—-

Multiresolution B-spline Radiosity

Yizhou Yu† and Qunsheng Peng

State Key Laboratory of CAD&CG, Zhejiang University
Hangzhou 310013, P.R.CHINA

Abstract

This paper introduces a kind of new wavelet radiosity method called multiresolution B-spline radiosity, which uses B-splines of different scales to represent radiosity distribution functions. A set of techniques and algorithms, such as function extrapolation, adaptive quadrature, scale adjustment and octree, are proposed to implement it. This method sets up hierarchical structures on surfaces, keeps radiosity distribution continuous at element boundaries, does not need postprocessing, and does not prevent the use of any surface whose parameter domain is rectilinear.

Keywords: Computer graphics, Radiosity, Wavelet, B-spline.

1 Introduction

In computer graphics, radiosity computation is a fundamental method to solve the global illumination problem in environments consisting entirely of Lambertian reflectors and emitters. The result is a radiosity distribution function over the domain of the surfaces in the scene. Classical radiosity[5](CR),
derived from the radiative heat transfer, approximates the radiosity distribution as piecewise constant. An energy balance argument gives rise to a linear system. In the rendering phase, these radiosity constants are interpolated to give a smooth appearance of the radiosity function. Heckbert has demonstrated that the linear system in radiosity can be derived by projecting the radiosity integral into a finite dimensional function space. The CR algorithm results from using the space of piecewise constant functions. Zhu and Peng[11] considered radiosity functions that are piecewise linear. Many researchers have explored the use of higher order bases in mesh construction and discontinuity meshing[10] as well as finite element methods[13, 14] which has been adapted to radiosity computation. However, discontinuity meshing needs to be generalized to be applicable in environments with curved surfaces. In such cases, neither shadow masks[14] nor adaptive meshing techniques seem accurate and efficient around shadow boundaries.

On the other side, Hanrahan et al.[9] presented a hierarchical radiosity method(HR) to lower the computational complexity of solving the linear system which arises in CR. Theoretically, it can reduce the O(n^2) form factors to O(n). The mathematical tools of wavelet analysis provide a general framework offering a unified view of both higher order element approaches, and the hierarchical radiosity methods. Gortler et al.[15] exploited wavelet techniques developed for the solution of integral equations to solve the radiosity equation. They used two class of wavelet-like bases to sparsely represent the kernel of the radiosity integral. The supporting sets of such basis functions on neighboring intervals do not overlap. They do not enforce any kind of continuity at element boundaries, possibly leading to blocky artifacts. Furthermore, the cost of projecting kernel increases significantly as the order of the basis increases.

A desirable radiosity method should be accurate, efficient, be able to produce visually acceptable images and convenient to control. These are our goals. This paper proceeds with a review of projection methods followed by a discussion of multiscale B-splines. Then we propose a progressive wavelet radiosity method called multiresolution B-spline radiosity. Finally, we discuss our implementation and report experimental results.

†This author is currently a graduate student at the Dept. of Computer Science, UCSB, Santa Barbara, CA 93106, USA. E-mail: yyz@cs.ucsb.edu

2 The Radiosity Integral Equation

In order to apply the appropriate mathematical tools to the solution of radiosity problems, it is convenient to express the radiosity equation and surface geometry in parametric form. Parametrically, the key radiosity variables(radiosity, emittance, reflectivity, etc.) are represented as functions of two variables, (s,t) or (u,v), over each surface i or j. By abstracting all the complexity of surface interaction into a single kernel function $K_{ij}(s,t,u,v)$, the radiosity equation can be written as an integral equation

$$B_i(s,t) = E_i(s,t) + \sum_j \int\int K_{ij}(s,t,u,v) B_j(u,v) du\, dv \tag{1}$$

where

$$K_{ij}(s,t,u,v) = \rho_i(s,t) F_{ij}(s,t,u,v) VIS_{ij}(s,t,u,v) A_j(u,v). \tag{2}$$

The form factors F_{ij} and area functions A_j can be further expanded in terms of the functions describing surface geometry $x_i(s,t)$ and normals $n_i(s,t)$:

$$F_{ij}(s,t,u,v) = \frac{[\vec{l_{ij}}(s,t,u,v) \bullet \vec{n}_i(s,t)][-\vec{l_{ij}}(s,t,u,v) \bullet \vec{n}_j(u,v)]}{\pi \parallel \vec{l_{ij}}(s,t,u,v) \parallel^4} \tag{3}$$

$$A_j(u,v) = \left\| \frac{\partial \vec{x}_j(u,v)}{\partial u} \times \frac{\partial \vec{x}_j(u,v)}{\partial v} \right\| \tag{4}$$

where $\vec{l_{ij}} = \vec{x}_j(u,v) - \vec{x}_i(s,t)$.

3 Mathematical Background

Basis Set Projection

Let $\{M_i(s)\}$ be a basis of a certain function space V, and $\{DM_i(s)\}$ be another basis of V. $\{DM_i(s)\}$ is called the *dual basis* of $\{M_i(s)\}$ in space V if it satisfies

$$< M_i(s), DM_j(s) >= \int M_i(s) DM_j(s) ds = \delta_{ij},\ \forall i,j \tag{5}$$

Orthonormal bases are self-dual bases. The approximation of a function $B(s)$ in this function space can be expressed as a linear combination of the basis functions:

$$B(s) \approx \hat{B}(s) = \sum_i B_i M_i(s) = \sum_i < B, DM_i > M_i(s) \tag{6}$$

where $\{B_i\}$ are scalar coefficients obtained by performing inner products.

