Supplemental Material IV: Ratio Estimator Properties

Zdravko Velinov, and Kenny Mitchell

Abstract—We discuss convergence rate, bias, progressive estimation and other properties of ratio estimators.

Index Terms—raytracing, color, shading, shadowing, texture.

1 BIAS

Monte Carlo techniques follow the definition of computing the expected value of a random variable, and as long as the probability density function correctly matches the distribution of samples generated by transforming a pseudorandom sequence by a sampling function, the estimator is going to be unbiased [1].

Moment-based estimators following the framework explained by Peters *et al.* [2] use heuristic functions that do not converge to the correct value, no matter whether we scale the resolution and increase the number of moments.

Ratio estimators are technically a composite estimator. They combine the results of multiple estimators to arrive at a value with improved convergence rate. We will demonstrate it after introducing some statistical framework to keep the discussion clear. Suppose that we have a random variable m with expected value μ, we define expectation as

$$\mu = \mathbb{E}\left[m\right] = \sum_{i=1}^{+\infty} m_i \cdot p_i = \int_{\mathbf{R}} m(x)\, p(x)\, \mathrm{d}x. \tag{1}$$

We can similarly define the expected value τ of another random variable t as $\tau = \mathbb{E}\left[t\right]$. Ratio estimators essentially represent the outcome of an unbiased estimator divided by the estimate of related variable and multiplied by the expected value of the same variable. Expressed mathematically,

$$\mu = \frac{\tau}{\mathbb{E}\left[t\right]} \mathbb{E}\left[m\right] = \mathbb{E}\left[m\right], \tag{2}$$

The following relation results in an unbiased estimate of m only if the expected value of the related variable t is not equal to 0 ($\tau \neq 0$). If we want to make this estimator avoid this case arithmetically, we can add an impulse function to both its numerator and denominator,

$$i(x) = \begin{cases} 1 & x = 0 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

$$\mu = \frac{\tau + i(\tau)}{\mathbb{E}\left[t\right] + i(\tau)} \mathbb{E}\left[m\right] = \mathbb{E}\left[m\right] \tag{4}$$

Note, that the outcome of the two estimators have to be computed independently before combining them in a single ratio estimator, otherwise bias will be introduced in the computation. We can demonstrate mathematically that it won't converge to the correct value if we directly compute the expectation of the ratio of these two related random variables,

$$\mu \neq \mathbb{E}\left[\frac{\tau}{t} m\right] \tag{5}$$

We can go as far as computing the bias of that estimator which is the difference of the two expected values,

$$\begin{aligned} \mathbb{B}\left(m, \frac{\tau}{t} m\right) &= \mathbb{E}\left[m\right] - \mathbb{E}\left[\frac{\tau}{t} m\right] = \mathbb{E}\left[m - \frac{\tau}{t} m\right] \\ &= \mathbb{E}\left[\left(1 - \frac{\tau}{t}\right) m\right] \end{aligned} \tag{6}$$

Obviously, the estimator in (5) is biases, but we already defined an estimator that we claim converges asymptotically to the correct solution (4). We can validate it in the same manner,

$$\begin{aligned} \mathbb{B}\left(m, \frac{\tau + i(\tau)}{\mathbb{E}\left[t\right] + i(\tau)} \mathbb{E}\left[m\right]\right) &= \mathbb{E}\left[m\right] - \mathbb{E}\left[\frac{\tau + i(\tau)}{\mathbb{E}\left[t\right] + i(\tau)} \mathbb{E}\left[m\right]\right] \\ &= \mathbb{E}\left[m - \frac{\tau + i(\tau)}{\mathbb{E}\left[t\right] + i(\tau)} m\right] \\ &= \mathbb{E}\left[\left(\frac{\tau + i(\tau)}{\mathbb{E}\left[t\right] + i(\tau)} - 1\right) m\right] \\ &= \mathbb{E}\left[\left(\frac{\cancelto{1}{\tau + i(\tau)}}{\cancelto{1}{\tau + i(\tau)}} - 1\right) m\right] \\ &= \mathbb{E}\left[(1 - 1)\, m\right] = 0, \end{aligned} \tag{7}$$

therefore, we prove arithmetically that the ratio estimator in (4) is unbiased only in the limit. However, when truncating the sequence the result will deviate from the expected value,

$$\mathbb{B}\left(m, \frac{\tau}{\hat{t}} \hat{m}\right) = \mathbb{E}\left[\left(1 - \frac{\tau}{\sum_{i=0}^{N} t_i\, \hat{p}_i}\right) \left(\sum_{j=0}^{N} m_j\, \hat{p}_j\right)\right]. \tag{8}$$

The most significant issue here is that directly averaging values might introduce bias and the result of each estimator must be

- *Zdravko Velinov was with Disney Research Los Angeles, Glendale, CA. ORCiD: 0000-0002-1258-1223*
- *Kenny Mitchell was with Disney Research Los Angeles, Edinburgh Napier University and Roblox Corporation, San Mateo, CA. ORCiD: 0000-0003-2420-7447*

Manuscript received December 14, 2020; revised December, 2021.

treated separately. If the numerator and denominator are treated separately, the estimator on average does not lead to bias,

$$\begin{aligned}\mathbb{B}^*\left(m,(\tau+i(\tau))\,\hat{m},\hat{t}+i(\tau)\right) =&\mathbb{E}\left[m\right]-\frac{\mathbb{E}\left[(\tau+i(\tau))\,\hat{m}\right]}{\mathbb{E}\left[\hat{t}+i(\tau)\right]}\\ =&\left(1-\frac{\tau}{\mathbb{E}\left[\sum_{i=0}^{N}t_i\,\hat{p}_i\right]}\right)\cdot\\ &\mathbb{E}\left[\sum_{j=0}^{N}m_j\,\hat{p}_j\right]=0. \end{aligned} \tag{9}$$

However, we use a much weaker requirement with respect to avoiding bias in the estimator and set certain requirements on the implementation side. Therefore, we can claim at least asymptotic convergence, if progressive estimation is carried correctly, which will be discussed in the following.

2 Progressive estimation

Progressive estimators or renderers in the case of graphics, spread the computation over multiple iterations. We can achieve progressive estimation by first considering that in practice ratio estimators combine the result of two estimators at each step. Therefore, our estimator can be defined as $\mu^*:\mathbf{R}^n,\mathbf{R}^n\to\mathbf{R}^n$,

$$\mu^*(E_t,E_m)=\frac{\tau+i(\tau)}{E_t+i(\tau)}E_m. \tag{10}$$

We can then define the expectation as a sum of values up to and past a certain point k,

$$\begin{aligned}\mathbb{E}\left[m\right]&=\mathbb{E}_{-,k}[m]+\mathbb{E}_{+,k}[m]\\ &=\sum_{i=1}^{k}m_i\cdot p_i+\sum_{i=k+1}^{+\infty}m_i\cdot p_i.\end{aligned} \tag{11}$$

The described estimator generally works, but we will need to split it into multiple steps which yield plausible intermediate results. Therefore, we need to normalize the two terms,

$$\begin{aligned}\mathbb{E}\left[m\right]=&\sum_{i=1}^{k}p_i\left(\frac{1}{\sum_{i=1}p_i}\sum_{i=1}^{k}m_i\cdot p_i\right)+\\ &\sum_{i=k+1}^{+\infty}p_i\left(\frac{1}{\sum_{i=k+1}^{+\infty}p_i}\sum_{i=k+1}^{+\infty}m_i\cdot p_i\right)\\ =&\left(\sum_{i=1}^{k}p_i\right)\hat{\mathbb{E}}_{1,k}[m]+\left(\sum_{i=k+1}^{+\infty}p_i\right)\hat{\mathbb{E}}_{k+1,+\infty}[m]\\ =&P_{1,k}\,\hat{\mathbb{E}}_{1,k}[m]+P_{k+1,+\infty}\,\hat{\mathbb{E}}_{k+1,+\infty}[m].\end{aligned} \tag{12}$$

