Bidirectional Reflection Functions from Surface Bump Maps

BRIAN CABRAL, NELSON MAX, and REBECCA SPRINGMEYER
UC Davis/Lawrence Livermore National Laboratory

In this paper, we approximate the bumpy surface by a collection of triangular facets, interpolating between the bump heights at a discrete array of points. The rays which hit a facet without being shadowed on the way in, and leave the surface directly without hitting another facet on the way out, define a parallel beam emerging from the non-obscured part of the facet. This reflected beam contributes a "delta function" to the bidirectional reflection, weighted by its cross-sectional area.

The bidirectional reflection function for the faceted approximation is the sum of these weighted delta functions, and is thus not continuous. We present two methods for capturing the information in the reflected beams. The first method divides the hemisphere of reflected directions into a number of bins, and assigns each reflected beam to one of the bins. This leads to a tabulated version of the bidirectional reflection function. The second method uses spherical harmonics to approximate the density per solid angle of the weighted delta functions. The result is a series of coefficients defining a continuous bidirectional reflection function as a linear combination of basis functions, effectively condensing the table into less data, and also smoothing it.

Section 1 gives several definitions from radiometry, leading up to a physical definition of the bidirectional reflection function. Section 2 summarizes previous models for rough surfaces and methods for finding their reflection properties. Section 3 summarizes previous work on using bidirectional reflection functions to calculate the effects of environmental illumination. Section 4 describes horizon mapping, an efficient way for finding the non-obscured part of each facet. Section 5 shows how to assign the beams into bins corresponding to a collection of tabulated reflection directions, and section 6 argues that a bidirectional reflection function tabulated in this way should approximately satisfy Helmholtz's Law of Reciprocity. Section 7 shows how a spherical harmonic expansion of the bidirectional reflection function can be used to efficiently compute the effects of environmental illumination, and section 8 shows how to determine the coefficients in this expansion from the collection of beams reflected by the facets. Finally, section 9 explains the construction of the bump tables, and presents images rendered with the resulting reflection functions.

Our methods generate anisotropic reflection functions. However the anisotropy can be averaged out if desired for isotropic surfaces. For simplicity we consider only isotropic reflection functions in sections 7 and 8.

1. Radiometry

There are two methods of quantifying light measurements: 1) by radiant energy, which measures physical quantities, and 2) by luminous energy, which measures the response of the eye. In this paper we discuss only radiant energy, integrated over all wavelengths. However, the quantities defined below can also be considered as spectral quantities, which vary with wavelength. They can then be converted to luminous quantities by multiplying by the CIE luminous efficiency function and integrating over all wavelengths. They can similarly be multiplied by...
the CIE color matching functions, and integrated to get CIE x, y, and z color coordinates, which can be converted to the appropriate output colors for a specific device (See Hall [15] and Cook and Torrance [8].) To measure solid angle, we use a spherical coordinate system with the CIE color matching functions, and integrated to get CIE x, y, and z color coordinates, which can be converted to the appropriate output colors for a specific device (See Hall [15] and Cook and Torrance [8].)

The radiant flux, $\Phi$, is the time rate of flow of energy traveling through a surface in the form of electromagnetic waves. It is usually measured in watts.

The radiant flux density, $D = \Phi / dA$, at an element of a surface is the ratio of the radiant flux through that surface element, to the area of the element. It can be measured in watts per square meter. When referring to radiant flux incident on a surface, the density is called irradiance.

The radiance, $L = \Phi / (dA \cos \theta d\Omega)$, in a direction $(\theta, \phi)$, at a point of a surface is the following ratio. The numerator is the radiant flux leaving, passing through, or arriving at an element of the surface surrounding the point, and propagated in directions within a cone containing the given direction. The denominator is the product of the solid angle of the cone and the area of the orthogonal projection of the element of the surface on a plane perpendicular to the given direction. Radiance can be measured in watts per square meter per steradian.

Consider a beam of incident light, with radiance $L_{in}$, coming from a solid angle $d\Omega_{in}$ about an incident direction $(\theta_{in}, \phi_{in})$ and reflecting from a surface. Let $d\Omega_{ef}$ be the radiance reflected in the direction $(\theta_{ef}, \phi_{ef})$. Then the bidirectional reflection function $f_{s}$ is the ratio:

$$f_{s}(\theta_{in}, \phi_{in}, \theta_{ef}, \phi_{ef}) = \frac{d\Omega_{ef}}{L_{in} \cos \theta_{ef} d\Omega_{in}}$$

(See Snel [28].) In the limit when $d\Omega_{in}$ approaches zero and the incident beam becomes parallel, the denominator becomes the irradiance onto the surface from the single direction $(\theta_{in}, \phi_{in})$. Since $f_{s}$ is a radiant divided by a radiant flux density, the units of $f_{s}$ are inverse steradians.