Multiresolution B-spline

Mallat[2] established multiresolution analysis based on wavelet bases. Let ψ be a wavelet function and $\{\psi_{j,k}\}$ be a Riesz basis of $L^2(R)$. For each $j \in Z$, let W_j be the closure of the subspace spanned by $\{\psi_{j,k} : k \in Z\}$, that is

$$W_j := clos_{L^2(R)} < \psi_{j,k} : k \in Z > \tag{7}$$

Then $L^2(R)$ can be decomposed to be the direct sum of subspaces $W_j, j \in Z$

$$L^2(R) = \dot{\sum_{j \in Z}} := \cdots \dot{+} W_{-1} \dot{+} W_0 \dot{+} W_1 \dot{+} \cdots \tag{8}$$

Let us consider the following closed subspaces of $L^2(R)$

$$V_j := \cdots \dot{+} W_{j-2} \dot{+} W_{j-1}\ ,\ \ j \in Z \tag{9}$$

These subspaces have the following properties: 1)$\cdots \subset V_{-1} \subset V_0 \subset V_1 \subset \cdots$; 2)$clos_{L^2}\left(\bigcup_{j\in Z} V_j\right) = L^2(R)$; 3)$\bigcap_{j\in Z} V_j = 0$; 4)$V_{j+1} = V_j \dot{+} W_j, j \in Z$; 5)$f(x) \in V_j \leftrightarrow f(2x) \in V_{j+1}, j \in Z$.
Let a function $\phi \in L^2(R)$, and

$$V_j = clos_{L^2(R)} < \phi_{j,k} : k \in Z >, j \in Z, \tag{10}$$

where $\phi_{j,k} = 2^{j/2}\phi(2^j x - k)$. Then ϕ produces a *multiresolution analysis*(MRA), if the nested series of subspaces $\{V_j\}$ satisfy 1), 2), 3), 5), and $\{\phi_{0,k} : k \in Z\}$ is a Riesz basis of V_0. ϕ is called a scaling function if ϕ produces a MRA.

Let $f_N \in V_N$, f_N has the unique decomposition,

$$f_N = f_{N-1} + g_{N-1} \tag{11}$$

where $f_{N-1} \in V_{N-1}, g_{N-1} \in W_{N-1}$. Repeating this process, we have

$$f_N = g_{N-1} + g_{N-2} + \cdots + g_{N-M} + f_{N-M} \tag{12}$$

where $f_j \in V_j, g_j \in W_j$. More and more details of f_N are peeled off when this process goes on.

For each positive integer m, let S_m^j be the function space of mth order B-splines with node sequence $2^{-j}Z, j \in Z$, that is,

$$S_m^j := \{f | f \in C^{m-2}, f|_{[k,k+1]} \in \pi_{m-1}, 2^j k \in Z\},$$

and V_j^m denotes the closure of $S_m^j \bigcap L^2(R)$ in $L^2(R)$. It has been demonstrated[1] that mth order B-spline basis $\{N_m(x-k) : k \in Z\}$ is a Riesz basis of V_0^m for $m \geq 2$. Checking $\{V_j^m\}$ with 1), 2), 3), 5), we find it is a MRA and its scaling function is $N_m(x)$. Make orthogonal decomposition $V_{j+1}^m = V_j^m \bigoplus W_j^m, j \in Z$. It has been showed that there exists compactly supported semiorthonormal wavelet ψ_m which produces subspaces $\{W_j\}$. The dual scaling function($DN_m(x)$) of $N_m(x)$ corresponding to ψ_m produces the same MRA $\{V_j^m\}$. So $\{DN_m(x-k) : k \in Z\}$ is the dual basis of $\{N_m(x-k) : k \in Z\}$ in V_0^m. Because both $\{N_m(x-k) : k \in Z\}$ and $\{DN_m(x-k) : k \in Z\}$ are Riesz bases of V_0^m, we can reconstruct a function $f \in V_j^m$ exactly by its projection coefficients in V_j^m, that is

$$f = \hat{f} = \sum_k < f, DN_m(2^j x - k) > N_m(2^j x - k)\ = \sum_k < f, N_m(2^j x - k) > DN_m(2^j x - k) \tag{13}$$

The computation of $DN_m(x)$ is outlined in appendix. The graphs of $DN_3(x)$ and $DN_4(x)$ are sketched in Figure 1.

(a) (b)

Figure 1. *(a)the graph of $DN_3(x)$, (b)the graph of $DN_4(x)$.*

The two-scale relation of mth order B-spline basis is

$$N_m(x) = \sum_{k=-\infty}^{\infty} p_{m,k} N_m(2x - k) \tag{14}$$

where $p_{m,k} = \left\{ \begin{array}{lll} 2^{-m+1} \left(\begin{array}{c} m \\ k \end{array} \right) & , & 0 \le k \le m; \\ 0 & , & otherwise \end{array} \right\}$.