We can take one of those partial terms and use them with the ratio estimator to perform an estimate with finite number of samples $\mu(\hat{\mathbb{E}}_{1,k}[t],\hat{\mathbb{E}}_{1,k}[t])$. After each iteration we can combine the results from multiple estimations similarly to (12),

$$\hat{\mathbb{E}}_{1,k_2}[m]=P_{1,k_1}\,\hat{\mathbb{E}}_{1,k_1}[m]+P_{k_1+1,k_2}\,\hat{\mathbb{E}}_{k_1+1,k_2}[m]. \tag{13}$$

3 Convergence of Ratio Estimators

In the general case, ratio estimators can be built out of two uncorrelated distributions. In that case, ratio estimators do not lead to any variance reduction, but in general they converge to the correct result, if they are built out of Monte Carlo estimators (cf. Fig. 2). The probability density function depends on the two estimators. The most straightforward approach to estimate it is to build a histogram out of generated samples. The distribution of errors of a Monte Carlo estimator follows a normal distribution with expected value μ and standard deviation $\hat{\sigma}_m=\sigma_m/\sqrt{N}$. To generate error samples of ratio estimator, we can use the quantile function of the normal distribution characterizing the error of each estimator,

$$x=\frac{(\tau+i(\tau))\left(\mu+\hat{\sigma}_m\sqrt{2}\,\mathrm{erf}^{-1}(2\xi_1-1)\right)}{\tau+\hat{\sigma}_t\sqrt{2}\,\mathrm{erf}^{-1}(2\xi_2-1)+i(\tau)}-\mu, \tag{14}$$

where ξ_1 and ξ_2 are random values drawn from uniform distribution in range [0, 1], and σ_t and σ_m are the standard deviation of each distribution which is integrated numerically. After generating the samples, they are binned and then normalized by dividing out the number of generated samples multiplied by the size of each bin. Another approach to solve this problem numerically is to solve the equation for one of the distribution to find the intersection with a given axis and integrate analytically the other distribution weighted by the probability density of having a sample at a given location. For convenience we will substitute the quantile $\psi=G^{-1}(\xi_2,\tau,\hat{\sigma}_t)=\tau+\hat{\sigma}_t\sqrt{2}\,\mathrm{erf}^{-1}(2\xi_2-1)$,

$$\begin{aligned}x=&\frac{(\tau+i(\tau))\left(\mu+\hat{\sigma}_m\sqrt{2}\,\mathrm{erf}^{-1}(2\xi_1-1)\right)}{\psi+i(\tau)}-\mu\\ \frac{(\psi+i(\tau))\,(x+\mu)}{(\tau+i(\tau))}=&\left(\mu+\hat{\sigma}_m\sqrt{2}\,\mathrm{erf}^{-1}(2\xi_1-1)\right)\\ \frac{(\psi+i(\tau))(x+\mu)}{(\tau+i(\tau))}-\mu=&\,\hat{\sigma}_m\sqrt{2}\,\mathrm{erf}^{-1}(2\xi_1-1)\\ \frac{1}{\hat{\sigma}_m\sqrt{2}}\left(\frac{(\psi+i(\tau))(x+\mu)}{(\tau+i(\tau))}-\mu\right)=&\,\mathrm{erf}^{-1}(2\,\xi_1-1)\\ \mathrm{erf}\left(\frac{1}{\hat{\sigma}_m\sqrt{2}}\left(\frac{(\psi+i(\tau))(x+\mu)}{\tau+i(\tau)}-\mu\right)\right)=&\,2\,\xi_1-1\\ \xi_1=&\,\frac{1}{2}+\frac{1}{2}\mathrm{erf}\left(\frac{1}{\hat{\sigma}_m\sqrt{2}}\left(\frac{(\psi+i(\tau))(x+\mu)}{\tau+i(\tau)}-\mu\right)\right)\\ \xi_1=&\,\Xi(x,\psi).\end{aligned} \tag{15}$$

The equation basically expresses the relation: given a sample from one distribution at a point, what is the mapping to the other distribution. Thus integrating the probability for each sample is achieved by drawing a sample at a given location from one distribution, then remapping it to the other distribution, and afterwards weighting them by the individual probability of selecting a sample from both independent distributions,

$$\begin{aligned}f(x)=\int_{-\infty}^{+\infty}&g\left(G^{-1}(\Xi(x,\xi_2),\mu,\hat{\sigma}_m),\mu,\hat{\sigma}_m\right)\cdot\\ &g\left(\psi,\tau,\hat{\sigma}_t\right)\mathrm{d}\psi.\end{aligned} \tag{16}$$

The equation can be further simplified by cancelling the error function by the quantile,

$$f(x)=\int_{-\infty}^{+\infty}g\left(\psi,\tau,\hat{\sigma}_t\right)g\left(\frac{(\psi+i(\tau))(x+\mu)}{(\tau+i(\tau))},\mu,\hat{\sigma}_m\right)\mathrm{d}\psi. \tag{17}$$

The second normal distribution can be re-written to transform its expected value and standard deviation,

$$\begin{aligned}f(x)=\left|\frac{\tau+i(\tau)}{\mu+x}\right|&\int_{-\infty}^{+\infty}g\left(\psi,\tau,\hat{\sigma}_t\right)\cdot\\ &g\left(\psi,\mu\frac{\tau+i(\tau)}{\mu+x}-i(\tau),\hat{\sigma}_m\left|\frac{\tau+i(\tau)}{\mu+x}\right|\right)\mathrm{d}\psi.\end{aligned} \tag{18}$$

We can then substitute the new expected value and standard deviation to keep the equation short,

$$
\begin{aligned}
\mu' &= \mu\,\frac{\tau+i(\tau)}{\mu+x} - i(\tau) = \frac{(\tau+i(\tau))\,\mu - i(\tau)(x+\mu)}{\mu+x} \\
&= \frac{\tau\,\mu - i(\tau)x}{\mu+x} \\
\hat{\sigma}'_m &= \hat{\sigma}_m \left|\frac{\tau+i(\tau)}{\mu+x}\right| \\
f(x) &= \frac{\hat{\sigma}'_m}{\hat{\sigma}_m}\int_{-\infty}^{+\infty} g\big(\psi,\tau,\hat{\sigma}_t\big)\, g\big(\psi,\mu',\hat{\sigma}'_m\big)\,\mathrm{d}\psi. \qquad (19)
\end{aligned}
$$

The two normal distributions can be combined into a single one,

$$
\begin{aligned}
f(x) =& \frac{\hat{\sigma}'_m}{\hat{\sigma}_m\sqrt{\hat{\sigma}_t^2+\hat{\sigma}_m'^2}} \exp\left(-\frac{1}{2}\,\frac{\tau^2\,\hat{\sigma}_m'^2 + \mu'^2\,\hat{\sigma}_t^2 - \frac{(\tau\,\hat{\sigma}_m'^2+\mu\,\hat{\sigma}_t^2)^2}{\hat{\sigma}_m'2+\hat{\sigma}_t^2}}{\hat{\sigma}_m'^2\,\hat{\sigma}_t^2}\right)\cdot \\
& \cancelto{1}{\int_{-\infty}^{+\infty} g\left(\psi,\frac{\tau\,\hat{\sigma}_m'^2+\mu\,\hat{\sigma}_t^2}{\hat{\sigma}_m'^2+\hat{\sigma}_t^2},\frac{\hat{\sigma}_t\,\hat{\sigma}'_m}{\sqrt{\hat{\sigma}_t^2+\hat{\sigma}_m'^2}}\right)\mathrm{d}\psi}. \qquad (20)
\end{aligned}
$$

