# Quick Image-Based Lighting of Hair

Ivan Neulander*
Rhythm and Hues Studios

We present a fast, approximative solution for image-based lighting of curve-based hair, capturing both diffuse and specular reflection with occlusion. Our technique draws on and extends a hair self-shadowing model originally developed for traditional point-source lighting.

## Hair Model

We model each hair curve using a set of control points that define a spline in world space. Each control point carries attributes that include world-space position and four floats needed by our occlusion model. Other attributes, such as thickness and color, are useful but not relevant to our technique.

Our lighting model is based on [Kajiya and Kay 1989], which treats each hair as an infinite cylinder. Because image-based lighting must consider numerous light directions, our diffuse and specular models are summations of the Phong diffuse and specular model over multiple shading normals, distributed orthogonally to the curve tangent. Whereas Kajiya analytically folds these summations into simple functions of the tangent, we cannot do so because each element in our summations is occlusion-weighted. For each diffuse and specular shading sample, we perform multiple lookups into the IBL texture, with an occlusion approximation per lookup. We restrict diffuse and specular sampling to a plane and cone of directions around the tangent by prefiltering the IBL texture using a pair of cosine kernels. This strategy eliminates cosine weighting at render time and vastly reduces the number of shading samples, with an acceptable loss in accuracy.

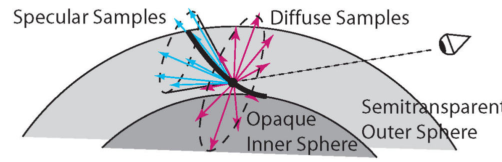
## Occlusion Model

We make use of an occlusion model proposed by [Neulander and van de Panne 1998], which places each hair shading point within a virtual sphere of homogeneous semitransparent material. This model involves storing two additional parameters per hair control point: a unit occlusion normal vector $\mathbf{N_O}$, and an occlusion height scalar $h_O$. With these, we can compute the distance from any point on the hair to the sphere's surface along any direction. An exponential based on this distance defines the point's fractional visibility along that direction. Given a unit incident light vector $\mathbf{L}$, and $\mathbf{N_O}$ and $h_O$ as above, the fractional visibility scalar is given by the expression $e^{-\rho(-h_O \mathbf{N_O}\cdot\mathbf{L}+\sqrt{1-h_O^2(1-\mathbf{N_O}\cdot\mathbf{L}^2)})}$ for some density constant $\rho$.

We extend the above model to include an opaque inner sphere centered at the same point as the above outer sphere. Rays intersecting the inner sphere are always considered fully occluded. This is useful for modeling shorter hair, which can be abruptly shadowed by skin. The inner sphere is quickly tested for intersection using the conditional $h_O^2(1-\mathbf{N_O}\cdot\mathbf{L}^2) < r^2$, where $r$ is the ratio between the inner and outer spheres' radii.

There are several ways to compute the occlusion normal and height for each control point. When the hair lies relatively close to skin geometry, the surface normal of the nearest point on the skin can serve as $\mathbf{N_O}$ and the normalized distance from this point can be used for $h_O$.

*email: ivan@rhythm.com




A more accurate solution for longer hair is achieved by using a ray tracer to compute volumetric ambient occlusion with bent normals (over a full sphere of directions) at each control point. While this can be expensive, it is a preprocess that does not affect subsequent rendering speed. Moreover, the ray tracing is limited to the set control points; during shading, we interpolate $\mathbf{N_O}$ and $h_O$ based on the values of nearby control points.

When animating deforming hair, we can accurately compute $h_O$ and $\mathbf{N_O}$ values for only one representative frame, and then keep the former fixed while rotating the latter based on the skin deformation at each frame. This procedure sacrifices accuracy for speed but works well in practice, especially with stiff hair.

## Conclusions and Further Work

Our technique is simple and well suited to GPU implementation. Its main limitation is its assumption of a locally spherical, homogeneous occluding medium, which precludes accurate shadowing. We are working to solve this by extracting bright areas of the IBL texture into explicit directional lights with traditional (e.g. depth map) shadowing, and applying our method to the residual texture from which the contribution of these lights has been subtracted.

Figure 1: A furry torus lit using HDRI images (courtesy of Paul Debevec). Rendered at 1024x768 resolution in 80 seconds. $\mathbf{N_O}$ and $h_O$ were quickly derived from the torus "skin", without ray tracing.

## References

KAJIYA, J. T., AND KAY, T. L. 1989. Rendering fur with three dimensional textures. In *SIGGRAPH '89 Proceedings*, Addison Wesley, vol. 23, ACM SIGGRAPH, 271–280.

NEULANDER, I., AND VAN DE PANNE, M. 1998. Rendering generalized cylinders with paintstrokes. In *Graphics Interface*, 233–242.