Monte Carlo Methods and the Challenge of Photo-Realism in Computer Graphics

Steven Collins*
Image Synthesis Group, Trinity College Dublin

Abstract

Computer Graphics have come a long way since the pioneering work of Ivan Sutherland in the early 60's and his SketchPad software developed during his postgraduate research studies at MIT. This system used a vector-based Cathode Ray Tube with a prototype "light-pen" device with which users of the system could draw line based images on the display.

In more recent times, computer graphics are not simply an interface between human and computer but a tool for visualisation, expression and communication. There have been many goals for the computer graphics industry and academic community: to simplify the interface to the computer; to allow manipulation and visualisation of data in a meaningful and insightful manner; to facilitate medical exploration of the human body; to provide faster drawing speeds for real-time applications including simulators and games.

The dominant objective, however, has remained the attainment of realism. If we have truly realistic computer graphics we can create digital actors and replace their live counterparts for particularly dangerous or difficult movie sequences. We can produce images of product designs with absolute faith that the image is a completely accurate and faithful representation of a manufactured version.

Realism has a price though. It is probably fair to say that we understand quite well the mechanics of image formation, optics and reflectance (at the very least for certain classes of scenes and environments). There are tremendous problems associated with acquiring accurate input data, however, and the final ingredient, the simulation, still presents tough challenges. Each image we perceive is the result of countless billions of interactions between photosensitive retinal cells and incident light energy. This energy that arrives at the eye has almost certainly had an interesting journey from its source experiencing many scattering events at the interfaces between media in the scene and within the media themselves.

Solution strategies have involved the use of both Finite Element and Monte Carlo Methods. We will examine how such methods may be employed in realistic image creation, assess their current application and suggest areas of future research and development.

*Email: Steven.Collins@cs.tcd.ie, Web: http://isg.cs.tcd.ie/scollins/

1 Introduction

Computer graphics are all pervasive, in today's information age, as a means of expression and communication. From the desktop to the virtual surgery computer graphics have revolutionised the way in which we interact with computers, information and each other. One goal of computer graphics is to compute synthetic images of real and imaginary scenes which are entirely realistic and are convincing enough to be used to validate industrial designs and to fool cinema audiences that the actor on the big screen really is made of mercury and slipping between the bars of his jail-cell. This paper will outline current research into photo-realism, otherwise known as realistic image synthesis, the basic ingredient of which is global illumination (the simulation of light transport in arbitrary scenes).

To a large extent the physics of light transport and light interation with surfaces and media is well known. We can draw on a huge body of research in areas from neutron transport to thermal radiation heat transfer [SH92] to radar theory and apply many of the techniques to solve the problem of visible light transport in scenes. However, the particular challenges of simulating light transport to produce images of photo-realistic[1] quality require new techniques in order to achieve acceptable quality and efficiency. We can exploit, for example, the independence of quanta of light energy as they interact with the scene[2]. We can ignore the wave nature of light considering only its particulate nature and finally we can assume the existence of a steady state solution with no non-linearities[3].

The nature of the required solution is quite different from many other disciplines. We are not interested in a set of measurements to be displayed using typical visualisation software. We require an image, or more precisely, a rectalinear array of samples of the *plenoptic function*[4] [AB91]. To compute an image we specify a view (position, orientation, size of sensor and direction of view) which guides the simulation phase. Given the view, we sample the plenoptic function at a sufficiently high density to reconstruct an accurate image of the scene. Each sample may not be expensive to compute relative to other disciplines, but the number of samples required for even the most modest of views is high and the complexity of the scene is usually much higher than in other applications. For example, the recent Disney film "Toy Story" included scenes which were modelled using in excess of 100 million polygonal elements [Apo98]. This introduces the final requirement: most designers/artists are not willing to wait very long for an image (typically less than 2 hours) particularly when a sequence of images are being generated ("Toy Story" required over 100 thousand) for animation purposes.

An image is formed when light energy arrives at the retina and is interpreted by the brain or when light energy strikes the photosensitive sensor in a camera. This light began its journey at a light source and probably experienced a complex series of scattering events alongs its journey altering the frequency, phase and amplitude of the light. To compute an image we must model the sensor, the light source and the physics

[1] By photo-realistic we mean that the image evokes a visual respose similar to that evoked by a photograph of the scene being simulated.

[2] All phase effects can be ignored at a macroscopic level; we are interested in phenomena that exist at magnitudes many orders greater than the wavelength of light.

[3] Although virtually all light sources have time variance due either to imperfections in the mechanics or to fluctuations in the power supply, the human visual system is particularly good at filtering these effects out. Therefore we can consider the average state of the system over a finite period. We will assume that the lights have been switched on already and the scene has reached equilibrium

[4] The *plenoptic punction* records the distribution of light in all directions flowing through scene. It is a function of position $\mathbf{x}$ and direction (θ, ϕ).

of light interation with surfaces and media in the scene.

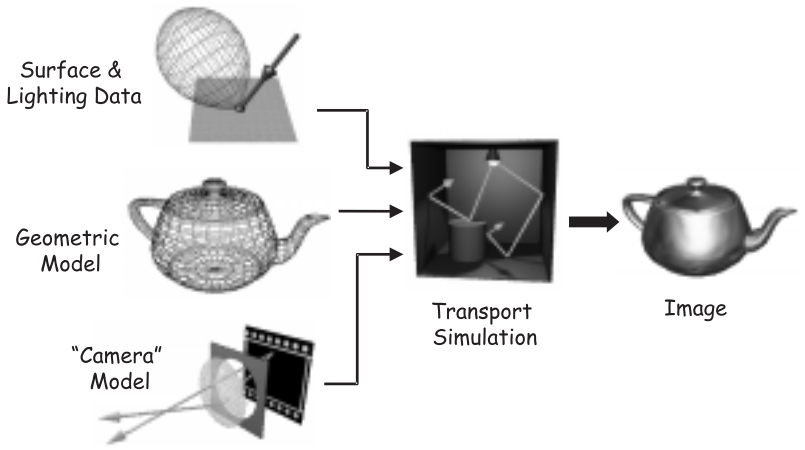


Figure 1: Requirements for realistic image synthesis.

Figure 1 illustrates the requirements for realistic image synthesis.

1.1 Geometry

We begin with an accurate geometric model of the scene to be rendered. Models are usually hybrid constructions of polygons, implicit surfaces, density functions, piecewise cubic manifolds or volumetric elements. Implicit in transport simulation is the ability to intersect lines with each of these elements to determine points of potential scattering events along the paths of light energy quanta (which we will call *photons*, though in this case "photon" is simply a convenient term for a finite quantity of light flux which, in the absense of a scattering event, travels linearly with infinite speed, and does not imply a quantum physics approach). The model allows us to establish regions in the scene through which photons will not undergo scattering. Usually in these regions we assume an isotropic (and transparent) medium. Photons travel from surface to surface, being scattered at each interface and terminate only when the associated flux drops below some threshold (i.e. the photon has been absorbed). If the photon arrives at the sensor it contributes to the image being formed. Various algorithms require a more restricted geometry (see, for example, the radiosity method outlined in Section 5.1). A vital component of the geometry, though often treated separately, is the light source (or *luminaire*). In more recent work, light sources are becoming increasingly complex, with non-isotropic emission characteristics, varied geometries and high spectral variation (required for the treatment of flourescent sources) [VG84, LT92].

