



Light Reflection Models for Computer Graphics

Donald P. Greenberg

Science, New Series, Vol. 244, No. 4901. (Apr. 14, 1989), pp. 166-173.

Stable URL:

<http://links.jstor.org/sici?sici=0036-8075%2819890414%293%3A244%3A4901%3C166%3ALRMFCG%3E2.0.CO%3B2-6>

Science is currently published by American Association for the Advancement of Science.

Your use of the JSTOR archive indicates your acceptance of JSTOR's Terms and Conditions of Use, available at <http://www.jstor.org/about/terms.html>. JSTOR's Terms and Conditions of Use provides, in part, that unless you have obtained prior permission, you may not download an entire issue of a journal or multiple copies of articles, and you may use content in the JSTOR archive only for your personal, non-commercial use.

Please contact the publisher regarding any further use of this work. Publisher contact information may be obtained at <http://www.jstor.org/journals/aaas.html>.

Each copy of any part of a JSTOR transmission must contain the same copyright notice that appears on the screen or printed page of such transmission.

JSTOR is an independent not-for-profit organization dedicated to creating and preserving a digital archive of scholarly journals. For more information regarding JSTOR, please contact support@jstor.org.

non-matches occurred because class registers were unavailable because of the closing of the elementary school over the last 20 years or lack of adequate identifying information for the abused and neglected children.

38. An obvious limitation of this study is that the number in the control group who were actually abused, but not reported as such, is unknown. If the control group included subjects who had been officially reported as abused, at some earlier or later time period, this would jeopardize the design of the study. Thus, any child who had an official record of abuse or neglect was eliminated from the study, regardless of whether the abuse or neglect occurred before or after the time period of the study. An alternative was to include these subjects and treat them as a separate group in the analyses. However, because the number of these subjects was small ($n = 11$), this was not done.
39. M. E. Wolfgang, R. M. Figlio, T. Sellin, *Delinquency in a Birth Cohort* (Univ. of Chicago Press, Chicago, 1972); P. Strasburg, *Violent Delinquents* (Monarch, New York, 1978); D. Rojek and M. Erikson, *Criminology* **20**, 5 (1982).
40. Violent crimes include arrests for robbery, assault, assault and battery, battery with injury, battery, aggravated assault, manslaughter/involuntary manslaughter/reckless homicide, murder/attempted murder, rape/sodomy, and robbery and burglary with injury.
41. A reanalysis of these findings was done, excluding abuse and neglect cases who did not have matches. Thus, the number of individuals in each group was 667. The results do not change with this smaller sample size. In cases where differences were significant, they became even more significant. In the few cases where differences were not significant, these results remained the same.
42. Because these findings are based on official records and official records overrepresent minority groups, the most obvious explanation for the higher rates of arrests for violent crimes among blacks would be the bias and discriminatory treatment by the criminal justice system. However, this explanation does not seem to explain the differences among blacks and the lack of difference for the whites, unless we postulate a "double jeopardy" theory. Another possible explanation is that parental violence is more severe among blacks than whites or that nonwhites are more physically abusive with their children and within their homes than whites; however, the data indicate that this is not the case. Among whites, approximately 20% suffered physical abuse, compared to less than 9% for blacks. Blacks suffered

more neglect, relative to whites in the sample.

43. M. Gottfredson and T. Hirschi, *Criminology* **26**, 37 (1988).
44. R. J. Baker and J. A. Nelder, *The GLIM System. Release 3.77: Generalised Linear Interactive Modeling Manual* (Numerical Algorithms Group, Oxford, 1986).
45. Separate logit analyses were done by using different methods of dividing the abuse and neglect groups in addition to the one presented here, which is based on pure groups. In these analyses, the same pattern emerged, indicating the importance of physical abuse only and neglect. One exception was in replicating the logit analysis by using only those abused or neglected cases with matches. Here, in addition to physical abuse and neglect as significant predictors, sexual abuse only was also significant.
46. L. P. Groeneveld and J. M. Giovannoni, *Soc. Work Res. Abstr.* **13**, 24 (1977).
47. R. J. Gelles, *Am. J. Orthopsychiatry* **45**, 363 (1975); E. H. Newberger, R. B. Reed, J. H. Daniel, J. N. Hyde, M. Kotelchuck, *Pediatrics* **60**, 178 (1977).
48. R. J. Gelles and C. P. Cornell, *Intimate Violence in Families* (Sage, Beverly Hills, CA 1985).
49. M. D. Pagelow, "Child abuse and delinquency: Are there connections between childhood violence and later deviant behavior?" Presented at the Tenth World Congress of the International Sociological Association, Mexico City, Mexico, 1982.
50. G. Bach-y-Rita and A. Venio, *Am. J. Psychiatry* **131**, 1015 (1974); J. Kagan, *Daedalus* (Boston) **106**, 33 (1977); H. P. Martin and P. Bezley, *Dev. Med. Child Neurol.* **19**, 373 (1977); A. H. Green, *Am. J. Psychiatry* **135**, 579 (1978).
51. A. Frodi and J. Smetana, *Child Abuse Negl.* **8**, 459 (1984); M. A. Lynch and J. Roberts, *Consequences of Child Abuse* (Academic Press, London, 1982).
52. N. Garmezy, in *Further Explorations in Personality*, A. I. Robin, J. Aronoff, A. M. Barclay, R. A. Zucker, Eds. (Wiley, New York, 1981), pp. 196–269.
53. K. Heller, personal communication.
54. Supported in part by the National Institute of Justice grant 86-IJ-CX-0033, by Indiana University Biomedical Research grant S07 RR07031, and by a Talley Foundation grant while the author was a visiting scholar in the Psychology Department at Harvard University, Cambridge, MA. I thank A. Ames, J. Lindsay, B. Rivera, and B. Tshanz for assistance with the data collection and B. Ross for assistance with the data analysis.

Light Reflection Models for Computer Graphics

DONALD P. GREENBERG

During the past 20 years, computer graphic techniques for simulating the reflection of light have progressed so that today images of photorealistic quality can be produced. Early algorithms considered direct lighting only, but global illumination phenomena with indirect lighting, surface interreflections, and shadows can now be modeled with ray tracing, radiosity, and Monte Carlo simulations. This article describes the historical development of computer graphic algorithms for light reflection and pictorially illustrates what will be commonly available in the near future.

