Appears in *ACM Transactions on Graphics* 25:3 (Proceedings of SIGGRAPH 2006)

Simulating Multiple Scattering in Hair Using a Photon Mapping Approach

Jonathan T. Moon Stephen R. Marschner

Program of Computer Graphics
Cornell University

Abstract

Simulating multiple scattering correctly is important for accurate rendering of hair. However, a volume of hair is a difficult scene to simulate because scattering from an individual fiber is very structured and forward directed, and because the radiance distributions that arise from many such scattering events remain quite directional. For these reasons, previous methods cannot compute accurate images substantially faster than Monte Carlo path tracing.

This paper proposes a new physically accurate method for rendering hair that is based on previous volumetric photon mapping methods. The first pass generates a photon map by tracing particles through the hair geometry, depositing them along paths rather than at scattering events. The second pass ray traces the hair, computing direct illumination and looking up indirect radiance in the photon map. Photons are stored and looked up in 5D position-direction space to allow for the very directional radiance distributions that occur in hair. Together with a new radiance caching method for fibers, our method simulates difficult scattering problems in hair efficiently and with low noise.

The new algorithm is validated against path tracing and also compared with a photograph of light scattering in real hair.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Shading

Keywords: hair, physics-based rendering, multiple scattering, photon mapping, density estimation

Figure 1: A photograph demonstrating multiple scattering in a blond ponytail. The color and translucent glow of the hair cannot be captured using only direct illumination.

1 Introduction

Multiple scattering is important in accurate rendering of hair. Particularly for light-colored hair, which is strongly forward scattering, multiple scattering contributes a large fraction of the overall color. However, because of the geometric and optical complexity of hair, most approaches to simulating multiple scattering fail to provide a correct and efficient solution, and it is quite rare in practice to account properly for multiple scattering when rendering hair.

Multiple scattering in hair is difficult for several reasons. Scattering from individual fibers is forward-scattering, and the structure of the scattering function tends to preserve directional variation in the radiance field. Visibility in hair is also extremely complex. On the other hand, one can readily observe that the effects of multiple scattering are smooth, producing a glow that fades gradually from the point where light enters the hair. As with subsurface scattering in translucent materials [Jensen et al. 2001], this smoothness invites more efficient computational approaches that average over the details of the individual hairs.

In this paper we propose a new approach to rendering hair with multiple scattering that combines explicit geometry of strands, which is required for views with individual fibers visible, with a smooth volumetric representation of the scattered radiance field within the hair. This representation, based on a generalization of volume photon mapping, is constructed by tracing paths from the light source through the hair and depositing photons in a geometric data structure. Unlike previous photon maps, our method stores and looks up photons by both position and direction, estimating their density in 5D space. This approach treats directional variation in the scattered radiance field on an equal basis with spatial variation, which is important because both the scattering function of the fibers and the radiance distribution in the volume are strongly directional. Another difference from previous work is that photons are deposited all along particle paths, rather than just at the interaction sites. The resulting photon density gives radiance directly, avoiding a difficult-to-correct dependence on hair density and minimizing boundary bias near the surface of the hair assembly. Finally, we incorporate a new fiber radiance cache, analogous to an irradiance cache in traditional photon mapping, that allows illumination computations to be reused whenever possible. Together, these innovations lead to an accurate method that is 1–2 orders of magnitude faster than Monte Carlo path tracing, the only method previously available for rendering these scenes correctly.

We validate our new method by comparing its output to the results of Monte Carlo path tracing and to a photograph of light spreading through real hair.

2 Prior Work

Our main goal in this work is to render human hair accurately. Previous work in hair rendering has focused on shadowing [Lokovic and Veach 2000] and direct illumination [Kajiya and Kay 1989; Marschner et al. 2003]. Kajiya and Kay's simple shading model can be used for rendering black hair accurately, and Marschner et al.'s model, which accounts for transmission and internal reflection, can be used for dark hair. However, even though the latter model accurately describes scattering from blond or white hair, these colors can't be rendered accurately with only single scattering. Our work incorporates the model put forth by Marschner et al. into a new particle tracing based technique that makes the accurate appearance of all hair scenes possible and practical. Multiple scattering in hair was also discussed by Zinke et al. [2004], but that paper focused on converting the model of Marschner et al. to a near-field scattering model, without proposing rendering methods other than Monte Carlo path tracing.

A key idea presented in our work is treating multiply scattered light as a continuous distribution. This approach is shared by methods for rendering participating media. For optically thick media like marble or milk, light transport can be approximated as a diffusion process, on the assumption that penetrating light will become isotropic [Stam 1995; Jensen et al. 2001]. However, this assumption does not hold in a volume representing hair, as light scattering in an assembly of hair remains strongly directional even after several interactions. Another class of methods, including volumetric path tracing [Kajiya and Herzen 1984] and volumetric photon mapping [Jensen and Christensen 1998; Jensen 2001], simulate scattering events in a medium based on a known density and phase function that may vary spatially. But in a volumetric representation of hair, both the density and the phase function depend on position and direction, and neither is known a priori. These fundamental differences prevent these and other traditional rendering techniques from being directly applied to hair.

Our particle tracing algorithm deposits photons uniformly along particle paths. A related modification to particle tracing is storing the path segments themselves in a ray map [Havran et al. 2005]. However, this technique has not yet been demonstrated to work with participating media, nor in the dense and complex geometry associated with hair. Our photon map also uses a 6D tree structure to locate photons nearby in both position and direction. Such high-dimensional structures have been successfully used to incorporate time-dependent effects into photon mapping [Cammarano and Jensen 2002], and to store directional illumination information for accelerating Monte Carlo ray tracing [Lafortune and Willems 1995].

