Improving Radiosity Solutions Through the Use of Analytically Determined Form-Factors

Daniel R. Baum, Holly E. Rushmeier⁺ and James M. Winget

Silicon Graphics Computer Systems 2011 N. Shoreline Blvd. Mountain View, CA 94039-7311

[†]George W. Woodruff School of Mechanical Engineering Georgia Institute of Technology Atlanta, GA 30332-0405

Abstract

Current radiosity methods rely on the calculation of geometric factors, known as form-factors, which describe energy exchange between pairs of surfaces in the environment. The most computationally efficient method for form-factor generation is a numerical technique known as the hemi-cube algorithm. Use of the hemi-cube is based on assumptions about the geometry of the surfaces involved. First, this paper examines the types of errors and visual artifacts that result when these assumptions are violated. Second, the paper shows that these errors occur more frequently in progressive refinement radiosity than in the originally proposed full matrix radiosity solution. Next, a new analytical technique for determining form-factors that is immune to the errors of the hemicube algorithm is introduced. Finally, a hybrid progressive refinement method that invokes the new technique to correctly compute formfactors when hemi-cube assumptions are violated is presented.

CR Categories and Subject Descriptors: I.3.3 [Computer Graphics]: Picture/Image Generation — display algorithms, viewing algorithms; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism — color, shading, shadowing, and texturing.

General Terms: Algorithms.

Additional Key Words and Phrases: analytical form-factor, global illumination, progressive refinement, radiosity, Z-buffer.

1. Introduction

Generating realistic computer images requires accurate modelling of visible light. The radiosity method has proven to be a useful approach for computer graphics illumination calculations. For efficiency, radiosity methods rely on a numerical algorithm known as the hemi-cube algorithm [4] to compute form-factors. This paper examines the sources of error in the hemi-cube algorithm and develops a new technique to analytically determine form-factors and eliminate these errors.

The radiosity method, borrowed from thermal engineering [15] and introduced to computer graphics by Goral et al. [10] and Nishita and Nakamae [13], models light interreflections between diffuse surfaces. In the original, full matrix radiosity method, the environment to be rendered is discretized into small surfaces. Geometric factors, known as form-factors, which describe the energy exchange between surfaces are calculated for each pair of discrete surfaces. The formfactors are then used to generate a set of simultaneous equations defining the interrelationships between surfaces in the environment. Given the emitted intensities of light sources, the simultaneous equations are solved to yield the intensity of light leaving each small surface. Once the intensities have been calculated, the environment can be viewed from any direction without performing additional illumination calculations.

The primary advantage of the radiosity method is that, unlike ray-tracing, the solution is view independent. Precomputed light intensities from the radiosity method can be used for interactively directed tours of synthetic environments. Because of its view independence, the full matrix method has been implemented in several image synthesis testbeds [9][18].

In the radiosity method, much of the computational burden is in the determination of the form-factors. In an environment of N surfaces, N^2 form-factors must be determined. Originally, in Goral's presentation, form-factors were calculated by numerical integration of contour integrals [16]. This approach, however, did not allow for hidden surfaces. In Nishita and Nakamae's approach, again using numerical integration, hidden surfaces were detected by performing visibility tests between pairs of vertices. Cohen replaced these techniques with the hemi-cube algorithm which efficiently determines form-factors in the presence of hidden surfaces by combining Nusselt's analogy from heat transfer with the Z-buffer algorithm from computer graphics [4].

Even when the hemi-cube algorithm is used, a disadvantage of the full matrix radiosity method is the large computation and storage requirement prior to image generation. Cohen et al. [6] developed a revised method that greatly reduced the time to first image and overall storage costs by applying progressive refinement [3] to the radiosity method. In progressive refinement radiosity, the process of finding all form-factors and then solving a set of equations is replaced by a process that sequentially computes the incremental effect on all surfaces of the light emitted and/or reflected from each surface. Generally, a small number of light sources and bright surfaces determine most of the global illumination in an environment. By processing these first, the progressive method quickly iterates to a good estimate of the global illumination. The disadvantage of a long wait to first image is converted to the advantage of quickly obtaining an estimate of the global illumination of the environment. Furthermore, since form-factors are computed on-the-fly for each iteration, storage costs are reduced from $O(N^2)$ to O(N).

For efficiency, the progressive refinement approach utilizes the hemi-cube algorithm to compute form-factors. The hemi-cube algorithm is based on various assumptions about environmental

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.



geometry. When these assumptions are violated, the hemi-cube algorithm produces inaccurate form-factors, which in turn produce visual artifacts in the image. The structure of the progressive refinement method causes the hemi-cube assumptions to be violated more frequently and to a greater extent than in the full matrix method. As a result, not only are the visual artifacts more pronounced, but the method generally will not converge to the same solution as the more exact full matrix method.