IN RECENT YEARS, THERE HAS BEEN AN ENORMOUS DEMAND for realism in computer imagery. Automobile designers would like to evaluate their new car designs without having to construct the full-size clay models commonly used in industry. Graphics simulations of dynamic systems are basic to today's aerospace, mechanical, and structural engineers. Modern pilot train-

ing is now conducted with real-time flight simulators, where the views from the cockpit are changing scenes simulating the landing at a specific airport. The display of biological organs, reconstructed from information obtained from tomographic scans, x-rays, or other noninvasive methods, greatly benefits the medical professions. And, of course, architects and interior designers would like to show their clients design concepts before they are constructed. The variety of uses of computer graphics are infinite, but all will require the ability to generate synthetic pictures of increasingly greater realism at increasingly greater speeds. Furthermore, as the complexity of these simulations increases with the inevitable availability of computer processing power, the ability to provide the visual cues such as shade, shadows, and motion and depth perception, will become necessary.

In general, to create a typical computer graphics image, it is necessary to perform the following five steps sequentially, in what is frequently called the graphics "pipeline".

1) *Three-dimensional model.* The initial step in the process is the modeling of the physical environment, including the geometry, the positions and orientations of all objects, and the material characteristics, textures, and finishes of all surfaces. The illumination, including the geometry of the light sources, the distribution of the light energy, and the color or spectral characteristics of the emission, must

The author is the Jacob Gould Schurman Professor of Computer Graphics and Director of the Program of Computer Graphics at Cornell University, Ithaca, NY 14853.

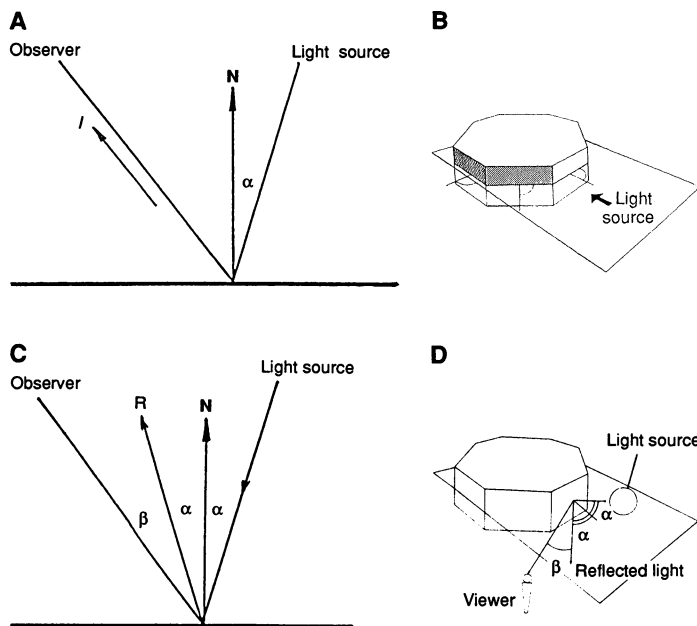


Fig. 1. (A) The intensity of light received by a surface is proportional to the cosine of the angle α between the light vector and the surface normal. $I = k_d \cos(\alpha)$. (B) As the surface turns more parallel to the direction of the light source, the intensity of the light reaching the surface becomes less. For a diffuse surface, the reflected light is equal in all directions. (C) The angle of reflection equals the angle of incidence. $I = k_s \cos^n(\beta)$. (D) For specular reflections, the amount of light reaching the observer is dependent on the angle β , the angle between the reflected ray and the sight vector.

also be defined. The computer model describes an environment that may be real or may be nonexistent.

2) *Perspective transformation.* Each vertex of the geometry describing the environment is mathematically transformed to generate a true perspective picture on an image plane while maintaining the necessary relative depth information. The picture is bounded within a preset cone or frustum of vision, and extraneous information outside of the field of view is "clipped" and discarded. Surfaces facing away from the observer are also removed in a "culling" operation.

3) *Visible surface determination.* The surfaces remaining after the perspective transformation and clipping operations are sorted in depth so that only the elements closest to the observer are displayed. In this way, opaque surfaces correctly occlude those that are further distant from the observer.

4) *Intensity or color determination.* The intensity or color of each element that is displayed on the image plane must be computed according to a light reflection model. This reflection model simulates the spatial and spectral distribution of the light reflected from each surface in the environment. Frequently, this step has been combined with the visible surface determination to reduce computational tasks.

5) *Image display.* The last stage in the graphics pipeline is the conversion of the image plane intensity information into a displayable form. For a television image, the picture is rendered by selecting the appropriate red, green, and blue intensities for each dot (pixel) in the visible scene.

In the high-performance graphics workstations of today, once the three-dimensional model has been defined, the perspective transformations (step 2), the visible surface determination (step 3), and the display routines (step 5) are commonly implemented in hardware. Furthermore, at least for simple direct illumination, the surface reflections can be approximated (step 4) and executed with hard-

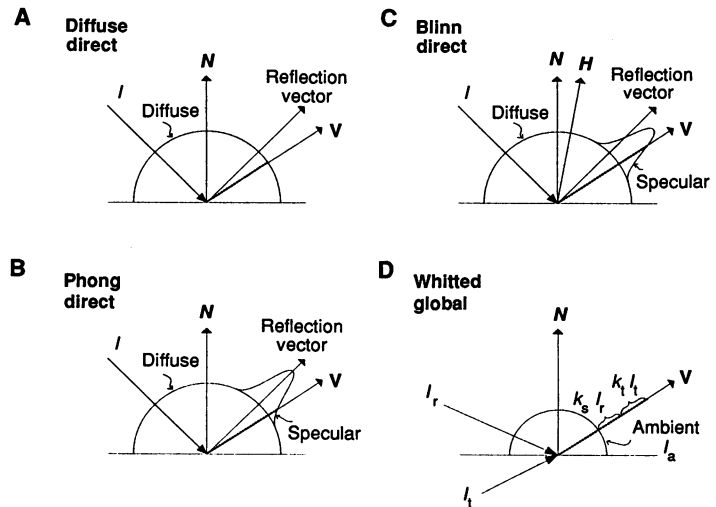


Fig. 2. These figures indicate the reflected intensity distribution for direct and indirect lighting according to different reflection models. In each case, the intensity seen by the viewer is shown by the dashed vector. (A) Pure diffuse reflection. Diffuse = $k_d (N \cdot L)$ object color. (B) The Phong model adds a specular distribution centered around the reflection direction from the direct lighting. Diffuse = $k_d (N \cdot L)$ object color; specular = $k_s (R \cdot V)^n$. (C) The Blinn model modifies the specular distribution and allows for "off-specular peaks". Diffuse = $k_d (N \cdot L)$ object color; Specular = $k_s (N \cdot H)^n$. (D) All three of the direct lighting models treat ambient reflections the same way with a constant global diffuse ambient term. The Whitted ray tracing model was the first to add the indirect specular and transmitted terms. Diffuse = I_a ; specular = $k_s I_r$; transmitted = $k_t I_t$.

