A hierarchical BRDF reconstruction using locally supported functions

N. Noé and B. Péroche

Ecole des Mines de Saint Etienne, Centre Simade, LISSE
158 Cours Fauriel 42023 Saint Etienne Cedex 2, France

Abstract. BRDFs are the spine of realistic rendering algorithms. But analytical models are sometimes inefficient since they cannot hold very particular material characteristics (retro-reflection, anisotropy, off-specularity, ...). Consequently we might want to directly use BRDF measured data, which leads to solve the problem of BRDF reconstruction. In this paper, we propose a solution which uses locally supported functions and which allows a hierarchical approach able to take irregular distributions of sample data into account. This method is not computationally expensive and gives the guarantee of a physically valid reconstruction. We also discuss about the quality of the reconstruction, by introducing some new error parameters.

1 Introduction

The bidirectional reflectance distribution function (BRDF) of a material describes its reflection behaviour under illumination. BRDFs are the spine of realistic rendering algorithms, because they determine the appearance of objects in a scene, through global illumination computation for example.

BRDFs, when wavelength is not taken into account, are functions from Ω_+^2 into $\mathbb{R}$, Ω_+ being the upper hemisphere and Ω the whole sphere. Hence, BRDFs have four independent real parameters (angles). Thus, it is difficult to represent them efficiently.

Analytical models of BRDF can't handle correctly every material behaviour (retro-reflection, off-specularity, anisotropy, ...). So we chose to work with physically measured or simulated BRDFs. As a consequence, we have to reconstruct a complete function over Ω_+^2 from a limited set of data. Our solution uses locally supported functions to be less computationally expensive and allows successive approximations of the BRDF to be able to take irregular distributions of sampled data into account.

The remainder of the paper is organized as follows. Section 2 presents the notations used and previous work on the representation of BRDFs. Our representation model is explained in section **??** while a hierarchical approach is described in section **??**. The problem of error measurement is tackled in section **??**. Implementation information and some results are discussed in sections **??** and **??**, and a conclusion and some further developments are suggested in section **??**.

2 Notations and previous work

2.1 Notations

To begin with, we shall expose some classical definitions. *Radiance* is the energical flux emitted by a surface, at point p, in a given direction $\vec{\omega}$, per unit of surface and unit of projected solid angle (in $W\,m^{-2}\,sr^{-1}$). If θ stands for the angle between $\vec{\omega}$ and the normal to surface dA, the exact definition of *Radiance* is:

$$L(p,\vec{\omega}) = \frac{d^2\varphi_r(\vec{\omega})}{dA\,\cos\theta\,d\omega}$$

Irradiance is the energical flux received by a surface, per unit of surface and for a given direction (in $W\,m^{-2}$):

$$E(p,\vec{\omega}) = \frac{d\varphi_i(\vec{\omega})}{dA}$$

BRDF is the ratio of reflected radiance along direction $\vec{\omega_r}$ and incident irradiance of the surface along direction $\vec{\omega_i}$:

$$F(p,\vec{\omega_i}\rightarrow\vec{\omega_r}) = \frac{dL_r(p,\vec{\omega_r})}{dE(p,\vec{\omega_i})} = \frac{dL_r(p,\vec{\omega_r})}{L_i(p,\vec{\omega_i})\,\cos\theta_i\,d\omega_i}$$

Let us remind that a BRDF must satisfy the crireria of physical plausibility, i.e. positiveness, reciprocity and energy conservation [**?**].

2.2 Previous approaches to BRDF representation

The main efforts in BRDF representation have focused on either analytic formulas or generic techniques for storing four-dimensional functions. We shall also present a recent work based on a neural network model. Some work on tabulated BRDF must also be mentioned [**?**].

Analytic models: Among these models, we find physically-based models [**?**, **?**, **?**]. But as they start with specific assumptions about the microgeometry of, the materials, they can only predict the reflectance of surfaces that closely match those assumptions. In particular, analytic models can't handle correctly every material behaviour (retro-reflection, off-specularity, anisotropy, ...). Phenomenological models have been introduced to widen the range of representable BRDFs. The most popular of them is Phong's model [**?**]. The models described in [**?**, **?**, **?**] can represent a wide range of phenomena and are physically plausible.

Decomposition into basis functions: A widely used technique for representing continuous functions is to describe this function as a linear combination of some set of basis functions. For BRDF representation, spherical harmonics are often used. We can use BRDF properties to ignore many of the coefficients [**?**]. But we still need a lot of coefficients because of the global control over the representation. And because of the global support (that is to say they must be evaluated everywhere), we'll need a lot of computations to get a good result. Furthermore, there are big problems of numerical instability with these representations, and it's very hard to handle specular peaks correctly.

It would be too long to detail wavelet theory in this paper. One will refer to [**?**] for an introduction to wavelets, and to [**?**, **?**] for their use in BRDF representation. Our method is somewhat related to wavelets, but is more BRDF specific.

BRDF are defined on the cartesian product of two hemispheres. Hence, we can easily project incident and reflected directions onto a plane. As a consequence, a BRDF can also be defined on the cartesian product of two planes. There are a lot of possible representations available. We can think about Zernike polynomials [**?**], which form an orthonormal basis of functions on the unit disk, or even surfaces such as *NURBS*.