1.2 Surface Data

The characteristics of each surface in terms of how that surface is perceived under various lighting conditions depends entirely on the manner in which the surface scatters incident light radiation. The selective absorption of light wavelengths gives rise to the colour of the surface. We are very good at determining the micro-scale geometry of the surface from the manner in which the scattering behaviour varies with direction (this allows us to distinguish between silk and cotton, even though they may be the same

colour). Surface scattering is quantified by the BSDF (bidirectional scattering distribution function) defined as a spherical density function. In the following discussion we will consider only opaque surfaces and thus the BSDF reduces to the hemispherical BRDF (bidirectional reflectance distribution function). The modelling, storage and application of BRDF data will be dealt with in detail in Section 3.

1.3 Viewing System

We need to define a viewing system in order to create an image. In general if we have a complete solution for the plenoptic function, our view is simply a 2D slice through this 5D function. For each point on the viewing sensor (which will normally be assumed to be a rectalinear quadrilateral) we determine the direction(s) that light can arrive through and sample the plenoptic function accordingly. This pre-supposes the existence of a full solution which will rarely be available. Instead, we adopt one of two approaches:

View dependent simulation: rather than compute the entire 5D plenoptic function, we evaluate it only on the 2D slice that is the sensor or viewing region. This is an importance sampling approach and involves either rejecting photons that do not arrive at the sensor or tracking photons in reverse beginning at the sensor. Both approaches will be discussed in section 5.2.

View independent simulation: view independent schemes make no simplifying assumptions regarding the view from which the image will be taken. Rather a more complete representation of the plenoptic function is simulated. In order to be tractable, we usually employ either a volumetric approach (the function is approximated as a lattice with accurate values at the nodes) or a surface based approach, where we assume non-scattering media between surfaces. In this latter case, we evalutate the lighting at nodes on the surfaces, requiring a mesh representation of the surface elements. We rarely compute the complete 5D sample even at these points, instead we adopt simplifying assumptions regarding the reflectance of the surfaces (for example, by assuming no dependence on direction). The *Radiosity* method employs such techniques and is discussed in Section 5.1. In such cases, the simulation provides a simple compact representation of the plenoptic function which is later queried by a viewing system in order to compute a view using simple projective geometry.

1.4 Simulation

The final ingredient is the simulation. Depending on the application there is a rich variety of rendering algorithms to choose from. A coarse taxonomy follows:

Projective Methods: early graphics algorithms computed images by projection of polygonal elements onto a viewing plane. These projections were then coloured according to a simple approximation of the lighting on the polygon, usually assuming lighting directly from light sources alone and therefore did not account for intermediate scattering. The greatest difficulty is in the determination of visibility and occlusion. These methods are employed in all real-time graphics hardware epitomised by the Silicon Graphics systems.

Finite Element Methods: the first example of the use of FE technology in computer graphics was work carried out by Goral et al. in 1984 [GTGB84]. Since then, the

Radiosity Method has been one of the most active areas of graphics research. The method requires a mesh geometry and expresses the global illumination problem as a linear system to be solved using traditional methods (including Gauss-Seidel, Southwell-relaxation schemes and more recently Wavelet methods). This method, as mentioned earlier, does not compute a single view, but rather evaluates the illumination at nodes in the scene only. The method is currently only applicable in scenes exhibiting highly coherent reflectance.

Monte Carlo Methods: since 1980 [Whi80], algorithms similar in nature to *random walk* MC simulations have been employed in computer graphics. The *ray tracing* algorithm is an example of such a system and is currently one of the most popular of all rendering algorithms. MC methods track the paths of photons as they interact with scene elements, probabalistically assigning new directions and energies based on the scattering characteristics of the material encountered. These methods scale extremely well to higher dimensional problems but exhibit error in the form of noise which can be problematic for applications requiring extremely high fidelity imagery. For example, the film industry demands noise-free images but are willing to accept other forms of error (e.g. consistently over-smooth illumination, sharp shadows rather than penumbrae).

Image Based Methods: recently, image based rendering (IBR) methods [GGSC96, LH96] have become a popular solution to the problem of geometry acquisition and fast lighting. Rather than computing images, we take real-world imagery (i.e. scanned photographs) and create new views of the scenes depicted in these images by extrapolation, interpolation and reconstruction. This is currently one of the most active areas of computer graphics research.

In this paper we are mostly concerned with Monte Carlo methods but will outline in more detail the Finite Element approach as a means of comparison. The projective and IBR approaches will be largely ignored as neither is concerned with the accurate simulation of light transport.

2 Radiometry and Photometry

In computer graphics, we are interested in the spatial and directional distribution of light flux. Instantaneous light energy is of no interest to us given our assumption of a steady state solution, so it is *power*[5] (with units of $Watts$ and denoted Φ) that we measure and simulate. Power will usually vary with position on a surface or within a volume. This distribution is described by the *radiosity*[6] ($B = \frac{d\Phi_{out}}{dA}$) or exitant power per unit area (with units $Watts\ m^{-2}$) or *irradiance* ($E = \frac{d\Phi_{in}}{dA}$) with similar units which expresses incident power per area. Finally, to express directional variation we employ *radiance* ($L = \frac{d^2\Phi}{dA\cos\theta d\omega}$) which expresses the power per unit projected area per unit solid angle (with units of $Watts\ m^{-2}\ sr^{-1}$). A geometric interpretation of radiance is illustrated in Figure 2. The spectral dependence of all the above terms is assumed but omitted for the purposes of clarity.

These are radiometric terms and as such include the entire electro-magetic spectrum as domain. Computer graphics involve visual results and therefore we can ignore

[5] Power and flux are used here interchangeably.

[6] Radiosity is more formally known as *radiant exitance*, but we will adopt the more common term for convenience.

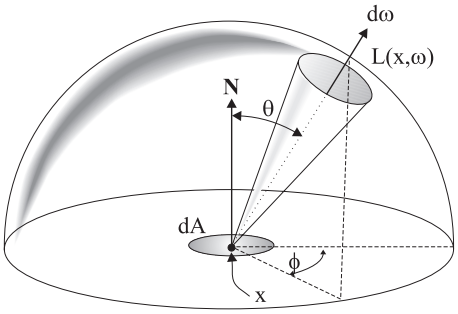


Figure 2: Geometry of the definition of radiance $L = \frac{d^2\Phi}{dA\cos\theta\, d\omega}$.

all wavelengths outside the visible range (approximately $380nm$ to $780nm$). The human visual system is not uniformly sensitive within the range however. In particular, we are very poor at descriminating differences in blues and relatively good when differentiating greens. When we talk about *brightness* which is entirely a psycho-physical sensation it is not sufficient to express the radiance, radiosity or power of a source. The science of *photometry* is the measurement of the visual response of average human observers to radiometric quantities. Implicit in photometry is the notion of an "average observer model"[7]. Associated with model is the luminous efficiency function $V(\lambda)$ which quantifies the sensitivity of the human visual system to all wavelengths of light. Thus, given the spectral power function of a source ($\Phi(\lambda)$) we can easily determine the apparent brightness (or luminous power P) of the source:

$$P = \int_{380}^{780} V(\lambda)\Phi(\lambda)\, d\lambda \tag{1}$$

As with the radiometric measures, the luminous power (units $Lumens$) can be quantified according to its spatial and directional distributions (luminous power per area = luminance, and luminous power per area per direction = luminosity). When computing an image for display, it is important to assume that the display device is not capable of reproducing the dynamic range of brightness present in the real scene, therefore it is necessary to account for the effect of viewing this dynamic range and computing an image which conveys the correct perceptual response.