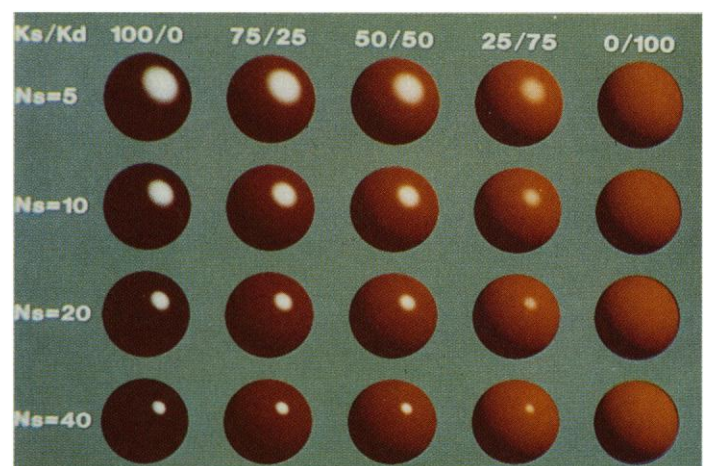


Fig. 3. This figure shows the effect of varying the diffuse and specular coefficients while the ambient term is held constant. The left-hand column is all specular, and thus the spheres are black since none of the object color is reflected (except for the ambient light). The right-hand column is all diffuse, reflecting the object color, but with no specular highlights. The exponent for the specular coefficient increases by row from top to bottom, depicting an increase in the glossiness of the surface. Image produced by Roy Hall.

ware implementations, creating a graphics pipeline capable of dynamically producing complex computer graphics simulations. One of the most interesting, challenging, and unresolved parts of the process is the determination of the true intensity or color of the visible surfaces seen in the final image.

To describe the propagation of light through an environment, a mathematical model of the physical laws governing electromagnetic radiation must be provided. These complex phenomena are most accurately simulated by means of a model rigorously based on wave optics and appropriate for surface reflections. These models can

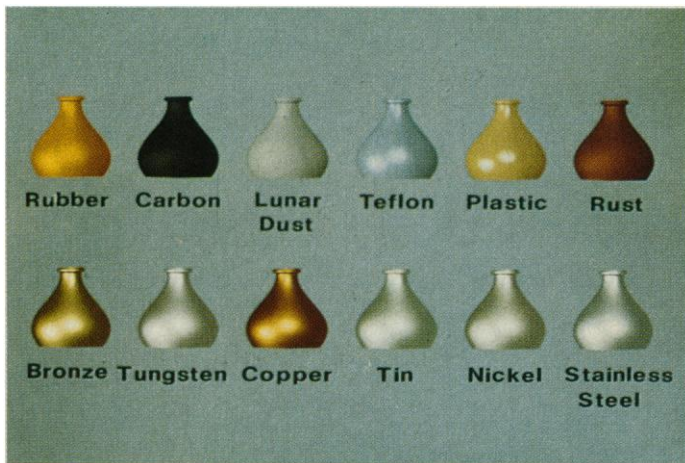


Fig. 4. Once the spectral reflectance curves are input into the computer's memory it is easy to display the visual results. This figure shows a sample of materials that exist in the material library. Image produced by Rob Cook.

predict the spatial and spectral distribution of the reflected light as a function of the physical properties of the incident and reflecting media and the geometric characteristics of the surface. They also must simulate the interreflections between surfaces within an environment, yielding the shading, shadows, and color bleeding necessary for photorealism. Although these algorithms currently require enormous amounts of computation, with the advent of custom chip technology and parallel processing, as well as algorithmic advances, they are rapidly becoming more tractable. Further computer graphics hardware will most certainly produce photorealistic images in real time.

Direct Illumination Models

Before describing the indirect illumination models of today, it is useful to review the historical precedents for the direct lighting algorithms. During the past 20 years, various computer models have been used for the simulation of the light reflection behavior. Early

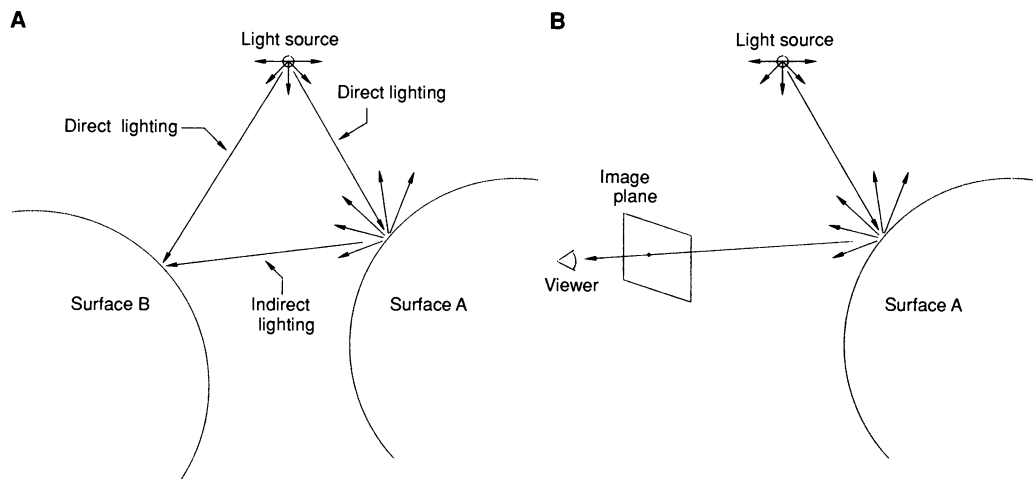


Fig. 5. (A) Direct and global illumination. Light can be thought of as consisting of rays emanating in all directions from each light source. Rays directly striking a surface provide direct illumination. Reflected or transmitted rays, or both, provide indirect illumination. (B) Light rays propagating from a source. Computations can be greatly simplified by only tracing those rays that reach the eye.

