Hair Occlusion: Rendering Hair-Like Objects with Global Illumination

CEM YUKSEL
Visualization Sciences Program
Department of Architecture

ERGUN AKLEMAN*
Visualization Sciences Program
Department of Architecture

College of Architecture
Texas A&M University

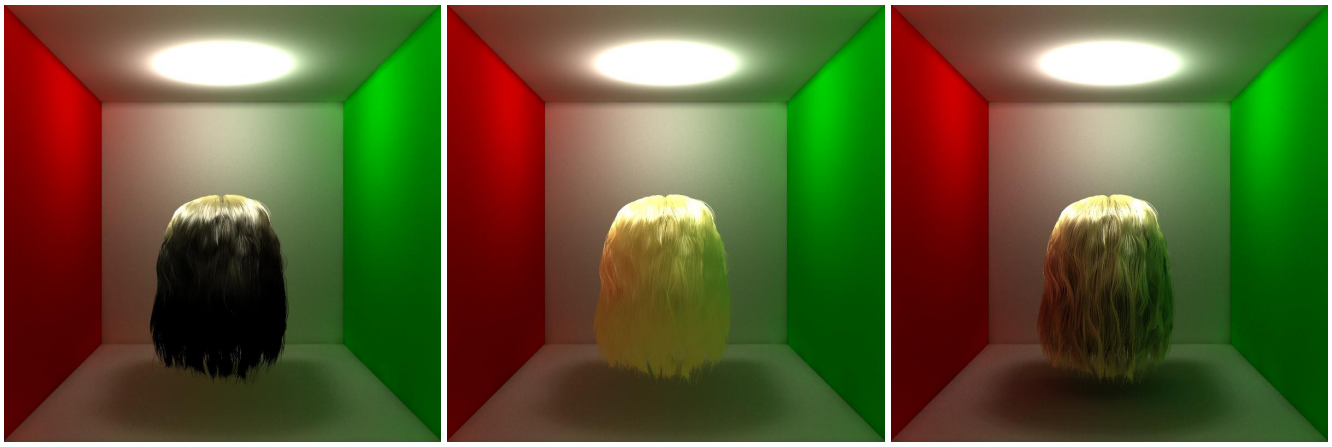


Figure 1: A Cornell-Box illuminated by a point light just below the ceiling. Shadows from the point light are computed using a deep shadow map. Photon mapping with final gathering is used for global illumination. **Left Image:** Hair is illuminated only by direct illumination. **Center Image:** Final gathering is included for hair without hair occlusion. **Right Image:** Our method, hair occlusion is included.

Abstract

In this paper we introduce *hair occlusion*, a general framework that is particularly suitable for global illumination of hair-like long, thin, and semi-transparent objects that can be represented by a series of line segments. Our framework is compatible with monte-carlo ray-tracing, and it can be integrated into the final gathering stage of photon mapping. Image based illumination techniques are also supported by this framework.

Our implementation of the framework allows us to render hair images with various realistic lighting simulations that are not supported by previous hair rendering methods. We also introduce optimization techniques to reduce the computational complexity, allowing several orders of magnitude faster hair occlusion calculations. Our framework is very simple, and can easily be implemented using any projection or ray-tracing based approach.

1 Introduction and Motivation

Simulating realistic lighting conditions using global illumination techniques is essential for realistic rendering. Hair rendering using only direct illumination in scenes rendered with global illumination techniques creates inconsistencies in the overall appearance. Global illumination techniques that are based on radiosity [Cohen and Greenberg 1985] and ray-tracing [Cook et al. 1984] cannot directly be applied to hair rendering, since even a single object with hair can be more complex than the rest of the scene, with a large number of extremely thin hair strands.

In this paper, we introduce the concept of hair occlusion and develop a framework for including hair occlusion in global illumination. For scenes that include hair-like structures, hair occlusion permits rendering consistent global illumination images by allowing shadows cast by indirect illumination over hair-like structures. Figure 1 shows the improvement with our method. The image on the left includes only deep shadow mapping and the hair does not seem to belong to the environment. The center image includes indirect illumination from the environment with hair occlusion. In this case, it is possible to see some color bleeding due to indirect illumination. But, the hair structure is still not visible. Only the image at the right that is created with our method shows the structure of the hair as a result of hair occlusion. Moreover, our method allows strong color bleeding from the environment also due to hair occlusion, and therefore, provides a consistent global illumination image.

Hair occlusion is not limited to indirect illumination computation. In our framework, hair can also be illuminated using image based lighting techniques that significantly improve the photo-realism of hair rendering. Previous shadow casting techniques for hair-like structures [Lokovic and Veach 2000; Kim and Neumann 2002] are applicable only for direct illumination of point or directional light sources. Skylight and area light sources, as well as indirect illumination cannot be computed by these methods. Figure 2 shows

*Corresponding Author: Visualization Sciences Program, Department of Architecture. Address: 216 Langford Center, College Station, Texas 77843-3137. email: ergun@viz.tamu.edu. phone: +(979) 845-6599. fax: +(979) 845-4491.