2.1 The Radiance Equation

The radiance equation (Equation 2 and illustrated in Figure 2.1(a)) introduced in 1986 by Kajiya[8] [Kaj86], is fundamental to global illumination and it is this that all rendering algorithms seek to solve or approximate.

We are interested in the radiance scattered by a surface towards the viewer which may be expressed as

[7]The foundation of all photometric research is the work of the CIE (Commission Internationale d'Eclairge) in 1932 [WS82] which tabulated the visual characteristics of a sample population of visually unimpaired subjects. This led to the establishment of a standard perceptually-based colour metric and the *standard observer model*.

[8]In Kajiya's paper, he uses the term "rendering equation" and expresses this in terms of intensity. Since then a more rigorous formulation employing radiance has been adopted by the rendering community.

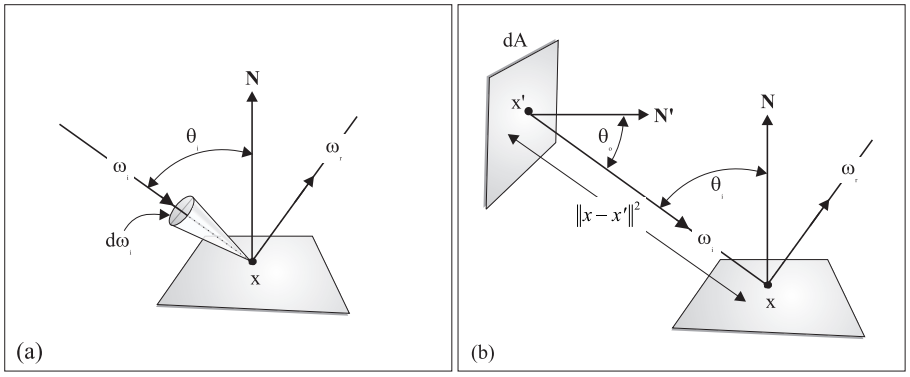


Figure 3: Geometry of the radiance equation. (a) illustrates an integration over all directions, whereas (b) illustrates a surface integral approach.

$$L_r(x,\omega_r) = L_e(x,\omega_r) + \int_\Omega f_r(x,\omega_i,\omega_r) L_i(x,\omega_i) \cos\theta_i \, d\omega \tag{2}$$

This accounts both for the radiance emitted in the direction ω_r by the surface at x, L_e, and the radiance L_i which has arrived at x via directions ω_i and scattered according to the BRDF f_r. This is an example of a Fredholm equation of the second kind. The domain of integration, Ω is hemispherical if the surface at x is opaque and spherical otherwise. Note the recursive but linear nature of the integral. The *driving term* L_e is non-zero for luminaires only.

The luminaires will tend to have smooth emission distributions and the BRDF is often approximated by spherical functions which are also highly smooth. However the implicit visibility term (in order to determine L_i, the incoming radiance, we must determine the surface visible in direction $-\omega_i$ from x) introduces the complexity. Alternatively we can make explicit this dependence on visibility by re-formulating the radiance equation as a surface integral [SW91], in which case we must relate the differential solid angle measure $d\omega$ to the differential area measure $dA(x)$. This is illustrated in Figure 2.1(b).

$$L_r(x,\omega_r) = L_e(x,\omega_r) + \int_A f_r(x,\omega_i,\omega_r) L_i(x,\omega_i) \cos\theta_i \, \frac{dA(x)\cos\theta_o}{|x-x'|^2} \tag{3}$$

Analytic expressions exist for certain simple scene geometries (e.g. a flat plane illuminated by spherical or polygonal source), but as geometric complexity increases (beyond 4 polygons in fact), analytic solutions become non-trivial and quickly become non-existent. Thus we must rely on numerical methods to solve such problems.

3 Modelling Reflectance

Before exploring numerical methods for solving the radiance equation we will outline the role of the BRDF in global illumination and discuss methods for aquiring, storing and implementing reflectance data. The BRDF (illustrated in Figure 3) is formally

defined as the ratio of reflected radiance to the incident differential irradiance and is expressed as

$$f_r(x, \omega_i, \omega_r) = \frac{dL_r(x, \omega_r)}{L_i(x, \omega_i) \cos\theta_i d\omega_i} \tag{4}$$

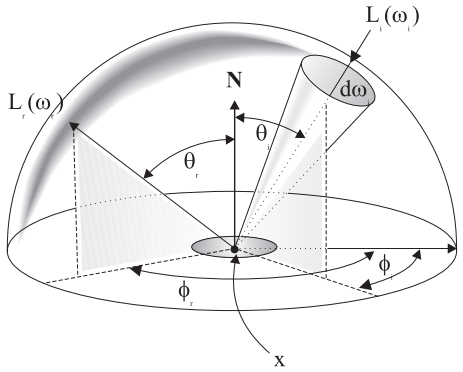


Figure 4: Geometry of the BRDF.

Note how the radiance equation is constructed by multiplying across the denominator of the BRDF ratio, including an emission term and integrating. The BRDF is a density function ($f_r \geq 0$ always) and thus directly applicable for importance sampling in Monte Carlo integration (refer to Section 5.2.2 for further details).

The simplest BRDFs are those of *ideal diffuse* and *ideal specular* surfaces. We use the term "ideal" to indicate that the surfaces do not simply exhibit diffuse or specular characteristics but are entirely diffuse or specular. Diffuse surfaces exhibit coherent reflectance. The BRDF for such surfaces is constant for all outgoing directions, therefore the BRDF may be replaced with a constant diffuse reflectance value, ρ_d, related to the diffuse BRDF by $\rho_d = \pi f_{r,d}$. It is no longer necessary to quantify radiance for such surfaces, as the radiance does not vary with viewing angle, so instead we use radiosity. To determine reflected radiance for a diffuse surface we scale the irradiance E by ρ_d/π (see [CW93] for further details).

$$\begin{aligned} L_{r,d}(x, \omega_r) &= \int_\Omega f_{r,d}(x, \omega_i, \omega_r) L_i(x, \omega_i) \cos\theta_i d\omega_i \\ &= \frac{\rho_d}{\pi} \int_\Omega L_i(x, \omega_i) \cos\theta_i d\omega_i \\ &= \frac{\rho_d}{\pi} E \end{aligned} \tag{5}$$

A BRDF which exhibits no coherence in reflection is that of an ideal specular surface, a good example of which is an ideal mirror. In this case, the reflected radiance depends entirely on the viewing direction and will vary considerably with changes in the viewing angles. An ideal specular BRDF is non-zero for a single direction only (the optically reflected direction) and is thus characterised by a delta function. Both the diffuse and specular BRDFs are illustrated in Figure 3.

