CEIG'09, San Sebastián, Sept. 9-11 (2009)
C. Andújar and J. LLuch (Editors)

BRDFLab: A general system for designing BRDFs

Adrià Forés[1], Sumanta N. Pattanaik[2], Carles Bosch[1], Xavier Pueyo[1]

[1]Geometry and Graphics Group, Universitat de Girona
[2]University of Central Florida

Abstract

This paper introduces a novel system for interactive modeling and designing of arbitrary BRDFs. The system is able to deal with BRDFs defined in a variety of forms, such as analytical models, measured data or data obtained by simulation. The system also allows designing BRDFs from scratch using a combination of different analytical lobes. Using the programmable graphics hardware, it then performs interactive display of the designed BRDF, and its rendering on objects lit by complex illumination. The system also allows the fitting of an input BRDF defined in any form to our analytical lobe combination, so that it can be efficiently evaluated with GPU based rendering. The idea behind this work is to make available a general system for designing, fitting and rendering BRDFs, that is intuitive and interactive in nature. We plan to use this as a tool for simulation and modeling of complex physically-based BRDFs, and thus provide access to a larger variety of material models to the rendering community.

Categories and Subject Descriptors (according to ACM CCS):
I.3.3 [Computer Graphics]: Picture/Image Generation—Viewing algorithms I.3.4 [Computer Graphics]: Graphics Utilities—Graphics editors

1. Introduction

Realistic material rendering is relevant in many applications of computer graphics. For this reason, a lot of research has been carried out for modeling and designing materials. All this research may be categorized into three different classes: development of analytical models, measurement of materials and simulation of materials.

In spite of the extensive research in the field, there still does not exist a common platform for visualizing and working with different types of BRDF (Bidirectional reflectance distribution function), including analytical models, measured data and simulations. Hence, any one working in BRDF design usually has to start from scratch to obtain basic functionalities like displaying, fitting or rendering.

In this paper, we present a novel system that provides all these functionalities for arbitrary BRDF models that takes advantage of current programmable graphics hardware. Our purpose is to provide an open framework that allows users to easily design and work with different models as well as to include new models on it. Such a system will also be valuable for the rendering community for carrying out further research in the field, for teaching purposes and for BRDF modeling in general.

Contributions: The main features of our system are:

- The ability to deal with different types of BRDFs. This includes analytical models as well as measured or simulated materials. For analytical models, a general representation based on combining different lobe models is also proposed.
- Interactive display of BRDF shape as a function of incident angles and other model specific parameters.
- Fitting from any BRDF type to our general analytical representation.
- Interactive rendering of any geometric object with the chosen BRDF and under different lighting conditions.
- An object-oriented, multi-platform and Open Source tool that can be easily extended to include new models and functionalities.

2. Background

In computer graphics, a lot of research effort has gone into representing material appearance. An accurate review can be found in [DRS07].

In the early days of Computer Graphics, analytical models that mimiced the light behavior over the materials were first developed [Pho75, Bli77]. Later on, more generalized models were proposed to better represent the light scattering over the surfaces [CT82, HTSG91, War92, LFTG97, AS00].

For more complex materials, simulations are usually carried out to measure how light interacts with the surface and its subsurface structure. From the simulation results, a BRDF describing the material appearance is then obtained [WAT92, HK93, APS00, JMLH01, Ger08].

In the recent years, the improved quality of digital image capturing devices has allowed faster ways to retrieve information of real materials by means of measurements. For this purpose, different setups have been designed to get the material properties efficiently [War92, DVGNK99, MPBM03, NDM05, GCG*05]. Most of these measured data are available to the rendering community as well.

In order to efficiently render data obtained from measurements or simulations, fitting to analytical models is also desirable. In [NDM05], a set of measured materials have been fitted to different analytic models. Fitting to measurements is also commonly used to demonstrate the strengths of new analytical models, as shown in [War92, LFTG97].

Interactive display of BRDF shapes is essential to understand the nature of the material. There is no doubt that most material researchers are developing such tools. However, to our knowledge BV [Rus98] is the only tool available to public. BV is able to display the shape of a few analytical models using different views, but it is currently outdated, does not take advantage of GPU techniques, and can not be easily extended.

There exists some interfaces in order to help defining and characterizing the behavior of a material as well. BRDF-Shop [CPK06], for instance, is an interface to create, model and render physically correct BRDFs based on defining the highlights of the desired model onto a spherical canvas. To modify the material properties in real time, some solutions have also been proposed in [BAOR06, BAEDR08].

In addition, commercial software like Maya, 3D Studio Max or LightWave allow to design materials based on some analytic models. However, only few models are usually supported and simple lit-sphere views are typically provided. None of these products support measured or simulated data nor functionalities such as displaying or fitting BRDFs.