The normal distribution obviously integrates to 1,

$$
\begin{aligned}
f(x) =& \frac{\hat{\sigma}'_m}{\hat{\sigma}_m\sqrt{(\hat{\sigma}_t^2+\hat{\sigma}_m'^2)\,2\,\pi}}\cdot \\
& \exp\left(-\frac{1}{2}\,\frac{\tau^2\,\hat{\sigma}_m'^2 + \mu'^2\,\hat{\sigma}_t^2 - \frac{(\tau\,\hat{\sigma}_m'^2+\mu'\,\hat{\sigma}_t^2)^2}{\hat{\sigma}_m'^2+\hat{\sigma}_t^2}}{\hat{\sigma}_m'^2\,\hat{\sigma}_t^2}\right) \\
=& \frac{\hat{\sigma}'_m}{\hat{\sigma}_m\sqrt{(\hat{\sigma}_t^2+\hat{\sigma}_m'^2)\,2\,\pi}}\cdot \\
& \exp\left(-\frac{1}{2}\,\frac{(\hat{\sigma}_m'^2+\hat{\sigma}_t^2)(\tau^2\,\hat{\sigma}_m'^2+\mu'^2\,\hat{\sigma}_t^2) - (\tau\,\hat{\sigma}_m'^2+\mu'\,\hat{\sigma}_t^2)^2}{\hat{\sigma}_m'^2\,\hat{\sigma}_t^2(\hat{\sigma}_m'^2+\hat{\sigma}_t^2)}\right) \\
=& \frac{\hat{\sigma}'_m}{\hat{\sigma}_m\sqrt{(\hat{\sigma}_t^2+\hat{\sigma}_m'^2)\,2\,\pi}} \exp\left(-\frac{(\tau-\mu')^2}{2(\hat{\sigma}_t^2+\hat{\sigma}_m'^2)}\right). \qquad (21)
\end{aligned}
$$

Expanding,

$$
\begin{aligned}
f(x) =& \left|\frac{\tau+i(\tau)}{\mu+x}\right| \frac{1}{\sqrt{\left(\hat{\sigma}_t^2+\hat{\sigma}_m^2\left(\frac{\tau+i(\tau)}{\mu+x}\right)^2\right)2\,\pi}}\cdot \\
& \exp\left(-\frac{1}{2}\,\frac{\left(\tau-\frac{\tau\,\mu-i(\tau)x}{\mu+x}\right)^2}{\left(\hat{\sigma}_t^2+\hat{\sigma}_m^2\left(\frac{\tau+i(\tau)}{\mu+x}\right)\right)}\right) \\
f(x) =& \frac{|\tau+i(\tau)|}{\sqrt{\left((\mu+x)^2\,\hat{\sigma}_t^2+(\tau+i(\tau))^2\,\hat{\sigma}_m^2\right)2\,\pi}}\cdot \\
& \exp\left(-\frac{1}{2}\,\frac{((\cancel{\mu}+x)\,\tau-(\tau\cancel{\mu}-i(\tau)x))^2}{((\mu+x)^2\,\hat{\sigma}_t^2+(\tau+i(\tau))^2\,\hat{\sigma}_m^2)}\right) \\
=& \frac{|\tau+i(\tau)|}{\sqrt{\left((\mu+x)^2\,\hat{\sigma}_t^2+(\tau+i(\tau))^2\,\hat{\sigma}_m^2\right)2\,\pi}}\cdot \\
& \exp\left(-\frac{1}{2}\,\frac{x^2\,(\tau+i(\tau))^2}{((\mu+x)^2\,\hat{\sigma}_t^2+(\tau+i(\tau))^2\,\hat{\sigma}_m^2)}\right) \\
=& \frac{1}{\sqrt{\left(\left(\frac{\mu+x}{\tau+i(\tau)}\right)^2\hat{\sigma}_t^2+\hat{\sigma}_m^2\right)2\,\pi}}\cdot \\
& \exp\left(-\frac{1}{2}\,\frac{x^2}{\left(\frac{\mu+x}{\tau+i(\tau)}\right)^2\hat{\sigma}_t^2+\hat{\sigma}_m^2}\right) \\
=& g\left(x,0,\sqrt{\left(\frac{\mu+x}{\tau+i(\tau)}\right)^2\hat{\sigma}_t^2+\hat{\sigma}_m^2}\right) \qquad (22)
\end{aligned}
$$

Finally we have a skewed normal distribution representing the distribution of errors of a ratio estimator. When $\hat{\sigma}_t \to 0$, it falls back to regular Monte Carlo error as expected,

$$
\lim_{\hat{\sigma}_t\to 0} f(x) = g(x,0,\hat{\sigma}_m) = \frac{1}{\hat{\sigma}_m\sqrt{2\,\pi}}\exp\left(-\frac{1}{2}\,\frac{x^2}{\hat{\sigma}_m^2}\right) \qquad (23)
$$

The distribution of errors is clearly skewed, i.e. shows distribution bias, but its standard deviation decreases with more samples and asymptotically converges to the correct result in the limit (cf. Fig. 2). The observed behavior is valid for relatively independent distributions where the unoccluded term does not provide good initial estimates. When occlusion is relatively low, the ratio estimator leads to formation of off-center error side lobes (cf. Tab. 3). Lower values means that it estimates the components as darker than expected. As more samples are accumulated, it tends to switch to overestimating values by having errors introducing slightly brighter pixels. We truncated the distribution along the y axis, so the probability of really low error values are not shown. What has to be outlined is that those side lobes can have much lower error than the initial error of estimators based on ray marching. The estimator has clear advantage in cases when unoccluded areas are still present in the view. WVDS, however, performs consistently better than regular distance sampling which introduces more errors around corners. These images were built by comparing the error for each color channel of each pixel in 128 images at 200x150 resolution with different initial seeds. The random number generator is a relatively simple hashing function and we use single-precision floating-point variables (32-bit) with the mathematics functions bundled in the C library as part of the MSVC C++ compiler. Better agreement with Monte Carlo error can be certainly achieved at higher precision and using high-quality random number generator, but most real-world implementations are not going to dedicate the extra budget for relatively minor precision improvements.

