A Self-Shadow Algorithm for Dynamic Hair using Density Clustering

Tom Mertens* Jan Kautz† Philippe Bekaert* Frank Van Reeth*

Limburgs Universitair Centrum* Massachusetts Institute of Technology†

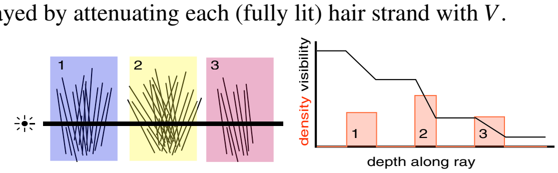
Introduction

We present a novel approach to rendering self-shadows for dynamic hair. Using programmable consumer graphics hardware, thousands of moving, individual hair strands can be rendered at interactive frame rates. Since we do not impose any constraints on the hair — hair is defined as a collection of line segments — any hair style can be given as input. The key innovation in our approach is the use of clustering.

Similar to previously presented methods [Lokovic and Veach 2000; Kim and Neumann 2001], the scene is rendered from the light source view, and a 1D visibility function $V(s)$ is constructed for each pixel. It encodes the fraction of visible (emitted) light in function of depth. The novelty of our algorithm lies in the way the visibility function V is constructed. In *deep shadow maps* [Lokovic and Veach 2000], for each shadow map pixel, many rays are pierced through the hair geometry while recording all intersections. From this data, a high resolution visibility function is computed, and finally compressed to facilitate efficient reconstruction. Obviously, such an approach remains too complicated for a graphics hardware implementation. In contrast to deep shadow maps, we perform both the intersection and compression at the same time.

Algorithm

We consider hair geometry as a *density field*. First, a density function d is computed along a pixel's line of sight. Integrating d will yield the visibility function: $V(s) = e^{-\int_0^s d(t)dt}$. Finally the hair is displayed by attenuating each (fully lit) hair strand with V.



In the above figure, the construction of V is depicted: on the left hand side, a ray pierces 3 bundles of hair in the input hair model; on the right the corresponding visibility function is shown. We rasterize all hair primitives, i.e. lines, into the frame buffer. Each rasterized fragment represents an intersection with the corresponding line of sight, and has a certain opacity value associated with it. All the intersection data, including opacities, are "summarized" into K clusters using the *k-means* method ($K = 3$ in the figure). The clusters are indicated by differently colored rectangles in the figure.

Next, the total "opacity weight" of each cluster is accumulated, effectively constructing a non-uniform histogram of the opacity density along the line of sight. The histogram corresponds to d (shown in red on the right). The width of the histogram bins are chosen proportional to the standard deviation of each cluster. Integrating d results in a piecewise linear approximation of the integral in V. In order to optimally utilize the 4-vector format in current graphics processors, we set K to 4. To avoid aliasing artifacts, filtering is mandatory. In our approach we render pre-filtered lines using a technique similar to "billboards" for particles: a textured screen space rectangle is constructed over each line segment. The texture contains a pre-filtered line profile. The amount of filtering controls the smoothness of the shadow.

Compared to Kim et al. [Kim and Neumann 2001], we achieve similar quality while rendering times are an order of magnitude faster. Kim et al.'s method renders N slices of the hair volume in order to construct V. For high quality results, N can run up to 80 [Kim and Neumann 2001]. While the rendering passes in our method are more complex, we only need O(K) passes, as opposed to O(N).

Results

We achieve interactive frame rates for animated hair models and dynamic lighting. E.g., the top figure shows a model consisting of 3600 hair strands (210K hair segments), running at 3Hz on an ATI Radeon 9800XT graphics board. Due to the absence of a certain extension on current hardware, we expect this performance to increase dramatically (4×) on next generation hardware. From left to right, the figure below shows an unshadowed and shadowed rendering of hair, indicating the tremendous impact that self-shadows have on the appearance. Finally, an image of curly hair is shown to illustrate rendering of a more complicated hair style.

References

KIM, T.-Y., AND NEUMANN, U. 2001. Opacity Shadow Maps. In *Proceedings of the Eurographics Workshop on Rendering*, 177–182.

LOKOVIC, T., AND VEACH, E. 2000. Deep Shadow Maps. In *Proceedings of SIGGRAPH 2000*, 385–392.

*{tom.mertens,philippe.bekaert,frank.vanreeth}@luc.ac.be

†kautz@graphics.csail.mit.edu