Neural networks: In [**?**], Gargan and Neelamkavil proposed a non-linear representation of BRDFs based on a neural network model. They used standard backpropagation networks with two or three weight layers, linear basis functions and sigmoid activation functions. Only the network weights together with a scaling factor are needed to reconstruct the BRDF.

3 A new approach

3.1 Intuitive idea

For a given pair of incident and reflected directions, we can bind the BRDF value to the bounce of a particle on the surface and to the loss of energy related. Consequently, each measure can be associated to a particle trajectory. Furthermore, we can think of a particle as an energy distribution similar to a Dirac function. With lots of such particles, the reflected light will become a continuous function. Let us notice that this idea was inspired by the work in [**?**].

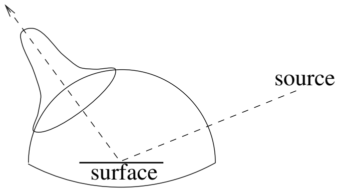

Fig. 1. a "particle" representation

The main benefit of this approach is that we may imagine an energy distribution on Ω^2. For a set of measures (X_i, Y_i, F_i), we may write:

$$F\left(p, X \to Y\right) = C \sum_{i=1}^{n} F_i\, K\left(d\left(\left(X, Y\right), \left(X_i, Y_i\right)\right)\right)$$

C is a normalization constant, K the energy distribution associated to the particles and d the distance between two pairs of directions. This model implies many drawbacks: which distribution shall we use ? How to compute the normalization constant ? Is there a natural distance in Ω^2 ? The chosen distribution must have a local support, to ensure a fast evaluation, which brings us to the issue of the size of this local support. We shall now give a formal definition of such a distribution.

3.2 A formal approach

Some definitions. Let $C(\Omega)$ be the set of continuous functions from Ω into $\mathbb{R}$ and let $\mathcal{L}^1([-1;1])$ be the set of integrable functions from $[-1;1]$ into $\mathbb{R}$. Let $F \in C(\Omega)$ and $f \in \mathcal{L}^1([-1;1])$. We define $F \otimes f \in C(\Omega)$, the convolution of F and f as:

$$\left(F \otimes f\right)(X) = \int_{\Omega} F(Y)\, f(X \cdot Y)\, d\omega(Y)$$

where $X \in \Omega$, $X \cdot Y$ is the dot product of vectors X and Y and $d\omega$ is the canonical measure on the sphere. For $\rho \in]-1;1[$, let $\{K(\rho,.)\}$ be a subfamily of $\mathcal{L}^1([-1;1])$. Then $\{I(\rho,.)\}$ is defined by the convolution $I(\rho, F) = K(\rho,.) \otimes F, F \in C(\Omega)$. $\{I(\rho,.)\}$ is called the *singular integral* corresponding to the integral kernel $\{K(\rho,.)\}$. We've got the following statement :

$$\int_{-1}^{1} K(\rho,t)\, dt = \tfrac{1}{2\pi} \text{ and } \lim_{\rho\to 1^-} \int_{-1}^{1} K(\rho,t)\, t\, dt = \tfrac{1}{2\pi} \text{ (*)}$$
$$\Rightarrow \int_{\Omega} I(\rho,F)\, d\omega = \int_{\Omega} F\, d\omega \text{ and } \lim_{\rho\to 1^-} I(\rho,F) = F \text{ (uniform convergence)}$$

An example of kernel. We'll use our own set of polynomials as kernel. We define:

$$B(\rho,t) = \begin{cases} 0 & \text{if t } \in [-1;\rho] \\ \frac{k+1}{2\pi(1-\rho)} \left(\frac{t-\rho}{1-\rho}\right)^k & \text{if t } \in [\rho;1] \end{cases}$$

$B(\rho, .)$ fulfils the (*) property Function $Y \to B(\rho, X \cdot Y)$ is then called a *spherical cap* at X. figure **??** is a representation of these functions, with $\rho = 0.5$ (the curve support is thus $[\frac{-\pi}{3}; \frac{\pi}{3}]$).

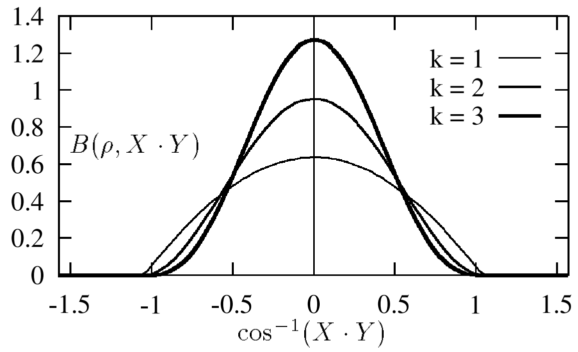

Fig. 2. kernels with different degrees

For $k = 2$, the shape is almost a Gaussian curve, but we've got more freedom with other values for k.

