Supplemental: On Learning the Best Local Balancing Strategy

D. Murray[1] and S. Benzait[1] and R. Pacanowski[2] and X. Granier[2,3]

1 - INRIA 2 - LP2N (IOGS, CNRS, U. Bordeaux) 3 - ARCHEOVISION (CNRS, U. Bordeaux, U. Bordeaux Montaigne)

This supplemental material is organized as follows. In Section 1, we present our 4 test scenes that were also used in the main paper. In Section 2, we provide, through many graphics, an in-depth analysis of the Monte-Carlo estimator efficiency compared to the previous techniques from Lu et al. [LPG13] and Sbert et al. [SHSK19]. In Section 3, we show different rendered images as well as the computed alpha-map with respect to the number of learning samples and iterations. In section 4, we provide the full mathematical derivation to compute α for two and three importance sampling strategies. Finally, a supplemental video illustrates, for two strategies, that our approach converges as long as $\alpha_0 \in [0.1, 0.9]$.

1. Test Scenes

We present our results for four types of scenes:

1. Classical Cornell box scene (cf. Figure 1a). The classical *indoor* scene (i.e., the Cornell box, from [Bit16]) where direct illumination is predominant but where some part of the scene (the cube faces pointing toward the walls) is mainly lit by indirect illumination.
2. Lucy scene (cf. Figure 1b) is an outdoor scene where one object (the Lucy statue from [Sta]) is laid on a table and is mainly lit by direct illumination. This type of scene is typical from the ones that were tested by Lu et al. [LPG13] and Sbert et al. [SHSK19]
3. Strong Indirect lighting scene (cf. Figure 1c) presents a modified version of the Cornell box scene where the ceiling light direction is reversed; therefore creating a scene where indirect illumination dominates.
4. Bathroom scene (cf. Figure 1d) is an indoor scene with many different types light-paths (highly specular, glossy, diffuse). This scene is a slightly modified version from the "Salle de bain" scene [McG17].

The materials composing these scenes are depicted in Figure 1, along with their rendering using direct and global illumination.

2. Estimator Efficiency

In this section we present additional results in terms of estimator efficiency with respect to the number of learning samples and the number of iterations used to optimize the α value. For each graphic shown in Figures 2,3 and 4, the bottom left point represents the efficiency obtained when using the balanced heuristic. All efficiency values are computed by averaging on the whole image the efficiency computed per-pixel , using the definition from [Vea97]:

$$\varepsilon(F_N) = \frac{1}{V[F_N]T[F_N]} \tag{1}$$

Where F_N is the estimator, $V[F_N]$, the average variance of the estimator over the whole image and $T[F_N]$ the total rendering time.

An overview of the these values (ε, V, T) is presented in Figures 6, 7, 8 and 9 for the different scenes and a few different learning configurations.

3. Detailed Rendering Results

3.1. Two sampling strategies

In this section, we present for each test scene the resulting rendered images and the computed alpha-map for a different number of iterations to optimize α. Unless said otherwise the number of learning samples is fixed to 128 per iteration for the figures presented in this section. For the balance map, a red color represents mostly light sampling strategy whereas a green color represents mostly BRDF sampling strategy. A yellow color represent equal distribution the two strategies. Remember that computed α values are clamped between 0.1 and 0.9 to avoid being restricted to one sampling technique only.

3.2. Three sampling strategies

In this section we present the balance map when using three sampling strategies analysis for the bathroom scene (cf. Figures 9). A red color represents full light sampling strategy, a green color represents full BRDF sampling strategy, a blue color represents an orthogonal sampling strategy (e.g., uniform sampling) and a white color represents the equal balance. The full color mapping is depicted on Figure 5.

4. Complete Mathematical Formulation of the $alpha$ computation

4.1. Formulation for 2 strategies

As pointed by Georgiev et al. [GKPS12], the MIS estimator between BRDF-based sampling (p_{BRDF}) and light-based sampling (p_{light}) of the Rendering Equation [Kaj86] can be rewritten in terms of Defensive importance sampling. For the case of two sampling strategies the MIS estimator becomes

$$F_N(\alpha_n) = \frac{1}{N}\sum_{i=1}^{N} \frac{\mathrm{f}(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega}_i)}{\mathrm{p}_{\alpha_n}(\boldsymbol{\omega}_i)}, \tag{2}$$

where α_n is the balancing factor as step n, $\mathrm{f}(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega}) = \rho(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega})(\boldsymbol{\omega}\cdot\boldsymbol{n})\mathrm{L}(\boldsymbol{x}\leftarrow\boldsymbol{\omega})$, $\boldsymbol{n}$ is the normal of the surface at $\boldsymbol{x}$, $\Omega_{\boldsymbol{n}}$ is the unit hemisphere, $\rho(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega})$ is the reflectance function (BRDF), and $\mathrm{L}(\boldsymbol{\omega}\rightarrow\boldsymbol{x})$ is the incoming radiance at position $\boldsymbol{x}$ from direction $\boldsymbol{\omega}$. The PDF p_α is

$$\mathrm{p}_{\alpha_n} = \alpha_n \mathrm{p}_{\mathrm{BRDF}} + (1-\alpha_n)\mathrm{p}_{\mathrm{light}}. \tag{3}$$

When computing many independent samples, the variance is $\mathrm{V}[F_N] = \mathrm{V}[F_1]/N$. Minimizing $\mathrm{V}[F_N]$ is then similar to minimizing $\mathrm{V}[F_1]$. In the following, we use $\mathrm{V}[F]$ for $\mathrm{V}[F_1]$ to simplify the notation.

