If the sample values are sufficiently dissimilar, then a higher final sampling rate is required to achieve a given image quality.

Adaptive rectangular jitter [Dipp85] recursively subdivides the original rectangular cells (pixels or subpixels) in one dimension only, using a randomly chosen subdivision plane with constant x, y or t. The original sample value is assigned to the appropriate subcell, and a new value is determined for the other subcell. Here, the local nature of jittering allows reuse of previously calculated sample values; thus, the total number of sample values (rays) is significantly reduced.

Importance sampling [Lee85; Lane94] is another stochastic sampling refinement. Again, the objective is to reduce the number of rays traced without reducing the image quality. Fundamentally, importance sampling divides the area of the filter or lighting function being sampled, e.g., the reflection function, into equal areas (or volumes) and assigns a fixed number of rays to each area (or volume). The rays are stochastically placed within the area or volume. As an example, if the function falls off quickly from its central core value, few rays are traced at the edges (extremes) of the function, while significant numbers are traced at the core of the function.

Lange [Lang94] uses importance sampling in an implementation of the rendering equation [Kaji86]. She uses a fixed number of rays, $N = 40$, at each pixel. The appropriate reflection model, Lambertian for diffuse reflection and Phong (see Sec. 5–3) for specular reflection, is used to distribute the rays over the surface hemisphere for a given pixel. For specular reflection it is not strictly necessary to sample the entire hemisphere; it is sufficient to consider a smaller solid angle centered around the mirror reflection direction. Varying the size of the solid angle cone determines the glossiness, or softness, of the specular reflection. Diffuse and specular reflection are combined by distributing the number of rays in proportion to k_d and k_s, i.e., $k_s N$ and $k_d N$. Light sources are assumed as circular disks and are sampled with either a uniform distribution over the area or uniformly along the edge in order to generate penumbra shadows. The center of the light sources is sampled by a single ray to determine the directed contribution to the illumination. This contribution is weighted by $w_{\ell s} = \cos^n \theta$, where θ is the angle between the light source direction and the surface normal. The direct light source intensity is then combined with the random intensity samples using $1 - w_{\ell s}$ as the weighting factor. Mitchell provides an interesting discussion of the improvement expected from stochastic (stratified) sampling [Mitc96].

Ray Tracing from the Light Source

Arvo [Arvo86] introduced the idea of illumination maps as a way of including diffuse reflection that results from the intersection of specularly reflected or refracted rays with diffusely reflecting objects in ray traced images. The concept extends nicely to combined ray tracing and radiosity algorithms (see Sec. 5–18). Arvo called the technique backward ray tracing and defined it as tracing rays from the light source into the scene. But to avoid confusion, the technique is more appropriately called *ray tracing from the light source*, or *bidirectional ray tracing*. Ray tracing from the light source is used *only* to determine the diffuse component of the reflected light. Normal ray tracing *from the eyepoint* is still used for visible surface determination, specular reflection and refraction, etc.

Ray tracing from the light source is a two-pass algorithm. In the first, or preprocessing, pass a dense shower of rays emanating *from each of the light sources* is cast into the scene. The rays are traced through the scene, including reflections and refractions. For each of the diffusely reflecting objects in the scene illuminated by specularly reflected/refracted light, an illumination map is constructed. The illumination map is a rectangular array on a 1×1 square, for which a one-to-one correspondence with the object exists, i.e., a parameterization function $T(u, v) \rightarrow S(x, y, z)$ and its inverse exists.

When a reflected/refracted ray hits a diffusely reflecting surface with an associated illumination map, the inverse parameterization function is used to deposit the ray's energy at the appropriate grid point using a bilinear interpolation to apportion the energy among the closest four grid points. Once the illumination maps for each surface are completed, the energy at each grid point in the map is converted to intensity by dividing by the corresponding surface area in object space. The surface area is approximated by the partial derivatives of the parameterization function, i.e.,

$$\text{Intensity at the grid point}(u, v) = \frac{\text{Energy}(u, v)}{\left| \dfrac{\partial T(u, v)}{\partial u} \times \dfrac{\partial T(u, v)}{\partial v} \right|}$$

The second pass uses conventional *from the eyepoint* ray tracing. When rendering a point on a surface with an associated illumination map, the parameterization function is used to access the illumination map. The intensity of the specularly reflected/refracted diffuse reflection contribution is obtained using bilinear interpolation among the closest four grid points in the illumination map, and is combined with the direct diffuse and specular reflection/refraction contributions.