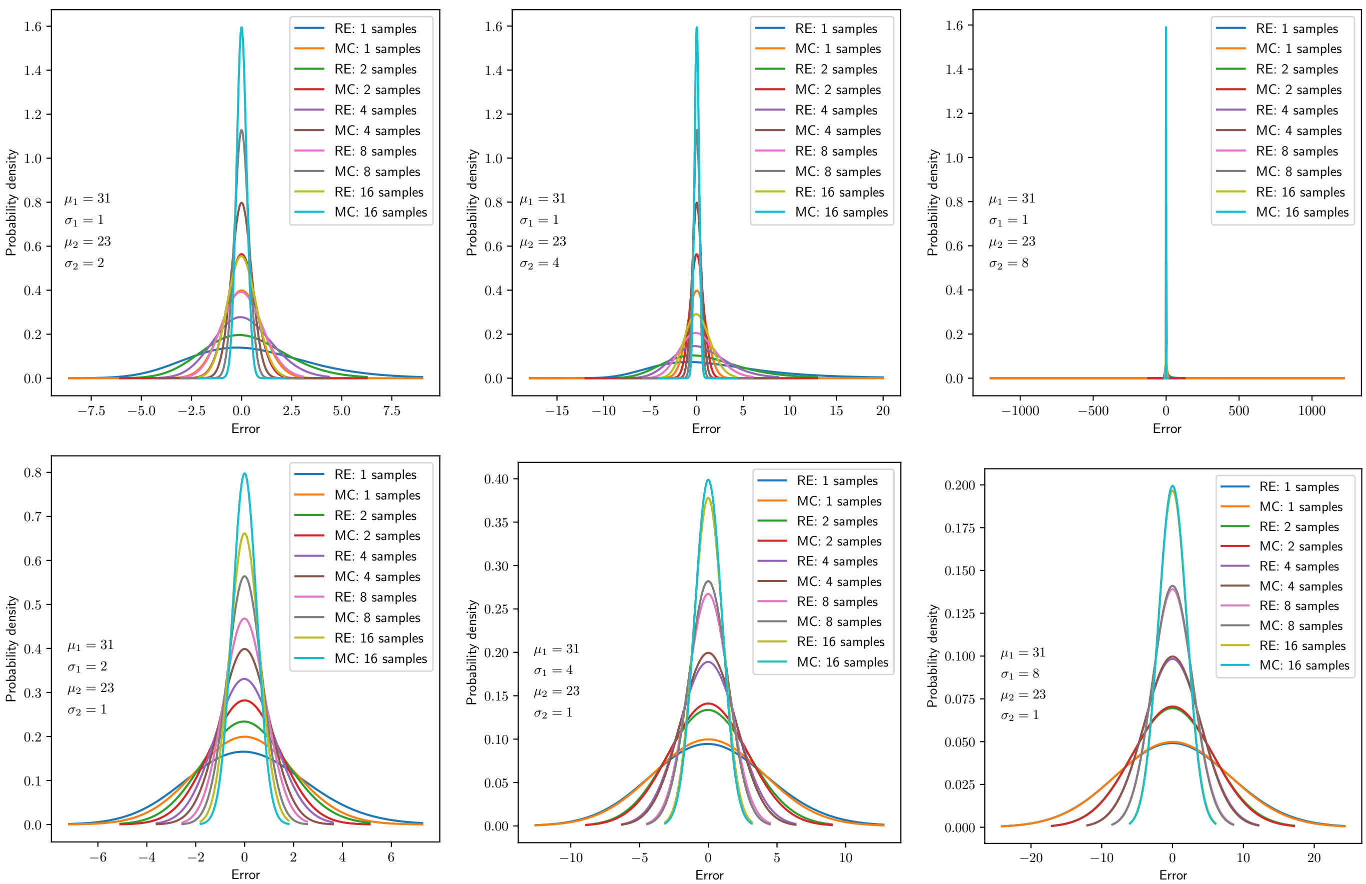


Fig. 1. Monte Carlo (MC) and Ratio estimator (RE) errors depending on the number of samples (analytic, independent distributions). Horizontal axis is probability density and vertical axis represents error in units of radiance.

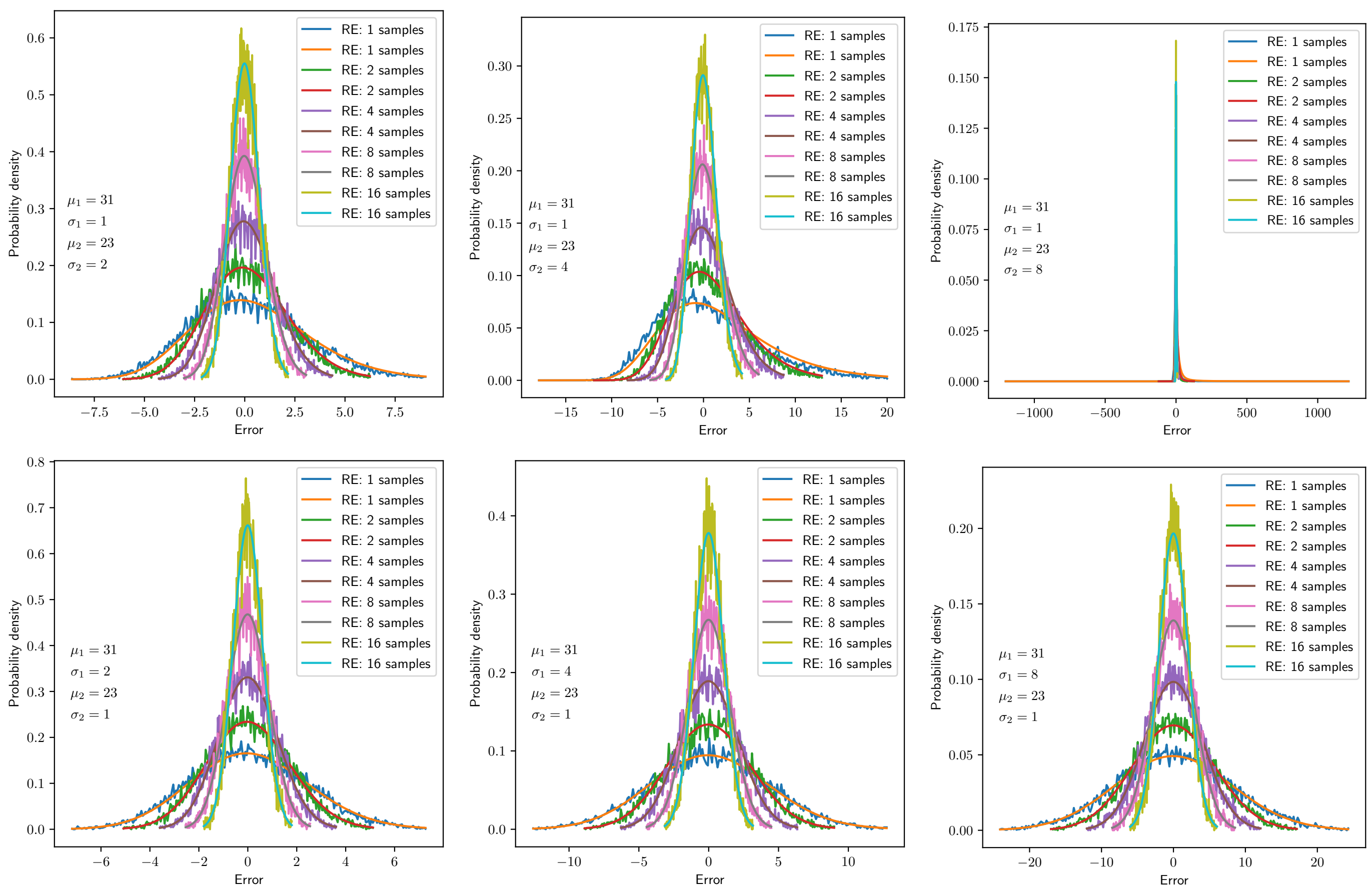
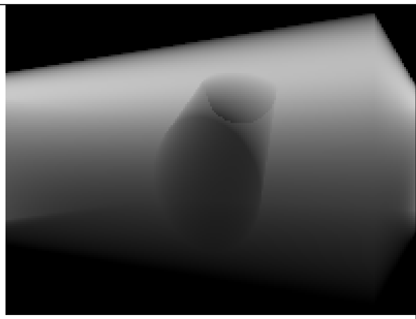
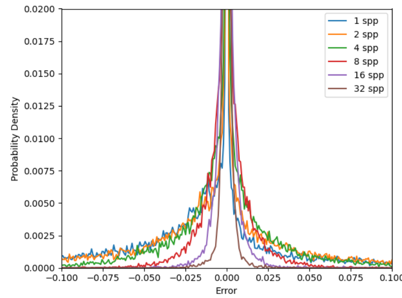
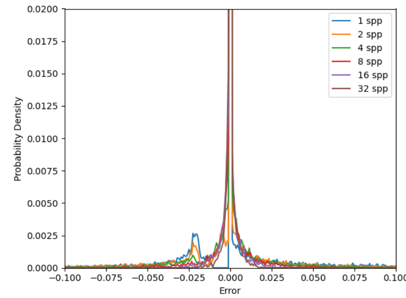
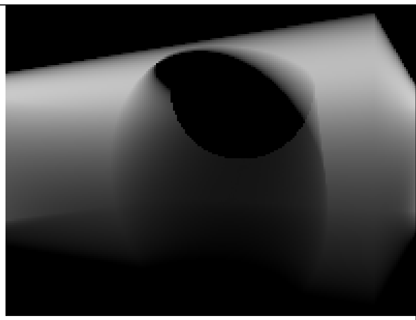
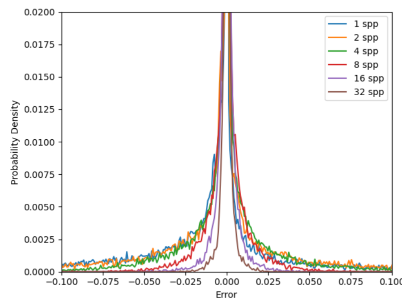
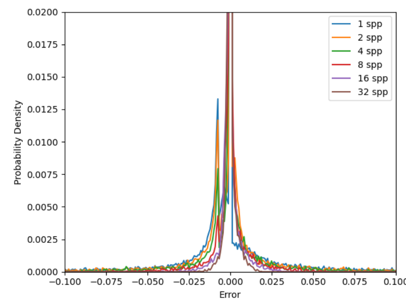
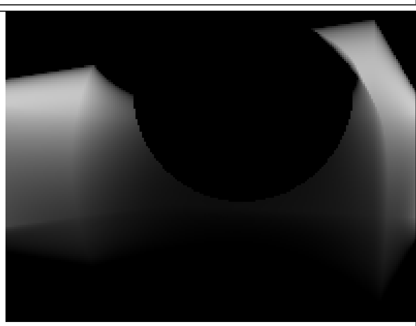
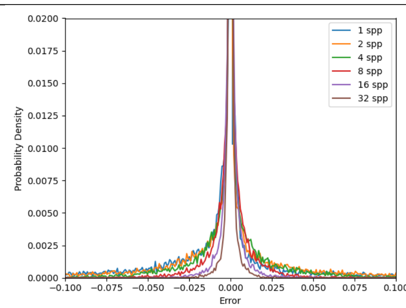
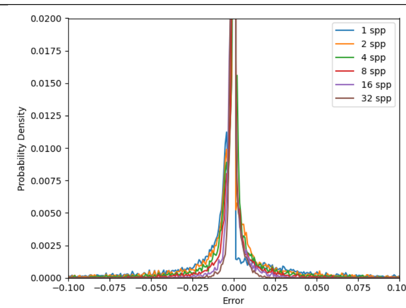
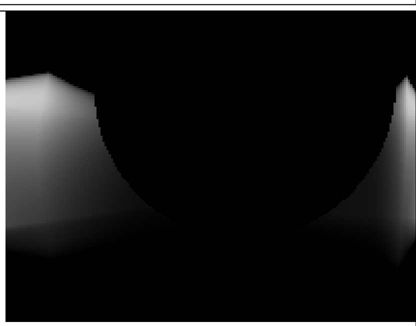
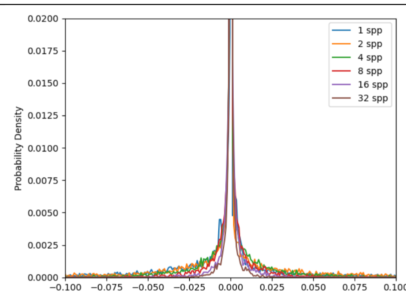
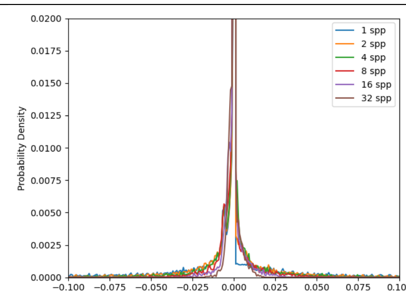
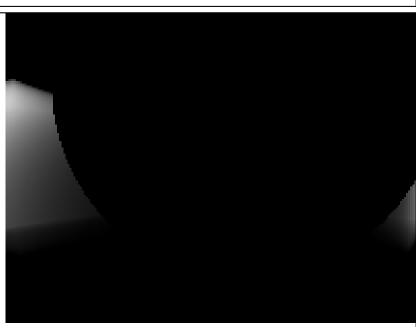
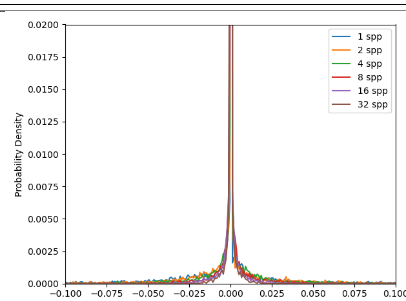
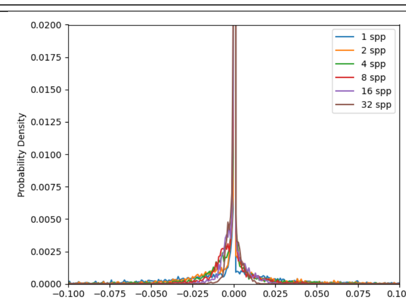
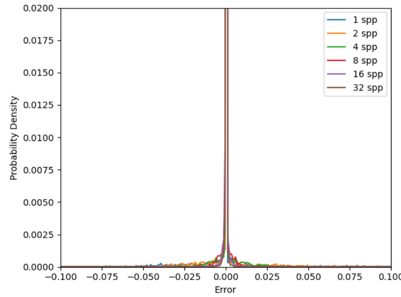