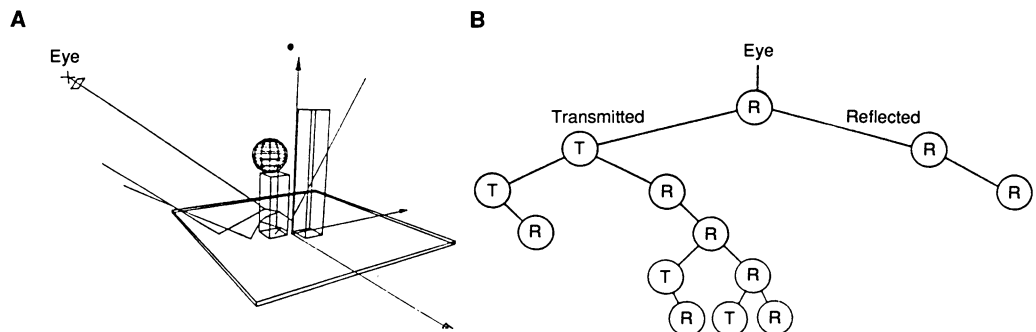


Fig. 6. (A) In ray tracing, a ray is traced from the eye through each pixel on the image plane into the environment. At each surface that is struck by the ray, a reflected or transmitted ray, or both, can be spawned. (B) As the ray is traced through the environment, an intersection tree consisting of branches (rays) and nodes (surface intersections) is constructed.

Fig. 7. Ray tracing images. (Left) Exterior simulation of an early design of the Performing Arts Center at Cornell University. Image produced by Phil Brock. (Right) Interior simulation of the John Soane Museum (breakfast room) with mirrored reflections and texturing. Image produced by Alan Polinsky.



models were obviously simplistic owing to the constraints of the computer hardware environments available. Limited memory and mass storage restricted the complexity of algorithms, but the dominant constraint was the length of the computation time. It was necessary to find fast numerical approximations of the complex physical behavior of light reflections so that solutions were computationally tractable.

In simple terms, reflected light can be thought of as consisting of three components: ambient light, diffuse light, and specular light. Ambient light originates from all directions, and is reflected back into the environment equally in all directions. Because of the presence of this scattered ambient light, one can discern objects even in a dark room at very low levels of illumination. Diffuse reflections originate from a particular light source direction but, because of the roughness of the surface, reflect light back equally in all directions. Reflections from a carpet or a matte finish are of this type. Specular reflections can be illustrated by the behavior of a narrow beam of light reflecting off of a highly polished mirror. The light originates from a particular direction, and reflects back primarily in a particular direction. Early simulations considered surfaces only illuminated directly from light sources and arbitrarily decomposed the complex reflection phenomena into these ambient, diffuse, and specular components.

Diffuse Reflections

The simplest reflection model used in computer graphics is one based on the reflection of light from a perfect diffuser (1). If a surface is perpendicular to the direction of the light, the amount of reflected light is a maximum (Fig. 1, A and B). As the surface turns more and more parallel with the direction of the light, the intensity of the light reaching the surface becomes less. When the surface is parallel to the direction of the light rays, no light reaches the surface. To prevent surfaces from completely disappearing, it is common to include a small amount of ambient reflection. For a perfect diffuser the reflected intensity is equal in all directions, and thus the viewer's position with respect to the surface orientation does not affect the intensity that is seen.

Mathematically, the dot product between the unit normal vector and the unit light vector ($\mathbf{N} \cdot \mathbf{L}$) yields the cosine of the angle between them. Thus, for diffuse reflection with a constant ambient term, the intensity of the diffusely reflected light can be expressed by:

$$I = I_a + k_d \sum_{i=1}^m \mathbf{N} \cdot \mathbf{L} \quad (1)$$

where I is the total intensity of the reflected light, I_a is the constant intensity due to ambient light, k_d is the surface coefficient of diffuse reflection, \mathbf{N} is the surface normal vector (unit length), \mathbf{L} is the light source vector of the i th light source (unit length), i is the light source index, and m is the number of light sources. The distribution of the diffusely reflected intensity is shown in Fig. 2A.

Specular Reflections

Specular reflected light bounces directly off the surface of an object without entering it. The angle of incidence is defined as the angle between the light ray striking the surface and the normal vector to the surface. The angle of reflection equals the angle of incidence. For a perfect mirror such a ray reflects off the surface at a specific angle and can be seen by the observer only if the eye is

located along the reflection direction. This implies that the determination of how much specularly reflected light an observer can see depends not only on the direction of the light vector and the normal vector of the surface but also on the sight vector, the direction in which the observer is looking (Fig. 1, C and D).

The reflected light vector indicates the direction of a light ray after it reflects off the surface of an object. How much of that light the eye can see is dependent on angle β . For smooth surfaces, the specularly reflected light is focused primarily along the reflection direction. For rough surfaces, the reflected light is more spread out. The distribution of specularly reflected light can be mathematically approximated by a cosine function raised to an exponential power. Higher exponents indicate a more mirror-like surface because the reflected light is more concentrated around the reflection direction. A lower exponent implies a greater spatial distribution of the reflected light and the specular component will be seen over a wide angle between the reflected ray vector and the sight vector. The combined effect is represented mathematically by:

$$I = I_a + k_d \sum_{i=1}^m \mathbf{N} \cdot \mathbf{L} + k_s \sum_{i=1}^m (\mathbf{R} \cdot \mathbf{V})^n \quad (2)$$

where k_s is the surface coefficient of specular reflection, \mathbf{R} is the unit vector of the maximum specular reflection direction, \mathbf{V} is the unit vector in the view direction, and n is the exponent that varies with the glossiness of the surface.

In this simple approximation, first proposed by Phong at the University of Utah (2), the color of the ambient and diffuse reflection is assumed to be the color of the object, and the color of the specular term is assumed to be the color of the light source. The results of the spatial distribution of this approximation are shown in Fig. 2B. The total reflected intensity is a linear combination of the diffuse reflection (including the ambient term) and the specular reflection.

The effect of varying the diffuse and specular model coefficients (3) is illustrated in Fig. 3. The ambient term I_a is held constant and k_s/k_d is varied for each column. As the magnitude of k_d is increased, the object color predominates. The exponent used for n (N_s in the figure) increases by row from top to bottom, depicting an increase in the glossiness of the surface.

Blinn (4) introduced an improved model that contained a more accurate function for the generation of specular highlights. His approach was based on earlier work and experimental measurements of light reflection by Torrance and Sparrow (5), whose semi-empirical model provided a better scientific basis for simulating reflections. The Blinn model differs from the Phong model in that the magnitude of the intensity, the spatial distribution, and the position of the specular highlight vary with the angle of incidence.

