Better Blending

Brian Wyvill *
Dept. of Computer Science
University of Calgary
Alberta, Canada, T2N 1N4

Geoff Wyvill †
Dept. of Computer Science
University of Otago
Dunedin, New Zealand

1 Introduction

Skeletal implicit surfaces [1] are useful for building models, particularly where smooth blends are required. Such models are generated as iso-surfaces in a scalar field, Blending is achieved by placing field generators (skeletal primitives) in close proximity so that their fields overlap. This process is termed, *proximity blending*.

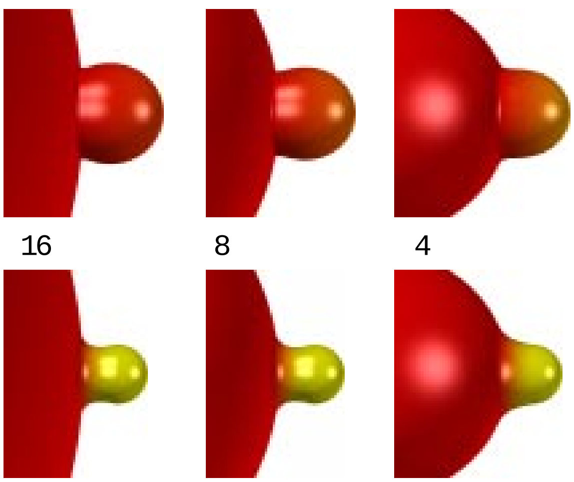

Figure 1. Top row: Regular blend. Bottom row: Restricted blend. Ratio of radii of large to small point primitives are (from left to right) 16, 8, 4.

2 Blending between large and small primitives

There is a problem with proximity blending when there is a large size difference between the primitives involved. The top row of Figure 1 shows a series of close-ups of a point primitive A, with a large radius blended with a point primitive B, with a much smaller radius. The ratios of radii of the large to the small point primitives range from 16 to 4 as indicated in the figure. It can be seen that the shape of the blend region is approximately equivalent to using union between the primitives and does not have the smooth blending shown in the close-up views in the bottom row of Figure 1. The effect is increased as the ratio of radii increases.

An iso-value of 0.5 is commonly used with the field function of [3] (the *Wyvill function*) as it gives a preferred shape in the blend region. To find the value of the field function at some point P due to a primitive S_i it is necessary to know the radius, R_i of S_i at which the field drops to zero. The nearest distance, r_i, to the primitive S_i, is found. The following is used to calculate the field value at P due to the ith primitive where the function $f(r_i)$ is the Wyvill function:

$$if\ (r_i > R_i)\ return\ 0\ else\ return\ f(r_i/R_i).$$

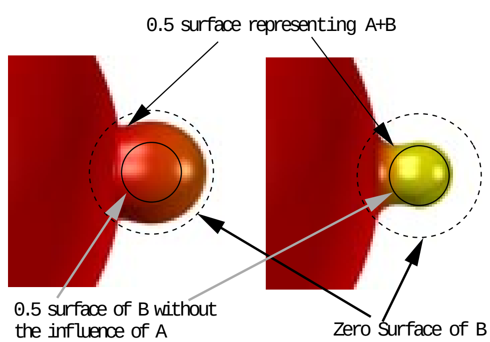

Figure 2. Contours for the large and small spheres showing what happens when the fields are summed. Left: Normal Blend. Right: Restricted Blend

Figure 2 shows the iso-value contours of B, $A + B$ and the zero surface of B. In this case the ratio of the

* blob@cpsc.ucalgary.ca
† geoff@cs.otago.ac.nz

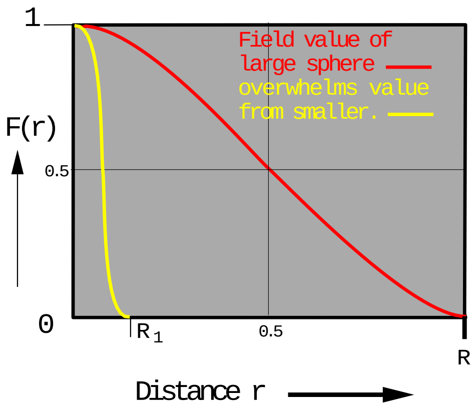

Figure 3. The graph shows the field functions for large and small primitives.