Ω^2 extension. We can easily extend the definition of section 3.2.2 to functions $F : \Omega^2 \to \mathbb{R}$. Let's define projections: $X \in \Omega, F_X(Y) = F(X,Y)$ and $Y \in \Omega, F_Y(X) = F(X,Y)$. Then, thanks to the uniform convergence of $I(\rho, F)$ toward F:

$$\lim_{\rho \to 1^-} \int_\Omega F_X(y)\, K(\rho, Y \cdot y)\, d\omega(y) = F_X(Y)$$

$$\lim_{\rho \to 1^-} \int_\Omega F_y(x)\, K(\rho, X \cdot x)\, d\omega(x) = \lim_{\rho \to 1^-} \int_\Omega F(x,y)\, K(\rho, X \cdot x)\, d\omega(x) = F_X(y) = F_y(X)$$

$$\lim_{\rho \to 1^-} \iint_{\Omega^2} F(x,y)\, K(\rho, X \cdot x)\, K(\rho, Y \cdot y)\, d\omega(x)\, d\omega(y) = F(X,Y) = F_X(Y)$$

To simplify notations, we introduce $K^{(2)}(\rho, u, v) = K(\rho, u) K(\rho, v)$.

A kernel on Ω^2. We've got a naturally defined kernel on Ω^2, which is the product of two kernels on Ω. This introduces a total decorrelation between the two parts of the cartesian product. We may avoid it by introducing a new kernel $K(\rho, u, v)$ on Ω^2. It must fulfill the following conditions:

$$\iint_{[-1;1]^2} K(\rho, u, v)\, du\, dv = \frac{1}{4\pi^2} \text{ and } \lim_{\rho \to 1^-} \iint_{[-1;1]^2} K(\rho, u, v)\, u\, v\, du\, dv = \frac{1}{4\pi^2}$$

We suggest to introduce $K(\rho, u, v) = B\left(\rho, \frac{u+v}{2}\right)$ as a non-normalized kernel. Hence, the following kernel is normalized and fulfills the (*) property for Ω^2:

$$K(\rho, u, v) = \frac{(k+1)(k+2)}{16\pi^2(1-\rho)^2}\, B\left(\rho, \frac{u+v}{2}\right)$$

Discretization. The former formulas are continuous and we are looking forward to applying them to discrete measures. As a matter of fact, we've just got sets of measures

$\{X_i, Y_i, F_i\}$, that do not present any particular decorrelated sets in X and Y. Consequently, it becomes very difficult to determine a good $\Delta\omega_{(2)i} = \Delta\omega(X)_i \Delta\omega(Y)_i$ that might be associated to every measure. Therefore, we'll use a constant solid angle:

$$F(X,Y) = \sum_{i=1}^{n} F_i \, K^{(2)}(\rho, X \cdot X_i, Y \cdot Y_i)\, \Delta\omega_{(2)}$$
$$f(X,Y) = \sum_{i=1}^{n} F_i \, K^{(2)}(\rho, X \cdot X_i, Y \cdot Y_i)$$

We'll choose the value of $\Delta\omega_{(2)}$ so that the RMS between the recomposed function and the original one is minimal. Hence:

$$E(\Delta\omega_{(2)}) = \sum_{i=1}^{n} (F(X_i, Y_i) - F_i)^2 \text{ has to be minimal}$$
$$\Rightarrow \frac{dE(\Delta\omega_{(2)})}{\Delta\omega_{(2)}} = 2\,\Delta\omega_{(2)} \sum_{i=1}^{n} f(X_i, Y_i)^2 - 2\sum_{i=1}^{n} f(X_i, Y_i)\, F_i = 0$$
$$\Rightarrow \Delta\omega_{(2)} = \frac{\sum_{i=1}^{n} f(X_i, Y_i)\, F_i}{\sum_{i=1}^{n} f(X_i, Y_i)^2}$$

This kind of approximation eliminates the kernel normalization problem. It is also very important to notice that this expression is just an approximation of the original function, and in no way an interpolation. That is to say we won't be able to compute the BRDF where we don't have any information.

4 A hierarchical approach

4.1 Main idea

Tuning the parameters. The main problem of the discretization method of section **??** is that it is difficult, and sometimes even impossible, to have a good parameter ρ. Actually, we haven't any particular information about the distribution of the samples. However, most of the time, this distribution is not regular at all on Ω^2. As a matter of fact, it is likely that during the measurement process, one will oversample the BRDF where it has a lot of variations. For instance, there will be oversampling around the mirror direction (where most of the time, the specular peak is). By choosing a too small ρ, we can create holes where there are few samples. In the same way, by choosing a bigger ρ, we'll get an extreme smoothing in the oversampling area. One solution is to choose a different ρ for every sample. But it raises many problems: we would have to perform a global optimization to compute the best ρ for each sample, a wide range of ρ could govern the aspect of the approximation and this method doesn't even work with some sampling distributions.

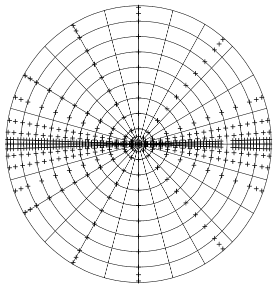

Fig. 3. sampling distribution of the reflected direction

For instance, figure **??** show real sampling directions (for a given incidence). It's obviously impossible to get a good ρ. The only solution might be to introduce two parameters, one for the θ direction, the other for the φ direction. But there wouldn't be any more mathematical explanation. Moreover, this is a particular case, and we are looking for a general method.