This paper presents a new approach for calculating formfactors, employing a hybrid combination of analytical and numerical techniques. The analytical technique does not rely on the geometric assumptions underlying the hemi-cube algorithm. Thus, by employing the new analytical technique when the hemi-cube assumptions are violated, inaccuracies of form-factors and subsequent visual artifacts can be reduced and often eliminated.

The next section reviews the formulations of the full matrix and progressive radiosity techniques. Section 3 explains the assumptions underlying the hemi-cube algorithm and examines the inaccuracies that result when these assumptions are violated. In Section 4, the impact of the hemi-cube form-factor errors on both the full matrix and progressive radiosity solutions is investigated. Additionally, Section 4 explains why, in general, the progressive refinement method and the full matrix method do not produce equivalent solutions. In Section 5, an analytical technique for computing formfactors is formulated that is independent of the geometric assumptions required for the hemi-cube. Section 5 also integrates the new technique into a hybrid progressive refinement method. The implementation and results of the new algorithm are presented in Section 6.

2. Full Matrix And Progressive Refinement Radiosity Algorithms

In this section, a brief review of the full matrix and progressive refinement radiosity methods is presented in preparation for later error analysis.

2A. Full Matrix Radiosity

In the full matrix (FM) radiosity method, the light intensity leaving each surface in the environment is found by solving the following matrix equation:

$$\begin{bmatrix} 1 - \rho_{1}F_{11} & -\rho_{1}F_{12} & \dots & -\rho_{1}F_{1N} \\ -\rho_{2}F_{21} & 1 - \rho_{2}F_{22} & \dots & -\rho_{2}F_{2N} \\ \vdots & \vdots & \dots & \vdots \\ -\rho_{N}F_{N1} & -\rho_{N}F_{N2} & \dots & 1 - \rho_{N}F_{NN} \end{bmatrix} \begin{bmatrix} I_{1} \\ I_{2} \\ \vdots \\ I_{N} \end{bmatrix} = \begin{bmatrix} I_{E1} \\ I_{E2} \\ \vdots \\ I_{EN} \end{bmatrix}$$
(1)

where: $\rho_i = \text{reflectivity of surface i}$

 I_i = intensity of surface i (for diffuse surfaces = radiosity of surface i $/\pi)$

 I_{Fi} = the emitted intensity of surface i

N = the number of surfaces in the environment

The form-factor, F_{ij} , equals the fraction of energy leaving surface i that arrives at surface j and is given by:

$$F_{ij} = \frac{1}{A_i} \int_{A_j} \int_{A_j} \frac{\cos \theta_i \cos \theta_j \operatorname{HID} dA_j dA_i}{\pi r_{ij}^2}$$
(2)

HID is equal to 1 if dA_i is visible to dA_j and is equal to 0 otherwise.



Figure 1. Form-factor Geometry

Implicit in Eq. (1) is the assumption that the intensity across each surface is constant. Equation (2) can be viewed as the average over area Ai of the integral over Aj.

Solving Eq. (1) for a coarse discretization of the environment yields a good global estimate of illumination, but does not give the detail required for a realistic image. Details are obtained without increasing the size of the matrix in Eq. (1) by using a substructuring method developed by Cohen et al. [5], based on a hierarchy of surfaces [14]. Each surface is divided into relatively large subsurfaces or *patches*. Each patch is further divided into subsurfaces, termed *elements*. Form-factors are found from each element to each patch. Patch-to-patch form-factors are then calculated using:

$$F_{ij} = \frac{1}{A_i} \sum_{e \in E_i} A_e F_{ej}$$
(3)

where: E_i is the set of elements in patch i

 F_{ii} = the form factor from patch i to patch j

 F_{ei} = the form factor from element e to patch j

 A_e , A_i = the areas of element e and patch i, respectively

Using these patch-to-patch form-factors, the intensity of each patch is found by solving Eq. (1). The intensity of each element is then calculated using back substitution:

$$\mathbf{I}_{e} = \boldsymbol{\rho}_{e} \begin{bmatrix} F_{e1} & F_{e2} & \dots & F_{eN} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{1} \\ \mathbf{I}_{2} \\ \vdots \\ \mathbf{I}_{N} \end{bmatrix}$$
(4)

where the I_N are the intensities of the N patches in the environment.

2B. Progressive Refinement Radiosity

Progressive refinement (PR) radiosity can be viewed as the radiosity equivalent of backward ray tracing [2] or two-way ray tracing [18]. Intensities in the environment are incremented by *shooting* energy out from surfaces for which estimates of intensity have already been found. The incremented intensities are found from the following equation:

where: θ_i , θ_j , and r_{ij} are shown in Fig. 1 326

$$\begin{bmatrix} \Delta I_{1} \\ \Delta I_{2} \\ \vdots \\ \Delta I_{N} \end{bmatrix} = I_{i} \begin{bmatrix} \rho_{1} F_{1i} \\ \rho_{2} F_{2i} \\ \vdots \\ \rho_{N} F_{Ni} \end{bmatrix}$$
(5)

where: I_i = the intensity of a surface i for which an intensity estimate has already been made

 ΔI_{e} = the increment in the estimated intensity for element e.

