DOI: 10.1111/j.1467-8659.2010.01744.x

Eurographics Symposium on Rendering 2010
Jason Lawrence and Marc Stamminger
(Guest Editors)

Volume 29 (2010), Number 4

A PCA Decomposition for Real-time BRDF Editing and Relighting with Global Illumination

Chuong H. Nguyen [1] Min-Ho Kyung [2] Joo-Haeng Lee [1] Seung-Woo Nam [1]

[1]Electronics and Telecommunications Research Institute, Korea
[2]Division of Digital Media, Ajou University, Korea

Abstract

We propose a novel rendering method which supports interactive BRDF editing as well as relighting on a 3D scene. For interactive BRDF editing, we linearize an analytic BRDF model with basis BRDFs obtained from a principal component analysis. For each basis BRDF, the radiance transfer is precomputed and stored in vector form. In rendering time, illumination of a point is computed by multiplying the radiance transfer vectors of the basis BRDFs by the incoming radiance from gather samples and then linearly combining the results weighted by user-controlled parameters. To improve the level of accuracy, a set of sub-area samples associated with a gather sample refines the glossy reflection of the geometric details without increasing the precomputation time. We demonstrate this program with a number of examples to verify the real-time performance of relighting and BRDF editing on 3D scenes with complex lighting and geometry.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.3]: Three-Dimensional Graphics and Realism—

1. Introduction

Since graphics hardware was first introduced, many techniques have been explored extensively for the real-time rendering of 3D scenes. This has contributed to a dramatic reduction in the cost and time of production as it provides instant feedback to designers working in the area of lighting and material setup. A common approach to supporting interactive rendering is to separate the static part of rendering from dynamically varying computations, as in the precomputed radiance transfer (PRT) method proposed in a recent study [SKS02].

Given the considerable size of the precomputed data, they are usually computed and stored in a per-vertex based manner. Per-vertex based computations, however, cannot generally capture pixel-scale details. Even worse, the stored data is highly compressed because the typical GPU memory size is not large enough to contain the data as it is. For a high compression ratio, a significant amount of data should be decimated and most of it is from high-frequency components. Thus, high-frequency changes of radiance transfer are usually smoothed out, which leads to a blurry reflection. Another drawback is that the radiance transfer data should be updated if the surface reflectance is modified, as the transferred light intensity can be affected by the new reflectance.

The goal of this work is to support real-time lighting and BRDF editing of a globally illuminated scene with enhanced glossy reflection. For BRDF editing, we convert an analytic BRDF model approximately to a linear combination of basis BRDFs in which the surface BRDF is edited simply by modifying the coefficient values. Basis BRDFs are obtained from a principal component analysis of a set of BRDFs, snapshots of the target BRDF model at different parameter values. For every pixel, the light transfer is precomputed with each basis BRDF, as it is in other PRT methods. At run time, the pixel illumination is computed by multiplying the light transfer data of basis BRDFs with the incoming radiance field and linearly combining the results with coefficients. The coefficient values are controlled by an editing parameter.

For enhanced accuracy of the glossy reflection, we employ a level-of-detail approach with sub-area samples. As in the direct-to-indirect light transfer by Hasan et al. [HPB06], our algorithm gathers radiance from pre-sampled surface points to compute the illumination of a pixel. The surface points are termed gather samples. This approach, how-

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd. Published by Blackwell Publishing, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main Street, Malden, MA 02148, USA.

ever, produces blurry and irregular highlight reflection for a glossy BRDF, the peak of which is often missed by the gather samples. We enhance the highlight reflection by increasing the sample density around a peak gather sample with additional sub-area samples. To minimize the computational cost, which is increased by the sub-area samples, they inherit their geometric properties from the associated gather samples.

2. Related Work

Many-light: Indirect lighting can be approximated with many virtual point lights attached to lit surfaces. Recently, instant radiosity technique [Kel97] approximated a point light attached to a surface patch as radiosity was emitted from the patch. Using many point lights, however, slows the rendering speed considerably as the number of lights increases. Walter et al. [WFA*05] introduced a light-cut method using a hierarchy of point lights to speed up rendering with many lights. Hasan et al. [HPB07] used a matrix row-column sampling method that accelerated the large-scale matrix-vector multiplication for many-light rendering.

All of these earlier works produced visually plausible results in a reasonable amount of time for diffuse surfaces; however, they often failed to render a scene with glossy surfaces, especially when the number of point lights was not sufficiently large and when their distributions were not properly created. To overcome this limitation, Hasan et al. [HKWB09] introduced virtual spherical lights. While they enable the high-quality rendering of glossy surfaces in the many-light framework, they are not appropriate for interactive rendering.

Precomputed Radiance Transfer (PRT): The PRT methods amortized the expensive rendering cost by factorizing the rendering equation into a static and a dynamic term. The static term, containing expensive computations such as visibility tests, is precomputed for real-time use. The precomputed data, too large to be stored, is transformed into a compact representation. The spherical harmonics, the Haar wavelets [NRH04], a mixture of Gaussian [GKMD06] and a spherical radial basis [TS06] have been used to represent the precomputed data compactly. Wang et al. [WRG*09] used a spherical Gaussian basis in their rendering algorithm to render glossy surfaces not well supported by other representations accurately.

Interactive BRDF editing: Sun et al. [SZC*07] proposed an interactive BRDF editing method based on the in-out BRDF factorization developed by [WTL04]. Cheslack-Postava et al. [CPWÅP08] proposed a nonlinear cut approximation method for efficient relighting with dynamically varying BRDFs. These methods showed good interactive performance with a reasonable level of quality but are nonetheless limits in terms of an accurate rendering of specular surfaces. A precomputed polynomial basis used for compact BRDF representation by Ben-Artzi et al. [BAEDR08] supported specular BRDF editing, but it limits the number of available BRDFs in a scene and does not allow the lighting to be changed at run time.

Method	R	E	D1	G1	DN	GN	M
[CPWÅP08]	√	√	√	√			√
[HPB06]	√		√	√	√		
[BAEDR08]		√	√	√	√	√	
Our Method	√	√	√	√	√		

Table 1: *Comparison of previous interactive rendering algorithms. Our method fully supports relighting (R) and BRDF editing (E) including* 1*-bounce glossy (G*1*), 1 and multi-bounce diffuse (D*1*, DN) indirect lighting. Multi-bounce glossy (GN) and moving camera (M) are not supported by our method.*

Interactive relighting system: Early efforts on relighting focused on direct illumination for only the direct reflection of light. However, recent works have been more concerned with indirect illumination. Here, we will briefly discuss two previous methods that support indirect illumination, as both are closely related to our work. Lehtinen et al. [LZT*08] proposed a relighting method using a hierarchical basis function induced from scattered data. However, its splatting scheme was incapable of computing glossy illumination. The direct-to-indirect transfer framework proposed by Hasan et al. [HPB06] precomputed radiance transfer matrices from a set of gather samples and compressed them by wavelet transformation. Both of these techniques do not allow surface reflectance to be edited at run time.