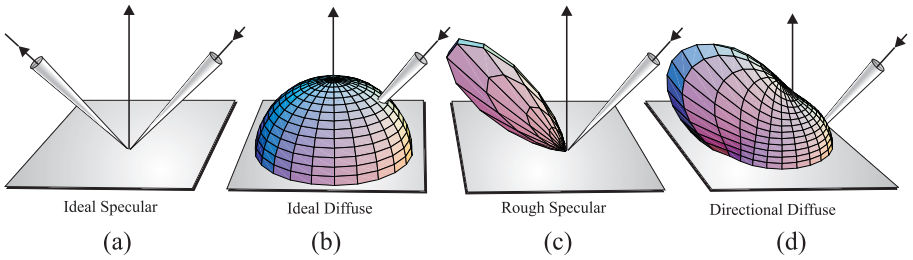


Figure 5: Hemispherical graphs of various idealised BRDFs. In all cases the incoming angle is fixed and the BRDFs are assumed to be isotropic.

Many real-world BRDFs lie somewhere between these two extremes. Two examples of less idealised BRDFs are shown in Figure 3. Many BRDFs however are significantly more complex and exhibit dependence on wavelength, incoming elevational *and* azimuthal angles and position (possibly even time). To model such BRDFs we have a number of options. We can attempt to measure real BRDFs using a *gonioreflectometer* and thereafter interpolate for parameters not captured directly by measurement [War92]. Alternatively we can derive an analytic model of surface reflectance based either on empirical observation (the phenomenological approach) [Pho75] or derive an analytic model for the behaviour of some simple micro-geometry [HTSG91]. A final approach involves a *virtual gonioreflectometer* which uses a Monte Carlo simulation to build a BRDF by firing photons at a geometric model of some micro-surface structure and recording the scattered directions of these photons in a directional data-structure [WAT92].

3.1 Empirical Methods

A good example of an empirical BRDF model is the *Phong Model*. This model is based on the observation that many specular surfaces (plastics in particular) have BRDFs that may be approximated by a linear combination of a constant diffuse term and a rough specular term (sometimes known as a glossy term) modelled as a $\cos^n$ lobe. The complete model is

$$f_r(x, \theta_i, \omega_r) = \frac{\rho_d}{\pi} + \rho_s \frac{n+2}{2\pi} \cos^n \alpha \tag{6}$$

The geometric interpretation of the model is given in Figure 3.1 and some samples of what is possible with the model are given in Figure 3.1. The BRDF has a maximum in the optically reflected direction. The reflectance drops as the angle α between this direction and the outgoing direction ω_r increases. The numerical simplicity of this model makes it very popular for rendering algorithms and so a large proportion of all computer graphic imagery employs the Phong model or a derivation thereof.

In many cases it transpires that the simplicity of the BRDF model is actually masked by the complexity of the radiance distributions and many of the most realistic images produced to date have employed this simple model.

Other empirical models, including that of Ward [War92] actually measure surface samples to determine their reflectances and attempt to fit this data to a function which is sufficiently general to allow a good approximation of the measurements. In the *Ward model*, an elliptical Gaussian is chosen as a basis.

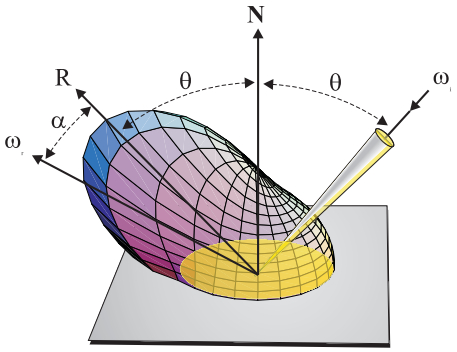


Figure 6: The geometry of the Phong BRDF model.

Figure 7: Some sample BRDFs using the Phong model.

3.2 Virtual Gonioreflectometry

A novel approach to the problem of BRDF data acquisition is to construct a geometric model of the micro-surface structure [WAT92] of the surface for which a BRDF is required (this model is typically of the order of millimeters in cross-sectional area). Many surfaces (for example silk, cotton, brushed metal) have well-known micro-geometry. Once this geometry is constructed, a large number of photons are fired at it and tracked as they interact with the micro-surfaces. Usually, we assume a simple BRDF model for the elements making up this micro-geometry[9] which allows us to simulate the scattering in a probabilistic manner. Once tracked, we determine the directions with which the photons leave the target geometry (if at all) and tabulate for all in-going directions the resulting out-going directions. Some sample micro-geometries are illustrated in Figure 3.2.

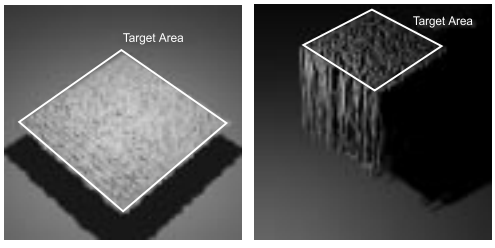


Figure 8: 2 micro-scale geometries for common surface materials.

[9]Salt crystals, for example, are entirely transparent when magnified, however when viewed from a distance they take on the appearance of a white powder, due entirely to the scattering of light through and off the salt crystals. At the micro-scopic level we might model the BRDF of salt as if it were an ideal specular transmitter like glass.

Storage of this data is another difficult problem. The BRDF can have dimension of greater than 5 depending on the application, and so some form of compression is required (particularly given that regions of the BRDF data-set can be both sparse and highly symmetric/coherent). We have successfully applied a feed-forward Neural Network to this problem and have reported significant compression ratios with efficient query times during light transport simulation [GN98].

4 Analytic Reflectance

In some cases, the reflectance of the surface is well-known. If the surfaces can be assumed to be optically smooth and are ideal conductors or dielectrics (for example gold, copper and glass) then we can directly employ the *Fresnel equations* which are solutions to Maxwell's wave equations for planar waves at the interface between media of different refractive indices. By employing these equations directly we can accurately capture the appearance of glasses and metals which is of particular importance in industrial design visualisation. Examples of analytic reflectance data of some materials is given in Figure 4. This data is the result of applying the Fresnel equations to the material's frequency dependent refractive index function.

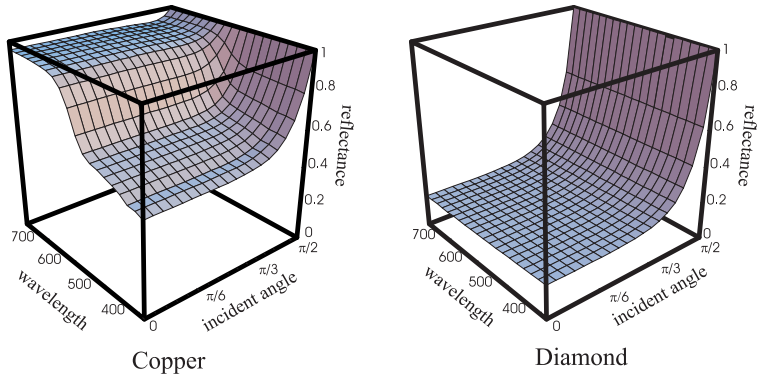


Figure 9: Fresnel reflectance for 2 materials.