As in the FM solution, a low cost, detailed solution is needed. Again, patch/element substructuring is employed. Intensity I_i on the right hand side of Eq. (5) is *shot* from the relatively large patches. Incremental intensities on the left hand side are calculated for the smaller elements.

3. Calculation Of Form-Factors Using The Hemi-Cube Algorithm

In this section the formulation and assumptions of the hemicube algorithm are reviewed. Then, for each assumption, the errors associated with its violation are discussed.

3A. The Hemi-Cube Algorithm

Both the FM and PR radiosity methods require form-factors F_{ij} between every pair of surfaces. The first step in simplifying the integral for F_{ij} in Eq. (2) requires two geometric assumptions: first, the proximity assumption, that the distance between surfaces A_i and A_j is great compared to the effective diameter of A_i ; second, the visibility assumption, that the visibility of surface i from dA_j does not change. The form-factor may then be approximated as:

$$F_{ij} \cong \int_{A_j} \frac{\cos \theta_i \cos \theta_i \text{ HID } dA_i}{\pi r_{ij}^2}$$
(6)

In the hemi-cube (HC) algorithm, form-factors are found by placing a HC over the center of surface i, as shown in Fig. 2. Each side of the HC is discretized into *hemi-cube pixels*. The form-factors from the center of surface i to each of these pixels are precomputed and are termed *delta form-factors*. All surfaces are projected onto each face of the HC using the Z-buffer algorithm to determine visibility.



Figure 2. The Hemi-cube

In the limiting case of an infinite resolution HC, the integral over A_j in Eq. (6) may be replaced with a summation over the associated HC pixels. In practice, only finite resolution HCs are possible. The third major assumption, the resolution or aliasing

assumption, is that the true projection of each visible surface onto the HC be accuractely accounted for using a finite resolution HC. This assumption and the equivalence of form-factors implied by Nusselt's analogy [15], allows the form-factor F_{ii} to be approximated by:

$$F_{ij} \cong \sum_{q \in Q_{ij}} \Delta F_q \tag{7}$$

where: Q_{ij} is the set of HC pixels through which surface j is visible to the center of surface i.

 ΔF_{q} is the delta form-factor associated with pixel q.

The HC algorithm is efficient for finding the form-factors F_{ij} from surface i to all other surfaces j (i.e. a matrix row). However, in the PR method, the form-factors F_{ji} from all surfaces j to a surface i are needed (i.e. a matrix column). In this case, the HC algorithm can be used to find F_{ij} , followed by application of the reciprocity relationship to yield:

$$F_{ji} = \frac{A_i}{A_j} F_{ij}$$
(8)

Note that the accuracy of F_{ji} depends strongly on the accuracy of F_{ij} , A_i , and A_j .

3B. Errors Created By The Violation Of The Proximity Assumption

The proximity assumption is violated whenever surfaces are adjacent to one another. Consider two perpendicular surfaces, S_1 and S_2 , of dissimilar size as shown in Fig. 3. The analytical values of the form-factors between these surfaces are $F_{21} = .247$ and $F_{12} = .0494$. The limiting values of these form-factors calculated with an infinite resolution hemi-cube are equal to the analytical values of the form-factors calculated from the centers of the surfaces: $F_{d21} = .238$ and $F_{d12} = .00857$. The approximation F_{d21} is relatively good, since the area of S_2 , is much smaller than the area of S_1 . In general, except for points near the boundary of S_2 and S_1 , the distance from S_2 to any particular point on S_1 is nearly the same for all points on S_2 . The approximation F_{d12} is quite poor, however. The distance from the center of S_1 to points in S_2 . Since form-factor dependence on distance is non-linear, the effects of changing distance do not cancel out, and the result is a poor estimate of the form-factor.



Figure 3. Two perpendicular surfaces of dissimilar size



Errors in form-factors are not limited to surfaces oriented at right angles to one another. In Fig. 4a we consider nine geometries: three possible HC pixel locations in combination with three possible element normal orientations. In Fig. 4b, the relative errors between true form-factors and form-factors calculated with the HC algorithm are plotted versus normalized distance for each geometry. Normalized distance is the distance between the patch center and the element center, divided by the patch diameter. For example, an element with a normal that is parallel to the patch normal, and that projects onto a pixel on the corner of the hemi-cube, the relative error in the formfactor is greater than 100% for normalized distances less than 0.3, and comes within 2.5% only for normalized distances greater than 1. For all of the geometries, the form-factors produced with the HC algorithm converge to the true values as the distance is increased. It is also true, however, that relative errors approach infinity as the normalized distance approaches zero. In the worst case, the two surfaces must be separated by at least five patch diameters (assuming a square patch) for the relative error to drop below 2.5 percent.