As demonstrated by Sbert at al. [SHSK19], the first and second derivatives according to α of the variance $\mathrm{V}[F]$ are

$$\frac{d\mathrm{V}[F](\alpha_n)}{d\alpha_n} = \int_{\Omega_{\boldsymbol{n}}} \frac{\mathrm{f}^2(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega})}{\mathrm{p}^2_{\alpha,n}}(\mathrm{p}_{\mathrm{BRDF}} - \mathrm{p}_{\mathrm{light}})d\boldsymbol{\omega} \tag{4}$$

$$\frac{d^2\mathrm{V}[F](\alpha_n)}{d\alpha^2} = 2\int_{\Omega_{\boldsymbol{n}}} \frac{\mathrm{f}^2(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega})}{\mathrm{p}^2_{\alpha,n}}(\mathrm{p}_{\mathrm{BRDF}} - \mathrm{p}_{\mathrm{light}})^2 d\boldsymbol{\omega}. \tag{5}$$

They may be estimated with $M < N$ samples [LPG13] with

$$\frac{d\mathrm{V}[F](\alpha_n)}{d\alpha_n} = \frac{1}{M}\sum_{i}^{M} \frac{\mathrm{f}^2(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega}_i)}{\mathrm{p}^2_{\alpha_n}} \frac{(\mathrm{p}_{\mathrm{BRDF}} - \mathrm{p}_{\mathrm{light}})}{\mathrm{p}_{\alpha_n}} \tag{6}$$

$$\frac{d^2\mathrm{V}[F](\alpha_n)}{d\alpha_n^2} = \frac{2}{M}\sum_{i}^{M} \frac{\mathrm{f}^2(\boldsymbol{x},\boldsymbol{o},\boldsymbol{\omega}_i)}{\mathrm{p}^2_{\alpha_n}} \frac{(\mathrm{p}_{\mathrm{BRDF}} - \mathrm{p}_{\mathrm{light}})^2}{\mathrm{p}^2_{\alpha_n}}. \tag{7}$$

4.2. Extension to S strategies

For S sampling strategies, the resulting PDF is a combination of each strategy's PDF, weighted by their respective weight $\alpha_{i=1..S}$ such that

$$p = \sum_{i}^{S} \alpha_i p_i, \tag{8}$$

meaning that we must find the optimal value for these S weights. But since

$$\sum_{i}^{S} \alpha_i = 1, \tag{9}$$

we can reduce the number of weights to find, from S to $S-1$, as the last one (α_S) can be deduced from the others α_i, with

$$\alpha_S = 1 - \sum_{i}^{S-1} \alpha_i. \tag{10}$$

For clarity, we now note $\boldsymbol{\alpha} = \alpha_{i=1..S-1}$ and $\boldsymbol{\alpha}_n$ for the iteration n. The second order approximation of the variance (or the first order approximation of the derivative of the variance) gives the following formulation for S strategies:

$$\nabla\mathrm{V}[F](\boldsymbol{\alpha}) \approx \nabla_{\boldsymbol{\alpha}}\mathrm{V}[F](\boldsymbol{\alpha}_n) + \mathcal{H}(\mathrm{V}[F](\boldsymbol{\alpha}_n)) \begin{bmatrix} \alpha_0 - \alpha_{0,n} \\ \vdots \\ \alpha_{S-1} - \alpha_{S-1,n} \end{bmatrix}, \tag{11}$$

where $\mathcal{H}(V[F](\alpha_i))$ is the $(M-1)\times(M-1)$ Hessian matrix of the variance. Note that in our case, $\mathcal{H}$ is symmetrical:

$$\mathcal{H}(\mathrm{V}[F](\boldsymbol{\alpha}_n)) = \begin{bmatrix} \frac{\partial^2\mathrm{V}}{\partial\alpha^2_{0,n}} & \cdots & \frac{\partial^2\mathrm{V}}{\partial\alpha_{0,n}\partial\alpha_{M-1,n}} \\ \vdots & \ddots & \\ \frac{\partial^2\mathrm{V}}{\partial\alpha_{S-1,n}\partial\alpha_{0,n}} & & \frac{\partial^2\mathrm{V}}{\partial\alpha^2_{S-1,n}} \end{bmatrix}. \tag{12}$$

For the 3 strategies, p reduces to

$$\mathrm{p} = \alpha\mathrm{p}_1 + \beta\mathrm{p}_2 + (1-\alpha-\beta)\mathrm{p}_3 \tag{13}$$

and the system to be solved becomes

$$\nabla\mathrm{V}[F](\alpha,\beta) \approx \nabla\mathrm{V}[F](\alpha_n,\beta_n) + \mathcal{H}(\mathrm{V}[F](\alpha_n,\beta_n)) \begin{bmatrix} \alpha-\alpha_n \\ \beta-\beta_n \end{bmatrix} \tag{14}$$

where

$$\mathcal{H}(\mathrm{V}[F](\alpha_n,\beta_n)) = \begin{bmatrix} \frac{\partial^2\mathrm{V}}{\partial\alpha^2} & \frac{\partial^2\mathrm{V}}{\partial\alpha\partial\beta} \\ \frac{\partial^2\mathrm{V}}{\partial\beta\partial\alpha} & \frac{\partial^2\mathrm{V}}{\partial\beta^2} \end{bmatrix}, \tag{15}$$

and the different derivatives are

$$\frac{\partial\mathrm{V}}{\partial\alpha} = \int \frac{\mathrm{f}(\boldsymbol{\omega})^2(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_1(\boldsymbol{\omega}))}{\mathrm{p}^2(\boldsymbol{\omega})} d\boldsymbol{\omega} \tag{16}$$

$$\frac{\partial V}{\partial\beta} = \int \frac{\mathrm{f}(\boldsymbol{\omega})^2(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_2(\boldsymbol{\omega}))}{\mathrm{p}^2(\boldsymbol{\omega})} d\boldsymbol{\omega} \tag{17}$$

$$\frac{\partial^2\mathrm{V}}{\partial\alpha^2} = 2\int \frac{\mathrm{f}(\boldsymbol{\omega})^2(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_1(\boldsymbol{\omega}))^2}{\mathrm{p}^3(\boldsymbol{\omega})} d\boldsymbol{\omega} \tag{18}$$

$$\frac{\partial^2\mathrm{V}}{\partial\beta^2} = 2\int \frac{\mathrm{f}(\boldsymbol{\omega})^2(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_2(\boldsymbol{\omega}))^2}{\mathrm{p}^3(\boldsymbol{\omega})} d\boldsymbol{\omega} \tag{19}$$

$$\frac{\partial^2\mathrm{V}}{\partial\alpha\partial\beta} = 2\int \frac{\mathrm{f}(\boldsymbol{\omega})^2(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_1(\boldsymbol{\omega}))(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_2(\boldsymbol{\omega}))}{\mathrm{p}^3(\boldsymbol{\omega})} d\boldsymbol{\omega} \tag{20}$$

$$\frac{\partial^2 V}{\partial\beta\partial\alpha} = \frac{\partial^2 V}{\partial\alpha\partial\beta}. \tag{21}$$

The last equality is ensured when the variance function is continuous according to α and β.