For a radiation function $L_{in}(\theta_{in}, \phi_{in})$ which varies continuously with the incident direction, we can get the total reflected radiance in the direction $(\theta_{ef}, \phi_{ef})$ by integrating $d\Omega_{ef}$ over the hemisphere $H$ of possible incident directions above a surface. Thus

$$L_{ef} = \int_{H} f_{s}(\theta_{in}, \phi_{in}, \theta_{ef}, \phi_{ef}) L_{in} \cos \theta_{in} d\Omega_{in}. \quad (1)$$

2. Bidirectional Reflection Models

Torrance and Sparrow [32] developed a geometric lighting model for specular reflection, which was applied to computer graphics by Blinn [1], and modified by Cook and Torrance [8], to take better account of colored light. It has been used to create spectacularly realistic shaded renderings of surfaces such as brushed metal.

The Torrance-Sparrow model assumes that an infinitely long, symmetrical, wedge-shaped groove has been cut into the surface. The normal $N$ to one of the two flat sides of this groove is used, together with the lighting direction and the viewing direction, to compute the fraction of the incoming light which is neither shadowed on the way in nor obscured on the way out by the opposite side of the groove. Random grooves, with some specific distribution of normals $N'$ about the surface normal $N$, generate a distribution of reflected rays from each incoming direction. One can thus compute a bidirectional reflection coefficient. However on any real surface the grooves will interfere with each other, so they cannot be both randomly oriented and infinitely long.

A real surface can be more explicitly modeled by a bump table, which gives the surface height at a two dimensional array of sample points. Blinn [2] has shown how to perturb the surface normal for use in a lighting model which takes account of the bumps. The method has become known as "bump mapping." It gives realistic light reflections for a level of detail at which the bumps are visible, but not needed to be modeled in full 3-D perspective. Max [21] has shown how to account for the shadows from the bumps, by using a table of horizon angles. This might appropriately be called "horizon mapping." In the present paper, we divide the surface defined by the bump table into triangular facets. For each facet and each tabulated lighting direction, we compute the reflection direction and use the horizon table to find the reflected beam, that is, the portion of the beam potentially intercepted by the facet, which is neither shadowed on the way in nor obscured on the way out.

We accumulate this reflected flux into a tabulated bidirectional reflection function, using the tabulated reflection direction closest to the actual reflected beam. Because it is derived from the bump table, the resulting lighting model will be consistent with the area average of the light from an image rendered with bump mapping and horizon mapping. This should permit smooth transitions of detail between these two methods of representing surface roughness. This is the chief justification of our method: consistency with explicitly modeled surface structure, including anisotropic reflections. Kajiya [20] has proposed smooth transitions between three roughness representations: lighting models, bump mapping, and full 3-D visible-surface calculations which models the bumps in the data base. However this method uses the wave theory of light, and is not computationally practical. If bump mapped surfaces are rendered with horizon mapping, they should in addition allow smooth transitions to full 3-D renderings including cast shadows.

Perlin [24] has also proposed a smooth transition of surface roughness components, between lighting models and normal perturbations. However his normal perturbations do not come from an explicitly specified bump table, and therefore do not permit the use of horizon mapping, or full 3-D modeling.
Ohira [23], Takagi [30], and Kajiya [20] have discussed bidirectional reflection functions which are anisotropic. These depend on all four parameters needed to define the lighting and viewing directions, with respect to a frame specifying the orientation of two surface tangents and the surface normal. Our calculations initially generate anisotropic reflections, since the two tangent vectors are needed to orient the bump tables on the surface. If isotropic reflection functions are desired, the anisotropic tables can be averaged over the possible orientations of the bump table on the surface, so that they depend on only three parameters. This will make more reflected rays contribute to each table entry, and improve the sampling statistics.

3. Environmental Illumination

Early illumination models handled only point light sources, but there have been many attempts to include the light coming from the whole environment, based on equation (1) above. Whitted and Cook [35] analyze various specific lighting models in terms of restrictions on the full generality of equation (1). For example, Blinn and Newell [5] created mirror reflections of a room in a teapot by assuming that $L_n$ is independent of position, and that $f_s$ is a delta function. If $L_n$ instead is a delta function, we have the case of a point light source. Lambert's law for diffuse reflection results when $f_s$ is constant. Phong [26], Blinn [1], and Cook and Torrance [8] have generated glossy highlights from point sources by using more general specular reflection functions $f_s$, concentrated near the mirror direction. Cook, Carpenter, and Porter [7] handle distributed ray tracing to model glossy reflections, with such a concentrated $f_s$ and an arbitrary $L_m$. Because the integral is estimated by tracing sample rays near the mirror reflection, $L_m$ can vary with position and multiple reflections can be rendered.

Kajiya [19] has demonstrated that by long calculation and a little pruning of the ray tree, it is possible to include diffuse as well as specular reflections in this scheme, allowing a completely general $f_s$, Immel, Cohen, and Greengberg [18] have also solved the same interreflection problem for general $f_s$, by dividing the surface into polygons, and solving a huge system of linear equations.