Figure 4b Relative errors between HC computed form-factors and true values (note Radial-Center is equivalent to Para.-Center)

3C. Errors Created By The Violation Of The Visibility Assumption

The visibility assumption requires that the variable HID stay constant across surface i for any given HC pixel j. Because HID is a discontinuous function (either 0 or 1), the single point evaluation is a possible source of significant error. In Fig. 5, the center of surface 1 has a complete view of surface 2, while intervening surface 3 obscures much of surface 1 from surface 2. In this case, the HC algorithm will overestimate F_{12} by using F_{d12} calculated from the center of surface 1.



Figure 5. Geometry where form-factor from center of surface 1 to surface 2 is a poor estimate of F₁₂ because of intervening surface 3

3D. Errors Created By The Violation Of The Aliasing Assumption

The aliasing assumption, that surfaces project *exactly* onto whole numbers of HC pixels, results in a problem very similar to that encountered in image display [8]. Consider the geometry shown in Fig. 6. An accurate value of the form-factor requires the true projected area of the surface onto the HC. Because of the finite resolution of the HC pixels however, the projected area and resultant form-factor may be either over or under estimated.



Figure 6. HC aliasing can cause over/underestimate of projected surface area

In many cases, HC aliasing results in intensity errors that are not visually detectable. However, HC aliasing can cause significant errors and disturbing visual artifacts when calculating form-factors to or from light sources. HC aliasing can be reduced by increasing the HC resolution or by using multiple HC samples and filtering the results (thus increasing the computational expense of finding formfactors).

4. Impact Of Hemi-Cube Form-Factor Error On Radiosity Solutions

In this section, the impact of errors arising from the violation of the proximity, visibility, and aliasing assumptions will be considered for both the FM and PR methods.

4A. Impact Of Errors In Full Matrix Radiosity

Using patch/element substructuring and invoking the HC algorithm from smaller elements, the FM method effectively numerically integrates the outer integral of Eq. (2).

In FM, the HC is located at the center of a small element and utilized to determine form-factors from the small element to larger patches. Therefore, violation of the proximity assumption is generally infrequent. This is demonstrated in Fig. 7. In the figures, S_1 is a light source, and S_2 is a white reflecting surface. Figure 7a shows the correct solution, i.e. when I_2 is equal to $F_{21}I_1$. Figure 7b shows the solution when the FM method is used with the approximation that I_2 equals I_1F_{d21} . As seen in the figures, the intensity of S_2 as computed by the FM method is quite close to the correct solution.



Figure 7a,b,c. Flat shaded renderings of geometry described in Fig. 3. Using true intensity values $I_1 = 1.0$, $I_2 = 0.247$ (left), using FM solution $I_1 = 1.0$, $I_2 = 0.238$ (center), and using PR solution $I_1 = 1.0$, $I_2 = 5(0.00857) = 0.0429$ (right).

Recall the situation in which the visibility assumption is violated as depicted in Fig 5. The probability of a violation decreases as the size of surface 1 decreases. Again, because the FM method utilizes the HC from a small element, visibility errors are less likely to occur.

Finally, consider the effects of aliasing error. In FM radiosity, for a given HC resolution, HC aliasing effects occur as a light source is made smaller [7]. As the light source size decreases, the number of HC pixels covered by the projection of the light onto the HCs of neighboring elements may vary substantially. This variation in computed projected area typically appears as a plaid pattern in the image.

4B. Impact Of Errors In Progressive Refinement Radiosity

By placing the HC at patches rather than elements, the PR method implicitly assumes that the outer integral of Eq. (2) is constant, rather than numerically integrating it as in the FM approach. For this reason, when patch/element substructuring is used, the PR method will not converge to the same solution as the FM method.

Since the PR method uses the HC to find form-factors from the patch to the smaller elements, the proximity assumption is frequently violated. Figure 7c shows the solution of the two perpendicular surfaces with the approximate form-factors obtained from a PR approach. Light is shot out from the source S_1 and the intensity of S_2 is given as $I_1F_{d12}A_1/A_2$. Clearly, the PR approach grossly underestimates intensity in this case. The global effects of this type of error are demonstrated in Figs. 8 and 9, both images of a cubical room containing a sphere resting on a pillar. Fig. 8 was generated using the FM method, Fig. 9 with the PR method. Note the areas where surfaces are in close proximity: the common edges between walls and the joining of the pillar with the floor. In Fig. 9 these areas are too dark in the PR method due to violation of the proximity assumption.



Figure 8. Flat shaded image of cubical room generated with FM method



Figure 9. Flat shaded image of cubical room generated with original PR method.

Visibility errors are more frequent in PR than in FM because form-factors are determined from relatively large patches. This effect



was noted in the original PR approach, and the recommended solution was to subdivide the shooting patch. The following criteria were given: subdivide a shooting patch if 1) there is a large intensity gradient across the patch and/or 2) if any element-to-patch formfactors exceed unity.

