Generalized Distance Metrics in Implicit Surface Modelling

Mark Tigges[1] M. S. T. Carpendale[2] Brian Wyvill[1]

[1]Dept. of Computing Science
University of Calgary
mtigges@cpsc.ucalgary.ca

[2]School of Computing Science
Simon Fraser University
carpenda@cs.sfu.ca

Abstract

Implicit surfaces are often designed through the definition and addition of offset surfaces. These are typically defined using a skeleton of primitives such as points and lines. A predefined offset from the skeleton determines the surface. Implied in this commonly used technique is the computation of the distance from the skeleton. It is natural to use the Euclidean definition of distance. In this paper we discuss previous and recent work that takes advantage of alternate definitions of distance to achieve shapes that have previously been difficult to attain.

1 Introduction

The shapes easily modeled with commonly used implicit surfaces have tended to be curved (giving rise to their nick name blobs). The shapes most easily modeled with CSG tend to be angular. Complex models that required combinations of these two types of shapes can be achieved by combining the two approaches to modeling through a process such as the *BlobTree* [10]. However, blending between the different types of shapes to achieve degrees of *roundedness* remains a problem, making some shapes difficult to design. For instance, basic geometric shapes with rounded corners such as commonly found in furniture and appliances. Through the use of general distance measures we show how to acheive shapes that are not just blobby but exhibit some degree of blobbiness while retaining most angular forms. We show also how these shapes can be achieved using an extension of super-quadrics [3] and how we can easily blend between them. Blending with the use of super-quadric distance metrics has been previously been avoided due to scaling and polygonization difficulties [12].

An implicit surface is a point set which satisfies some implicit function, ie. $F(x, y, z) = 0$. The functions used generate a scalar field. The surface is then a level set of the scalar field. An advantage of these surfaces is the ease with which complex models can be constructed through blending of scalar fields of different implicit functions.

While any implicit function can be used, for example that of a sphere, recent use of this class of surfaces computes a scalar field based on a function of distance from a skeleton of primitives. The primitives are generally lines, points, circles etc. These techniques were introduced by Blinn [5], and refined by Wyvill et al. [11] and Nishimura et al. [8]. A common formulation is shown in function 2.

$$f(r) = \begin{cases} 1 & r < 0 \\ \left(1 - \frac{4}{9}r^2\right)\left(1 - r^2\right)^2 & 0 \leq r \leq 1 \\ 0 & r > 1 \end{cases} \quad (1)$$

$$F(\mathbf{p}) = \sum_{i=1}^{i<=n} f_i(|\mathbf{p}|) \quad (2)$$

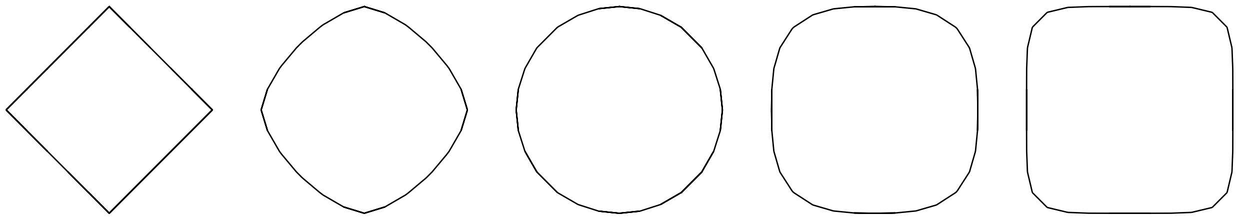
Where $\mathbf{p}$ is the point (x, y, z), and f_i is the field function for the i^{th} skeleton in the composite implicit surface. The function used for the field function f_i can be as shown in function 1 however within some constraints [4, 12] any function can be used.

In this paper we describe how to integrate general distance metrics into a general implicit surface modeling system. The use of general distance metrics for implicit surfaces has been described before, most notably by [4], however assumptions and simplifications were made in that paper. We describe a solution for how to use different metrics on non-point skeletons, and how to use metrics that have heretofore been avoided. The end result of this work is to replace $|\mathbf{p}|$ (a shorthand for Euclidean distance) in function 2 with some other function.

This paper is organized in the following sections. In section 2 we discuss the previous work both on implicit surfaces and the use of general distance metrics for point skeleton implicit surfaces. In section 3 we discuss use of the super-quadric distance metric. In section 4 we discuss how to use these general distance metrics in non-point skeleton implicit surfaces. Some results are shown in section 5 and conclusions and future work are outlined in 6.

