Multiple Importance Sampling for Emissive Effects

Ryusuke Villemin, Christophe Hery

Pixar Technical Memo 13-02

1 Introduction

With the advent of global illumination, emissive volumes interact with objects in the scene. These volumes need to be visible to ray-tracing, and volumetric effects rendering can no longer be a separate entity. If the energy emitted from volumes is fairly low, it can be treated as any normal emissive object; however if the emission becomes strong, we need to treat the volume as a light source with its own sampling strategy, in order to reduce variance.

In the past, such effects would be achieved efficiently by converting the volume data into multiple point light sources (as described in paragraph 7.1 of [Chr04]), or by considering only the envelope of the volume and simplifying it to a 2D surface sampling problem. With diffuse or highly glossy surfaces, those approaches can give good results, but for low gloss or mirror surfaces, the trick can become apparent. In many cases, these techniques usually require parameters tweaking and setup from the artists and are very time consuming.

In this report we present a method for creating and sampling volumetric light sources directly using the volumetric data, without any kind of imposters, obtaining high quality results with any mirror, glossy or diffuse objects, and integrating with any global illumination framework.

Our algorithm does not require any preprocessing or baking, so it is compatible with any type of progressive renderer.

Similar strategies have been employed before for infinite area lights by [PH04]: we are extending this method to volumetric lights and showing how to handle the extra dimension introduced by the volume compared to a standard 2D environment texture.

Our work can also be seen as complementary to other recent techniques focusing on scattering effects in volumes [JC98], or more recently [NNDJ12] and [KF12]. (See [CPCP+05] for an overview).

In part 2, we present the simplified volume rendering equation which is the basis our algorithm. In part 3, we discuss how light and brdf sampling can be done in a homogeneous volume, and in part 4 how to extend those results to heterogeneous volumes. Finally, we give a few implementation details in part 5 and conclude in part 6 showing our results.

2 Volumetric Rendering Equation

The full scattering equation [KVH84] [Gla95] [KPH+03] is:

$$(\omega \cdot \nabla)L(x,\omega) + \sigma_t(x)L(x,\omega) = \epsilon(x,\omega) + \sigma_s(x)\int_\Omega p(x,\omega,\omega')L(x,\omega')d\omega' \tag{1}$$

this can be rewritten, by first considering only what's happening in a fixed ω_0 ray direction:

$$L'(t) + \sigma_t(t)L(t) = \epsilon(t) + \sigma_s(t)\int_\Omega p(t,\omega')L(t,\omega')d\omega' \tag{2}$$

this is obviously still too difficult to solve analytically, so we simplify further by keeping only what interests us in the case of an isotropic volumetric light, that is absorption σ_a and emission ϵ (scattering σ_s is 0) [Max95]

$$L'(t) + \sigma_a(t)L(t) = \epsilon(t) \tag{3}$$

this is now a standard first order linear equation, and solving the lighting in our given ray direction:

$$L(t) = \int_0^t \epsilon(t')e^{-\int_{t'}^t \sigma_a(t'')dt''} dt' + L(x_0, \omega_0)e^{\int_0^t -\sigma_a(t')dt'} \tag{4}$$

Now we consider an isotropic homogeneous volume, emission ϵ and density σ_a are constant within it, so the equation becomes:

$$L(t) = \frac{\epsilon}{\sigma_a}(1 - e^{-\sigma_a t}) + L(x_0, \omega_0)e^{-\sigma_a t}, \forall \sigma_a > 0 \tag{5}$$

Another useful equation is the special case of attenuation, where $\epsilon(t)$ is null:

$$L'(t) + \sigma_a(t)L(t) = 0 \tag{6}$$

the solution is:

$$L(t) = L(x_0, \omega_0)e^{\int_0^t -\sigma_a(t')dt'} \tag{7}$$

and for a homogeneous volume:

$$L(t) = L(x_0, \omega_0)e^{-\sigma_a t} \tag{8}$$

We will use the following notation in the report:

$L(x, x')$ $L(x, \omega)$	radiance from x to x' $[Wm^{-2}sr^{-1}]$ radiance from x in the ω direction $[Wm^{-2}sr^{-1}]$
$f_s(x, x', x'')$ $f_s(x, \omega_o, \omega_i)$	surface brdf at x' from x to x'' $[sr^{-1}]$ surface brdf at x from direction ω_i to direction ω_o $[sr^{-1}]$
$\tau(x, x')$ $\tau(x, \omega, t)$	volumetric attenuation between x and x' volumetric attenuation between x and $x + \omega t$
$\delta_{vis}(x, x')$ $\delta_{vis}(x, \omega, t)$	geometric visibility between x and x' geometric visibility between x and $x + \omega t$
θ	angle between the surface normal N and ω_i (or segment $[x;\ x']$) $[r]$
ϵ	total volumetric emission $[Wm^{-3}sr^{-1}]$
ϵ_i	i-th voxel volumetric emission $[Wm^{-3}sr^{-1}]$
V	total volume $[m^3]$
V_i	i-th voxel volume $[m^3]$
σ_t	transmission coefficient $[m^{-1}]$
σ_s	scattering coefficient $[m^{-1}]$
σ_a	absorption coefficient $[m^{-1}]$

3 Homogeneous Emitting Volume

An isotropic homogeneous emitting volume has an emitting intensity of ϵV. Since everything is constant, for the light sampling we can use a uniform point sampling of the volume. For the brdf sampling, the intersection between the sampled direction and the volume is going to be a segment, so we have to compute the intensity by estimating the line integral over that segment.

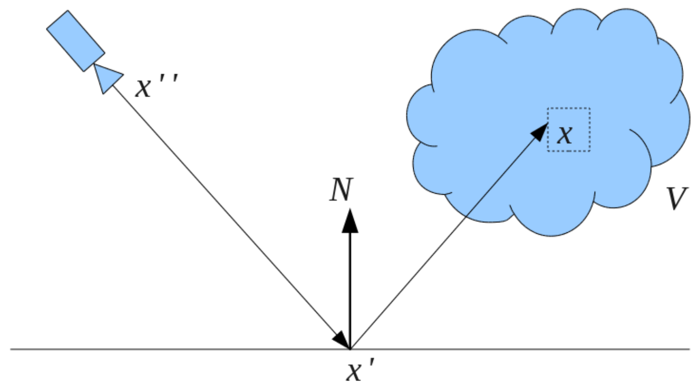

Figure 1: Light Sampling

3.1 Light Sampling

The light sampling is pretty straightforward. We need to sample a point inside the volume, and then compute the attenuation due to the density of the volume.

At a shading point on a surface, the contribution of an emissive volume is:

$$L(x', x'') = \int_V f_s(x, x', x'') cos(\theta) G(x, x') \tau(x, x') \epsilon dV(x) \tag{9}$$

with $G(x, x') = \frac{\delta_{vis}(x,x')}{\|x-x'\|^2}$

Since we are taking a sample randomly in the volume, $p(x) \propto \frac{1}{V}$. The resulting estimate is :

$$L(x', x'') \approx \frac{f_s(x, x', x'') cos(\theta) G(x, x') \tau(x, x') \epsilon}{p(x)} \tag{10}$$

The attenuation $\tau(x, x')$ is given by equation 8 for a homogeneous medium, in this case t is the distance traveled in the volume on segment $[x;\ x']$. For non-convexe volume, or volume with "holes" in it, there will be multiple segments. These will be handled by the algorithm for heterogeneous volumes in the next section.

3.2 BRDF Sampling

The brdf sampling is a sampling done on the surface side, so in the volume light we still need to estimate a line integral; i.e. from the input direction, we need to provide a point, emission value and pdf. The first step is to compute the intersection between the line and the volume to get the segment range. Once we have this segment, we sample a point on it.

$$L(x', \omega'_o) = \int_\Omega f_s(x', \omega'_i, \omega'_o) cos(\theta) L(x', \omega'_i) d\omega'_i \tag{11}$$

with

$$L(x', \omega'_i) = \int_S \delta(x', \omega_i, t) \tau(x', \omega_i, t) \epsilon dt \tag{12}$$