A hierarchical approach. In order to solve this problem, we'll use a wavelet similar approach. We'll represent the function at different levels of details, with more and more refinement. We'll use two grids on Ω^2. Actually, $I\left(\rho, I\left(\rho, F\right)\right)$ (named $I^{(2)}\left(\rho, F\right)$) also converges toward F. We'll then discretize on the samples grid and on a regular grid too.

Fitting to a BRDF analytical model. First of all, it might be a good idea not to represent the BRDF directly. One just needs to represent the error between the BRDF and a given analytical model. The generalized cosine lobe model [**?**] is very simple to use, since it can handle as many peaks as wanted. But, when the BRDF is mainly diffuse, we can simply use a lambertian model. The computation of the diffuse constant to use is not as simple as it looks. Instead of choosing the average sample value (so as to minimize the error), it's more efficient to choose the natural diffuse component of the samples, the same way it appears in lots of analytical models ($F(X, Y) = F_d + F_s(X, Y)$). To compute this value we sort the data by increasing BRDF values and search for largest interval with minimal RMS [1].

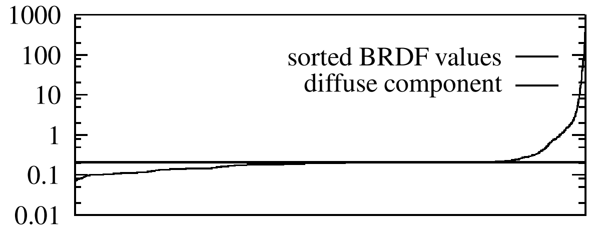

Fig. 4. the natural diffuse part of an experimental BRDF (ceramic tiles)

This choice is very relevant when we want to use very few caps to reconstruct the BRDF, since the error is neglectable on a lot of samples. It's then a better choice than the minimal value of the BRDF.

We could also perform a change-of-variables transformation on the samples directions (as in [**?**]), and then use our reconstruction method.

4.2 Formalization

Our goal is to represent a function F (the BRDF itself, or the error between it and an analytical model). We've got the experimental data: $\mathcal{E}_0$, the set of samples of Ω^2_+ we've measured. We also need $\mathcal{R}_0 \supset \mathcal{R}_1 \supset \mathcal{R}_2 \ldots$ finer and finer discrete parts of Ω^2_+. We associate to these parts caps' parameters $\rho_0 < \rho_1 < \rho_2 \ldots$, that are closer and closer to 1 (i.e. smaller and smaller caps). We apply the discretization two times: the first one on $\mathcal{E}_0$, the second one on $\mathcal{R}_0$:

$$I\left(\rho_0, F\right)(X, Y) = C_0\left(X, Y\right) = \sum_{(x,y)\in\mathcal{E}_0} F(x, y)\, K\left(\rho_0, X \cdot x, Y \cdot y\right)\ d\omega_{\mathcal{E}_0}$$
$$I^{(2)}\left(\rho_0, F\right)(X, Y) = F_0\left(X, Y\right) = \sum_{(x,y)\in\mathcal{R}_0} C_0(x, y)\, K\left(\rho_0, X \cdot x, Y \cdot y\right)\ d\omega_{\mathcal{R}_0}$$

Then we have an approximation of F, called F_0. But this approximation is very smooth and still far from F, because ρ_0 is small enough to avoid holes in $\mathcal{E}_0$ distri-

[1] Experimental data come from Laboratoire Central des Ponts et Chaussées goniophotometer [**?**].

bution and because it's the result of two smoothings. Hence we have to improve this approximation. That's why we are now going to represent $F - F_0$, the error left after the first level approximation. However, for a good subset of $\mathcal{E}_0$, the error will be likely insignificant. Consequently, we can choose an error threshold, and extract from $\mathcal{E}_0$ a new subset $\mathcal{E}_1$, where the error is greater than the chosen threshold.

$$\mathcal{E}_1 = \{(X, Y) \in \mathcal{E}_0 \; / \; \mid F(X, Y) - F_0(X, Y) \mid > error\}$$

We now have to represent $F - F_0$ and we write down:

$$I\left(\rho_1, F - F_0\right)(X, Y) = C_1\left(X, Y\right) = \sum_{(x,y)\in\mathcal{E}_1} F(x, y)\, K\left(\rho_1, X\cdot x, Y\cdot y\right)\, d\omega_{\mathcal{E}_1}$$
$$I^{(2)}\left(\rho_1, F - F_0\right)(X, Y) = F_1\left(X, Y\right) = \sum_{(x,y)\in\mathcal{R}_1} C_1(x, y)\, K\left(\rho_1, X\cdot x, Y\cdot y\right)\, d\omega_{\mathcal{R}_1}$$

And we iterate the process, until all the data get an error lower than the chosen threshold. But it's not worth going too far, if the number of samples is too small. As a matter of fact, the caps related to ρ_i will only span one sample, and won't smooth anymore, but will introduce slots.