These criteria are necessary but not sufficient. Criterion 1 fails to improve the most critical form-factors — those from light sources. Figure 4 shows that element-to-patch form-factors computed with the HC can erroneously exceed unity when the element is extremely close to the patch. Thus, criterion 2 only corrects the size of the shooting patch when other surfaces lie very close to the patch.

In PR, HC aliasing is most apparent when shooting from a light source. For a given HC resolution, aliasing effects will appear and worsen as surfaces are subdivided into smaller elements. Fig. 9 shows the plaiding effect which can occur when computing form-factors from the light source.

5. Calculation Of Accurate Form-Factors

The inaccuracies due to violation of HC assumptions can be reduced or eliminated by recognizing that the HC serves two distinct functions: first, visibility determination between surfaces, second, form-factor calculation at the HC pixel level. For pairs of surfaces which do not violate the HC assumptions discussed in the previous section, the HC algorithm can be used for both visibility and formfactor calculation. When computing form-factors involving surfaces which are too close relative to their size, or for form-factors involving light sources, the HC should be used only for visibility determination. For these special cases, the form-factors should be determined by an alternate more accurate method. In this paper, the form-factors for close surfaces and the form-factors involving light sources are determined using an analytical formula.

A general approach to the generation of efficient algorithms is the appropriate application of specialized analytical simplifications prior to invocation of more general numerical methods. Many analytical solutions for form-factors have been determined for specific geometries [12]. However, for environments containing surfaces of arbitrary size and orientation a more general approach is needed. Some general analytical simplifications of the form-factor integral which are useful in numerical calculations have been outlined by Walton [17]. The most common simplification is application of Stoke's theorem to reduce the form-factor double area integral to a double contour integral. As mentioned earlier, Goral used the contour integral form to calculate form-factors.

In this paper, we propose an algorithm in which the outer integral in Eq. (2) will be integrated numerically while the inner integral will be evaluated analytically. By applying Stoke's theorem, the inner area integral can be reduced to a contour integral. For an arbitrarily oriented planar polygon, this contour integral can be evaluated pointwise for a differential surface area dA_j in closed form as:

$$F_{dA_jA_i} = \frac{1}{2\pi} \sum_{g \in G_i} N_j \bullet \Gamma_g$$
(9)

where: G_i is the set of edges in surface i

 \mathbf{N}_{i} is the surface normal for the differential surface j

 $\Gamma_{\rm g}$ is a vector with magnitude equal to the angle gamma (in radians) illustrated in Fig. 10, and direction given by the cross product of the vectors $\mathbf{R}_{\rm g}$ and $\mathbf{R}_{\rm g+1}$ as illustrated in Fig. 10.

A detailed derivation of Eq. (9) can be found in [11].



Figure 10. Geometry for evaluating analytical form-factor

Next we describe a procedure for combining Eq. (9) with the HC algorithm within the framework of the PR method. To minimize computational expense we employ a hybrid combination of the HC and analytical calculations. The hybrid method processes HC pixels in the standard fashion until detecting a violation of the geometric assumptions. For those selected pixels analytic evaluation is used. The new procedure uses the same patch/element substructuring as the previous radiosity methods. We first describe the application of this new approach to simple convex environments, then to more complex environments with hidden surfaces.

5A. Application To Convex Environments

In a convex environment all surfaces are in full view of one another. Variations of intensity across individual surfaces are continuous since there are no shadows. The procedure has two stages: first, the highly radiative surfaces are processed, then the remaining surfaces are considered. Usually the highly radiative surfaces correspond to light sources. To facilitate the description of the new method, light sources are used in place of highly radiative surfaces.



Figure 11. Using the HC to identify element sample points

Initially, the energy is shot from each light L. All elements in the environment are projected onto a HC positioned over the center of L. As shown in Fig. 11, each HC pixel corresponds to a sample point on a visible element. Each sample used in determining the form-factor F_{eL} is weighted by the area of the pixel projected onto the element, ΔA_{ep} , divided by the total visible element area. The increment of intensity for each element e is equal to the element-to-light form-factor, F_{eL} , times the light intensity, I_L . The form-factors F_{eL} are found by:

Initialize: For all elements e{

$$\mathbf{F}_{eL} = 0$$

Process: Project all elements onto HC centered over L For each HC pixel p and associated sample point on element e{

$$F_{eL} += F_{eL}^{analytical} \Delta A_{ep}$$

Normalize: For all elements e{

$$F_{eL} = \sum \Delta A_{ep}$$

where: $F_{eL}^{analytical}$ is computed using Eq. (9) ΔA_{ep} is the element area associated with pixel p.

