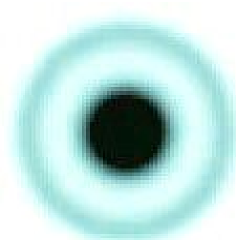
A Simple Mathematical Approximation For Efficient and Faster Rendering of Blobs

M.N. Karthik
mnk@acm.org
National Institute of Technology & Science, University of Madras.

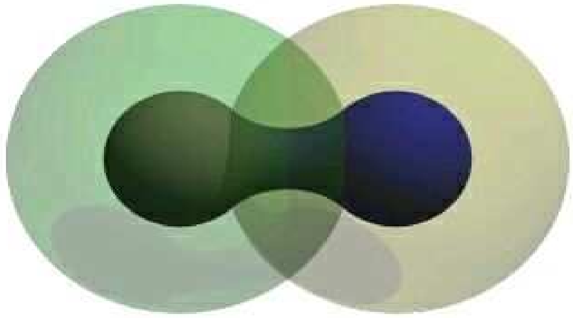
Abstract: Blobs are isosurfaces, which require extensive mathematical processing for efficient rendering. This paper looks at one of the methods of optimizing a certain aspect of the render process by substituting a complex exponential calculation by a simpler approximation.

Introduction to Blobs

A Blob can be defined as an isosurface in 3d space, which is continuous in all of the regions in which it is defined. A Blob is drawn by taking a function in 3d space, taking its co-ordinates as input. A threshold value is then calculated for this function output, and points falling below the threshold are given a value greater than zero (say one) and those above the limit are assigned the value zero. So essentially, the 3D space is divided into regions having a defined surface, and empty region. The intersection of such functions overlaps in a continuous region, which is called an isosurface. A necessary condition for an isosurface to exist is that both the functions must be defined and be continuous in the region for which they are defined. For example, the function **[$F(x,y,z) = x^2 + y^2 + y^2 + C$]** will yield a perfect sphere for an isosurface. Blobs have yet another source, from Physics. The equation **[$F(x,y,z) = x^2 + y^2 + y^2 + C$]** also corresponds to the Electric Field Strength due to a point charge at the origin. The strength is a function of the vector distance of the point considered from origin, or $G(x,y,z) = G(1/r^2)$ where $r = (x^2 + y^2 + y^2)^{1/2}$ This is how a the electric field due to a single charge look like –



This means that the electric field is infinity at the point where the charge lies, is one at unit vector distance from the origin and is zero at infinity. But at any point that we may consider, each point charge will contribute some non-zero amount of electric field. But for the convenience, infinity can be approximated to a relatively bigger distance. Now, if two such charges are near each other, near enough to have some effect on one another, there will a region of overlap of the electric field due to each component. This overlap is a function of both the charges, and their co-ordinates, and gives rise to a blob.