To ensure that we minimize the variance, we must ensure that

$$\nabla\mathrm{V}[F](\alpha,\beta) = \begin{bmatrix} \mathrm{V}'_\alpha[F](\alpha,\beta) \\ \mathrm{V}'_\beta[F](\alpha,\beta) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{22}$$

$$\mathrm{determinant}(\mathcal{H}(\mathrm{V}[F](\alpha,\beta))) > 0 \tag{23}$$

$$\mathrm{trace}(\mathcal{H}(\mathrm{V}[F](\alpha,\beta))) > 0. \tag{24}$$

This is straightforward for trace$(\mathcal{H})$ as $\frac{\partial^2 \mathrm{V}}{\partial \alpha^2}$ and $\frac{\partial^2 \mathrm{V}}{\partial \beta^2}$ only contain positive terms.

We are also in a case where determinant$(\mathcal{H}) > 0$ due to Cauchy-Schwarz inequality. The demonstration can be done quickly as follows:

If

$$u = \frac{\mathrm{f}(\boldsymbol{\omega})(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_1(\boldsymbol{\omega}))}{\mathrm{p}^{3/2}(\boldsymbol{\omega})} \tag{25}$$

$$v = \frac{\mathrm{f}(\boldsymbol{\omega})(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_2(\boldsymbol{\omega}))}{\mathrm{p}^{3/2}(\boldsymbol{\omega})}, \tag{26}$$

then

$$\frac{\partial^2 \mathrm{V}}{\partial \alpha^2} = 2\int u^2 \tag{27}$$

$$\frac{\partial^2 \mathrm{V}}{\partial \beta^2} = 2\int v^2 \tag{28}$$

$$\frac{\partial^2 \mathrm{V}}{\partial \alpha \partial \beta} = 2\int uv, \tag{29}$$

so

$$\text{determinant}(\mathcal{H}) = 4\left(\int u^2 \int v^2 - \left(\int uv\right)^2\right). \tag{30}$$

The Cauchy-Schwarz inequality directly ensures that

$$\left(\int uv\right)^2 \leqslant \int u^2 \int v^2, \tag{31}$$

so

$$\text{determinant}(\mathcal{H}) \geqslant 0. \tag{32}$$

Now we must ensure that

$$\left(\int uv\right)^2 \neq \int u^2 \int v^2 \tag{33}$$

so that determinant$(\mathcal{H}) \neq 0$. There are two situations where determinant$(\mathcal{H}) = 0$. The first one is if the integrals are estimated with a low number of samples and none of theses samples contribute to the final image, thus $u = v = 0$. So as long as we use a sufficient number of learning samples, we should avoid this situation. The second case is given by definition of the Cauchy-Schwarz inequality: this situation occurs if there exists a constant value λ, independent of $\boldsymbol{\omega}$, so that for any $\boldsymbol{\omega}$

$$\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_1(\boldsymbol{\omega}) = \lambda(\mathrm{p}_3(\boldsymbol{\omega}) - \mathrm{p}_2(\boldsymbol{\omega})) \tag{34}$$

$\iff$

$$\mathrm{p}_3(\boldsymbol{\omega}) = \frac{1}{1-\lambda}(\mathrm{p}_1(\boldsymbol{\omega}) - \lambda \mathrm{p}_2(\boldsymbol{\omega})). \tag{35}$$

Therefore, as long as we can avoid this situation by using three sampling strategies which are not correlated, we are sure that

$$\text{determinant}(\mathcal{H}) > 0. \tag{36}$$

In the case where one uses light, BRDF and uniform sampling, this inequality is verified as neither of them are correlated.

When all our conditions are verified, to obtain the solution for α_{n+1} and β_{n+1}, we just have to solve

$$\begin{bmatrix}\alpha_{n+1}\\ \beta_{n+1}\end{bmatrix} = \begin{bmatrix}\alpha_n\\ \beta_n\end{bmatrix} - \mathcal{H}^{-1}(\mathrm{V}[F](\alpha_n,\beta_n))\nabla\mathrm{V}[F](\alpha_n,\beta_n). \tag{37}$$

References

[Bit16] Benedikt Bitterli. Rendering resources, 2016. https://benedikt-bitterli.me/resources/. 1

[GKPS12] I. Georgiev, J. Křivánek, S. Popov, and P. Slusallek. Importance caching for complex illumination. In *Computer Graphics Forum*, volume 31, pages 701–710. Wiley Online Library, 2012. 1

[Kaj86] James T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143–150, August 1986. 1

[LPG13] H. Lu, R. Pacanowski, and X. Granier. Second-order approximation for variance reduction in multiple importance sampling. *Computer Graphics Forum*, 32(7):131–136, October 2013. 1, 2, 5, 6, 7, 8, 9, 10

[McG17] Morgan McGuire. Computer graphics archive, July 2017. `https://casual-effects.com/data`. 1

[SHSK19] M. Sbert, V. Havran, and L. Szirmay-Kalos. Optimal deterministic mixture sampling. In *Eurographics (Short Papers)*, pages 73–76, 2019. 1, 2, 5, 6, 7, 8, 9, 10

[Sta] The stanford 3d scanning repository. http://graphics.stanford.edu/data/3Dscanrep/. 1

[Vea97] Eric Veach. *Robust Monte Carlo methods for light transport simulation*, volume 1610. Stanford University PhD thesis, 1997. 1