5–17 Radiosity

The radiosity method describes an equilibrium *energy* balance within an enclosure. Fundamentally, radiosity accounts for only diffuse reflection. It is based on concepts originally developed for radiative heat transfer in thermal engineering. It was first adapted to computer graphics by Goral et al. [Gora84], and in a somewhat different form by Nishita and Nakamae [Nish85]. Goral's solution was limited to diffuse reflection within convex environments in which no object obscured another. Goral's work is unique in that the computational results were compared to an actual physical environment. Subsequently, Meyer et al. [Meye86b] conducted an extensive comparison of physically real and radiosity generated scenes. Test observers showed no more than a 50% chance of picking the actual physical environment over the computational results, i.e., no better than random guessing. Both the physically real and the computer generated

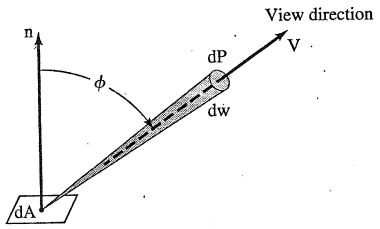

Figure 5–79 Radiant energy falling on a differential area; $d\omega$ is the differential solid angle in steradians.

scenes are shown in Color Plate 19. Cohen and Greenberg [Cohe85] extended the radiosity method to complex environments.

Consider a differential element of area dA. Radiant energy, dP, is assumed to emanate in all directions from dA. The energy, in the current context, is in the form of visible light. The radiant intensity, i, is given by

$$i = \frac{dP}{\cos\phi \, d\omega}$$

where i is the radiant energy per unit time, per unit projected area (in the viewing direction) per unit solid angle, $d\omega$ (see Fig. 5–79).

As shown in Fig. 5–80, for Lambertian diffuse reflection, the distribution of reflected light energy is

$$\frac{dP}{d\omega} = k\cos\phi \qquad k = \text{constant}$$

or

$$dP = k\cos\phi \, d\omega$$

Consequently, the intensity i of the diffusely reflected light is

$$i = \frac{k\cos\phi \, d\omega}{\cos\phi \, d\omega} = k \qquad \text{a constant}$$

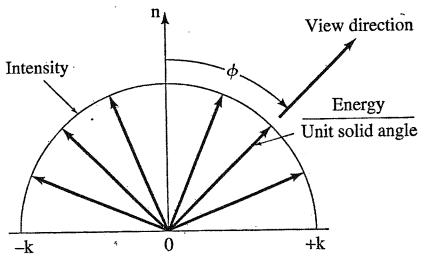

Figure 5–80 Lambertian diffuse reflection.

The total energy is found by integrating over the surface

$$P = \int_{2\pi} dP = \int_{2\pi} i\cos\phi \, d\omega = i\int_{2\pi} \cos\phi \, d\omega = i\pi$$

Thus, the energy and the intensity for ideal Lambertian diffuse reflection differ by a factor of π.

Enclosures

In order to determine the light energy at a surface, *all* the radiant energy from *all* directions in space must be accounted for. Consider a hypothetical enclosure consisting of a set of N surfaces, as shown in Fig. 5–81. The surfaces can be light sources, e.g., emitting surfaces; reflective; or fictitious, e.g., a window. Thus, the surfaces are considered as ideal diffuse reflectors, ideal diffuse emitters, a combination of diffuse reflector and emitter or of uniform composition with uniform illumination, reflection and emission intensities. Light sources are emulated by treating them as surfaces with specified illumination intensities. Directed light sources are handled by first independently computing their direct contribution to surfaces and then treating those surfaces as illuminated (emitting) surfaces in the radiosity solution.

The radiosity $B(i)$ is the hemispherical integral of the energy leaving the surface i. To an observer the surface i appears to be emitting a flux, B_i, from the imaginary surface. The flux consists of two parts: E_i, the rate at which the surface emits energy as a *source*, and $\rho_i H_i$, the rate at which incident energy is reflected back into the environment. Here ρ_i is the reflectivity of the surface, i.e., the fraction of incident energy reflected from the surface; and H_i is the radiant energy incident on the surface. Thus

$$B_i = E_i + \rho_i H_i$$

where the units are energy/unit time/unit area.

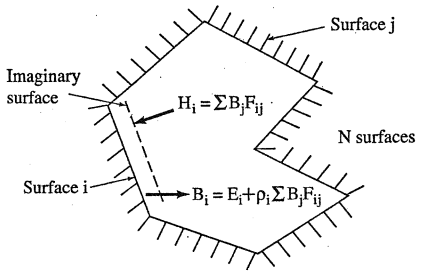

Figure 5–81 A hypothetical enclosure.

