CUCS 291-87

Recovery of superquadrics from depth information

Terrance E. Boult and Ari D. Gross
Columbia University Department of Computer Science
New York City, New York, 10027.
tboult@cs.columbia.edu, ari@sylvester.columbia.edu

Abstract

Superquadrics are a a class volumetric primitive which can model objects including rectangular solids with rounded corners, ellipsoids, octaheadrons, 8-pointed stars, hyperbolic sheets, and toroids with cross sections ranging from rectangles with rounded corners to elliptical regions. They can be stretched, bent, tapered and combined with boolean operations to model a wide range of objects. This paper discusses our progress at attempting to recover a subclass of superquadrics from 3D depth data.

The first section of this paper presents a mathematical definition of superquadrics. Some of the rationale for using superquadrics for object recognition is then discussed. Briefly, superquadrics are flexible enough to represent a wide class of objects, but are simple enough to be recovered from 3d data. Additionally, the surface and its normal surface both have well defined inside-out functions which provide a useful tool for their recovery.

The third section examines some of the difficulties to be encountered when modeling objects with superquadrics, or attempting to recover superquadrics from 3D data. These difficulties include the general problems of a non-orthogonal representation, difficulties of dealing with objects which are not exactly representable with CSG operations on the primitives, the need to recognize negative objects, certain numerical instabilities and some problems caused by using the inside-out function as an approximation of the distance of a point from the superquadric.

Our current system employs a nonlinear least square minimization technique on the inside-out function to recover the parameters. After discussing the details of the current system, the paper presents examples, using noisy synthetic data, where the system successfully uses multiple views to recover underlying superquadrics. Also presented are examples using range data, including the recovery of a negative superellipsiod.

Some pros and cons of our approach as well as few conclusions, and a discussion of our planned future work appear in the final section. The main result is that least square minimization using the inside-out function allows both positive and negative instances of superellipsoids to be recovered from depth data. A second preliminary result is that a single view of a superquadric may not be sufficient for reconstruction without additional assumptions.

Mathematical Definition of Superquadrics

Mathematically, superquadric solids are a spherical product of two superquadric curves. Superquadric curves are similar to traditional quadric curves except the terms in the definition are raised to parameterized exponents (not necessarily integers). For example, a superellipse (see [Gardiner-65]), is defined such that: $\left(\frac{x}{a}\right)^{\frac{2}{\epsilon}} + \left(\frac{y}{b}\right)^{\frac{2}{\epsilon}} = 1$. When ϵ, the *relative shape parameter* is 1, the curve describes an ellipse. As the relative shape parameter varies from 1 down to 0, the shape becomes progressively squarish; as it varies from 1 toward 2, the shape transforms from a ellipse to a diamond shaped bevel. When the parameter is greater than 2, the shape becomes pinched and as the parameter approaches infinity, the shape approaches a cross.

The result of the spherical product of two such curves is conveniently represented in a parametric form, e.g., a superquadric ellipsoid can be represented as (see figure 1):

$$\vec{s}(\eta,\omega) = \begin{bmatrix} a_1 \cdot \cos^{\epsilon_1}\eta \cdot \cos^{\epsilon_2}\omega \\ a_2 \cdot \cos^{\epsilon_1}\eta \cdot \sin^{\epsilon_2}\omega \\ a_3 \cdot \sin^{\epsilon_1}\eta \end{bmatrix}, \quad \begin{matrix} -\frac{\pi}{2} \leq \eta \leq \frac{\pi}{2} \\ -\pi \leq \omega \leq \pi, \end{matrix} \quad \begin{matrix} \text{for any fixed positive} a_1, \\ a_2, a_3, \epsilon_1, \text{ and } \epsilon_2. \end{matrix}$$

The parameters a_1, a_2, a_3 effect the size of the superellipsoid along the x, y and z respectively. The parameters ϵ_1 and ϵ_2 effect the relative shape of the superellipsoid in the latitudinal (xz) and longitudinal (xy) directions. When the 5 parameters are all unity, the superellipsoids define the unit sphere.

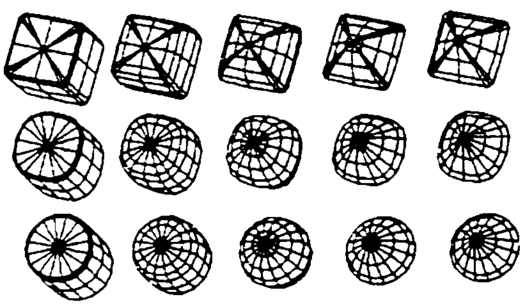

Figure 1: Superellipsoids with relative share parameter ϵ_1 having values .1, .5, 1, 1.5 and 2 (left to right), and ϵ_2 having values .1, .6 and 1 (top to bottom).