In this paper we apply our bidirectional reflection distribution function to the case that $L_m$ is independent of position, as if the environment were painted on a sphere at infinity. In this case $L_m(\theta_{in}, \phi_{in})$ can be sampled into a texture table. Greene [13] has called techniques based on such a table “environment mapping”, and surveyed methods for obtaining the integral for $L_{mef}$ efficiently from the table. For example, if $f_s$ is constant (diffuse reflection) $L_{mef}$ depends only on the normal $N$, and Miller and Hoffman [22] suggest that it be precomputed and tabulated. In the Phong reflection model [26], $f_s$ depends only on the mirror reflection $M = (\theta_{mirror}, \phi_{mirror})$ of the viewing direction $V = (\theta_{v}, \phi_{v})$, so $L_{mef}$ can again be precomputed and tabulated as a function of $(\theta_{mirror}, \phi_{mirror})$. (See [22].)

Consider a somewhat more general $f_s$, which, as a function of $(\theta_{in}, \phi_{in})$ for fixed $V$ and fixed normal $N$, has elliptical symmetry about some “off specular” peak direction $P = (\theta_{peak}, \phi_{peak})$ near $M$. (See [1] or [22].) That is, contours of constant $f_s$ correspond to small concentric ellipses about $P$ on the sphere of $(\theta_{in}, \phi_{in})$ directions. (See figure 1.) The shape of $f_s$ is determined by the eccentricity, which is the same for all these ellipses, and by a curve on a cross section plane through $N$ and $P$, specifying $f_s$ as a function of the ellipse size. The shape of $f_s$ usually varies with $\theta_{in}$, and the orientation of the ellipses varies with $N$, so the integrals cannot be precomputed as a function of $(\theta_{peak}, \phi_{peak})$ alone.

The elliptical weighted average filter of Greene and Heckbert [14] can take advantage of the elliptical symmetry of $f_s$ to estimate the integral for $L_{mef}$ from the tabulated values of $L_m(\theta_{in}, \phi_{in})$. Its computation time is proportional to the number of entries in the table which contribute to the integral, so this method is most efficient for shiny surfaces, with narrow specular peaks. Greene [13] has suggested using Williams' pyramidal parametrics [36] together with elliptical weighted averages, so that the number of entries in the selected table is bounded, independent of the width of specular peak. Glassner [12] has shown how to use summed area tables (See Crow [10]) to approximate the integral of $L_m$ over larger ellipses. But his method requires that $f_s$ be constant inside the ellipse, and zero outside.

Perlin [25] has generalized summed area tables to higher order, and shown how to approximate an elliptical gaussian by piecewise quadratic polynomials, whose convolutions with $L_m$ can be obtained using the second order tables, in time independent of the width of the gaussian. (See Heckbert [16] for a more detailed explanation.) The ellipses must be oriented with their major and minor axes horizontal or vertical in the texture table. This can be arranged in the following case. Suppose the view of the reflecting object is from far away, so that the viewing direction $V$ can be assumed to be constant, and that $L_m$ is tabulated in spherical coordinates, with $V$ as the north pole. An isotropic bidirectional function will have mirror symmetry with respect to a plane through $V$ and $N$, so its contour ellipses must also. This guarantees the required orientation, since a plane through the pole is a line of constant azimuthal angle in the texture table.

Greene [14] also considers the problem of antialiasing reflections in a curved, perfectly shiny mirror, on account for the fact that the ray from the eye reflects in directions which vary over the area of the pixel. He proposes elliptical weighted average filtering, using values of $L_m$ tabulated on the six faces of an environment cube. A gaussian weighting about the pixel center, for slowly varying mirrors, will transform to an approximate elliptical gaussian in the environment map, in spherical coordinates as well as on an environment cube. To generalize this to the case of a glossy curved mirror, the appropriate filter is the convolution of the weighting function with the specular bidirectional reflection function $f_s$. If $f_s$ can be approximated by an elliptical gaussian, the convolution of these two functions is again an elliptical gaussian, which may no longer be oriented consistently with the spherical axes. The method of Perlin [24] no longer applies, but the method of Greene and Heckbert [13,14] can still be used. Miller and Hoffman [22] also include antialiasing in their technique. Of course, in distributed ray tracing [27], the rays can be distributed simultaneously over both the subpixel positions and the glossy reflection directions.

The method of integration we propose in section 7 below works under restricting assumptions similar to those above: a) $L_m$ is independent of position, b) $V$ is constant, and c) $f_s$ is isotropic. However, we can handle arbitrarily shaped $f_s$, which can combine specular and diffuse reflection. We expand both $\cos \theta_{ref}$ and $L_m$ in spherical harmonics, with $V$ as the north pole. The integral for $L_{mef}$ can then be computed efficiently as the dot product of the coefficient vectors for $\cos \theta_{ref}$ and $L_m$. Because the computation time is proportional to the number of coefficients, our method works best for rougher surfaces, where $f_s$ is
wide and can be approximated by fewer terms. (This is in contrast to the methods of Greene and Heckbert [14] and Immel, Cohen, and Greenberg [18], which work better for narrow $f$.)