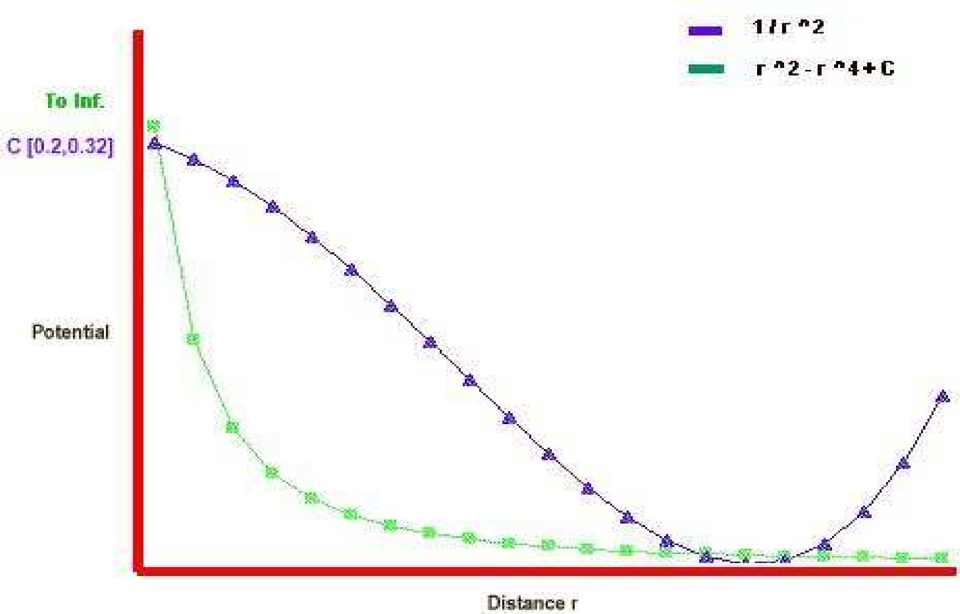


Here, both the electric fields for each of the blobs have been given different colors so that the overlap can be seen. Both the spheres are of equal radii since there are equal electric charges at both their centers. This is an example of a simple blob, and rendering this is very easy since the polynomials involved end with quadratics. The overlapping strength boundaries can also be seen.

These can also be represented by another means - by mapping the strength of the electric field using discrete color bands, by sub-pixel plotting as shown is the first picture of a single charge. The contours that can be seen above are in fact **equi-potential lines** along which the electric field strengths are equal at all points within that line integral, or in case of 3D space at all points within that surface integral.

An Approximation & Optimization

The equation **$G(x,y,z) = G(1/r^2)$** involves a division, which slows down the rendering process. An efficient optimization would be to find an equation that corresponds to the B-Spline curve taken for that **$G(1/r^2)$**, and has a finite value for **$G(1/r^2)$** at **$G(1/0)$**. Such an optimization is possible, when you consider **$G = [r^2 - r^4 + C]$**, where **C** corresponds to value at **0**. Ideally, **C** should be in the range **[0.20-0.32]** for the blob to maintain a large percent of the characteristics of the B-Spline obtained for **$G(x,y,z) = G(1/r^2)$**. The image below shows the both the curves for Distance r Vs. Potential:



As we can see, the functions taken as purely increasing for both the curves represent the same generic pattern. This has another advantage, for **r=0**, the curve does not tend to infinity, the value taken is **C**. Also, for $\mathbf{r=2^{1/2}/2}$, the curve does not take any value, i.e. becomes **G=0**. Hence, we need not plot the curve for points beyond that.

Conclusion

We can safely assume that the value for **r>0.70710678**, there is no contribution to the electric field in the region of space considered. This does not affect the basic characteristics of blobs, in fact it makes the process a lot faster. The factors that provide the blob like nature and appearance are not affected, the only difference is that this process makes the blobs appear as though pressure has been applied to them, from inside/outside, depending on their position, orientation, strength, etc. The advantages of this method are -

- Avoids division, hence making the rendering a lot faster.
- Gives a definite value for G at r=0
- Sharp boundary value for field strength (usually .707)
- Does not modify the basic properties & general look of the blobs

and the disadvantages are -

- Not suitable for rendering blobs where r tends to zero
- Although there is no basic structural change, there is a slight difference in appearance. The blobs look as though they have been pressurized, either from interior or exterior.
- When rendering a very large number (>30000) of small blobs, there is no marked behavior of blob groups as observed when rendering blobs through the usual method.

Together with other methods of optimizations in graphics programming, this method has been proven to render faster and efficient blobs. However, this method would not be suitable for very precise renditions.

References

- Metaballs & Blobs, Institut for Computer Graphik, TU Braunschweig.
- Blinn, J. F., "A Generalisation of Algebraic Surface Drawing", ACM Trans. Graphics, Vol. 1, No 3, July 1982, pp 135-256
- Graves, G., "The Magic of metaballs", Computer Graphics World, May 1993, pp 27-32
- Blinn, J. F., "A Generalisation of Algebraic Surface Drawing", ACM Trans. Graphics, Vol. 1, No 3, July 1982, pp 135-256