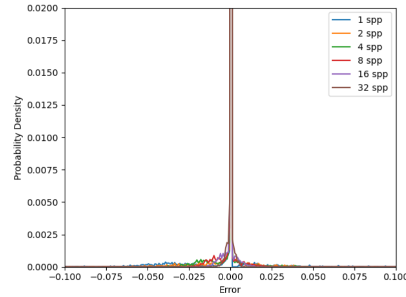
Fig. 2. Stochastic validation of Ratio estimator (RE) errors depending on the number of samples (independent distributions). Horizontal axis is probability density and vertical axis represents error in units of radiance.

TABLE 1: Converge behavior.

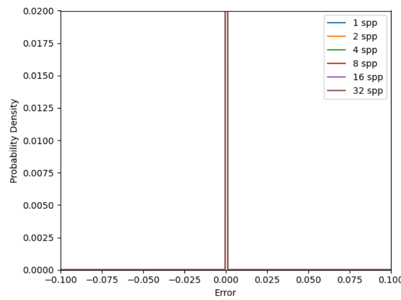
<table>
<tr><th>Sphere Size</th><th>Rendering (Reference)</th><th>RM (Convergence)</th><th>REDS (Convergence)</th></tr>
<tr><td>0.5</td><td></td><td></td><td></td></tr>
<tr><td>1</td><td></td><td></td><td></td></tr>
<tr><td>1.5</td><td></td><td></td><td></td></tr>
<tr><td>2</td><td></td><td></td><td></td></tr>
<tr><td>2.5</td><td></td><td></td><td></td></tr>
</table>

3


0.0200
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
Probability Density
−0.100 −0.075 −0.050 −0.025 0.000 0.025 0.050 0.075 0.100
Error
1 spp
2 spp
4 spp
8 spp
16 spp
32 spp


0.0200
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
Probability Density
−0.100 −0.075 −0.050 −0.025 0.000 0.025 0.050 0.075 0.100
Error
1 spp
2 spp
4 spp
8 spp
16 spp
32 spp

3.5


0.0200
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
Probability Density
−0.100 −0.075 −0.050 −0.025 0.000 0.025 0.050 0.075 0.100
Error
1 spp
2 spp
4 spp
8 spp
16 spp
32 spp

