Eurographics Symposium on Rendering (2007)
Jan Kautz and Sumanta Pattanaik (Editors)

Rendering Discrete Random Media Using Precomputed Scattering Solutions

Jonathan T. Moon, Bruce Walter, and Stephen R. Marschner

Department of Computer Science and Program of Computer Graphics, Cornell University

Abstract

This paper addresses light transport through a discrete random medium, which we define as a volume filled with macroscopic scattering geometry generated by a random process. This formulation is more general than standard radiative transport, because it can be applied to media that are made up of closely packed scatterers. A new approach to rendering these media is introduced, based on precomputed solutions to a local multiple scattering problem, including a new algorithm for generating paths through random media that moves through the interior of the medium in large strides without considering individual scattering events. A method for rendering homogeneous isotropic random media is described that generates paths using precomputed scattering solutions compressed and randomly sampled using Nonnegative Matrix Factorization. It can efficiently render discrete media, such as a large pile of glass objects, in which the individual scatterers are visible. The method is demonstrated on scenes containing tens of thousands of transparent, specular objects that are nearly impossible to render with standard global illumination techniques.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism

1. Introduction

This paper addresses the problem of rendering complex, random three-dimensional transparent structures. A motivating example for our work is hair, which is made up of many fibers that are visible and therefore need to be modeled individually, but which exhibit smooth large-scale behavior due to multiple scattering. Accurate multiple scattering solutions can be computed by tracing rays through the geometry itself [MM06], but a statistical method for multiple scattering has the potential to be much more efficient. In this paper we take the first steps toward this goal by developing methods for a class of similar but non-oriented materials.

Many other scenes have similar properties of visible structure but smooth large-scale behavior: air-liquid foams, such as soap foam; crystalline materials, such as sugar or salt; a pile of glass beads or other transparent objects; a forest canopy made up of randomly distributed leaves (see Figure 1 for an example). We refer to these structures as *discrete random media*, a term borrowed from the optical literature [TDSK98]. When specular reflection dominates, these scenes are essentially impossible to render using established global illumination techniques. For our purposes a discrete random medium is a volume filled with macroscopic scattering geometry that is generated by a random process: the specific arrangement of the objects is not important for multiple scattering, and we can instead look for the average behavior over the distribution of geometries that could be produced by the generating process.

Discrete random media differ from the continuous media commonly simulated in graphics. In continuous media, light is assumed to interact with infinitesimal scattering particles. This ensures that scattering events along a path are statistically independent. In a discrete random medium, scatterers are often packed close together, which invalidates the independence assumptions underlying the standard formulation of volumetric light transport.

This paper introduces a new approach to analyzing and simulating light transport in homogeneous, isotropic random media, which relies on *shell transport functions*, a novel description of a medium's scattering behavior across finite dis-

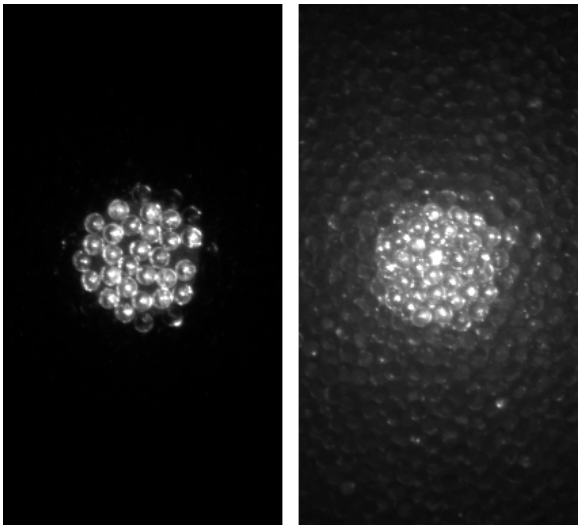

Figure 1: *Two photographs of glass beads illuminated by a spotlight. Left: a single layer of beads. Right: a deep container of beads, in which light spreads by multiple scattering in the volume.*

tances. Using this formulation we show how the results of local scattering simulations can be stored and packaged so that they can later be used to generate paths through random media that cross long distances without considering individual scattering events. Finally, we describe a new method for rendering homogeneous, isotropic random media that generates paths using precomputed scattering solutions that are compressed and randomly sampled using Nonnegative Matrix Factorization.

2. Prior Work

Light transmission in continuous media (known as volumetric light transport, or radiative transfer) has been studied extensively in many fields; Chandrasekhar [Cha60] and Ishimaru [Ish78] are good references on the basic theory. Continuous media are used in computer graphics [CPP*05] to simulate visibility and other atmospheric effects, light scattering in clouds, and light reflection from translucent solids and liquids. Clouds and translucent materials are most similar to the scenes we examine in this paper because they are dominated by multiple scattering. For these cases methods based on diffusion approximations [Sta95, JMLH01] are often used. But because of the independence assumptions underlying conventional radiative transport none of these methods can be used directly on discrete random media.

Our method is based on statistical path tracing and is related to the many methods in this general class, including classic path tracing [KV84] and two-pass photon mapping approaches [JC98, MM06]. Like the hybrid method of Li et al. [LPT05], we use distance from the surface of a medium to switch from path tracing near the surface to a more efficient approach in the interior, but we address discrete rather than continuous media and use an entirely different method in the interior.

One method in the photon-mapping category, described by Moon and Marschner [MM06], also handles light transport in volumes filled with scattering objects. That method, which is specialized to rendering hair, actually traces paths through the geometry itself, and therefore relies on a memory-intensive 5D photon map to smooth and cache the multiply scattered radiance. By contrast, our new method constructs paths efficiently using a stochastic model, without considering individual scatterers, leading to a simple and lightweight from-the-eye path tracing method.

Addressing a similar problem to ours from a very different approach, Tong et al. [TWL*05] presented an appearance capture system for quasi-homogeneous materials based on the idea of using dense BTF-like data to describe effects of local geometry and texture, coupled to a diffusion model that describes longer-scale transport. Like our method, theirs simulates the appearance of multiple scattering as viewed through a local geometry, but their goals and methods are quite different.

The term *discrete random medium* comes from work in the optics literature motivated by scattering problems involving rain, ice, snow, colloids, and other media with closely spaced scatterers [GMS06, GSI*03]. Our observations about attenuation [IK82] and retroreflection [KI84] are well known from experiments in this field. In these fields long wavelengths or small scatterers are often important, and full electromagnetic scattering analyses and computations are the norm [TDSK98]. Cost and macroscopic scale have led us to methods based on geometric optics instead.

At the core of our scattering method is a numerical representation of a precomputed shell transport function, which is stored in a factored form computed using Nonnegative Matrix Factorization [LS00], a method that has been successfully applied to compression, representation, and sampling of BRDFs [LRR04], illumination [LRR05], subsurface scattering [PvBM*06], and spatially-varying BRDFs [LBAD*06].