Consider the B-spline function

$$f_{j_0}(x) = \sum_l a_l^{(j_0)} N_m(2^{j_0} x - l) \tag{15}$$

Its expression in $V_{j_0+1}^m$ can be written as

$$f_{j_0+1}(x) = f_{j_0}(x) = \sum_l a_l^{(j_0+1)} N_m(2^{j_0+1} x - l) \tag{16}$$

where

$$a_l^{(j_0+1)} = \sum_k p_{m,l-2k} a_k^{(j_0)}, l \in Z. \tag{17}$$

At $x = k/2^{j_1} (k \in Z, j_1 \ge j_0)$, we have

$$f_{j_0}(k/2^{j_1}) = \sum_l \omega_{m,k-l} a_l^{(j_1)} \tag{18}$$

where $\omega_{m,k} := N_m(k), k \in Z$, and $\{a_l^{(j_1)}\}$ can be obtained by applying (17) recursively.

In two dimensional function space $L^2(R^2)$,we use *tensor product B-spline basis* and its dual $\{DN_m(x-k)DN_m(y-l) : k,l \in Z\}$. They can also produce a MRA in $L^2(R^2)$. A relation similar to (13) also holds in this case. It plays an important role in our radiosity algorithm.

4 Sampling and Reconstruction

From section 3, we know there are two steps to compute the projection($\hat{f}$) of function f in function space V.

First,

$$f_i =< f, DM_i >, i \in Z \tag{19}$$

Second,

$$\hat{f} = \sum_i f_i M_i \tag{20}$$

In radiosity computation, the function to be projected is a radiosity distribution $B(s,t)$ on a certain surface. The right side of (19) is an integral. But the radiosity distribution $B(s,t)$ is an unknown function at first. We can compute values of $B(s,t)$ at some sample points, and perform the above inner product by some quadrature rule. Again, we can not directly obtain the balanced sample values of $B(s,t)$. We can only compute the increments in an iteration. Progressive refinement is adopted here. In each iteration, one shooting surface is selected and all others are receiving surfaces. On each receiving surface, the sample values of received energy times albedo are increments of radiosity at sample points. The projection coefficients in (19) are evaluated by quadrature and the increment of radiosity function is reconstructed by equation (20). Let A_j be the shooting surface, $\hat{B}_j(u,v)$ be the shot energy distribution, A_i be the receiving surface, (s_0, t_0) be the parameters of a sample point. The increment at this point is

$$\Delta B_i(s_0,t_0) = \int\int K_{ij}(s_0,t_0,u,v)\hat{B}_j(u,v)du\,dv \tag{21}$$

which is a double integral and can be evaluated by quadrature again.

The whole process of our algorithm is outlined as follows, where $\hat{B}_i(s,t)$ is the total accumulated radiosity, $\Delta\hat{B}_i(s,t)$ is the accumulated unshot radiosity and $d\hat{B}_i(s,t)$ is the reconstructed distribution of radiosity increment in an iteration.

for each emitting surface L_i
    compute the projection of its emittance distribution $\hat{E}_i(s,t)$;
    $\hat{B}_i(s,t) = \hat{E}_i(s,t)$;
    $\Delta\hat{B}_i(s,t) = \hat{E}_i(s,t)$;
while(!converged)
    pick out the surface A_{j_m} which has the largest value of $\int\int_{A_j} \Delta\hat{B}_j$
    for each receiving surface A_i
        for each sample point $p_k(s_k,t_k)$ on A_i
            compute $\Delta B_i(s_k,t_k)$;
        compute projection coefficients by quadrature and obtain $d\hat{B}_i(s,t)$ from them;
        $\hat{B}_i(s,t) = \hat{B}_i(s,t) + d\hat{B}_i(s,t)$;
        $\Delta\hat{B}_i(s,t) = \Delta\hat{B}_i(s,t) + d\hat{B}_i(s,t)$;
    $\Delta\hat{B}_{j_m} = 0$;

In fact, only projection coefficients are stored, the reconstruction formula (20) is only applied in the rendering phase. The addition of two distribution functions is actually carried out by performing addition of the coefficients corresponding to the same basis function. So it is convenient to perform superposition by basis set projection. In our algorithm, the shooting face can be a whole large surface, not a small patch in classical progressive refinement. So the influence of one shooting to the environment might be quite obvious.

5 Radiosity with Multiresolution B-splines

As we know, both tensor product B-spline basis and its dual can produce a MRA in $L^2(R^2)$. They can be substituted into (20) and (19). The *overlapping* support of these basis functions makes it infeasible to project the kernel. Therefore, we use progressive refinement and project the radiosity distribution function on each surface. Because the variation of radosity is different on different surfaces. It is more efficient to perform projection on different scale corresponding to different receiving surfaces or different areas on the same surface.

Energy Transfer between a Pair of Surfaces

We define the symmetric center of a B-spline basis function or a dual basis function to be its fulcrum. A two dimensional fulcrum can be defined likewise.

In the following, we will discuss one dimensional reconstruction first. Let f be a segment of polynomial of order m-1 on interval [t,t+1]. It can be represented accurately on this interval by a linear combination of m basis functions in terms of mth order B-spline basis. The fulcrums of these basis functions are $x - \frac{m}{2} + 1, x - \frac{m}{2} + 2, ..., x + \frac{m}{2}$. If we want to reconstruct this segment of polynomial by sampling, we must perform projection on m dual basis functions which have the same sequence of fulcrums. But the integral intervals of these projection are greater than [t,t+1]. So some sample values of f outside [t,t+1] should be known.

