



# Frequency-Based Kernel Estimation for Progressive Photon Mapping

Laurent Belcour, Cyril Soler

## ► To cite this version:

Laurent Belcour, Cyril Soler. Frequency-Based Kernel Estimation for Progressive Photon Mapping. ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques in Asia, ACM SIGGRAPH, Dec 2011, Hong Kong, China. pp.Article No. 47, 10.1145/2073304.2073357 . inria-00633940

**HAL Id: inria-00633940**

**<https://inria.hal.science/inria-00633940>**

Submitted on 20 Oct 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

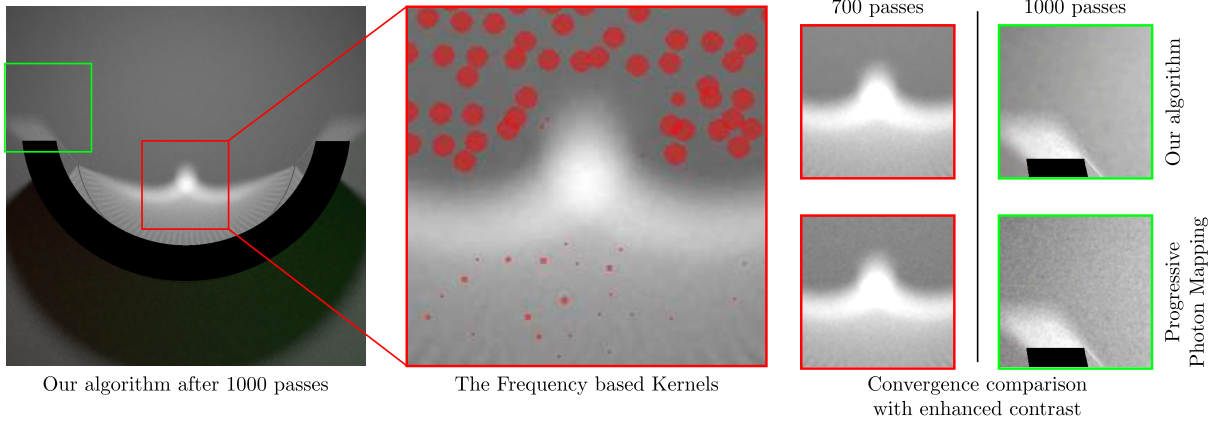
L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Frequency based Kernel Estimation for Progressive Photon Mapping

## Supplementary Material

Laurent Belcour\*  
Grenoble Universite

Cyril Soler†  
Inria



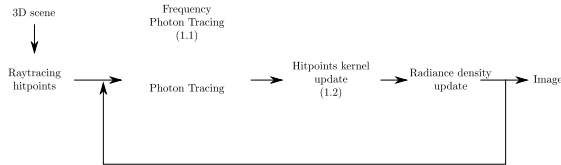
**Figure 1:** In this example, a caustic is created using a curved reflector. We estimate conservative density estimation kernels using frequency information. That permits to speed-up the convergence of the progressive photon mapping algorithm.

### Abstract

We present an extension to Hachisuka *et al.*'s Progressive Photon Mapping (or **PPM**) algorithm [Hachisuka et al. 2008] in which we estimate the radius of the density estimation kernels using frequency analysis of light transport [Durand et al. 2005]. We predict the local radiance frequency at the surface of objects, and use it to optimize the size of the density estimation kernels, in order to accelerate convergence. The key is to add frequency information to a small proportion of photons: *frequency photons*. In addition to contributing to the density estimation, they will provide frequency information.

### 1 Algorithm Overview

Our algorithm works as illustrated in Figure 2: First, we ray trace *hitpoints* in the scene from the camera. Second, we iteratively trace a proportion of *frequency photons* and *"classical" photons* in the scene (Section 1.1). These frequency photons are used to update the frequency estimate at *hitpoints* (Section 1.2).



**Figure 2:** Our algorithm overview. The blue part represent the original progressive photon mapping algorithm, the green part highlight our contributions.

Classical progressive photon mapping progressively reduces the size of the density estimation kernel to cancel the bias

of density estimation, at the expense of increasing variance. However, if the radiance is smooth enough, this size can be kept constant, and additional photons will only decrease variance of the estimate. The role of frequency photons is precisely to conservatively predict this phenomenon.

Consequently, we update the hitpoints' collecting size using the following heuristic: if the kernel size of classical PPM is greater than our prediction we decrease the radius just as the standard progressive photon mapping method does. Otherwise, we don't decrease the kernel size. We make this process iterative by always using a fixed proportion of frequency photons to update the desired size of all hitpoints.