The incident flux on surface i, H_i, is the sum of the flux from all the surfaces in the enclosure that 'see' i (see Fig. 5–81). The fraction of the flux leaving surface j, B_j that reaches surface i is given by the form factor F_{ji}. Thus, the incident flux on surface i is

$$H_i = \sum_{j=1}^{N} B_j \frac{A_j F_{ji}}{A_i}$$

where the term within the summation represents the energy leaving patch j, $(B_j A_j)$, that reaches patch i, (F_{ji}), per unit area of patch i, $(1/A_i)$. Intuitively (but see below for more detail), the energy per unit time leaving patch j that reaches patch i is proportional to the area A_j of patch j. Similarly the energy per unit time leaving patch i that reaches patch j is proportional to the area of patch i. Although the energies of the two patches are not necessarily equal, for uniform energy distributions on each patch, the geometry intuitively suggests that $A_i F_{ij} = A_j F_{ji}$ where F_{ij} is the form factor from patch i to patch j. This is the classical radiosity reciprocity relation.

Substituting for H_i and using the reciprocity relation yields

$$B_i = E_i + \rho_i \sum_{j=1}^{N} B_j F_{ij} \qquad 1 \le i \le N$$

Multiplying by the area of patch i gives the energy per unit time leaving patch i, i.e.

$$B_i A_i = E_i A_i + \rho_i \sum_{j=1}^{N} B_j F_{ij} A_i \qquad 1 \le i \le N$$

Dividing each surface into patches and applying this equation to each patch yields a set of linear equations represented in matrix form as

$$\begin{bmatrix} 1-\rho_1 F_{11} & -\rho_1 F_{12} & \cdots & -\rho_1 F_{1N} \\ -\rho_2 F_{21} & -\rho_2 F_{22} & \cdots & -\rho_2 F_{2N} \\ -\rho_N F_{N1} & -\rho_N F_{N2} & \cdots & -\rho_N F_{NN} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \cdot \\ \cdot \\ B_N \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \cdot \\ \cdot \\ E_N \end{bmatrix}$$

The E_is represent the sources of illumination. If all the E_is equal zero, there is no illumination and all the B_is are zero. This system of equations is applied monochromatically for each bandwidth, for each 'color' (RGB). To get the radiant *intensity*, the B_is and E_is are divided by π. The system of equations can be solved using any standard equation solver. However, a Gauss-Seidel iterative technique is advantageous because the matrix is strictly diagonally dominant, hence rapid convergence to a solution is guaranteed.

Form Factors

To determine the form factors shown in Fig. 5–81, we consider the solid angle subtended by differential surface dA_j (see Fig. 5–82), as seen from differential surface dA_i. This is

$$d\omega = \frac{\cos\phi_j}{r^2}\, dA_j$$

Recalling that

$$i = \frac{dP}{\cos\phi\, d\omega}$$

and

$$P = i \int_{2\pi} \cos\phi\, d\omega = i\pi$$

yields

$$dP_i\, dA_i = i_i \cos\phi_i\, d\omega\, dA_i = \frac{P_i \cos\phi_i \cos\phi_j}{\pi r^2}\, dA_i\, dA_j$$

Recalling that the form factor is the *fraction* of the total energy emanating from dA_j, directly incident on dA_i, yields

$$F_{dA_i - dA_j} = \frac{P_i \cos\phi_i \cos\phi_j\, dA_i\, dA_j/\pi r^2}{P_i\, dA_i} = \frac{\cos\phi_i \cos\phi_j}{\pi r^2}\, dA_j$$

Integrating over A_j yields

$$F_{dA_i - A_j} = \int_{A_j} \frac{\cos\phi_i \cos\phi_j}{\pi r^2}\, dA_j$$

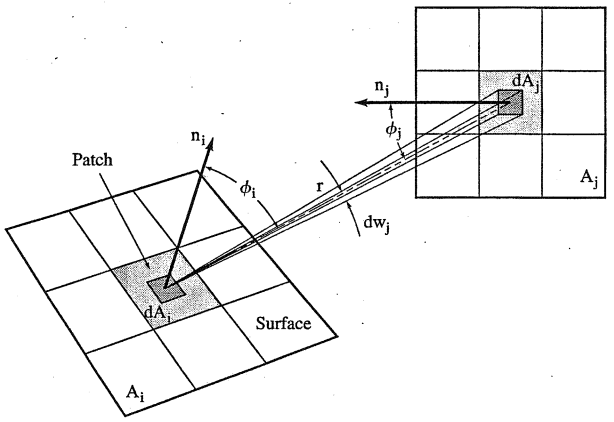

Figure 5–82 The solid angle subtended by dA_j.

which is the fraction of energy leaving dA_j and reaching A_i. The form factor between the finite areas is defined as the area average, i.e.

$$F_{A_i-A_j} = F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\phi_i \cos\phi_j}{\pi r^2} \, dA_i \, dA_j$$

where in arriving at this result the radiosity across each area (patch) is assumed constant. In addition, this result assumes that every part of each area is visible to the other area and that the distance between the patches is large compared to their size. If the distance between the patches is not large compared to their size, then the patches are further subdivided.