2 Previous Work

The framework described in the original work on implicit surfaces [5, 11, 8] defines the surface as a level set of an implicit function $F(x, y, z) = 0$, and yielded shapes that had a spherical offset surface. This is to say that a point skeleton yields a sphere, a line skeleton yields a cylinder with hemispherical caps, a circle yields a torus, etc. This is a result



(a) L_1 (b) $L_{1.5}$ (c) L_2 (d) L_3 (e) L_6

Figure 1: A summary of the affect of variation of p for L_p metrics for two dimensional implicit surfaces.

of using a Euclidean distance metric to determine the distance of a point from the skeleton of the implicit surface.

When introducing implicit surfaces, Blinn [5] used only the Euclidean distance measure. He did mention however that in general any distance measure could be used. It was briefly pointed out in a paper from Blanc and Schlick [4] that using other methods for distance measure can extend the modeling power of the system. Their suggestion was to use the L_p metric, expression 3.

$$L_p = (|x_1 - x_2|^p + |y_1 - y_2|^p + |z_1 - z_2|^p)^{1/p} \tag{3}$$

Expression 3 can be used to replace $|\mathbf{p}|$ in function 2, In this case $\mathbf{p} = < x_1 - x_2, y_1 - y_2, z_1 - z_2 >$, where $< x_1, y_1, z_1 >$ is the closest point on the skeleton to the query point $< x_2, y_2, z_2 >$.

The L_p metric is a generalization of the Euclidean metric. Note that L_2 is exactly the Euclidean definition. L_∞ is known as the Manhattan distance. The interesting use of the L_p metric occurs for $p > 2$. Note that in the limit as p goes to ∞ the result of the metric is simply $\mathtt{max(}|x_1 - x_2|, |y_1 - y_2|, |z_1 - z_2|\mathtt{)}$. A summary of the affect of p to surface shape for two dimensional implicit surfaces is shown in figure 1.

The Blanc and Schlick paper, however did not provide the method for computation of the normal in 3D. (The normal is the gradient of the difference vector from the skeleton, see appendix A). This problem is an issue when the skeleton is not a point, section 4. The main purpose of their work was to survey existing field and distance functions and present new ones. The majority of the work in that paper on different distance metrics was through the use of user defined distance metrics. Their formulation for this is similar in principle to the use of a star-shaped set as input for a Minkowski metric [9].

3 Super-quadric Distance

A super-ellipse is a generalization of a circle. Their introduction was by Faux and Pratt [7]. They allow the specification of all shapes between squares, circles and pinched circles.

The definition of super ellipses was extended to three dimensions by Barr [3] to form super-quadrics. The formulation for the surface of a super-quadric is given in equation 5.

$$d_{sq}(x,y,z) = \left(\left|\frac{x}{a}\right|^{\frac{2}{ew}} + \left|\frac{y}{b}\right|^{\frac{2}{ew}}\right)^{\frac{ew}{ns}} + \left|\frac{z}{c}\right|^{\frac{2}{ns}} \tag{4}$$

$$f(x,y,z) = d_{sq}(x,y,z) - 1 = 0 \tag{5}$$

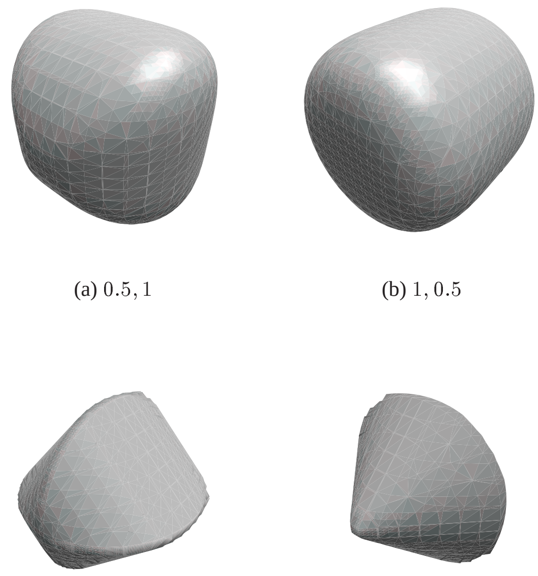
The parameters ew and ns are roundedness parameters in *east/west* and *north/south* directions. The a, b, and c parameters simply imply the scale in the axis aligned directions. The roundedness parameters are the main shape control parameters. The shapes attainable range from a cube to a sphere if ew and ns together change uniformly from 0 to 1. These parameters can be changed independently to attain cylindrical surfaces in either direction. If the ew and ns parameters go above 2 the surface becomes pinched. A full inventory of shapes can be found in [2], figure 2 shows a small sampling of the possible shapes.