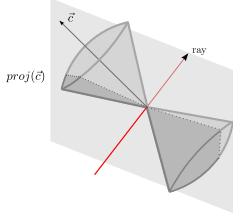
#### 1.1 Tracing Frequency Photons

Durand [2005] presents a Fourier analysis of the light transport in 4D. For efficiency reasons, we assume the radiance function to be isotropic in space and isotropic in angle, reducing the dimensionality of the spectrum of the radiance function to 2. We also approximate this spectrum by a Gaussian  $\hat{l}(\Omega_x, \Omega_\theta) \simeq e^{-(a\Omega_x^2 + 2c\Omega_x\Omega_\theta + b\Omega_\theta^2)}$  whose compact representation  $(a, b, c)$  is stored per photon and per hitpoint. The  $a$  coefficient denotes the spread in space of the radiance spectrum (e.g. accounting for occlusions or textures), the  $b$  coefficient denotes its spread in angle (e.g. due to reflectance properties), and the  $c$  coefficient allows for exact computation of the travel shear. At light sources, frequency photons are initialized with  $(a, b, c)$  values corresponding to the local radiance spectrum of the light source. When a photon is traced in the scene, we update these coefficients for each light transport atomic operation (travel in free space, partial occlusion, and reflection). Indeed, each of these atomic operation translates into a simple arithmetic operator on  $(a, b, c)$ . The derivation of the different operations are described in the appendix A.

**Occlusion** For occlusion detection, we precompute into a regular voxel grid the distance between the closest occluder and the voxels's centers. When testing for an occlusion

\*e-mail: laurent.belcour@inrialpes.fr

†e-mail: cyril.soler@imag.fr



**Figure 3:** When testing for an occlusion, we use a symmetrical bi-cone approximating the normal distribution in the voxel. We only have to test for the intersection between the bi-cone and the ray's orthogonal plane.

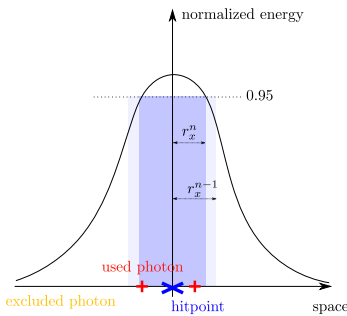
along a ray, we ray march the voxel grid and test whether the distance to the closest occluder is below a threshold holding for locality. Once an occluder is found, we convolve the photon's Gaussian with a Gaussian approximating the occluder's isotropic spectrum. For that aim, we assume that the visibility function is a Gaussian of width equal to the distance to the object.

**Distinguishing reflections** When looking to voxels close to a surface, the distance test is not able to distinguish between an occluding surface and a reflecting one. To avoid false positives in this scenario, we store an estimate of the normals distribution inside each voxel as a symmetrical bi-cone, whose direction is the mean normal direction and extent is the extent of the normals distribution. During occlusion detection, we test for the intersection between the bi-cone and the plane orthogonal to the ray as shown in figure 3.

**Frequency computation cost** Since we need to update only three variables ( $a, b, c$ ) for each transport operation, the cost overhead for frequency estimation is dominated by the occlusion ray marching's cost. For a  $64^3$  grid, we measured the time overhead to compute a frequency photon around 25%. Using no more than 25% of frequency photons leads to a total overhead of 6% in the photon tracing computation.

**Accumulating Frequency Information** At each reflection of a frequency photon we query for neighboring hitpoints and update each hitpoint's spectrum with the photon's spectrum if the latter falls in the hitpoint's kernel. We update the hitpoint's coefficients using its assigned ( $a, b, c$ ) coefficients. To ensure a conservative frequency kernel size, for each coefficient, we take the max value across collected frequency photons.

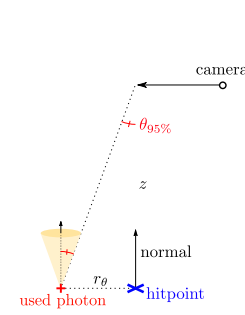
## 1.2 Updating the Hitpoint's Kernel Size



**Figure 4:** At pass  $n$ , we update the Gaussian using the photons inside the radius  $r_x^{n-1}$  of the hitpoint  $x$ . We then compute the new spatial radius using the updated Gaussian in the primal space.

