

Rendering Microgeometry with Volumetric Precomputed Radiance Transfer

John Kloetzli

February 14, 2006

Although computer graphics hardware has made tremendous advances over the last few years, there are many things which are still not feasible to render in realtime. One of these is the subset of surfaces referred to as 'microgeometry,' where many small individual pieces combine to create a complex surface. Microgeometry is a problem for typical polygonal rendering systems, the method used in the vast majority of realtime rendering, because of the large number of triangles needed to accurately represent microgeometry and the complex optical effects that these surfaces can possess. I propose a new technique for rendering microgeometry in realtime with realistic lighting effects through a unique combination of realtime and offline rendering methods. It consists of a volumetric microgeometry rendering system based upon the Texel¹ rendering method [Kajiya and Kay 1989], a concentric shell surface rendering method [Lengyel et al. 2001], an approximation of global illumination based upon Precomputed Radiance Transfer [Sloan et al. 2002], and a local lighting model.

1 The Challenges of Microgeometry

Even without the stringent restrictions of the realtime computer graphics environment, rendering realistic microgeometry is a difficult problem. The term 'micro' implies that there is very high frequency activity present in surfaces of this type, but 'geometry' then further describes the material as concrete in nature (each peice is still visible). A good example of a surface of this type is fur, where the complexity of the surface is very high but individual hairs are still distinguishable. Obviously, traditional polygonal methods of rendering bog down when applied to surfaces of this type, because of both the number of triangles needed to accurately represent these surfaces and the complexity of light interaction between them.

¹This is not to be confused with a texel (lowercase), which is one element of a texture. Although the terms are related, the Texel is a more specific term associated with a particular rendering method and will always be capitalized in this document.

One approach that was developed to solve the problem was the Texel rendering system [Kajiya and Kay 1989], which discarded polygonal meshes entirely in favor of a special volumetric entity called a Texel. Each voxel in the Texel contained both the density and the direction(s) of the microgeometry being represented in what they called a 'frame bundle', and could be used, along with a lighting model which was designed for the particular surface type, in a ray-tracer to render realistic microgeometry. The system produced very realistic images of fur, capturing both geometric and lighting properties very well. Although the original method was only demonstrated with the specific microgeometry of fur, it is generalizable to other types of surfaces with appropriate volume and lighting data.

My proposed method depends on being able to precompute lighting of microgeometry off-line and then reconstruct that lighting in a realtime environment. This precomputation will be based upon the Texel rendering system, with a few changes. Because the final lighting reconstruction is going to take place in realtime, the lighting model is not going to be computed at this stage since it might depend on viewing angle or other terms that are not yet known. Instead, only the volumetric data and visibility of each voxel will be computed. These two things will be passed on to the next stage of the method, which is realtime rendering.

2 Shell Rendering

Rendering microgeometry in realtime requires a different technique from the one described above. The Texel method was designed for use with ray-tracing, so for realtime application I adopt a shell-based rendering method [Lengyel et al. 2001] that can be used to produce realistic results of the same type of volumetric microgeometry representation. I propose to compute a small toroidally tiling patch of microgeometry using the Texel method above and then map this patch to a polygonal model by rendering it onto a set of concentric 'shells' that are created around the model. Each shell would render one level of the Texel volume, using alpha blending on graphics hardware to blend the levels together. Although the original method used 'fin' textures on model silhouettes to cover for problems when the shells are parallel, I do not plan to address that issue at present.

3 Realtime Illumination

As noted before, realistic rendering of microgeometry depends heavily on having a realistic illumination of the surface. A large part (but not all, as discussed later) of this illumination consists of self-shadowing within the surface, as noted for the specific case of fur by Lokovic and Veach [Lokovic and Veach 2000]. This self-shadowing can be viewed as a type of global illumination that is local to a specific

surface. Local-global illumination, as I will refer to it, is not a contradiction of terms, but rather a way of describing global illumination effects (specifically shadowing) limited to a local region. In this case, the local region is the tile of microgeometry that is mapped to a model.