Note that using this procedure:

- 1.) The accuracy of F_{eL} is independent of the point chosen on the light source with respect to geometrical errors.
- 2.) Here ΔA_{ep} plays the role of dA_j in Eq. (9), thus the assumption that the area from which the form-factor is calculated is small is satisfied by calculating from these small, sub-element sized areas.
- 3.) F_{eL} is a weighted average of samples, rather than a sum depending on sample count. Because the magnitude of F_{eL} does not depend directly on the number of HC pixels onto which element e projects, HC aliasing effects will begin to occur only when some elements entirely miss being projected onto the HC, rather than when the projection of elements cover a relatively small number of pixels. To minimize sensitivity to aliasing, we apply this analytical approach to find all form-factors from elements to light sources, regardless of whether the elements and light source are in close proximity.

In the second stage, the remaining surfaces are processed. Since these surfaces have a smaller effect on the overall illumination, the analytical approach is only required for surfaces found in close proximity. For *far* surfaces the patch-to-element form-factor is computed using the original HC algorithm followed by the application of reciprocity to yield the desired element-to-patch form-factor. Given an upper bound on the acceptable form-factor error, the criterion for whether two surfaces are close to each other can be determined using information from Fig. 4. With this criterion the procedure for non-light source patch is:

Initialize: For all elements e{

Process:

$$\mathbf{F}_{ei}^{A} = \mathbf{F}_{ie}^{N} = 0$$

Project all elements onto HC centered over patch i

For each HC pixel p and associated sample point on element e{

$$IF (D_{ep} < D_{min}) F_{ei}^{A} += F_{ei}^{analytical} \Delta A_{ep}$$

ELSE
$$F_{ie}^{N} += \Delta F_{p}$$

}

Normalize: For

For all elements e whose form-factors were computed analytically{

$$F_{ei}^A = \sum \Delta A_{ep}$$

For all elements

$$F_{ei} = F_{ei}^{A} + \left(\frac{A_i}{A_e}\right) F_{ie}^{N}$$

where: F_{ei}^{A} is the analytically computed portion of F_{ei}

 F_{ie}^{N} is the numerically computed portion of F_{ei}

 D_{ep} is the distance from the origin of the HC on patch i to sample point p on element e

 D_{\min} is the minimum distance derived from Fig. 4 such that the HC form-factor satisfy acceptable error bounds

 ΔF_{n} is the delta form-factor associated with pixel p from Eq. (7)

Using the analytical formula together with the closeness criterion substantially increases accuracy at a small computational expense.

5B. Application To A Non-Convex Environment

In a non-convex environment, not all surfaces have an unobstructed view of one another. While the HC is useful for determining visibility of an element from one particular point on a shooting patch, it provides no information on whether the element has unobstructed view of the entire shooting patch. Using either reciprocity alone to find F_{ij} (as in the original PR method) or using the analytical formula described above assumes a full view of the shooting patch and gives an incorrect result when part of the shooting patch is occluded.

To avoid violation of the visibility assumption, each shooting patch must be subdivided until its components are either fully visible to or fully hidden from each element in the environment. We propose two methods for detecting when such subdivision is necessary. First, visibility errors in using the procedure described in Section 5A can be detected in a non-convex environment by forming the following sum after finding all the element-to-patch form-factors to patch i:

$$TOTAL_F = \sum_{e} \frac{A_e}{A_i} F_{ei}$$
(10)

The value of TOTAL_F in Eq. (10) is the approximation of the sum of F_{ie} for all elements e in the environment. For the interior of a closed region, this sum should be unity. If the sum is greater than 1.0, some form-factors have been over estimated because of partially hidden surfaces. If the sum exceeds unity by a small amount (e.g. less than a few percent), the form-factors are approximately correct. In this instance, to avoid shooting out more energy from patch i than it actually has, the form-factors F_{ei} should be normalized by:

$$F_{ei} \approx TOTAL_F$$
 (11)

If the sum exceeds 1.0 by a significant amount then visibility errors exist for one or more surfaces in the environment. In this instance, the shooting patch needs to be subdivided, and the procedure recursively applied to each *sub-patch*.

A precise estimate of visibility is particularly crucial when shooting from light sources. The above procedure generates an average visibility error estimate over the entire environment. Errors in form-factors from lights, even for a relatively small number of elements, can produce undesirable artifacts. To avoid such errors, we



employ the following more stringent procedure when shooting from light sources:

Initialize:	Subdivide the shooting patch into four sub-patches $\Delta V = 0$
Process:	For each sub-patch s { N _{se} = [number of HC pixels covered by element e]
	For each element e in the environment { $\Delta V_e = \max N_{se} - \min N_{se}$ IF ($\Delta V_e > tol_e$) $\Delta V += \Delta V_e$
Check:	<pre>} IF (ΔV > tol) { Subdivide each sub-patch and apply this procedure recursively }</pre>

where: tol_e and tol are preset tolerances that are used for all light sources

This light source subdivision criteria is much more effective than the criteria put forth in the original PR method (as outlined in Section 4B). In particular, a light source does not typically develop an intensity gradient, so the original criterion based on these gradients never produces light source subdivision. Additionally, the second criterion, an element-to-patch form-factor exceeding unity, will produce subdivision only if a visible element lies very close to the light source.