Cohen and Greenberg [Cohe85] introduce the factor H_{ij}, where $H_{ij} = 1$ if a straight line between the centers of the differential areas dA_i and dA_j does not intersect any other element in the scene, and $H_{ij} = 0$ if it does. H_{ij} accounts for occluding objects between the areas. Effectively, it produces the projection of area j visible from differential area i. Thus

$$F_{A_i-A_j} = F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\phi_i \cos\phi_j}{\pi r^2} \, H_{ij} \, dA_i \, dA_j$$

From the geometry and the definition of H_{ij}, $H_{ij} = H_{ji}$; and it is obvious that the integrals in this relation are identical. Thus, for uniform diffuse distributions

$$A_i \, F_{ij} = A_j \, F_{ji}$$

which is the reciprocity relation for form factors. Consequently, when calculating form factors only those for $i < j$ need be determined directly. Also, for a plane or convex surface that does not see itself, $F_{ii} = 0$; hence, only $N(N-1)/2$ form factors need actually be calculated.

Because F_{ij} is the fraction of energy that leaves a surface j, which reaches a surface i, and because the environment is enclosed, conservation of energy requires that

$$\sum_{j=1}^{N} F_{ij} = 1 \qquad \text{for } i = 1, N$$

Using this relation further reduces the number of form factors that actually require explicit calculation. Alternatively, it can be used to check the accuracy of numerical calculation of the form factors.

For a limited number of simple geometries analytic formulas give exact form factors. However, in general, numerical methods must be used. For example, Goral [Gora84] used Stoke's theorem to analytically convert the area integrals into contour integrals, that is

$$F_{ij} = \frac{1}{2\pi A_i} \oint_{C_1} \oint_{C_2} [\ln(r)\, dx_i\, dx_j + \ln(r)\, dy_i\, dy_j + \ln(r)\, dz_i\, dz_j]$$

However, it was necessary to solve the resulting integrals numerically by discretizing the boundaries. Thus, because the complexity of the resulting integrals requires a numerical solution, the analytical/numerical approach based on Stokes theorem is also limited to relatively simple environments.

The Hemicube

A geometric analog for the form factor equation is given by the Nusselt Analog (see Fig. 5–83). For a finite area, the form factor is equivalent to the fraction of the circle forming the base of a hemisphere covered by projecting the area onto the surface of the hemisphere and thence orthogonally downward onto the circle. If during projection the hidden (nonvisible) portions of patches are removed, then the procedure includes the effects of hidden surfaces.

Projection onto a sphere is difficult. However, note in Fig. 5–84 that any two patches which have the same area when projected onto the hemisphere have the same form factor. This is also true for any other *surrounding* surface. Cohen and Greenberg [Cohe85] introduced the idea of a hemicube, i.e., half of a cube, to determine a numerical approximation to the integral over the hemisphere, as shown in Fig. 5-85.

An imaginary hemicube is constructed around the center of the patch i in question. For convenience, the environment is transformed such that the center of the patch is at the origin and the normal is along the positive z-axis. The hemicube is divided into subareas, or hemicube 'pixels' (see Fig. 5-86).

All other patches j in the environment are projected onto the hemicube (see Fig. 5–86). If more than one patch projects onto a hemicube subarea, a visible

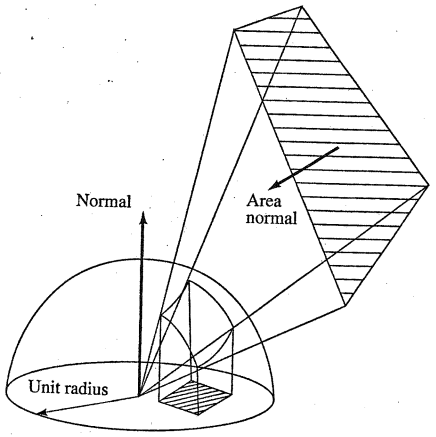

Figure 5–83 The Nusselt Analog. The form factor is equal to the fraction of the base covered by the projection.