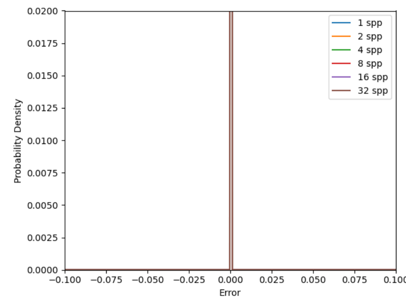
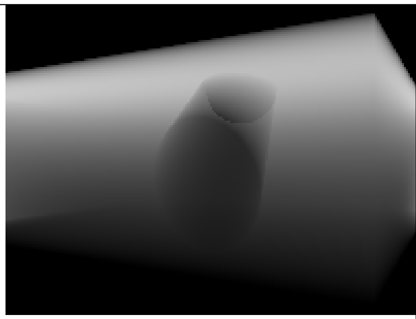
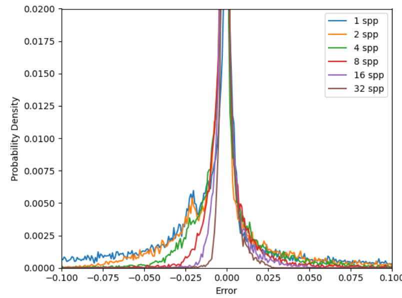
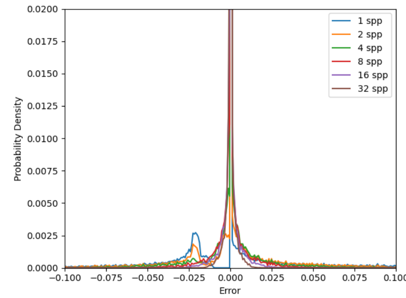
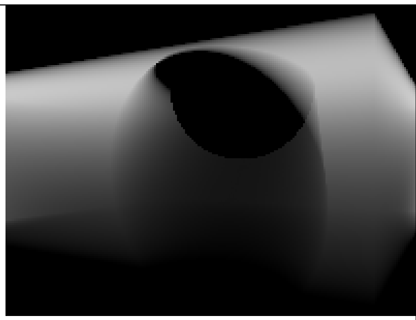
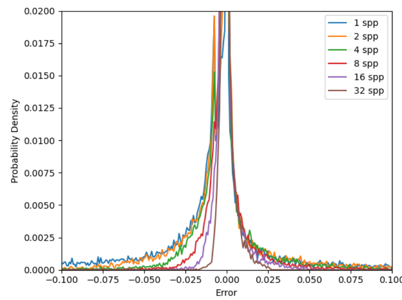
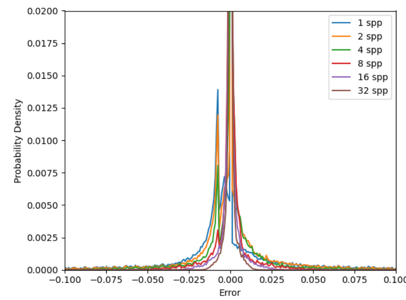
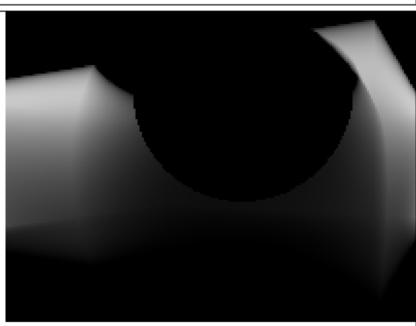
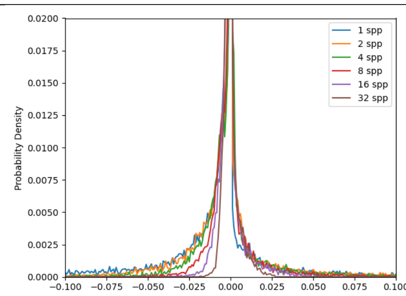
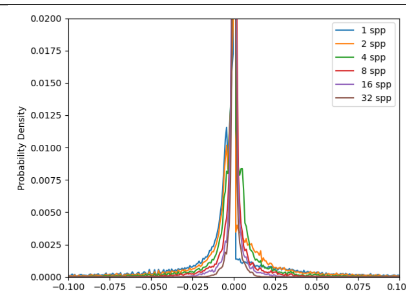
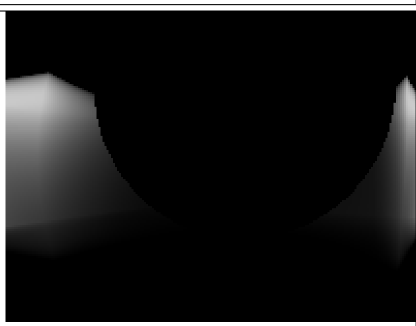
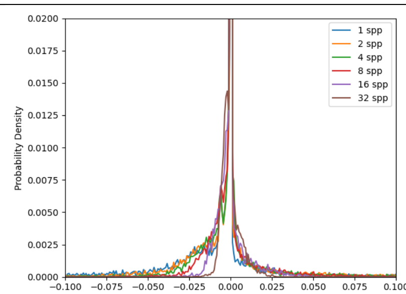
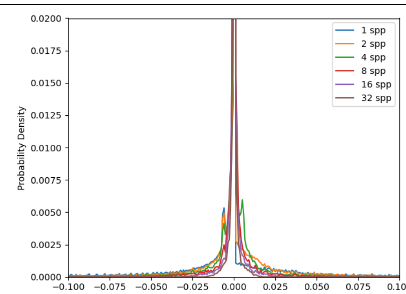
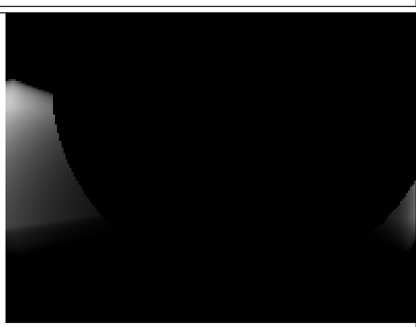
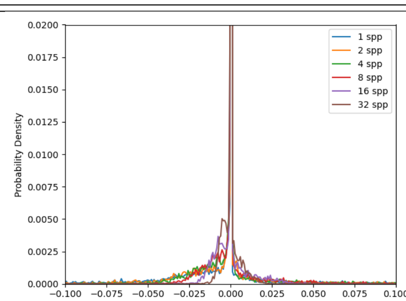
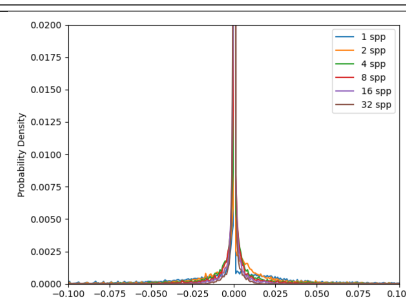

0.0200
0.0175
0.0150
0.0125
0.0100
0.0075
0.0050
0.0025
0.0000
Probability Density
−0.100 −0.075 −0.050 −0.025 0.000 0.025 0.050 0.075 0.100
Error
1 spp
2 spp
4 spp
8 spp
16 spp
32 spp

TABLE 2: Converge behavior.

Sphere Size	Rendering (Reference)	DS (Convergence)	WVDS (Convergence)
0.5			
1			
1.5			
2			
2.5			