We saw in part 2 that we can analytically compute the emission for a given direction in a homogeneous volume. We will directly use this function as a pdf, the CDF is equation 5 and is easily invertible:

$$p(t \mid \omega_i) \propto \tau(x', \omega_i, t) \epsilon \tag{13}$$

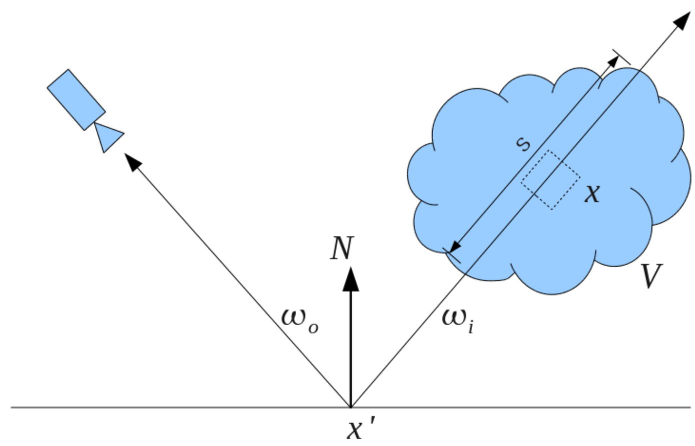

Figure 2: BRDF Sampling

(the CDF inversion gives us t for any random number χ, $t = \frac{-1}{\sigma_a} ln(1 - (1 - e^{-\sigma_a(t_1 - t_0)})\chi))$

Using $p(\omega_i')$ to denote the pdf resulting from the brdf sampling, the resulting estimate can be written as:

$$L(x', \omega') \approx \frac{f_s(x, \omega_i', \omega_o') cos(\theta) \delta_{vis}(x', \omega_i, t) \tau(x', \omega_i, t) \epsilon}{p(\omega_i') p(t \mid \omega_i')} \tag{14}$$

which reduces to:

$$L(x', \omega') \approx \frac{f_s(x, \omega_i', \omega_o') cos(\theta) \delta_{vis}(x', \omega_i, t)}{p(\omega_i')} \tag{15}$$

This gives us a nearly optimal sampling, the only part not accounted for is the visibility function $\delta_{vis}(x', \omega_i, t)$.

4 Heterogeneous Emitting Volume

Now let's consider that a heterogeneous volume is a collection of a finite number of homogeneous sub-volumes. For a grid volume, each voxel is going to be a cubic homogeneous isotropic emitting volume. By initially sampling the set of voxels, we can utilize our homogeneous sampling strategies for heterogeneous volumes.

4.1 Light Sampling

For the light sampling we sample the voxels based on the distribution of ϵ. Since we have a finite number of voxels, we can use the same method used in [PH04] for the environment map sampling, replacing the pixel colors by ϵ and adding one more dimension.

Once we have a voxel, we just need to uniformly sample a point inside it, which is exactly the process described in the previous section. For the following part, we will consider we have a grid η with total emission $\epsilon = \sum_{i \in \eta} \epsilon_i$:

$$L(x', \omega') = \sum_{i \in \eta} \int_{V_i} f_s(x, x', x'') cos(\theta) G(x, x') \tau_\eta(x, x') \epsilon_i dV_i(x) \tag{16}$$

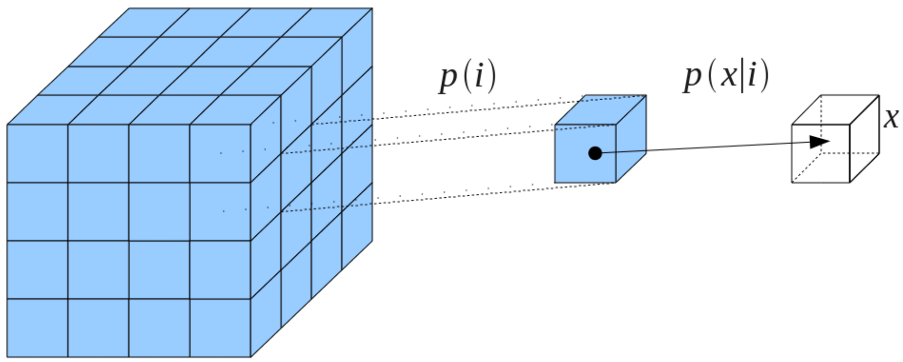

Figure 3: Light Grid

Also note that τ_η is not only the attenuation within the voxel i, but the attenuation due to all the voxels between the shading point and the sampled point. The exponential function representing the attenuation is now a piecewise exponential.

By taking a voxel based on a discrete PDF $p(i) \propto \epsilon_i$, and then a point inside this voxel using $p(x \mid i) \propto \frac{1}{V_i}$, the resulting final estimate is :

$$L(x', x'') \approx \frac{f_s(x, x', x'')cos(\theta)G(x, x')\tau_\eta(x, x')\epsilon_i}{p(i)p(x \mid i)} \tag{17}$$

This will reduce to:

$$L(x', x'') \approx \frac{f_s(x, x', x'')cos(\theta)G(x, x')\tau_\eta(x, x')\epsilon}{p(x \mid i)} \tag{18}$$

Again the attenuation is given by equation 8 for all the voxels between the shaded point and the sampled point.