To finally appreciate the appearance of a material, it is important to allow interactive rendering of the BRDF directly onto geometric objects. A number of techniques have been proposed to render some analytical models with complex lighting [MLH02, CK07]. Also, some works have presented solutions for directly render measured or arbitrary BRDFs [KM99, MAA01, SM02, Mat03, LRR04, DWd*08].

3. System Overview

We present a modular system that is able to create, display, fit and render BRDFs given in different forms. Figure 1 shows the main structure of our system.

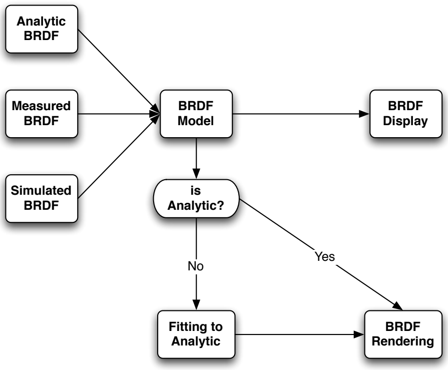


Figura 1: *System structure.*

The system takes as input a BRDF model defined in a variety of forms. The BRDF model can be displayed directly to let the user understand how the light interacts with the surface material. Analytical models can be rendered in real time as surface materials of complex objects, and for the other BRDF models; the fitting step is required to convert them into analytical equivalents before they can be rendered in real-time.

The system is able to create analytic models as a combination of different lobes, load measured data and perform material simulations. See Section 4.

Displaying and interacting with all types of supported BRDFs is also possible with the presented framework. The displaying of the BRDF allows to see the shape of the directional distribution of radiance as a function of incident irradiance due to light interaction with the material. See Section 5.

Fitting is another important feature of the presented system, allowing to fit from any BRDF model to any analytic model. See Section 6.

The rendering is also supported by the framework. This module allows to render geometric objects with complex lighting using the current BRDF as a material. See Section 7.

These features together gives us a system capable to design and analyze complex materials that is suited for use in real time rendering applications.

Section 8 presents some results obtained with our system, while conclusions and future work can be seen in Section 9.

4. Input models

As inputs, our system is able to handle different BRDF representations including analytic models, measured materials and simulated materials.

Analytical models tend to be computationally efficient and are commonly used by the rendering community due to their easy implementation and fast computation. Our system allows to create from scratch a combination of a set of different analytical models. A simple example will be a Lambertian lobe defining the diffuse term and a Lafortune lobe defining the specular term. Currently, Lambertian, Blinn-Phong [Bli77], Ward [War92], Lafortune [LFTG97] and Ashikhmin [AS00] lobe models are supported by the framework, although more models can be easily added.

The system is able to handle two different representations for the analytical models. Both representations can deal with multiple number of channels (for example, red, green and blue) and multiple number of lobes. The first representation is named Linear, as it uses a linear combination of different lobes. In this case, all the used channels share the same lobes but each lobe has channel-specific scale factors. The other approach, named Fixed, uses a different lobe for each channel in order to represent a completely different behavior for each one. In this case, no scale factors are necessary. In either case, the system guarantees energy conservation.

Measured materials are also supported by the framework. Our system uses measured data available from the MERL database [NDM05], the car paint database [GCG*05] and the Reflectance and Texture of Real World Surfaces database [DVGNK99]. The number of supported measured databases is easily expandable by simply writing the corresponding data parsing code.

Simulated materials are supported by linking specific libraries for material simulation or carrying out the full simulation on the system. At the moment, our system supports one external library, Scatmech simulation library [Ger08]. This library allows to perform a variety of material simulations over complex material structures to obtain the resulting BRDF.

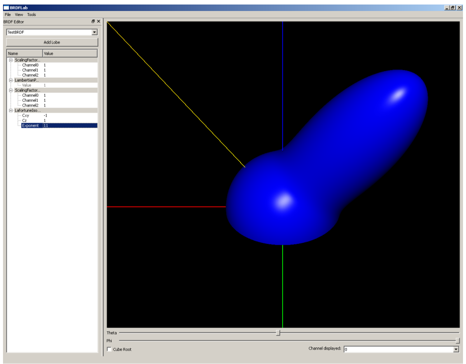
To increase the application usability our system defines a new XML filetype to load and save the designed BRDFs. For analytical models, all the lobes and parameters are stored in the file. For measured data, we store information about the database used and any material relevant information. The simulations store information about the library used and the parameter values. As our system allows to perform as many fittings as needed for a given material, all of them are also stored and retrieved from the same file.