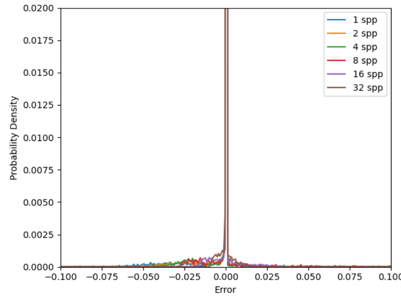
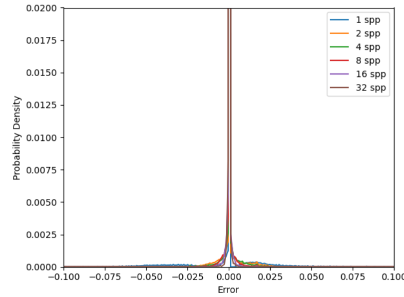
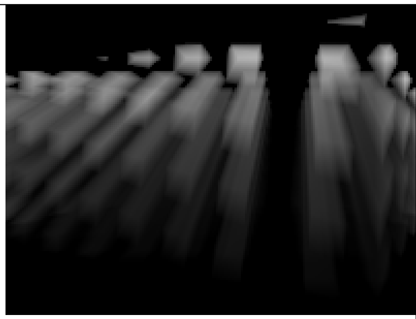
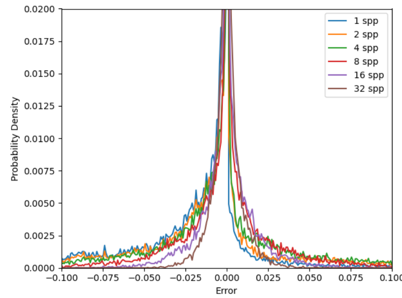
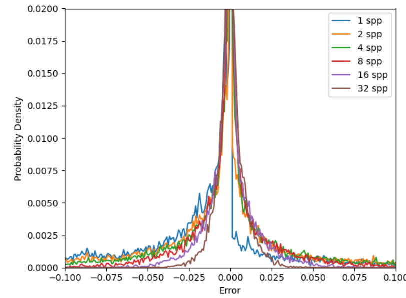
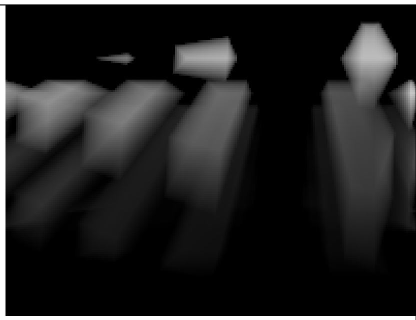
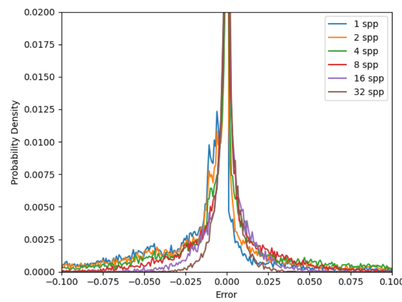
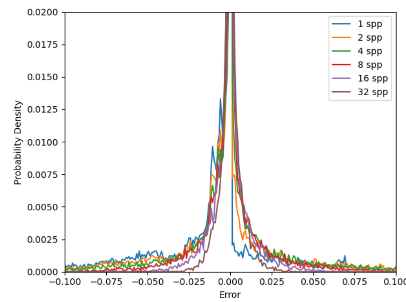
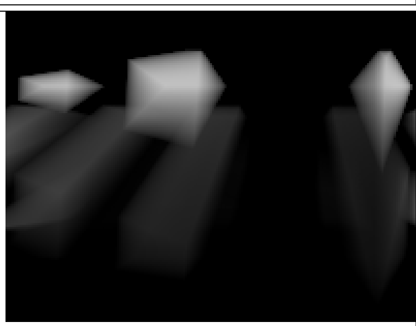
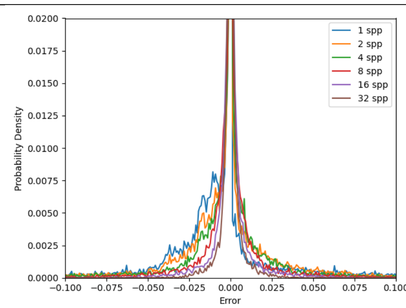
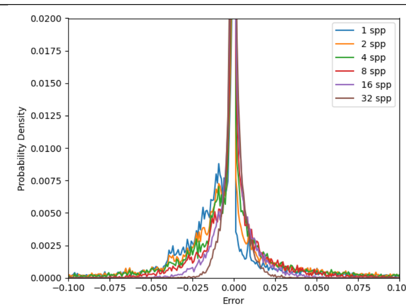
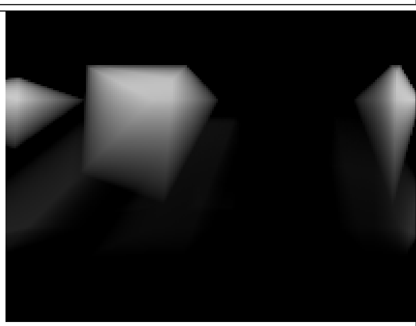
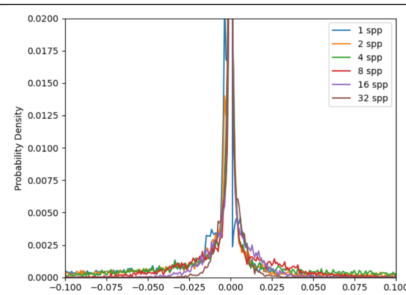
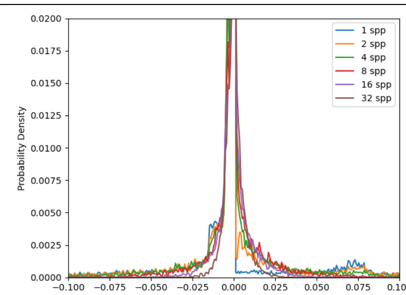
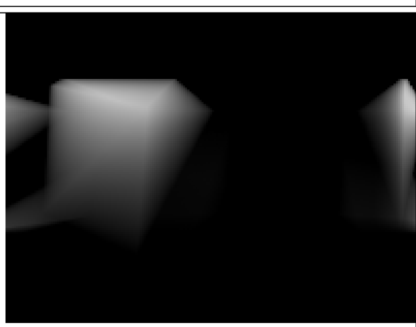
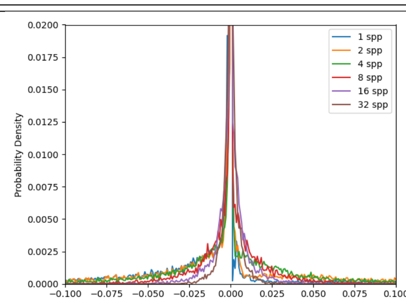
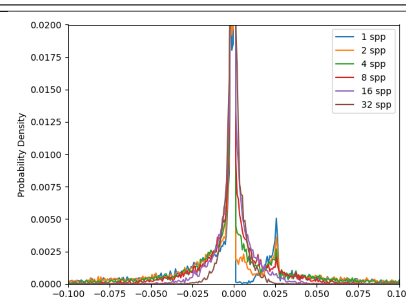
3		Probability Density Error 1 spp 2 spp 4 spp 8 spp 16 spp 32 spp	Probability Density Error 1 spp 2 spp 4 spp 8 spp 16 spp 32 spp

TABLE 3: Converge behavior.

Grid Size	Rendering (Reference)	RM (Convergence)	REDS (Convergence)
0.5			
1			
1.5			
2			
2.5			