The analytical model assumes a surface to be composed of many small randomly oriented microfacets with each facet reflecting the incident light similar to a perfectly smooth mirror. Thus, only facets whose normal orientation is in the mirror direction contribute to the specular component of reflection. The resultant specular reflection distribution is a function of the angle between the mirror direction (defined to be the angular bisector between the light and view directions) and the normal to the surface, and is dependent on the combination of three factors. One factor represents the distribution of the microfacets and predicts the probability of a microfacet having the necessary orientation to reflect light from the source to the viewer. A second factor describes the amount by which the facets shadow and mask each other. The third factor is a function of the index of refraction of the material and calculates the ratio of reflected energy to the incident energy according to the Fresnel equations. The distribution is shown in Fig. 2C. The model correctly predicts

that the maximum of the specular function does not necessarily lie exactly along the reflection vector. These “off-specular peaks” are particularly noticeable at grazing angles. Blinn’s model thus provides a better match to experimental data than the previous formulations.

Cook and Torrance presented a more general model for rough surfaces (6). Although it is similar to previous models in that it is based on geometrical optics, it correctly produces the directional distribution and the spectral composition of the reflected light. In particular, the model produces the color shift that occurs in the reflected light as the angle of incidence changes. Because the model is based on an energy basis, it can also relate the brightness of an object to the intensity and size of the light source. The facet distribution function was derived from a wave-optics model originally developed by Beckmann (7) and applied to rough surfaces, those surfaces where the irregularities are considered to be large compared to the visible wavelengths of light. In the Cook-Torrance formulation, two or more distributions could be combined, thus simulating surfaces with two or more roughness scales.

For the first time, wavelength-dependent spectral energy distributions and reflectances were used. The spectral energy of the reflected light was found by multiplying the spectral energy distribution of the illuminant by the reflectance spectrum of the surface. This provided a more accurate simulation than one that only considers the red, green, and blue components. Furthermore, it is easy to store in the computer’s memory a library of spectral reflectances for common materials and assign these properties to any geometry (Fig. 4).

The specular highlight, which is a function of the surface material property, can also be computed more precisely. For a surface with known wavelength-dependent material characteristics, the Fresnel equations can predict the spectral composition as a function of wavelength and angle of incidence. At normal incidence, the color of reflected light is dependent on the reflectance spectrum. As the angle of incidence approaches the grazing angle, the composition of the reflected light approaches the color of the light source. The reflection of the sunlight off a newly polished car at sunset illustrates this behavior. The color of the reflected highlight is the color of the sun and is independent of the color of the automobile.

Recently, in the continual quest for more detail, the reflection models have been further extended to anisotropic materials (8) for which the reflection and refraction of light exhibit preferred directions. To date, however, the computational cost has precluded its popular usage.

Global Illumination Models

Although the approaches presented so far are quite sophisticated, results of computer simulations are still easily recognized as such. The primary reason is that with the exception of the constant ambient term, the computer simulations described do not consider the effect of the intra-environment surroundings. In real scenes, the lighting and reflections are far more complicated and subtle. Every surface receives light directly from light sources or indirectly from reflections off of neighboring surfaces. The indirect lighting is frequently called the “global illumination.” This phenomena is very difficult to model accurately, but for realistic image generation these global effects must be modeled in greater detail.

An integral equation, which accounts for the global illumination and perhaps is the most comprehensive description of light propagation appearing in the computer graphics literature, was presented by Kajiya (9). His generalized equation, called the “rendering equation,” considered the emissions from all light sources plus the interreflections from all surfaces in determining the light intensity at

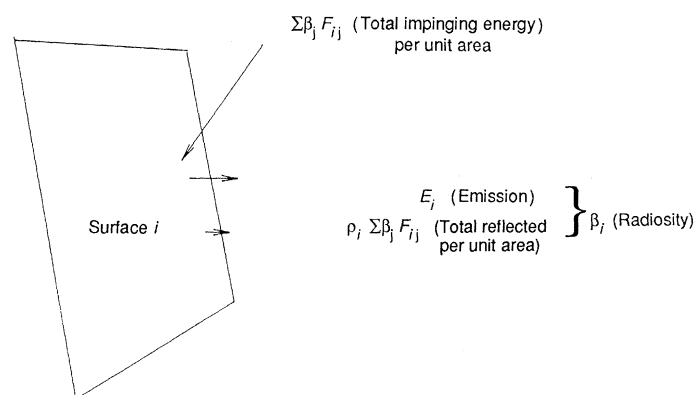


Fig. 8. Total radiosity. The light leaving a surface (its radiosity) consists of its self-emitted light, E , plus the light reflected from its surface. The amount of light reflected depends on ρ and the light arriving from all other sources or surfaces.

a surface. Arbitrary surface reflectance characteristics were assumed, and thus the equations were not restricted to diffuse or specular reflections. Both the direct and reflected illumination was modified by a geometry term that accounted for occluding surfaces and the dissipation of energy with distance. Kajiya’s proposed solution to the complex integral equations was a Monte Carlo approach, which although it extended the range of optical phenomena that could be simulated, was extremely time-consuming and thus, not practical.

During the past few years, two more tractable, but restrictive methods have become popular: ray tracing and radiosity. Both methods provide limited global illumination effects, but produce very realistic images. In ray tracing, which is particularly appropriate for specular environments, the discretization and sampling step occurs at the image plane, and the results are view dependent. In the radiosity approach, which is excellent for diffuse scenes, the environment is discretized, and the results are view independent.

Ray Tracing

One can conceptualize light propagation through an environment to consist of rays emanating in all directions from each light source. When these rays impinge on a surface, they are either absorbed, reflected, or transmitted. Each of the secondary reflected or transmitted rays can then act as an illuminating source (Fig. 5A). Therefore, any point on a surface may receive light, either directly from a light source or indirectly from a reflected or transmitted ray. The computations required to trace all of the rays from each light source and each surface in all directions in a complex environment are obviously too prohibitive. However, if one wishes to simulate the electromagnetic radiation as seen from a particular viewpoint, the process can be greatly simplified. Because it is only necessary to compute the intensity function on the image plane to render a picture, the procedure can be reversed. Only intensities of those rays reaching the eye need to be computed (Fig. 5B).

For computer-generated imagery, each image is composed of a series of pixels. The technique, basic to television displays, relies on the properties of the human visual and perceptual system to spatially and spectrally integrate the collection of dots to depict a continuous scene. The Pointillist painters of the Impressionist era relied on these same phenomena and accurately reproduced very subtle lighting conditions. By sending a ray through each pixel location into the environment and computing the intensity reaching the eye, the color of each pixel can be determined. Although the idea existed previous-

ly, the most popular solution to model the global illumination effects by ray tracing was presented by Turner Whitted (10).

