Pacific Graphics (2016), pp. 1–4
E. Grinspun, B. Bickel, and Y. Dobashi (Editors)

Modified Filtered Importance Sampling for Virtual Spherical Gaussian Lights (Supplemental Material)

Yusuke Tokuyoshi†

Square Enix Co., Ltd., Japan

1. Introduction

This supplemental material explains virtual spherical Gaussian lights (VSGLs) [Tok15b] which is an effective application of our filtered importance sampling (FIS).

2. Spherical Gaussians

A spherical Gaussian (SG) is a type of spherical function and is represented using a Gaussian function γ with respect to a direction vector $\boldsymbol{\omega} \in \mathrm{S}^2$ as follows:

$$G(\boldsymbol{\omega},\boldsymbol{\xi},\lambda) = \gamma\left(\|\boldsymbol{\omega}-\boldsymbol{\xi}\|, \frac{1}{\lambda}\right) = \mathrm{e}^{-\frac{\lambda}{2}\|\boldsymbol{\omega}-\boldsymbol{\xi}\|^2} = \mathrm{e}^{\lambda((\boldsymbol{\omega}\cdot\boldsymbol{\xi})-1)},$$

where $\boldsymbol{\xi} \in \mathrm{S}^2$ is the lobe axis, and λ is the lobe sharpness. $\boldsymbol{\xi}$ and $\frac{1}{\lambda}$ correspond to the mean and variance for the Gaussian function, respectively. The integral of an SG is given by

$$A(\lambda) = \int_{\mathrm{S}^2} G(\boldsymbol{\omega},\boldsymbol{\xi},\lambda)\mathrm{d}\boldsymbol{\omega} = \frac{2\pi}{\lambda}\left(1-\mathrm{e}^{-2\lambda}\right).$$

A normalized SG $\frac{G(\boldsymbol{\omega},\boldsymbol{\xi},\lambda)}{A(\lambda)}$ is known as the Von Mises-Fisher distribution. For VSGLs, this distribution is used for representing reflection lobes.

2.1. SG approximation of reflection lobes

Diffuse lobes. For the Lambert bidirectional reflectance distribution function (BRDF) ρ_d, the diffuse reflection lobe can be approximated with an SG taking energy conservation into account as follows:

$$\boxed{\rho_d(\mathbf{y},\boldsymbol{\omega}',\boldsymbol{\omega})\max(\boldsymbol{\omega}\cdot\mathbf{n},0) \approx R_d\frac{G(\boldsymbol{\omega},\mathbf{n},\lambda_d)}{A(\lambda_d)}}, \tag{1}$$

where $\rho_d(\mathbf{y},\boldsymbol{\omega}',\boldsymbol{\omega}) = \frac{R_d}{\pi}$, $\boldsymbol{\omega}' \in \mathrm{S}^2$ is the incoming direction, $\mathbf{n} \in \mathrm{S}^2$ is the surface normal at the world position $\mathbf{y} \in \mathbb{R}^3$, R_d is the diffuse reflectance, and $\lambda_d \approx 2$ is the sharpness of the diffuse lobe which is obtained by using the least square method.

Specular lobes. For the microfacet BRDF ρ_s, the specular reflection lobe is fitted with a single SG by using Wang et al. [WRG*09]'s analytical approximation. The BRDF is separated into two factors: the unnormalized normal distribution function (NDF) $D(\boldsymbol{\omega}_h)$ whose maximum is one, and the rest of the factors $C(\boldsymbol{\omega})$ as follows:

$$\rho_s(\mathbf{y},\boldsymbol{\omega}',\boldsymbol{\omega})\max(\boldsymbol{\omega}\cdot\mathbf{n},0) = C(\boldsymbol{\omega})D(\boldsymbol{\omega}_h),$$

where $\boldsymbol{\omega}_h = \frac{\boldsymbol{\omega}+\boldsymbol{\omega}'}{\|\boldsymbol{\omega}+\boldsymbol{\omega}'\|}$ is the halfway vector of the incoming direction and outgoing direction. Bell-shaped NDFs (e.g., Phong [Bli77], Beckmann [BS63] and GGX [TR75, WMLT07] NDFs) can be approximated with an SG as

$$D(\boldsymbol{\omega}_h) \approx G(\boldsymbol{\omega}_h,\mathbf{n},\lambda_h).$$