5 Numerical Methods

The solution of the radiance equation for complex scenes is a non-trivial problem and one which has received a lot of attention in the computer graphics research community since 1980. Over the years since then two major families of algorithms have developed: the Finite Element based methods and the Monte Carlo methods. The choice of algorithm depends very much on the final application. For example, in the case of interior lighting design, a diffuse reflectance model will often suffice so a Finite Element approach will usually produce the most desirable results. When visualising industrial designs with varied surface types and possibly significant spectral variation (as in the case of automobile paint or crystal glass) Monte Carlo approaches are more appropriate. Each algorithm family has its strengths and weaknesses. Whereas the FE methods produce smooth images with soft lighting they do not scale well and in particular are poor at handling specular surfaces and discontinuous flux distributions. Monte Carlo

methods are more general, can adapt to most surface geometries and reflectances but are very slow to converge and can suffer from serious noise artifacts.

5.1 Finite Element Methods

In order to employ FE methodologies, it is first necessary to simplify the radiance equation in order to reduce it to a finite linear system of equations. This involves a number of simplifying assumptions:

1. All surfaces are diffuse, therefore we can drop the BRDF f_r in favour of the simpler reflectance ρ_d. We can also work directly with radiosity rather than radiance as all the surfaces exhibit no reflectance variation with respect to direction.

2. The scene is composed entirely of surface elements, usually planar, over which the radiosity is assumed to vary in a known manner (often this is linear or cubic).

Given these assumptions we can reduce the radiance equation to the simpler discrete *radiosity equation* which is expressed as

$$B_i = E_i + \rho_i \sum_{j=1}^{n} B_j F_{ij} \tag{7}$$

$$\begin{bmatrix} 1-\rho_1 F_{11} & \dots & -\rho_1 F_{1n} \\ \vdots & & \vdots \\ -\rho_n F_{n1} & \dots & 1-\rho_1 F_{nn} \end{bmatrix} \begin{bmatrix} B_1 \\ \vdots \\ B_n \end{bmatrix} = \begin{bmatrix} E_1 \\ \vdots \\ E_n \end{bmatrix} \tag{8}$$

Equation 8 summarises the linear system of n equations resulting from a scene with n elements. B_i and ρ_i are the radiosity and reflectance respectively of patch i, E_i is the emitted radiosity of patch i and F_{ij} is the form factor from patch i to patch j which accounts for the geometric relationship between patches i and j and thus gives the ratio of the radiosity leaving patch j which arrives at patch i.

To solve this system we can employ many different matrix methods. The matrix is *diagonally dominant* and can be made symmetric via a simple transformation and so iterative schemes converge rapidly. Currently the most popular method is an adapted form of *Southwell relaxation* known as the *Progressive Refinement Radiosity Method* [CCWG88]. More recently, clustering algorithms[10] [RPV93] and wavelet methods[11] [GSCH93] are being used to accelerate the radiosity simulation with minimal error.

Appropriate meshing is vital in order to effectively capture important features in the radiosity solution. The most commonly used approach is an adaptive meshing refinement scheme, usually employing a quadtree decomposition on each element, with further subdivision being flagged by an *oracle* which tracks the local error. An a-priori approach, that of *discontinuity meshing* attempts to construct a mesh in advance of simulation which tracks the gradients of the illumination, ideally matching element edges with discontinuities in the radiosity function. Although it appears that the solution to the radiosity system is a requirement for such meshing, in practice the most important discontinuities are those arising from direct lighting. It is therefore possible to determine the limits of umbral and penumbral given an area light source, a planar receiver and blocker and these limits are used to align mesh elements.

[10] If elements are densely distributed in certain regions of the scene, these elements may be *clustered* and treated as a single meta-element when evaluating radiosity exchange with a distant element.

[11] Wavelet methods exploit smoothness in the radiosity kernel to reduce the number of element interactions in regions where the radiosity is slowly varying whilst preserving discontinuous regions.

5.2 Monte Carlo Methods

Realistic image synthesis in complex environments requires sampling over many domains. Consider the spectral radiance arriving at a point on the sensor of a camera. Light arrives at this point x from all points on the lens assembly. Therefore we must sample all points on the lens u. This radiance varies with wavelength requiring that samples λ be drawn from the visible wavelength domain. The radiance arriving at the lens in any particular direction may vary with time if objects are in motion. Thus we must sample the time dimension t to account for this behaviour. Finally, the radiance leaving each point in the scene in the direction of the lens depends on the radiance arriving at those points from all possible scattering directions. We are now sampling from a domain of dimension 8. It is precisely for this type of problem that Monte Carlo methods [KW86] are highly applicable.

In practise, most Monte Carlo algorithms for global illumination solutions are *random walks*. Before discussing these methods we will summarise Monte Carlo integration. Given a real-valued function $f(u)$, we wish to integrate this over domain Ω:

$$F = \int_{\Omega} f(u) d\mu(u) \quad f : \Omega \rightarrow \mathcal{R} \tag{9}$$

We will draw samples x from Ω with probability density function $p(x) : \Omega \rightarrow \mathcal{R}^{+}$. A primary estimator for F is

$$F \approx \frac{f(x)}{p(x)} \tag{10}$$

A secondary estimator is

$$F = \int_{\Omega} \frac{f(x)}{p(x)} p(x) d\mu(x) = E\left[\frac{f(x)}{p(x)}\right] \approx \frac{1}{n} \sum_{i=1}^{n} \frac{f(x_i)}{p(x_i)} \tag{11}$$

The error in this estimator is quantified by the *variance* which is defined as

$$\sigma^2 = \text{var}\left[\frac{1}{n} \sum_{i=1}^{n} \frac{f(x_i)}{p(x_i)}\right] = \frac{1}{n} \text{var}\left[\frac{f(x)}{p(x)}\right] \tag{12}$$

As can be seen, the variance decreases with n but in fact the real situation is worse than this. It is σ which actually corresponds to visual error in our images and as a result we must quadruple the number of samples n in order to reduce the error by half. In the limit, $n \rightarrow \infty$, we will converge to the exact solution. In practice this convergence is asymptotic and we must stop after a certain threshold error level has been attained. Setting this threshold is more often than not a matter of trial and error.