In the simplest form of ray tracing, a single ray is traced from the eye through each pixel into the environment. At each surface that is struck by the ray, a reflected or refracted ray can be spawned (Fig. 6A). Each of these must be recursively traced to establish what surfaces they intersect. As the ray is traced through the environment, an intersection tree is constructed for each pixel (Fig. 6B). The intersection tree consists of branches that represent the propagation of the ray through the environment, and of nodes that represent the intersection of rays with surfaces in the environment.

The final pixel intensity is determined by traversing the tree and computing the intensity contribution of each intersected surface according to the reflection model. The intersection tree is traversed from the bottom up. The intensity information for the children nodes of the node currently being evaluated is retained, and the reflection model is applied to generate the intensity at the current node. Once the recursive tree traversal procedure has been executed, the final intensity for the pixel consists of the cumulative intensity contributions of all of the nodes of the tree. In this way, the global illumination effects of the environment are gathered and contained in the final image. By tracing rays, three stages of the standard graphics pipeline—the perspective transformations, the visible surface calculation, and the intensity determination—are combined in a single step. In a slightly modified form, the Whitted model is expressed as:

$$I = I_a + k_d \sum_{i=1}^m \mathbf{N} \cdot \mathbf{L} + k_s \sum_{i=1}^m (\mathbf{N} \cdot \mathbf{H})^n + k_s I_r + k_t I_t \quad (3)$$

where \mathbf{H} is the unit vector in mirror direction (halfway between the \mathbf{L} and \mathbf{V} vector), $k_s I_r$ is the global reflected term, and $k_t I_t$ is the global transmitted term (Fig. 2D).

The difference between the ray tracing model and previous models is the contribution of the reflected and transmitted rays in the specular direction. As with the previous models, the two summation terms represent the diffuse and specular reflections from direct lighting. The constant ambient term is also similar, but the global illumination, which accounts for the increased realism of the images, is represented by the addition of the global reflected and global transmitted terms.

The intensive computational operations in a ray tracing system are the intersection tree building and the “shadow testing” operations. To construct the intersection tree, each ray must be tested against every surface in the environment to determine the closest intersection point, and this must be done for each pixel on the image plane. To compute the intensity at each node on the tree, it must first be determined if that intersection point can be illuminated by a light source. This determination, called shadow testing, is accomplished by establishing a vector from the intersection point to each light source. If the vector intersects any opaque surface, then the intersection point is in shadow. Shadow testing, which must be performed for all intersection points for all light sources, is obviously very time consuming. In fact, for environments with complex lighting, shadow testing is frequently the dominant computational expense.

The results of ray tracing have been some of the most realistic images to date (Fig. 7). However, there are many scenes that cannot be adequately modeled by ray tracing, and there are several shortcomings to the method. First, the method combines the modeling of global illumination and the computation of the image plane intensity function into a single operation, thus requiring new calculations for every change of view. Second, the computational expense is already high, and despite the fact that algorithmic advances have accelerated the computations (11–14), new calculation methods

must be found to make the solutions more tractable. Third, the ray tracing method provides only point-sampled information, which causes some aliasing problems, and makes it difficult to simulate area light sources. Lastly, and most important, the reflection models used to date are empirical and do not account for the required energy equilibrium conditions. Some of these shortcomings can be eliminated by the use of multiple sample points for each pixel, or ray tracing with cones instead of point samples (15). One particularly effective method called distributed ray tracing (16) determines the directions of the rays according to the analytic function they sample, and thus can incorporate fuzzy phenomena such as motion blur, depth of field, and penumbras. However, despite the improved picture quality, the energy equilibrium conditions still cannot be satisfied with the ray tracing procedure.

Radiosity

As previously shown, ray tracing methods take into consideration the global illumination contribution by the addition of specularly reflected and transmitted terms in their respective directions only. The dissipation of intensity with distance cannot be accurately simulated with this point sampling procedure. Furthermore, ray tracing neglects the interaction of diffusely reflecting surfaces. In most environments, the object-to-object reflections between diffuse surfaces have a major influence on scene illumination. The majority of surfaces in a real environment are diffuse reflectors and, in general, specular reflections account for only a small proportion of the total reflected light energy.

A new procedure for image generation, based on methods in thermal engineering (17, 18) and applicable to environments composed of ideal diffuse emitters and reflectors, was developed by Goral *et al.* (19). This procedure, known as the radiosity method, determines surface intensities for diffuse environments independent of observer position.

Light leaving an object surface, its radiosity, originates from the surface by direct emission, as from a light source, or by the reflection of incident light (Fig. 8). The amount of light arriving at a surface (the incident light) comes from all other surfaces and lights within the environment. Thus, the amount of light arriving at a surface requires a complete specification of the geometric relations between all reflecting surfaces, as well as the amount of light leaving every other surface. This relation is expressed by:

$$\beta_i A_i = E_i A_i + \rho_i \sum_{j=1}^n \beta_j F_{ji} A_j \quad (4)$$

where β_i is the radiosity of surface i , E_i is the emissivity of surface i , A_i is the area of surface i , ρ_i is the reflectivity of surface i , β_j is the radiosity of surface j , F_{ji} is the fraction of the energy leaving surface j and landing on another surface i , A_j is the area of surface j , and n is the number of discrete surfaces.

The term F_{ji} is known as the “form-factor” and can be geometrically determined. However, a reciprocity relation exists between form-factors such that $F_{ij} A_i = F_{ji} A_j$. Therefore, by dividing through by the area A_i , the more familiar radiosity equation is obtained:

$$\beta_i = E_i + \rho_i \sum_{j=1}^n \beta_j F_{ij} \quad (5)$$

Assume the environment is subdivided into a set of small discrete surface elements or “patches,” each with a constant radiosity. The radiosity of each patch can be mathematically expressed as described.

The unknown quantities in the equation are the patch radiosities, since the radiosity of each surface depends on the radiosity of every other surface. However, since this same equation can be written for each patch, a set of simultaneous equations can be generated to describe the interaction of the diffuse light energy within the environment. The solution to this set of equations yields the correct radiosities and fully accounts for the global illumination within diffuse environments. Both the emissivities and reflectivities are wavelength-based functions and thus, the equations are only valid for a given wavelength interval, or "color band." For the generation of computer pictures, it is usually sufficient to use three color bands corresponding to red, green, and blue. The entire process can be described by the following set of sequential modules.