For Beckmann or GGX NDFs, $\lambda_h = \frac{2}{\alpha^2}$ where α is the roughness parameter. Using spherical warping, this can be approximated with a function of $\boldsymbol{\omega}$ as

$$G(\boldsymbol{\omega}_h,\mathbf{n},\lambda_h) \approx G(\boldsymbol{\omega},\boldsymbol{\xi}_s,\lambda_s),$$

where $\boldsymbol{\xi}_s$ is the reflection vector given by $\boldsymbol{\xi}_s = 2(\boldsymbol{\omega}'\cdot\mathbf{n})\mathbf{n}-\boldsymbol{\omega}'$, and $\lambda_s = \frac{\lambda_h}{4|\boldsymbol{\xi}_s\cdot\mathbf{n}|}$. Hence, the specular lobe is approximated with the following equation:

$$\rho_s(\mathbf{y},\boldsymbol{\omega}',\boldsymbol{\omega})\max(\boldsymbol{\omega}\cdot\mathbf{n},0) \approx C(\boldsymbol{\omega})G(\boldsymbol{\omega},\boldsymbol{\xi}_s,\lambda_s).$$

Moreover, since microfacet BRDFs mostly preserve energy for highly glossy surfaces, the specular lobe can be approximated using a normalized SG as follows:

$$\boxed{\rho_s(\mathbf{y},\boldsymbol{\omega}',\boldsymbol{\omega})\max(\boldsymbol{\omega}\cdot\mathbf{n},0) \approx R_s\frac{G(\boldsymbol{\omega},\boldsymbol{\xi}_s,\lambda_s)}{A(\lambda_s)}}, \tag{2}$$

where R_s is the specular reflectance. Anisotropic spherical Gaussians (ASGs) [XSD*13] are also usable in the same manner.

3. Virtual Spherical Gaussian Lights (VSGLs)

This paper approximates a cluster of virtual point lights (VPLs) [Kel97] with a VSGL. For a VSGL, the total radiant intensity and positional distribution of VPLs are represented using an SG and isotropic Gaussian distribution respectively. This representation can be computed using a simple summation operation.

3.1. Radiant intensity

The radiant intensity of the jth VPL is given as

$$I_j(\boldsymbol{\omega}) = \Phi_j \rho(\mathbf{y}_j, \boldsymbol{\omega}'_j, \boldsymbol{\omega}) \max(\boldsymbol{\omega} \cdot \mathbf{n}_j, 0),$$

where Φ_j is the power of the jth photon emitted from the light source, $\boldsymbol{\omega}'_j \in \mathbb{S}^2$ is the incoming direction of the photon, and $\mathbf{n}_j \in \mathbb{S}^2$ is the surface normal at the VPL position $\mathbf{y}_j \in \mathbb{R}^3$, and $\rho(\mathbf{y}_j, \boldsymbol{\omega}'_j, \boldsymbol{\omega})$ is the BRDF. This paper first divides this BRDF into diffuse and specular components (i.e., ρ_d and ρ_s). Then, the total radiant intensity of clustered VPLs is approximated with a single SG for each component by using Toksvig [Tok05]'s filtering. For ease of explanation, this subsection hereafter describes only a single BRDF component. The total radiant intensity of a VPL cluster $\mathbb{S}$ is represented as

$$I_v(\boldsymbol{\omega}) = \sum_{j \in \mathbb{S}} I_j(\boldsymbol{\omega}) \approx c_v G(\boldsymbol{\omega}, \boldsymbol{\xi}_v, \lambda_v).$$