5. Displaying BRDFs

By displaying the shape of the BRDF the user can get a feeling of the kind of BRDF he/she is dealing with and the shape of the different lobes (for example: a diffuse or specular like BRDF). The BRDF is a 4-dimensional function of the incoming and the outgoing directions. In order to display its shape, the user interactively modifies the incoming direction and the surface resulting from the BRDF evaluation for all possible outgoing directions is shown (see Figure 2).

For BRDF display, the idea is to start with a hemisphere where each vertex represents the outgoing intensity along direction $(\theta_{out}, \phi_{out})$. Given an incoming direction (θ_{in}, ϕ_{in}), a vertex shader is then used to evaluate the BRDF value corresponding to each $(\theta_{out}, \phi_{out})$. This gives the radial distance of each vertex with respect to the center of the hemisphere according to the given model. The position of the vertex, situated over a geodesic hemisphere of radius 1, is multiplied by this radial distance to determine the final position of the vertex in world coordinates.

For analytical models, the BRDF is directly evaluated on the GPU. This is performed by evaluating on the vertex shader all the lobes that compose a BRDF. The number of lobes, the lobe type, and its parameters are encoded in a list, allowing to parse efficiently this information on the GPU.



TestBRDF

Name	Value
ScalingFactor...	
Channel0	1
Channel1	1
Channel2	1
LambertianP...	
Value	1
ScalingFactor...	
Channel0	1
Channel1	1
Channel2	1
LafortuneIso...	
Cxy	-1
Cz	1
Exponent	11

☐ Cube Root

Channel displayed: 0

Figura 2: *Linear combination of a Lambertian lobe and an isotropic Lafortune lobe. Our system allows to interactively modify the incoming direction and the parameters of the lobes to see how the shape looks like.*

For displaying other models, like measured data, another approach must be taken. As measured data is not continuous, we offer the possibility to directly render the set of points representing the BRDF shape. Another option is to store the different values onto a texture indexed by $(\theta_{out}, \phi_{out})$ and reconstruct a continuous shape by taking advantage of texture interpolation. For a different incoming direction this texture must be replaced by the corresponding values in the CPU, but this is easily done at real time rates.

A similar approach can be used for simulated data, with the difference that the texture needs to be recomputed according to the change in incoming direction or change in the input parameters. The simulation result is stored in a texture indexed by $(\theta_{out}, \phi_{out})$ and the BRDF shape is reconstructed. By increasing the texture size, the quality of the displayed BRDF increases. The user can improve the quality of the displayed BRDF interactively at the expense of computational cost incurred due to change in input direction or parameter modification.

6. Fitting

The purpose of the fitting step is to approximate a complex BRDF with a set of analytic models. Analytic models are usually desired for rendering, as they tend to be computationally efficient and can be importance sampled.

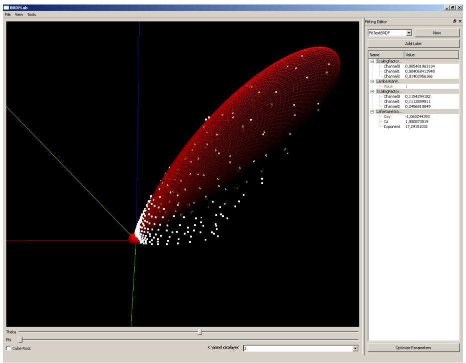
The system implements two strategies commonly used in the literature to fit a generic BRDF to an analytic model. The first one, used in [NDM05], is based on the linear combination of lobes, where all the BRDF channels share the same lobes with a different scaling factor for each channel. In this approach, the scaling factors are computed at each iteration step by using linear Least Squares optimization and the lobe parameters are computed by using non-linear optimization. The other approach used, as in [MWL*99], has a different lobe for each one of the channels. In this approach, all the parameters are optimized by non-linear optimization and the scaling factors are absorbed into the lobe parameters.

Non-linear optimization algorithms do not always converge to optimal solution. As many authors point out [LFTG97,MPBM03,NDM05], providing right initial parameter values is key to finding the best approximation. Our system allows the user to set the initial guess values that will be used on the optimization step. During the editing of these parameters, both the original and the modified BRDF models are displayed at the same time so that the user can see in real time how the parameter modifications approximate the original model. This is also used to display the modified model after the fitting process to evaluate the obtained results (see Figure 3). This allows the user to see how well a set of parameters approximate the data to be fitted and also see the fitting results. Also, the system allows to iterate the fitting step as many times as needed.

The non-linear optimization algorithm attempts to minimize the Least Square Error, defined by:

$$Error = \sum_{i=1}^{N} [y_i - f(x_i)]^2 \qquad (1)$$

where N is the number of directions, y_i is the input BRDF value for a specific direction, and $f(x_i)$ is the BRDF function evaluated at that given direction.