4. Horizon Mapping

Horizon tables were originally developed to create shadows on bump-mapped surfaces [21]. Imagine a bump height function, tabulated on an $m \times m$ grid, representing the altitude of a mountainous terrain, with $z = \text{east}$, $y = \text{north}$, and $x = \text{up}$. Suppose the sun sets to the west in a vertical plane, as it would at the equator on an equinox. Then a point on the terrain is in shadow whenever the sun is below the horizon, that is, whenever the angle $\theta$ from the $Z$ axis to the sun is greater than the angle $\beta$ from the $Z$ axis to the western horizon. The angle to the western horizon can be computed at each tabulated position, by looking at the slopes of lines from that position to other tabulated positions to its west, and finding the maximum slope. The bump map is doubly periodic in $z$ and $y$, so that the slope can be determined by the $m - 1$ points to the west, even for a starting point near the western edge of the bump map. The angle of the horizon can be determined similarly for any sunset direction, that is, for any half plane through the $Z$ axis. Max [21] proposed using only the eight principal compass directions, for which the slope to the horizon can be easily determined by interpolation. Mountains which are missed by the sampled compass directions from a data point will fail to cast shadows on the point, so more compass directions will give more accurate results.

In this work, we have used 24 directions. Let $PQ$ represent a ray from the data point $P$ in one of these directions, shown intersecting the triangles in Figure 2. The bump heights at an intersection point with a triangle side are determined by interpolation from the tabulated vertices at endpoints of the side. Note that the interpolation factors, and the positions of the two endpoints relative to the start of the ray, depend only on the ray’s direction. To speed up the calculation of the horizon table, this information is computed ahead of time for each ray direction [5].

Max [21] shows how to use the horizon table computed for a flat surface, to find the bump shadows when the flat surface is parametrized onto a curved surface patch. Here we will use it to estimate energy reflected from the bumpy surface.

5. Reflection From Facets

Consider a collimated parallel beam of light coming to the surface $S$ from the direction $W_{in} = (\theta_{in}, \phi_{in})$, with radiant flux density $D_{in}$, measured per unit area normal to $W_{in}$. This is the limit for a source of radiant $L_{in}$ and solid angle $d\omega_{in}$, as the source moves away to infinity, and $d\omega_{in}$ approaches zero, while the radiant increases to keep $L_{in}d\omega_{in}$ equal to $D_{in}$. Since $W_{in}$ is tilted away from the normal $N$ by an angle $\theta_{in}$, the irradiance incident on $S$ is $D_{in} \cos \theta_{in} = D_{in} (W_{in} \cdot N)$.

We divide $S$ into $2m^2$ triangular facets, as shown projected in Figure 2. If $U_1$ and $V_1$ represent the vectors along the edges of facet $S_i$, which project to the horizontal and vertical edges in Figure 2, then the area $A_i$ of $S_i$ is $|U_1 \times V_1|/2$, and the normal $N_i$ to $S_i$ is $(U_1 \times V_1)/(|U_1 \times V_1|)$. Given $N_i$ and $W_{in}$, we can compute the reflected direction $R_i = 2(W_{in} \cdot N_i)N_i - W_{in}$ (See Whitted [34]).

The obscuring effects of other facets can be estimated from the horizon table. We can determine from the entries for the compass direction $\theta_{in}$, the three horizon angles $\beta_1, \beta_2$, and $\beta_3$ for the three vertices of $S_i$. If the samples for $\theta_{in}$ for the tabulated bidirectional reflection function correspond to those in the horizon table, no interpolation is required. When $\beta_1, \beta_2$, and $\beta_3$ are all greater than $\theta_{in}$, the facet $S_i$ is completely lit, and when all are less than $\theta_{in}$, the whole facet is in shadow. When some are greater and some are less, the partial shadowing is determined as follows. Along the two sides where $\beta_j - \theta_{in}$ changes sign, we interpolate to estimate two points where $\beta = \theta_{in}$. The line joining these two points separates the triangle into a quadrilateral and a smaller triangle, one of which is illuminated and the other, shadowed.

Shadowed light contributes its flux instead to the specular reflection from the facet which intercepted it. But, as in the Torrance-Sparrow model, reflected light which is obscured on the way out is lost from the specular energy computation, and assumed to contribute instead to diffuse reflection or absorption. The illuminated portion of the facet sends out a reflected beam in the direction $R_i = (\theta_{ref}, \phi_{ref})$. The portion of this beam which is not obscured can be found similarly, using the convention of Duff [11]. Figure 1, to treat the case where $\theta - \theta_{ref}$ changes sign along all four sides of a quadrilateral. Since $\phi_{ref}$ can be an arbitrary azimuthal angle, interpolation in the horizon table is now required, which introduces another approximation in our calculations.

Let $B_r$ be the area of the facet $S_i$ which reflects the non-obscured, non-shadowed portion of the beam. This area projects to an area $B_i(N_i, W_{in})$ normal to the incident beam, and therefore intercepts a flux of $D_{in} B_i(N_i, W_{in})$. We assume the facet is perfectly smooth, so that the reflected flux is this incident flux multiplied by the Fresnel factor $F(N_i, W_{in})$. (See Born and Wolf [4].)