3. Theory

In this section we explain the distinction between continuous and discrete random media and introduce a framework for describing light transport in discrete random media using precomputed scattering solutions.

3.1. Continuous and discrete random media

The radiative transport equation [Ish78] is frequently used to model light propagation through scattering media. In its

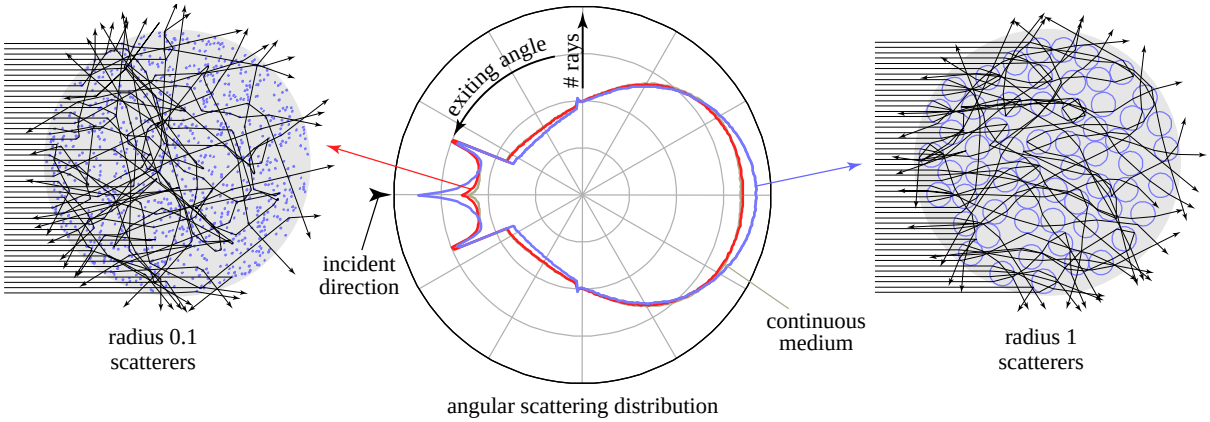

Figure 2: *A 2D demonstration of the effects of non-independent scattering events. A circular region is filled with randomly distributed circles; on the right, the circles are as big as fits in the space, and on the left they are a tenth that size. The density of circles is increased on the left to match the overall mean free path of the medium. A large number of rays is traced through this geometry, and the far-field distribution of scattered rays is shown in the center, together with the result of a volumetric simulation. The small, well-separated circles nearly match the volumetric solution, but the closely packed circles behave differently.*

integral equation form, this standard model relates the radiance in the volume to an integral of itself. The radiance at a given point $\mathbf{x}$ and direction ω, denoted $L(\mathbf{x},\omega)$, is:

$$L(\mathbf{x},\omega) = \int_0^{t_\mathbf{z}} e^{-\sigma_t t}\, \sigma_s \int_{S^2} f_p(\omega,\omega')L(\mathbf{y}(t),\omega')\,d\omega'\,dt + e^{-\sigma_t t_\mathbf{z}} L_i(\mathbf{z},\omega) \quad (1)$$

where $\mathbf{y}(t)$ is the arc-length parameterization of the ray, $\mathbf{z}$ is the point at which it exits the volume, and $t_\mathbf{z}$ is the distance from $\mathbf{x}$ to $\mathbf{z}$.

The first term in this equation describes the contribution of scattering to $L(\mathbf{x},\omega)$. At the point $\mathbf{y}(t)$ on the ray, the light scattered in the direction ω is the scattering coefficient σ_s times the integral of the phase function f_p times the radiance L in the medium at $\mathbf{y}(t)$ over the sphere S^2. This scattered light is attenuated over the distance t at the rate σ_t, known as the attenuation coefficient. The second term describes the contribution of transmitted light to $L(\mathbf{x},\omega)$: the radiance L_i entering the medium at the end of the ray is attenuated along the length of the ray before it adds with the scattered light. This formulation of the radiative transport equation is commonly used in computer graphics [CPP*05].

This model is so widely used that we may easily forget that it is a heuristic approximation, based on certain assumptions about what causes scattering [Ish78]. In particular, scattering is assumed to happen by interaction of light with scattering particles that are far apart relative to their size; and the outcome of one scattering event is assumed to be statistically independent of other events along the path.

In many media, particularly solids, this independence assumption does not hold. When scatterers are closely packed, but still randomly arranged, they can be treated as a random medium, in that the statistical properties of the geometry are more important than the specific arrangement of parts. But they do not satisfy the requirements of the radiative transport model, as is discussed in Section 3.2. We call this kind of collection of scatterers a discrete random medium.

For example, marble is a discrete random medium because it is made of many separate crystals separated by discontinuities that create reflecting surfaces. Because these crystals are disorganized the material acts like a scattering medium on a large scale, but because they are close together the interactions of paths with nearby crystals are correlated. Other examples include soap foam, dense suspensions of droplets or bubbles in liquid, and loose granular materials like table salt or glass beads. All these materials can be treated as random media for multiple scattering computations, even though the complete geometry is required for low-order scattering whenever the structure is visible in the image.

3.2. Light transport in discrete random media

We can think of the radiative transport model as a stochastic process for generating paths through the medium. Its job is to describe the distribution of paths that a light particle might take after passing through a particular position and direction. The description given by the radiative transport model is: "the path goes straight for a distance drawn from an exponential distribution, then turns through an angle drawn from the phase distribution. The statistics of the rest of the path are the same." The three underlying assumptions are that path

lengths are exponentially distributed, that the turning angle is independent of the distance traveled, and that subsequent events are independent of this one.

To describe a discrete random medium in similar terms, we need to describe the distribution of paths that pass through a particular point and direction, averaged over all instances of the random geometry. This stochastic model is a valid description of the medium in any situation where replacing the specific geometry with another geometry drawn from the same statistical distribution would not change the image. The stochastic model is not good for directly viewed parts of the medium when individual scatterers are visible, but it can be used for computing scattering deeper inside the medium, where one can no longer tell from the image which particular arrangement of scatterers is present.

However, the same kind of description as used in the continuous case, based on independent scattering events, can't be used. This can be illustrated by a simple example in 2D. Figure 2 shows a few paths from a simulation of scattering in a circle filled with dielectric circles in a Poisson distribution. For small, well separated circles, the angular distribution of scattered light can be adequately predicted using a continuous medium in which the phase function is the phase function of scattering from one circle and the attenuation coefficient is computed from the size and density of the circles. However, when we instead use larger, closely packed circles with the same mean free path, the result is different. In the case of circles, the overall distribution is biased more toward forward scattering, and a strong retroreflection peak emerges.[†] Attenuation through closely packed circles or spheres also follows a non-exponential decay, as shown in Figure 8.