4.2 BRDF Sampling

Similar to the homogeneous volume case, we can directly use the emission function as a pdf. The difference is that, like we did with the attenuation function, we have a piecewise exponential function instead of a single exponential.

$$L(x', \omega'_o) = \int_\Omega f_s(x', \omega'_i, \omega'_o)cos(\theta)L(x', \omega'_i)d\omega'_i \tag{19}$$

with

$$L(x', \omega'_i) = \int_S \delta_{vis}(x', \omega_i, t)\tau_\eta(x', \omega_i, t)\epsilon(t)dt = \sum_{i \in S} \int_{s_i} \delta_{vis}(x', \omega_i, t)\tau_\eta(x', \omega_i, t)\epsilon_i(t)dt \tag{20}$$

The resulting final estimate is :

$$L(x', \omega') \approx \frac{f_s(x, \omega'_i, \omega'_o)cos(\theta)\delta_{vis}(x', \omega_i, t)\tau_\eta(x', \omega_i, t)\epsilon_i}{p(\omega'_i)p(t \mid \omega'_i)} \tag{21}$$

which simplifies again to (equation 15):

$$L(x', \omega') \approx \frac{f_s(x, \omega'_i, \omega'_o)cos(\theta)\delta_{vis}(x', \omega_i, t)}{p(\omega'_i)} \tag{22}$$

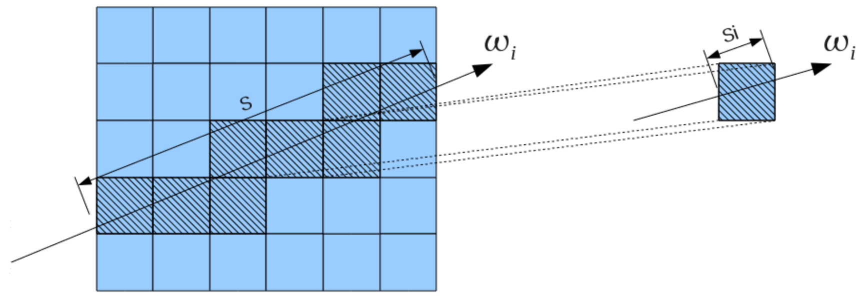

Figure 4: BRDF Grid

5 Implementation Details

Combining these two strategies with Multiple Importance Sampling is straightforward [VG95], but the implementation in a raytracer can be tricky. Usually the light sampling is done by light shaders, and the brdf sampling is done by the surface shader. This makes it difficult (or impossible) for the light shader to modify the pdfs produced by the surface.

Let's suppose we are using a squared heuristic for the MIS weight, the Light Sampling estimation is (a few parameters have been omitted and notation simplified for brevity):

$$L_{light} = \frac{p(x)^2}{(p(\omega)p(t \mid \omega))^2 + p(x)^2} \frac{f.L}{p(x)} \tag{23}$$

$p(\omega)$, f are returned by the brdf shader.

$p(x)$, L are returned by the light shader, along with an additional $p(t \mid \omega)$ that needs to be applied on $p(\omega)$.

A optimization we have used for this problem, is to return pre-divided values in our light shader. By making the light shader return $p(x)/p(t \mid \omega)$, $L/p(t \mid \omega)$, we can avoid the third extra parameter.

$$L_{light} = \frac{(p(x)/p(t \mid \omega))^2}{p(\omega)^2 + (p(x)/p(t \mid \omega))^2} \frac{f.(L/p(t \mid \omega))}{p(x)/p(t \mid \omega)} = \frac{p(x)^2}{(p(\omega)p(t \mid \omega))^2 + p(x)^2} \frac{f.L}{p(x)} \tag{24}$$

We can apply the same method to the BRDF sampling:

$$L_{brdf} = \frac{p(\omega)^2}{p(\omega)^2 + (p(x)/p(t \mid \omega))^2} \frac{f.(L/p(t \mid \omega))}{p(\omega)} = \frac{(p(\omega)p(t \mid \omega))^2}{(p(\omega)p(t \mid \omega))^2 + p(x)^2} \frac{f.L}{p(\omega)p(t \mid \omega)} \tag{25}$$

6 Conclusion

We were able to provide two sampling strategies for a volumetric light based on heterogeneous voxel data:

- light sampling based on the voxel emission distribution (equation 16)
- brdf sampling based on the product of transmitance and emission along a line (equation 21)

Since those two strategies are sampling the same domain, MIS is fully functional. The algorithm is robust and can handle a large variety of surfaces and volume (diffuse to mirror) without any special case handling the range of specular to diffuse surfaces or large to small arealights.