To estimate the kernel size from the frequency estimation, we take as new radius the min between the radii estimated from the spatial and the angular frequencies. The spatial radius equals the abscissa corresponding to 95 percent of the total energy of the Gaussian in ordinate:  $r_x = \pi \sqrt{a} \beta_{95\%}$ ,

where  $\beta_{95\%} = -\log(0.95)$ . Figure 4 illustrate this equation.

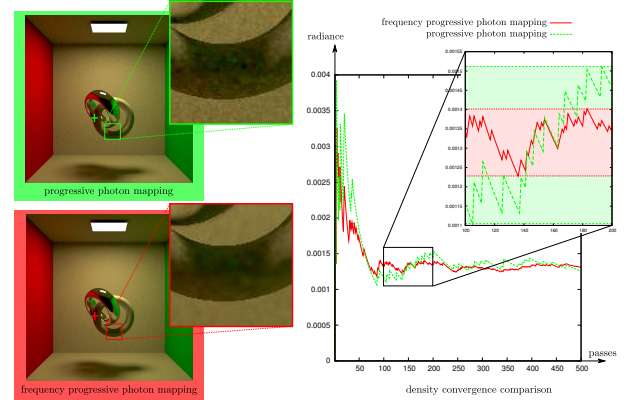


**Figure 5:** To estimate the angular radius, we look at the case where the reflection to the camera is along the normal direction. Using the alternate angles law, estimating the hitpoint's minimum radius  $r_\theta$  so that any contributive photon's angular region  $[0, \theta_{95\%}]$  contain the projected camera point boils down to finding the radius value using the same angle  $\theta_{95\%}$  from the projected camera. We chose  $\theta_{95\%}$  so that each photon's contribution remains above 95% of its max reflected intensity.

The angular radius is conservatively estimated as the radius such that the angle from the hitpoint's center to its border viewed by the camera projected on the hitpoint's normal equals the abscissa corresponding to 95 percent of the Gaussian's energy ordinate in the primal space along the angular direction:  $r_\theta = \tan(\pi \sqrt{b} \beta_{95\%})z$ , where  $z$  is the distance from the hitpoint center to the camera projected on the hitpoint's normal. We illustrate this equation in figure 5.

## 2 Results

As shown in the figure 6 our algorithm converges faster for low frequency parts such as diffuse non-occluded regions or indirectly lit diffuse regions. The frequency analysis enables us to detect specific lighting situations, such as a caustic's focus point, which corresponds to a frequency spectrum with no energy along the angular axis. Figure 1 is an example of a caustic produced by a curved reflector. As expected, the high frequency content in the spatial domain at the focus point causes smaller reconstruction kernels.



**Figure 6:** In this figure we compare against progressive photon mapping with our algorithm for the convergence of an indirectly lit part of the scene. In the closeup, we show that our algorithm produces a lower varying estimate at an earlier stage of its execution. The images were produced using 100.000 photons per pass and 25% of frequency photons to make timing comparable.

## 3 Ongoing works

Due to the presence of outliers mostly caused by occlusion testing, our frequency estimation is somewhat conservative in certain regions such as the diffuse part in the top of Figure 1. This leads to reduced convergence rate of the density

estimate compared to regions without outliers. One solution to this issue would be the use of a histogram for the parameters  $(a, b, c)$  based on the photon contribution at a given hitpoint. Such a method would exclude outliers, but would require additional memory space per hitpoint.

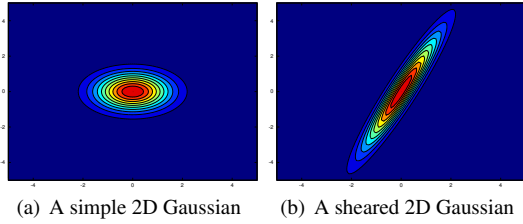
## 4 Acknowledgments

We would like to thank Pierre-Edouard Landes and Jean-Dominique Gascuel for their advises and comments during the redaction of this work.

## A Approximation of the Lightfield

We approximate the frequency lightfield function as a Gaussian. It carries the occupation in the Fourier space of the spectrum, with the knowledge of the spread of the Gaussian. This is a crucial information since our goal is to determine the minimum area around the central ray with no variation of the primal function.

Since most of the light sources we handle are planar and of a fixed size, their representation in the Fourier space is approximatively a cardinal sinus. We can map a Gaussian on the power spectrum of this function while preserving both the energy and the predominant shape: the lobe. The figure 7 shows the Gaussian application.