To successfully represent closely packed scatterers as a sequence of independent random events, we must look at statistics of long enough subpaths that the behavior of preceding and following sections of path are sufficiently independent.

3.3. Computing statistics of local multiple scattering

The core idea of our rendering method is to think of scattering not in terms of single events but by describing the total effect of all scattering, including multiple scattering, within a local, homogeneous region. For a particular starting point $\mathbf{x}$ and direction ω, we define the *shell transport function* as the 4D distribution over points and directions with which random paths will exit a spherical *shell* centered at $\mathbf{x}$ (Figure 3).[‡]

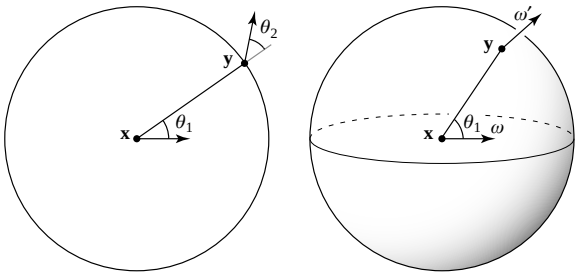

Figure 3: *Geometry of shells in 2D (left) and 3D (right). In 3D, paths start at* $\mathbf{x}$ *traveling in the direction* ω*, and their exit from the shell is recorded as a function of* $\mathbf{y}$ *and* ω'*. Because of symmetry, the probability of exiting at* $\mathbf{y}$ *depends only on* θ_1*, the angle between* ω *and the direction to* $\mathbf{y}$*.*

$$T_r(\mathbf{x}, \omega; \mathbf{y}, \omega') = p((\mathbf{y}, \omega') \mid (\mathbf{x}, \omega))$$

This function gives the probability that a path will *first exit* a sphere of radius r around the point $\mathbf{x}$ at $(\mathbf{y}, \omega')$ given that it passes through $(\mathbf{x}, \omega)$. Considered as a function of $\mathbf{y}$ and ω', T_r is a probability density over a sphere cross a hemisphere: $T_r\, dA\, d\Omega$ is the probability that the path exits the sphere in the area dA with a direction in the solid angle $d\Omega$ (in projected solid angle measure). The probability space for T_r is all paths through all geometries that might be generated by the random process that describes the medium. One can think of T_r as predicting the result of an experiment: suppose we trace many paths through many random geometries, and select out paths that pass through $\mathbf{x}$ in the direction ω, and record the point and direction where each one first exits the sphere of radius r centered at $\mathbf{x}$. The density of the resulting (point, direction) pairs, with respect to surface area times projected solid angle, will be described by T_r.

The shell transport function for a particular random process applies to any other sphere in the medium as long as the geometry inside the sphere is generated by the same random process. In this paper we consider homogeneous, orientation-independent media, whose statistics are stationary under all translations and rotations of 3D space, so a single shell describes any same-size sphere that fits inside the medium, for any starting direction. Furthermore, the symmetry with respect to rotation around ω means that the transport function $T_r(\mathbf{y}, \omega')$ depends on only three variables, not four.

In order to capture long-distance transport while also fitting into the corners of the volume, we store a set of shell transport functions for a range of radii, ranging from the medium's autocorrelation length up to the largest sphere that

[†] This phenomenon was observed years ago in the laboratory: a suspension of spheres in water shows enhanced retroreflection as the density increases [KI84].

[‡] This distribution is essentially the light field [LH96] that would exit the volume if the medium was illuminated by a narrow directed beam and everything outside the shell surface was removed.

fits in the volume. The resulting 4D function can answer the question:

> Of all paths that pass through **x** in the direction ω, what is the density of paths, with respect to projected area and solid angle, exiting the sphere of radius r at the point **y** in the direction ω′?

Knowing the answer to this question is very valuable in computing lighting in a medium, because it allows the contribution of light arriving at the shell directly from the light source to be evaluated. In discrete random media, the shell transport representation has an additional advantage: by rolling together all the scattering that happens over a volume, we capture the effects of correlation between subsequent events. As long as the shell radius is larger than the autocorrelation distance of the geometry, the shells can be chained together to make a valid statistical model for scattering in the medium.

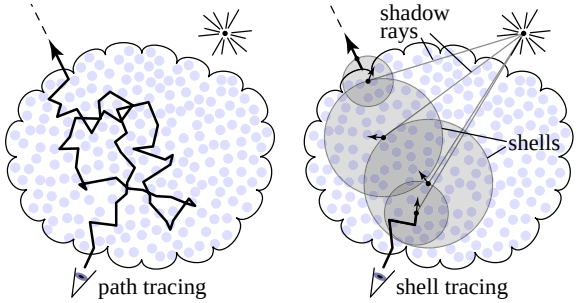

Figure 4: *The main idea of our method. In path tracing, all events along the path must be considered, and in a purely specular scene a path makes no contribution if it fails to hit the light source. With our new method, we use precomputed shells to advance in large steps, and at each step we can compute illumination from the source.*

3.4. The shell transport equation

Using the shell transport function, we can write a new rendering equation that directly relates radiance across large distances through the medium:

$$L(\mathbf{x},\omega) = \int_{S_r}\int_{H} T_r(\mathbf{x},\omega;\mathbf{y},\omega')L(\mathbf{y},\omega')\,d\mu(\omega')\,dA(\mathbf{y})$$

where S_r is a sphere of radius r centered at **x**, T_r is the shell transport function, H is the hemisphere, and μ is projected solid angle measure about the normal to S_r at **y**. Figure 3 illustrates the notation.

This equation can be used for any r above the medium's autocorrelation distance, but it applies only when the medium is homogeneous within the shell, and for this reason it cannot be used too close to the surface. Therefore this method must be coupled to another rendering equation to provide the boundary condition (or an approximate boundary condition can be used, as described in the next section).

By repeatedly drawing samples according to T, we can generate paths through a medium using large steps when we are in large homogeneous regions of random geometry. As we approach the edges we are forced to take smaller and smaller steps and eventually invoke the boundary condition. The distribution can also be used to compute direct illumination, as described in Section 4.3.

4. Method

We have developed a simple rendering algorithm based on shell transport functions. It applies to homogeneous, isotropic, random collections of discrete scatterers in which multiple scattering is dominant. In a preprocess, we trace paths through a representative part of the scene to compute an approximation of the shell transport function. We compress this function using matrix factorization. During the main rendering process, we use the stored shell transport function both to generate paths that advance rapidly through dense multiple scattering media, and to compute direct illumination within the volume, as shown in Figure 4.

4.1. Preprocessing

For the preprocessing phase we manually select a sphere that just fits within the largest region of the homogeneous medium. We also specify the autocorrelation distance, which becomes the minimum shell radius r_0. A set of radii linearly or logarithmically spaced between these two distances become the shell radii r_i.