Ray tracing can be accelerated by caching illumination information during rendering and interpolating nearby illumination values from the cache whenever possible. Ward [1988] proposed caching indirect irradiance for ray tracing diffuse interreflection, and later improved the cache using hemispherical sampling [Ward and Heckbert 1992]. Support for glossy materials has also been incorporated, by caching directional incoming radiance rather than irradiance [Krivanek et al. 2005]. These techniques are similar in spirit to our proposed fiber radiance cache, but because they interpolate across surfaces they do not directly apply to assemblies of hair fibers where interpolation can only be done along individual fibers.

3 Multiple Scattering in Hair

Individual human hair fibers have a very distinctive scattering function. Measurements have shown that a large fraction of the light scattered from light colored fibers goes into the hemisphere facing away from the light source, and that the scattered light is confined to the fairly sharply defined cone of directions with inclinations near that of the incident direction [Marschner et al. 2003].[1] Figure 2 illustrates this property. This highly directed, strongly forward scattering behavior leads to spatially spread out but still directionally focused radiance distributions in hair, breaking the assumptions on which previous efficient rendering algorithms for participating media are based.

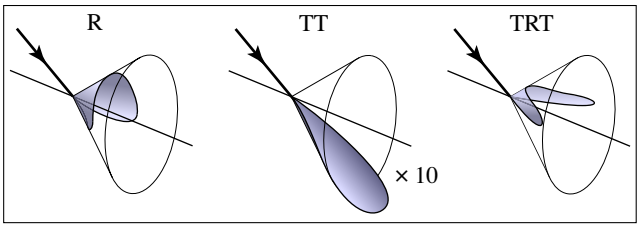

Figure 2: The three significant modes of scattering from a single hair fiber. The R mode is a surface reflection, the TT mode is transmission through the fiber, and the TRT mode is internal reflection. Radiance emerges roughly in the specular cone, remaining highly directional. For light colored hair the TT lobe contains a large fraction of the energy, making the scattering function strongly forward scattering.

Multiple scattering has important consequences for the appearance of hair, with significance that increases as the albedo of the strands goes up. In a strongly forward scattering material, we may expect to see a large contribution from multiple scattering compared to single scattering, since the majority of the incident energy is scattered into the material where it cannot be directly observed. As with diffusion-type multiple scattering in homogeneous media, such as most translucent solids, the multiply scattered light both contributes to the aggregate albedo of the hair and causes a spatial spreading of the light, softening geometric features and blurring hard shadow edges. The first of these effects is apparent in the results in Figure 10, as the overall reflectance in the direct illumination image is significantly less than that of the images with indirect illumination included. The second effect can be observed in Figure 8, a log-scaled photograph of a blond hair assembly illuminated by a sharp-edged spot of light. A translucent-like glow can be seen around the edges of the spot, indicating that light travels over significant distances through the hair before it emerges. Both of these effects must be contributed by multiple scattering.

Scattering in hair is very different from scattering in more isotropic media, however, because of the fibers' distinctive scattering properties. Unlike materials with no oriented structure, in which only the scattering angle matters, scattering in hair depends strongly on orientation. Light in hair is scattered much more broadly in the azimuthal direction and remains tightly focused in the tangential direction. This means that the spread of light through the hair is anisotropic—it spreads faster across the strands than along them. This effect may be observed in Figure 8. The angular distribution of light also remains quite directional even after several scattering events.

3.1 Modeling radiance in hair volumetrically

Because hairs passing through a small volume are arranged fairly randomly in space—though their directions are often closely aligned—we expect that the incident radiance distribution arriving at any given hair from the other hairs around it is statistically similar to what would be observed if the hair were in a slightly different location. That is, we assume that the incident radiance on a hair

[1] The surface structure of hair fibers directs the reflected components to slightly different cones, but for general discussion of scattering we will think of all scattered light being roughly in the specular cone.

can be treated as a smooth function of its position, even though the exact radiance field in a hair volume is very discontinuous. In regions where hairs are pressed tightly together, such as where a ponytail is bound by a rubber band, the fibers may tend to pack in a semi-ordered structure, which could invalidate this smoothness assumption, but for the more typical regions of loosely packed hair it seems reasonable.

This assumption of smoothness underlies our approach to accelerating the computation of multiple scattering: to evaluate the contribution of multiple scattering to the image, we need to represent the function $L_s(x,\omega)$ that gives the average indirect radiance observed from points near the 3D point x when looking in directions near the direction ω. Because of the assumption of smoothness, this average will generally be computed over a volume of radius considerably larger than the spacing between hairs, and over a solid angle at least as wide as the width of the cone in the hair's scattering function. Note that L_s is a function on a 5D domain (three spatial and two angular dimensions). Direct illumination from light sources is not included in L_s. The next section discusses our approach to precomputing and representing this function.

4 Simulating Multiple Scattering in Hair

Our rendering system is based on modeling scattering from one-dimensional fibers using a scattering function. This means we disregard variation in radiance across the width of a fiber, computing only the average radiance and treating the fiber as having that radiance across its entire width. To determine that radiance, we integrate the scattering function against the incident light distribution:

$$L_o(x,\omega_o) = \int_{S^2} f_s(\omega_i,\omega_o) L_i(x,\omega_i) \sin(\omega_i,u) d\omega_i. \qquad (1)$$

Here $L_o(x,\omega_o)$ is the outgoing radiance in direction ω_o, and $L_i(x,\omega_i)$ is the incident radiance from direction ω_i. Since the incident light is assumed constant across the fiber's width, L_i and L_o are both described at a single point x on the fiber. The scattering function f_s describes how light from the direction ω_i is distributed over exitant directions ω_o, and $\sin(\omega_i,u)$ is the sine of the angle between the incident direction and the fiber tangent u. The integral is taken with respect to solid angle.