Interactive global illumination: Wang et al. [WWZ*09] implemented a global illumination scheme based on photon mapping, final gathering and radiance caching fully on the GPU. Imperfect shadow map method [RGK*08] approximates visibility test in many point-lights approach. Another study [NSW09] exploits an image-based approach to produce plausible indirect lighting. All of these techniques cannot easily handle highly complex scenes with high glossy interreflections.

Here, a rendering algorithm based on the direct-to-indirect light transfer framework [HPB06] that supports high-frequency details of reflection and BRDF editing is developed. The extension can also be applied to other PRT-based rendering algorithms. A functional comparison of our algorithm with those in previous works is shown in Table 1.

3. Rendering Framework

The rendering algorithm proposed in this paper is based on a precomputed light transfer framework. We extend it using sub-area sampling and linear BRDF approximation to support the interactive editing of the lighting and material properties.

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

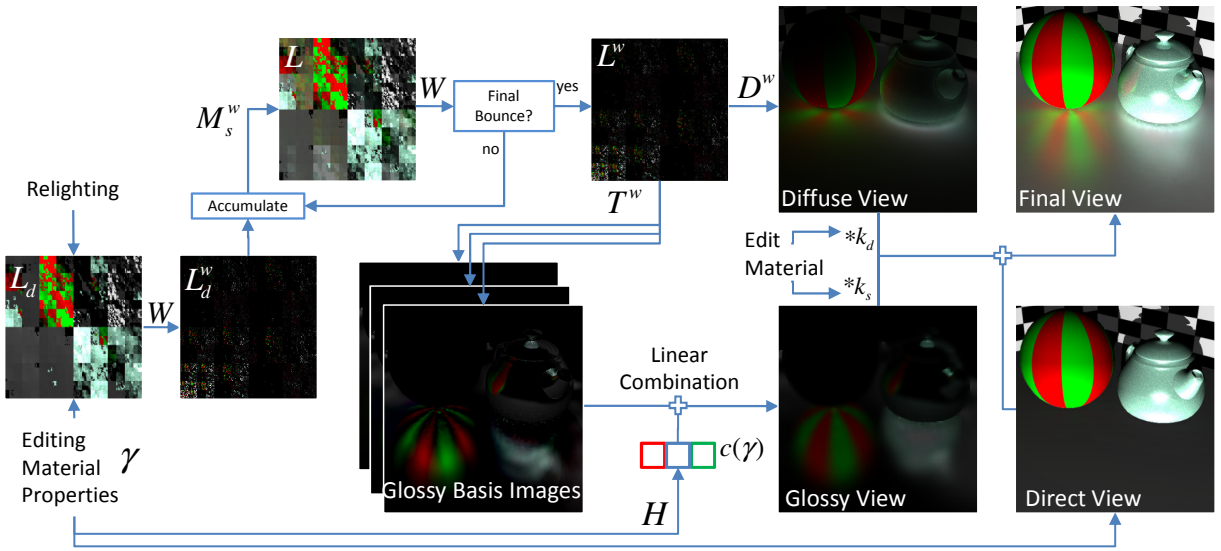

Figure 1: *An overview of the proposed algorithm (see Table 2 for notation description).* $\mathbf{M}_s^w$ *(Eq. 17),* $\mathbf{T}^w$ *(Eq. 7),* $\mathbf{D}^w$ *([HPB06]),* $\mathbf{H}$ *(Eq. 15) are precomputed and loaded into GPUs at run time. Direct lighting on the view and gather samples is computed using deep frame buffer caching shading parameters.*

Symbol	Description
$\mathbf{W}$	Wavelet transform
$\mathbf{L}_d$	Direct on gather samples
$\mathbf{L}$	Indirect on gather samples
$\mathbf{M}_s$	Single bounce gather matrix
$\mathbf{T}$	Glossy transfer basis matrix
$\mathbf{D}$	Diffuse transfer matrix
γ	BRDF editing parameters.
$\mathbf{H}$	BRDF basis projection matrix
$\mathbf{c}(\gamma)$	Projection of γ into basis space.
k_d	Diffuse coefficients
k_s	Glossy coefficients

Table 2: *Important notations. We use the superscript w to denote a matrix (or vector) projected into the wavelet space.*

3.1. Light Transfer Equation

We use the same light transfer setup used in the direct-to-indirect light transfer method [HPB06]. There are two sets of samples: a set of *view samples* and a set of *gather samples*, denoted by X and G, respectively. A view sample is a surface point visible from the viewpoint that is later mapped to a pixel of the rendered image. A gather sample is a surface point sampled in the target scene as a source of indirect illumination.

For a gather sample $g \in G$, the intensity of the reflection to the ray direction ω_o at view sample x, denoted by $I_g(x,\omega_o)$, is computed as follows:

$$I_g(x,\omega_o) = \int_{C_g} f(x,\omega_q,\omega_o)V(x,q)L(x,\omega_q)\cos\theta_q d\omega_q \quad (1)$$

Here, C_g is a cone of rays from x toward the surface region A_g, represented by a gather sample g, and $q = x+\omega_q \in A_g$. The function $V(x,q)$ is a visibility function between x and q, and θ_q is the angle between ω_q and the normal at x. $L(x,\omega_q)$ is the incoming radiance at x along ω_q; it is actually the radiance emitted from q. The total intensity of the reflection at x is simply the sum of all values of I_g:

$$I(x,\omega_o) = \sum_{g\in G} I_g(x,\omega_o) \quad (2)$$

Unlike the direct-to-indirect light transfer method, a gather sample is not regarded as a point light source but as an area light source. Although many gather samples are used to compute the indirect reflection, they are not sufficient enough to ensure that the details of specular reflection are accurate when using highly glossy BRDFs. Moreover, increasing the number of gather samples also increases the compression ratio of the radiance transfer data, which leads to a loss of the details acquired by the added samples. Importance-based sampling could be a solution to enhance the rendering accuracy, but this would require extensive runtime sampling because surface BRDFs vary dynamically. An area light is more capable of catching the peak reflection than a point light and using it can therefore improve the accuracy of the glossy reflection details.

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

3.2. Sub-area Sampling

As there is no known closed-form solution to Eq. 1, the Monte-Carlo integration is used to approximate it here. If we have a point set M_g sampled on A_g with a PDF p, Eq. 1 is written as

$$I_g(x,\omega_o) = \frac{1}{|M_g|}\sum_{q\in M_g}\frac{f(x,\omega_q,\omega_o)V(x,q)L(x,\omega_q)\cos\theta_q}{p(\omega_q)} \tag{3}$$

A point q in M_g is termed a sub-area sample of g.