The use of a super-quadric formulation for the measure of distance has been avoided due to difficulty with blending [4, 12]. This is because when equation 5 (without the -1 radius term) is used to measure distance in an implicit surface system the scale of the offset surface is not consistent. The $\left|\frac{z}{c}\right|^{\frac{2}{ns}}$ term causes the offset to grow when ns decreases. This needs to be controlled in order to make the function useful as a distance measure.

The first step to getting a useful distance measure from function 5 is to note that when $ew = ns = 1$ the definition is equivelent to the implicit form of a sphere, $f(x,y,z) = x^2 + y^2 + z^2 - 1$. This infers that we need to have an extension to the implicit equation of a super-quadric such that when $ew = ns = 1$ the result is the L_2 distance metric. Only a small amount of thought is required to see that we need to raise the function 4 to the exponent $ns/2$. This is shown in function 6.

$$f(x,y,z) = d_{sq}(x,y,z)^{\frac{ns}{2}} \tag{6}$$

The above function for distance, 6, is a generalization of



(a) $0.5, 1$ (b) $1, 0.5$

(c) $2, 1$ (d) $1, 2$

Figure 2: An inventory of super-quadric shapes showing the values for the ew and ns parameters. The scale parameters are all set to 1.

the L_p metric. This can be shown quite easily, an informal discussion follows. If $ew = ns = 1$ then it is easily seen that we have the L_2 metric, in fact if $ew = ns = n$ we have the following (the scale parameters a, b and c are omitted for brevity):

$$\left(\left(|x|^{\frac{2}{n}} + |y|^{\frac{2}{n}}\right)^1 + |z|^{\frac{2}{n}}\right)^{\frac{n}{2}} = L_{\frac{2}{n}} \tag{7}$$

Interesting cases which should be outlined are when ew and ns simultaneously go to zero and to ∞:

$$\begin{aligned}\lim_{ew,ns\to 0} d_{sq}(x,y,z)^{\frac{ns}{2}} &= \left(\left(|x|^{\infty} + |y|^{\infty}\right)^1 + |z|^{\infty}\right)^0 && (8)\\ &= \max(|x|, |y|, |z|) && (9)\\ &= L_{\infty} && (10)\end{aligned}$$

Finally if ew and ns approach ∞ then we have the following:

$$\begin{aligned}\lim_{ew,ns\to \infty} d_{sq}(x,y,z)^{\frac{ns}{2}} &= \left(\left(|x|^{0} + |y|^{0}\right)^1 + |z|^{0}\right)^{\infty} && (11)\\ &= \left(|x|^0 + |y|^0 + |z|^0\right)^{\infty} && (12)\\ &= L_0 && (13)\end{aligned}$$

Since this metric is nothing more than a special case of the L_p metric its use in implicit surface modelling is no more difficult than the use of the L_p metric. Some examples of the shapes attainable with this super-quadric distance can be found in section 5.

In order to find the normal on the surface of an implicit surface generated with this metric we use the vector of partial derivatives of the difference vector from the skeleton. See appendix A.

4 Non-point Skeletons

All of the above discussion is relevant only for point skeletons. If we have a circle skeleton which lies say in the xy plane then we will not generate a torus as the surface from the L_p metric with $p > 2$. There will be pinches in the neighbourhoods on the surface where $x \approx z$ and the normal to the surface will no longer be the vector of partial derivatives for the difference vector.

What is required is a local basis defined at the closest point on the skeleton for the point in question. When the field function is evaluated for a given point the typical implicit surface system computes the value from the distance of that point to the closest point on the skeleton. This distance must be computed with respect to a local basis defined at that closest point.