To compute SG parameters c_v, $\boldsymbol{\xi}_v$ and λ_v efficiently, each reflection lobe is approximated using Eq. 1 or Eq. 2 as follows:

$$\begin{aligned} I_v(\boldsymbol{\omega}) &= \sum_{j \in \mathbb{S}} \Phi_j \rho(\mathbf{y}_j, \boldsymbol{\omega}'_j, \boldsymbol{\omega}) \max(\boldsymbol{\omega} \cdot \mathbf{n}_j, 0) \\ &\approx \sum_{j \in \mathbb{S}} \Phi_j R_j \frac{G\left(\boldsymbol{\omega}, \boldsymbol{\xi}_j, \lambda_j\right)}{A(\lambda_j)} \\ &= \left(\sum_{j \in \mathbb{S}} \Phi_j R_j\right) \frac{\sum_{j \in \mathbb{S}} \Phi_j R_j \frac{G(\boldsymbol{\omega}, \boldsymbol{\xi}_j, \lambda_j)}{A(\lambda_j)}}{\sum_{j \in \mathbb{S}} \Phi_j R_j}, \end{aligned}$$

where R_j is the reflectance, and $\boldsymbol{\xi}_j$ and λ_j are the axis and sharpness of the reflection lobe at the jth VPL. Then, the weighted average of the normalized SGs weighted by the VPL power $\Phi_j R_j$ is approximated with a single SG as

$$\frac{\sum_{j \in \mathbb{S}} \Phi_j R_j \frac{G(\boldsymbol{\omega}, \boldsymbol{\xi}_j, \lambda_j)}{A(\lambda_j)}}{\sum_{j \in \mathbb{S}} \Phi_j R_j} \approx \frac{G(\boldsymbol{\omega}, \boldsymbol{\xi}_v, \lambda_v)}{A(\lambda_v)}.$$

Using Toksvig's filtering, the jth normalized SG is first approximately converted into its averaged direction as $\bar{\boldsymbol{\xi}}_j = \frac{\lambda_j}{\lambda_j + 1} \boldsymbol{\xi}_j$. Next, the weighted average of the directions is computed by

$$\bar{\boldsymbol{\xi}}_v = \frac{\sum_{j \in \mathbb{S}} \Phi_j R_j \bar{\boldsymbol{\xi}}_j}{\sum_{j \in \mathbb{S}} \Phi_j R_j}.$$

Finally, the filtered SG is obtained from the weighted average direction as $\boldsymbol{\xi}_v = \frac{\bar{\boldsymbol{\xi}}_v}{\|\bar{\boldsymbol{\xi}}_v\|}$, $\lambda_v = \frac{\|\bar{\boldsymbol{\xi}}_v\|}{1 - \|\bar{\boldsymbol{\xi}}_v\|}$. The coefficient c_v is given by $c_v = \frac{\sum_{j \in \mathbb{S}} \Phi_j R_j}{A(\lambda_v)}$.

3.2. Positional distribution

In this paper, the positional distribution of VPLs is represented with a single isotropic Gaussian distribution for a VSGL. Unlike radiant intensity, this distribution is not divided into diffuse and specular components in order to avoid the increase of visibility tests (i.e., shadow maps). The weighted mean of VPL positions is computed by

$$\boldsymbol{\mu}_v = \frac{\sum_{j \in \mathbb{S}} \Phi_j (R_{d,j} + R_{s,j}) \mathbf{y}_j}{\sum_{j \in \mathbb{S}} \Phi_j (R_{d,j} + R_{s,j})},$$

where $R_{d,j}$ and $R_{s,j}$ are the diffuse reflectance and specular reflectance at the jth VPL, respectively. The positional variance is also calculated using weighted average as

$$\sigma_v^2 = \frac{\sum_{j \in \mathbb{S}} \Phi_j (R_{d,j} + R_{s,j}) \|\mathbf{y}_j\|^2}{\sum_{j \in \mathbb{S}} \Phi_j (R_{d,j} + R_{s,j})} - \|\boldsymbol{\mu}_v\|^2.$$

Assuming VPLs are distributed on a planar surface, the emitted radiance of a VSGL is represented as follows:

$$\boxed{L_e(\mathbf{y}, \boldsymbol{\omega}) \approx \frac{I_v(\boldsymbol{\omega})}{2\pi\sigma_v^2 |\boldsymbol{\omega} \cdot \mathbf{n}|} \gamma\left(\|\mathbf{y} - \boldsymbol{\mu}_v\|, \sigma_v^2\right)}, \tag{3}$$

where $\mathbf{n}$ is the surface normal which will be eliminated in shading (§4.1).