As shown in Figure 1 and the appendix, the dual basis of B-spline has exponential decay. The integral intervals can be made finite by truncation within a certain error tolerance. If the function f has no definition outside [t,t+1], it should be extended to cover all necessary integral intervals as well as satisfy a certain degree of continuity.

In radiosity computation, let the shooting surface be A_h, the receiving surface be A_i, its parametric domain be a square R[a,b;c,d]. The procedure of computing the increment of radiosity distribution, $d\hat{B}_i(s,t)$, on A_i could be divided into the following stages.

Adaptive Subdivision of the Parametric Domain This is a kind of predictive top down subdivision. After each subdivision, the smoothness of the increment radiosity distribution —-shadow is not taken into account—-on each subdomain is predicted. If it can be approximated by a double (m-1)th order tensor product polynomial, the subdivision is considered nearly sufficient in terms of (15) and will be stopped after once more. Otherwise,

the subdivision continues. In each subdivision, a subdomain is divided into four smaller subdomains of the same size. This is required by discrete wavelet analysis.

Adjustment of the Scale of Adjacent Subdomains and Computation of Sample Values If the scale of one subdomain is not smaller than or equal to twice the scale of one of its neighboring subdomains, it is recursively subdivided until the above condition is satisfied. This is to reduce the error incurred by reconstruction. After scale adjustment, the increment of radiosity at each sample point on each subdomain is computed.

Computation of Radiosity Increment and Its Partial Derivatives at Boundaries of the Parametric Domain Let $\Delta B_i(s_0, c)$ and $\Delta B_i(s_0, c + \Delta t)$ be computed by (21). At boundary P(s_0,c), the partial derivative of $\Delta B_i(s_0, c)$ with respect to parameter t could be evaluated by the following divided-difference:

$$\Delta B_{it}(s_0, c) = \frac{\Delta B_i(s_0, c + \Delta t) - \Delta B_i(s_0, c)}{\Delta t} \tag{22}$$

where $a \leq s_0 \leq b$.

At boundary $(s, c)(a \leq s \leq b)$, the radiosity increment and its partial derivative at each sample point could be evaluated as follows:

compute $\Delta B_i(a, c), \Delta B_{it}(a, c), \Delta B_i(b, c), \Delta B_{it}(b, c)$;
call function *edgefill*(a,b),

where the recursive function *edgefill*(s_1, s_2) could be written as

Begin
  $s_3 = \frac{s_1+s_2}{2}$;
  compute $\Delta B_i(s_3, c), \Delta B_{it}(s_3, c)$;
  if $|\Delta B_i(s_3, c) - \Delta B_i(s_1, c)|$ and $|\Delta B_{it}(s_3, c) - \Delta B_{it}(s_1, c)|$ are small enough
    compute the radiosity increment and its partial derivative with respect to t
      at boundary segment $(s, c)(s \in [s_1, s_3])$ by linear interpolation;
  else *edgefill*(s_1, s_3);
  if $|\Delta B_i(s_3, c) - \Delta B_i(s_2, c)|$ and $|\Delta B_{it}(s_3, c) - \Delta B_{it}(s_2, c)|$ are small enough
    compute the radiosity increment and its partial derivative with respect to t
      at boundary segment $(s, c)(s \in [s_3, s_2])$ by linear interpolation;
  else *edgefill*(s_3, s_2);
End.

The radiosity increment and its partial derivatives at boundaries $(s, d), (a, t), (b, t)(a \leq s \leq b, c \leq t \leq d)$ could be computed likewise.

Adaptive Subdivision and Sampling on Expanded Domain The scope of the expanded domain is determined according to scales of subdomains at the lowest level, which are produced in the first and second stages. The four boundaries of domain R[a,b;c,d] partition the expanded region into eight subregions. They are adaptively subdiveded as we have done in the first stage. The scales of subdomains at boundaries of R[a,b;c,d] and their neighboring expanded subdomains should also be adjusted. The value of radiosity on the expanded domain is determined by extrapolating the radiosity distribution on R[a,b;c,d].

Selection of Dual Basis Functions In order to perform projection, a set of dual basis functions and their scales should be determined. On R[a,b;c,d], when m is an odd integer, we determine the centre of each subdomain at the lowest level is the fulcrum of a selected dual basis function, and the scale of a selected basis function is the scale of the corresponding subdomain; when m is even, all corners of lowest-level subdomains are the fulcrums of selected dual basis functions(solid circles in Figure 2), and there may be two selected basis functions whose fulcrums coincide at one location, which happens when the location is the corner of two subdomains of different scale. On the expanded domain, fulcrums, that are necessary in reconstruction, should be selected. When m=3 or 4, they are the centres or corners of those expanded subdomains adjoining R[a,b;c,d]. This is shown in Figure 2, where square ABCD represents R[a,b;c,d]. As m increases, the number of required fulcrums also increases and they spread outward from the neighborhood of R[a,b;c,d].

Figure 2. *Selection of fulcrums*

Figure 3. *Shape of parameter region after truncation*

Projection Similar to one dimensional situation, the dual of tensor product B-spline basis has no compact support, but has exponential decay. So its support needs truncation before projection. According to the result of calculation, when a small $\varepsilon(\varepsilon > 0)$ is given, the parameter region, where the absolute value of a basis function is greater than ε, is approximately a rhombus (Figure 3). That is, the value of the basis function has the same order of magnitude on the four edges of the rhombus. This result can also be obtained from the property of exponential decay.

