Real-Time Relief Mapping on Arbitrary Polygonal Surfaces

Fábio Policarpo*
Paralelo Computação

Manuel M. Oliveira†
Instituto de Informática
UFRGS

João L. D. Comba‡
Instituto de Informática
UFRGS

Abstract

We present a technique for mapping relief textures onto arbitrary polygonal models in real time, producing correct self-occlusions, interpenetrations, shadows and per-pixel lighting. The technique uses a pixel-driven formulation based on an efficient ray-height-field intersection implemented on the GPU. It has very low memory requirements, supports extreme close-up views of the surfaces and can be applicable to surfaces undergoing deformation.

1 Introduction

Proper simulation of the appearance of surface details is crucial for producing realistic images. As a result, several techniques have been proposed along the past three decades for rendering surface details. Such techniques, however, usually present some drawbacks such as requiring large numbers of polygons or considerable amount of memory; others are unable to produce important shading features such as self-occlusion and self-shadowing. We show how to use a normal and a depth map, both packed into a single 32-bit texture (a relief texture) to map and render surface details onto arbitrary polygonal models. This approach presents several desirable features: it is fast and compact, is easy to implement, supports close-up views without introducing noticeable texture distortions, and supports mipmapping and anisotropic filtering.

2 Relief Mapping on Polygonal Surfaces

Assigning a relief texture to a polygonal model is similar to assigning a regular texture, since it only requires the specification of a pair of texture coordinates per vertex. The rendering is based on an efficient ray-height-field intersection performed on a per-fragment basis, taking place completely inside the GPU using only 2D texture operations. The process can be summarized as follows: for each fragment, the viewing ray is transformed to the fragment's tangent space, where it is intersected with a height-field surface resting under the polygonal surface. The height-field surface is defined by the depth map, whose values are normalized to the [0,1] range. The apparent depth of the surface details can be interactively changed during rendering time by adjusting the direction of the view vector. The intersection is performed using a binary search in texture space, which guarantees the method its speed and accuracy. However, since the binary search may miss some structures in certain situations, a linear search is used to find an approximate intersection, which is then refined by the binary search. The (s,t) texture coordinates of the point where the viewing ray enters the polygonal surface are given by the rasterization process. The (u,v) coordinates of the point where the ray leaves the scaled version of the [0,1] height field are computed using (s,t) and the direction of the viewing ray. Thus, the search is performed against the depth map, starting at (s,t) and moving towards (u,v). In practice, we have noticed that 10 steps of linear search and 5 steps of binary search seem to produce satisfactory results (Figure 1).

Figure 1: The surface details for the character, the teapot and the walls were rendered using relief mapping.

3 Conclusion

Relief mapping renders realistic surface details in real time requiring very low memory footprint. The technique as described does not render surface details at the silhouettes. Except for this, the results are comparable to the rendering of displacement maps and, in some sense, smoother. We are currently investigating ways to properly render silhouettes details.

References

POLICARPO, F., OLIVEIRA, M. M., AND COMBA, J. 2005. Real-time relief mapping on arbitrary polygonal surfaces. In *Proceedings of ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games 2005*, ACM Press, ACM, 155–162.

*e-mail:fabio@paralelo.com.br
†e-mail:oliveira@inf.ufrgs.br
‡e-mail:comba@inf.ufrgs.br