5.2.1 Random Walks

To formulate the radiance equation as a random walk, we first recall its recursive form:

$$L(x) = L_e(x) + \int_{\Omega} \kappa(x, x') L(x') d\mu(x) \tag{13}$$

The kernel $\kappa(x, x')$ will account for all geometric terms involved in the scattering of a particle x in new direction x', thus the cosine term disappears. Essentially, a photon, represented by sample x_i, upon hitting a surface will be scattered by that surface in a new random direction and will lose energy. If this photon initially carried radiance $L(x_i)$, then the scattered photon radiance $L(x_{i+1})$ is given by

$$L(x_{i+1}) = L(x_i)\frac{\kappa(x_i, x_{i+1})}{p(x_{i+1})} \tag{14}$$

In this case, the radiance is the old radiance scaled by the kernel (given the randomly selected scattering direction) divided by the probability of chosing that random direction $p_1(x_1)$. We can apply this basic technique to express the radiance equation as a *Markov Chain*:

$$\begin{aligned} E[L(x_0)] &= L_e(x_0) + \frac{\kappa(x_0, x_1)}{p_1(x_1)L(x_1)} && (15)\\ &= L_e(x_0) + \frac{\kappa(x_0, x_1)}{p_1(x_1)}\left[L_e(x_1) + \frac{\kappa(x_1, x_2)}{p_2(x_2)L(x_2)}\right] && (16)\\ &= L_e(x_0) + \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}\frac{\kappa(x_{j-1}, x_j)}{p_j(x_j)}\right]L_e(x_i) && (17) \end{aligned}$$

This is a *gathering* approach where we evaluate the illumination arriving at the sensor by recursively tracking a path back through the scene (in effect, we are tracking the photon back in time to determine its origin). In this way we are guaranteed to track only those paths which will contribute to the final image. This approach is known as *path tracing* [Kaj86]. A geometric interpretation of this is shown in Figure 5.2.1(a). The problem with this brute-force approach is that we are not guaranteed to hit a light source before the error threshold is reached (and in practice we terminate paths long before this, usually according to a user-specified maximum recursive depth). An alternative approach is to begin at the light sources and emit photons into the scene, tracking their paths until they hit the sensor. Again, this is problematic, as a large proportion of the paths will never reach the sensor. This is a *shooting* approach and is known as *particle tracing* [PM92] in graphics research literature.

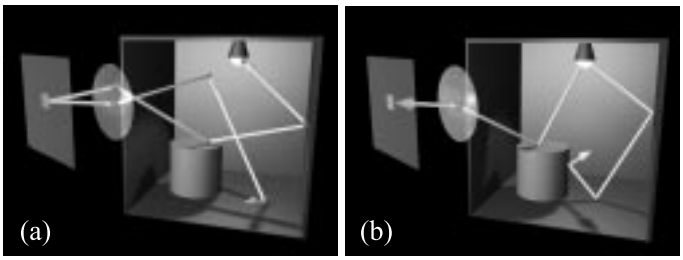
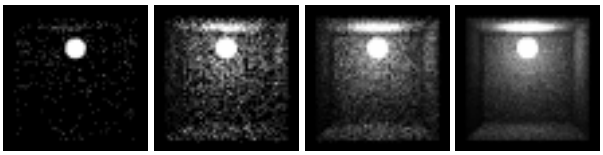


Figure 10: Random walk strategies. In (a) we track photons back in time from sensor to source. Some of these paths may never reach a light source. In (b) we track photons from the light source to the sensor. A large number of these paths will never actually hit the sensor.

Even for the simplest of scenes, as shown in Figure 5.2.1, a very large number of paths must be simulated in order to compute an image that is recognisable (but still far from noise-free).



Paths/pixel	1	16	64	400
Time (seconds)	14	89	306	1871

Figure 11: Results of path tracing a simple box scene with a single spherical isotropic light source and ideal diffuse BRDFs. Note that the number of samples quoted is on a per pixel basis. Each image is a 200×200 array of pixels thus for the scene with 400 samples/pixel a total of 16 million paths have been traced.

Simply increasing the number of samples will not provide a sufficient improvement in image quality. We must therefore examine other methods for reducing the variance.

5.2.2 Variance Reduction

A large number of variance reduction techniques exist and most of these have been applied to path tracing for computer graphics at some time or another. The most popular strategies are

- Stratified sampling
- Low-discrepance sampling (otherwise known as *Quasi-Monte Carlo*)
- Importance sampling
- Control Variates
- Next event estimation

Importance sampling is probably the single most effective way to reduce variance if some information about the domain is known. The basic idea behind this approach is to sample more densely in regions where the function being integrated has large magnitude. In this way we attach most importance to regions of the integrand that have most influence over the final outcome. Knowledge of the function however presupposes that we have a solution (the classic "chicken and egg" scenario). In theory we can achieve zero variance by chosing the ideal probability density function:

$$p(x) = \frac{f(x)}{\int_\Omega f(x) d\mu(x)} \tag{18}$$

In practice we want $p(x)$ to be high when $|f(x)|$ is high. To achieve this we can factor out known elements of $f(x)$ and sample according to these. Examples of this include the BRDF or the lightsource emission characteristics. If we sample with $p(x) = k f_r(x, \omega_i, \omega_r)$ then our new scattering directions will be randomly distributed

around areas of high reflectance. This has no effect for diffuse surfaces, but specular or glossy surfaces are sampled far more successfully using this approach.

In order to implement importance sampling using the BRDF as p.d.f. we must first normalise the BRDF and then find the inverse of the associate cumulative distribution. More formally if we wish have a function $g(u)$ that we wish to use for importance sample, we first construct the cumulative distribution $G(x) = \int_{u=0}^{x} g(u)d\mu(u)$. To draw samples distributed according to $G(x)$ we transform a canonical uniform random variable ξ by $x = G^{-1}(\xi)$. Note that the cumulative distribution function is monotonic increasing and therefore an inverse will always exist. Finding the inverse of arbitrary BRDFs or light source emission characteristics, however, is often a difficult problem and may not admit an analytic form requiring an expensive numerical inversion approach. Frequently, we simplify even further and draw samples from functions which approximate the shape of the functions we really want to sample.

For example, consider the Phong model BRDF of Equation 6. The diffuse and specular components are separable and therefore the simplest approach is to draw two samples, one from each distribution (or probabilistically decide in advance which component to sample). Therefore we sample the diffuse component with p.d.f. $p_d(\omega_i) = \frac{\cos\theta_i}{\pi}$ and the specular component using $p_s(\omega_i) = \frac{n+1}{2\pi}\cos^n\alpha_i$.

A second method is to use *next event estimation*. This technique attempts to partition the integral into sub-domain with each of these domains having an associated importance. An excellent example is that of *direct light sampling* [SWZ96]. At a given position x we are required to estimate the radiance leaving x in a certain direction ω_r given by $L_r(x, \omega_r)$. Rather than sampling the hemisphere uniformly, or simply importance sampling according to the BRDF we may observe that the most important directions are likely to be those that point in the directions of the light sources (i.e. these directions will probably contribut most to the final solution). By missing these important directions, the variance of our estimator increases. The next event estimation approach involves sampling the directions to the light sources separately from the hemispherical sampling scheme and weighting the samples accordingly to eliminate bias[12]. Figure 5.2.2 illustrates this partitioning of the integral.