When projection is being performed, a rhombus region is determined according to a given error tolerance, the scale and location of the basis function. Only when a parametric subdomain completely or partially overlaps the rhombus, should quadrature be performed on this subdomain. Finally, all these quadratures on different subdomains are accumulated to obtain the result of projection.

Adaptive Subdivision in Shadow Computation In this stage,we determine the fraction of unoccluded energy from A_h to all points on A_i corresponding to those selected fulcrums. If the fraction corresponding to two adjacent fulcrums differs significantly, the corresponding subdomains are subdivided. This subdivision continues if the above condition is satisfied. Here, the scale of adjacent subdomains is also adjusted. And shadowed radiosity distribution will be computed after this subdivision.

Scale Unification and Superposition The parametric subdomains of the final partition generated from previous stages have different scale. Therefore, the dual basis functions corresponding to these subdomains and the projection coefficients stored at fulcrums also have different scales. In order to facilitate storage and superposition, we unify their scales to be 2^{-j} by equation (17). The resulting radiosity increment $d\hat{B}_i(s,t)$ on surface A_i belongs to the two dimensional counterpart of V_j^m. Here, 2^{-j} is the smallest scale of those selected dual basis functions.

When unifying their scale, we investigate all fulcrums of dual basis functions of scale 2^{-j} in and around the domain R[a,b;c,d]. If there is no projection coefficient of scale 2^{-j} at some fulcrum, it is obtained by applying (17) to those coefficients stored around the fulcrum whose scale is $2^{-(j-1)}$. If some of these required coefficients are still unknown, (17) is applied recursively to obtain them.

When performing superposition, we again apply (17) to unify the scales of $d\hat{B}_i(s,t)$ and $\Delta\hat{B}_i(s,t)$ to be the smaller one.

Iteration

In the algorithm introduced in section 4, if the radiance on light source L_i is a constant, the projection coefficients of $E_i(s,t)$ must be the same constant. Otherwise, it should be projected to obtain $\hat{E}_i(s,t)$.

In each iteration, the shooting surface is the one which has the largest value of the integral $\int\int_{A_j} \Delta\hat{B}_j$. Although it is difficult to compute this integral accurately, it could be evaluated approximately.

6 Implementation

Adaptive Quadrature

If we want to integrate $f(x)$ on interval [a,b] and the absolute error should be less than ε, the rule of adaptive quadrature is as follows: choose a specific quadrature rule Q and the number of sample points n; apply Q on [a,b] with n and $n+1$ sample points and obtain two results F_0 and F_1, if $|F_0 - F_1| < \varepsilon$, stop; otherwise, divide [a,b] into $[a, \frac{a+b}{2}]$ and $[\frac{a+b}{2}, b]$; repeat the above process on these two intervals recursively with error tolerance $\frac{\varepsilon}{2}$. In this way, the final accumulated error will not exceed ε.

In this paper, two dimensional adaptive Gauss-Legendre quadrature is used because Gauss-Legendre quadrature rule has high algebraic accuracy. So the previously mentioned sample points on surfaces correspond to those sample points in Gauss-Legendre quadrature. Two dimensional Gauss-Legendre rule can be written as follows

$$\sum_{i=1}^{n}\sum_{j=1}^{n}\omega_i\omega_j f(x_i, x_j),$$

where $\omega_1, \omega_2, \cdots, \omega_n$ and $x_1, x_2, \cdots, x_n$ are the weights and sample points in one dimensional case.

Adaptive quadrature is used to evaluate (21) at each sample point on the receiving surface A_i. It needs adaptive subdivision of the shooting surface A_h. In order to accelerate convergence, the radiosity distribution $\Delta\hat{B}_h(s,t)$ on A_h is replaced with a lower resolution version by applying the decomposition relation in wavelet analysis to remove higher resolution details. In the quadrature, one reparameterization factor should be included. It consists of two parts. One is the area $A_j(u,v)$ in the kernel. The other is the reparameterization factor between parametric domains of the shooting surface and Gauss-Legendre quadrature.

Oracle

It is required that at the end of subdivision, the increment radiosity distribution on each subdomain of the receiving surface can be approximated by a double (m-1)th order tensor product polynomial. During the subdivision, this is predicted by twice Gauss-Legendre quadrature with $\lceil \frac{m}{2} \rceil \times \lceil \frac{m}{2} \rceil$ and $m \times m$ sample points on current subdomain respectively. If the difference between them is below a threshold, the above requirement is considered to be satisfied and subdivision is stopped on current subdomain.

Scale Adjustment

As shown in Figure 4, there are three other subdomains adjoining each corner of a subdomain(D). So there are totally twelve such neighboring subdomains($D_i, i = 0, 1, \cdots, 11$). We implemented a recursive procedure *NeighAdjust*() to carry out scale adjustment. This procedure makes the requirement in section 5 satisfied by checking these twelve neighboring subdomains only. The details of this procedure are left out in this paper. When subdividing the expanded domain, we require the scale of boundary subdomains in R[a,b;c,d] be equal to that of its neighboring expanded subdomains.