Eq. 3 is still too heavy to evaluate for all instances of $g \in G$ at run time. For further approximation, we assume that region A_g has uniform light intensity and that its visibility is identical to that of gather point g. Under these assumptions, $L(x,\omega_q) = L_g$ and $V(x,q) = V_g(x)$. Thus, Eq. 3 is simplified to

$$\begin{aligned} I_g(x,\omega_o) &= L_g\frac{V_g(x)}{|M_g|}\sum_{q\in M_g}\frac{f(x,\omega_q,\omega_o)\cos\theta_q}{p(\omega_q)} \\ &= L_g\cdot T_g(x,\omega_o) \end{aligned} \tag{4}$$

For a fixed view point, we can write $T_g(x,\omega_o)$ simply as $T_g(x)$. Although $T_g(x)$ is not dependent on the lighting, the summation in $T_g(x)$ is not yet computable in the preprocessing time due to the dynamically varying BRDF f. Later, f will be represented as a linear combination of basis BRDFs, which will make the summation precomputable.

As a consequence of Eq. 4, the sub-area sampling is equivalent to increasing the total number of surface samples, which clearly enhances the rendering accuracy. The sub-area samples are distributed in a small area around a gather sample, thus inheriting the radiance value and visibility from the gather sample. Consequently, sub-area sampling does not increase the overall complexity of the run-time computation, as shown in Eq. 4, whereas the gain in accuracy is significant (See Figure 2).

3.3. BRDF Decomposition

To enable BRDF editing, we first linearly parametrize a target standard BRDF model, including analytic and measured BRDF models. An analytic model, such as the Blinn-Phong model, Ward model, and Lafortune model, defines its reflectance lobe by an algebraic formula with one or two editable parameters. Meanwhile, a measured BRDF model is simply an array of reflectance values, it does not have any intrinsic parameters to control its reflectance. It is possible to define a BRDF space linearly spanned by a set of measured BRDFs, as in an earlier study [MPBM03] and change the weight values to construct a new BRDF. In this paper, the edited target is restricted to an analytic BRDF because defining editing parameters that are generally suitable for measured BRDFs remains unclear.

A BRDF model is first factorized into two terms, as

$$f(\gamma,\omega_i,\omega_o) = f_E(\gamma,\omega_i,\omega_o)f_F(\omega_i,\omega_o) \tag{5}$$

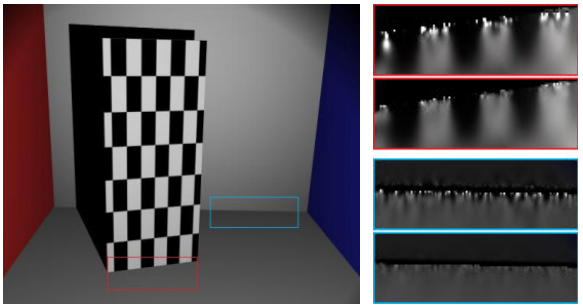

Figure 2: *Cornell box: the left image is rendered in direct lighting, and the right images are magnified views of the red and the blue box areas. Here, they only show indirect glossy reflection to demonstrate the effect of sub-area sampling. The first and the second images are rendered without and with sub-area sampling, respectively. Image artifacts appearing along the sharp edges are less noticeable in the images with the sub-area sampling (second images).*

where γ is a vector of the editing parameters; the first term $f_E(\gamma,\omega_i,\omega_o)$, determining the BRDF characteristics, is considered as an *edited term*, and the second term $f_F(\omega_i,\omega_o)$ is a scale factor known as a *fixed term* not affected by γ. The edited and fixed terms of various BRDF models are summarized in Table 4. f_F is separated from f_E to minimize the complexity of the target function to be linearized.

Suppose that the function f_E is approximated with a linear combination of k basis functions, $b_j(\omega_i,\omega_o)$,

$$f_E(x,\gamma,\omega_i,\omega_o) \simeq \mathbf{c}(x,\gamma)\cdot\mathbf{b}(\omega_i,\omega_o) \tag{6}$$

where $\mathbf{c}$ is a coefficient vector and $\mathbf{b}$ is a vector of b_j. In this case, $f_j = b_j\cdot f_F$ is regarded as a basis BRDF of f. If $T_{j,g}$ is the j-th term of T_g computed with a basis BRDF f_j, Eq. 2 is rewritten as

$$\begin{aligned} I(x,\omega_o) &= \sum_{g\in G} L_g\cdot(\mathbf{c}(x,\gamma)\cdot\mathbf{T}_g(x)) \\ &= \mathbf{c}(x,\gamma)^T\cdot\mathbf{T}(x)\cdot\mathbf{L} \end{aligned} \tag{7}$$

where $\mathbf{T}$ is a matrix of $T_{j,g}$, and $\mathbf{L}$ is a vector of L_g. As $\mathbf{T}$ is not affected by changes of lighting or BRDF, it is precomputed at every pixel and stored into a texture object. If multi-bounce reflection among the gather samples is taken into account, it can be formed into another light transfer matrix $\mathbf{M}_L$; hence

$$\mathbf{L} = \mathbf{M}_L\cdot\mathbf{L}_d \tag{8}$$

where $\mathbf{L}_d$ is a radiance vector from the gather points illuminated by direct lighting.

3.4. Wavelet Compression

To reduce the storage of the matrices and the computation cost, we use wavelet compression. All of the rows of $\mathbf{T}$ are

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

projected into the wavelet space and compressed by keeping the most significant coefficients. The projected matrices are denoted by $\mathbf{T}^w$. The radiance vector $\mathbf{L}$ is also similarly transformed to $\mathbf{L}^w$ whenever $\mathbf{L}$ is re-evaluated. As a consequence, Eq. 7 is simplified as a product of wavelet coefficient matrices, as

$$I(x,\omega_o) = \mathbf{c}(x,\gamma)^T \cdot \mathbf{T}^w(x) \cdot \mathbf{L}^w \tag{9}$$

4. PCA-based BRDF Linearization

Let the parameter vector $\gamma \in [\gamma_{min}, \gamma_{max}]$. We first separate the mean function b_0 over the range of γ in Eq. 6, as shown below:

$$f_E(\gamma,\omega_i) = b_0(\omega_i) + \sum_{i=1}^{k} c_i(\gamma) b_i(\omega_i). \tag{10}$$

ω_o is dropped from the equation because it is fixed at a specific x. The basis function $b_i(\omega_i)$ is a continuous function; it could be a well-known basis function such as a spherical harmonic function, a Fourier basis function or a polynomial basis function. However, these bases have a common drawback in that a highly specular peak is not correctly represented. For this reason, we choose to utilize a principal component analysis (PCA).