It turns out that for a point skeleton (where the point is at the origin) the basis to use is always $[<1,0,0>, <0,1,0>, <0,0,1>]$. However what is it for a line skeleton where the line lies in the y axis? The answer is $[<1,0,0>, <0,1,0>, <0,0,1>]$. This basis slides up and down the line as the closest point on the skeleton changes. The distance function $d(x,y,z)$ is then evaluated with respect to this basis. If $\mathbf{p}$ is a point for which the field value is requested and $\mathbf{s}$ is the closest point on the skeleton to $\mathbf{p}$ then we evaluate d as follows: $d((\mathbf{p} - \mathbf{s}) * \mathcal{B})$. Where $\mathcal{B}$ is the basis on the skeleton at $\mathbf{s}$ and multiplication of a vector $\mathbf{v}$ by a basis $\mathcal{B}$ is meant as the vector of dot products between v and each of the axial vectors of $\mathcal{B}$.

For most skeletons the basis is always the default of $[<1,0,0>, <0,1,0>, <0,0,1>]$. In our implementation this means; points, lines, planes, disks and squares. Their are two skeletons in our implementation which require a special basis, they are; circle and polyline. The basis for a circle which lies in the xz plane and centered at the origin is $[<x_c, \hat{0}, z_c>, <0,1,0>, <-z_c, \hat{0}, -x_c>]$ (where the subscript c is meant as the coordinates of the closest point on the skeleton). It is with respect to this basis that the difference vector $\mathbf{p} - \mathbf{s}$ is computed. Now also the vector of partial derivatives of the distance function used is appropriate for the normal of the surface.

The basis for the polyline is more complicated. A polyline skeleton is a set of n-lines defined by an ordered set of $n+1$ vertices. Each pair of contiguous line segments are connected with a circular arc to provide a piecewise continuous field. The radius of the connecting arcs is a user

Figure 3: Five primitives with distance measured with L_4 blended to form a table.

defined parameter allowing for a primitive with considerable flexibility. A basis is defined by associating a basis with the first point and ensuring a continuous frame based on the relationship between contiguous line segments. For the frames of the line segments we define an arbitrary frame F_0 for the first point p_0. The F_i, $0 < i < n$ frames are computed with respect to the F_{i-1} frames by rotating around the normal vector of the F_{i-1} frame by the angle between the two line segments. It remains to compute a frame for any point along a curve joining two segments of the polyline. This can be done using an interpolation. See [6] for a complete discussion of calculating frames along a curve.

5 Results

In this section we present some visual results of models build with the L_p and the super-quadric distance metrics. Figure 3 shows four line segments blended with a plane segment to form a table. All of the field functions are evaluated using an L_4 metric.

Figure 4 shows the result of blending super-quadric distance measures from point skeletons. As is shown by the figure there is no problem blending between fields generated with equation 6. There are three skeletons in the figure. The left most uses super-quadric distance with $ew = 2$ and $ns = 1$ and the right most uses super-quadric distance with $ew = 1$ and $ns = 0.2$. The middle skeleton uses Euclidean distance.

The third example, figure 5, shows an odd barbell/pipe type of object. The point of the object is to show the use of super-quadric and L_p-distances blended inside a complex hierarchical soft object using non-point skeletons. The object is built by blending two circle primitives using an L_6 metric with the CSG difference of a line primitive from super-quadratic distance line primitive. The difference gives the hole through the middle, the subtracting surface was accomplished with a standard Euclidean distance metric surface. The super-quadratic distance measure was used for the outside line primitive so that we could easily avoid the hemispherical caps.

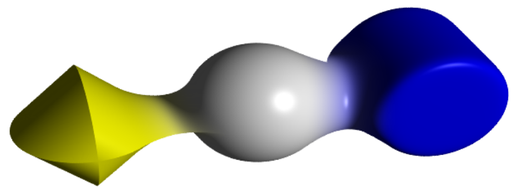


Figure 4: Blending of three primitives using the super-quadric distance metric.

Figure 5: Various distance measures in a more complex hierarchy.

6 Conclusions

In this paper we have discussed the use of generalized distance metrics for implicit surface modeling. We have shown that using both the L_p metric and the super-quadric metric presents no problems in integration with an implicit modeling system. In order to use the super-quadric formula as a distance metric we extended the formula to be a special case of the L_p metric.

There are currently two extensions that we are considering for future work. The first is to implement parameteric distance metrics. This would be to make the L_p metrics p parameter, or super-quadric ns and ew parameters dependent on a set of direction vectors. This would allow truly odd shapes, with protruding rounded *corners*. The other extension is to implement ray quadrics [1] for input to the Minkowski metric [9].