6. Results

The algorithm introduced in this paper is implemented in C on the Silicon Graphics Iris 4D series of superworkstation running the UnixTM operating system. We took advantage of various features of the superworkstation architecture to accelerate the performance of our testbed implementation. All projection/visibility/item-buffer computations for the HC are performed in hardware using the Geometry PipelineTM and hardware Z-buffer [1]. Once the itembuffers for each face of the HC are generated by the raster system, they are read back to host memory and processed by the host processor(s).

Comparisons of three different radiosity algorithms against a reference solution were performed. The environment chosen was a sphere resting on a pillar in a cubical room. A FM solution where the test environment was discretized into 328 patches and 8176 elements was used as the reference solution. The same surface discretization was used in all test runs. The methods compared to the reference solution were the original PR method, the new hybrid PR method that computes form-factors analytically when surfaces are close, and a version of the new method that always determines form-factors analytically. All three variants of the progressive refinement method used identical HC resolutions: 400 by 400 when shooting from the light source and 200 by 200 elsewhere.

To help validate the accuracy of the algorithms, provide an objective error measure, and supplement visual image comparison, a numerical error estimate was developed. The error estimate is based on the L_2 norm of the piecewise constant **rgb** field over non-emitting surfaces. The relative error measure compared to a reference solution is defined as:

$$\text{Error} = \frac{\left| \frac{Ref_{-1}Compare}{2} \right|_{2}}{\left| \frac{Ref_{-1}}{2} \right|_{2}}$$
(12)

where the Lp norm of an rgb field I is defined by:

$$|\mathbf{I}|_{p} = \sqrt{\sum_{e \in E_{n}} \sum_{c \in rgb} (I_{ec})^{p} A_{e}}$$
(13)

and E_n is the set of non-emitting elements with areas A_n .

Intermediate and final element intensities for each variant of the PR method were recorded. These intensities were compared with the reference solution using the error measure above. Test results are summarized in Fig. 12. It should be noted that the *ambient term* as originally proposed by Cohen [6] was not used. The term, a heuristic approximation of the ambient illumination, was defined for display purposes only. Since it is not part of the actual solution, it was removed to more clearly portray the accuracy of the different algorithms.

The original progressive algorithm converged to within 19% of the reference solution. Comparing the resultant image, Fig. 9, with that of the reference solution, Fig. 8, reveals rather large differences.



Figure 12. Relative error by iteration for original and new hybrid PR methods



Figure 13. Flat shade hybrid PR

Flat shaded image of cubical room using the new hybrid PR method

As shown in Fig. 12, the new hybrid PR method converged to within 4% of the reference FM solution. Additionally, the difference in the relative errors between the 100% analytic and the hybrid was less than 0.01% at every iteration thus validating the closeness criterion. For the test environment, the hybrid method processed 62% of the HC pixels analytically. Converging to virtually the same solution, the hybrid method took 47.5 minutes compared to 70.9 minutes for the 100% analytical variant. A visual comparison between

the reference solution (Fig. 8) and the solution of the new hybrid method (Fig. 13) shows that visual differences are almost imperceptible.

To further demonstrate the viability of the new techniques, the hybrid method was run on a more complex environment. The modelled environment is the Barcelona Pavilion designed by Mies van der Rohe in 1929. The model consists of 2,676 patches and 10,041 elements; it includes numerous light sources of varying size



Figure 14. View of main room in Barcelona Pavilion



Figure 15. View from utility room in Barcelona Pavilion



and a high level of geometric detail. Although the number of elements has increased only slightly, the number of patches is almost an order of magnitude larger. For this reason it was impractical to compute the $O(N^2)$ storage FM solution; in contrast, the various O(N) PR methods were viable. The results of the new hybrid PR method are shown in two scenes from a walk-thru of the pavilion in Figs. 14 and 15. In the images, note the shadow detail, diffuse interreflections, and the lack of HC aliasing artifacts (plaiding)

7. Conclusion

This paper detailed the three assumptions underlying the numerical HC algorithm: proximity, visibility, and aliasing. When any of these assumptions are violated, the HC algorithm produces inaccurate form-factors causing visual artifacts in the resultant images. Although both the FM and PR methods as originally proposed rely on the HC algorithm to determine form-factors, it was shown both theoretically and empirically that the PR approach is much more likely to violate the HC assumptions. As a result, the PR method will generally not converge to the same solution as the more accurate FM method.

A new analytical technique to compute form-factors that is immune to the errors of the HC algorithm was introduced. Finally, a hybrid PR method that combines the new analytical technique with the original HC algorithm was presented. The hybrid method computes form-factors analytically when violations of the HC assumptions are detected and computes form-factors numerically otherwise.