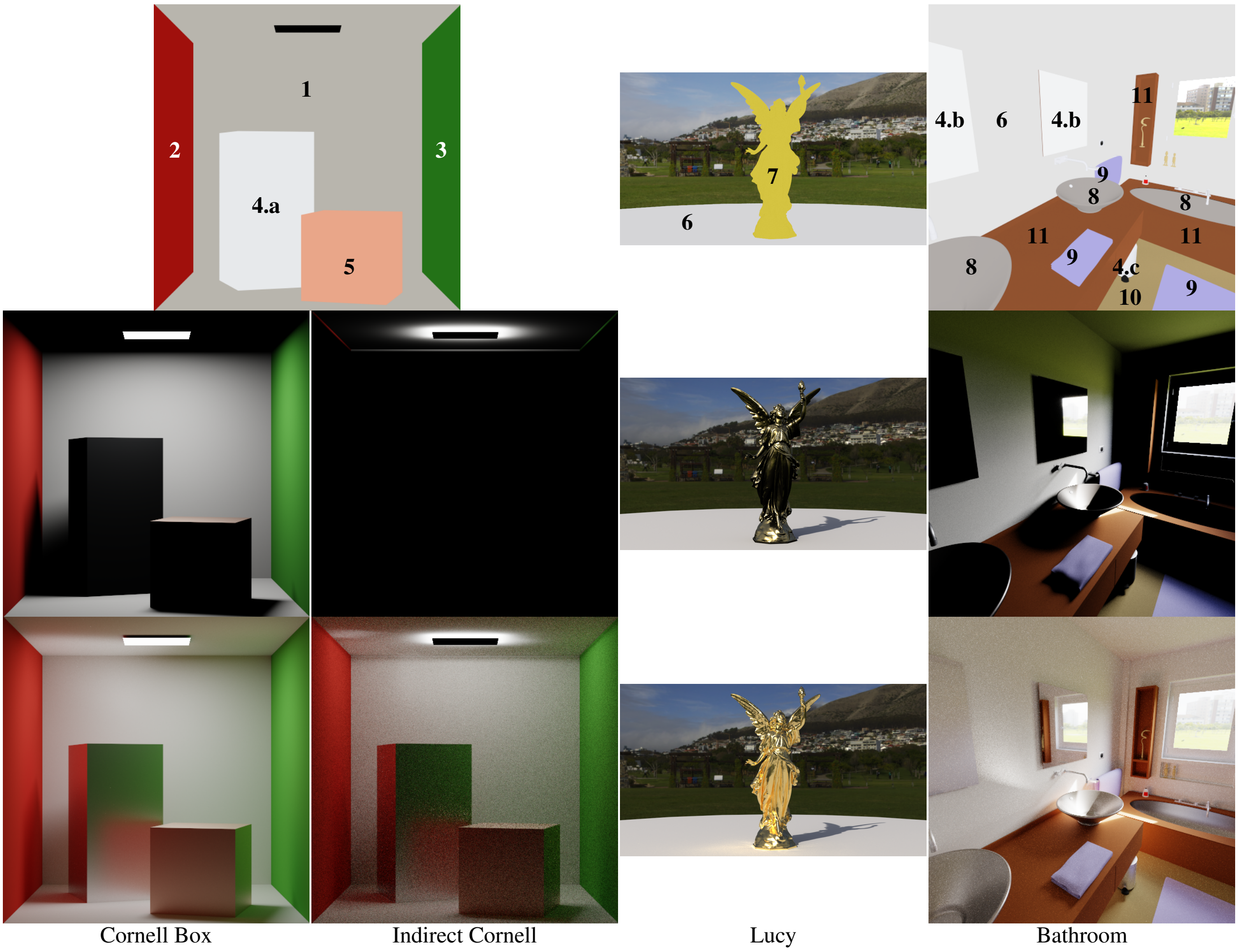
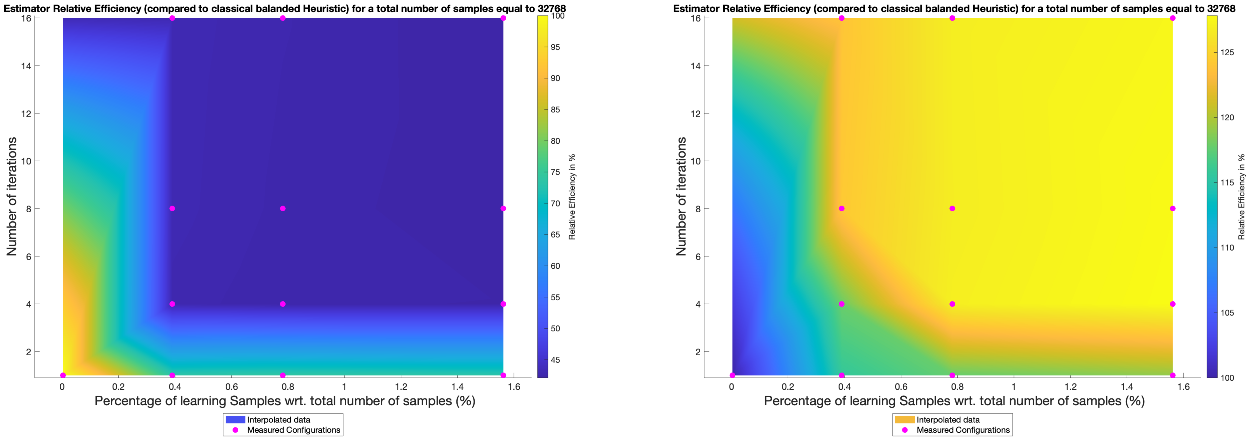


Figure 1: *The four test scenes. **Top row**: the material map for the scenes. The Cornell Box with the reversed light uses the sames materials as the original one. **Middle row**: the four scenes under direct illumination. **Bottom row**: the four scenes under global illumination.*

Type	#	Parameters			Comments
Lambert		ρ_{RGB}			
	1	(0.68, 0.71, 0.725)			
	2	(0.63, 0.065, 0.05)			
	3	(0.14, 0.45, 0.091)			
	6	(0.8, 0.8, 0.8)			
	9	(0.481, 0.457, 0.8)			
GGX		Roughness	η_{RGB}	κ_{RGB}	
	4.a	0.1	(1.5580, 1.0152, 0.63324)	(7.7124, 6.6273, 5.4544)	Aluminium
	4.b	0.001			
	4.c	0.2			
	5	0.2	(0.27527, 0.91110, 1.2404)	(3.3726, 2.6202, 2.3929)	Copper
	7	0.05	(0.11114, 0.29172, 1.4733)	(1.7702, 2.3914, 3.5041)	Gold
	8	0.1	(3.1170, 2.9936, 2.8555)	(2.7282, 2.5065, 2.4078)	Titanium Carbide
	10	0.9	(3.6777, 3.3376, 2.5480)	(7.1884, 4.0247, 1.3820)	
	11	0.65	(3.0272, 2.0552, 1.5506)	(2.6332, 0.62672, 0.17958)	