Now take a collection $\{V_k|k = 1, n\}$ of sampled reflection directions, and let $V_{nearest}(V)$ be the index $k$ of the $V_k$ nearest to the unit vector $V$. (Choose the lower index in case of a tie.) Then $C_k = \{V \in D|V_{nearest}(V) = k\}$ represents the bin on the hemisphere $H$ assigned to $V_k$. Let $d\omega_k$ be the solid angle measure of $C_k$.

To accumulate the reflected flux into the appropriate bins, we assign the flux from facet $S_i$ to bin $k = V_{nearest}(R_i)$. When all $2m^2$ facets have been considered the total flux $\Phi_k(\theta_{in}, \phi_{in})$ in bin $k$ is

$$\Phi_k(\theta_{in}, \phi_{in}) = \sum_{i=1}^{2m^2} d(k - V_{nearest}(R_i))F(N_i, W_{in})D_{in} B_i(N_i, W_{in} \cdot N_i)(2).$$

We find the reflected radiance $L_{ref}$. In the direction $V_k$ for bin $k$, by dividing the flux $\Phi_k$ by the area $dA$ of $S$ normal to the outgoing direction $V_k$, and by the solid angle $d\omega_k$ of the bin. The area $dA$ of $S$ is taken as the area of the flat lattice, shown in Figure 2, on which the bumps are based. Then $dA = dA(V_k \cdot N)$. Thus
By definition, the bidirectional reflection function \( f_r \) is this reflected radianse, divided by the incopming irradince. Since the incoming beam has a radiant flux density \( D_{in} \) measured normal to its direction \( W_m \), its irradiance on the surface \( S \) is \( D_{in} \cos \theta_i = D_{in}(W_m \cdot N) \), so

\[
f_r(\theta_i, \phi_i; \theta_r, \phi_r) = \frac{L_{ref,k}}{D_{in}(W_m \cdot N)}
\]

Putting the above three equations together, we get

\[
f_r(\theta_i, \phi_i; \theta_r, \phi_r) = \frac{\sum_{k=1}^{2n^2} \mathbf{D}_{in}(k - \text{Nearest}(R_i))f_r(N_i, W_m)B_i(W_m \cdot N_i)}{D_{in}(W_m \cdot N)\sum_{k=1}^{2n^2} \mathbf{D}_{in}(k - \text{Nearest}(R_i))G_i}
\]

Note that \( D_{in} \) can be cancelled from both the numerator and denominator, so that \( G_i = F(N_i, W_m)B_i(W_m \cdot N_i) \) / \( D_{in}(W_m \cdot N) \) represents the fraction of the incoming flux reflected by facet \( S_i \), and

\[
f_r(\theta_i, \phi_i; \theta_r, \phi_r) = \sum_{k=1}^{2n^2} \frac{\mathbf{D}_{in}(k - \text{Nearest}(R_i))G_i}{\sum_{k=1}^{2n^2} \mathbf{D}_{in}(k - \text{Nearest}(R_i))}
\]

If this computation is repeated for each of a finite collection of incident directions \( W_m \), we can build up a bidirectional reflection table \( f_r(\theta_i, \phi_i; \theta_r, \phi_r) = f_r(W_m, V_k) \).

Torrance and Sparrow also discuss a continuous distribution of hypotheetical facets, all of the same projected area, with a density \( \delta(N') \) per steradian having facet normal \( N' \). They apply the formula \( \sum_{k=1}^{2n^2} \mathbf{D}_{in}(k - \text{Nearest}(R_i))f_r(N_i, W_m)B_i(W_m \cdot N_i) / D_{in}(W_m \cdot N) \) represents the fraction of the incoming flux reflected by facet \( S_i \), and

\[
f_r(\theta_i, \phi_i; \theta_r, \phi_r) = \sum_{k=1}^{2n^2} \frac{\mathbf{D}_{in}(k - \text{Nearest}(R_i))G_i}{\sum_{k=1}^{2n^2} \mathbf{D}_{in}(k - \text{Nearest}(R_i))}
\]

We can speed up the calculation of the integral in equation (1) by approximating the factor \( \cos \theta_i f_r(W, V) \) with spherical harmonics. We reorganize the spherical coordinates so that the pole is in the viewing direction \( V \). The isotropic bidirectional reflection function \( f_r(W, V) \) really depends on the three vectors \( W, V, \) and \( N \). For non-isotropic reflection functions, tangent vectors are also required. In the previous section, \( N \) was assumed fixed, but now \( V \) is fixed. So we change notation, and let \( f_r(\theta, \phi) \) denote the bidirectional reflection function of the illumination direction \( W = (\theta, \phi) \) for \( V \) permanently fixed, and \( N \) temporarily fixed and indicated in the subscript. We have taken \( V \) along the \( Z \) axis, and will at first assume that \( N \) lies in the \( XZ \) plane, with spherical coordinates \( (\alpha, \phi) \). Since \( f_r \) is isotropic, \( f_r(\theta, \phi) \) is symmetric with respect to the \( XZ \) plane, and \( f_r(\alpha, \phi) = f_r(\alpha, \phi) \). As a result, \( \cos \theta_i f_r(W, V) \) can be expanded in spherical harmonics with terms in \( \cos \theta_i \) but none in \( \sin \theta_i \), that is, there are no \( a_{\ell m} \) terms for \( n < 0 \). This expansion is defined over the whole unit sphere \( Q \), instead of just the hemisphere \( H \), so we replace \( \cos \theta_i \) by \( \cos \theta_i(\cos \theta_i, 0) \). Thus the expansion of \( \cos \theta_i f_r(W, V) \) is