We generate paths from the center of the sphere using standard importance sampling methods so that all rays carry the same weight. A random point and orientation $(\mathbf{x},\omega)$ along each path is chosen to serve as the path's starting frame of reference, to avoid correlation between the paths and the geometry, effectively tracing a different instance of the random geometry with each path. When a path first ventures a distance r_i from **x** and has scattered at least once, we record θ_1 and ω', in coordinates relative to ω (see Figure 3), in a data structure associated with shell i. The result is a set of samples on each shell whose density is proportional to the shell transport function.

After the paths have been traced, we estimate their density with respect to θ_1 and ω' using k-nearest neighbor density estimation in the style of photon mapping. We use $(\cos\theta_1, \omega'_u, \omega'_v)$ as the coordinates for the lookup, where ω'_u and ω'_v are coordinates of ω' projected to the unit square under a mapping that preserves projected solid angle on the hemisphere around the normal to the shell [SC97]. In this way the computed density is naturally exactly proportional to the shell transport function. Densities are estimated on a coarse 3D grid for each shell.

function **advanceRayUsingShells**($\mathbf{x}_i, \omega_i$):

r = max shell radius fitting around $\mathbf{x}_i$, else r_0
d = min(r, distance to surface along ω_i from $\mathbf{x}_i$)
with probability $f_a(d)$
  $\mathbf{x}_{i+1} = \mathbf{x}_i + r * \omega_i$; $\omega_{i+1} = \omega_i$
else
  choose $\mathbf{y}$ with probability $p_r(\mathbf{y})$ from T_r
  choose ω' with probability $p_r(\omega' \mid \mathbf{y})$ from T_r
  $\mathbf{x}_{i+1} = \mathbf{y}$; $\omega_{i+1} = \omega'$ in world coordinates
if $\mathbf{x}_{i+1}$ is outside of the volume, $\omega_{i+1} = \mathbf{x}_{i+1} - \mathbf{x}_i$

Figure 5: *Pseudocode for advancing rays inside a medium using shell transport functions.*

function **advanceRayUsingShellsWithDirect**($\mathbf{x}_i, \omega_i$):

r = max shell radius fitting around $\mathbf{x}_i$, else r_0
d = min(r, distance to surface along ω_i from $\mathbf{x}_i$)
cast a shadow ray $(\mathbf{x}_i, \omega_l)$ to light, hitting S_r at $\mathbf{x}_l$.
if distance to surface along ω_l from $\mathbf{x}_i$ < r
  pixel value += light sample $* p_r(\mathbf{y} = \mathbf{x}_l)$
with probability $f_a(d)$
  $\mathbf{x}_{i+1} = \mathbf{x}_i + r * \omega_i$; $\omega_{i+1} = \omega_i$
else
  choose $\mathbf{y}$ with probability $p_r(\mathbf{y})$ from T_r
  choose ω' with probability $p_r(\omega' \mid \mathbf{y})$ from T_r
  $\mathbf{x}_{i+1} = \mathbf{y}$; $\omega_{i+1} = \omega'$ in world coordinates
  cast a shadow ray $(\mathbf{x}_{i+1}, \omega_l)$ to light
  d_l = distance to surface along ω_l from $\mathbf{x}_{i+1}$
  pixel value += light sample $* f_a(d_l) * p_r(\omega_l \mid \mathbf{x}_{i+1})$
if $\mathbf{x}_{i+1}$ is outside of the volume, $\omega_{i+1} = \mathbf{x}_{i+1} - \mathbf{x}_i$

Figure 6: *Pseudocode for advancing rays inside a medium using shell transport functions, with direct illumination included.*

The resulting four-dimensional array can require a considerable amount of storage, depending on the number of shells used and the coarseness of the 3D sampling grids. However, the shell transport functions tend to vary smoothly with respect to both position and direction, and shells with similar radii tend to have similar shell transport functions, making the data highly compressible. Using Nonnegative Matrix Factorization, we can factor our 4D array into a weighted sum of a small number of outer products of a function of (r, θ_1) with a function of ω', each of which is stored as a 2D array of nonnegative samples. As a result, this factored representation requires considerably less storage than the 4D input array. In addition, this representation can be used to generate points $\mathbf{y}$ for a given r according to the marginal probability $p_r(\mathbf{y})$ and to generate directions ω' according to the conditional probability $p_r(\omega' \mid \mathbf{y})$, allowing us to importance sample outgoing paths from a given shell during the path tracing process described in Section 4.2. This factored representation also permits direct evaluation of the probabilities $p_r(\mathbf{y}_0)$ and $p_r(\omega'_0 \mid \mathbf{y})$, which are needed to estimate the contribution of direct illumination within the volume and are discussed in Section 4.3.

While the paths are being traced, we also record the 1D distribution of free path lengths in a dense histogram. From this distribution we can derive a discretized attenuation function $f_a(d)$ for the medium, which represents the probability of a ray continuing unimpeded for a distance d within the medium. This function can be used directly, with no need for compression or smoothing.

As a result of this preprocess we now have a compact representation of the shell transport functions for a set of shells with radii that range from the autocorrelation distance of the medium to the radius of the volume in the scene. These shell transport functions can be used when inside the core of the volume to provide a smooth, easily sampled representation of the path distribution that would result from tracing paths through spheres of discrete medium. Moreover, these functions are dependent only upon the statistics of the medium, not on the lighting or viewing conditions or the overall shape of the volume, meaning they could be reused in any other scene with the same discrete random medium.

4.2. Path tracing

The simplest operation to do with the factored transport functions is tracing paths. Once a viewing ray has scattered a minimum number of times and has penetrated to a distance r_0 from the boundary of the medium, we switch from tracing rays to generating samples from the shells as shown in pseudocode in Figure 5. To extend a path, we simply choose the largest available shell radius r that fits entirely within the volume, then importance sample the next point and direction according to the factored distribution of T_r. This process is illustrated in Figure 4.

When the path comes within a distance r_0 from the boundary again, the use of the shell transport functions is no longer precisely valid, as the shell transport functions are derived from spheres full of media. It might seem desirable at this stage to stop using the shell functions and drop back into standard geometric path tracing. This is not feasible, however, because if the path is restarted arbitrarily without regard for the local geometry, the generated paths will not be representative of paths that would actually occur in the discrete medium. For instance, we might start a path inside a sphere where total internal reflection will prevent its escape.