FitTestBRDF

Name	Value
ScalingFactor...	
Channel0	0,005481963134
Channel1	0,004068413948
Channel2	0,01403956166
LambertianP...	
Value	1
ScalingFactor...	
Channel0	0,1154294182
Channel1	0,1112899511
Channel2	0,2456810849
LafortuneIso...	
Cxy	-1,060244393
Cz	1,000873519
Exponent	17,29151031

☐ Cube Root

Channel displayed: 2

Figura 3: *The blue metallic paint material of the MERL database is fitted to a linear combination of a Lambertian lobe and an isotropic Lafortune lobe.*

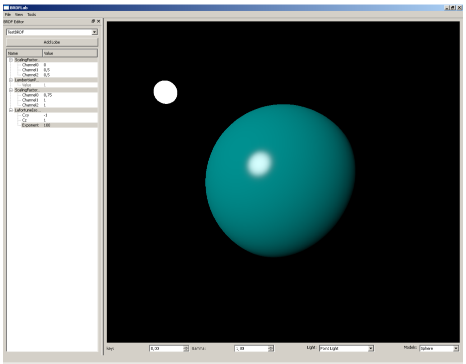
Generally speaking, what we are doing in the fitting step is to optimize a set of equations in order to find which parameters better approximate the input data. When analytic or simulated data is fitted, more precise results can be obtained by evaluating more directions. For example, when we are fitting from a simulated model to an analytic model, we can increase the sample density in order to gain fidelity. This implies having more equations to optimize and consequently increased computation to find out a good approximation. The isotropic BRDF models only have to be optimized considering three dimensions, as rotation around the surface normal (ϕ angle) does not change the model behavior.

The Levenberg-Marquardt non-linear optimization algorithm [Lou04] is currently used in the system. More optimization routines can be easily added to the system.

7. Rendering

The objective of the rendering part is to obtain images of a 3D objects with the given BRDF. We provide rendering of complex objects with analytic BRDF models as their surface material and under different lighting conditions. Only direct illumination is considered at the moment, but global effects could also be incorporated as well.

The first lighting model provided by the system is based on a point light source, with the ability to use either of the two analytical BRDF representations presented above, Linear and Fixed. This model consists of evaluating the given input model for each pixel by using the light direction as incoming direction and the eye direction as outgoing direction. This is computed in a fragment shader, where the light and eye vectors are transformed to tangent space in order to evaluate the BRDF. After this evaluation, the resulting BRDF value is multiplied by the intensity of the point light source and the cosine of the incident angle. A rendering of sphere using a Linear and a Fixed combination of analytic BRDF models can be seen in Figure 4 and 5, respectively. Complex models can also be interactively rendered using this technique, as can be seen in Figure 6.

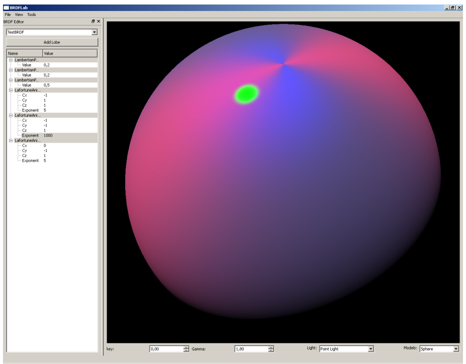


TestBRDF

Name	Value
ScalingFactor...	
Channel0	0
Channel1	0,5
Channel2	0,5
LambertianP...	
Value	1
ScalingFactor...	
Channel0	0,75
Channel1	1
Channel2	1
LafortuneIso...	
Cxy	-1
Cz	1
Exponent	100

key: 0,00 Gamma: 1,80 Light: Point Light Models: Sphere

Figura 4: *Point light rendering of a sphere. The BRDF uses a Linear combination of a Lambertian lobe and an isotropic Lafortune lobe, where all the channels share the same lobe.*