To separate out the direct illumination, which has different characteristics from the indirect, we express L_i as a sum of direct and scattered light, $L_d + L_s$. This separates the rendering integral into two pieces, which we evaluate independently:

$$L_o(x,\omega_o) = \int_{S^2} f_s(\omega_i,\omega_o) L_d(x,\omega_i) \sin(\omega_i,u)\, d\omega_i + \int_{S^2} f_s(\omega_i,\omega_o) L_s(x,\omega_i) \sin(\omega_i,u)\, d\omega_i. \qquad (2)$$

Both integrals are evaluated using Monte Carlo integration. For the direct illumination integral, random directions toward light sources are chosen and f_s is evaluated using the approximation of Marschner et al. [2003]. For the indirect illumination integral, random directions are chosen according to f_s by the cylinder model described below, and the incoming radiance $L_s(x,\omega_i)$ is evaluated using a 5D photon map lookup, also described below.

4.1 Scattering functions for fibers

Our system uses two slightly different approximations to the scattering function of a hair fiber: one for evaluating the scattering function (as for direct illumination) and the other for generating random directions according to the scattering function. For evaluation, we use the model of Marschner et al., which uses a number of approximations to make it practical to compute. It is not simple to generate random directions proportional to this model, though, yet it is straightforward to generate random directions using the cylinder model that the scattering model approximates.

Therefore, when we need to generate random scattered directions we simply trace rays through an elliptical cylinder, simulating surface scales and roughness by rotating and jittering the normals at each interaction. Reflection or refraction is chosen according to the appropriate probability at each interface. The ray intersection computations are done in isolation from the rest of the geometry, leading to an efficient procedure for choosing random directions.

There is a small difference between these two scattering functions (that which is evaluated and that which is the probability density function of the scattered directions), and this could in principle affect the comparison of our method to the path tracer.

4.2 Photon mapping for hair

We use a two-pass method to compute the illumination on visible hairs due to multiple scattering. In the first pass, particles are traced from light sources into the hair volume and followed through multiple scattering events, and their positions and directions are stored into a 5D hierarchical data structure to record the flow of particles through space. In the second pass a density estimate is performed simultaneously in position and direction to estimate the radiance arriving at a hair from a particular direction.

The method is most closely related to volume photon mapping [Jensen and Christensen 1998], but there are several important differences. First, all ray tracing is done using geometry; no continuous medium is used. Second, we deposit photons along the particle paths with uniform probability, rather than at the interactions themselves. Third, we use a 5D density estimate rather than 3D, to better handle strong directional variations in radiance. Together these changes result in a weighted photon density that is simply equal to the indirect radiance $L_s(x,\omega)$.

Depositing photons During particle tracing, photons are generated along the path of each particle with a constant probability per unit length. We use Russian Roulette to terminate the paths, so that the weights of the photons remain constant.[2] In this way, the expected number of photons recorded in a particular volume with directions in a particular solid angle is directly proportional to the total length of particle paths within that volume and solid angle. This path length is in turn proportional to particle flux, and for suitably small volumes and solid angles the density of particles is an estimate of the radiance. Note that depositing photons in this fashion avoids any bias near the surface of the hair, as photons are placed along paths until they exit the bounding volume.

In volumetric photon mapping, particles are conventionally stored at each scattering event, producing a density that is proportional to the amount of outscattered radiance [Jensen and Christensen 1998]. The photon map is then used to estimate the inscattering term of the volume rendering equation, which redistributes the outscattered radiance, so this density is exactly what is needed. In our application, however, since the individual scatterers (fibers) are visible, we intend to use the photon map to compute illumination on an individual fiber. To do this we must estimate the radiance distribution in the volume *before* it interacts with the hair. If photons are placed in the map when paths interact with hairs, their powers must be divided by the local hair density in their direction in order for photon density to be used to illuminate fibers [Jensen

[2] The colored absorption in the fibers causes the weights in channels with higher absorption to decrease and those in less absorptive channels to increase as photon depth increases, but the average weight is fixed.

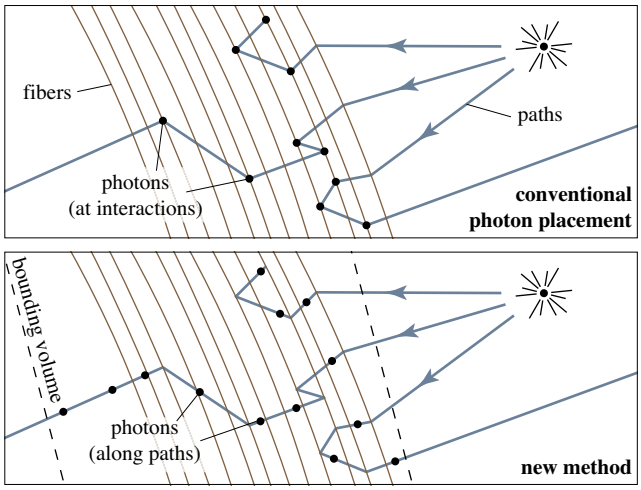

Figure 3: Depositing photons in the photon map. Top: depositing photons at interaction points, as done in previous work, results in a photon density that is proportional to the hair density, which is difficult to correct for. Bottom: depositing photons uniformly along paths results in a photon density that is proportional to radiance.

and Christensen 1998]. As we show in the next section our photons therefore need to carry a quantity that is power times distance. The difference in units is due to the factor of the volume scattering coefficient σ_s, with units of inverse distance, that is built into their density [Jensen 2001].