**Figure 7:** In this setting the 2D-Gaussian capture the envelop of the signal. We keep track of this envelop by applying all operations to the Gaussian like the shear on the left (b)

As we force the signal to be of a certain shape, this affect the different operations it will undergo. We must then give an analytical versions of each operator. For this, we model the Gaussian using its polynomial:

$$G(\Omega_x, \Omega_\theta) = Ee^{-(a\Omega_x^2 + 2c*\Omega_x\Omega_\theta + b\Omega_\theta^2)}$$

In the following lines, we present how we translated the different operations on the Gaussian model.

### A.1 Light sources:

Based on the type of light sources, we fit the spatial  $a$  and angular  $b$  coefficients to ensure that the Gaussian matched the local spectrum at the starting position.

### A.2 Transport:

A travel in space is a shear in the Fourier space. It means that the angular component of the function is shifted linearly with respect to the spatial axis and to a factor  $d$ : the distance of the shear.

With a Gaussian, we update our three parameters using the following equations:

$$\begin{aligned} a' &= a - 2cd + bd^2 \\ c' &= c - bd \\ b' &= b \end{aligned}$$

### A.3 Occlusion :

An occlusion in space is a convolution in Fourier space. Soler *et al.* [2009] showed that it can be well approximated using the following function as the occluder power spectrum in the Fourier space:

$$f(\Omega_x, \Omega_\theta) = \delta(\Omega_x) \frac{1}{\Omega_\theta}$$

We cannot use this model since the inverse function has an infinite bandwidth. Instead, we approximate the visibility function as a zero centered Gaussian in the primal which spread is the distance to the blocker. The associated Gaussian in Fourier has a spread proportional to the inverse of the primal one.

Using this approximation we can use the formula on centered Gaussians convolution to get the new coefficients:

$$\begin{aligned} a' &= \frac{a * a_{occ}}{a + a_{occ}} \\ c' &= c \\ b' &= b \end{aligned}$$

### A.4 Reflection :

A reflection on a surface is a complex operation. The most important part of it is the multiplication of the signal by the BRDF transform.

To compute the attenuation of the signal on the surface, we need to project our orthogonal plane to the surface plane (define by the normal of the surface) and then apply the deformation of the surface (simplified to a first order curvature in Durand *et al.* [2005] model).

We use the local incoming angle to perform the reparametrization to the tangent plane of the surface. We only have to scale our signal by  $\cos(\theta_i)$  in the spatial dimension.

$$\begin{aligned} a' &= a \cos^2(\theta_i) \\ c' &= c \cos(\theta_i) \\ b' &= b \end{aligned}$$

We assume here that the BRDF Fourier transform can be mapped onto a Gaussian. With this assumption done, the result of the operation is a multiplication by a 1D-Gaussian in angle. That means we have a single addition to perform.

$$\begin{aligned}
a' &= a \\
c' &= c \\
b' &= b + b_{brdf}
\end{aligned}$$

Where the transform of the BRDF is  $\mathcal{F}\{brdf\}(\Omega_\theta) = e^{-b_{brdf}\Omega_\theta^2}$ . We use the same notation as previously.

When the light leave the surface, we have to reparametrize our light plane to the orthogonal plane of the ray. This is done by scaling the power spectrum by  $\frac{1}{\cos(\theta_o)}$ .

$$\begin{aligned}
a' &= \frac{a}{\cos^2(\theta_i)} \\
c' &= \frac{c}{\cos(\theta_i)} \\
b' &= b
\end{aligned}$$

For this transformation to be effective, we need to estimate the parameter  $b_{brdf}$ . We used Ramamoorthi *et al.* [2004] approach and converted all materials to frequency equivalent Phong lobes.

## References

- DURAND, F., HOLZSCHUCH, N., SOLER, C., CHAN, E., AND SILLION, F. X. 2005. A frequency analysis of light transport. In *ACM SIGGRAPH 2005 Papers*, ACM, New York, NY, USA, SIGGRAPH '05, 1115–1126.
- HACHISUKA, T., OGAKI, S., AND JENSEN, H. W. 2008. Progressive photon mapping. In *ACM SIGGRAPH Asia 2008 papers*, ACM, New York, NY, USA, SIGGRAPH Asia '08, 130:1–130:8.
- RAMAMOORTHI, R., AND HANRAHAN, P. 2004. A signal-processing framework for reflection. *ACM Transactions on Graphics (TOG)* 23, 4.
- SOLER, C., SUBR, K., DURAND, F., HOLZSCHUCH, N., AND SILLION, F. X. 2009. Fourier depth of field. *ACM Transactions on Graphics (TOG)* 28, 2.