Figure 4. *Scale adjustment*

Figure 5. *Eight patches of the expanded domain*

Extrapolation of Radiosity Distribution

There are two ways of extrapolation to obtain sample values of radiosity on the expanded domain. One is linear extrapolation. The other is higher order extrapolation. In Figure 5, R[a,b;c,d] is denoted by the square ABCD. There are eight patches($0, 1, \cdots, 7$) of the expanded domain around it. The linear extrapolation on patch 1 is implemented as follows:

$$\Delta B_i(s_0, t_0) = \Delta B_i(s_0, c) + (t_0 - c)\Delta B_{it}(s_0, c), \quad a \le s_0 \le b, t_0 < c \tag{23}$$

The linear extrapolation on patch 3,5 or 7 is implemented likewise. The linear extrapolation on patch 2 is implemented as follows:

$$\Delta B_i(s_1, c) = \Delta B_i(b, c) + (s_1 - b)\Delta B_{is}(b, c), \quad s_1 > b \tag{24}$$

$$\Delta B_i(s_1, t_1) = \Delta B_i(s_1, c) + (t_1 - c)\Delta B_{it}(b, c), \quad t_1 < c \tag{25}$$

The linear extrapolation on patch 0,4 or 6 is implemented likewise. If $\Delta B_{is}(s, t)$ or $\Delta B_{it}(s, t)$ in the above equations is too large, the error incurred by truncation may increase significantly. So when this does happen, the value of $\Delta B_{is}(s, t)$ or $\Delta B_{it}(s, t)$ is reduced into a predetermined range.

The second order extrapolation has the following restrictions on patch 1:

$$\begin{array}{l} \Delta B_i(s_0, c_-) = \Delta B_i(s_0, c), \\ \Delta B_{it}(s_0, c_-) = \Delta B_{it}(s_0, c), \\ \Delta B_i(s_0, (c-1)_-) = \Delta B_i(s_0, (c-1)_+), \\ \Delta B_{it}(s_0, (c-1)_-) = \Delta B_{it}(s_0, (c-1)_+), \\ \Delta B_i(s_0, t) = \Delta B_{it}(s_0, t) = 0, \ t \le c - 2. \end{array} \tag{26}$$

We can also impose the following restrictions on the third order extrapolation on patch 1:

$$\begin{array}{l} \Delta B_i(s_0, c_-) = \Delta B_i(s_0, c), \\ \Delta B_{it}(s_0, c_-) = \Delta B_{it}(s_0, c), \\ \Delta B_i(s_0, t) = \Delta B_{it}(s_0, t) = 0, \ t \le c - 1. \end{array} \tag{27}$$

Higher order extrapolation on patch 1,3,5 or 7 is a piecewise polynomial satisfying restrictions similar to (26) or (27). And on patch 2,4,6 or 8, it is a tensor product polynomial. Higher order extrapolation should be more efficient than linear extrapolation because its value and derivative equal zero beyond a certain neighborhood of R[a,b;c,d].

Projection

Upon the domain R[a,b;c,d] and each patch of the expanded domain, there is a hierarchical structure constructed during subdivision. When performing projection, we start from the roots of these structures and test whether the rhombus overlaps the current subdomain. If they do overlap, we continue checking the four sons of the current subdomain; otherwise, the current subdomain is given up. When the current subdomain is a leaf, 2D $m \times m$ point Gauss-Legendre quadrature is performed because the product of the dual basis function and radiosity distribution on the current subdomain can be approximated by a tensor product polynomial of order at most double 2m-2. The sample values of the dual basis function are approximated by precalculated discrete values. The sample values of radiosity are those $m \times m$ samples calculated in section 6.2. If the scale of the current subdomain is larger than that of the dual basis function, we fit a double (m-1)th order polynomial to those $m \times m$ samples, subdivide the current subdomain to the desired scale and resample the fitted polynomial on each resulting leaf subdomain. In the quadrature, we should include the reparameterization factor between parametric domains of the receiving surface and Gauss-Legendre quadrature, but not the area $A_i(s,t)$.

Shadow

Corresponding to each selected fulcrum on the domain of the receiving surface, there is a partition of the shooting surface formed during the evaluation of sample values by (21) and by adaptive quadrature. In order to evaluate the fraction of the unoccluded energy, test rays are cast from the fulcrum being considered to each patch of the partition. Each patch is hit by a constant number of random rays regardless of its size. Let e_i be the energy shot by patch i toward the fulcrum, r_i be the fraction of the unoccluded rays among those hitting patch i. Then the fraction of total energy received by the receiving surface is approximated by

$$\frac{\sum r_i e_i}{\sum e_i} \tag{28}$$

The subdivision in shadow computation is based on this fraction. During subdivision, (17) is applied to those projection coefficients computed in the previous section to obtain projection coefficients at a smaller scale. After subdivision, each resulting coefficient is multiplied by its corresponding fraction to obtain projection coefficients of the shadowed radiosity distribution.

Octree

An octree is built before radiosity computation to accelerate the test of ray-object intersection during shadow computation. It is not a linear octree and similar to those introduced in [16], stored as a tree structure with its nodes linked by pointers.

Each node in the octree represents a cube in the object space. If the surfaces of objects are composed of polygons, we use the following algorithm to test node-polygon intersection when the octree is being built.

1)If the node intersects the bounding box of the polygon, go to step 2); otherwise, there is no intersection.