In order to apply the PCA, it was necessary to set up the problem in a discrete domain by sampling γ and ω_i. Most analytic BRDFs, as summarized in Table 4, have only one scalar parameter. Therefore, γ *is set to a scalar variable*. For ω_i, we transform it to δ, the angle between the normal N and the half vector $H(\omega_i)$ so as to avoid sampling over the hemisphere of ω_i. One exception is the Lafortune model, where the edited term does not have a clearly δ. Thus, we make it to be a function of δ by setting $\delta = \left(\frac{C_x l_x v_x + C_y l_y v_y + C_z l_z v_z}{\max(|C_x|,|C_y|,|C_z|)}\right)$ in the range $[0+\varepsilon, 1-\varepsilon]$ (We omit 0 and 1 to avoid singularity of the PCA linearization, $\varepsilon = 10^{-3}$).

Let the γ samples be evenly spaced as $\{\gamma_1, \gamma_2, ..., \gamma_{k'}\}$ in the range $[\gamma_{min}, \gamma_{max}]$, and the δ samples also be evenly spaced as $\{\delta_1, \delta_2, ..., \delta_h\}$ in its range. For each γ_i, we construct a h-dimensional column vector $\mathbf{p}_i$:

$$\mathbf{p}_i = \begin{pmatrix} f_E(\gamma_i,\delta_1) \\ \vdots \\ f_E(\gamma_i,\delta_h) \end{pmatrix} \tag{11}$$

It then becomes possible to apply PCA to the set of $\mathbf{p}_i$'s to determine the basis matrix $\mathbf{B}$ such that

$$\mathbf{p}_i = \mathbf{B} \cdot \mathbf{c}(\gamma_i) \tag{12}$$

where

$$\mathbf{b}_0 = \frac{1}{k'}\sum_{i=1}^{k'} \mathbf{p}_i, \quad \mathbf{c}(\gamma) = [1 \;\; c_1(\gamma) \;\; \cdots \;\; c_k(\gamma)]^T \tag{13}$$

$$\text{and} \quad \mathbf{B} = [\mathbf{b}_0 \;\; \mathbf{b}_1 \;\; \cdots \;\; \mathbf{b}_k] \, (k \ll k') \tag{14}$$

BRDFs	Formulation
BP	$\rho_s \frac{\gamma+2}{2\pi}\cos^\gamma\delta$
W	$\rho_s \frac{1}{\sqrt{(N.L)(N.V)}} \frac{\exp[-\tan^2\delta/\gamma^2]}{4\pi\gamma^2}$
WD	$\rho_s \frac{1}{(N.L)(N.V)} \frac{\exp[-\tan^2\delta/\gamma^2]}{4\pi\gamma^2}$
La	$\rho_s \frac{(\gamma+2)[C_x l_x v_x + C_y l_y v_y + C_z l_z v_z]^\gamma}{2\pi[\max(\lvert C_x\rvert,\lvert C_y\rvert,\lvert C_z\rvert)]^\gamma}$
CT	$\frac{\rho_s}{\pi} \frac{DG}{(N\cdot L)(N\cdot V)} F(F_0, V\cdot H)$
AS	$\frac{\gamma+1}{8\pi} \frac{(N\cdot H)^\gamma}{(V\cdot H)\max((N\cdot L),(N\cdot V))} F(F_0, V\cdot H)$

Table 3: *Analytic BRDF models and their original formulations: Blinn-Phong (BP), Ward (W), Ward-Duer (WD), Lafortune (La), Cook-Torrance (CT), Ashikhmin-Shirley (AS). N is the normal vector, V is the view vector, L is an incoming vector, R is the mirror reflection of L, H is the half vector, δ is the angle between N and H, and* $G = \min\{1, \frac{2(N\cdot H)(N\cdot V)}{(V\cdot H)}, \frac{2(N\cdot H)(N\cdot L)}{(V\cdot H)}\}$, $D = \frac{1}{\gamma^2\cos^4\delta} e^{-[(\tan\delta/\gamma)^2]}$. *In our framework, F is the Fresnel term fixed with the parameters F_0, V and H.*

Model	Fixed term f_F	Edited term f_E
BP	ρ_s	$\frac{\gamma+2}{2\pi}\cos^\gamma(\delta)$
W	$\rho_s \frac{1}{\sqrt{(N\cdot L)(N\cdot V)}}$	$\frac{\exp[-\tan^2\delta/\gamma^2]}{4\pi\gamma^2}$
WD	$\rho_s \frac{1}{(N\cdot L)(N\cdot V)}$	$\frac{\exp[-\tan^2\delta/\gamma^2]}{4\pi\gamma^2}$
La	ρ_s	$\frac{(\gamma+2)}{2\pi}\left(\frac{C_x l_x v_x + C_y l_y v_y + C_z l_z v_z}{\max(\lvert C_x\rvert,\lvert C_y\rvert,\lvert C_z\rvert)}\right)^\gamma$
CT	$\rho_s \frac{G.F(F_0,V\cdot H)}{(N\cdot L)(N\cdot V)cos^4\delta}$	$\frac{\exp[-\tan^2\delta/\gamma^2]}{\pi\gamma^2}$
AS	$\frac{F(F_0,V\cdot H)}{(V\cdot H)\max((N\cdot L),(N\cdot V))}$	$\frac{\gamma+1}{8\pi}\cos^\gamma\delta$

Table 4: *Decomposition of analytic BRDF models given in Table 3 into fixed and edited terms.*

Each column vector in $\mathbf{B}$ represents the basis function $b_j(\omega_i,\omega_o)$ in Eq. 6.

The coefficient vector $\mathbf{c}(\gamma)$ is not expressed in a closed form for γ. Hence, we compute $\mathbf{c}(\gamma)$ at run time with the least-squares method when the user specifies a new value of γ. A vector $\mathbf{p}$ is constructed for a given γ value, and $\mathbf{c}(\gamma)$ is then obtained as

$$\mathbf{c}'(\gamma) = (\mathbf{B}'^T \cdot \mathbf{B}')^{-1} \cdot \mathbf{B}'^T \cdot (\mathbf{p}_i - \mathbf{b}_0) = \mathbf{H} \cdot (\mathbf{p}_i - \mathbf{b}_0) \tag{15}$$

where $\mathbf{c}'(\gamma) = [c_1(\gamma) \cdots c_k(\gamma)]^T$ and $\mathbf{B}' = [\mathbf{b}_1 \cdots \mathbf{b}_k]$.

We surveyed standard BRDF models, as shown in Table 3, to identify their primary editing parameters. The fixed terms factored out of the BRDF models are summarized in Table 4. The Fresnel term F, taking two indices of refraction as input parameters, goes into the fixed terms because the indices are assumed to be fixed during the rendering process.