TestBRDF

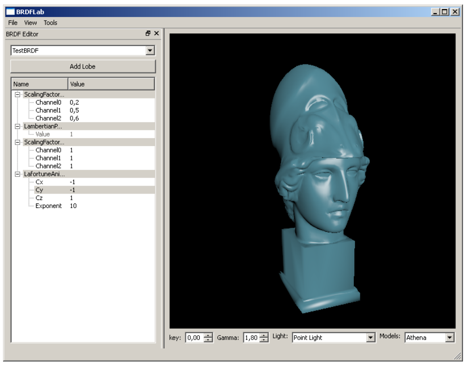
Name	Value
LambertianP...	
Value	0,2
LambertianP...	
Value	0,2
LambertianP...	
Value	0,5
LafortuneAni...	
Cx	-1
Cy	1
Cz	1
Exponent	5
LafortuneAni...	
Cx	-1
Cy	-1
Cz	1
Exponent	1000
LafortuneAni...	
Cx	0
Cy	-1
Cz	1
Exponent	5

key: 0,00 Gamma: 1,80 Light: Point Light Models: Sphere

Figura 5: *Point light rendering of a sphere. The BRDF uses a Fixed combination of a Lambertian lobe and an anisotropic Lafortune lobe, where each channel uses a different lobe. The flexibility of this representation is shown.*

To render geometric objects with complex lighting, the system also provides a rendering based on a HDR environment map. In this case, we use a modification of the real-time shading with filtered importance sampling proposed by Colbert and Krivanek [CK07]. Their approach uses spherical harmonics for the diffuse term, and importance samples an analytic lobe to compute the specular term. The specular term uses the appropriate mipmap level of the environment map to reduce the sampling rate normally required for noise free rendering. The samples generated in the low gradient areas of the PDF (probability density function) access higher mipmap levels, and samples from high gradient areas access lower mipmap levels.

Our modification of the rendering method allows to use more than a single analytical model to compute the BRDF. The number of samples selected by the user are distributed over the different lobe models that compose the BRDF. For Lambertian lobe, if it exists, instead of using spherical harmonics as proposed by Colbert and Krivanek [CK07], we uniformly sample the hemisphere and access the environment light from a higher mipmap level. For the other analytical lobes, importance sampling and appropriate mipmap level access are done as described in the previous paragraph. All the directions are computed on the CPU and passed on to the GPU. A rendered image using this technique can be seen in Figure 7.

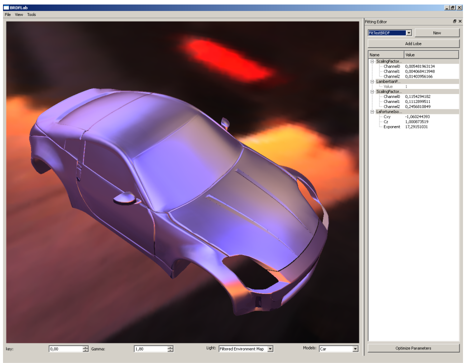


TestBRDF

Name	Value
ScalingFactor...	
Channel0	0,2
Channel1	0,5
Channel2	0,6
LambertianP...	
Value	1
ScalingFactor...	
Channel0	1
Channel1	1
Channel2	1
LafortuneAni...	
Cx	-1
Cy	-1
Cz	1
Exponent	10

key: 0,00 Gamma: 1,80 Light: Point Light Models: Athena

Figura 6: *Point light rendering of the Athena model. The BRDF used is a Linear combination of a Lambertian lobe and an isotropic Lafortune lobe.*



FitTestBRDF

Name	Value
ScalingFactor...	
Channel0	0,005481963134
Channel1	0,004068413948
Channel2	0,01403956166
LambertianP...	
Value	1
ScalingFactor...	
Channel0	0,1154294182
Channel1	0,1112899511
Channel2	0,2456810849
LafortuneIso...	
Cxy	-1,060244393
Cz	1,000873519
Exponent	17,29151031

key: 0,00 Gamma: 1,80 Light: Filtered Environment Map Models: Car

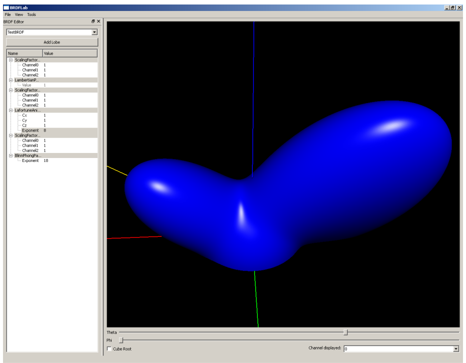
Figura 7: *Real time rendering of a car model with the fitted material in Figure 3, using filtered importance sampling environment mapping light.*

The objective of the material measurements and simulations is to capture the complex behavior of certain materials and to render objects as close as possible as they look in the real world. Once the fit is performed on those materials, they can be efficiently rendered in real time on our system.

8. Results

The presented system has been developed using OGRE [Str01] as the graphics engine and Qt for the GUI interface. The presented results were obtained using an Intel Core 2 Duo at 2.40 GHz, 4 GB of RAM memory and a NVIDIA GeForce 8800 GTX graphics card.