2)From the coordinate components of the vertices of the polygon, obtain three projections of the polygon on the planes xy, yz and zx, respectively; obtain the same three projections(three squares) of the node.

If one of the projections of the polygon is surrounded by its corresponding projection of the node, the node and the polygon surely intersect;

if one of the projections of the polygon is seperate from its corresponding projection of the node, the node and the polygon are also seperate;

otherwise, go to step 3).

3)If the eight vertices of the node are located at the same side of the plane where the polygon resides, they are seperate; otherwise, they intersect.

In this algorithm,a 3D test is converted to three 2D tests. Such 2D tests could be carried out by the method used in Warnock's hidden surface removal algorithm. We also propose a new method to perform the test of ray-polygon intersection.

1)Through each edge of each polygon, build a bounding plane with its normal pointing outward(Figure 6).This step can be carried out on the fly or before radiosity computation.

2)Locate the intersect between the ray and the plane where the polygon resides. Substitute the coordinates of the intersect into the equation of each bounding plane. If all the results are negative, the intersect is inside the polygon.

This approach is more reliable, and maybe more efficient.

The ways of rendering and back face removal are trivial and are left out in this paper.

Figure 6. *Bounding planes and their normals*

Figure 9. *The configuration of Figure 8*

7 Experimental Results

In order to obtain good visual effects, we should use third or even higher order B-spline basis,which has continuous first order derivative, to approximate radiosity distribution. [15] only gives some images using first and second order wavelet bases. On the other hand, wavelet radiosity adopts hierarchical subdivision. Each basis function only covers a local area. So low order basis is able to achieve good approximation. We feel 3rd or 4th order basis is enough. In the following, we give some experimental results using 4th order B-spline basis and linear extrapolation. 3rd order B-spline basis can produce similar results.

Error computations for a single energy transfer between parallel and perpendicular squares is used to indicate the accuracy of the method of this paper. All comparisons are made against an analytic formula advocated in [7]. The relative error metric used is

$$rea = \left\langle \frac{|B_{spline}(s,t) - B_{exact}(s,t)|}{B_{exact}(s,t)} \right\rangle_{s,t} \tag{29}$$

where the error is evaluated on a 256 by 256 grid of sample points on the receiving surface. Shooting and receiving squares are the same size. Let the edgelength of these squares be 1.0, h be the distance between two parallel squares, and d be the number of hierarchies on receiving squares. The following two tables exhibit the computed errors.

Figure 7(a) and (b) give pictures of errors in the above two cases. In Figure 7(a), h=0.5 and d=4. In Figure 7(b), d=3. In each picture, the upper left corner gives the accurate result; the upper right gives the approximated result; the lower right gives magnified relative errors at sample points; the lower left is the scale of error.

The lower blue receiving surface in Figure 8 is actually composed of three narrow coplanar rectangles, as shown in Figure 9. The radiosity distribution on these rectangles are computed seperately. But we can not

Table 1: *Relative error between parallel squares*

h	d	rea(percent)
1.0	2	0.880
1.0	3	0.126
0.5	2	2.020
0.5	3	0.402
0.1	3	1.700
0.1	4	0.534

Table 2: *Relative error between perpendicular squares*

d	rea(percent)
2	4.670
3	0.676

see any discontinuity of radiosity at boundaries between adjacent rectangles. It indicates that the technique of function extrapolation in this paper can keep a certain degee of continuity between coplanar surfaces.

Figure 10 is an image of a room with a table and a stool.

All these pictures are rendered without any postprocessing.

(a) (b)

Figure 7. *Relative error of single energy transfer between (a)parallel and (b)perpendicular squares*

8 Conclusion and Future Work

In this paper we have presented a new wavelet radiosity method called multiresolution B-spline radiosity which is basically different from that introduced in [15]. We also proposed a set of new techniques and algorithms to implement it.

Our method is essentially hierarchical. Theoretically, it has linear time complexity. It has overcome the problem of blocky artifacts at element boundaries[15] and keeps radiosity distribution continuous at these locations. It uses progressive refinement. On different surfaces in an environment, different B-spline bases can be used. Our algorithm does not require albedo to be constant over a surface. It also does not prevent the use of any surface whose parameter domain is rectilinear. And superposition can be easily performed.

However, better methods are needed to handle intersections between curved surfaces, accelerate convergence of quadrature at these intersections and improve shadow computation. To environments made up of planar objects, the preprocessing introduced in [17] can be used to accelerate testing ray-object intersection. Topological ralations of surfaces could be set up[18] to improve efficiency of computation at surface boundaries. They also may be helpful to keep radiosity distribution continuous at boundaries between coplanar surfaces.

Acknowledgements

This work was partially supported by China Natural Science Foundation and Natural Science Foundation of Zhejiang Province. We would like to thank professor Daren Huang for helping us get farmiliar with wavelet analysis. We would also like to thank Hujun Bao and all those who gave us help during the research.

References

[1] C.K.Chui, An Introduction to Wavelets, vol.1 of Wavelet Analysis and its Applications, Academic Press Inc., 1992.

[2] S.G.Mallat, "A Theory for Multiresolusion Signal Decomposition: The Wavelet Representation," IEEE PAMI–11(7), 1989, pp.674–693.

[3] I.Daubechies, "Orthonormal Bases of Compactly Supported Wavelets," Comm. Pure and Appl. Math, 1988, pp.909–996.