Instead, we continue to use the smallest shell, and if a point is generated outside the boundary of the medium we consider the path to have escaped. In this case the exiting ray's direction is that of the ray connecting the previous position to the escaped position. While this approximate exit

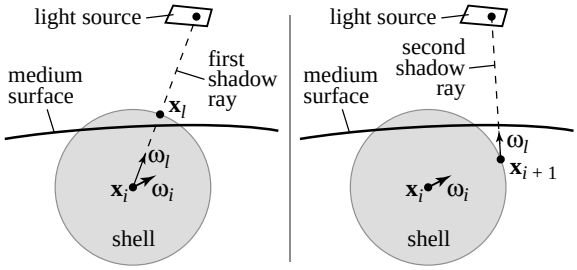

Figure 7: *The two cases for direct illumination. A path from* $(\mathbf{x}, \omega)$ *could hit a given light source point by choosing* $\mathbf{x}_l$ *and escaping; or the direction* ω_l *might be chosen as the next starting direction. The two ways of escaping the medium are handled by two shadow rays.*

strategy does introduce some error, experiments have shown that the resulting exiting path distribution is similar on average to the correct distribution, because shell-generated paths have scattered through large distances of medium and are significantly randomized by the time they exit the volume.

4.3. Direct illumination

Tracing paths with shells speeds up computation by taking large steps within a medium, but even accelerated path tracing of specular discrete media is very inefficient. Since every contributing path through the volume is part of a caustic or highlight, there is no way to directly estimate the contribution of light sources and a traditional path tracer must just wait for rays to hit the source, which is infrequent for small sources. A major advantage of using the shell transport functions is that they give the continuous probabilities required for direct lighting calculations.

At each event along a shell-generated path, we would like to estimate the probability that the next path segment will exit the medium and hit a light source by sampling positions on the light sources. There are two ways in which shell-generated paths can exit the medium; either by choosing not to scatter along a path segment near the surface of the volume, or by generating a point that lies outside of the volume with a shell transport function. We can calculate the probability of generating each of these types of exiting paths to estimate direct illumination.

Figure 6 contains the path tracing pseudocode from Figure 5 except with direct illumination calculations included. Suppose point $\mathbf{x}_i$ is close to the boundary of the volume, as in the left panel of Figure 7. Point $\mathbf{x}_{i+1}$ may be generated outside of the volume, in which case the path continues into the scene in the direction normal to the last shell at $\mathbf{x}_{i+1}$. This outgoing ray may hit a light source, and we would like to estimate this contribution using the continuous shell transport function centered at $(\mathbf{x}_i, \omega_i)$. By casting a shadow ray $(\mathbf{x}_i, \omega_l)$ from $\mathbf{x}_i$ to a light source, we can find a point $\mathbf{x}_l$ on the shell that may contribute this type of direct illumination. Its contribution is then simply the intensity from the light sample multiplied by the probability of generating the point $\mathbf{x}_l$ using the current shell, which is $p_r(\mathbf{y} = \mathbf{x}_l)$. We can read this probability directly from the factored representation of T_r.

Alternatively, a ray $(\mathbf{x}_{i+1}, \omega_{i+1})$ may be generated from a shell around the ray $(\mathbf{x}_i, \omega_i)$, as in the right panel of Figure 7. This path may never scatter again in the medium, and may then hit a light source, so that contribution needs to be accounted for here using the current shell transport function. We cast a shadow ray $(\mathbf{x}_{i+1}, \omega_l)$ to find one such contributing exit direction. The probability of generating $(\mathbf{x}_{i+1}, \omega_l)$ using the current shell function is simply $p_r(\omega_l \mid \mathbf{x}_{i+1})$, and the probability of this contributing path exiting the medium without scattering again is given by $f_a(d_l)$, where f_a is the attenuation function for the medium and d_l is the distance between $\mathbf{x}_{i+1}$ and the volume boundary in the direction ω_l. The contribution of the path is thus the product of these two probabilities and the intensity of the light sample.

By estimating both of these contributions at every shell-generated event in paths through the medium, direct illumination is fully included in this algorithm. This leads to a smooth estimate of the multiple scattered radiance in the scene, regardless of the size of the light sources, which is unachievable with conventional path tracing in specular discrete random media.

5. Results

The primary media used to test our algorithm were closely packed glass spheres, Poisson distributed or physically simulated in the presence of gravity. We have shown that these media have significant correlation between consecutive scattering events, as seen in Figure 2; unlike other methods based on continuous formulations, our new shell-based algorithm is able to account for this correlation. We present images of test scenes that vary the shape and size of a volume of spheres, with different types of illumination, rendered with our new approach as well as with a standard geometric path tracer. We validate our packed sphere results against a photograph of multiple scattering in an actual volume of glass spheres, and finally present the results of our algorithm on a scene full of small glass Buddha meshes.

We first preprocess each scene to produce factored shell transport functions for the discrete random media. There are several parameters that affect the run-time of this stage, and we present our settings here, although most parameters seem quite flexible. We choose 12 to 16 linearly spaced sphere radii, and then trace 500,000 paths from the center of the scene, which takes around 5 minutes on a single Pentium 4 based machine. We then estimate the sample density in each shell at 50x50x25 resolution, which takes another 4 minutes.

The resulting sampled shell transport functions are then factored using NMF with 4-6 terms, which takes 4 minutes in MATLAB and reduces data storage from roughly 30 MB to 250 KB. The resulting factored representation, as well as the 1D attenuation function derived from the observed histogram of free path lengths, are stored on disk with the scene description files. Figure 8 shows the distribution with respect to θ_1 for a set of shells used to render packed spheres, as well as the observed free path length distribution and the attenuation function f_a that was derived from it. In all cases, the total preprocessing time is a very small fraction of the time required to render high quality results.

Our new rendering algorithm uses shell transport functions to smoothly estimate the contribution of multiple scattering in discrete scattering media. Low order scattering from scatterers near the surface of the volume is inherently not smooth, as reflections and refractions of the light sources in the scene are distinctly visible. In light of this, our new results are computed as the sum of two images: an image containing just the low order scattering, which is rendered using standard geometric path tracing with a large number of rays, and another image containing a much smaller number of higher order paths generated using shell transport functions. Therefore our result images and timings are each broken out into these two components. This second component really represents the contribution of our new algorithm, as we can produce a low-noise image of the high order scattering several orders of magnitude faster than a geometric path tracer can.

Figure 9 shows renderings of a scene composed of 120,000 identical glass spheres, Poisson distributed within a cubic volume. It is illuminated from above by a small rectangular area light source, and a rectangular blocker is positioned above the volume so as to cast a hard shadow edge across the cube. The spheres are large enough to be seen individually near the surface, providing visible texture and specular reflections to the result. But high-order scattering is visible as well, as light can be seen penetrating deep into the cube and into the shadowed region of the volume, emerging as a smoothly varying glow across the image.

The output from a path tracer running on a compute cluster with 128 1.7 Ghz processors for 236 minutes is shown in Figure 9. A considerable amount of noise remains in the image, particularly in the shadowed and lower regions of the volume. The two output components from our algorithm are displayed combined and then separately in Figure 9 as well; the low-order scattering image was computed by a standard path tracer on the cluster in 12 minutes. The high-order scattering image was produced using shell transport functions for path generation and direct illumination, and took just 2.25 minutes to render on the cluster. The combined result contains considerably less noise than the result of the path tracer but was rendered over 16 times faster.