Photon storage and density estimation In conventional photon mapping, photons generated during the particle tracing pass are stored in a 3D tree, so that the n nearest photons to a given 3D point can be found efficiently. The photons are selected without regard to direction. For anisotropic scattering, nonuniformity in the distribution of photon directions is then handled by weighting the collected photons by the scattering function for the photon's direction [Jensen and Christensen 1998]:

$$(\omega \cdot \nabla) L_{\text{in}}(x,\omega) \approx \sum_{p=1}^{n} f_s(x,\omega_p,\omega)\frac{3P_p}{4\pi r^3} \quad \left(\frac{\text{W}}{\text{m}^3\text{sr}}\right), \qquad (3)$$

where L_{in} is inscattered radiance, P_p is the power of photon p, and r is the radius of the smallest sphere centered at x that contains all n photons.

This reweighting procedure works well if radiance is mildly directional or the scattering function is fairly isotropic. But it performs poorly when the radiance distribution is highly directional and the scattering function has strong peaks. If most of the nearby photons are traveling in directions that don't align with the peaks of the scattering function, very few photons will contribute significantly, leading to high variance. The situation is analogous to rendering reflection from a specular surface using samples distributed according to the incident light distribution [Veach and Guibas 1995].

A 3D photon map reduces noise through smoothing in the spatial dimensions, but has no similar mechanism for smoothing in the angular dimensions. In estimating the scattering integral the individual photons are treated as point samples. In our system we solve this problem by performing density estimation with respect to position and direction together (Figure 5). We retrieve a set of photons that are nearby in space to a particular point and also nearby in angle to a particular direction, and use their density to directly estimate radiance in that position and direction. This density is measured in the 5D space that is the Cartesian product of the set of possible

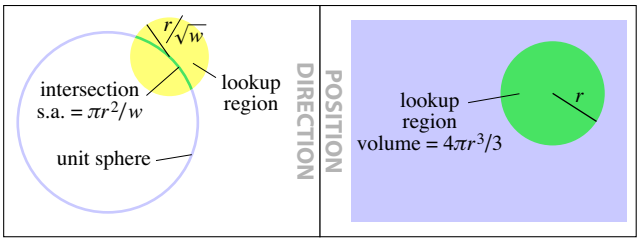

Figure 4: A 5D lookup in 6D Euclidean space. In 3D position space, a lookup of radius r simply encompasses a volume of $4\pi r^3/3$. In the 3D space of direction vectors, a corresponding lookup of radius $r/\sqrt{w}$ centered at a point on the unit sphere will include a 2D circular solid angle of size $\pi r^2/w$. Thus the total 5-volume contained in the lookup is the product of this solid angle and 3D volume.

directions (the direction sphere) with the set of possible positions (3D Euclidean space).

To perform this new type of density estimate we must efficiently locate photons near a point in 5D space, then calculate the size of the region in which they were found. Both these operations require choosing a metric for position-direction space.

To allow the use of a simple spatial data structure, we embed the 5D space in 6D Euclidean space by representing the directions as unit vectors. That is, the point (p,ω) is represented by the 6-tuple $(p_x,p_y,p_z,\omega_x,\omega_y,\omega_z)$. This embedding is convenient because it is easy to define a circular solid angle around a direction ω: it is the intersection of a sphere centered at ω with the direction sphere (see Figure 4). The solid angle (that is, the area on the direction sphere) that is selected by a lookup of radius r around a given direction vector is

$$\Omega = \begin{cases} \pi r^2 & 0 \le r < 2 \\ 4\pi & 2 \le r \end{cases}. \qquad (4)$$

Once the lookup region has a radius of 2, it envelops the entire direction sphere, and so further increases in radius have no effect.

These 6D points are stored in a 6D tree (that is, a k-D tree with $k = 6$), which is a straightforward generalization of the usual 3D tree that allows for efficient n-nearest-neighbor queries using reasonable metrics in 6D Euclidean space.

Because the dimensions of position-direction space are not all of the same type, choosing a metric defines a conversion between distances and angles. If one pair of photons with the same direction has positions separated by a distance r, while a second pair of photons with the same position has directions separated by an angle θ, the metric must choose which pair is closer. This establishes a key tradeoff between directional and spatial resolution in our algorithm, which we control by explicitly defining a weight w that can be used to adjust the relative importance of direction and position in the metric.

Since we actually define and apply the metric in the 6D space in which the photons are stored, we also must ensure that it is easy to compute the 5-volume of the region of position-direction space that falls within a particular distance according to the metric. For our metric we chose the maximum of the Euclidean metrics in position and direction space, compared using the weight w:

$$d((p_1,\omega_1),(p_2,\omega_2)) = \max(\|p_1 - p_2\|, \sqrt{w}\|\omega_1 - \omega_2\|). \qquad (5)$$

The weight w is defined so that a distance of $\sqrt{w}$ in position space equates to a distance of one unit in direction space. The value of w will depend on the scene's scale, and has units of $\frac{\text{m}^2}{\text{sr}}$. In practice, w is chosen to provide a desirable combination of spatial and angular accuracy for a particular scene. It also is often

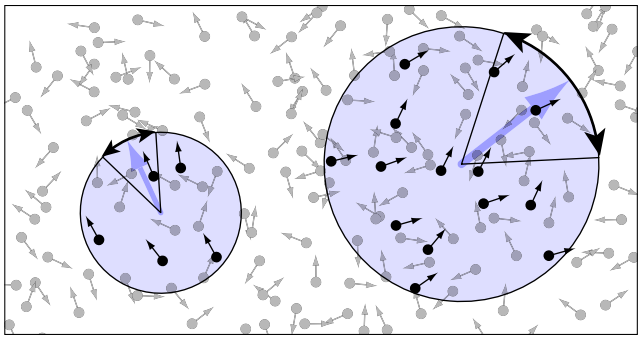

Figure 5: A 2D example of gathering photons in both position and direction. The smaller-radius gather on the left collects photons that have directions that fall within a particular angle from the lookup direction and also have positions that fall inside a circle centered at the lookup point. When the radius increases on the right, the area and the angle of the gathered region grow together.

useful to choose w and a maximum lookup radius r_{max} such that $r_{max} \leq \sqrt{2w}$ to prevent any photon lookup from encompassing a solid angle greater than one hemisphere.