[4] B.Alpert, "A Class of Bases in L^2 for the Sparse Representation of Integral Operators," SIAM Journal on Mathematical Analysis 24(1), 1993.

[5] C.M.Goral, K.E.Torrance, D.P.Greenberg and B.Battaile, "Modeling the Interaction of Light Between Diffuse Surfaces," Computer Graphics, 18(3), 1984, pp.213–222.

[6] M.F.Cohen, S.E.Chen, J.R.Wallace, D.P.Greenberg, "A Progressive Refinement Approach to Fast Radiosity Image Generation," Computer Graphics, 22(4), 1988, pp.75–84.

[7] D.R.Baum,H.E.Rushmeier and J.M.Winget,"Improving Radiosity Solution Through the Use of Analytically Determined Form-Factors,"Computer Graphics,23(3),1989,pp.325–334.

[8] D.R.Baum, S.Mann, K.P.Smith and J.M.Winget, "Making Radiosity Usable: Automatic Preprocessing and Meshing Techniques for the Generation of Accurate Radiosity Solutions," Computer Graphics, 25(4), 1991, pp.51–60.

[9] P.Hanrahan, D.Salzman and L.Aupperle, "A Rapid Hierarchical Radiosity Algorithm," Computer Graphics, 25(4), 1991, pp.197–206.

[10] D.Lischinski, F.Tampieri and D.P.Greenberg, "Discontinuity Meshing for Accurate Radiosity," IEEE CG&A, 12(6), 1992, pp.25–39.

[11] Y.Zhu and Q.Peng, "Generalized Radiosity Method for Environments with Curved Surfaces," Progress in Natural Science(China), 2(5), 1992, pp.425–437.

[12] D.Lischinski, F.Tampieri and D.P.Greenberg, "Combining Hierarchical Radiosity and Discontinuity Meshing," Computer Graphics Proceedings, Annual Conference Series, 1993, pp.199–208.

[13] R.Troutman, N.L.Max, "Radiosity Algorithms Using Higher Order Finite Element Methods," Computer Graphics Proceedings, 1993, pp.209–212.

[14] H.R.Zatz, "Galerkin Radiosity: A Higher Order Solution Method for Global Illumination," Computer Graphics Proceedings, 1993, pp.213–220.

[15] S.J.Gortler, P.Schröder, M.F.Cohen and P.Hanrahan, "Wavelet Radiosity," Computer Graphics Proceedings, 1993, pp.221–230.

[16] J.D.MacDonald and K.S.Booth, "Heuristics for ray tracing using space subdivision," Visual Computer, (1990)6, pp.153–166.

[17] S.Teller and P.Hanrahan, "Global Visibility Algorithms for Illumination Computations," Computer Graphics Proceedings, 1993, pp.239–246.

[18] M.Mäntylä, An Introduction to Solid Modeling, Computer Sci. Press, 1988.

A Numerical Method of Computing Dual Basis of B-spline

Consider "matrix operator"

$$M_m := [\gamma_{m;k,l}], \quad m,k,l \in Z, m \geq 2 \tag{30}$$

where $\gamma_{m;k,l} =< N_m(\bullet - k), N_m(\bullet - l) >$. M_m is a positive definite matrix. Its inverse is denoted by

$$M_m^{-1} := [\mu_{m;k,l}], \quad m,k,l \in Z, m \geq 2$$

Because

$$\sum_s \mu_{m;k,s}\gamma_{m;s,l} = \delta_{k,l}, \quad k,l \in Z, \tag{31}$$

$$DN_m(x-k) = \sum_l \mu_{m;k,l} N_m(x-l), \tag{32}$$

M_m has the property: $\gamma_{m;k,l} = \gamma_{m;k+1,l+1}$. So $\mu_{m;k,l} = \mu_{m;k+1,l+1}$ and only $\{\mu_{m;0,l}\}$ needs to be evaluated. Given a positive integer N and an error tolerance ε, we use numerical method to invert a series of matrices and evaluate the values $\{\mu_{m;0,l} : |l| \leq N\}$. Consider matrix

$$M_{m;N} = [\gamma_{m;k,l}], \ -N \leq k,l \leq N.$$

Its inverse is denoted by $M_{m;N}^{-1} = [\mu_{m;k,l}^N], -N \leq k,l \leq N$. A series of matrices, $M_{m;N}^{-1}, M_{m;N+1}^{-1}$, $M_{m;N+2}^{-1}, \cdots$, are computed until $|\mu_{m;0,l}^{N+p+1} - \mu_{m;0,l}^{N+p}| < \varepsilon, |l| \leq N$. The following table lists some approximate values in $\{\mu_{3;0,l}\}$, which were obtained in this way.

Table 3:*Some values in* $\{\mu_{3;0,l} : l \in Z\}$

l	$\mu_{3;0,l}$	l	$\mu_{3;0,l}$	l	$\mu_{3;0,l}$	l	$\mu_{3;0,l}$
0	2.842171	5	-0.045804	10	0.000678	15	-0.000010
1	-1.321729	6	0.019722	11	-0.000292	16	0.000004
2	0.573326	7	-0.008492	12	0.000126	17	-0.000002
3	-0.247042	8	0.003656	13	-0.000054	18	0.000001
4	0.106378	9	-0.001574	14	0.000023		