Figure 10 shows photographs of two different collections

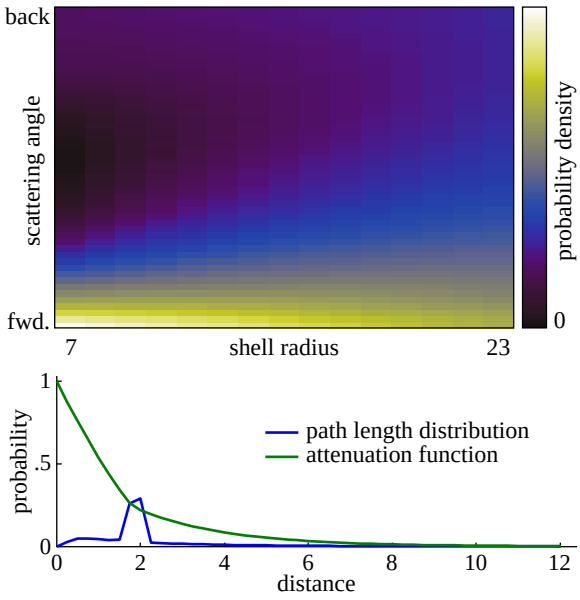

Figure 8: *Results of preprocessing a medium filled with densely packed spheres. Top: the exit position probabilities for different shell radii r and turning angles θ_1 (this plot represents the marginal distribution $p_r(\mathbf{y})$). Strong forward and weaker backward scattering are both observed, in distributions that widen for larger radii. Bottom: the observed attenuation function of the medium, along with the distribution of path segment lengths from which it is derived. Neither function has the exponential form assumed by radiative transport.*

of 1 mm diameter glass spheres illuminated by hard-edged spotlights. At left, the spheres are arranged in a cubic volume, and a considerable amount of light can be seen spreading outside of the spotlit area. At right, the spheres are arranged in just a single layer, backed by light absorbing material, and it is clear that the amount of light spreading outside of the lit area as well as the intensity of the lit area itself is considerably less than when the spheres are a part of a volume. This underscores the importance of multiple scattering in volumes of such scatterers, and demonstrates the smoothness typical of high-order scattering.

We modeled a scene matching the spotlit volume scene using measurements of the spheres and volume, and rendered it with both path tracing and our new shell-based approach. The results are shown in Figure 11. Due to the small size and solid angle of the spotlight, path traced rays are very unlikely to exit the medium and hit the light source—the image at left is the result of 28.1 hours of path tracing on the cluster, and the texture of the spheres outside of the spotlight is still barely visible. Our method required 0.78 hours to path trace the low-order contribution, and another 0.27 hours to

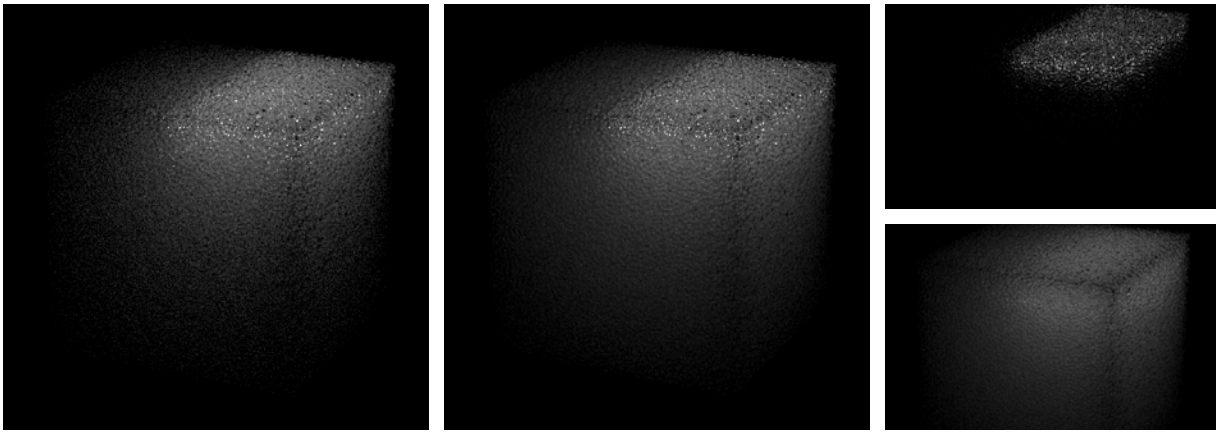

Figure 9: *Cubes of glass spheres with a hard shadow edge across the top face. Left: path traced image, in 236 minutes on a compute cluster. Center: combined output of our algorithm, in 14.25 minutes. Top right: path traced low order scattering, in 12 minutes. Bottom right: high order scattering using shells, in 2.25 minutes. Precomputation time was 14 minutes on a single machine.*

compute the high-order scattering image. The sum of these images is also shown in Figure 11. Again we achieve far superior image quality in less than 1/40 the amount of time as the path tracer. Compared to the renderings, the spheres in the photograph appear a bit cloudy, with broader specular highlights within the spotlight and slightly less spreading light. We conjecture that this is due to non-zero absorption in the scatterers and contaminants on their surfaces. However, the spotlit region of the photograph contains approximately 65% of the intensity of the image, while the same region in our rendering contains 57% of that image's total intensity, implying that our scattering simulation is a good approximation of real-world scattering behavior.

Next we produced a scene in which 100,000 glass spheres sit piled in a bowl on a table, illuminated by a moderately sized area light source above and to the left of the camera. Rendered results are shown in Figure 12, including the component images from our method. The path tracer required 175 minutes to produce the image on the left, while our new method took 11.5 minutes to produce the low-order image and 4.2 minutes for the high-order image. Here our shell-based approach is able to produce images comparable to the path traced reference image in 1/11 of the time, which shows that the new method excels even in more path tracer friendly scenes like this one, with a larger light source compared with our other scenes.

Finally we made a scene similar to the bowl of spheres, but filled the bowl with 10,000 glass Buddha meshes, each composed of 10,000 triangles. Figure 13 shows the results of our new algorithm on this very complex scene, both from afar and when zoomed in on the edge of the bowl. The full bowl image required 113 minutes of low order path tracing and 21 minutes of high-order shell tracing on the cluster, while the zoomed image took 135 minutes and 40 minutes for low and high order images, respectively. While we could not obtain a photograph of 10,000 miniature glass Buddhas for comparison, we feel that our algorithm produces convincingly realistic renderings of this scene, much more efficiently than other existing rendering methods.

5.1. Limitations

The method we present is specialized to orientation-independent homogeneous media and has so far only been demonstrated on non-absorbing media, as they represent the most difficult multiple scattering problems. Absorbing media are supported by our method without additional precomputation or rendering time, but extending the method to support heterogeneous or orientation-dependent media would increase the amount of required precomputation. Additionally, we ignore refraction at media boundaries when calculating direct illumination, which would be present if the host medium surrounding the scatterers differed from the background medium.