Workflows are greatly simplified and integration in any raytracer with sampled light capability is immediate. An additional advantage is that since this method only relies on important sampling and does not require any preprocessing or extra steps, it will integrate naturally in a progressive renderer.

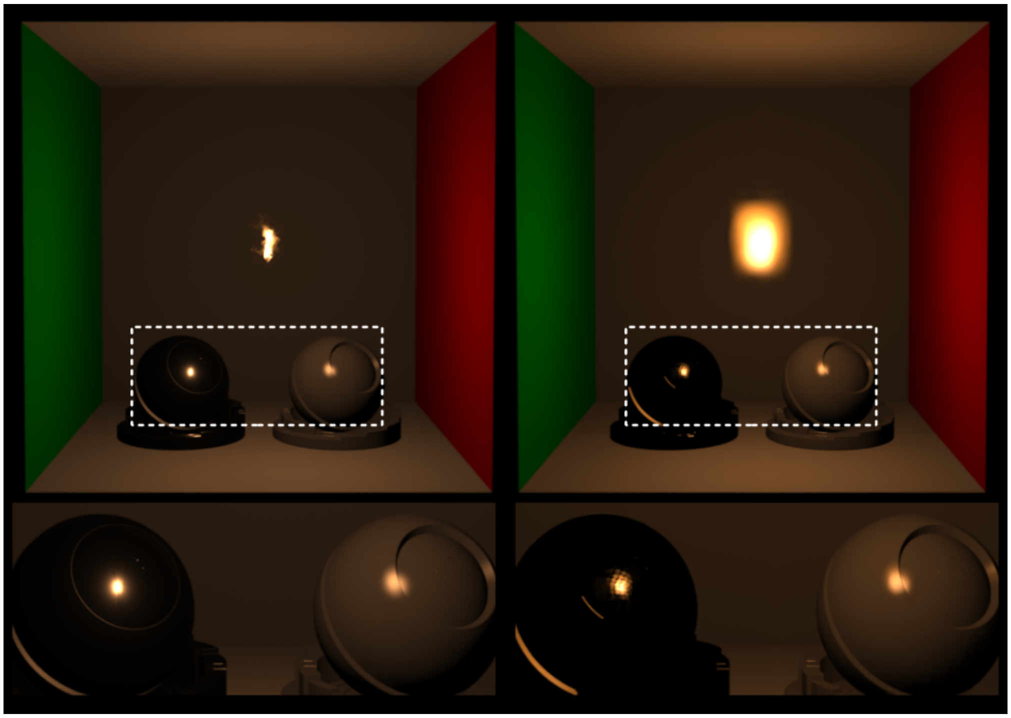

Figure 5: Our method (left) maintains high quality lighting even on specular surfaces where artefacts are visible with volumetric point-based method (right). Both images were rendered without occlusion and GI. A large amount of manual work was necessary to get a visual match with the point-based method, whereas our method gives us the right result automatically.

References

[Chr04] Per H. Christensen. Point-based approximate color bleeding. `http://graphics.pixar.com/library/PointBasedColorBleeding/paper.pdf`, 2004.

[CPCP+05] Eva Cerezo, Frederic Perez-Cazorla, Xavier Pueyo, Francisco Seron, and François Sillion. A survey on participating media rendering techniques. *the Visual Computer*, 2005.

[Gla95] A.S. Glassner. *Principles of digital image synthesis. 1.* The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling. Morgan Kaufmann, 1995.

[JC98] Henrik Wann Jensen and Per H. Christensen. Efficient simulation of light transport in scences with participating media using photon maps. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98, pages 311–320, New York, NY, USA, 1998. ACM.

[KF12] Christopher Kulla and Marcos Fajardo. Importance sampling techniques for path tracing in participating media. *Comp. Graph. Forum*, 31(4):1519–1528, June 2012.

[KPH+03] Joe Kniss, Simon Premoze, Charles Hansen, Peter Shirley, and Allen McPherson. A model for volume lighting and modeling. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):150–162, April 2003.