Advantages. This method presents lots of advantages. First there is no need for *a priori* ρ values. Second we've got control over a progressive refinement of the representation. Third all kernels have a local support, which means a fast evaluation. Whereas the cardinal of $\mathcal{R}_i$ sets increases very quickly, the number of non-zero coefficients is very small. Actually, we can rewrite our formalization:

$$\mathcal{E}_i = \{(X, Y) \in \mathcal{E}_{i-1} \; / \; \mid F(X, Y) - F_0(X, Y) - \ldots - F_{i-1}(X, Y) \mid > error\}$$
$$F_i\left(X, Y\right) = \textstyle\sum_{(x,y)\in\mathcal{R}'_i} C_i(x, y)\, K\left(\rho_1, X\cdot x, Y\cdot y\right)\, d\omega_{\mathcal{R}'_i}$$
$$C_i\left(X, Y\right) = \sum_{(x,y)\in\mathcal{E}_i} \left(F(x, y) - F_0(x, y) - \ldots - F_{i-1}(x, y)\right)\, K\left(\rho_1, X\cdot x, Y\cdot y\right)\, d\omega_{\mathcal{E}_i}$$

where $\mathcal{R}'_i$ is the part of $\mathcal{R}_i$ where $C_i(X, Y)$ is non-zero, that is to say the intersection of $\mathcal{R}_i$ and $\mathcal{E}_i$ increased by the caps size. Then: $F = F_0 + F_1 + F_2 + \ldots$

4.3 Fulfilling BRDF constraints

In this paragraph, we'll try to improve our kernel, so that the reconstructed function fulfills essential BRDF properties: reciprocity, energy conservation and if needed isotropy.

Reciprocity. Since F is a BRDF, we know that $F(X, Y) = F(Y, X)$.

$$I\left(\rho, F\right)(X, Y) = \textstyle\iint_{\Omega^2_+} F(x, y)\, K^{(2)}\left(\rho, x\cdot X, y\cdot Y\right)\, d\omega(x)\, d\omega(y)$$
$$= \textstyle\iint_{\Omega^2_+} F(x, y)\, \frac{1}{2}\left(K^{(2)}\left(\rho, x\cdot X, y\cdot Y\right) + K^{(2)}\left(\rho, y\cdot X, x\cdot Y\right)\right)\, d\omega(x)\, d\omega(y)$$

We just have to introduce a new kernel:

$$L^{(2)}\left(\rho, (X, Y), (x, y)\right) = \frac{1}{2}\left(K^{(2)}\left(\rho, x\cdot X, y\cdot Y\right) + K^{(2)}\left(\rho, y\cdot X, x\cdot Y\right)\right)$$

This kernel is obviously symetric in relation to X and Y. We can now use it to fulfill the reciprocity law, in addition to the fact we can double the number of measures thanks to this property.

Energy conservation. It's not possible to get a simple analytic expression of the energy reflected by the reconstructed BRDF. The only thing that may be said is that if we got enough samples, the approximation will be close enough to the real BRDF, which fulfills energy conservation.

Isotropy. A material is anisotropic when its reflection properties change when it is rotated around its normal. A lot a materials are isotropic and only a few analytic models [**?**, **?**, **?**, **?**] can handle anisotropy. Anisotropy may be natural (wood) or artificial (brushed aluminum for instance). A BRDF is isotropic if and only if:

$$\forall (X, Y) \in \Omega_+^2, \forall \varphi \in [0; 2\pi]\; F(X, Y) = F(X + \varphi, Y + \varphi)$$

where $X + \varphi$ is the rotation of X around the surface normal (by angle φ). We can show that singular integrals preserve isotropy:

$$\begin{aligned} I^{(2)}(\rho, F(X{+}\varphi, Y{+}\varphi)) &= \iint_{\Omega_+^2} F(x, y)\, K^{(2)}(\rho, x \cdot (X{+}\varphi), y \cdot (Y{+}\varphi))\; d\omega^2(x, y) \\ &= \iint_{\Omega_+^2} F(x, y)\, K^{(2)}(\rho, (x - \varphi) \cdot X, (y - \varphi) \cdot Y)\; d\omega^2(x, y) \\ &= \iint_{\Omega_+^2} F(x + \varphi, y + \varphi)\; K^{(2)}(\rho, x \cdot X, y \cdot Y)\; d\omega^2(x, y) \\ &= \iint_{\Omega_+^2} F(x, y)\, K^{(2)}(\rho, x \cdot X, y \cdot Y)\; d\omega^2(x, y) = I^{(2)}(\rho, F(X, Y)) \end{aligned}$$

Therefore, we'll naturally define an isotropic kernel:

$$K^{iso}(\rho, (x, y), (X, Y)) = \frac{1}{2\pi} \int_{\varphi=0}^{2\pi} K^{(2)}(\rho, x \cdot (X + \varphi), y \cdot (Y + \varphi))\; d\varphi$$

An analytic expression of this kernel can be computed. But instead we'll use a slighty different one, with a maximum replacing the integral. Since $B(\rho, .)$ is a growing function, it's easier to compute.