Figure 2 and Figure 8 first show two analytic BRDF models designed with our system. Figure 2 shows a combination of a Lambertian lobe and an isotropic Lafortune lobe, while Figure 8 shows a combination of a Lambertian lobe, an anisotropic Lafortune lobe and a Blinn-Phong lobe. In the later, the anisotropic Lafortune lobe models the retro-reflection effect, meaning that light is also reflected in the incident direction. These BRDFs are displayed in real-time at 430 frames per second by using a 328192 triangle geodesic hemisphere.



TestBRDF

Name	Value
ScalingFactor...	
Channel0	1
Channel1	1
Channel2	1
LambertianP...	
Value	1
ScalingFactor...	
Channel0	1
Channel1	1
Channel2	1
LafortuneAni...	
Cx	1
Cy	1
Cz	1
Exponent	8
ScalingFactor...	
Channel0	1
Channel1	1
Channel2	1
BlinnPhongPa...	
Exponent	18

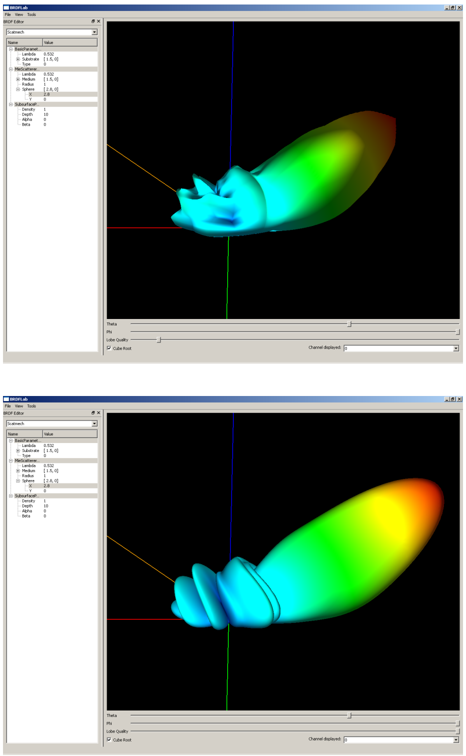
☐ Cube Root

Channel displayed: 0

Figura 8: *Linear combination of a Lambertian lobe, an anisotropic Lafortune lobe and a Blinn-Phong lobe.*

Figure 9 shows a complex BRDF obtained by simulation using the subsurface particle model from the Scatmech library. The texture size used to generate the top image is of 32 × 32 and took 0.125 seconds to compute, the bottom one uses a texture size of 150 × 150 and took 2.5 seconds to compute. As can be seen, lower texture sizes can be used to get faster feedback of the model when the user is modifying the parameters. Once the texture is computed, the user can look over the BRDF shape in real time at 980 frames per second. As reflectance values are computed on the CPU during the simulation, the maximum value can be easily determined and a transfer function can then be applied to better understand those complex models, as shown in the figure. In Figure 10, another simulation using the same model with different parameters is shown.

Figure 3 shows the fitting of the blue metallic paint material of the MERL database with a linear combination of a Lambertian lobe and a Lafortune lobe. The least square error obtained is 24.074 and the optimization step takes 45.2 seconds. Performing the same fitting with a linear combination of a Lambertian lobe, a Lafortune lobe, and an Ashikhmin lobe an error of 7.611 is obtained in 94.2 seconds.



Scatmech

Name	Value
BasicParamet...	
Lambda	0.532
Substrate	[1.5, 0]
Type	0
MieScatterer...	
Lambda	0.532
Medium	[1.5, 0]
Radius	1
Sphere	[2.8, 0]
X	2.8
Y	0
SubsurfaceP...	
Density	1
Depth	10
Alpha	0
Beta	0

☑ Cube Root

Channel displayed: 0

Scatmech

Name	Value
BasicParamet...	
Lambda	0.532
Substrate	[1.5, 0]
Type	0
MieScatterer...	
Lambda	0.532
Medium	[1.5, 0]
Radius	1
Sphere	[2.8, 0]
X	2.8
Y	0
SubsurfaceP...	
Density	1
Depth	10
Alpha	0
Beta	0

☑ Cube Root

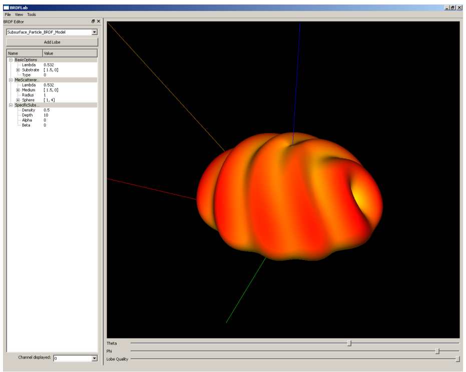
Channel displayed: 0

Figura 9: *Simulation obtained with the Scatmech library using a subsurface particle model. Top using a* 32*x*32 *resolution texture and bottom* 150*x*150*.*

Figure 4 shows the rendering of a sphere with a point light source. The BRDF uses a Linear combination of a Lambertian lobe and an isotropic Lafortune lobe. The framerate obtained on this rendering is 480 frames per second.