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

5. Precomputation

For a rapid computation of Eq. 9, $\mathbf{T}^w, \mathbf{D}^w$ and $\mathbf{M}_s^w$ (Section 5.2) should be precomputed before interactive rendering. The detailed procedure of the precomputation is shown below:

1. decompose a BRDF into basis vectors,
2. construct a gather sample set G and sub-area sample sets M_g's for each gather sample g,
3. compute the glossy transfer basis matrix $\mathbf{T}$, the diffuse transfer matrix $\mathbf{D}$ and the single bounce gather matrix $\mathbf{M}_s$, and
4. compress $\mathbf{T}$, $\mathbf{D}$ and $\mathbf{M}_s$ and pack $\mathbf{T}^w$, $\mathbf{D}^w$ and $\mathbf{M}_s^w$ into texture objects.

5.1. Sampling Surface Points

Gather Samples

As in [HPB06], we use a backward photon shooting method starting from the view point to sample the surfaces. A photon is shot from the view point through a pixel and is reflected at the first hit point. The reflected photon makes a second hit at a surface point which is chosen as a gather sample point. To handle the diffuse and glossy reflections effectively, we divide the total gather samples into two groups: diffuse and glossy samples. The reflection directions to find diffuse samples are uniformly sampled on the hemisphere. The reflection directions for glossy samples, ω_s, are randomly chosen by the probability density function

$$p(\omega_s) \propto (\omega_o \cdot R(\omega_s))^e \tag{16}$$

where $R(\omega_s)$ is the mirror reflection of ω_s and e is a roughness constant, usually set to 200. The ratio of glossy samples to diffuse samples is empirically chosen as $\frac{1}{6}$, as a peak reflectance lobe usually has a very narrow bell shape.

Here, the gather samples are stored in a hierarchical cluster structure to accelerate construction of the light transfer matrix.

Sub-area Samples

In the proposed rendering framework of Eq. 3, we introduced sub-area samples $q \in M_g$ in surface region A_g represented by gather sample g. For efficient sub-area sampling, we approximate surface region A_g as disk K_g centered at g with a radius derived from the area $|A_g|$. To estimate $|A_g|$, we generate uniform samples s over all of the scene surfaces such that at least one sample is assigned for each triangle of the scene. Subsequently, we classify s by the nearest gather sample g and estimate $|A_g|$ with the number of s's assigned to g.

For a gather sample g, the number of sub-area samples is determined by the maximum spherical angle of K_g viewed from a view sample. $|M_g|$ points are then uniformly sampled from K_g and their PDF is defined as $|x-g|/(|A_g||N_g \cdot \omega_i|)$, where N_g is the surface normal vector at g and x is the position of a view sample.

In the computation of Eq. 3, the actual number of sub-area samples used for a view sample x is generally less than $|M_g|$, as the summation of Eq. 3 stops when converging within a certain threshold.

5.2. Light Transfer Matrix Computation

The light transfer matrix $\mathbf{M}_L$ represents the contribution of each gather sample to the other gather samples through infinite bounces. It can be decomposed into a sum of multiples of a single bounce matrix, denoted as $\mathbf{M}_s$. From Eq. 8, the final radiance vector $\mathbf{L}$ including indirect illumination is

$$\begin{aligned} \mathbf{L} &= \mathbf{M}_L \cdot \mathbf{L}_d \\ &= (I + \mathbf{M}_s + (\mathbf{M}_s)^2 + (\mathbf{M}_s)^3 + \cdots) \cdot \mathbf{L}_d \\ &= \mathbf{L}_0 + \mathbf{L}_1 + \mathbf{L}_2 + \mathbf{L}_3 + \cdots \end{aligned}$$

where $\mathbf{L}_i = (\mathbf{M}_s)^i \cdot \mathbf{L}_d$. Projecting them into the wavelet space for a fast computation gives

$$\mathbf{L}^w = \mathbf{L}_0^w + \mathbf{L}_1^w + \mathbf{L}_2^w + \mathbf{L}_3^w + \cdots \tag{17}$$

with $\mathbf{L}_0^w = \mathbf{L}_d^w$ and $\mathbf{L}_n^w = (\mathbf{M}_s^w \cdot \mathbf{L}_{n-1}^w)^w$ for $n \geq 1$. Since $\mathbf{L}_i$ converges to zero as i increases, Eq. 17 is calculated only up to n specified by users.

Similarly to $\mathbf{T}$ and $\mathbf{D}$, $\mathbf{M}_s$ is compressed using the wavelet during precomputation time. In the construction of $\mathbf{M}_s$, we assume that all the surfaces of scene are Lambertian for simplicity. This assumption is not physically true, but supporting

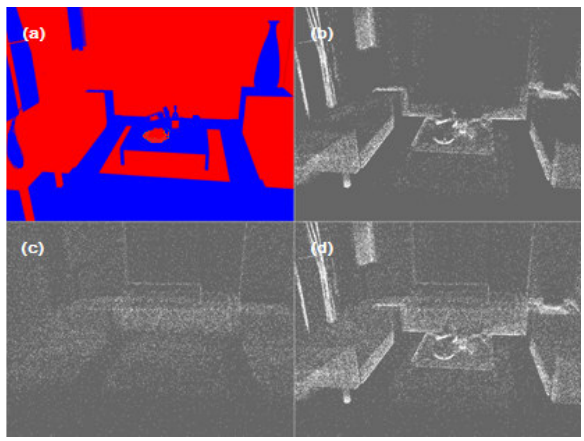

Figure 3: *An example of gather samples in an interior scene: (a) BRDF types of view samples (diffuse and glossy in red and blue, respectively), (b) glossy gather samples, (c) diffuse gather samples and (d) all of the gather samples. Note that the gather samples are distributed more densely on the surface regions next to glossy surfaces.*

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

glossy BRDFs raises the computational complexity significantly, while glossy effects quickly dissipate through subsequent bounces.

6. Implementation

6.1. Data Quantization for GPUs Computation

In Eq. 9, $\mathbf{c}(\gamma)$ and $\mathbf{T}^w(x)$ are dependent on a view sample, which corresponds to an image pixel. Supposing that an object has a single BRDF over its surface, we can use the object index pointing to $\mathbf{c}(\gamma)$ which is stored separately. Hence, each pixel contains only the object index and $\mathbf{T}^w(x)$. As $\mathbf{T}^w$ is a sparse matrix, it is stored using a the row-indexed scheme to save storage space. $\mathbf{M}_s^{\mathbf{w}}$ is also stored in a separate texture object using the row-indexed scheme. $\mathbf{D}^{\mathbf{w}}$ is packed as in [HPB06].