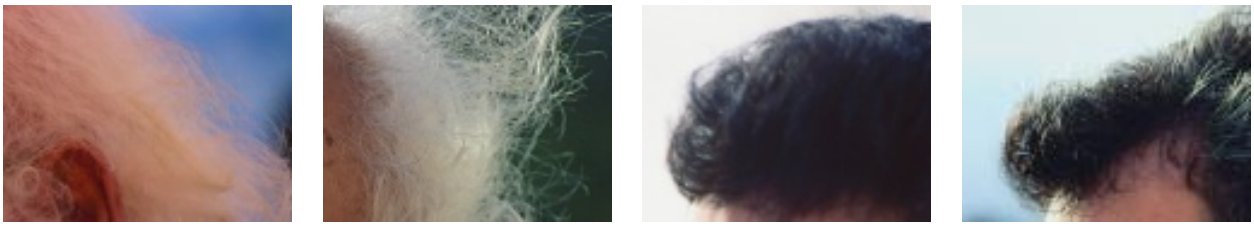


Figure 3: We can almost always can see subtle contrast changes in hair caused by indirect illumination in photographs. This effect can be seen even for black hair. The effect is important to perceive an image as hair even when no individual strand is visible as shown in these low resolution details from actual photographs that were taken under sunlight.

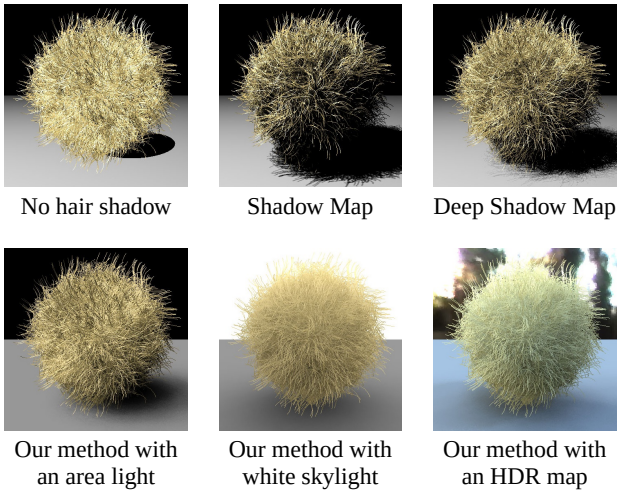


Figure 2: Comparison using environment maps. Area light here is given as an environment map.

a comparison of our technique with shadow mapping based methods. Note that shadow mapping provides a substantial improvement over the appearance as compared to the rendering without hair shadows. Deep shadow maps further improve this result by including the transparency term, however realistic lighting conditions can only be simulated using our framework.

The significant improvement that we gain by including hair occlusion comes from the fact that in real life our hair is usually illuminated from every direction and indirect illumination almost always exists. Thus, slightly-transparent hair strands cast shadows over each other causing subtle changes in contrast regardless of hair color, so that we can always perceive hair structure even in a very low resolution image (see Figure 3). To include such subtle changes in hair rendering, special effects companies rely on the expertise of lighting technical directors. Although, lighting technical directors can significantly improve the overall quality using more lights that illuminate hair with deep-shadow or shadow maps, their methods require extensive experience and are hard to generalize for realistic rendering.

In the next section we discuss our contributions and the previous work on hair rendering. Our theoretical framework is explained in section 3. Section 4 describes our procedure, and our simplifying assumptions that significantly reduce the rendering time is discussed in section 5. Our implementation is discussed in section 6. We show our results in section 7, concluding in section 8.

2 Contributions and Previous Work

Rendering hair has long been a challenge for computer graphics, since hair models often consist of many more than ten thousand hair strands, formed by a number of line segments that vary according to the length of the hair. Early methods used texturing techniques [Kajiya and Kay 1989; Goldman 1997] to avoid rendering of high complexity hair models. Recent advances in hair modeling, animation and rendering have made it possible to use hair in virtual environments [Lokovic and Veach 2000; Thalmann and Hadap 2000; Kim and Neumann 2002].

Our main contribution in this paper is to develop a novel *framework* for global illumination of hair-like structures. Our framework is compatible with monte-carlo ray-tracing, and so it can be included in the final gathering stage of photon mapping [Jensen 1996]. We can also render hair with image based lighting techniques [Debevec 1998], and include all basic computer graphics lights such as point, directional or area lights in cooperation with the previous shadow casting methods [Lokovic and Veach 2000; Kim and Neumann 2002].

Our framework is very simple, and can easily be implemented using any projection or ray-tracing based approach. In our implementation, we use a projection-based method to achieve hair occlusion with global illumination. We introduce simplifying assumptions to significantly reduce rendering time (from days to minutes) without loss of visual quality. Our implementation is fast enough to permit practical usage of global illumination for hair rendering.

3 Theoretical Framework

To develop a theoretical framework, we first assume that the image consists of a set of regions (pixels). If a given pixel region includes some hair segments, it will most likely be covered with a large number of semi-transparent hair segments that are projected to the pixel as shown in Figures 4 and 5. It is very unlikely that any one of the hair segments will completely cover the pixel, since hair strands are extremely thin. Even when a single hair segment completely covers the pixel, because of its transparency other hair segments will still be visible behind it.