3.3. VSGL generation using reflective shadow maps

As mentioned in §3.1 and 3.2, a VSGL is generated by calculating the total VPL power $\sum_{j \in \mathbb{S}} \Phi_j R_j$, total weighted emission direction $\sum_{j \in \mathbb{S}} \Phi_j R_j \bar{\boldsymbol{\xi}}_j$, total weighted position $\sum_{j \in \mathbb{S}} \Phi_j (R_{d,j} + R_{s,j}) \mathbf{y}_j$, and total weighted squared norm of the position $\sum_{j \in \mathbb{S}} \Phi_j (R_{d,j} + R_{s,j}) \|\mathbf{y}_j\|^2$. Therefore, these values are stored into reflective shadow maps (RSMs) [DS05], and then they are mipmapped to obtain the total values. The ith VPL cluster is represented by the unnormalized filtering kernel $g((\mathbf{x} - \mathbf{x}_i)/s_i)$ on the RSM. For example, Let $f(\mathbf{x})$ be VPL power stored in the RSM, then the total VPL power of ith VPL cluster is given by

$$\sum_{j \in \mathbb{S}} \Phi_j R_j = \int_{[0,1]^2} f(\mathbf{x}) g((\mathbf{x} - \mathbf{x}_i)/s_i) \, \mathrm{d}\mathbf{x} \approx \frac{4^{l_i}}{M} \bar{f}(\mathbf{x}_i, l_i).$$

We are also able to calculate the total weighted emission direction, total weighted position, and total weighted squared norm of the position in the same manner. In this paper, the image-space position $\mathbf{x}_i$ and mip level l_i are sampled based on FIS.

4. Shading

For each shading point $\mathbf{y}_p$ with view direction $\boldsymbol{\omega}_p$, the reflected radiance is calculated using the rendering equation [Kaj86] defined by

$$L(\mathbf{y}_p, \boldsymbol{\omega}_p) = \int_{\mathbb{S}^2} L_{in}(\mathbf{y}_p, \boldsymbol{\omega}) \rho(\mathbf{y}_p, \boldsymbol{\omega}_p, \boldsymbol{\omega}) \max(\boldsymbol{\omega} \cdot \mathbf{n}_p, 0) \mathrm{d}\boldsymbol{\omega}, \tag{4}$$

where $L_{in}(\mathbf{y}_p, \boldsymbol{\omega})$ is the incoming radiance, and $\mathbf{n}_p$ is the surface normal at the shading point. This paper approximates the incoming radiance using SGs for the analytical approximation of the rendering integral [WRG*09, XSD*13].

4.1. Incoming radiance

Using Eq. 3, the approximated incoming radiance is given by

$$\begin{aligned} L_{in}(\mathbf{y}_p, \boldsymbol{\omega}) &= V(\mathbf{y}_p, \mathbf{y}) L_e(\mathbf{y}, -\boldsymbol{\omega}) \\ &\approx \frac{V(\mathbf{y}_p, \boldsymbol{\mu}_v) I_v(-\boldsymbol{\omega})}{2\pi\sigma_v^2 |\boldsymbol{\omega} \cdot \mathbf{n}|} \gamma\left(\|\mathbf{y} - \boldsymbol{\mu}_v\|, \sigma_v^2\right), \end{aligned} \tag{5}$$

where $\boldsymbol{\omega} = \frac{\mathbf{y}-\mathbf{y}_p}{\|\mathbf{y}-\mathbf{y}_p\|}$, $V(\mathbf{y}_p, \boldsymbol{\mu}_v)$ is the visibility between $\mathbf{y}_p$ and $\boldsymbol{\mu}_v$ obtained from a shadow map. The position $\mathbf{y}$ is assumed to be on the planar surface defined by the normal $\mathbf{n}$ and position $\boldsymbol{\mu}_v$. However, $\mathbf{n}$ and $\mathbf{y}$ are unknown for shading. Therefore, we project the positional distribution onto a sphere centered at a shading point instead. To correct the energy for this projection, $|\boldsymbol{\omega}\cdot\mathbf{n}|$ is multiplied similar to virtual spherical lights [HKWB09]. Since it is divided by $|\boldsymbol{\omega}\cdot\mathbf{n}|$, $\mathbf{n}$ is eliminated. This is reasonable because the actual surface normal distribution is taken into account by the radiant intensity $I_v(-\boldsymbol{\omega})$. Therefore, Eq. 5 is approximated with the following equation:

$$L_{in}(\mathbf{y}_p, \boldsymbol{\omega}) \approx \frac{I_v(-\boldsymbol{\omega})}{2\pi\sigma_v^2}\gamma\left(\|\mathbf{y}_r - \boldsymbol{\mu}_v\|, \sigma_v^2\right),$$

where $\boldsymbol{\omega} = \frac{\mathbf{y}_r-\mathbf{y}_p}{\|\mathbf{y}_r-\mathbf{y}_p\|}$, and $\mathbf{y}_r$ is the position on the sphere defined by the center $\mathbf{y}_p$ and radius $\|\boldsymbol{\mu}_r - \mathbf{y}_p\|$. This is derived assuming a small σ_v or large radius, but it does not produce noticeable artifacts in practice for a large σ_v and small radius. The Gaussian term can be rewritten into an SG as

$$\gamma\left(\|\mathbf{y}_r - \boldsymbol{\mu}_v\|, \sigma_v^2\right) = G(\boldsymbol{\omega}, \boldsymbol{\xi}_\mu, \lambda_\sigma), \tag{6}$$

where $\boldsymbol{\xi}_\mu = \frac{\boldsymbol{\mu}_v-\mathbf{y}_p}{\|\boldsymbol{\mu}_v-\mathbf{y}_p\|}$, and $\lambda_\sigma = \frac{\|\boldsymbol{\mu}_v-\mathbf{y}_p\|^2}{\sigma_v^2}$. This SG represents the spherical region of the VSGL viewed from $\mathbf{y}_p$. Using Eq. 6, the incoming radiance is approximated with the product of two SGs which yields an SG as follows:

$$\begin{aligned} L_{in}(\mathbf{y}_p, \boldsymbol{\omega}) &\approx \frac{c_v}{2\pi\sigma_v^2} G(\boldsymbol{\omega}, -\boldsymbol{\xi}_v, \lambda_v)\, G(\boldsymbol{\omega}, \boldsymbol{\xi}_\mu, \lambda_\sigma) \\ &= \boxed{c_{in} G(\boldsymbol{\omega}, \boldsymbol{\xi}_{in}, \lambda_{in})}, \end{aligned} \tag{7}$$

where $\xi_{in} = \frac{\lambda_\sigma \boldsymbol{\xi}_\mu - \lambda_v \boldsymbol{\xi}_v}{\|\lambda_\sigma \boldsymbol{\xi}_\mu - \lambda_v \boldsymbol{\xi}_v\|}$, $\lambda_{in} = \|\lambda_\sigma \boldsymbol{\xi}_\mu - \lambda_v \boldsymbol{\xi}_v\|$, and $c_{in} = \frac{c_v}{2\pi\sigma_v^2} e^{\lambda_{in}-\lambda_v-\lambda_\sigma}$.

4.2. Shading via product integrals of SGs

Since the reflection lobe $\rho(\mathbf{y}_p, \boldsymbol{\omega}_p, \boldsymbol{\omega}) \max(\boldsymbol{\omega}\cdot\mathbf{n}_p, 0)$ can be approximated using SGs and ASGs, Eq. 4 can be calculated using the analytical product integral.

Diffuse reflection. Using Eq. 1 and Eq. 7, the rendering integral of the diffuse component is calculated using the analytical product integral of two SGs. This approach is efficient for a few VSGLs [Tok15a]. However, a light leak error caused by the SG approximation of reflection lobes cannot be reduced by increasing the number of VSGLs. Unlike the secondary bounce represented by VSGLs, light leaks are noticeable at the first bounce which is more visually important. Therefore, for thousands of VSGLs, the cosine factor at the first bounce is assumed to be a constant and pulled out of the integral [WRG*09] as follows:

$$\begin{aligned} L_d(\mathbf{x}_p, \boldsymbol{\omega}_p) &= \int_{S^2} L_{in}(\mathbf{x}_p, \boldsymbol{\omega}) \rho_d(\mathbf{x}_p, \boldsymbol{\omega}_p, \boldsymbol{\omega}) \max(\boldsymbol{\omega}\cdot\mathbf{n}_p, 0) d\boldsymbol{\omega} \\ &\approx \frac{c_{in} R_d}{\pi} A(\lambda_{in}) \max(\boldsymbol{\xi}_{in}\cdot\mathbf{n}_p, 0). \end{aligned}$$