\[
\cos \theta_i f_r(W, V) = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} a_{\ell m}(\alpha)Y_{\ell m}(\theta, \phi).
\]
The expansion is an infinite series, but if we replace the infinite upper bound in the first summation by a finite upper bound \( M \), we get an approximation to \( \tilde{L}(\theta, \phi) \). We will calculate the coefficients \( a_{n}(\alpha_{j}) \) for a finite sample \( N_{j} \) of normals in the \( XZ \) plane, with spherical coordinates \((\alpha_{j}, 0)\), and find \( a_{n}(\alpha) \) for other normals \( N = (\alpha, 0) \) by interpolation.

Now let \( N \) be an unrestricted normal, with spherical coordinates \((\alpha, \beta)\). A rotation of \(-\phi\) about the \( Z \) axis takes \((\alpha, \beta)\) to \( N_{0} = (\alpha, 0)\), and \( W = (\theta, \phi - \beta) \). Since the reflection function is isotropic, and only depends on the relative angles between \( V, N, \) and \( W \), \( f_{s}(\theta, \phi) = f_{s}(\theta, \phi - \beta) \). Also, \( \tilde{L}_{s}(\theta, \phi) \) depends only on the angle between \( W \) and \( N \). Therefore

\[
\tilde{L}_{s}(\theta, \phi, \gamma) = \sum_{n=0}^{M} \sum_{m=-n}^{n} a_{n}(\alpha) M_{n} P_{n}^{m}(\cos \theta) \cos(n(\phi - \beta)).
\]

But

\[
\cos n(\phi - \beta) = \cos(n\phi - n\beta) = \cos n\phi \cos n\beta + \sin n\phi \sin n\beta.
\]

Therefore

\[
\tilde{L}_{s}(\theta, \phi, \gamma) = \sum_{n=0}^{M} \sum_{m=-n}^{n} a_{n}(\alpha) \cos n\phi \sin n\beta,
\]

where

\[
a_{n}(\alpha) = \left\{ \begin{array}{ll}
a_{n}(\alpha) \cos n\beta, & n \geq 0; \\
a_{n}(\alpha) \sin n\beta, & n < 0.
\end{array} \right.
\]

We now have terms in \( \sin \phi \beta \), so there are \( a_{n} \) for \( n < 0 \). The computation of the \( a_{n}(\alpha) \) is explained in section 4.

The radiance of the environment, \( L_{e}(\theta, \phi, \gamma) \), may also be expanded in real spherical harmonics:

\[
L_{e}(\theta, \phi) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} b_{n} Y_{n}^{m}(\theta, \phi).
\]

Since the \( Y_{n} \) are an orthonormal basis, the coefficients \( b_{n} \) are computed by integrating the product of the illumination function and the spherical harmonics over the unit sphere:

\[
b_{n} = \int_{Q} b_{n}(\theta, \phi) Y_{n}(\theta, \phi) d\omega.
\]

Having expanded both the reflection function and the environmental radiance function in spherical harmonics, we can now estimate the integral of equation (1):

\[
L_{e}(\theta, \phi) = \int_{Q} \cos \theta_{1} f_{s}(W, V) L_{e}(\theta, \phi) d\omega
= \int_{Q} \tilde{L}_{s}(\theta, \phi) d\omega
= \int_{Q} \sum_{n=0}^{M} \sum_{m=-n}^{n} a_{n}(\alpha) M_{n} P_{n}^{m}(\cos \theta) \cos(n(\phi - \beta)) d\omega
\]

Thus \( L_{e}(\theta, \phi) \) becomes the inner product of these two coefficient vectors. If \( L_{e} \) has high spatial frequencies, then \( b_{n} \) does not, and then \( f_{s} \) acts like a filter in the integral for \( L_{e} \), averaging out the high frequencies in \( L_{e} \). Thus when the terms up to \( l = M \) are sufficient to represent \( \tilde{L}_{s}(\theta, \phi, \gamma) \), the higher terms in \( L_{e} \) are eliminated from the integral by the orthogonality of the \( Y_{n} \).

For a curved or bump-mapped surface, the bidirectional reflection function should be further widened and frequency limited, to account for the variation of surface normals within a pixel, as mentioned in the introduction above. An inexpensive way to do this is to limit the range of \( l \) indices in the expansion for \( \tilde{L}_{s}(\theta, \phi, \gamma) \) to some maximum \( M' \) less than \( M \). The values of \( a_{n} \) can be gradually decreased for large \( l \), so that the expansion varies smoothly with the range of normals within a pixel. Decreasing the weight for terms with large \( l \) also helps eliminate the "ringing", or Gibbs phenomenon, at sharp edges or in \( L_{e}(\theta, \phi) \).