Therefore, to simulate the look of hair on a pixel we need to consider the contribution from more than one hair segment. As a result, incoming radiance from a pixel region to the eye point will be an integral over all visible surfaces. For simplicity we assume that non-hair surfaces in the scene are completely opaque. Let x_e be the eye point and $\mathscr{A}$ be the set of all visible regions, let x_c denote the center of any infinitesimal visible area dA where $dA \in \mathscr{A}$, and θ_c is the direction vector of the light that reaches the eye from x_c. Total

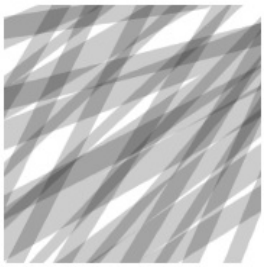


Figure 4: Hair strands projected to a pixel

incoming radiance to the eye point x_e through pixel p is

$$L_i(x_e \leftarrow p) = \int_{\forall dA \in \mathscr{A}} \gamma(x_e \rightarrow x_c) L_r(x_c, \theta_c) dA \qquad (1)$$

where $L_r(x_c, \theta_c)$ is the reflected radiance for the surface element dA in the direction of θ_c (Note that this equation does not include the illumination coming directly from light sources to the pixel). $\gamma(x_e \rightarrow x_c)$ is the percentage of radiance loss due to occlusion when the ray goes from x_c to x_e. Figure 5 shows a side view of this integral domain. In this formula,

$$\gamma(x_e \rightarrow x_c) = \exp\left(\int_{x_c}^{x_e} \ln\left(\gamma(x \rightarrow x + dx)\right) dx\right) \qquad (2)$$

for any given point x_c. The integral in equation 2 is associative and if γ represents a constant density it provides an exponential decay as expected. Moreover, the discrete versions of equations 1 and 2 provide associative color operations, consistent with standard volume rendering.

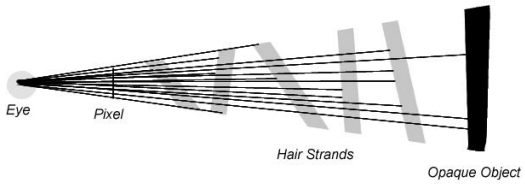


Figure 5: Side view of the visible regions.

For global illumination, the incoming radiance to a point in the scene should be computed similarly. In this case, the integral domain will be a unit sphere S instead of a pixel, leading to

$$L_i(x \leftarrow S) = \int_{\forall dA \in \mathscr{A}} \gamma(x \rightarrow x_c) L_r(x_c, \theta_c) dA \qquad (3)$$

This equation includes inter-reflections between hair segments since $L_r(x_c, \theta_c)$ is a function of $L_i(x \leftarrow S)$. Fortunately, we can safely ignore these inter-reflections between hair strands because of the high forward scattering property of hair [Marschner et al. 2003]. Based on this assumption, equation 3 is simplified by replacing L_r with $\Lambda(x, \theta_c)$, which is defined as the part of $L_r(x_c, \theta_c)$ that excludes inter-reflections between hair strands. Now, the integral is simplified since the function $\gamma(x \rightarrow x_c)$ comes only from hair and $\Lambda(x, \theta_c,)$ comes from the rest of the scene. As a result, we can redefine equation 3 as an integral of two functions

$$\Gamma(x, \theta) = \gamma(x \rightarrow x + t\theta) \qquad \text{and} \qquad \Lambda(x, \theta) = L_r(x + t\theta, \theta_c)$$

on the unit sphere. Here θ is a direction vector and $x + t\theta$ is the first non-hair surface seen from x in the direction of θ. We call Γ the occlusion function and Λ the illumination function. Now equation 3 becomes

$$L_i(x \leftarrow S) = \int_S \Gamma(x, \theta) \Lambda(x, \theta) d\theta \qquad (4)$$

This formalization leads to following observations regarding implementation. **(1)** When the point x is on an opaque surface, we can use just half of the sphere in computing equation 4. **(2)** The occlusion and illumination functions can be implemented separately using different methods. For instance, $\Lambda(x, \theta)$ can be implemented using ray tracing and $\Gamma(x, \theta)$ using a projection based method such as a Cube (i.e. two hemicubes). **(3)** $\Lambda(x, \theta)$ is the final gathering term that is computed to create the final image in photon mapping. Thus, the only additional price for including hair is the computation of the function $\Gamma(x, \theta)$.

Reflected radiance will be computed using $\Gamma(x, \theta, \phi)$ and $\Lambda(x, \theta)$ as

$$L_r(x, \phi) = \int_S F(x, \theta, \phi) \Gamma(x, \theta) \Lambda(x, \theta) d\theta \qquad (5)$$

where $F(x, \theta, \phi)$ is the BRDF combined with the geometry term, and ϕ is the direction from x to the eye point x_e. The $L_r(x, \theta)$ term will only be used in the computation of incoming radiance from pixel $L_i(x_e \leftarrow p)$ since we have ignored inter-reflections between hair strands by assuming a strong forward scattering.

4 Procedure

Based on this theoretical framework, our procedure consists of two stages. The first stage is the scan-conversion of the scene geometry (polygonal surfaces and hair line segments) onto the image plane as pixel fragments. In the second stage we compute the color of the pixel with illumination computation and shading.