With the metric proposed above, a 5-ball is simply the Cartesian product of a 3D sphere with a circular solid angle. The 5-volume is then just the product of the two volumes:

$$V(r) = \left(\frac{\pi r^2}{w}\right)\left(\frac{4\pi}{3}r^3\right) = \frac{4\pi^2}{3w}r^5. \tag{6}$$

Other choices of metric certainly exist, such as 6D Euclidean distance. This would provide a continuous trade off between space and direction that could select points spatially close yet distant in direction as well as points nearby in direction but far away in space. This behavior, while possibly desirable in some situations, makes it more difficult to reason about the extent of the lookup, and complicates the calculation of the 5-volume.

If we let Q_p be the value of photon p with units of W · m, then the estimated radiance corresponding to a 5D lookup of n photons that are found in a 5-ball of radius r is:

$$\tilde{L}_s(x,\omega) = \sum_{p=1}^{n} \frac{3wQ_p}{4\pi^2 r^5} \quad \left(\frac{\mathrm{W}}{\mathrm{m}^2\mathrm{sr}}\right). \tag{7}$$

It is important to set a maximum radius for lookups to prevent lookups in very dark areas from wasting time searching very large 5-volumes to find enough photons [Jensen 2001].

One important implication of a 5D lookup volume is that as the number of photons in the map increases, the radius shrinks very slowly, as the inverse fifth root of the number of photons. To double resolution, the number of photons must increase by a factor of 32. By comparison, normal volume photon mapping requires a factor of 8 increase to double resolution, and surface photon mapping only 4.

4.3 Radiance caching for hair

Ordinary surface photon mapping gains a great deal of efficiency from irradiance caching [Ward et al. 1988], which allows expensive final gather computations to be reused across a smoothly varying illumination field on a surface. Without an irradiance cache, a great deal of computation is wasted performing final gathers on nearly identical illumination fields for adjacent points on the same surface.

Our system shares the same property: close-together viewing rays will generally land in spatially close-by positions and if similar scattered directions are chosen, will wind up gathering nearly

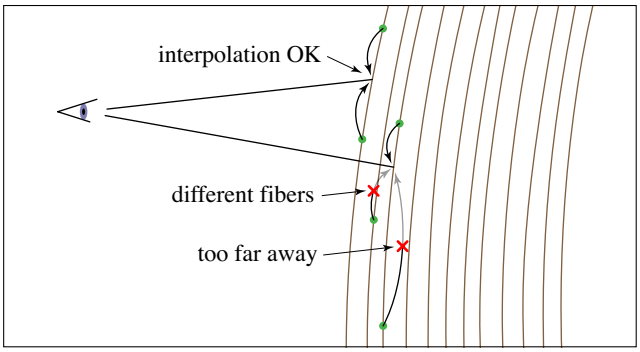

Figure 6: The fiber radiance cache. Cached radiance values are stored at points on fibers, and they can be reused for other sample points on the same fiber that are within a distance determined by the radius of the photon lookup used to compute them.

the same sets of photons. However, in hair there is no assumption of diffuse illumination, and the scattering functions of adjacent hairs can be quite different from one another (if the properties of the fibers vary, or if there is eccentricity and the hairs have different orientations). This generally prevents direct reuse of results from one hair when rendering another hair, even if it is only a fraction of a millimeter away. However, *along* the fibers we may expect the scattered radiance to vary as smoothly as the incident light distribution varies with respect to space, provided the amount of fiber twist is small over the typical distance of interpolation.

Taking these constraints into account, our method incorporates a *fiber radiance cache* that allows for reuse of previously computed scattered radiance values for additional rays that hit the same fiber. The idea is simple: each fiber has a list of radiance samples, and if a ray lands between two samples and is close enough to both, its radiance will be interpolated from those samples. If there are no nearby samples, its radiance is calculated using the photon map and the result is added to the cache. If there is exactly one nearby sample, radiance is calculated at a point farther along the fiber and cached, so the radiance at the original point can be interpolated from the cache. The maximum reuse distance for a cached radiance value is defined as a fixed fraction of the minimum photon map lookup radius used in computing that value; we have generally used the fraction 0.5, so that a sample can be interpolated only from samples whose lookup volumes all overlap by at least a full radius. Figure 6 illustrates the basic cache behavior, and the cache logic is detailed in pseudocode in Figure 7.

To prevent the radiance cache from occupying a lot of memory, the image is computed in blocks, and the cache is emptied before starting each block. The per-fiber sample lists are stored in a hash table by fiber ID, so only as many lists as there are visible fibers in one block will ever be created. In general, the cache performed very well in our test cases, reducing the number of photon map lookups per block by 90% or more. Further details on performance can be found in the results section.

4.4 Algorithm recap

We now summarize our new method in its entirety. Pseudocode is also included in Figure 7.