Input geometry. The environment geometry is described by a set of polygon descriptions, each with their appropriate vertex coordinates. Associated with each polygon are reflectivity and emission values for each color band and a parameter for subdividing the polygon into patches.

Form-factors. After the patches are defined by subdividing the polygons, the form-factors from each patch to every other patch are calculated. The form-factor, which specifies the fraction of the energy leaving one surface that is received by the other, is dependent only on the geometric relation between the two surfaces. Variables include the shape, area, and orientation of each patch, the distance between them, and the portion of each patch visible to the other. By taking into account occluding surfaces, complex environments can be accurately modeled (20).

Radiosity solutions. For each color band, a matrix of simultaneous equations is constructed with the form-factors and the appropriate set of reflectivities. The corresponding emission values are then used to solve the equations for patch radiosities.

Rendering. An eye position, view direction, and frustum angle are specified from which an image is rendered. For each pixel, the location of the view ray-patch intersection is computed. The color, or colors, are found through a bilinear interpolation of the known vertex color values.

Display. Once the color of each pixel has been computed, the results are transformed to digital output for each of the red, green, and blue electron guns of a television monitor. If the chromaticity coordinates of the phosphors are known, accurate renditions can be obtained with the use of color science principles.

The introduction of the radiosity method has led to a complete decoupling of the light reflection simulation from the final image rendering. The illumination calculations are independent of viewing parameters and can be performed on an individual wavelength basis or on any number of independent wavelength bands. Form-factors need be computed only once for static environments. If the lighting conditions remain constant, the radiosity solution is also valid for any viewing position. Thus the environmental intensity information can be preprocessed and subsequently used for multiple views. Since only the rendering process has to be repeated for each image, and this part of the standard graphics pipeline is now executed in hardware on many graphics workstations, dynamic sequences can be displayed (Fig. 9).

Furthermore, the inclusion of all elements of the environment in calculating the global illumination effects yields more accurate solutions than those previously achieved. The phenomena of "color bleeding" from one surface to another, variable shading within shadow envelopes, the effect of area light sources, and penumbra effects along shadow boundaries can all be reproduced. However, there are several disadvantages to the radiosity approach. Because of the large computational cost in computing the form-factors, the method is practically limited to static environments, ones in which the geometric relation between surfaces remains constant. As pre-

sented, the approach is also limited to purely diffuse scenes. Lastly, the solution is only as accurate as the discretization of the environment.

During the past few years, the limitations of the radiosity algorithms have been substantially extended. To reduce the computational expense without sacrificing image quality, a two-level adaptive subdivision approach was formulated (21). First, a coarse patch solution, in which the patches act as light sources and reflectors, is obtained. Then the patches themselves are further subdivided into elements. The elements act as receivers of the light from the coarse patch solution and provide the sample values for the final rendering process. The advantage is that the element subdivision can be continued adaptively as high radiosity gradients are discovered without recomputing the patch radiosities. Effectively, this means that the picture quality improves by sampling the environment in areas where the changes are the greatest.

The restriction of complete diffusivity has also been removed. Although the approach is not tractable because the number of equations becomes so large, different reflection functions have been incorporated into the radiosity solution allowing for specular reflections and "reflection tracking" (22). The addition of a post-processing step incorporated specularly more efficiently (23) (Fig. 10, right), and the standard radiosity solution was further extended to include the effects of scattering due to a participating media (24) (Fig. 10, left).

However, most importantly, the adaptive procedures mentioned previously reveal another important phenomenon, that is, a coarse solution that generates a reasonable approximation of incoming energy combined with a more detailed, finer solution to generate outgoing intensities can produce excellent results for the modeling of global illumination effects. Carried a step further, the incoming energy can be estimated by the use of only the patches providing the largest energy to the entire environment. Thus, rather than solving a complete set of simultaneous equations, an iterative approach that considers the contribution of one patch at a time can be used. This approach monotonically converges on the correct solution, but interim results can now be displayed as the solution is progressively refined (25).

With this technique, one can obtain images that are almost correct (98 percent) at speeds two orders of magnitude faster than with standard radiosity approaches. Furthermore, the enormous memory requirements are also reduced, allowing the simulation of very complex environments. This progressive refinement technique almost immediately provides a useful solution that progresses gracefully and continuously to the complete radiosity solution.

Conclusion

The ray tracing and radiosity approaches demonstrate two methods for producing realistic images, but both are only restrictive solutions to the integral equations required to simulate global illumination (9). The best way to obtain the correct solution is to use Monte Carlo simulations, ones that can incorporate different reflection functions and motion phenomena similar to the distributed ray tracing, but these are computationally intractable. Perhaps a more scientific and comprehensive basis for combining the benefits of ray tracing and radiosity may prove worthwhile.

Despite the quality of the images shown, the simulation methods presented are not sufficient. It is still difficult to model the attenuation or the scattering effects of light as it travels through a medium. Environments with changing geometry or topology are difficult to compute. Foliage, texture, water, clouds, and many natural phenomena cannot easily be handled with current simulation technology.

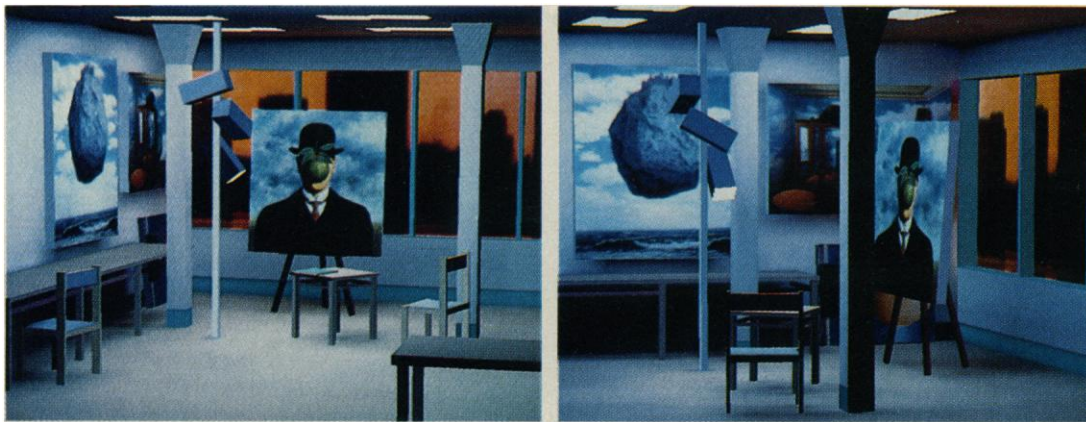
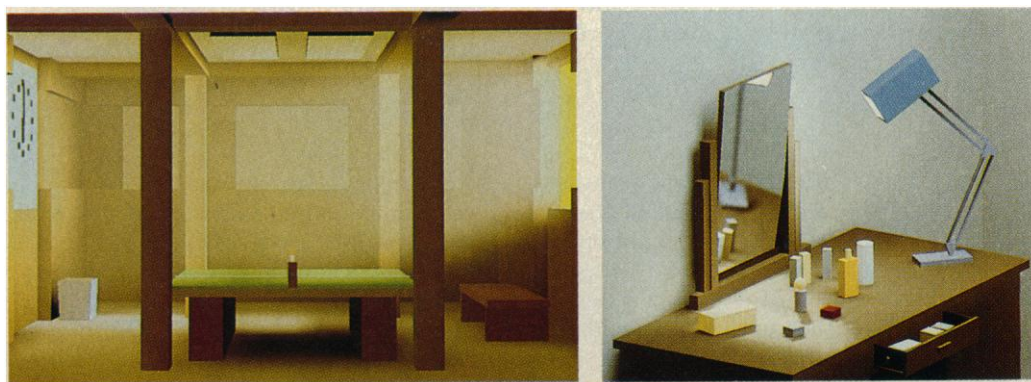