6.2. Interactive BRDF Editing

There are two editing modes for BRDF and light. In the BRDF editing mode, a single BRDF is edited at a time, while other BRDF's $\mathbf{c}(\gamma)^T \cdot \mathbf{T}^w(x)$ are left unchanged. Thus, we freeze these vectors to improve the rendering performance and free the memory occupied by them. As the entries of $\mathbf{T}(x)$ are highly coherent, nonzero wavelet coefficients are likely to be clustered. Thus, this freezing process can be done efficiently (See Table 5). We choose the most important wavelet coefficients of $\mathbf{c}(\gamma)^T \cdot \mathbf{T}^w(x)$ packed and loaded them into the GPU at run time. In the light editing mode, $\mathbf{c}(\gamma)^T \cdot \mathbf{T}^w(x)$ is frozen at every pixel in a similar manner, while $\mathbf{L}_d$ is computed with the user-controlled lights.

7. Results and Discussion

7.1. PCA Basis Analysis

A root mean squared (RMS) error metric is used to assess the accuracy of a linearized BRDF. Although it is not obvious to determine the threshold of the RMS for reasonable rendering quality, a lower RMS value implies a better rendering quality.

Figure 5 shows the RMS errors measured for the Blinn-Phong model. For different numbers of basis vectors, we computed the errors of the linearized function over the exact edited term. We observed that 6 basis vectors were sufficient to ensure at least 10^{-5} accuracy over the parameter range of $[3,300]$. This level of accuracy is feasible considering that the peak of the editing parameter is in the range of $[10,250]$. Figure 4 shows a comparison of the rendering result with $k = 6$ basis BRDFs and a reference image rendered by the PBRT renderer with final gathering.

7.2. Results

We tested our algorithm on a Intel Quad-core i7 CPU with 4GB of RAM and the NVidia GTX295 graphics accelerator. We ran the precomputation module in a single thread.

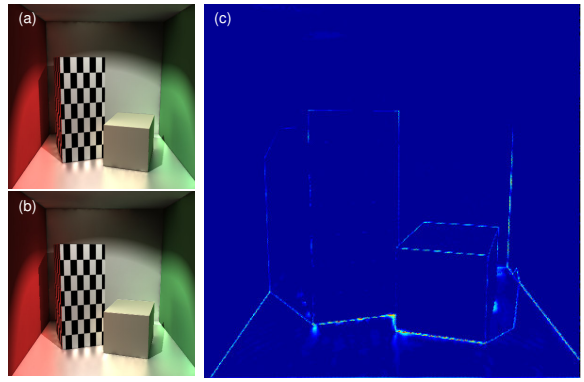

Figure 4: *Illustration of the error of our method compared to the PBRT renderer with final gathering: (a) reference (b) our method (c) the difference is computed on the original images. Subtle differences are noticeable along the object edges and shadow borders.*

We chose the Cornell box, Tableau and a still-life scene with different BRDF models to test the BRDF and light editing features of our renderer. The interior scene was chosen to show the performance of our renderer for a complex scene with a reasonable number of gather samples. As the most important concern is the accuracy of the glossy surface rendering, we ensured that the tested scenes had many glossy surfaces.

Except for Cornell box scene with 16k gather samples, we used 64k gather samples for all other test scenes. In most cases, keeping $k = 6$ basis BRDFs is sufficient to capture the important features of the BRDF model. To compute the illumination on high glossy surfaces, it was necessary to keep more wavelet coefficients of transfer matrices than diffuse surfaces. As the basis BRDFs have different factors of importance, we adaptively determined the number of coefficients proportionally to their eigenvalues: i.e., (250,150,150,100,100,100,100) for the floor in the Cornell box scene. With this setting, it is nearly impossible to load all of the precomputed data into any GPU memory; however, using the freezing technique discussed in the previous section, we were able to load all of the necessary data into the GPU. The memory requirement and the freezing time are summarized in Table 5.

8. Conclusions and Future work

We have presented a novel algorithm that supports interactive relighting and BRDF editing of a globally illuminated 3D scene. For interactive BRDF editing, a given analytic BRDF with parameter γ is discretized into a set of vectors on which the PCA is applied to identify the basis BRDFs. For each basis BRDF, the light transfer from the gather samples to the view samples is precomputed and stored in a wavelet-

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

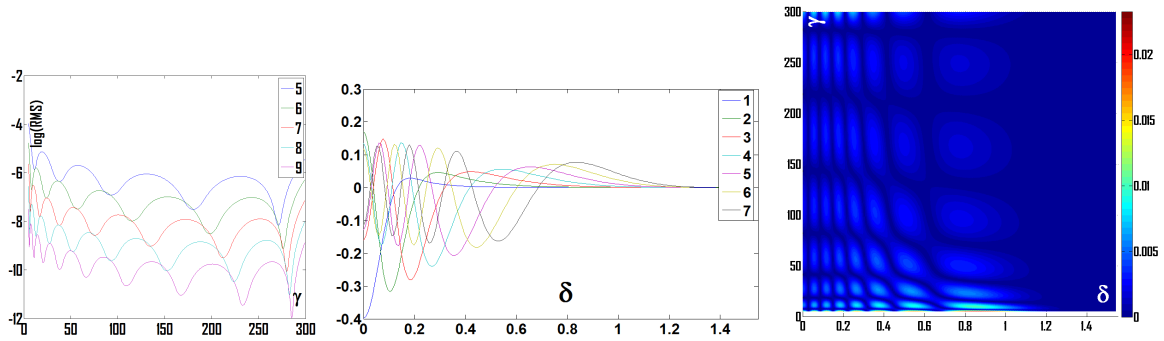

Figure 5: *Example of fitting the edited term f_E of BP with $\gamma \in [3,300]$ (a) the number of basis vectors and the RMS errors in log scale, (b) the first 7 basis vectors, (c) the RMS errors along the γ when 6 basis vectors are used.*

Scene					Precomputation			Storage (MB)				f(s)
Name	res.	objects	model	$[\gamma_{min}, \gamma_{max}]$	$\|G\|$	k	t(h)	$\mathbf{T}^w$	$\mathbf{D}^w$	$\mathbf{M}_s^w$	max	
Cornell box	400×400	floor	BP	$[3,300]$	$16k$	6	1	152 (41)	60	20	232	6
Tableau	400×400	floor	BP	$[3,300]$	$64k$	6	4	423 (109)	66	32	627	17
		bunny	W	$[0.05,2.0]$		6		197 (45)				5
		dragon	W	$[0.05,2.0]$		6		247 (61)				8
Still-life	400×400	floor	CT	$[0.1,2.0]$	$64k$	6	4	228 (93)	66	32	437	18
		ball	LF	$[3,300]$		6		158 (39)				7
		bottle	AS	$[3,300]$		6		45 (10)				4
		bowl	BP	$[3,300]$		6		163 (46)				10
		fruits	BP	$[3,300]$		6		79 (16)				5
Interior	640×480	-	-	-	$64k$	6	10	2580 (–)	168	32	707	25

Table 5: *For the scenes used in the experiments, the statistics including scene complexity, precomputation time, storage requirement for the matrices ($\mathbf{T}^w$ before (after freezing), $\mathbf{D}^w$, $\mathbf{M}_s^w$ and memory requirement (max)) and freezing time for each objects in seconds, are summarized. k and t are the number of wavelet coefficients and the construction time of transfer matrices in hours.*

compressed matrix. Given that the target BRDF is formed in a linear combination of basis BRDFs, the final value of illumination is computed as a linear combination of the incoming radiance field multiplied by the basis BRDFs.