The particle tracing pass takes as parameters the total desired photon count and the mean distance d between photons along a ray. Paths are generated through the hair volume, using the cylinder model to generate the scattered direction at each interaction. After the first interaction, photons are deposited randomly along the path, one in every interval of length d along the path. The photons are generated even in empty space, so a bounding volume is

```
RenderEyeRay(ray):
    hair, t = IntersectScene(ray)   [t is a parameter along the hair]
    singleScatter = GetDirectIllumination(hair, t, ray.dir)
    multipleScatter = GetCachedRadiance(hair, t, ray.dir)
    return singleScatter + multipleScatter

GetCachedRadiance(hair, t, viewDir):
    sample0, sample1 = FindNeighboringSamplesOnHair(hair, t)
    if at least one sample is present and close enough
        if only one sample is present and close enough
            let sample0 be the valid sample
            tnew = ChooseNewSampleLocation(sample0, t)
            sample1 = (tnew, IlluminateHair(hair, tnew, viewDir))
            AddSample(hair, sample1)
        radiance = InterpolateSamples(sample0, sample1 , t)
    else   [no valid samples found]
        radiance = IlluminateHair(hair, t, viewDir)
        AddSample(hair, (t, radiance))
    return radiance

IlluminateHair(hair, t, viewDir):
    shadingPoint = hair.position(t)
    reflectedRadiance = 0
    for i = 1 to NUM_STABS
        atten, dir = ChooseScatteredDirection(hair, viewDir)
        inRadiance = PhotonMapLookup(shadingPoint, dir)
        reflectedRadiance += inRadiance * atten / NUM_STABS
    return reflectedRadiance

PhotonMapLookup(point, dir):
    photonList, r = LocatePhotons(point, dir, MAX_RADIUS)
    totalPower = sum of power of photons in photonList
    lookupMeasure = 4/3 * π^2 * r^5 / WEIGHT
    return totalPower / lookupMeasure
```

Figure 7: Pseudocode of the new rendering method.

used that extends at least the maximum lookup radius beyond the hair in all directions; photons are only deposited until the ray exits the volume.[3] This process is continued until the desired number of photons have been stored.

The rendering pass traces eye rays into the hair volume, and at the first intersection it computes direct illumination using the scattering model. It combines this with the scattered radiance from the hair at that point, which it first attempts to interpolate from the fiber radiance cache. If this is not possible, the illumination computation proceeds by generating a fixed number of directions according to the cylinder scattering procedure and performing a 5D photon map lookup for each direction. Each of these yields a radiance estimate, which are then averaged together weighted by the attenuation from the scattering procedure to get the result. We use enough directions to get a low-variance estimate that is then added to the cache, to be later interpolated for nearby calculations on the same fiber.

5 Results

To validate this new rendering technique, we developed a number of test scenes that include lighting of highly complex hair geometry from the front, from the back, and from a sharp-edged spotlight. In each case, we compare our result to a result from a Monte Carlo path tracer that generates paths using the cylinder scattering procedure and calculates direct illumination using the approximation from Marschner et al. [2003] at each interaction. In each scene, the hair fibers have identical properties: the radius is 41 μm, the eccentricity is 0.9, the azimuthal orientation is chosen randomly, and the color is reddish blond, with absorption coefficient $\sigma_a = (0.03, 0.11, 0.2)$ per hair radius in the spotlight scene, $\sigma_a = (0.045, 0.11, 0.2)$ in the backlit scene, and $\sigma_a = (0.03, 0.07, 0.15)$ in the frontlit scene. All photon mapped images were rendered with 64 eye rays per pixel for anti-aliasing. Details of the parameters used for the photon map in each of the scenes are listed in Table 1.

The first test scene consists of several thousand nearly parallel hair fibers illuminated perpendicularly by a spotlight 2 cm in diameter from a distance of 1 m. The camera is positioned 40 cm from the center of the hair, in a direction 15° toward the root of the fibers. Figure 8 shows the results of the path tracer and of our new method. Both results appear quite similar, displaying important features that result from multiple scattering. The presence of light outside the spotlight must be due to multiple scattering, and its spreading primarily across the fibers is consistent with the discussion in Section 3. The increased amount of light toward the top of the image (toward the camera) is indicative of the strong directionality of the scattered radiance field. While the new photon mapping method does introduce a small amount of blur, the fine features of individual hair fibers and the overall character of the image are both preserved. In the photograph at the right in Figure 8, you can see effects similar to those in the rendered images, although the hair is darker in color and less well aligned than the model used for rendering, which reduces the directionality and intensity of multiply scattered light.

Figure 9 shows another scene that cannot be accurately rendered without multiple scattering. A 25 cm long blond ponytail composed of 27,000 fibers hangs directly between a point light source and the camera. With direct illumination only, the hair appears unrealistic, with sharp shadows, little color, and opacity everywhere but at the edges. Including multiple scattering softens the sharp direct illumination and introduces a colorful translucent quality that varies with the density of the hair fibers. These additions greatly increase realism, similar to the photograph in Figure 1. This test case is very challenging for our approach, because the majority of the photons are deposited on the back side of the model and we calculate illumination on the front. In spite of this difficulty, our method performs very well, producing a low noise image that accurately conveys the contribution of multiple scattering. The final image in Figure 9 shows the result of our method using directional weight $w = 0$, meaning photons are selected based on position and not direction. The brightening at the sides of the ponytail, where the sheet of scattered light is aligned with the camera direction, is lost when directional variation is ignored in this way.

Finally, we set up a more typical scene in which the same ponytail is illuminated from a point 20° to the left of the camera. In Figure 10, we show the results of rendering with and without multiple scattering included. While the direct illumination image does convey the shape and texture of the ponytail, it fails to account for a significant amount of energy in the scene, which is evident in the added color, glow, and softness present in the path traced and photon mapped images. Here, the photon mapping tends to spread the very directional low order scattering over a slightly broader solid angle, causing some blurring of surface features, but the overall result is quite similar to the path traced result. The $w = 0$ photon mapping result, ignoring direction in the indirect radiance, appears much darker and less saturated than the correct result. This is because, in the correct solution, the camera direction is receiving more multiply scattered light than average.