In addition, when λ_{in} is not small, $A(\lambda_{in}) \approx \frac{2\pi}{\lambda_{in}}$ can be assumed [IDN12]. Therefore, diffuse reflection is inexpensively calculated using the following equation:

$$\boxed{L_d(\mathbf{x}_p, \boldsymbol{\omega}_p) \approx \frac{2 c_{in} R_d}{\lambda_{in}} \max(\boldsymbol{\xi}_{in}\cdot\mathbf{n}_p, 0)}.$$

Specular reflection. While SGs are used for VSGLs, this paper employs an ASG to approximate a specular lobe at a shading point. This is because a specular lobe can be anisotropic even if it is an isotropic BRDF model, especially for shallow grazing angles. For simplicity, ASGs are used only for the first bounce which is more visually important than the second bounce. In addition, the product integral of an ASG and SG [XSD*13] has a reasonable computation cost. An ASG is defined as

$$\acute{G}(\boldsymbol{\omega}, \boldsymbol{\xi}_x, \boldsymbol{\xi}_y, \boldsymbol{\xi}_z, \eta_x, \eta_y) = \max(\boldsymbol{\omega}\cdot\boldsymbol{\xi}_z, 0) e^{-\eta_x(\boldsymbol{\omega}\cdot\boldsymbol{\xi}_x)^2 - \eta_y(\boldsymbol{\omega}\cdot\boldsymbol{\xi}_y)^2},$$

where $\boldsymbol{\xi}_x, \boldsymbol{\xi}_y, \boldsymbol{\xi}_z$ are orthonormal vectors, and η_x, η_y are the bandwidth parameters. Since a specular lobe is approximated with an ASG as $\rho_s(\mathbf{y}_p, \boldsymbol{\omega}_p, \boldsymbol{\omega}) \max(\boldsymbol{\omega}\cdot\mathbf{n}_p, 0) \approx C(\boldsymbol{\omega}) \acute{G}(\boldsymbol{\omega}, \boldsymbol{\xi}_x, \boldsymbol{\xi}_y, \boldsymbol{\xi}_z, \eta_x, \eta_y)$, the rendering integral is calculated as

$$\begin{aligned} L_s(\mathbf{y}_p, \boldsymbol{\omega}_p) &= \int_{S^2} L_{in}(\mathbf{y}_p, \boldsymbol{\omega}) \rho_s(\mathbf{y}_p, \boldsymbol{\omega}_p, \boldsymbol{\omega}) \max(\boldsymbol{\omega}\cdot\mathbf{n}_p, 0) d\boldsymbol{\omega} \\ &\approx c_{in} C(\boldsymbol{\xi}_{in}) \int_{S^2} G(\boldsymbol{\omega}, \boldsymbol{\xi}_{in}, \lambda_{in}) \acute{G}\left(\boldsymbol{\omega}, \boldsymbol{\xi}_x, \boldsymbol{\xi}_y, \boldsymbol{\xi}_z, \eta_x, \eta_y\right) d\boldsymbol{\omega} \\ &\approx \boxed{\frac{\pi c_{in} C(\boldsymbol{\xi}_{in}) \acute{G}\left(\boldsymbol{\xi}_{in}, \boldsymbol{\xi}_x, \boldsymbol{\xi}_y, \boldsymbol{\xi}_z, \frac{\eta_x \nu}{\eta_x+\nu}, \frac{\eta_y \nu}{\eta_y+\nu}\right)}{\sqrt{(\eta_x+\nu)(\eta_y+\nu)}}}, \end{aligned}$$

where $\nu = \frac{\lambda_{in}}{2}$.

5. Implementation Details of VSGL Generation

VSGL generation using FIS. Our implementation is based on Tokuyoshi [Tok15b], and uses DirectX® 11. After rendering an RSM, an additional RSM (which stores VPL positions, squared VPL positions, and average emission directions to calculate VSGL parameters) is generated using a compute shader. Then, these RSMs are mipmapped using a graphics API (i.e., *GenerateMips* of DirectX). Finally, VSGLs are generated based on FIS. The proposed mip level l_i is calculated using Algorithm 1.