6. Conclusion

We have presented a new framework for solving light transport problems in discrete random media. Discrete random media are different from the commonly used continuous media, because the lack of independence between events causes transport behavior that—even when averaged over random geometry—does not behave like a continuous medium. This makes discrete random media difficult to model at the small scale, based on individual scattering events, other than by brute force approaches. The key insight of this paper is that we can regain the independence of events if we redefine the

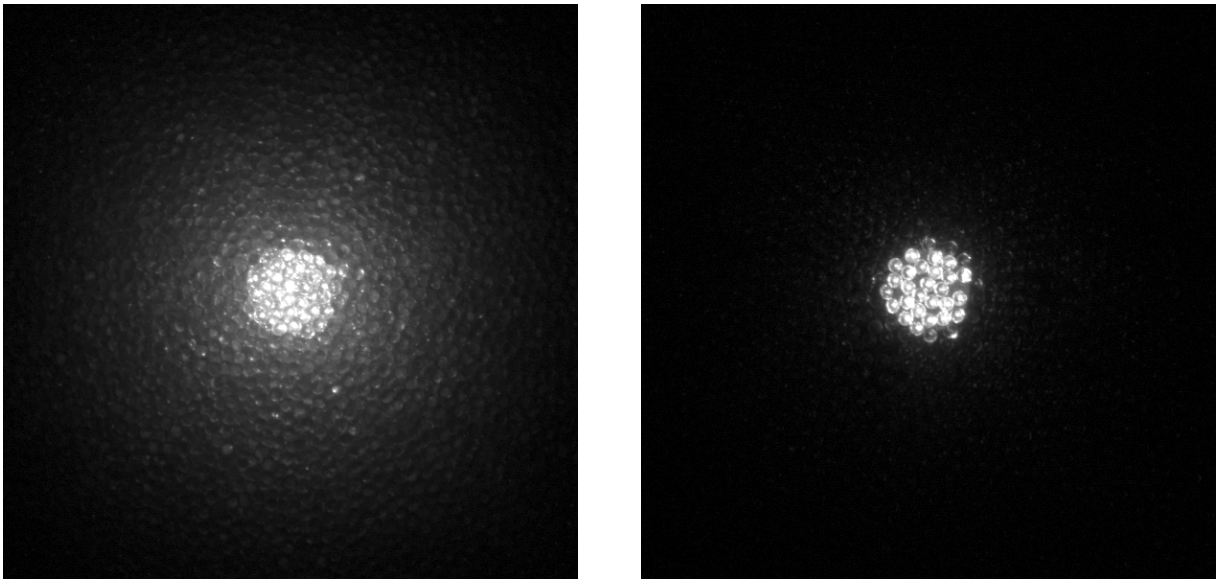

Figure 10: *Photographs of glass spheres. Left: a cubic volume of spheres under spot illumination. Right: a single layer of spheres under the same conditions.*

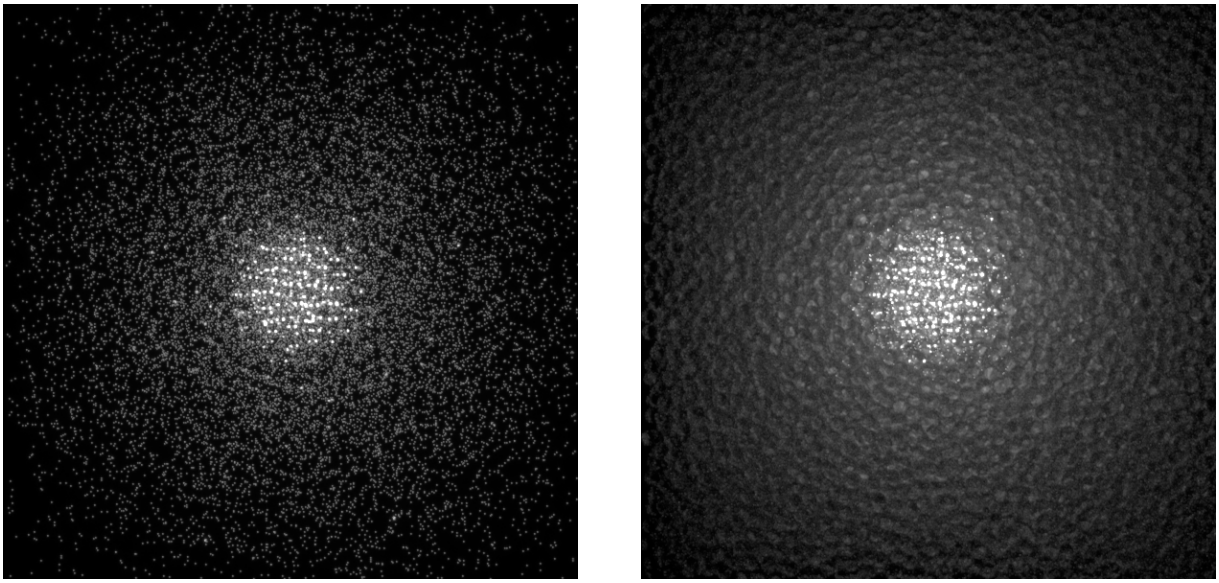

Figure 11: *Renderings of the volume photographed in Figure 10. Left: path traced image, 28.1 hours on a compute cluster. Right: output of our algorithm, 1.05 hours on the cluster. Precomputation time was 10 minutes on a single machine.*

Figure 12: *A bowl of 100,000 glass spheres. Left: path traced image, 175 minutes on a compute cluster. Center: combined output of our algorithm, 15.7 minutes on the cluster. Top right: path traced low order scattering, in 11.5 minutes. Bottom right: high order scattering using shells, in 4.2 minutes. Precomputation time was 10 minutes on a single machine.*