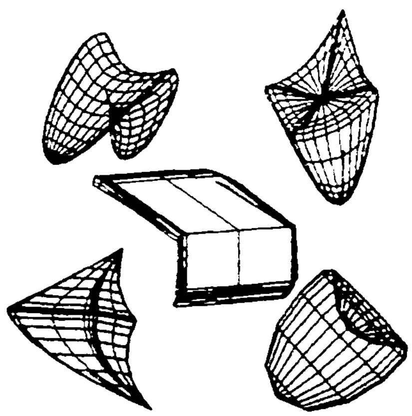

Figure 2: Examples of superquadrics deformed by bending and tapering

Superellipsoids have a well defined inside-out function:

$$f(x,y,z) = \left(\left(\frac{|x|}{a_1}\right)^{\frac{2}{\epsilon_2}} + \left(\frac{|y|}{a_2}\right)^{\frac{2}{\epsilon_2}} \right)^{\frac{\epsilon_2}{\epsilon_1}} + \left(\frac{|z|}{a_3}\right)^{\frac{2}{\epsilon_1}} \tag{1}$$

$$\text{where if} \begin{cases} f(x_0,y_0,z_0) = 1, & \text{then } x_0, y_0, z_0 \text{ is on the surface boundary;} \\ f(x_0,y_0,z_0) < 1, & \text{then } x_0, y_0, z_0 \text{ lies inside the surface boundary,} \\ f(x_0,y_0,z_0) > 1, & \text{then } x_0, y_0, z_0 \text{ lies outside the surface boundary.} \end{cases}$$

The absolute values are introduced to extend the inside-out function beyond the first octant.*

Superellipsoids (hereafter referred to as SEs) define the simplest of the superquadrics, and the only subclass which can model convex solids. One can also define superhyperboloids of one or two sheets and

*Alternatively, one could insure that fractional exponentiation was broken into two stages, squaring and then raising to the remaining fractional power.

supertoroids, see [Barr-81]. In addition to the variety of shapes defined by the basic superquadrics. Barr also discusses the application of angle-preserving transforms which allow translation, rotation, bending and twisting. With the addition of tapering and traditional boolean combination operations, superquadrics become a powerful modeling tool, see Figure 2.

Why use superquadric solids?

This section discusses some of the rational for using superquadrics for object recognition. In short, they are flexible enough to represent a wide class of objects, but are simple enough to be recovered from 3d data. Their inside-out function provides a useful tool for their recovery. In addition, they have a useful duality principle with their normal surface, which also has an inside-out function.

Traditional constructive solid modeling systems use boolean operations to combine primitives, such as, spheres, cylinders, and rectangular solids. By extending the primitive shapes to SEs, we allow the user to easily produce a continuum of forms, from spheres, to cuboids with rounded corners, to cubes, to diamond shapes. Such objects are difficult to model with traditional constructive solid geometry (hereafter CSG) systems. Making them primitives simplifies the job of the designer for any problem that contains objects with these properties.[†]

However, adding flexibility above that of a traditional CSG system is not the only reason to choose superquadrics. For added flexibility, one could make the primitives of the system Generalized Cylinders or Generalized Cones, as introduced in [Brooks-Binford-80]. The problem with GC's is that recovering them is a difficult process, partially because each GC may require hundreds of parameters to describe it.[‡] As in [Brooks-85] or [Rao-Nevatia-86], one generally has to greatly restrict the class of GCs allowed before one can reliably recover them. Superquadrics provide such a restriction, and in addition, provide mathematical properties (the inside-out function and normal surface duality principle) that will make the recovery of superquadrics simpler and more robust than the recovery of unrestricted GCs.

The inside-out function provides a useful tool for recovery, because it provides a simple way to determine which of the data points are inside the surface and the value of the function grows as points are moved farther from the surface. Thus it can be used in conjunction with a minimization technique to recover the surface.

Canonical superquadrics (except the supertoroids) have a desirable duality property. The superquadric normal vectors lie on a dual superquadric form; that is, if the normal vectors were translated to the origin, they would generate another superquadric of the same class such that the normal of the new surface (if translated to the origin) would produce a copy of the original superquadric (except for a translation). One can easily derive the form of the normal surface as a spherical product as well as its own inside-out function, see [Barr-81].

Note that the inside-out function for the surface normals may be used to find superquadrics fitting surface normal information as would be available from shape-from-X methods.

In summary, the authors are interested in recovering superquadrics because they are a flexible primitive for a CSG type system, but are simple enough to be recovered from 3d data. In addition, the surface and surface normals have easily computed inside-out functions.