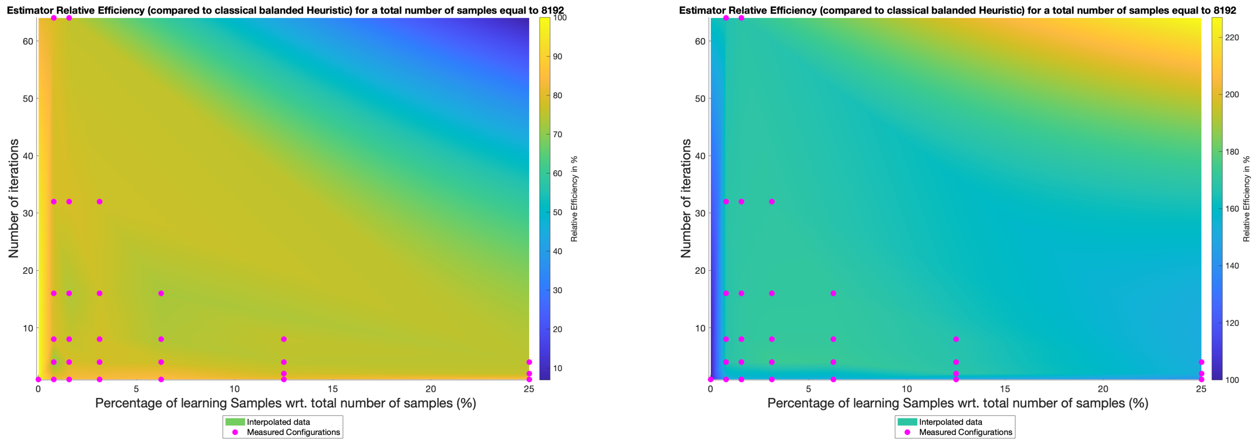
Table 1: *List of materials composing the scenes in Figure 1. In the bathroom scene, small objects not identified explicitly due to their size in the image are either rough gold (#7) or rough aluminium (#4.c). In our implementation, GGX materials are characterized by their RGB refractive indices $n = \eta + i\kappa$, through the Fresnel Term. Some refractive indices (for which the actual material is identified) are extracted from the database https://refractiveindex.info/, the others are custom made.*



(a)

(b)

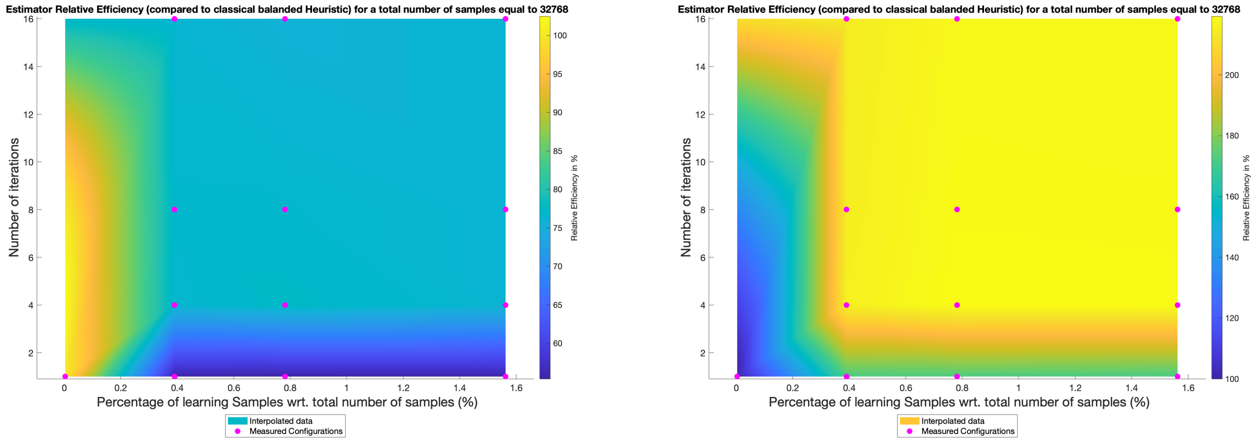
Figure 2: *Cornell Box with metallic boxes Scene. The efficiency of the Monte-Carlo estimator, relative to the balance heuristic with equal allocation between light and BRDF samples. The X-axis is given per iteration.* ***(a)*** *α is computed with direct illumination only vs.* ***(b)*** *global (direct+indirect) illumination.* ***Using our improved version of learning (RIGHT) can achieve 130% efficiency against the balanced heuristic whereas the previous techniques ([LPG13, SHSK19]) achieve only 80%. In other words, for this scene, [LPG13] and [SHSK19] are less efficient than the balanced heuristic.*** *These timings were measured using an Nvidia® RTX 2080 graphics card.*



(a)

(b)

Figure 3: *Lucy scene. Relative efficiency of the Monte-Carlo estimator compared to the balanced heuristic with equal allocation between light and BRDF samples. The X-axis is given per iteration.* ***(a)*** α *is computed with direct illumination only vs.* ***(b)*** *global (direct+indirect) illumination.* ***Using our improved version of learning (RIGHT) can achieve 180% efficiency against the balanced heuristic whereas the previous techniques ([LPG13, SHSK19]) achieve only 90%.*** *These timings were measured using an Nvidia® Titan V graphics card.*



(a)

(b)

Figure 4: *Modified Cornell box with strong indirect lighting. Relative efficiency of the Monte-Carlo Estimator compared to the balanced heuristic with equal allocation between light and BRDF samples. The X-axis is given per iteration.* ***(a)*** α *is computed with direct illumination only vs.* ***(b)*** *global (direct+indirect) illumination.* ***Using our improved version of learning (RIGHT) can achieve 200% efficiency against the balanced heuristic whereas the previous techniques ([LPG13, SHSK19]) achieve only 90%.*** *These timings were measured using an Nvidia® RTX 2080 graphics card.*

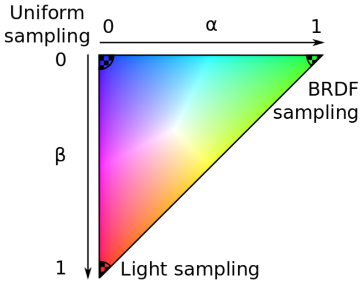


Figure 5: *The color map used to identify the sampling pattern. A red color represents full light sampling strategy, a green color represents full BRDF sampling strategy, a blue color represents an orthogonal sampling strategy (here, uniform sampling) and a white color represents the equal balance. The hatched part represents the values excluded by our clamping* $[0.1;0.9]$. *When using only two strategies, only the diagonal from light to BRDF sampling is used.*