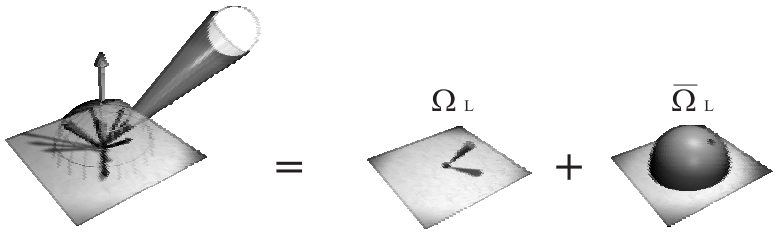


Figure 12: Normally the domain of integration for lighting calculations is the hemisphere, with associated solid angle measure $\Omega = 2\pi$. We can partition this domain into Ω_L, the solid angle subtended by a light source and $\bar{\Omega_L}$ the remaining solid angle. Samples drawn from each distribution are weighted according to the solid angle subtended.

To sample the set of directions not pointing towards the light sources we can employ rejection sampling as the solid angles of the light sources will typically be small.

[12]Bias is the error due to convergence to the incorrect solution and is usually due to attaching too much importance to certain directions.

Sampling according to light sources requires examination of the kernel for direct lighting. In some cases we can distribute samples uniformly across the surface of the light source geometry. In other cases we will samply uniformly within the solid angle subtended by the light source. Some results of various sampling strategies for spherical light sources are given in Figure 5.2.2. These results draw heavily from the work of Pete Shirley et al. [SWZ96].

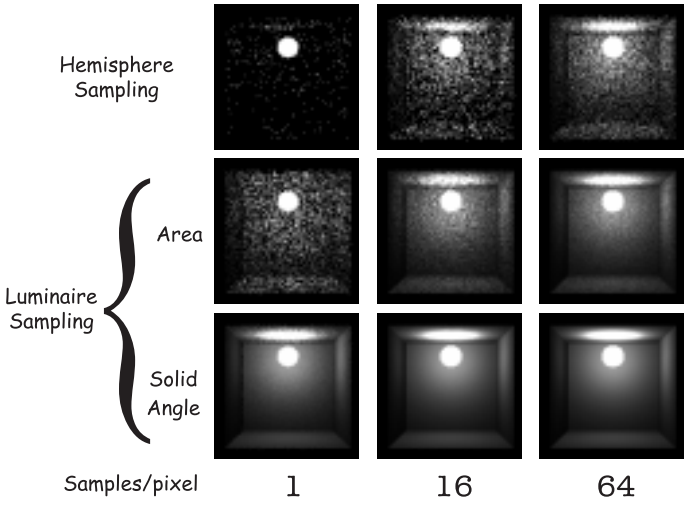


Figure 13: In the first row, we employ no variance reduction techniques and simply sample the hemisphere of directions uniformly ($p(\omega) = \frac{1}{2\pi}$). The second row demonstrates the improvement when directly sampling the spherical light source. In this case we sample uniformly over the surface of the sphere, $p(\omega) = \frac{1}{A_L}$. We still have high variance due to rejection of sample points on rear facing regions of the light source. The final row employs uniform sampling within the solid angle subtended by the sphere, $p(\omega) = \frac{1}{\Omega_L}$. For very little extra computation we dramatically reduce the variance in the solution.

5.3 Multi-pass Methods

Attempting to solve for all possible modes of light transport quickly becomes an extremely computationally expensive problem. In many cases, we are more interested in certain modes and will happily sacrifice accuracy in order modes. A good example of this is the visualisation of crystal glass. To produce an image of crystal glass we are rarely concerned with the full global illumination solution. Rather we wish to compute a highly accurate solution locally around the glass. We can focus our efforts by employing a *multi-pass method*. Such methods decouple various modes of light transport, simulate each mode separately and then reconstruct a solution based on the combined results of the various passes. To simplify the solution we neglect certain paths. In the case of crystal glass visualisation we can neglect the transfer of energy between diffuse surfaces and consider only the energy due to light interaction with specular surfaces. This will give a good approximation to the crystal glass design. If we wish to place this design in an environement (a display cabinet for example), we can introduce further modes. If we are interested in the complex pattern of caustics produced by light being

refracted through the facets of the glass we will simulate a specular to diffuse transport mode and add this to the final solution. This 2-pass scheme is illustrated in Figure 5.3.

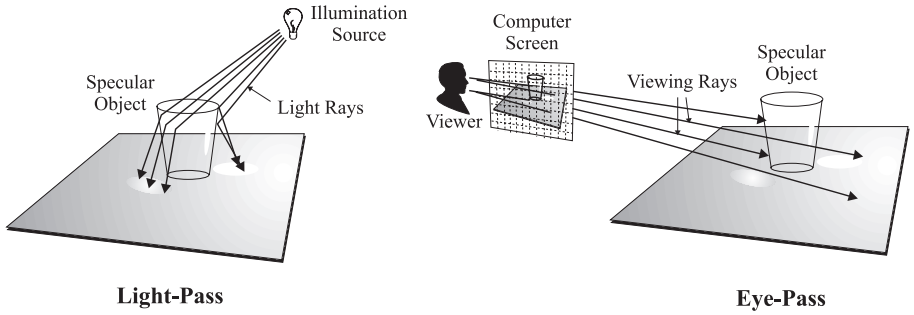


Figure 14: In this example of a 2-pass solution, the first pass uses a shooting approach to distributed photons into the scene. As these photons interact with surfaces their energies are stored in illumination maps attached to the surfaces. In the second pass we perform a gather and pick up the energy deposited in the illumination maps to construct a view of the scene.

One of the main problem with such multi-pass methods is the storage and retrieval of partial solutions from each mode. We have investigated the use of *illumination maps* (uniform meshes recording irradiance on diffuse surfaces) and *density estimation*[13] for the accurate reconstruction of caustic effects [Col96].

5.4 Sampling the Camera Model

To improve the apparent realism of the image we can construct an accurate model of the sensor (either as a camera system or the human eye). This increases the parameter space within which we must sample, but allows us to reproduce effects including motion blur, depth of field and vignetting. To implement depth of field we must model the camera lens using, at the very least, a thin lens approximation[14]. We sample the lens (usually uniformly) by chosing points on the lens and firing rays from here through the focal point and out into the scene. Objects positioned off the focal plane will appear blurred. Figure 5.4 illustrates the geometry of the thin-len approximation and shows the results of applying this model within a global illumination framework.

6 Conclusions and Future Work

There are many avenues for future research. Certainly the fact the global illumination has yet to make its impact on TV and film graphics illustrates that current techniques fall short of addressing the needs of the industry. Elimination of noise is paramount to the future success of these techniques. This will probably require the introduction of biased filtering techniques to consistently remove noise from the images, though exactly how this may be done is a matter for future research. Many current algorithms are brittle and require simple geometries and reflectances. The challenge to the research

[13] Density estimation is a statistical technique for reconstruction of a function from a sparse non-uniformly distributed set of sample points.

[14] To simulate non-linear lens artifacts including coma and astigmatic effects we must implement a thich lens approximation.