Since true realtime global illumination of complex scenes is still a long way off, methods to approximate global illumination are the next best alternative. One approximation method which seems especially promising is Precomputed Radiance Transfer. This technique uses offline global illumination algorithms to precompute the lighting of an object under a basis set of lighting environments. This results in a set of 'light transfer functions' that model how light from a particular basis function actually lights the object when all local-global illumination effects have been accounted for. When an arbitrary target lighting environment is projected into that particular basis set as a set of coordinates, the precomputed light transfer on the object for those coordinates can be reconstructed. Although this technique can give good results, it does present serious limitations when compared to full global illumination. First, the object must be static in order for the precomputed lighting to be correct during reconstruction, since any precomputed lighting would change if the object deformed. Second, finding a set of basis functions that will allow the target lighting environment to be projected into the basis set with minimal error is difficult. Some popular choices include Spherical Harmonics(SH), Zonal Harmonics(ZH), and Harr Wavelets.

I propose to use the ZH basis functions in my paper as described in "Local, Deformable Precomputed Radiance Transfer" [Sloan et al. 2005] because they have the advantage of being easily rotatable at runtime. Since the mapping from the precomputed tile onto the model is not linear (it will require 'bending'), the basis representation will have to be rotated at each vertex. The ZH basis set allows this rotation to be performed at lower cost than it would with SH functions.

4 Lighting Models

Another important part of rendering realistic microspheres is the local lighting model used. This can be seen by noting the success of microfacet lighting models such as Cook-Torrance [Cook and Torrance 1981] for general lighting and Fakefur [Goldman 1997] specifically for microgeometric fur. However, these microfacet lighting models are not appropriate in this application because there is no polygonal surface representation to use such a lighting model on. The basic problem is that these models were meant to be used with 2D objects (surfaces) in 3D space, which we do not have. However, the 'frame bundle' concept which we borrow from the Texel rendering method does give microsphere orientation at each point in the volume, which means that we do have a 1D object at these points. The Banks lighting model [Banks 1994] discusses exactly this situation,

and I propose to use a version of it to light my microgeometry volume.

5 Conclusion

I propose to render microgeometry in realtime with local-global illumination and a realistic local lighting model. Realtime rendering is achieved by using a concentric shell method where the microgeometry is sampled onto the shells. The local-global illumination effects are precomputed according to a portion of the Texel rendering method, and reprojected in realtime with Precomputed Radiance Transfer using Zonal Harmonics basis functions. Finally, I propose to add a unique lighting model that works with my microgeometry method to allow for view-dependent lighting.

References

- [Banks 1994] BANKS, D. C. 1994. Illumination in diverse codimensions. *Computer Graphics 28*, Annual Conference Series, 327–334.
- [Cook and Torrance 1981] COOK, R. L., AND TORRANCE, K. E. 1981. A reflectance model for computer graphics. In *SIGGRAPH 1981 Proceedings*, ACM.
- [Goldman 1997] GOLDMAN, D. B. 1997. Fake fur rendering. In *Proceedings of SIGGRAPH 1997*, ACM Press / ACM SIGGRAPH, ACM.
- [Kajiya and Kay 1989] KAJIYA, J. T., AND KAY, T. L. 1989. Rendering fur with three dimensional textures. In *ACM SIGGRAPH Computer Graphics Vol. 23, Issue 3*, ACM SIGGRAPH.
- [Lengyel et al. 2001] LENGYEL, J., PRAUN, E., FINKELSTEIN, A., , AND HOPPE, H. 2001. Real-time fur over arbitrary surfaces. In *Proceedings of the 2001 symposium on Interactive 3D graphics*, ACM Press / ACM SIGGRAPH, ACM.
- [Lokovic and Veach 2000] LOKOVIC, T., AND VEACH, E. 2000. Deep shadow maps. In *Proceedings of SIGGRAPH 2000*, ACM Press / ACM SIGGRAPH, ACM.
- [Sloan et al. 2002] SLOAN, P.-P., KAUTZ, J., AND SNYDER, J. 2002. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. In *Proceedings of SIGGRAPH 2002*, ACM Press / ACM SIGGRAPH, ACM.

- [Sloan et al. 2005] SLOAN, P.-P., LUNA, B., AND SNYDER, J. 2005. Local, deformable precomputed radiance transfer. In *Proceedings of SIGGRAPH 2005*, ACM Press / ACM SIGGRAPH, ACM.