$$\begin{gathered} K^{iso}(\rho, (x, y), (X, Y)) = \max_{\varphi \in [0; 2\pi]} K^{(2)}(\rho, x \cdot (X + \varphi), y \cdot (Y + \varphi)) = \\ \hat{B}\left(\rho, \tfrac{1}{2} \max_{\varphi \in [0; 2\pi]} x \cdot (X{+}\varphi), y \cdot (Y{+}\varphi)\right) = \tfrac{1}{2}\Big(\big(\sin^2\theta_x\, sin^2\theta_X + \sin^2\theta_y \sin^2\theta_Y + \\ 2 \sin\theta_x \sin\theta_X \sin\theta_y \sin\theta_Y \cos(\Delta\varphi_{xy} - \Delta\varphi_{XY})\big)^{\frac{1}{2}} + \cos\theta_x \cos\theta_X + \cos\theta_y \cos\theta_Y\Big) \end{gathered}$$

5 Error measurement

Once a BRDF is reconstructed, we need to evaluate the error with the original function. As far as image synthesis is concerned, the only criterium is the visual quality of the rendered image. But scenes can be very different, and a mathematic error between BRDF seems to be the only reliable method to quantify our results.

5.1 Measures definition

Since BRDFs are not mere functions, we'll use two different measures. First, a classic integral over the definition domain. Second, the same integral, but for the product of the BRDF with the cosine of the reflected direction. The former is a function measure, the latter an energetic measure:

$$\mu_f(F) = \frac{1}{4\pi^2} \iint_{(X,Y) \in \Omega_+^2} F(X, Y)\, d\omega_X\, d\omega_Y \quad \mu_e(F) = \frac{1}{2\pi^2} \iint_{(X,Y) \in \Omega_+^2} F(X, Y)\, \cos\theta_Y\, d\omega_X\, d\omega_Y$$

5.2 Errors definition

Let $\tilde{F}$ be the recomposed BRDF, and F the original one. Usually, the error between the two functions is written as:

$$\epsilon = \frac{1}{4\pi^2} \iint_{(X,Y)\in\Omega_+^2} \frac{\left|\tilde{F}(X,Y) - F(X,Y)\right|}{F(X,Y)} \, d\omega(X)\, d\omega(Y)$$

This error is the average percentage of error between values of $\tilde{F}$ and F. Now let's define:

$$\epsilon_f = \sqrt{\frac{\mu_f\left((\tilde{F}-F)^2\right)}{\mu_f(F^2)-\mu_f(F)^2}} \quad \epsilon_e = \sqrt{\frac{\mu_e\left((\tilde{F}-F)^2\right)}{\mu_e(F^2)-\mu_e(F)^2}}$$

These errors are not only RMS between the two functions, since they are divided by the own RMS of the original function. This ratio ensures us a scale independent error. Furthermore, if the recomposition is a constant function evaluating to the average value of the original function, the error will be 1 (100%). ϵ_f is called *functional error*, wheras ϵ_e is called *energetic error*. We'll see in section **??** that they can be very different.

5.3 Discretization

It's straightforward to discretize those errors ($c_i = \cos\theta_{X_i} + \cos\theta_{Y_i}$):

$$\epsilon_f = \sqrt{\frac{\sum_{i=1}^{n}\left(\tilde{F}(X_i,Y_i)-F_i\right)^2}{\sum_{i=1}^{n}F_i^2-\frac{1}{n}\left(\sum_{i=1}^{n}F_i\right)^2}} \quad \epsilon_e = \sqrt{\frac{\sum_{i=1}^{n}\left(\left(\tilde{F}(X_i,Y_i)-F_i\right)c_i\right)^2}{\sum_{i=1}^{n}(F_i\,c_i)^2-\frac{1}{n}\left(\sum_{i=1}^{n}(F_i\,c_i)\right)^2}}$$

5.4 Visual aspect

The visual aspect of the BRDF itself (not a rendered image, but the function itself) is a criterium we don't know reliability. For instance, on figure **??**, we've got two different representations (the first one uses spherical harmonics, the second one uses Zernike polynomials), that have about the same error with the original function ($\epsilon_f = 54\%$, $\epsilon_e = 51\%$ for the middle one, $\epsilon_f = 51\%$, $\epsilon_e = 49\%$ for the right one, the left one being the original BRDF).

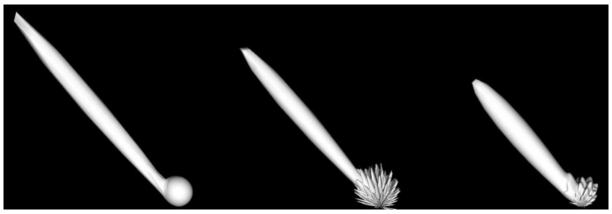

Fig. 5. original BRDF of a ceramic tile, spherical harmonics, Zernike polynomials

Which one will produce the best image ? The middle one has a specular peak that's closer to the original one, but the right one has a diffuse part that's less noisy. Visual aspect is an interesting point of view, but we can't get any serious conclusion from it.

6 Implementation

6.1 Hierarchical subdivision

In order to use the method defined in section **??**, we need a hierarchical and regular subdivision of Ω. A suitable structure is the icosahedron (20 equilateral faces), where each face can be divided into four new equilateral smaller faces (see figure **??**). Then we can hold the non-zero coefficients in a sparse matrix.