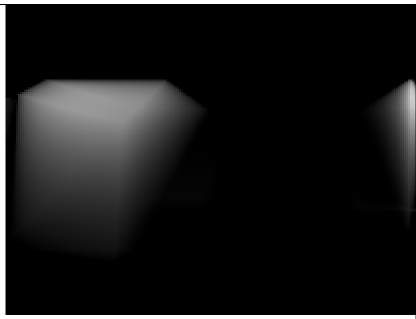
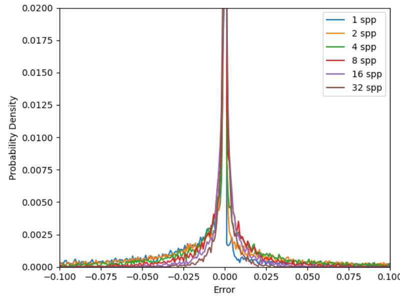
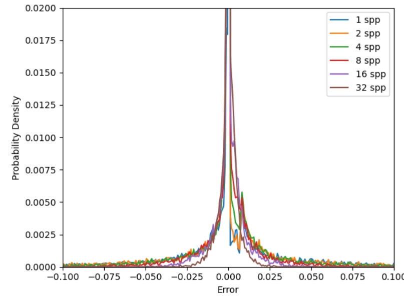
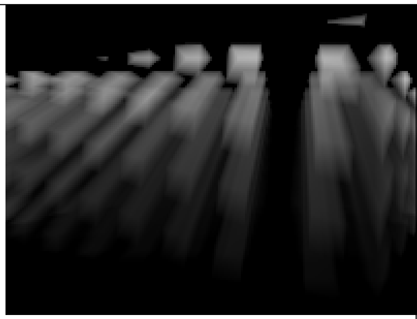
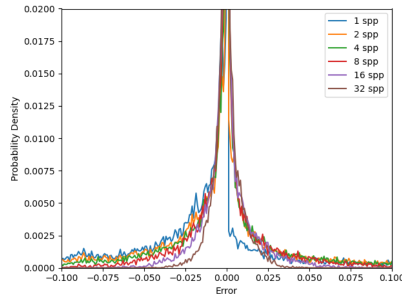
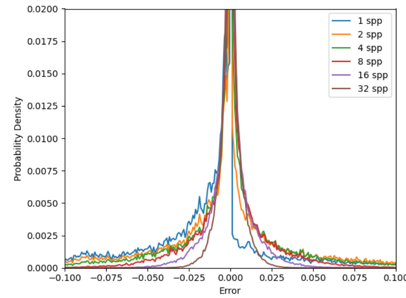
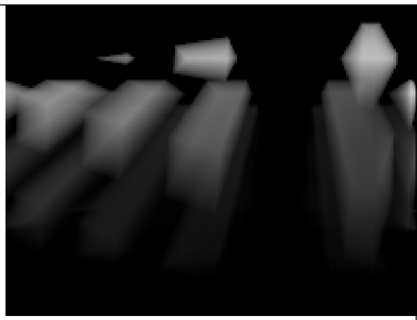
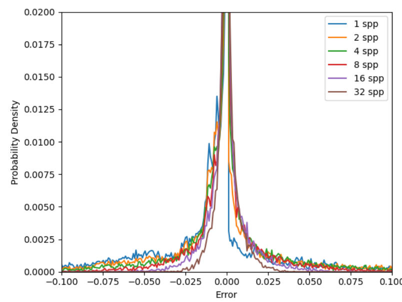
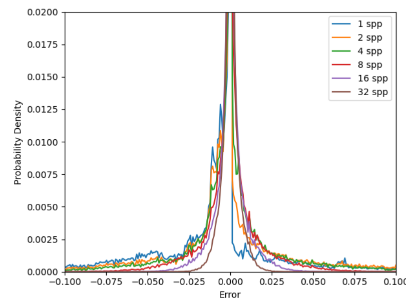
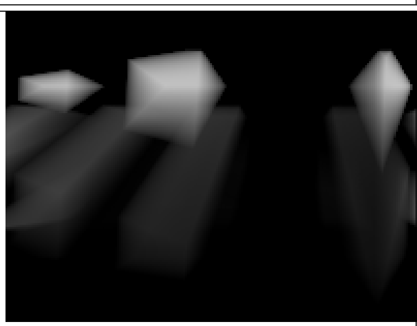
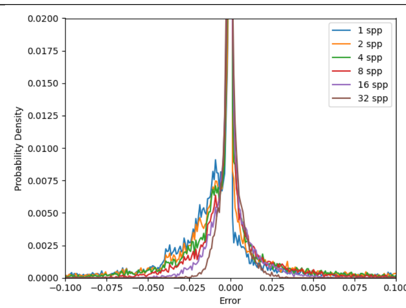
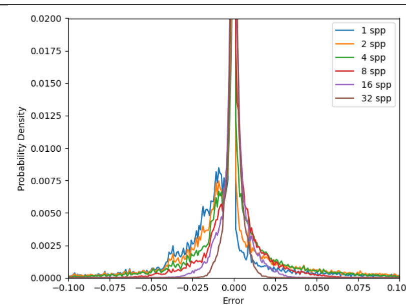
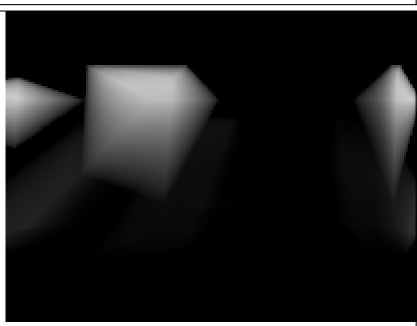
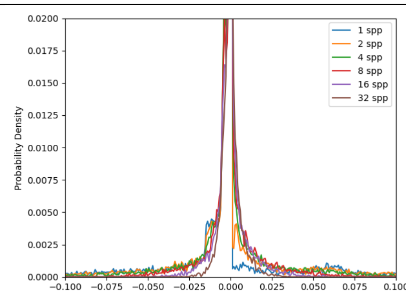
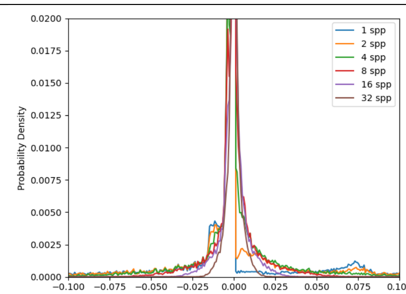
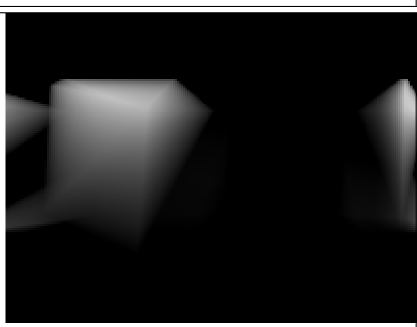
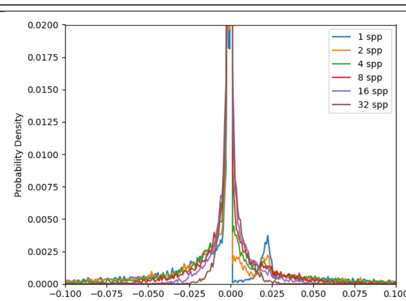
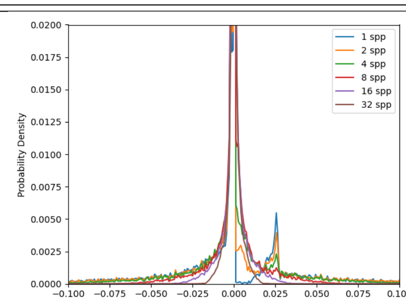
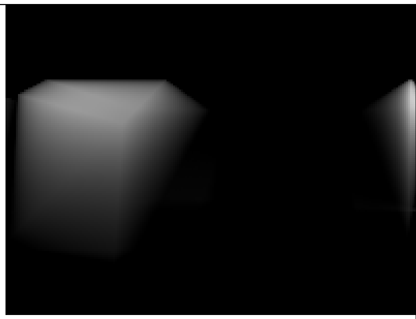
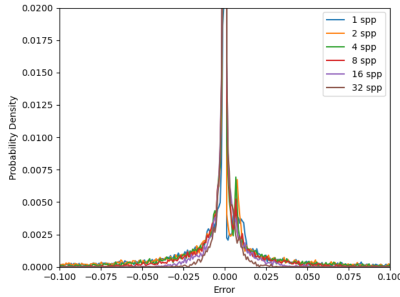
3		 Probability Density Error 1 spp 2 spp 4 spp 8 spp 16 spp 32 spp	 Probability Density Error 1 spp 2 spp 4 spp 8 spp 16 spp 32 spp

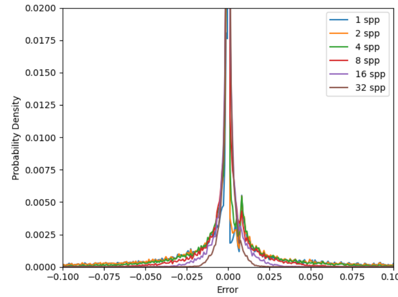
TABLE 4: Converge behavior.

Grid Size	Rendering (Reference)	DS (Convergence)	WVDS (Convergence)
0.5			
1			
1.5			
2			
2.5			

3




Probability Density
Error
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
0.0150
0.0175
0.0200
−0.100 −0.075 −0.050 −0.025 0.000 0.025 0.050 0.075 0.100
1 spp
2 spp
4 spp
8 spp
16 spp
32 spp


Probability Density
Error
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
0.0150
0.0175
0.0200
−0.100 −0.075 −0.050 −0.025 0.000 0.025 0.050 0.075 0.100
1 spp
2 spp
4 spp
8 spp
16 spp
32 spp

REFERENCES

[1] M. Pharr, W. Jakob, and G. Humphreys, *Physically Based Rendering: From Theory to Implementation*, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2016.

[2] C. Peters, C. Münstermann, N. Wetzstein, and R. Klein, "Improved moment shadow maps for translucent occluders, soft shadows and single scattering," *Journal of Comput. Graph. Techniques (JCGT)*, vol. 6, no. 1, March 2017.