[KVH84] James T. Kajiya and Brian P Von Herzen. Ray tracing volume densities. In *Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '84, pages 165–174, New York, NY, USA, 1984. ACM.

[Max95] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, June 1995.

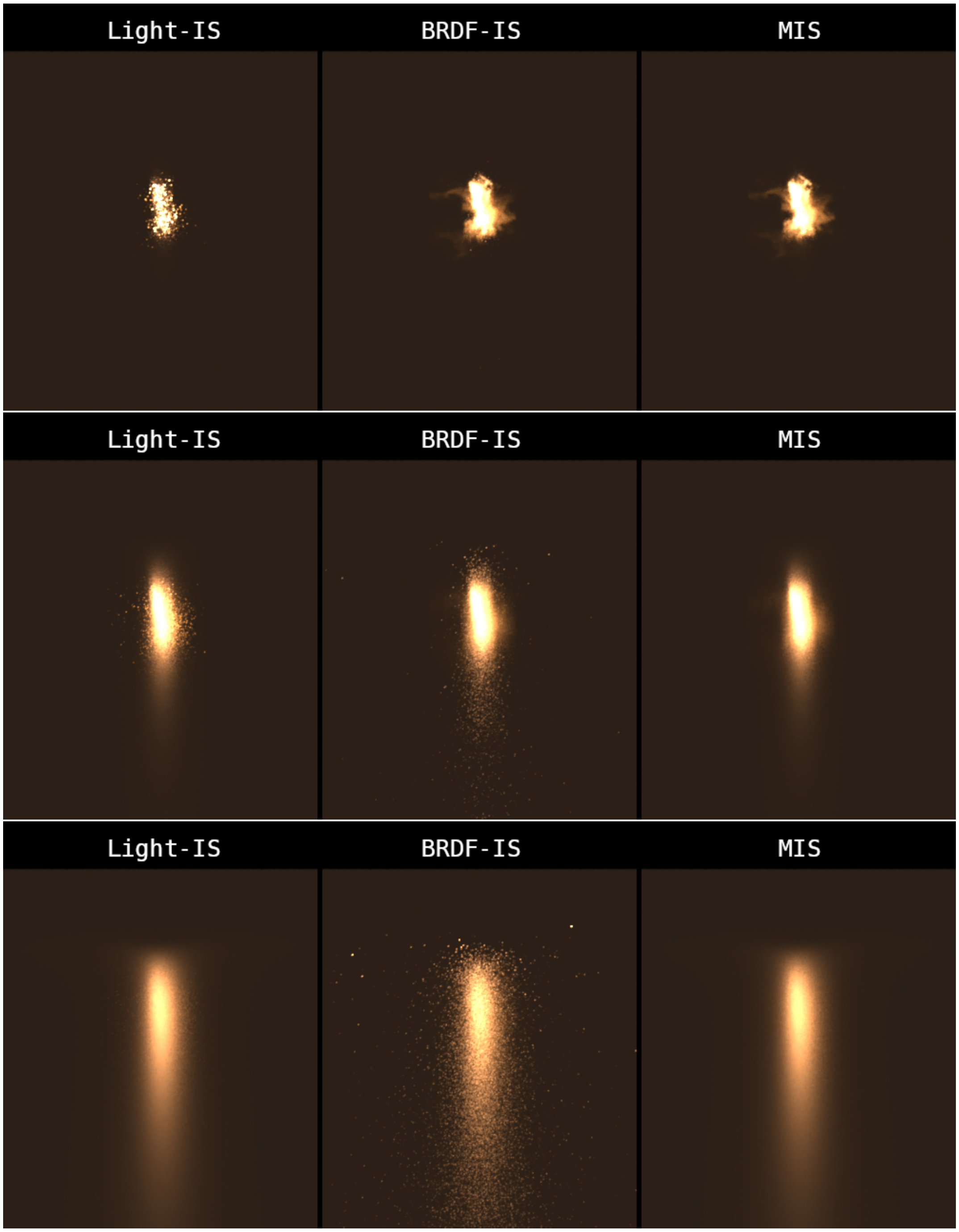

Figure 6: Flame Reflection on a Glossy Plane with different roughnesses

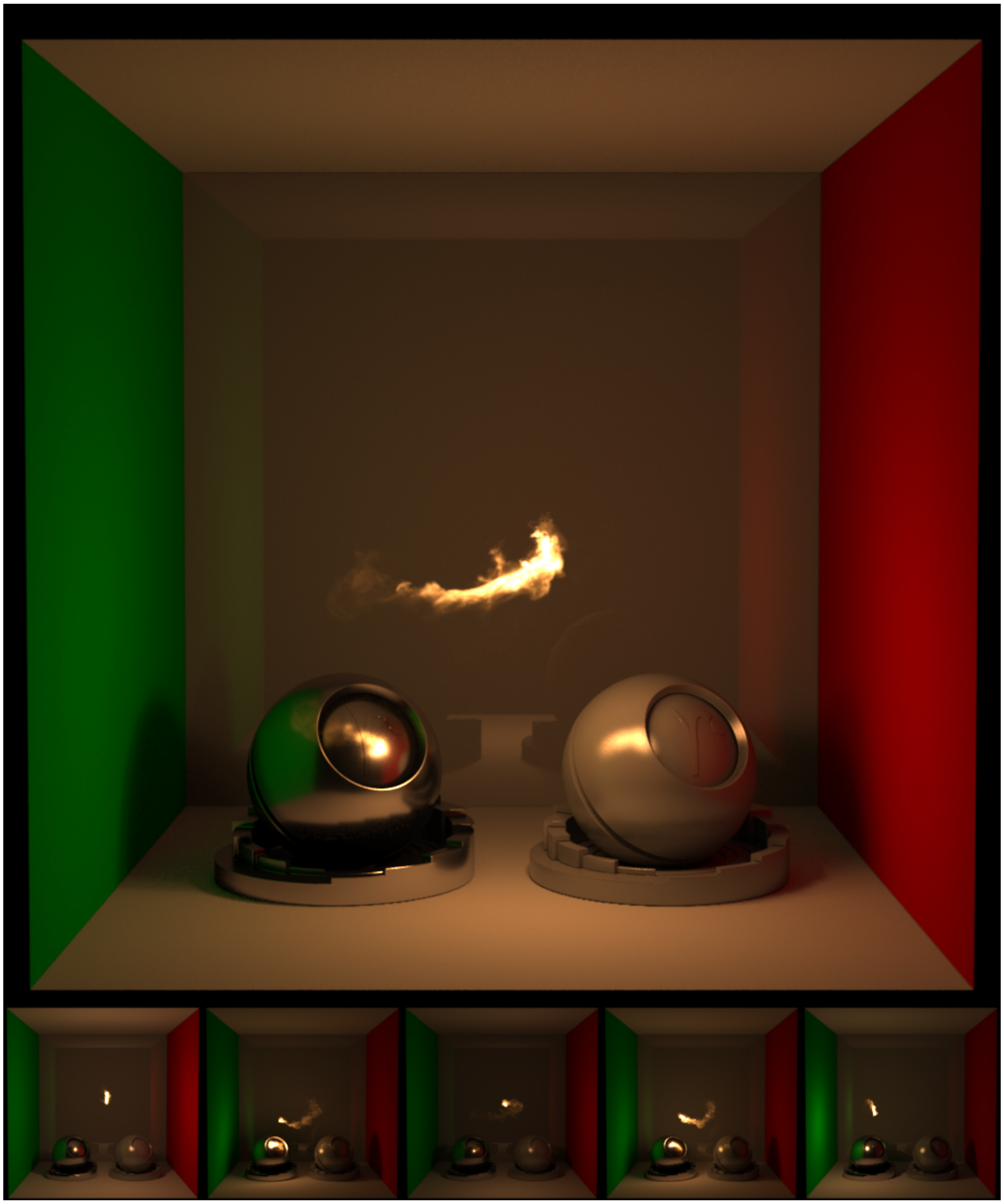

Figure 7: Cornell Box With Specular and Diffuse Objects. This scene with full global illumination was rendered at 1K resolution in 40 seconds on a 12 core machine with 64 light samples and 64 brdf samples per shading point.

[NNDJ12] Jan Novák, Derek Nowrouzezahrai, Carsten Dachsbacher, and Wojciech Jarosz. Progressive virtual beam lights. *Comp. Graph. Forum*, 31(4):1407–1413, June 2012.

[PH04] Matt Pharr and Greg Humphreys. Infinite area light source with importance sampling. `http://www.pbrt.org/plugins/infinitesample.pdf`, 2004.

[VG95] Eric Veach and Leonidas J. Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '95, pages 419–428, New York, NY, USA, 1995. ACM.