4.1 Stage 1: Scan-conversion

In the scan-conversion stage we convert the 3D scene to fragments on the image plane. For any given pixel region we compute the discretized estimate of the integral

$$L_i(x_e \leftarrow p) \approx \sum_{n=0}^{N} a_n L_r(x_n, \phi_p), \qquad (6)$$

where N is the number of samples, x_n is the position of fragment n, a_n is the pixel coverage of the fragment, and ϕ_p is the direction from x_e to p. Equation 9 suggests that the integral region of a pixel can be represented with a set of transparent layers that cover the whole region as shown in Figure 6. Let N layers be handled starting from 0 at the nearest layer. Let the transparency and color associated with layer n be denoted by γ_n and C_n respectively. Then, if we choose

$$a_n = \prod_{i=0}^{n-1} \gamma_i \qquad \text{and} \qquad C_n = L_r(x_n, \phi_p)$$

the composited layers give the same result as equation 9. As a result, it is crucial to estimate the value of γ for each layer for a good approximation.

To compute γ values, we approximate hair segments intersecting with the pixel region as transparent layers (fragments) that cover the region completely as shown in Figure 6. Let the projected length of

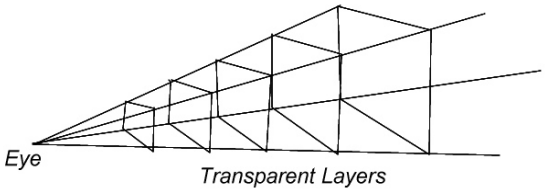


Figure 6: Computing a_n values by assuming hair segment as transparent layers.

a hair segment n in a pixel be denoted by l_n and the average distance of the fragment to the eye be z_n.

It is crucial to estimate the transparency of these layers to compute a good approximation of $L_i(x_e \leftarrow p)$. As we have discussed earlier, the hair segments cover only a portion of the pixel region. Let the thickness of hair segments be the same and be denoted by d. The total area of the hair segment in the pixel can be approximated as $l_n d/z_n$ by assuming that the projection has a rectangular shape. If the area of the pixel region is normalized to 1, the area of projected hair segment directly gives the ratio of the area covered by the hair segment. Let the transparency of a hair segment be denoted by t_n, then we can estimate the transparency of the layer associated with this particular hair segment as $(1-(1-t)l_n d/z_n)$.

It is important to estimate the value of t based on hair transparency. Let the actual length of the hair segment visible from the pixel region be given as L_n and the transparency of a unit hair segment be a constant denoted by σ, then based on equation 2, $t = \gamma_n^L$. As a result the transparency of the layer associated with hair segment n will be

$$\gamma_n = \left(1 - \frac{l_n d}{z_n}\left(1-\sigma_n^L\right)\right). \tag{7}$$

In practice for efficient computation, we record only the N smallest γ_n values in computing $L_r(z_n,\theta)$, where N is a user defined number. By increasing N we can improve the rendering quality with the cost of extra computation time for illuminating and shading the new layers.

4.2 Stage 2: Illumination and Shading

In stage 2 we compute an estimate to $L_r(x,\phi)$. We first subdivide the unit sphere into a set of spherical regions Ω_n such that the intersection of any two solid angles is empty and the union of all solid angles gives us the complete unit sphere. Then we discretize occlusion and illumination functions as follows:

$$\overline{\Gamma(x,\Omega_n)} = \frac{\int_{\Omega_n}\Gamma(x,\theta,\phi)d\theta}{\Omega_n} \tag{8}$$

$$\overline{\Lambda(x,\Omega_n)} = \frac{\int_{\Omega_n}\Lambda(x,\theta,\phi)d\theta}{\Omega_n} \tag{9}$$

Using the above functions, we compute

$$L_r(x,\phi) = \sum_{n=0}^{N} F(x,\Omega_n,\phi)\overline{\Gamma(x,\Omega_n)}\ \ \overline{\Lambda(x,\Omega_n)}$$

Since only F is based on the choice of the shading model, the problem reduces to computing average occlusion $\overline{\Gamma}$ and average illumination $\overline{\Lambda}$ for the fragment. The computation of $\overline{\Lambda}$ can be done exactly like final gathering is done in photon mapping.

For computing the average occlusion function $\overline{\Gamma}$, we assume that each line segment can be considered a transparent layer that covers a solid angle. It is important to identify the transparency of these layers. In this case, transparencies of all the hair segments that can be projected to a solid angle γ_n will contribute to the final transparency $\overline{\Gamma}_i$ of a given solid angle i as

$$\overline{\Gamma}_i = \prod_{n=0}^{N}\gamma_n \tag{10}$$

where N is the number of hair segments that are projected to the solid, and γ_n is as in equation 7.

5 Simplifying Assumptions

The computation we have described in the previous section is done for every shading point on the image and is bounded by KN, where K is the number of pixels and N is the number of allowable layers in a pixel. Moreover, a simple hair consists of a large number of hair strands and each strand represents a number of line segments. Therefore, we need to improve the computation speed of $\overline{\Gamma}$ as much as possible. We introduce two major approaches to speeding up computation based on level of detail (LOD) methods. Using these methods, it is possible to increase the computation speed by several orders of magnitude without having visible effects on the final results.