For each of these scenes, our photon mapping approach produces effectively noise-free images 1–2 orders of magnitude faster than the equivalent path traced image can be produced. The fiber radiance cache has a profound impact on the running time of our algorithm, particularly for higher resolution, anti-aliased images such as these. As listed in Table 1, the cache hit rate was consistently high, leading to performance that scales extremely well as the number of

[3] If the bounding volume is not convex, the ray must still be traced to see if it re-enters the volume.

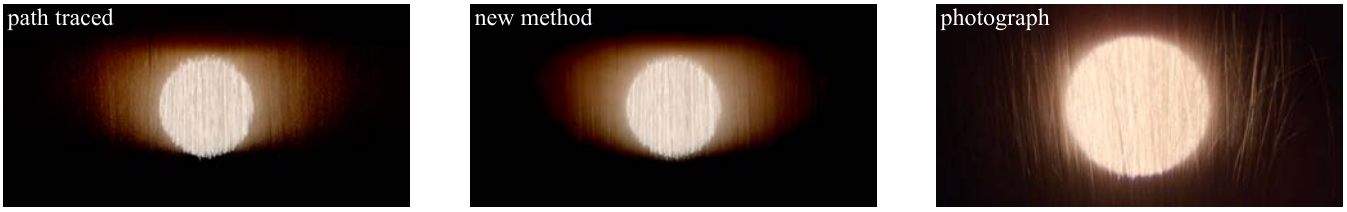

Figure 8: Blond hair illuminated by a spotlight, resulting in a diffused glow of multiply scattered light, and viewed from 15° above. Our method produces the same spread of illumination with lower noise in 2.2 hours than path tracing does in 90 hours. The photon map slightly blurs the indirect light, but the texture of the hair remains sharp. The photograph shows a similar glow, though not identical because of differences in the properties of the hair. These images are logarithmically scaled to compress dynamic range.

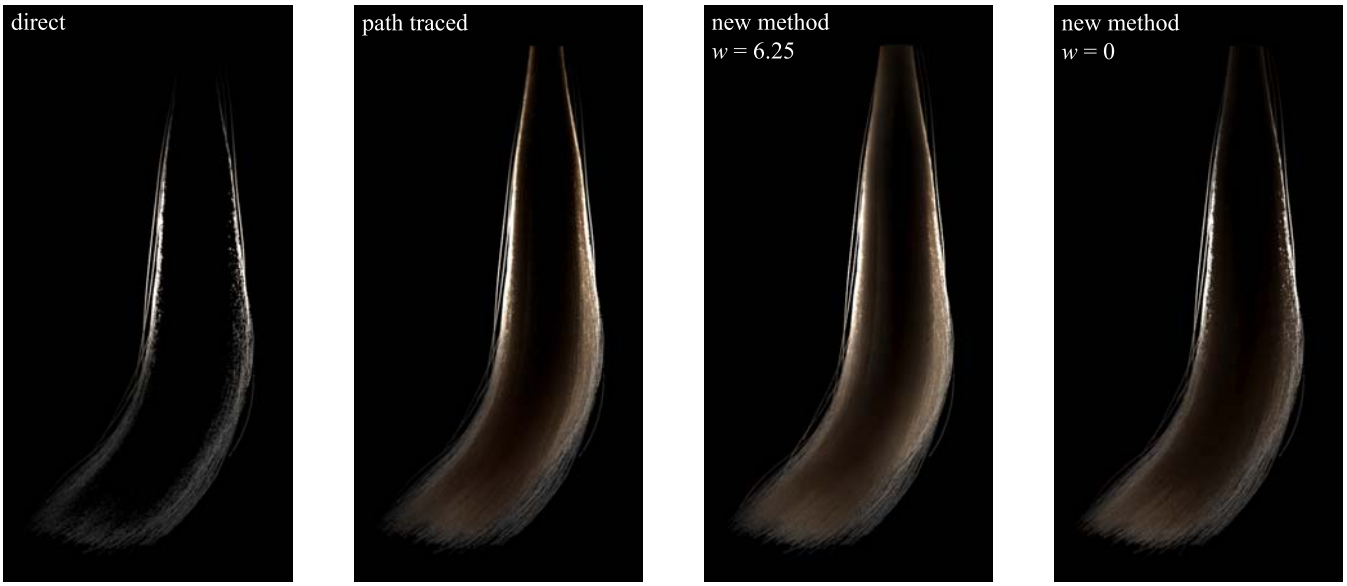

Figure 9: Renderings of a ponytail lit from behind. With direct illumination only, the hair appears too opaque, with a hard-edged highlight. Accounting for multiple scattering adds softness and a translucent quality, as seen in the results of a 100 hour path tracing and our method in 2.7 hours. The $w = 0$ image, which ignores directionality in the radiance field, produces the wrong intensity near the plane of the light source.

Figure 10: Renderings of a ponytail lit from the front. Direct illumination captures the surface highlight but has entirely the wrong color. Our new method correctly simulates the color and softness caused by multiple scattering in 2.5 hours, agreeing well with the 60 hour path traced result. The $w = 0$ result, which assumes multiply scattered light is isotropic, demonstrates the importance of maintaining direction in the scattered radiance.

Scene	Size	# Hairs	Photons	Spacing	Dirs	Number	$w\ (\frac{cm^2}{sr})$	Pass 1	Pass 2	Total	Hit %	Path Trace
Spot	800x400	13,000	80M	0.9 mm	75	75	0.81	0.6 hr	1.6 hr	2.2 hr	98	90 hr
Backlit	462x924	27,000	90M	3.0 mm	15	50	6.25	0.9 hr	1.8 hr	2.7 hr	94	100 hr
Frontlit	462x924	27,000	90M	2.0 mm	20	50	0.5625	0.8 hr	1.7 hr	2.5 hr	93	60 hr

Table 1: The various scene attributes, parameters to our method, and performance results. The Spacing column represents the mean distance between photons deposited along paths. Dirs is the number of directions in which the photon map is queried for each indirect illumination calculation, and Number is the number of photons gathered in each query. Pass 1 represents the particle tracing phase, and Pass 2 includes both the direct and indirect illumination calculations. Hit % represents the average cache hit rate at the given settings. All methods were implemented in Java as single-threaded applications and ran on a dual Intel Xeon 3.8 Ghz workstation.

pixels and rays/pixel increase. In all, our method represents an efficient and accurate way of including the contributions of multiple scattering in renderings of hair assemblies.