The \( M \)-term approximation for \( \tilde{L}_{s}(\theta, \phi, \gamma) \) cannot be exactly zero in the hemisphere \( Q - H \) behind the surface, where \( \tilde{L}_{s} \) is zero. This can cause slight reflections of objects behind the surface, but such artifacts do not appear disturbing in the images shown here.

If \( L_{e}(\theta, \phi) \) is represented as a cube whose six faces are texture maps, as in Greene [13], the integral in equation (5) can be estimated by summation over the texture pixels. This integral must be computed only once, for the first viewing direction \( V \). For each new viewing direction the set of coefficients \( b_{n} \) may be transformed by taking a linear combination of the original coefficients:

\[
b_{\tilde{n}} = \sum_{n=-\tilde{n}}^{\tilde{n}} b_{n} D_{\tilde{n} n}(\alpha, \beta, \gamma)
\]

where the matrices \( D_{\tilde{n} n} \) describe how the spherical harmonics transform under rotation through the Eulerian angles \( \alpha, \beta, \) and \( \gamma \). (See Tinkham [31].) Note that each \( D \) matrix will be of size \( 2\tilde{n} + 1 \), with the largest matrix requiring \( (2M + 1)^{2} \) elements. The use of these rotation matrices is clearly more efficient than recomputing the coefficients for each new viewing direction.

The expansion

\[
L_{\text{ref}} \approx \sum_{l=0}^{M} \sum_{n=-l}^{l} a_{n}(\alpha) b_{n}
\]

has \( (M + 1)^{2} \) terms, each requiring interpolation and rotation, and it must be computed once per pixel. This is better than doing an integral per pixel, but it can be simplified still further by precomputing some of the interpolation, as follows.

Once per surface texture, for \( j = 1, \ldots, J \), for \( l = 0, \ldots, M \), and for \( n = 0, \ldots, l \), compute

\[
a_{n}(\alpha_{j}) = \int_{Q} \cos \theta_{1} f_{s}(W, V) Y_{n}(\theta, \phi) \sin \theta d\theta d\phi
\]

for \( \alpha_{j} = (\alpha_{j}, 0) \). Once per viewing direction/surface-texture combination, and for \( j = 1, \ldots, J \), compute

\[
b_{n} = \sum_{l=0}^{M} a_{n}(\alpha_{j}) b_{l,n}.
\]
Now suppose we are at a specific pixel on a specific surface, where the normal \( N \) has spherical coordinates \((\alpha, \beta)\). For each \( n = -M, \ldots, M, \) we interpolate \( g_n(\alpha) \) from the \( g_n(\alpha_j) \) for the appropriate surface texture. In order not to introduce derivative discontinuities in the shading, which could be visible as Mach bands, some form of \( C^r \) interpolation, such as Catmull-Rom splines [6], should be used. We can precompute ahead of time the coefficients of the interpolation polynomials for each interval \( \alpha_l \leq \alpha \leq \alpha_{l+1} \), for \( n = 0, \ldots, M \), and for each surface texture. Once the \( g_n(\alpha) \) are found, the radiance \( L_{ref} \) becomes

\[
L_{ref} = \sum_{n=0}^{N} (g_n(\alpha) \cos n\beta + g_{-n}(\alpha) \sin n\beta)
\]

which results when equation (8) and the definitions for \( \tilde{a}_n \) are combined with equation (6). This equation represents \( L_{ref} \) as a Fourier series in \( \beta \) for fixed \( \alpha \), which is a useful way of approximating a smooth function on a circle. It is only necessary to evaluate \( \cos \beta \) and \( \sin \beta \), as \( \cos n\beta \) and \( \sin n\beta \) can be obtained from these inductively, with two multiplies and three adds each. Therefore, the total cost per pixel is two trigonometric function evaluations, about 12M multiplies, and about 9M adds. On a vector computer like the Cray-1, it is easier to do 2M trig evaluations, 8M multiplies and 7M adds, because everything can then be vectorized.

### 8. Computing the coefficients \( a_{ln}(\alpha_j) \)

The bidirectional reflection function enters into the shading computations of the previous section, in the integral of equation (7) for the coefficients \( a_{ln}(\alpha_j) \). We wish to apply the method of section 5 for the bidirectional reflection function from an explicit surface model. We could estimate this integral by assuming \( h_{\text{surf}}(W) \) was constant over each of the bins in the table. (Note that these bins are in spherical coordinates with \( N \) as axis, not \( V \).) But it is possible to get a better estimate, taking advantage of the exact flux and direction of the beam reflected from each facet.

For each of the sampled normals \( N_j \) in the \( XZ \) plane, place the bump texture in an orientation perpendicular to \( N_j \). The texture can be rotated about \( N_j \), and the calculations below can be summed over all the rotated facets to get better statistics for an isotropic bidirectional reflection function. Neglecting the \( \sin n\phi \) terms in the expansion automatically enforces averaging over the reflection of the texture in the \( XZ \) plane, by projecting onto the subspace of functions \( f_r \) with \( f_r(\theta, \phi) = f_r(\theta, -\phi) \).