8. Acknowledgements

We gratefully acknowledge the support of the entire Silicon Graphics team. In particular, Efi Fogel cofounded the radiosity project and wrote much of the testbed software. Dave Ligon modeled the Barcelona Pavillion. Tom Davis provided valuable input on the derivation of Eq. 9., and Rolf Van Widenfelt wrote visualization software. Val Jermoluk and Bill Staab expedited required equipment for this collaborative research effort. Paul Haeberli and Mark Compton prepared all color images in this paper for accurate reproduction by creating digital color separations using image processing tools that come standard with SGI's IRIX[™] operating system.

Finally, we thank George Walton of the National Bureau of Standards for providing useful information about existing computer programs for computing form-factors.

9. Reference

- 1. Akeley, Kurt, Tom Jermoluk, "High Performance Polygon Rendering," *Computer Graphics*(*SIGGRAPH'88 Proceedings*), Vol.22, No.4, August 1988, pp.239-246.
- 2. Arvo, James, "Backward Ray Tracing," Developments in Ray Tracing(SIGGRAPH'86 Course Notes), Vol.12, August 1986.
- Bergman, Larry, Henry Fuchs, Eric Grant, Susan Spach, "Image Rendering by Adaptive Refinement," *Computer Graphics* (*SIGGRAPH '86 Proceedings*), Vol.20, No.4, August 1986, pp.29-38.
- Cohen, Michael F., Donald P. Greenberg, "The Hemi-Cube: A Radiosity Solution for Complex Environments," *Computer Graphics(SIGGRAPH '85 Proceedings)*, Vol.19, No.3, July 1985, pp.31-40.

- Cohen, Michael F., Donald P. Greenberg, David S. Immel, Philip J. Brock, "An Efficient Radiosity Approach for Realistic Image Synthesis," *IEEE Computer Graphics and Applications*, Vol.6, No.2, March 1986, pp.26-35.
- Cohen, Michael F., Shenchang Eric Chen, John R. Wallace, Donald P. Greenberg, "A Progressive Refinement Approach to Fast Radiosity Image Generation," *Computer Graphics* (SIGGRAPH '88 Proceedings), Vol.22, No.4, August 1988, pp.75-84.
- 7. Cohen, Michael F., "A Consumer's and Developer's Guide to Radiosity," A Consumer's and Developer's Guide to Image Synthesis(SIGGRAPH'88 Course Notes), 1988.
- 8. Crow, Franklin C., "The Aliasing Problem in Computer-Generated Shaded Images," *Communications of the ACM*, Vol.20, No.11, November 1977, pp.799-805.
- 9. Domancich, Micheline. "Graphics Research: A Rambling Tour of French Research Labs Finds Them Hard at Work," *Computer Graphics World* (July 1988) pp. 113-114.
- Goral, Cindy M., Kenneth E. Torrance, Donald P. Greenberg, Bennett Battaile, "Modeling the Interaction of Light Between Diffuse Surfaces," *Computer Graphics(SIGGRAPH '84 Proceedings)*, Vol.18, No.3, July 1984, pp.213-222.
- 11. Hottel, Hoyt C., Adel F. Sarofim, *Radiative Transfer*, McGraw-Hill, New York, NY, 1967.
- 12. Howell, J. R., A Catalog of Radiation Configuration Factors, McGraw-Hill, New York, 1982.
- Nishita, Tomoyuki, Eihachiro Nakamae, "Continuous Tone Representations of Three Dimensional Objects Taking Account of Shadows and Interreflection," *Computer Graphics* (SIGGRAPH '85 Proceedings), Vol.19, No.3, July 1985, pp.23-30.
- Samet, Hanan, Robert E. Webber, "Hierarchical Data Structures and Algorithms for Computer Graphics, Part II: Applications," *IEEE Computer Graphics and Applications*, Vol.8, No.4, July 1988, pp.59-75.
- 15. Siegel, Robert, John R. Howell, *Thermal Radiation Heat Transfer*, Hemisphere Publishing Corp., Washington DC, 1981.
- 16. Sparrow, E. M., "A New and Simpler Formulation for Radiative Angle Factors," *Transactions of the ASME, Journal of Heat Transfer*, Vol.85, No.2, 1963, pp.81-88.
- Walton, George. N., "Algorithms for Calculating Radiation View Factors Between Plane Convex Polygons with Obstructions," *Fundamentals and Applications of Radiation Heat Transfer (24th National Heat Transfer Conference and Exhibition)*, HTD-Vol.72, August, 1987, pp.45-52.
- Zhu, Yining, Qunsheng Peng, Youdong Liang, "PERIS: A Programming Environment for Realistic Image Synthesis," *Computers and Graphics*, Vol.12, No.3/4, 1988, pp.299-308.