Fig. 9. (Left) This image of an artist's studio was generated by means of the radiosity approach and shows soft shadows and subtle shading on the surfaces. The picture took approximately 5 hours to compute on a mini-computer (DEC VAX 11/780), including the calculation of the form-factors, the radiosity solution, and the rendering. (Right) By changing the observer's position, another view can be easily obtained. Since only the rendering stage had to be repeated, the image was generated in software in 15 min. As the rendering software migrates into hardware, images of this quality can be dynamically generated. Images produced by Michael Cohen.

Fig. 10. (Left) By discretizing not only the surfaces but the enclosed volume, the radiosity method can be generalized to account for the emission, scattering, and absorption by a participating medium, such as the dispersion of light in a smoky room. Image produced by Holly Rushmeier. (Right) The radiosity algorithms have been extended to include the specular to diffuse energy. The illumination of the top of the vanity along with the objects on it results primarily from the direct emission from the lamp as well as the specular reflection from the mirror. Image produced on an HP 320 SRX by John Wallace.



gy except in very special cases. Better and more accurate light reflection models, those that are physically and energy based and can model arbitrary spectral and spatial distributions, need to be developed. The entire process must be made computationally faster to provide dynamic capability and to enhance our opportunities for scientific and design exploration. We would like to visualize experiments that have not been conducted, perform noninvasive diagnostic techniques for medicine, and walk through environments that have not yet been built. Because graphics will become the dominant form of communication between humans and computers, our quest for realism will continue.

REFERENCES AND NOTES

1. H. Gouraud, *IEEE Trans. Graphics C-20* (no. 5), 623 (1971).
2. B. T. Phong, *Commun. ACM*, **18**, 311 (1975).
3. R. A. Hall and D. P. Greenberg, *IEEE Comput. Graphics Appl.* **3** (no. 8), 10 (1983).
4. J. F. Blinn, *Comput. Graphics ACM SIGGRAPH 1977 Proceedings* **11** (no. 2), 192 (1977).
5. K. E. Torrance and E. M. Sparrow, *J. Opt. Soc. Am.* **57**, 1105 (1967).
6. R. L. Cook and K. E. Torrance, *Comput. Graphics ACM SIGGRAPH 1981 Proceedings* **15** (no. 3), 307 (1981).
7. P. Beckmann and A. Spizzichina, *The Scattering of Electromagnetic Waves from Rough Surfaces* (Macmillan, New York, 1963).
8. J. T. Kajiya, *Comput. Graphics ACM SIGGRAPH 1985 Proceedings* **19** (no. 3), 15 (1985).
9. ———, *Comput. Graphics ACM SIGGRAPH 1986 Proceedings* **20** (no. 4), 143 (1986).
10. T. Whitted, *Commun. ACM* **23**, 343 (1980).
11. S. M. Rubin and T. Whitted, *Comput. Graphics ACM SIGGRAPH 1980 Proceedings* **14** (no. 3), 110 (1980).
12. H. Weghorst, G. Hooper, D. P. Greenberg, *ACM Trans. Graphics* **3**, 52 (1984).
13. E. A. Haines and D. P. Greenberg, *IEEE Comput. Graphics Appl.* **6** (no. 9), 6 (1986).
14. T. Kay and J. T. Kajiya, *Comput. Graphics ACM SIGGRAPH 1986 Proceedings* **20** (no. 4), 269 (1986).
15. J. Amanatides, *Comput. Graphics, ACM SIGGRAPH 1984 Proceedings*, **18** (no. 3), 129 (1984).
16. R. L. Cook, T. Porter, L. Carpenter, *ibid.*, p. 137.
17. E. M. Sparrow and R. D. Cess, *Radiation Heat Transfer* (McGraw-Hill, New York, 1980).
18. R. Siegel and J. R. Howell, *Thermal Radiation Heat Transfer* (McGraw-Hill, New York, 1980).
19. C. Goral, K. E. Torrance, D. P. Greenberg, B. Battaile, *Comput. Graphics ACM SIGGRAPH 1984 Proceedings* **18** (no. 3), 213 (1984).
20. M. F. Cohen and D. P. Greenberg, *Comput. Graphics ACM SIGGRAPH 1985 Proceedings* **19** (no. 2), 311 (1985).
21. ———, D. S. Immel, P. J. Brock, *IEEE Comput. Graphics Appl.* **6** (no. 2), 26 (1986).
22. D. S. Immel, M. F. Cohen, D. P. Greenberg, *Comput. Graphics ACM SIGGRAPH 1986 Proceedings* **20** (No. 4), 133 (1986).
23. J. R. Wallace, M. F. Cohen, D. P. Greenberg, *Comput. Graphics ACM SIGGRAPH 1987 Proceedings* **21** (no. 4), 311 (1987).
24. H. E. Rushmeier and K. E. Torrance, *ibid.*, p. 293.
25. M. F. Cohen, S. E. Chen, J. R. Wallace, D. P. Greenberg, *Comput. Graphics ACM SIGGRAPH 1988 Proceedings* **22** (no. 4), 75 (1988).
26. Much of the work has been produced at Cornell University's Program of Computer Graphics. The author acknowledges the support from NSF, and the contribution of the Digital Equipment Corporation and Hewlett-Packard. Thanks go to my many students, past and present, who performed most of the work depicted.