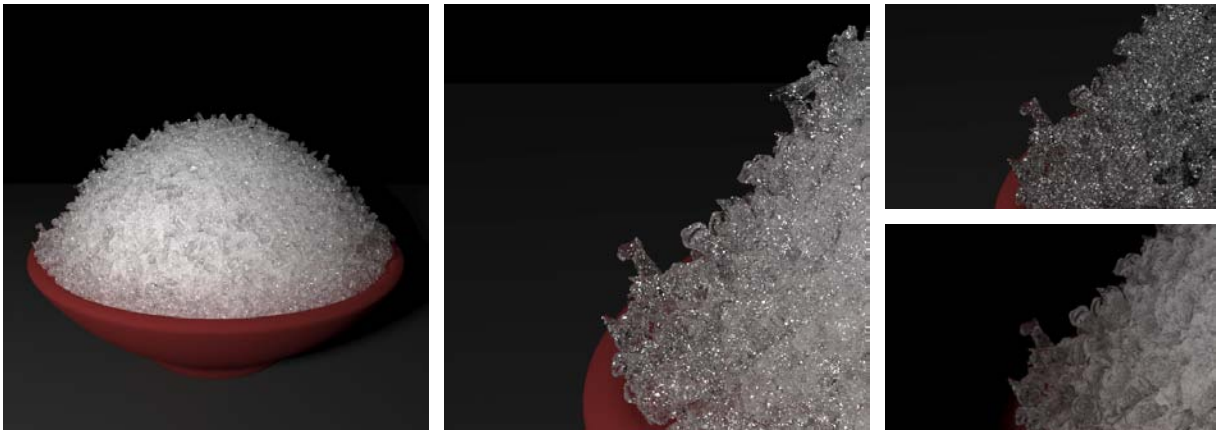

Figure 13: *Renderings of 10,000 glass Buddha meshes, each with 10,000 triangles, using our new method. Left: the full bowl of Buddhas, 133 minutes on the cluster. Center: inset of the left edge of the bowl of Buddhas, 175 minutes on the cluster. Top right: path traced low order scattering for the inset, in 135 minutes. Bottom right: high order scattering using shells, in 40 minutes. Precomputation time was 17 minutes on a single machine.*

problem in terms of "larger" events that encapsulate several scattering events. We have identified a convenient and workable way to do this, based on shell transport functions that encapsulate all scattering within a local volume.

Using these larger-scale primitives in path tracing has two important advantages. First, it is faster to construct the paths because fewer events need to be considered. Second, and no less important, it allows the use of shadow rays, because the shell functions can predict the probability (over random geometries) of a path going toward the light source.

An interesting feature of our results is that, after transforming the scattering problem into multidimensional functions that describe long-distance transport directly, the results tend to be fairly smooth, simple functions for large radii. This suggests looking for simple representations, such as empirical functions or continuous scattering problems that are equivalent for sufficiently large scales.

Acknowledgments

The authors would like to thank Jonathan Kaldor for generating many of the scenes that we rendered. This research was supported by funding from Unilever Corporation, NSF CAREER award CCF-0347303, NSF grants CCF-0541105 and ACI-0205438, and an Alfred P. Sloan Research Fellowship. Computing equipment was donated by Intel Corporation.

References

[Cha60] CHANDRASEKHAR S.: *Radiative Transfer*. Dover Publications, New York, 1960.

[CPP*05] CEREZO E., PÉREZ F., PUEYO X., SERON F. J., SILLION F. X.: A survey on participating media rendering techniques. *The Visual Computer 21*, 5 (2005), 303–328.

[GMS06] GUÉRIN C.-A., MALLET P., SENTENAC A.: Effective-medium theory for finite-size aggregates. *JOSA A 23*, 2 (2006).

[GSI*03] GIUSTO A., SAIJA R., IATÌ M. A., DENTI P., BORGHESE F., SINDONI O. I.: Optical properties of high-density dispersions of particles: application to intralipid solutions. *Applied Optics 42*, 21 (2003), 4375–4380.

[IK82] ISHIMARU A., KUGA Y.: Attenuation constant of a coherent field in a dense distribution of particles. *JOSA 72*, 10 (1982).

[Ish78] ISHIMARU A.: *Wave Propagation and Scattering in Random Media*, vol. 2. Academic Press, New York, 1978.

[JC98] JENSEN H. W., CHRISTENSEN P. H.: Efficient simulation of light transport in scenes with participating media using photon maps. In *Proceedings of ACM SIGGRAPH 98* (July 1998), pp. 311–320.

[JMLH01] JENSEN H. W., MARSCHNER S. R., LEVOY M., HANRAHAN P.: A practical model for subsurface light transport. In *Proceedings of ACM SIGGRAPH 2001* (Aug. 2001), pp. 511–518.

[KI84] KUGA Y., ISHIMARU A.: Retroreflectance from a dense distribution of spherical particles. *JOSA A 1*, 8 (1984).

[KV84] KAJIYA J. T., VON HERZEN B. P.: Ray tracing volume densities. In *Computer Graphics (Proceedings of ACM SIGGRAPH 84)* (July 1984), vol. 18, pp. 165–174.

[LBAD*06] LAWRENCE J., BEN-ARTZI A., DECORO C., MATUSIK W., PFISTER H., RAMAMOORTHI R., RUSINKIEWICZ S.: Inverse shade trees for non-parametric material representation and editing. *ACM Transactions on Graphics 25*, 3 (July 2006), 735–745.

[LH96] LEVOY M., HANRAHAN P. M.: Light field rendering. In *Proceedings of SIGGRAPH 96* (Aug. 1996), Computer Graphics Proceedings, Annual Conference Series, pp. 31–42.

[LPT05] LI H., PELLACINI F., TORRANCE K. E.: A hybrid monte carlo method for accurate and efficient subsurface scattering. In *Rendering Techniques 2005: 16th Eurographics Workshop on Rendering* (June 2005), pp. 283–290.

[LRR04] LAWRENCE J., RUSINKIEWICZ S., RAMAMOORTHI R.: Efficient BRDF importance sampling using a factored representation. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 496–505.

[LRR05] LAWRENCE J., RUSINKIEWICZ S., RAMAMOORTHI R.: Adaptive numerical cumulative distribution functions for efficient importance sampling. In *Rendering Techniques 2005: 16th Eurographics Workshop on Rendering* (June 2005), pp. 11–20.

[LS00] LEE D. D., SEUNG H. S.: Algorithms for non-negative matrix factorization. In *Neural Information Processing Systems* (2000), pp. 556–562.

[MM06] MOON J. T., MARSCHNER S. R.: Simulating multiple scattering in hair using a photon mapping approach. *ACM Transactions on Graphics 25*, 3 (July 2006), 1067–1074.

[PvBM*06] PEERS P., VOM BERGE K., MATUSIK W., RAMAMOORTHI R., LAWRENCE J., RUSINKIEWICZ S., DUTRÉ P.: A compact factored representation of heterogeneous subsurface scattering. *ACM Transactions on Graphics 25*, 3 (July 2006), 746–753.

[SC97] SHIRLEY P., CHIU K.: A low distortion map between disk and square. *Journal of Graphics Tools 2*, 3 (1997), 45–52.

[Sta95] STAM J.: Multiple scattering as a diffusion process. In *Eurographics Workshop on Rendering 1995* (June 1995), pp. 41–50.

[TDSK98] TSANG L., DING K. H., SHIH S. E., KONG J. A.: Scattering of electromagnetic waves from dense distributions of spheroidal particles based on monte carlo simulations. *JOSA A 15*, 10 (1998).

[TWL*05] TONG X., WANG J., LIN S., GUO B., SHUM H.-Y.: Modeling and rendering of quasi-homogeneous materials. *ACM Transactions on Graphics 24*, 3 (Aug. 2005), 1054–1061.