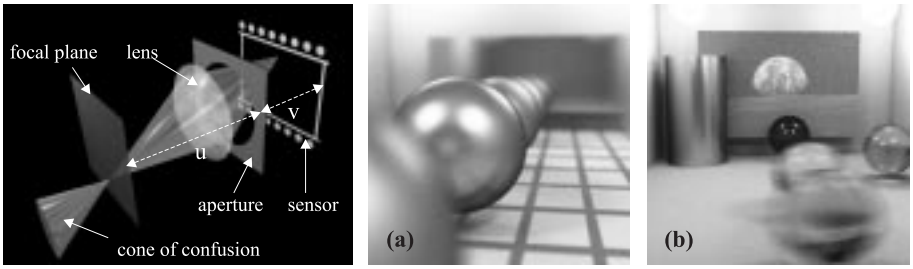


Figure 15: The geometry of the thin lens camera model is shown on the left. The thin lens approximation with focal length f establishes a focal plane at distance u from the lens given v, the distance from the sensor to the lens, according to the well-known formula $\frac{1}{u} + \frac{1}{v} = \frac{1}{f}$. In image (a) we see the result of this model. The focal planes lies at the centre of the 2nd sphere. In image (b) we sample the time domain while pivoting the camera about the sphere at the rear of the room. Note the motion blur which increases with speed of object motion. Both images are full path traced solutions and took approximately 6 hours on a Pentium 100 class PC.

community is to facilitate global illumination solutions in scenes involving very large data-sets (100 to 1000 million polygons) and highly accurate BRDF data.

As with most numerical systems, an obvious choice for improving performance is to utilise a parallel system. Much research has already been conducted into the effective use of parallel systems for computer graphics, examining caching and distributed memory issues as well as dynamic load-balancing and problem partitioning schemes. Each particular algorithm requires special attention with the implemention of parallel FE approaches being substantially different from the implementation of MC algorithms. Of prime importance, as always, is the minimisation of inter-node communication during simulation, and assuming that the scene will never fit on a single processing element this necessitates a distributed memory implementation. The challenge is to implement such a distribution scheme that can maximise the work that each node can perform independently of other nodes.

We hope to examine such schemes and in particular assess the applicability of lazy evaluation and delayed computation approaches in conjunction with compression schemes to reduce the inter-node bandwidth.

Acknowledgements

The work described in this paper has been the result of research carried out by a number of researchers within the Image Synthesis Group, Department of Computer Science, Trinity College Dublin. Thanks also to Pete Shirley and Eric Veach for permission to use their images in the presentation accompanying this paper. Most of this work has been sponsored by Forbairt grant ST/96/106 and is supported by Waterford Crystal. We are delighted that future work on parallel rendering is to be supported jointly by Forbairt grant ST/98/001 and the Hitachi Dublin Laboratory.

References

[AB91] E. H. Adelson and J.R. Bergen. *The plenoptic function and the elements of early vision*, chapter 1. The MIT Press, Cambridge, Mass., 1991.

[Apo98] Tom Apodaca. Photosurrealism. In *Proceedings of the 9th Eurographics Workshop on Rendering, Vienna*, pages 140–152, 1998.

[CCWG88] M. F. Cohen, S. E. Chen, J. R. Wallace, and D. P. Greenberg. A progressive refinement approach to fast radiosity image generation. In *Computer Graphics (ACM SIGGRAPH 88 Proceedings)*, pages 75–84, 1988.

[Col96] S. Collins. *Wavefront Tracking for Global Illumination Solutions*. PhD thesis, Trinity College Dublin, Ireland, 1996.

[CW93] M. F. Cohen and J. R. Wallace. *Radiosity and Realistic Image Synthesis*, chapter 2. Academic Press Professional, 1993.

[GGSC96] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. In *Computer Graphics (ACM SIGGRAPH '96 Proceedings)*, pages 43–54, 1996.

[GN98] D. G. Gargan and F. Neelamkavil. Approximating reflectance functions using neural networks. In *Proceedings of the 9th Eurographics Workshop on Rendering, Vienna, Austria*, pages 23–34, 1998.

[GSCH93] S. J. Gortler, P. Schr oder, M. F. Cohen, and P. M. Hanrahan. Wavelet radiosity. In *Computer Graphics (ACM SIGGRAPH '93 Proceedings)*, 1993.

[GTGB84] C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile. Modelling the interaction of light between diffuse surfaces. In *Computer Graphics (ACM SIGGRAPH '84 Proceedings)*, pages 212–222, 1984.

[HTSG91] X. D. He, K. E. Torrance, F. X. Sillion, and D. P. Greenberg. A comprehensive physical model for light reflection. In *Computer Graphics (ACM SIGGRAPH 91 Proceedings)*, pages 175–186, 1991.

[Kaj86] J. T. Kajiya. The rendering equation. In *Computer Graphics (ACM SIGGRAPH '86 Proceedings)*, pages 143–150, 1986.

[KW86] M. H. Kalos and P. A. Whitlock. *Monte Carlo Methods*. Hemisphere Publishing Corporation, 3rd edition, 1986.

[LH96] M. Levoy and P. Hanrahan. Light field rendering. In *Computer Graphics (ACM SIGGRAPH '96 Proceedings)*, pages 31–42, 1996.

[LT92] E. Languenou and P Tellier. Including physical light sources and daylight in global illumination. In *Proceedings of the 3rd Eurographics Workshop on Rendering, Bristol, UK*, pages 217–225, 1992.

[Pho75] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.

[PM92] S. N. Pattanaik and S. P. Mudur. Computation of global illumination by monte carlo simulation of the particle model of light. In *Proceedings of the 3rd Eurographics Workshop on Rendering, Bristol, UK*, pages 71–83, May 1992.

[RPV93] H. Rushmeier, C. Patterson, and A. Veerasamy. Geometric simplification for indirection illumination calculations. In *Proceedings of Graphics Interface '93, Toronto, Canada*, 1993.

[SH92] R. Siegel and J. R. Howell. *Thermal Radiation Heat Transfer*. Hemisphere Publishing Corporation, 3rd edition, 1992.

[SW91] P. Shirley and C. Wang. Direct lighting calculation by monte carlo integration. In *Proceedings of the 2nd Eurographics Workshop on Rendering, Barcelona, Spain*, 1991.

[SWZ96] P. Shirley, C. Y. Wang, and K. Zimmerman. Monte carlo techniques for direct lighting calculations. *ACM Trans. on Graphics*, 15(1):1–36, 1996.

[VG84] C. P. Verbeck and D. P. Greenberg. A comprehensive light-source description for computer graphics. *IEEE Computer Graphics and Applications*, 4(7):66–75, 1984.

[War92] G. J. Ward. Measuring and modelling anisotropic reflection. In *Computer Graphics (ACM SIGGRAPH 92 Proceedings)*, pages 265–272, 1992.

[WAT92] S. H. Westin, J. R. Arvo, and K. E. Torrance. Predicting reflectance functions from complex surfaces. In *Computer Graphics (ACM SIGGRAPH 92 Proceedings)*, pages 255–264, 1992.

[Whi80] T. Whitted. An improved illumination model for shaded display. *Communications of the ACM*, 23(6):343–349, 1980.

[WS82] G. Wszyecki and W. S. Stiles. *Colour Science: concepts and methods, quantatitive data and formulae*. Wiley, New York, 2nd edition, 1982.