Figure 5 also shows a sphere with point light rendering, but the Fixed representation is used to combine different Lambertian and anisotropic Lafortune lobes for each channel. The Fixed representation uses isotropic Lafortune coefficients for the green channel (i.e. Cx = Cy) and anisotropic coefficients for the red and blue channels. The later, however, use different anisotropic directions. The framerate obtained on this rendering is 400 frames per second.

Figure 6 shows the rendering of a car with a point light source. Figure 7 shows the rendering of the blue metallic paint material, previously fitted, with our implementation of [CK07] based on importance sampling. The framerate obtained on this rendering is 490 fps on the point light source and 85 frames per second on the environment map version. For the later, 80 samples were used.



Subsurface_Particle_BRDF_Model

Name	Value
BasicOptions	
Lambda	0.532
Substrate	[1.5, 0]
Type	0
MieScatterer...	
Lambda	0.532
Medium	[1.5, 0]
Radius	1
Sphere	[1, 4]
SpecificSubs...	
Density	0.5
Depth	10
Alpha	0
Beta	0

Channel displayed: 0

Figura 10: *Simulation obtained with the Scatmech library using a subsurface particle model and a* 150*x*150 *resolution texture.*

9. Conclusions & Future Work

We have presented a new system for interactively designing, fitting and rendering a wide set of BRDFs. The system provides interactive visualization of analytical models, measured data and simulation. It also provides interactive rendering of complex objects with general analytical models as the material and different lighting conditions.

This system on completion will be released in the public domain as an Open Source software. We believe that this tool will be valuable to the rendering community for teaching, research and development of reflective materials. We also foresee its use in other areas of material research such as Optics and Remote Sensing.

Before releasing the system, we plan to add the following features to the system.

- Capability to render objects directly with the measured/simulated BRDF.
- Tools to compare the rendering of the objects with measured/simulated data vs the rendering with the corresponding fitted analytical models.
- Tools to directly perform simulations in the GPU and compute BRDFs by varying surface and subsurface properties.

10. Acknowledgements

We want to thank Ismael García and Francisco González for the helpful discussions during the application development. This project was funded by grant TIN2007-67120 from the Spanish government.

References

[APS00] Ashikmin M., Premože S., Shirley P.: A microfacet-based BRDF generator. In *SIGGRAPH '00: Proceedings of the 27th Annual Conference on Computer graphics and Interactive Techniques* (2000), pp. 65–74.

[AS00] Ashikhmin M., Shirley P.: An anisotropic Phong BRDF model. *Journal of Graphic Tools 5*, 2 (2000), 25–32.

[BAEDR08] Ben-Artzi A., Egan K., Durand F., Ramamoorthi R.: A precomputed polynomial representation for interactive BRDF editing with global illumination. *ACM Transactions on Graphics 27*, 2 (2008), 1–13.

[BAOR06] Ben-Artzi A., Overbeck R., Ramamoorthi R.: Real-time BRDF editing in complex lighting. *ACM Transactions on Graphics 25*, 3 (July 2006), 945–954.

[Bli77] Blinn J. F.: Models of light reflection for computer synthesized pictures. In *SIGGRAPH '77: Proceedings of the 4th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1977), ACM, pp. 192–198.

[CK07] Colbert M., Křivánek J.: *GPU Gems 3*. Addison-Wesley, 2007, ch. GPU-Based Importance Sampling, pp. 459–474.

[CPK06] Colbert M., Pattanaik S., Krivanek J.: BRDF-Shop: Creating physically correct bidirectional reflectance distribution functions. *IEEE Comput. Graph. Appl. 26*, 1 (2006), 30–36.

[CT82] Cook R. L., Torrance K. E.: A reflectance model for computer graphics. *ACM Transactions on Graphics 1*, 1 (1982), 7–24.

[DRS07] Dorsey J., Rushmeier H., Sillion F.: *Digital Modeling of Material Appearance*. Morgan Kaufmann / Elsevier, 2007.

[DVGNK99] Dana K., Van-Ginneken B., Nayar S., Koenderink J.: Reflectance and texture of real world surfaces. *ACM Transactions on Graphics 18*, 1 (Jan 1999), 1–34.

