 (2013), 87–96. 3, 6

[CJAMJ05] CLARBERG P., JAROSZ W., AKENINE-MÖLLER T., JENSEN H. W.: Wavelet importance sampling: Efficiently evaluating products of complex functions. *ACM Trans. Graph. 24*, 3 (2005), 1166–1175. 2, 6

[DGR*09] DONG Z., GROSCH T., RITSCHEL T., KAUTZ J., SEIDEL H.-P.: Real-time indirect illumination with clustered visibility. In *VMV'09* (2009), pp. 187–196. 2

[DS05] DACHSBACHER C., STAMMINGER M.: Reflective shadow maps. In *I3D'05* (2005), pp. 203–231. 1

[HKL16] HEDMAN P., KARRAS T., LEHTINEN J.: Sequential monte carlo instant radiosity. In *I3D'16* (2016), pp. 121–128. 6

[KC08] KŘIVÁNEK J., COLBERT M.: Real-time shading with filtered importance sampling. *Comput. Graph. Forum 27*, 4 (2008), 1147–1154. 1, 2

[Kel97] KELLER A.: Instant radiosity. In *SIGGRAPH'97* (1997), pp. 49–56. 1

[PKD12] PRUTKIN R., KAPLANYAN A. S., DACHSBACHER C.: Reflective shadow map clustering for real-time global illumination. In *EG'12 Short Papers* (2012), pp. 9–12. 2, 3

[REH*11] RITSCHEL T., EISEMANN E., HA I., KIM J. D., SEIDEL H.-P.: Making imperfect shadow maps view-adaptive: High-quality global illumination in large dynamic scenes. *Comput. Graph. Forum 30*, 8 (2011), 2258–2269. 2

[SJ94] SLOAN I. H., JOE S.: *Lattice Methods for Multiple Integration*. Clarendon Press, Oxford, 1994. 3

[Tok15] TOKUYOSHI Y.: Virtual spherical Gaussian lights for real-time glossy indirect illumination. *Comput. Graph. Forum 34*, 7 (2015), 89–98. 1, 2, 3, 4, 5

[TS06] TSAI Y.-T., SHIH Z.-C.: All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation. *ACM Trans. Graph. 25*, 3 (2006), 967–976. 2