We also introduced an auxiliary set of samples, termed sub-area samples, to enhance the accuracy of the glossy reflection. A set of sub-area samples is attached to each gather sample and is used to compute more accurate light transfer during the precomputation time.

The proposed algorithm has two limitations which remain unsolved. Solution to these limitations will be sought in a future study.

Textures. As with other point-based global illumination methods, our algorithm also has a limitation when rendering the glossy reflection of a textured surface with high frequency patterns. Increasing the number of sub-area samples may mitigate this problem, but it is not a perfect solution because small texture features are still missed by any limited number of samples.

Measured BRDFs. Currently, measured BRDFs are not yet supported, but technically can be handled with the proposed algorithm. Matusik et. al. [MPBM03] showed that 30-40 PCA basis BRDFs are sufficient to parametrize the 104 measured BRDFs with unnoticeable errors. These basis BRDFs can be directly used, but the matrix size $\mathbf{T}^w$ will easily exceed the GPU memory capacity. If we reduce the number of basis vectors for a smaller set of measured BRDFs, there are no other technical barriers to support them.

Acknowledgments

This work was supported by the IT R&D program of MKE. [No. 1002552, Development of User-motion based UX (user-experience) Platform and Stereoscopic 3D Interaction Technology for Next Generation UX Contents]. Min-Ho Kyung was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2010-0012021).

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

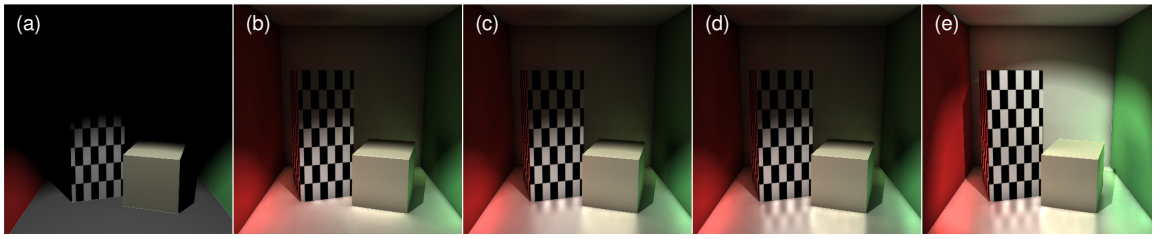

Figure 6: *Cornell box scene rendered with two-bounce diffuse reflection. From left to right: direct lighting (a), floor while increasing γ to 3 (b), 100 (c) and 300 (d). The last image (e) was rendered by modifying the light direction.*

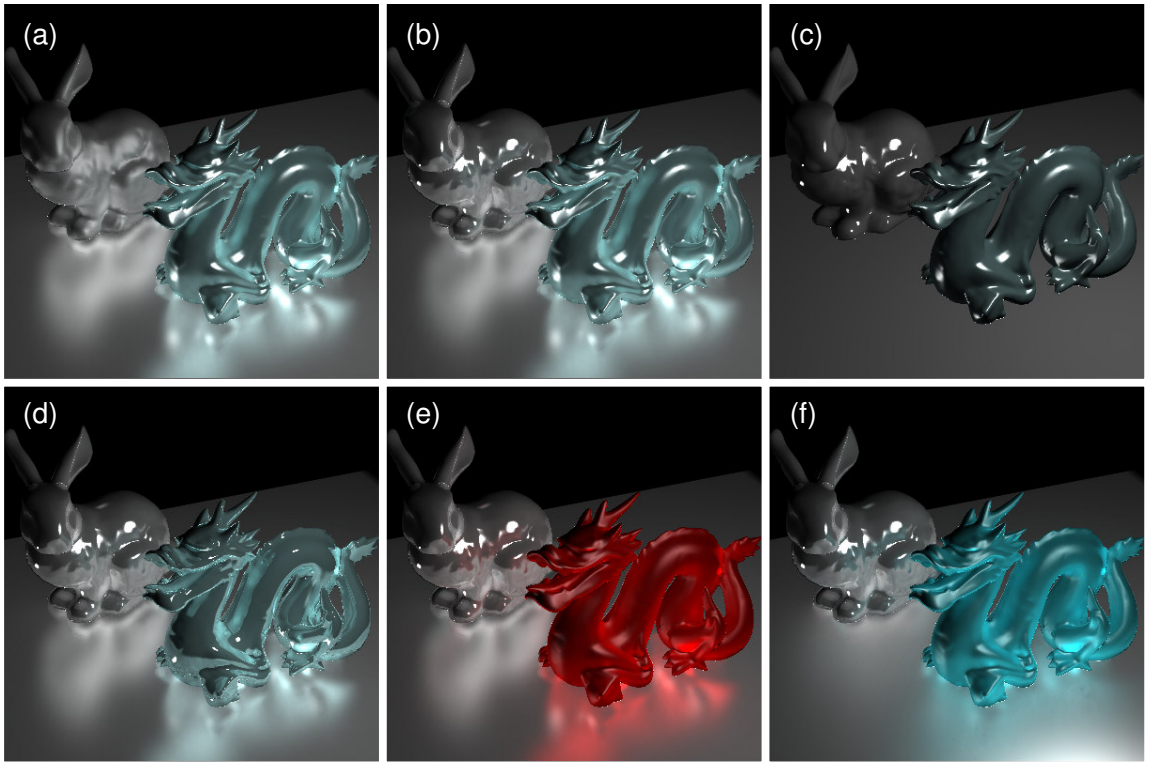

Figure 7: *Dragon and bunny scene rendered with two-bounce diffuse reflection. The top images demonstrate editing the BRDF on the bunny surface: γ =0.5 (a), 1.5 (b) and direct lighting only (c) for comparison. The two bottom images demonstrate the effect when editing the BRDF on the dragon surface: $\gamma = 0.5$ (d) and 1.5 (e). In the final image, the floor BRDF is modified to $\gamma = 5$ (f). The dragon color is also edited simultaneously at run time.*

References

[BAEDR08] BEN-ARTZI A., EGAN K., DURAND F., RAMAMOORTHI R.: A precomputed polynomial representation for interactive BRDF editing with global illumination. *ACM Trans. Graph. 27*, 2 (2008), 1–13. 2

[CPWÅP08] CHESLACK-POSTAVA E., WANG R., ÅKERLUND O., PELLACINI F.: Fast, realistic lighting and material design using nonlinear cut approximation. *ACM Trans. Graph. (Proceedings of SIGGRAPH Asia 2008) 27*, 5 (2008), 128. 2