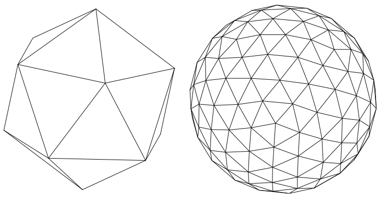

Fig. 6. icosahedron

6.2 Kernel localization

So as to efficiently evaluate our recomposition, we need to localize the kernels. Each type of kernel has its own localization properties, and with a few computations it's rather easy to find it's support. For instance, let us consider the isotropic kernel. We'll do a backward reasoning, to get a sufficient condition of our kernel being zero. If we choose to localize it using θ_x and θ_y we can write:

$$\begin{gathered}
\text{the kernel located at } (X,Y)\text{, with parameter } \rho \text{ doesn't need to be taken into account} \\
\Leftarrow \forall \varphi_{xy} \in [0:\pi]\ ,\ \tfrac{1}{2}\big(\cos\theta_x \cos\theta_X + \cos\theta_y \cos\theta_Y + \\
\sqrt{\sin^2\theta_x \sin^2\theta_X + \sin^2\theta_y \sin^2\theta_Y + 2\sin\theta_x \sin\theta_X \sin\theta_y \sin\theta_Y \cos(\varphi_{xy} - \varphi_{XY})}\big) \leq \rho \\
\Leftarrow \text{(maximum value reached for } \varphi_{xy} = \varphi_{XY}\text{)}\ \cos\theta_x \cos\theta_X + \cos\theta_y \cos\theta_Y + \\
\sqrt{\sin^2\theta_x \sin^2\theta_X + \sin^2\theta_y \sin^2\theta_Y + 2\sin\theta_x \sin\theta_X \sin\theta_y \sin\theta_Y} \leq 2\rho \\
\Leftarrow \cos\theta_x \cos\theta_X + \cos\theta_y \cos\theta_Y + \sin\theta_x \sin\theta_X + \sin\theta_y \sin\theta_Y \leq 2\rho \\
\Leftarrow \cos(\theta_x - \theta_X) \leq 2\rho - 1 \text{ or } \cos(\theta_y - \theta_Y) \leq 2\rho - 1
\end{gathered}$$

We then just have to subdivide $[0 : \frac{\pi}{2}]$ into bands, and a list of potentially non-zero kernels will be stocked into each band.

7 Some results

7.1 Comparison method

In this section, we'll compare our methods with classical methods, that is to say spherical harmonics (SH) and Zernike polynomials (ZP). We'll express the differences with our error criteria, with very simplistic images, and with visual aspects. We'll use different parameters to compare BRDF representations: the number of non-zero coefficients (n_c), functional error (ϵ_f), energetic error (ϵ_e), average evaluation time (t_e), the computation time of the representation (t_c). The most important parameters are obviously the average evaluation time and the two kinds of error.

7.2 Explicit results

All the code was written in `C++`, in the same library, so as to use the same level of optimisation. All the computations were done on `Indigo`^{2} `R10000` workstations.

type	SH	SH	ZP	ZP	1 level	2 levels	3 levels	4 levels
n_c	285	819	285	819	245	700	962	1326
ϵ_f	72%	54%	68%	51%	49%	34%	27%	21%
ϵ_e	69%	51%	67%	49%	51%	38%	29%	24%
t_e	0.22ms	0.41ms	0.19ms	0.34ms	0.05ms	0.08ms	0.08ms	0.09ms
t_c	15mn	42mn	15mn	42mn	8s	25s	43s	1mn

Fig. 7. comparison of representations for a ceramic tile BRDF

First example. First, we give some results for the BRDF of a ceramic tile. This BRDF has a strong specular peak, which intensity widely increases for grazing incident directions. Rightmost columns are using our method, with a different number of levels in the hierarchical representation.

This BRDF cannot be easily represented with other methods, because of the very strong specular peak. As a consequence, there's a lot of noise on reconstructed BRDFs (see figure **??**).

Second example. We'll study the visual aspect of our recomposition. Figure **??** is the recomposition of a slate BRDF [**?**], using very few caps (about 60). The computed error is $\epsilon_f = 12\%$ and $\epsilon_e = 15\%$, which is a very satisfying value.

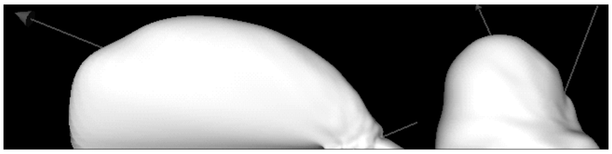

Fig. 8. recomposed BRDF for two different incident directions (60^o and 30^o)

We can see that the recomposed BRDF has a smooth aspect, but there is also a problem of irregularity in the specular peak, especially for the 60^o incidence.

Third example. We'll do a test render, with an ultra-simplistic scene, to show that the most important thing in BRDF reconstruction is the quality of a rendered image.