[DWd*08] Donner C., Weyrich T., d'Eon E., Ramamoorthi R., Rusinkiewicz S.: A layered, heterogeneous reflectance model for acquiring and rendering human skin. *ACM Transactions on Graphics 27*, 5 (2008), 1–12.

[GCG*05] Günther J., Chen T., Goesele M., Wald I., Seidel H.-P.: Efficient acquisition and realistic rendering of car paint. In *Proceedings of 10th International Fall Workshop - Vision, Modeling, and Visualization (VMV) 2005* (Nov. 2005), pp. 487–494.

[Ger08] Germer T.: SCATMECH: Polarized light scattering C++ class library. [web page] http://physics.nist.gov/Divisions/Div844/facilities/scatmech/, May. 2008. [Accessed on 20 April 2009].

[HK93] Hanrahan P., Krueger W.: Reflection from layered surfaces due to subsurface scattering. In *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1993), ACM, pp. 165–174.

[HTSG91] He X. D., Torrance K. E., Sillion F. X., Greenberg D. P.: A comprehensive physical model for light reflection. *SIGGRAPH Comput. Graph. 25*, 4 (1991), 175–186.

[JMLH01] Jensen H. W., Marschner S. R., Levoy M., Hanrahan P.: A practical model for subsurface light transport. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2001), ACM, pp. 511–518.

[KM99] Kautz J., McCool M. D.: Interactive rendering with arbitrary BRDFs using separable approximations. In *SIGGRAPH '99: ACM SIGGRAPH 99 Conference abstracts and applications* (New York, NY, USA, 1999), ACM, p. 253.

[LFTG97] Lafortune E. P., Foo S.-C., Torrance K. E., Greenberg D. P.: Non-linear approximation of reflectance functions. In *Proc. SIGGRAPH '97* (1997), vol. 31, pp. 117–126.

[Lou04] LOURAKIS M.: levmar: Levenberg-Marquardt nonlinear least squares algorithms in C/C++. [web page] http://www.ics.forth.gr/~lourakis/levmar/, Jul. 2004. [Accessed on 20 April 2009].

[LRR04] LAWRENCE J., RUSINKIEWICZ S., RAMAMOORTHI R.: Efficient BRDF importance sampling using a factored representation. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers* (New York, NY, USA, 2004), ACM, pp. 496–505.

[MAA01] MCCOOL M. D., ANG J., AHMAD A.: Homomorphic factorization of BRDFs for high-performance rendering. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 2001), ACM, pp. 171–178.

[Mat03] MATUSIK W.: *A Data-Driven Reflectance Model*. PhD thesis, Massachusetts Institute of Technology, 2003.

[MLH02] MCALLISTER D. K., LASTRA A., HEIDRICH W.: Efficient rendering of spatial bi-directional reflectance distribution functions. In *HWWS '02: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware* (Switzerland, 2002), Eurographics Association, pp. 79–88.

[MPBM03] MATUSIK W., PFISTER H., BRAND M., MCMILLAN L.: A data-driven reflectance model. *ACM Transactions on Graphics 22*, 3 (July 2003), 759–769.

[MWL*99] MARSCHNER S. R., WESTIN S. H., LAFORTUNE E. P., TORRANCE K. E., GREENBERG D. P.: Image-based BRDF measurement including human skin. In *Rendering Techniques* (1999), Lischinski D., Larson G. W., (Eds.), Springer, pp. 131–144.

[NDM05] NGAN A., DURAND F., MATUSIK W.: Experimental analysis of BRDF models. In *Proceedings of the Eurographics Symposium on Rendering* (2005), Eurographics Association, pp. 117–226.

[Pho75] PHONG B. T.: Illumination for computer generated pictures. *Communications of the ACM 18*, 6 (1975), 311–317.

[Rus98] RUSINKIEWICZ S.: bv - a BRDF browser. [web page] http://graphics.stanford.edu/~smr/brdf/bv/, March 1998. [Accessed on 20 April 2009].

[SM02] STEIGLEDER M., MCCOOL M. D.: Factorization of the Ashikhmin BRDF for real-time rendering. *J. Graph. Tools 7*, 4 (2002), 61–68.

[Str01] STREETING S.: OGRE - Open Source 3D Graphics Engine. [web page] http://www.ogre3d.org/, 2001. [Accessed on 20 April 2009].

[War92] WARD G. J.: Measuring and modeling anisotropic reflection. *SIGGRAPH Comput. Graph. 26*, 2 (1992), 265–272.

[WAT92] WESTIN S. H., ARVO J. R., TORRANCE K. E.: Predicting reflectance functions from complex surfaces. *SIGGRAPH Comput. Graph. 26*, 2 (1992), 255–264.