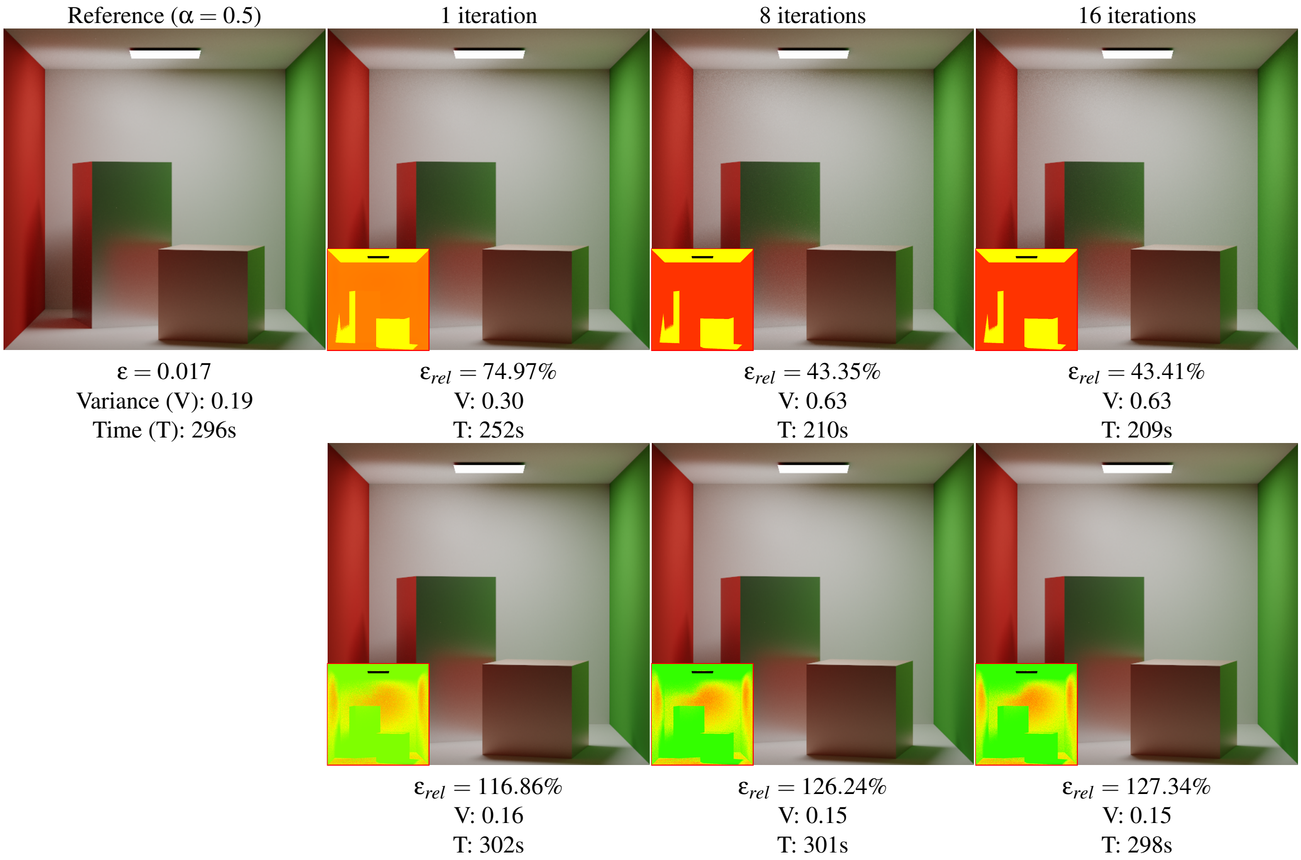


Figure 6: *Comparisons for the Cornell box scene. We compare in terms of relative efficiency, between Sbert et al. [SHSK19] and Lu et al. [LPG13]. The number of learning samples for each technique is set to 128 per iteration and the total number of samples to compute the rendered images is 32768 (learning samples included). The insets show* α *values per pixel where red color means mainly light sampling and green mainly BRDF sampling.* ***Top Part*** *Learning phase limited to direct lighting only [LPG13, SHSK19]. In this case, the optimal solution is to use only one strategy in visible areas. In occluded areas, no strategy outperforms the other due to the limitation to direct lighting.* ***Bottom Part*** *In occluded areas and area lit mainly by indirect paths, the BRDF strategy logically outperforms the light one, whereas in areas mainly lit by direct paths the light strategy outperforms the BRDF one. We can also see that for diffuse material, the transition between one strategy and the other is smooth. All timings were measured on a PC with an Nvidia® Titan V graphics card.*