By reciprocity, we can trace the beams from the fixed viewing direction \( V \), and accumulate the flux at the different lighting directions \( W \). Recall from Section 2 that \( B_i \) represents the non-observed area of the \( i^{th} \) facet, \( N_i \) its normal, and \( F(N_i, V) \) the Fresnel law, so that

\[
G_i = B_i F(N_i, V)(N_i \cdot V)/(V \cdot N) d\sigma
\]

is the fraction of the incoming flux reflected by the facet, in the direction \( W_i = 2(V \cdot N_i)N_i - V \).

Let \( h_{\text{surf}}(W) \) be a smoothing function of \( W \) which is maximum at \( W_i \), zero far away from \( W_i \), and has integral over the unit sphere equal to one. Then \( G(W) = \sum_{i=1}^{2\pi} h_{\text{surf}}(W)G_i \) is an approximation to the flux fraction per solid angle reflected from \( V \) to the unit sphere \( Q \), by the faceted surface, smoothed so as to be integrable.

In Equation (4), we divided this flux fraction per solid angle by \( V_i \cdot N_i \), which corresponds to \( W_i \cdot N \) here, to get the bidirectional reflection function. We need not divide now by \( W_i \cdot N = \cos \theta_n = \cos \theta_{in} \), since we want to estimate \( \cos \theta_{in} f_{\text{surf}} \) in the integral (7). Thus

\[
a_{ln}(\alpha_j) = \int_Q \cos \theta_{in} f_{\text{surf}}(\theta, \phi) Y_{lm}(\theta, \phi) d\omega
\]

\[
= \int_Q G(W) Y_{lm}(\theta, \phi) d\omega
\]

\[
= \sum_{i=1}^{2\pi} G_i \int_Q h_{\text{surf}}(W) Y_{lm}(\theta, \phi) d\omega
\]

Denote the spherical coordinates of \( W_i \) by \((\theta_i, \phi_i)\), and take the limit as \( h_{\text{surf}}(W) \) approaches the delta function. We get

\[
a_{ln}(\alpha_j) \approx \sum_{i=1}^{2\pi} G_i Y_{lm}(\theta_i, \phi_i)
\]

Therefore we can estimate the \( a_{ln}(\alpha_j) \) by computing \( G_i \) for each facet, and then for each \( l \) and \( n \), adding \( G_i Y_{lm}(\theta_i, \phi_i) \) to the appropriate sum.

The resulting spherical harmonic expansion defines a smooth function which is a least squares approximation to the density distribution of the flux in the various reflected beams.

### 9. Bump Maps and Results

The bidirectional reflection algorithm presented in section 5 relies on precomputed height or bump maps. We generated several bidirectional reflection function tables based on several different \( 258 \times 256 \) bump maps. These bump maps represented different types of surface micro geometries.

![Figure 3a. Sample white noise surface.](image)

Figure 3a. Sample white noise surface.

![Figure 3b. BDRF with \( \theta_{in} = 0.81 \) and \( \phi_{in} = 0.0 \)](image)

Figure 3b. BDRF with \( \theta_{in} = 0.81 \) and \( \phi_{in} = 0.0 \)
The random bump map was made up of scaled white noise. This surface was supposed to model isotropic diffuse surfaces much like the models used by Torrence and Sparrow [32]. The scale of the noise was chosen to be such that ratio of the heights to the inter-height spacing was similar to real surfaces. According to [29] this ratio is on the order of 0.01 - 0.1. We used a scale of 0.3 for our images. A scale of 0.3 results in surface with an RMS slope of m = 0.218 [8]. Figure 3a is a sample of this type of random surface. Figure 3b is a bidirectional reflection function of the surface in figure 3a for the incident angles θ = 0.81 and ϕ = 0.0. We built a 24 x 24 x 24 x 24 bidirectional reflection function table. The took 12 hours of VAX 11/785 cpu time to build the table.

Figure 4a and 4b are front and back renderings of an environment map (from NYIT) as reflected off a sphere. The sphere surface's BDRF was generated with the method discussed in section 5. The BDRF was combined with the spherical harmonics with the method described in section 8.

Conclusion

Table generated bidirectional reflection functions based on bump mapped micro surfaces can be used with spherical harmonics to render environments reflected off of various surfaces. We presented methods to approximate the micro surfaces, light reflecting off the surfaces, and the environment-reflection integral (eq. 1).

There are a number of future directions that need to be explored. Much more study into the micro surface properties of real surfaces needs to be done in order to more accurately model various micro surfaces. A bidirectional reflection function algorithm based on ray tracing could produce a more accurate horizon approximation and model higher order reflections. This ray tracer could be designed to exploit the surface geometry and thus be made to run faster than an ordinary ray tracer. Lastly a technique could be developed for taking the bidirectional reflection tables and blending them into texture maps for specific viewing and lighting angles.

Acknowledgements

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract number W-7405-ENG-48. We wish to thank Don Vickers, Ken Joy, Chuck Grant, Michael Gwilliam, Jeff Kaliman, and Dave Temple for helpful discussions and suggestions, and Ned Greene and the New York Institute of Technology for the environment map from [19].

References