5.1 Level of Detail Shadowing by Eliminating Higher Order Terms

Equation 10 allows us to make this speedup. Note that γ_n is in the form

$$\gamma_n = 1 - a_n + a_n b_n$$

where $a_n = \frac{l_n d}{z_n}$ and $b_n = \sigma^{L_n}$ and both a_n and b_n are in the interval $[0,1]$. Thus, equation 10 can be rewritten as

$$\overline{\Gamma}_i = \prod_{n=0}^{N}(1-a_n+a_n b_n). \tag{11}$$

In this equation, higher order terms will quickly approach zero. Therefore, the linear term essentially determines the shadowing effect. If we ignore higher order terms, the effect of the remaining terms is

$$\overline{\Gamma}_i \approx 1 - \sum_{n=0}^{N} a_n + \sum_{n=0}^{N} a_n b_n. \tag{12}$$

Normally natural hair clusters due to electrostatic forces between hair strands. Since each hair strand in a cluster shares approximately the same a_n and b_n values, we can use only one hair strand to represent each cluster without affecting the linear term. Let us assume that hair consists of N/k clusters, with each cluster having k hair strands. Using one representative for each cluster and scaling a_n by k we can write an equation that is equivalent to equation 12:

$$\overline{\Gamma}_i \approx \left(1 - \sum_{n=0}^{N/k} k a_n + \sum_{n=0}^{N/k} k a_n b_n\right) \tag{13}$$

In other words, by using one hair per cluster our linear term stays approximately the same. All terms higher order than N/k are eliminated. We still have higher order terms up to N/k, but they are approximate. In practice, we get almost the same result if, when we reduce the number of hair segments k times, we adjust γ_n as follows:

$$\gamma_n = 1 - \frac{kl_n d}{z_n} + \frac{kl_n d\sigma^{L_n}}{z_n}$$

In our experiments we found that using this assumption it is possible to retain only a very small fraction of hair strands to get the same visual quality as when using all strands.

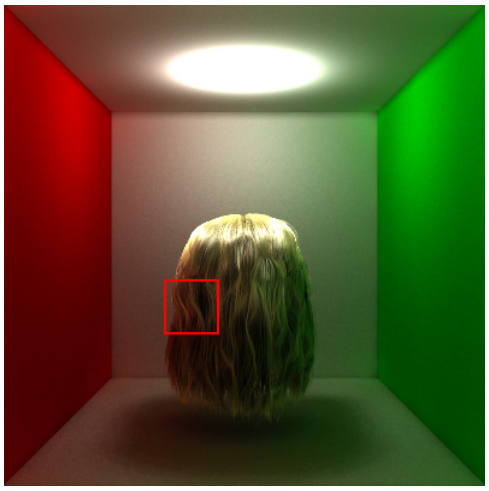
5.2 View Dependent Level of Detail Shadowing

Using a view dependent level of detail approach, it is possible to eliminate z_n from the terms

$$\sum_{n=0}^{N} \frac{a_n}{z_n} \quad \text{and} \quad \sum_{n=0}^{N} \frac{a_n b_n}{z_n}$$

by simply using $1/z_n th$ of hair segments that are approximately at the same distance. For instance, we use every other hair segment of the hair segments that are at an approximate distance $\lfloor z+0.5 \rfloor = 2$; we use every third hair segment of the segments with an approximate distance $\lfloor z+0.5 \rfloor = 3$ and so on... In this way, by eliminating the term z_n, we do much less computation. We multiply L_n by z_n to make the results approximately the same.

With our simplifying assumptions, we can speed up the computation by several orders of magnitude. In fact, without the assumptions these images cannot be created in a reasonable time frame. We can, therefore, make comparisons only with very low-resolution images. As shown in Figure 7, for a 32×32 image our assumptions reduce the computation time from approximately 9.5 hours to 2.5 minutes on a Pentium 4 3GHz processor, without any loss of quality.



(A)

(B) (C) (D) (E)

Figure 7: (A) 400×400 image that is rendered in 25 minutes with our assumptions (Using 10% of hairs and view dependent LOD). (B) A 32×32 region from image (A) rendered in 9.5 hours without our assumptions. (C) The same region rendered in 2.5 minutes with our assumptions. Note that we have an approximately 230 times speed-up without a visible change in quality. (D) The difference image $|B-C|$. The maximum difference is 16 in 256 color levels. Average differences are 5 in Red, 3 in Green and 2 in Blue. (E) Exaggerated difference by multiplying the difference image by 16. Assuming that the speed up will be the same for every region of the image, to create (A) would take at least 4 days without the assumptions. However, the actual computation might take much longer since speed-up for pixel regions that are away from hair is much larger than 200 due to view dependent LOD.

6 Implementation

In our implementation of the framework, we first project the polygonal surfaces in the scene and line segments that form the hair strands onto the image plane pixels as fragments. Then we shade these fragments and combine them to find the final color of the pixel. Illumination and hair occlusion is computed during this shading stage. For computing the illumination function $\overline{\Lambda}$ we use monte-carlo ray-tracing with an importance sampling just like final gathering in photon mapping [Jensen 1996]. We use a projection based algorithm to evaluate γ_n and $\overline{\Gamma}$.