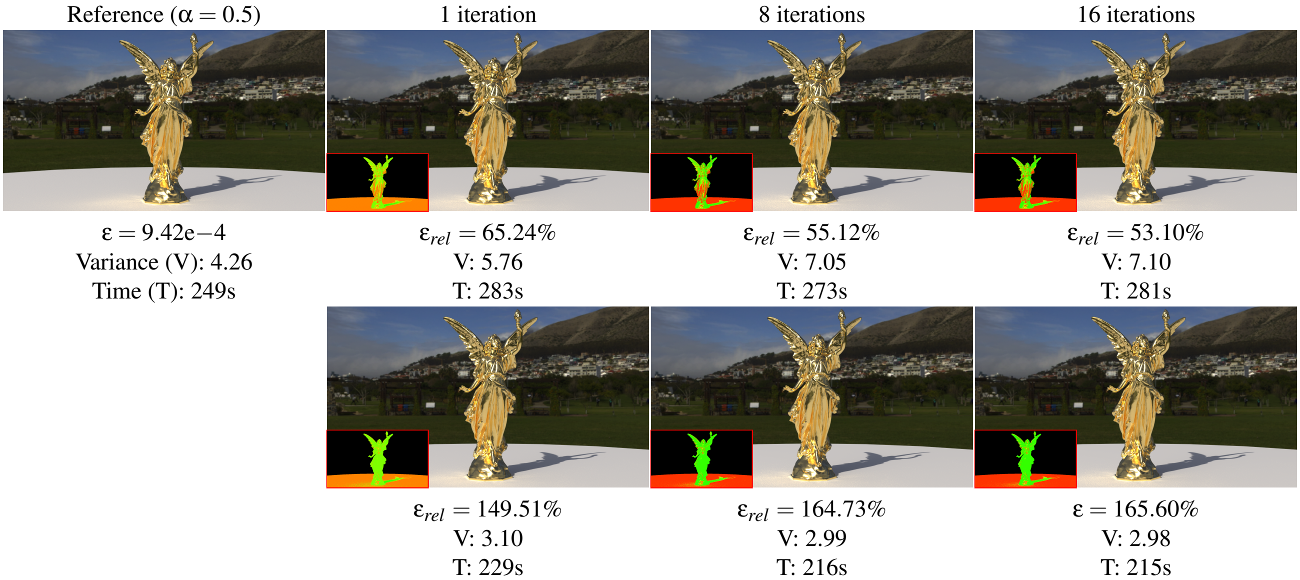

Reference ($\alpha = 0.5$)
1 iteration
8 iterations
16 iterations
$\varepsilon = 9.42\text{e}{-4}$
Variance (V): 4.26
Time (T): 249s
$\varepsilon_{rel} = 65.24\%$
V: 5.76
T: 283s
$\varepsilon_{rel} = 55.12\%$
V: 7.05
T: 273s
$\varepsilon_{rel} = 53.10\%$
V: 7.10
T: 281s
$\varepsilon_{rel} = 149.51\%$
V: 3.10
T: 229s
$\varepsilon_{rel} = 164.73\%$
V: 2.99
T: 216s
$\varepsilon = 165.60\%$
V: 2.98
T: 215s

Figure 7: *Comparisons for a scene mainly lit by* ***direct*** *lighting, with a high frequency environment map. We compare in terms of relative efficiency between Sbert et al. [SHSK19] and Lu et al. [LPG13]. The number of learning samples for each technique is set to 128 per iteration and the total number of samples to compute the rendered images is 32768 (learning samples included). All timings were measured on a PC with an Nvidia® Titan V graphics card.* ***Top Part*** *Learning phase limited to direct lighting only [LPG13, SHSK19].* ***Bottom Part*** *Learning phase with global illumination, thus improving the efficiency.*

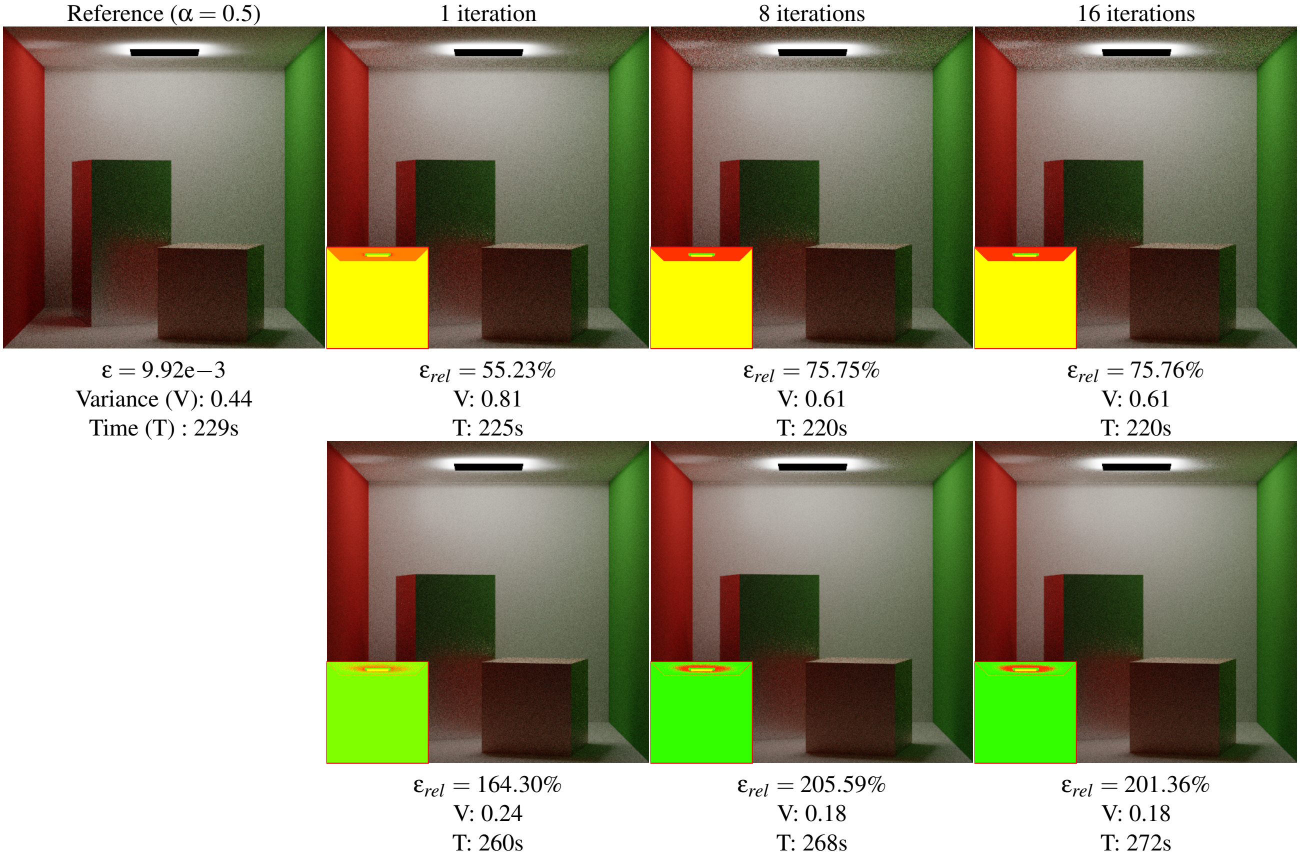


Figure 8: *Indirect Cornell box scene. The number of learning samples is set to 128 per iteration and the total number of samples to compute the rendered images is 32 768 (learning samples included). All timings were measured on a PC an Nvidia® Titan V graphics card.* ***Top Part*** *Learning phase limited to direct lighting only [LPG13, SHSK19].* ***Bottom Part*** *Learning phase with global illumination, thus improving the efficiency.*