Algorithm 1 Mip level calculation using the bisection method.

```
l_min ← 0
l_max ← the top mip level of p̄
for k = 1 to the user-specified iteration count do
    l ← (l_min + l_max)/2
    if (4^l / M) p̄(x_i, l) < K/N then
        l_min ← l
    else
        l_max ← l
    end if
end for
l_i ← (l_min + l_max)/2
```

VSGL generation using *k*-means. For comparison, this paper uses k-means VPL clustering using 2D image space and 3D world space. The k-means algorithm first samples the cluster center according to the probability density function, and then the closest center is computed for each VPL. In our implementation, all the texels are assigned to clusters for high-frequency geometries and textured glossy materials unlike Dong et al. [DGR*09]. To accelerate the search of the closest cluster center for each texel, a kd-tree of cluster centers is built using parallel construction of a binary radix tree [Kar12]. This tree-based search is more efficient than using a 2D uniform grid proposed by Prutkin et al. [PKD12] for densely distributed cluster centers. 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]. Unlike Prutkin et al., we use a GPU radix sort [MG10] instead of bitonic sort for the high-resolution RSM and G-buffer. Although k-means clustering can be improved by updating cluster centers in an iterative fashion, we do not update iteratively in this paper.

References

[Bli77] BLINN J. F.: Models of light reflection for computer synthesized pictures. *SIGGRAPH Comput. Graph. 11*, 2 (1977), 192–198. 1

[BS63] BECKMANN P., SPIZZICHINO A.: *Scattering of Electromagnetic Waves from Rough Surfaces*. MacMillan, 1963. 1

[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. 4

[DS05] DACHSBACHER C., STAMMINGER M.: Reflective shadow maps. In *I3D'05* (2005), pp. 203–231. 2

[HKWB09] HAŠAN M., KŘIVÁNEK J., WALTER B., BALA K.: Virtual spherical lights for many-light rendering of glossy scenes. *ACM Trans. Graph. 28*, 5 (2009), 143:1–143:6. 3

[IDN12] IWASAKI K., DOBASHI Y., NISHITA T.: Interactive bi-scale editing of highly glossy materials. *ACM Trans. Graph. 31*, 6 (2012), 144:1–144:7. 3

[Kaj86] KAJIYA J. T.: The rendering equation. *SIGGRAPH Comput. Graph. 20*, 4 (1986), 143–150. 2

[Kar12] KARRAS T.: Maximizing parallelism in the construction of BVHs, octrees, and k-d trees. In *HPG'12* (2012), pp. 33–37. 4

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

[MG10] MERRILL D. G., GRIMSHAW A. S.: Revisiting sorting for GPGPU stream architectures. In *PACT'10* (2010), pp. 545–546. 4

[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. 4

[Tok05] TOKSVIG M.: Mipmapping normal maps. *J. Graph. Tools 10*, 3 (2005), 65–71. 2

[Tok15a] TOKUYOSHI Y.: Fast indirect illumination using two virtual spherical Gaussian lights. In *SA'15 Posters* (2015), pp. 12:1–12:1. 3

[Tok15b] TOKUYOSHI Y.: Virtual spherical Gaussian lights for real-time glossy indirect illumination. *Comput. Graph. Forum 34*, 7 (2015), 89–98. 1, 3

[TR75] TROWBRIDGE T. S., REITZ K. P.: Average irregularity representation of a rough surface for ray reflection. *J. Opt. Soc. Am 65*, 5 (1975), 531–536. 1

[WMLT07] WALTER B., MARSCHNER S. R., LI H., TORRANCE K. E.: Microfacet models for refraction through rough surfaces. In *EGSR'07* (2007), pp. 195–206. 1

[WRG*09] WANG J., REN P., GONG M., SNYDER J., GUO B.: All-frequency rendering of dynamic, spatially-varying reflectance. *ACM Trans. Graph. 28*, 5 (2009), 133:1–133:10. 1, 2, 3

[XSD*13] XU K., SUN W.-L., DONG Z., ZHAO D.-Y., WU R.-D., HU S.-M.: Anisotropic spherical Gaussians. *ACM Trans. Graph. 32*, 6 (2013), 209:1–209:11. 1, 2, 3