References

[1] Ergun Akleman. Construction of convex and star shapes with yontsal functions. In *International Symposium on Computer and Information Sciences*, November 1993.

[2] David Kirk ed. Alan Barr. *Graphics Gems III*, chapter Rigid Physically Based Super Quadrics, pages 137–159. Academic Press, 1995.

[3] Alan Barr. Superquadrics and angle-preserving transformations. *IEEE Computer Graphics and Applications*, 1(1), 1981.

[4] Carole Blanc and Christophe Schlick. Extended field functions for soft objects. In *Implicit Surfaces'95*, pages 21–32, apr 1995.

[5] James F. Blinn. A generalization of algebraic surface drawing. *ACM Transactions on Graphics*, 1(3):235–256, July 1982.

[6] Jules Bloomenthal. *Graphics Gems*, chapter Calculation of Reference Frames along a Space Curve. Academic Press, 1990.

[7] I. D. Faux and M. D. Pratt. *Computational Geometry for Manufacture and Design*. Halstead Press, 1979.

[8] H. Nishimura, M. Hirai, T. Kawai, T. Kawata, I. Shirakawa, and K. Omura. Object modeling by distribution function and a method of image generation. *Trans. IECE Japan, Part D*, J68-D(4):718–725, 1985.

[9] Valentin. *Convex Sets*. ACM Press, 1980.

[10] Brian Wyvill, Eric Galin, and Andrew Guy. Extending the csg tree - warping, blending and boolean operations in an implicit surface modeling system. In *Proceedings of the Third Eurographics Workshop on Implicit Surfaces*, June 1998.

[11] Brian Wyvill, Craig McPheeters, and Geoff Wyvill. Data structure for soft objects. *The Visual Computer*, 2(4):227–234, 1986.

[12] Brian Wyvill and Geoff Wyvill. Field functions for iso-surfaces. *The Visual Computer*, 5(1/2):75–82, March 1989.

A Normals

In this appendix we present the computation of the normal vector to the offset surface generated for the non-Euclidean distance metrics discussed in this paper. For both cases this is simply the the vector of partial derivatives of the offset vector in the local basis on the skeleton (see section 4). The distance function is given as f complete with the parameters for the function, and its partial derivates are shown below.

L_p Metric

$$f(x,y,z,n) = (|x|^n + |y|^n + |z|^n)^{\frac{1}{n}} \tag{14}$$
$$\tag{15}$$

$$\frac{\partial f}{\partial x} = \frac{f(x,y,z)|x|^n}{x\,(|x|^n + |y|^n + |z|^n)} \tag{16}$$
$$\frac{\partial f}{\partial y} = \frac{f(x,y,z)|y|^n}{y\,(|x|^n + |y|^n + |z|^n)} \tag{17}$$
$$\frac{\partial f}{\partial z} = \frac{f(x,y,z)|z|^n}{z\,(|x|^n + |y|^n + |z|^n)} \tag{18}$$

Super-quadric Metric

$$f(x,y,z) = \left(\left(|x|^{\frac{2}{ew}} + |y|^{\frac{2}{ew}}\right)^{\frac{ew}{ns}} + |z|^{\frac{2}{ns}}\right)^{\frac{ns}{2}} \tag{19}$$
$$\tag{20}$$

$$\frac{\partial f}{\partial x} = \frac{f(x,y,z)\left(|x|^{\frac{2}{ew}} + |y|^{\frac{2}{ew}}\right)^{\frac{ew}{ns}} |x|^{\frac{2}{ew}}}{x\left(|x|^{\frac{2}{ew}} + |y|^{\frac{2}{ew}}\right) d_{sq}(x,y,z)} \tag{21}$$
$$\frac{\partial f}{\partial x} = \frac{f(x,y,z)\left(|x|^{\frac{2}{ew}} + |y|^{\frac{2}{ew}}\right)^{\frac{ew}{ns}} |y|^{\frac{2}{ew}}}{y\left(|x|^{\frac{2}{ew}} + |y|^{\frac{2}{ew}}\right) d_{sq}(x,y,z)} \tag{22}$$
$$\frac{\partial f}{\partial z} = \frac{f(x,y,z)|z|^{\frac{2}{ns}}}{z\; d_{sq}(x,y,z)} \tag{23}$$