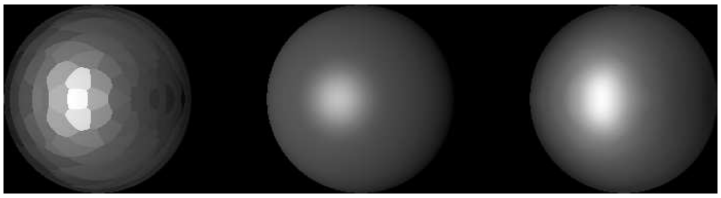

Fig. 9. spheres rendered with measured, analytic and recomposed BRDF

The leftmost sphere is rendered using real BRDF data of a slate (the closer sample is used, leading to that piecewise look). The middle one is rendered with a cosine lobe BRDF [**?**] ($\epsilon_f = 33\%, \epsilon_e = 34\%$). The rightmost one is rendered with our method, with very few caps (about 50, $\epsilon_f = 24\%, \epsilon_e = 27\%$). Errors are on the same level, but middle and right images are rather different. Furthermore, the radiance values in both images are very different too. It's an important fact, since the BRDF look is rather common (a diffuse part and a "Phong-like" part).

8 Conclusion

The method we exposed in this article allows us to reconstruct coherent BRDF data, fulfilling physical constraints. We think this method presents some advantages.

First, it's a hierarchical approach and we may control the error. Second it is also a fast method, thanks to locally supported basis. As a matter of fact, the evaluation time doesn't increase a lot, even if the number of coefficients increases. It is also very useful when few samples were made. Finally it is very easy to implement and can handle very different sample scattering.

We've begun a study of error evaluation issue. But it is only a first approach, and we need to extend this research. We're also integrating our method in our ray-tracer to render complex scenes with real materials.

References

1. R. LEWIS, *Making Shaders More Physically Plausible*, Fourth Eurographics Workshop on Rendering, 1993, pp. 47-61.
2. R. L. COOK AND K. E. TORRANCE, *A reflectance model for computer graphics*, Computer Graphics, vol. 15, 1981, pp. 307-316.
3. M. OREN AND S. NAYAR, *Generalisation of Lambert's Reflectance Model*, Computer Graphics, vol. 22(4), 1994, pp. 239-246.
4. X.D. HE, K. E. TORRANCE, F. X. SILLION AND D. P. GREENBERG, *A Comprehensive Physical Model for Light Reflection*, Computer Graphics, vol. 25, 1991, pp. 175-186.
5. B. PHONG, *Illumination for Computer-generated Pictures*, Communications of the ACM, vol. 18(8), 1975, pp. 311-317.
6. C. SCHLICK, *An Inexpensive BRDF Model for Physically-Based Renderi ng*, Computer Graphics Forum, vol. 13., 1994, pp. 233-246.
7. G. J. WARD, *Measuring and Modeling Anisotropic Reflection*, Computer G raphics, vol. 26, 1992, pp. 265-272.
8. E. LAFORTUNE, S. FOO, K. E. TORRANCE AND D. P. GREENBERG, *Non-Linear Approximation of Reflectance Functions*, Computer Graphics, vol. 31, 1997, pp. 117-126.
9. S. WESTIN AND J. ARVO AND K. TORRANCE, *Predicting Reflectance Functions from Complex Surfaces*, Computer Graphics, vol. 26(2), 1992, pp. 255-264.
10. P. SCHRÖDER AND W. SWELDENS, *Spherical Wavelets: Efficiently Representing Functions on the Sphere*, Computer Graphics, vol. 29, 1995, pp. 161-172.
11. P. LALONDE AND A. FOURNIER, *Generating Reflected Directions from BRDF Data*, Computer Graphics Forum, vol. 16, 1997, pp. 293-300.
12. P. LALONDE, *Representations and Uses of Light Distribution Functions*, PhD thesis, The University of British Columbia, 1997.
13. K. J. DANA, S. K. NAYAR, B. VAN GINNEKEN AND J. J. KOENDERINK, *Reflectance and Texture of Real-World Surfaces*, CUCS-048-96, Columbia University Technical Report, 1996.
14. D. GARGAN AND F. NEELAMKAVIL, *Approximating Reflectance Functions using Neural Networks*, 9th Eurographics Workshop on Rendering, 1998, pp. 23-34.
15. T. PFLAUM, *Colour and Reflectance in Image Synthesis*, Master Thesis, University of Waterloo, 1996.
16. J.M. COULOMB AND C. BRUSQUE, *Propriétés de réflexion des surfaces. Le goniophotomètre du LCPC. Performances et perspectives d'études*, Bulletin des Laboratoires des Ponts et Chaussées, vol. 204, 1996, pp. 25-35.
17. J. T. KAJIYA, *Anisotropic Reflection Models*, Computer Graphics vol. 19, 1985, pp. 15-21.
18. P. POULIN AND A. FOURNIER, *A Model for Anisotropic Reflection*, Computer Graphics vol. 24, 1990, pp. 273-282.
19. J. S. GONDEK, G. W. MEYER AND J. G. NEWMAN, *Wavelength Dependent Reflectance Functions*, Computer Graphics, vol. 28, 1994, pp. 213-220.
20. S. M. RUSINKIEWICZ, *A New Change of Variables for Efficient BRDF Representation*, 1998 Eurographics Workshop on Rendering.
21. J DEYOUNG AND A. FOURNIER, *Properties of Tabulated Bidirectional Reflectance Distribution Functions*, Graphics Interface '97, pp. 47-55.