Hair occlusion is computed differently for non-hair surfaces and hair fragments. This is because opaque surfaces can only be illuminated from directions above the surface normal, whereas hair may as well be illuminated from behind. To compute $\overline{\Gamma}$ for non-hair surfaces, we project hair segments onto the hemisphere that is centered on the surface point and aligned with the surface normal. Then we sample the projection to identify the directions occluded by the projected hair segment. When computing $\overline{\Gamma}$ for hair fragments, we use two hemispheres facing in opposite directions that are arbitrarily oriented.

For sampling on the hemisphere surface we subdivide the hemisphere into faces of approximately equal areas. The subdivision method that we use results in an igloo like structure (see Figure 8).

When a hair segment is projected onto the hemisphere, we identify the igloo faces covered by the hair segment, and apply the occlusion effect of the hair segment onto those faces.

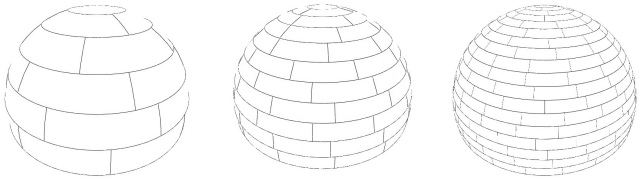


Figure 8: Igloo models in different resolutions.

This procedure is somewhat difficult, since line segments become curves on the hemisphere surface. To avoid curve computations, we first project the igloo onto the plane tangent to the top of the hemisphere (Figure 9). (For projecting the igloo onto the tangent plane we use a simple perspective projection centered on the center of the igloo. By doing so, we neglect the contributions from horizontal directions. Using 32 bit single precision floating point numbers, the neglected horizontal angles are smaller than 10^{-37} degrees.) Hair strand segments are then projected onto this plane as line segments using a simple perspective projection, and checked for intersections with hemisphere faces on the plane. The area of each face that is covered by the line segment increases the occlusion value of the face.

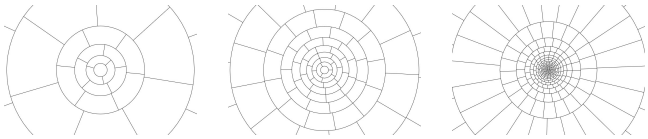
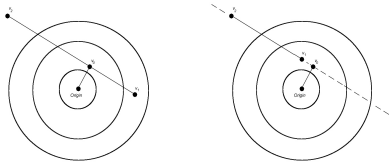


Figure 9: Projections of Igloos in Figure 8 on the tangent plane.

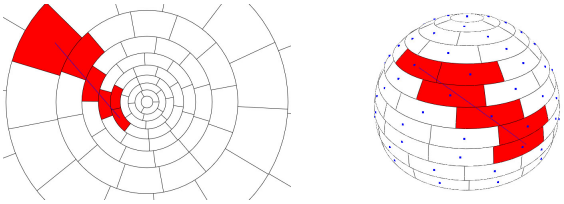
The projection of the igloo to the infinite plane tangent to the top of the hemisphere, gives us concentric circles as shown in Figure 9. It is easy to calculate the intersections of projected line segments with these concentric circles. Once intersections with two consecutive nested circles are computed, it is simple to identify intersections of the line segment with the faces within the given circular region. As seen in Figure 10, there are two cases that can easily be solved with ray-circle intersections (see appendix for details.). Note that intersections with two consecutive nested circles give us the first and last face that the line has intersected while passing through the circular region. To find the intersection points of the lines with those faces, we intersect this line with the line segments that separate the faces.



Case 1 Case 2

Figure 10: Intersection of a line segment with nested circles.

For occlusion, we assume each hair segment can be considered as a transparent layer that covers one face of the igloo as shown in Figure 11. We compute γ_n as in equation 7. Once we have all γ_n values, we obtain $\overline{\Gamma}$. Now, we know both $\overline{\Gamma}$ and $\overline{\Lambda}$, and we can shade and compute the color of the given fragment.



(a) (b)

Figure 11: A line segment that creates transparent layers on igloo faces.

7 Results

We have performed a number of rendering experiments using our approach. Figure 12 shows the result of illumination using three colored area lights and with HDR maps. The simplifying assumptions come from the fact that the real hair strands usually form clusters. However, it is important to note that the hair model in Figure 12 is completely random, but the assumptions still hold.

Figure 13 show additional comparisons with deep shadow maps in an environment globally illuminated by photon mapping. Notice that our approach provides a favorable result even for straight or dark hair. Figure 14 shows the effect of transparency on hair occlusion. Figure 15 shows a hairy teapot illuminated by a white skylight and Figure 16 shows details from Figure 1 and 13.

We use the Kajiya-Kay shading model [Kajiya and Kay 1989] for hair fragments, though it can easily be replaced by a more realistic BRDF function [Marschner et al. 2003] since the computation of $\overline{\Gamma}$ and $\overline{\Lambda}$ are independent of the shading model.

Even in very uniform lighting conditions, hair self shadows help to see the 3D structure of the hair, which would otherwise look flat. Moreover, animated sequences show that we do not get any artifacts or flickering in consecutive frames since our random strand segment selection is deterministic.