6 Conclusion

We have demonstrated that the very difficult problem of simulating multiple scattering in hair, previously only approachable using path tracing, can be simulated much more efficiently by using a two-pass particle tracing and density estimation approach. Our method builds on the idea of volumetric photon mapping for participating media, but with some important new modifications. These changes are made necessary by two factors. The density of stored photons needs to be proportional to radiance, not scattered radiance, because the map is being used to light geometric hairs, not to compute scattering in a volume. The generalization to a 5D position-direction space is required because of the optical nature of hair: light spreads anisotropically and remains quite directional even after several scattering events. This directionality, combined with the sharply peaked scattering function of a hair fiber, makes a smooth estimate in direction, as well as space, crucial to performance.

The general idea of using a smooth volumetric model for light scattered from complex geometry has implications that go beyond hair. Other scenes with densely packed scatterers in which multiple scattering is important (for example, a pile of soap bubbles appears white due to multiple scattering) may be rendered using very similar techniques. This idea can also be carried further, to using volumetric rendering methods for the scattering simulation as well as for reconstructing the radiance field.

In this work we have used commercial tools to generate the hair models. There is no reason to expect the geometric arrangement of strands in the resulting model to be similar to that in real hair, and this limits our ability to compare to measurements. Determining and simulating the typical geometric properties of real hair assemblies is required to get results that can be quantitatively validated against measurements.

7 Acknowledgements

The authors would like to thank Donivan Patwell for donating the hair samples photographed for this paper, and Andrew Butts for modeling the hair scenes that we rendered. Our hair renderer was built on the `skar` rendering system, developed at the Program of Computer Graphics, and we are grateful to the many contributors to that code over the years. We particularly thank Bruce Walter for his help with `skar`, and for many other valuable discussions. This research was supported by NSF CAREER award CCF-0347303, NSF grant CCF-0541105, and by an Alfred P. Sloan Research Fellowship.

References

CAMMARANO, M., AND JENSEN, H. W. 2002. Time dependent photon mapping. In *Rendering Techniques 2002: 13th Eurographics Workshop on Rendering*, 135–144.

HAVRAN, V., BITTNER, J., HERZOG, R., AND SEIDEL, H.-P. 2005. Ray maps for global illumination. In *Rendering Techniques 2005: 16th Eurographics Symposium on Rendering*, 43–54.

JENSEN, H. W., AND CHRISTENSEN, P. H. 1998. Efficient simulation of light transport in scenes with participating media using photon maps. In *Proceedings of ACM SIGGRAPH 98*, 311–320.

JENSEN, H. W., MARSCHNER, S. R., LEVOY, M., AND HANRAHAN, P. 2001. A practical model for subsurface light transport. In *Proceedings of ACM SIGGRAPH 2001*, 511–518.

JENSEN, H. W. 2001. *Realistic image synthesis using photon mapping*. A. K. Peters, Ltd., Natick, MA, USA.

KAJIYA, J. T., AND HERZEN, B. P. V. 1984. Ray tracing volume densities. In *Computer Graphics (Proceedings of ACM SIGGRAPH 84)*, vol. 18, 165–174.

KAJIYA, J. T., AND KAY, T. L. 1989. Rendering fur with three dimensional textures. In *Computer Graphics (Proceedings of ACM SIGGRAPH 89)*, vol. 23, 271–280.

KRIVANEK, J., GAUTRON, P., PATTANAIK, S., AND BOUATOUCH, K. 2005. Radiance caching for efficient global illumination computation. *IEEE Transactions on Visualization and Computer Graphics 11*, 5 (September-October), 550–561.

LAFORTUNE, E. P., AND WILLEMS, Y. D. 1995. A 5D tree to reduce the variance of Monte Carlo ray tracing. In *Rendering Techniques 1995: 6th Eurographics Workshop on Rendering*, 11–20.

LOKOVIC, T., AND VEACH, E. 2000. Deep shadow maps. In *Proceedings of ACM SIGGRAPH 2000*, 385–392.

MARSCHNER, S. R., JENSEN, H. W., CAMMARANO, M., WORLEY, S., AND HANRAHAN, P. 2003. Light scattering from human hair fibers. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2003) 22*, 3, 780–791.

STAM, J. 1995. Multiple scattering as a diffusion process. In *Eurographics Workshop on Rendering 1995*, 41–50.

VEACH, E., AND GUIBAS, L. J. 1995. Optimally combining sampling techniques for Monte Carlo rendering. In *Proceedings of ACM SIGGRAPH 95*, 419–428.

WARD, G. J., AND HECKBERT, P. 1992. Irradiance gradients. In *Rendering Techniques 1992: 3rd Eurographics Workshop on Rendering*, 85–98.

WARD, G. J., RUBINSTEIN, F. M., AND CLEAR, R. D. 1988. A ray tracing solution for diffuse interreflection. In *Computer Graphics (Proceedings of ACM SIGGRAPH 88)*, vol. 22, 85–92.

ZINKE, A., SOBOTTKA, G., AND WEHER, A. 2004. Photo-realistic rendering of blond hair. In *Vision, Modeling, and Visualization 2004*, 191–198.