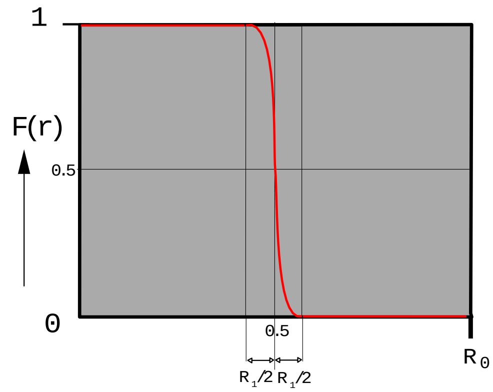

Figure 4. The graph shows the modified field function used for the larger sphere, the *restricted* blend.

radii of the large to the small sphere is 8:1. The left hand of Figure 2 is the regular blending method and it can be seen that the iso-surface of $A + B$ is very close to the zero-surface of B. This is due to the functions used to calculate the field values for A and B, which are shown in Figure 3. It can be seen that the values returned for A overwhelm the contributions from B.

The *blending distance* for the Wyvill function is defined as the distance over which the field function drops from 1 to 0. To improve the shape of the blend, we propose a method we call, *restricted blending.* In the example below, the blending distance for A is R_0 and R_1 for B. Figure 3 represents the field functions for A (shown in red) and B (shown in yellow). Restricted blending involves modifying the blending distance of A to match that of B. The field function of A is scaled, then translated, resulting in the field function shown in Figure 4. The field function is finally clipped so that $f(r) = 1 \; if \; r < (R_0 - R_1)/2$ and $f(r) = 0 \; if \; r > (R_0 + R_1)/2$.

The right hand of Figure 2 shows a dramatic change in the position of the contour representing the sum of the two primitives when using restricted blending. The resulting blends for various ratios of radii are shown in Figure 1. The field function has been restricted to vary within the blending distance of the smaller sphere in each case. Another problem which is solved by the restricted blend is the blending of primitive colours. Each primitive is assigned a colour. The colour at any point on the surface is calculated by weighting the colours of the contributing primitives by the field value contributed by that primitive. It can be seen in Figure 1 that without restricted blending the red colour of the larger primitives completely overwhelms the smaller yellow primitives. By using the restricted blend the smaller primitives appear yellow after blending, as intended.

With primitives other than the sphere the same approach applies. We are currently exploring various functional methods of calculating the restricted blend when blending with arbitrary *BlobTrees*.

3 Conclusion and Future work

Restricted blending works well for two primitives by choosing the appropriate blending distance. Unfortunately this approach does not necessarily work when it is required to blend more than two primitives. We are currently experimenting with continuous functions which provide a blending distance depending on the distance to the primitives involved in the blend. Such a function will return an appropriate blending distance at any chosen point. Our preliminary experiments show that such an approach may be an alternative way of eliminating bulging as described in [2].

4 Acknowledgements

The authors would like to thank the many students, past and present who have contributed towards the implicit modeling research project at the University of Calgary, in particular Callum Galbraith. We would also like to thank the department of computer science and the MACI project for the use of a large cluster of workstations for high performance computing. This work is partially sponsored by the Natural Sciences and Engineering Research Council.

References

[1] J. Bloomenthal. *Introduction to Implicit Surfaces.* Morgan Kaufmann, ISBN 1-55860-233-X, 1997. Edited by Jules Bloomenthal With Chandrajit Bajaj, Jim Blinn, Marie-Paule Cani-Gascuel, Alyn Rockwood, Brian Wyvill, and Geoff Wyvill.

[2] J. Bloomenthal and K. Shoemake. Convolution Surfaces. In *Computer Graphics (Proc. SIGGRAPH 91)*, volume 25, pages 251–256, August 1991.

[3] G. Wyvill, C. McPheeters, and B. Wyvill. Data Structure for Soft Objects. *The Visual Computer*, 2(4):227–234, February 1986.