8 Conclusions

In this paper, we have presented *hair occlusion*, a framework for rendering hair like structures with global illumination. Our results show that this framework can effectively be used for illuminating hair-like structures by both image based illumination and photon mapping. We also introduced simplifying assumptions that significantly reduce the rendering time.

We used a projection method for computing hair occlusion. However, our framework may also be implemented using both ray-tracing and other projection based methods like hemicube [Cohen and Greenberg 1985].

References

COHEN, M. F., AND GREENBERG, D. P. 1985. The hemi-cube: A radiosity solution for complex environments. In *Proceedings of SIGGRAPH 1985*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 31–40.

COOK, R. L., PORTER, T., AND CARPENTER, L. 1984. Distributed ray tracing. In *Proceedings of SIGGRAPH 1984*, ACM

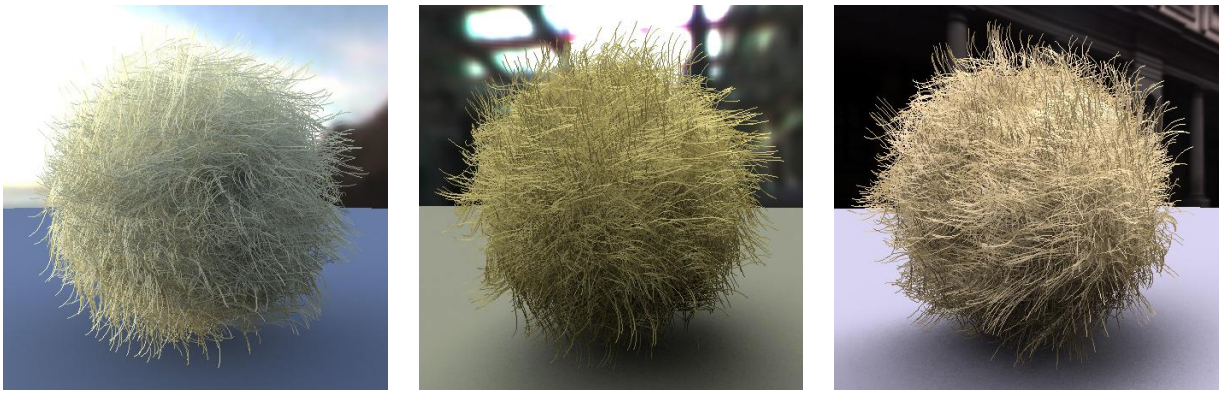


Figure 12: Hair model illuminated by a HDR maps (from www.debevec.org) and three area lights.

Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 137–145.

DEBEVEC, P. 1998. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proceedings of SIGGRAPH 1998*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 189–198.

GOLDMAN, D. B. 1997. Fake fur. In *Proceedings of SIGGRAPH 1997*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 127–134.

JENSEN, H. W. 1996. Global Illumination Using Photon Maps. In *Rendering Techniques '96 (Proceedings of the Seventh Eurographics Workshop on Rendering)*, Springer-Verlag, 21–30.

KAJIYA, J. T., AND KAY, T. L. 1989. Rendering fur with three dimensional textures. In *Proceedings of SIGGRAPH 1989*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 271–280.

KIM, T.-Y., AND NEUMANN, U. 2002. Interactive multiresolution hair modeling and editing. In *Proceedings of SIGGRAPH 2002*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 620–629.

LOKOVIC, T., AND VEACH, E. 2000. Deep shadow maps. In *Proceedings of SIGGRAPH 2000*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 385–392.

MARSCHNER, S., JENSEN, H. W., CAMMARANO, M., WORLEY, S., AND HANRAHAN, P. 2003. Light scattering from human hair fibers. In *Proceedings of SIGGRAPH 2003*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 780–791.

THALMANN, N. M., AND HADAP, S. 2000. State of the art in hair simulation. In *International Workshop on Human Modeling and Animation*, Korea Computer Graphics Society, International Workshop on Human Modeling and Animation, 3–9.

A Appendix

This appendix shows how we have implemented the igloo structure and how we compute intersections with line segments on the tangent plane.

A.1 Making an Igloo with an Iterated Procedure

Our iterated subdivision procedure allows creating igloos in any given resolution such as the ones shown in Figure 9. We will explain our procedure in a spherical coordinate system (r, θ, ϕ). Starting from a top circle a new circular region is created in each iteration. The sizes of these circular regions are chosen in such a way that each circular region can be further divided into equal area rectangular regions.[1] Let C denote the user-defined area. Using this area, we first compute the top circle. For small values of C, its area can approximately be given as $C = \pi\theta_0^2$. Therefore, we can choose $\theta_0 = \sqrt{C/\pi}$.

Let N denote the user defined number of circles along θ direction, then we compute approximate $\Delta\theta = (\pi/2 - \theta_0)/N$.

The iteration starts from $n = 1$. Let θ_n denote the total angle change in the n^{th} iteration. Using θ_{n-1}, we compute the total area of the circular region that will be created in the n^{th} iteration as $A_n = \pi(Sin(\theta_{n-1} + \Delta\theta) + Sin(\theta_{n-1}))\Delta\theta$. Our goal is to separate this circular area into k_n equal parts such that C approaches $\frac{A_n}{k_n}$.