[GKMD06] GREEN P., KAUTZ J., MATUSIK W., DURAND F.: View-dependent precomputed light transport using nonlinear Gaussian function approximations. *Proceedings of the 2006 Symposium on Interactive 3D Graphics and Games* (2006), 7–14. 2

[HKWB09] HAŠAN M., KŘIVÁNEK J., WALTER B., BALA K.: Virtual spherical lights for many-light rendering of glossy scenes. *ACM Trans. Graph. (Proceedings of SIGGRAPH Asia 2009) 28*, 5 (2009), 1–6. 2

[HPB06] HAŠAN M., PELLACINI F., BALA K.: Direct-to-indirect transfer for cinematic relighting. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2006) 25*, 3 (2006), 1089–1097. 1, 2, 3, 6, 7

[HPB07] HAŠAN M., PELLACINI F., BALA K.: Matrix row-

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

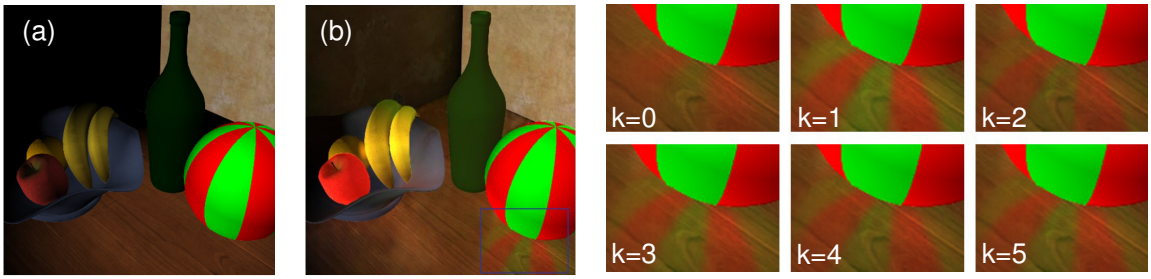

Figure 8: *Still-life scene: using direct lighting (a), setting the floor BRDF to the CT model with* $\gamma = 0.1$ *(b). The small images on the right are magnifications of the blue box with k basis BRDFs.*

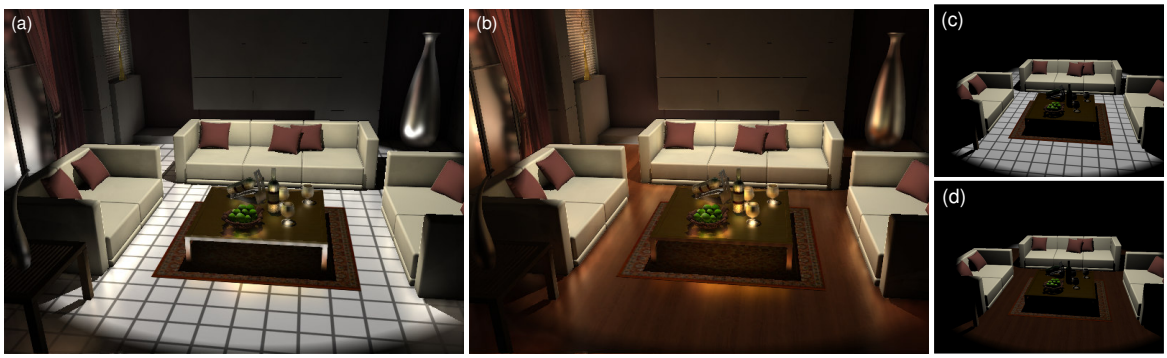

Figure 9: *Interior scene rendered with three-bounce diffuse reflection. The floor texture was changed from grid (a) to wood (b). The effect of indirect lighting is clearly visible in the scene compared to images (c) and (d) which were rendered with direct lighting only. Minor visual artifacts appear on the curtain surface to the right of the window due to insufficient gather samples on the curtain.*

column sampling for the many-light problem. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2007) 26*, 3 (2007), 26. 2

[Kel97] KELLER A.: Instant radiosity. *Proceedings of SIGGRAPH 1997* (1997), 49–56. 2

[LZT*08] LEHTINEN J., ZWICKER M., TURQUIN E., KONTKANEN J., DURAND F., SILLION F. X., AILA T.: A meshless hierarchical representation for light transport. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2008) 27*, 3 (2008), 1–9. 2

[MPBM03] MATUSIK W., PFISTER H., BRAND M., MCMILLAN L.: A data-driven reflectance model. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2003) 22*, 3 (2003), 759–769. 4, 8

[NRH04] NG R., RAMAMOORTHI R., HANRAHAN P.: Triple product wavelet integrals for all-frequency relighting. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2004) 23*, 3 (2004), 477–487. 2

[NSW09] NICHOLS G., SHOPF J., WYMAN C.: Hierarchical image-space radiosity for interactive global illumination. *Comput. Graph. Forum (Proceedings of the Eurographics Symposium on Rendering 2009) 28*, 4 (2009), 1141–1149. 2

[RGK*08] RITSCHEL T., GROSCH T., KIM M. H., SEIDEL H.-P., DACHSBACHER C., KAUTZ J.: Imperfect shadow maps for efficient computation of indirect illumination. *ACM Trans. Graph. (Proceedings of SIGGRAPH Asia 2008) 27*, 5 (2008), 129. 2

[SKS02] SLOAN P.-P., KAUTZ J., SNYDER J.: Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2002) 21*, 3 (2002), 527–536. 1

[SZC*07] SUN X., ZHOU K., CHEN Y., LIN S., SHI J., GUO B.: Interactive relighting with dynamic BRDFs. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2007) 26*, 3 (2007), 27. 2

[TS06] TSAI Y.-T., SHIH Z.-C.: All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2006) 25*, 3 (2006), 967–976. 2

[WFA*05] WALTER B., FERNANDEZ S., ARBREE A., BALA K., DONIKIAN M., GREENBERG D. P.: Lightcuts: a scalable approach to illumination. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2005) 24*, 3 (2005), 1098–1107. 2

[WRG*09] WANG J., REN P., GONG M., SNYDER J., GUO B.: All-frequency rendering of dynamic, spatially-varying reflectance. *ACM Trans. Graph. (Proceedings of SIGGRAPH Asia 2009) 28*, 5 (2009), 1–10. 2

[WTL04] WANG R., TRAN J., LUEBKE D. P.: All-frequency relighting of non-diffuse objects using separable BRDF approximation. In *Rendering Techniques (Proceedings of the Eurographics Symposium on Rendering 2004)* (2004), pp. 345–354. 2

[WWZ*09] WANG R., WANG R., ZHOU K., PAN M., BAO H.: An efficient GPU-based approach for interactive global illumination. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2009) 28*, 3 (2009), 1–8. 2

© 2010 ETRI
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.