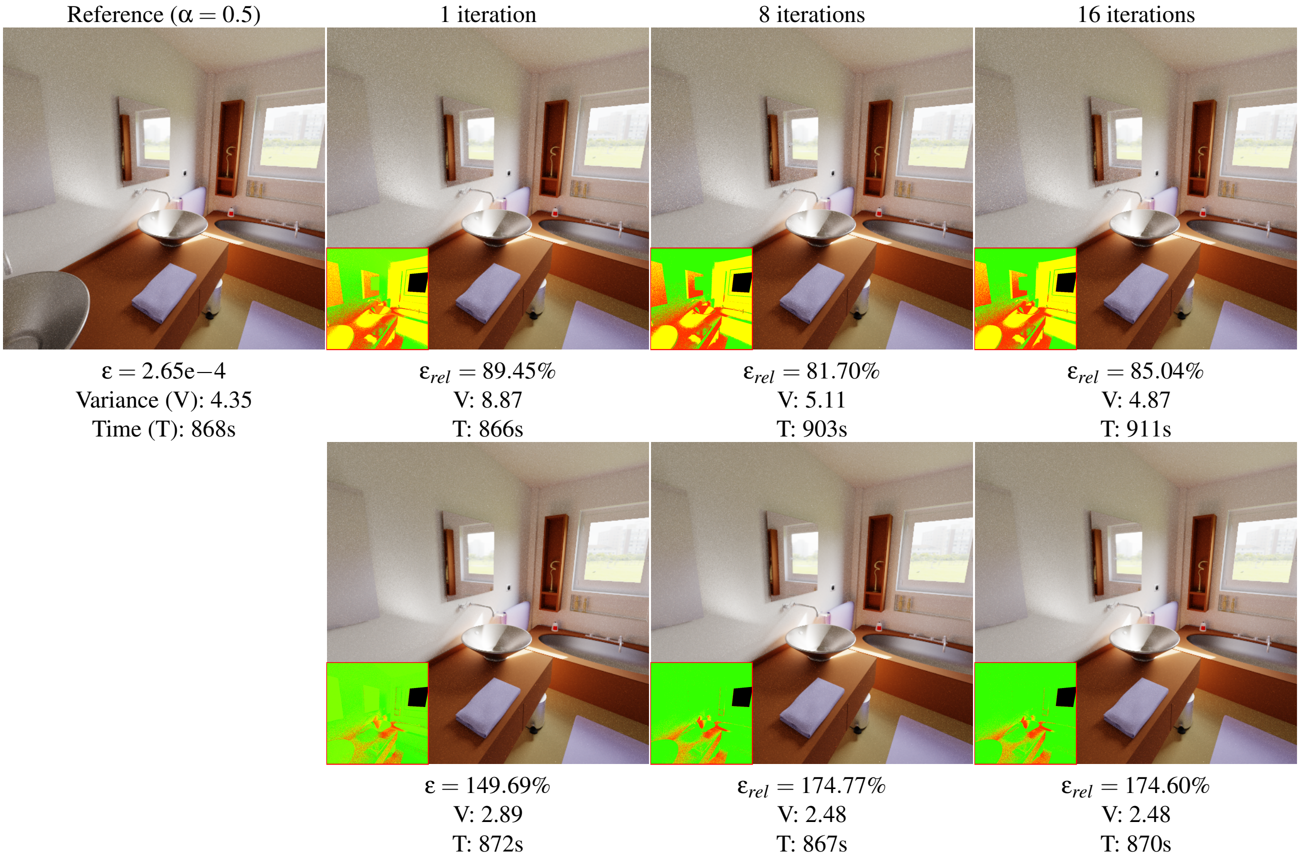


Figure 9: *Bathroom scene. The number of learning samples is set to 512 per iteration and the total number of samples to compute the rendered images is 32768 (learning samples included). All timings were measured on a PC with an Nvidia® Titan V graphics card.**Top Part** Learning phase limited to direct lighting only [LPG13, SHSK19]. **Bottom Part** Learning phase with global illumination.*

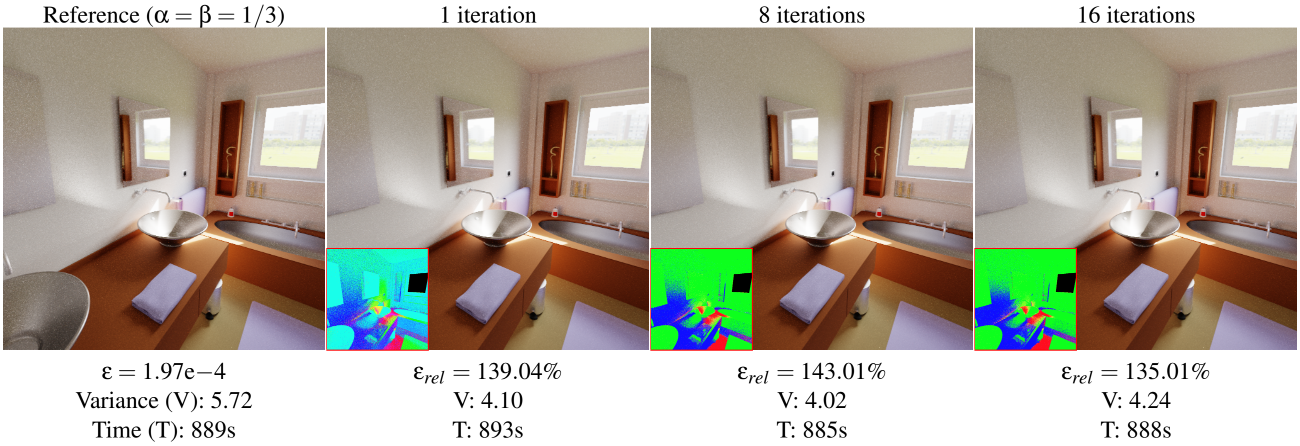


Figure 10: *Efficiency comparisons using three sampling strategies on the bathroom scene. The number of learning samples is set to 512 per iteration and the total number of samples to compute the rendered images is 32768 (learning samples included). All timings were measured using an Nvidia® Titan V graphics card.*