As a result, we can simply choose

$$k_n = \lfloor \frac{A_n}{C} + 0.5 \rfloor$$

and

$$\theta_n = \theta_{n-1} + \frac{k_n C}{\pi(Sin(\theta_{n-1} + \Delta\theta) + Sin(\theta_{n-1}))}.$$

To create each rectangle boundary, we start from a random angle ϕ_n and divide the circumference into k_n equal parts.

In the last iteration, we cannot adjust θ_n freely since $\theta_n = \pi$. Therefore, we will not be able to get an exact integer value for k_n. Thus, we will have an error. Fortunately, this error is negligible in practice.

A.2 Computing Intersections with Line Segments

As we discussed earlier, we project hair strands to the infinite plane. Each hair strand consists of a number of line segments $L(v_1, v_2)$ where v_1 and v_2 denote two endpoints of the line segment, and we assume that $|v_1| \leq |v_2|$. Let $n_0 = \frac{v_2 - v_1}{|v_2 - v_1|}$ be a unit vector in the

[1] These are not really rectangles in the spherical domain. Their edges are circular.

No global illumination on hair
(deep shadow map only)

Global illumination on hair
(no hair occlusion)

Our method
(hair occlusion)

Figure 13: Cornell-Box with three different hair models (curly, straight, and dark) illuminated by a point light just below the ceiling. Shadows from the point light are computed using a deep shadow map. Photon mapping with final gathering is used for global illumination. **Left Images:** Hair is illuminated only by direct illumination. **Center Images:** Final gathering is included for hair without hair occlusion. **Right Images:** Our method, hair occlusion is included.

Figure 14: A lock of hair with changing transparency.

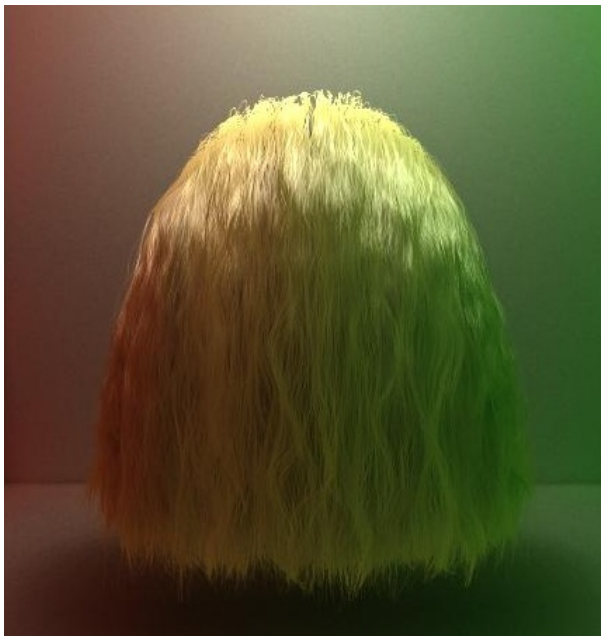
Figure 15: Hairy Teapot.

direction of the line. To simplify the intersection of the hair line segments with nested circles, we will introduce a new point. Let $v_0 = v_1 - (n_0 \bullet v_1)n_0$ be the point closest to the origin on the infinite line that includes the line segment $L(v_1, v_2)$. As seen in Figure 10, there are two cases:

- **Case 1.** $v_0 \in L(v_1, v_2)$: In this case, we separate the line into two pieces starting from v_0 as
  Line 1: $v = v_0 + n_0 t; \quad 0 \leq t \leq |v_2 - v_0|$
  Line 2: $v = v_0 - n_0 t; \quad 0 \leq t \leq |v_1 - v_0|$
- **Case 2.** $v_0 \notin L(v_1, v_2)$: In this case, we use the
  Line 1: $v = v_0 + n_0 t; \quad |v_1 - v_0| \leq t \leq |v_2 - v_0|$

Using a line equation starting from the nearest point to the origin, we greatly simplify the intersection with nested circles. Let r_i denote the radius of nested circle *i*, then the circle can be represented by the implicit equation $v \bullet v = r_i^2$. Combining the line equations and the circle equations we find $t = \mp\sqrt{r_i^2 - v_0 \bullet v_0}$. Based on this equation, the algorithm becomes extremely simple:

- **Case 1.** Start from the first circle in which $r_i^2 > v_0 \bullet v_0$ then
  - Find two intersection points $t_{1,2} = \mp\sqrt{r_i^2 - v_0 \bullet v_0}$ that correspond to line until $r_i^2 > v_1 \bullet v_1$, then
  - Find only one intersection point $t_1 = \sqrt{r_i^2 - v_0 \bullet v_0}$ until $r_i^2 > v_2 \bullet v_2$.
- **Case 2.** Start from the first circle in which $r_i^2 > v_1 \bullet v_1$ then
  - Find only one intersection point $t_1 = \sqrt{r_i^2 - v_0 \bullet v_0}$ until $r_i^2 > v_1 \bullet v_1$.



Curly hair from Figure 13

Straight hair from Figure 13

Hair model from Figure 1

Dark hair from Figure 13

Figure 16: Hair rendered with our method: Details from Figures 1 and 13.