Some Difficulties with Superquadrics

The difficulties of modeling with and recovering superquadrics can be divided into two classes: those difficulties common to any CSG based system and those difficulties particular to superquadrics. The former class includes general problems of non-orthogonality of the representation, difficulties of dealing with objects which are not exactly representable with CSG operations on the primitives, and the need

[†] A simple CAD system has been developed by A. Pentland, see [Pentland-86b] that combines SEs (constrained to only those that are convex solids) with CSG type operations plus bending and tapering. The easy with which people became accustomed to modeling with this system may be tied to their own internal representation of the objects. [Pentland-86a] presents arguments to assert motivate the use of superquadrics as modeling primitives by showing correspondences to human vocalization of object descriptions.

[‡] The number of parameters necessary to define a GC depends on the complexity of the cross-section function, the spine, and sweeping function. It is interesting to note that superellipsoids and supertoroids can be defined as a subclass of GC; e.g., SEs are those straight spined GC's with their cross-section and sweeping functions defined by superellipses. If the SEs are bent or tapered, these deformations are applied to the spine and sweeping function respectively.

to recognize negative objects. The latter class of difficulties includes certain numerical instabilities and particular types of parameter ambiguity. This section briefly discusses these difficulties.

One of the most difficult problems of any CSG system is the non-orthogonality of the representation. One can generate the same volume by a number of operations on primitive objects. This makes the matching of recovered objects with a database more difficult. Since the underlying objects are symmetric with respect to certain sequences of rotation, these rotations form a equivalence class which can be anticipated by the matching.

When the primitive objects are SEs some of the symmetries are nonintuitive. For example a cube with just slightly rounded corners can be represented as a SE with $\epsilon_1 = .01, \epsilon_2 = .01, a_1 = a_2 = a_3 = 1$ and also (after a rotation of $\frac{\pi}{4}$ around the z-axis) by one with $\epsilon_1 = .01, \epsilon_2 = 1.99, a_1 = a_2\sqrt{2}$, and $a_3 = 1$.** Other nonintuitive symmetries may also exist.

The two basic CSG operations are addition (set union) and subtraction (set difference). Thus if one is interested in recovering models created with boolean operations on superquadrics, one must be able to recognize them from partial boundaries, and also be able to recognize negative superquadrics as the result of a difference operation rather than a partial boundary of some other positive superquadric. Here again, a nonintuitive symmetry of SEs becomes apparent. Consider a small patch of strictly concave data. It can be modeled as a patch of a negative convex object or part of a positive concave (pinched) SE.

As with any CSG system, there are the difficulties of approximately representing objects. Two of the most difficult problems to solve are what level of detail needs to be preserved, and given that level, how does one allocate deviation from the actual object to different pieces of the construction. The importance of this became quite apparent as we attempted to reconstruct a soda-can from the utah range data, [utah-85]. The actual object (the can upside down) may, at one level of detail be modeled as just a cylinder. However, if a better model is desired, one can use a cylinder minus an ellipsoid. Finally more accurate models exist (because the small concave bevel which helps makes the can stable when stacked must also be modeled), e.g. a cylinder minus the ellipsoid plus a small section of a superhyperboloid of one sheet. Thus the level of accuracy chosen for the modeling can greatly effect the resultant model. As with the other problems mentioned in this section, the major impact of this difficulty is not really in the recovery phase, but becomes quite important when one attempts to match a recovered model (or part thereof) against some internal models.

A second, and more difficult problem in approximate modeling using superquadrics is how to measure "error of fit". The intuitive idea is simply to take $\sum_i [1 - f(x_i, y_i, z_i)]^2$ where f is the inside-out function. Unfortunately, this measure is *not* even proportionally related to the distance of the points from the surface. This problem is reminiscent of the problem of defining goodness of fit for conic filters, see [Turner-74], and in the future, these authors will be attempting to apply results from that work to the recovery of SEs. The problem of measuring error is exasperated as one attempts to apportion the error to individual data points, especially if there is any nonuniformity in data densities.

By examining the inside-out functions which define superquadrics††, one might expect to find problems of numerical instability because of the exponents ϵ_1 and ϵ_2. While these researchers have encountered some problem (most notably effects of roundoff errors and floating point overflows) when $\epsilon_1 \ll 1$ for reasonable values (say $> .1$), the instabilities encountered have not been insurmountable.

A final problem with the use of superquadrics as a modeling primitive is that they are often not well defined by a single view. This is especially a problem for the size parameters. An example of this is flat objects, for which information on a single face does not determine the objects size or rotation in the plane of the face. If the object is large enough one can incorporate multiple views (see the discussion below). However, if the flat object is just a patch of a negative object (e.g. the result of an intersection of a negative cube with another SE), one cannot get another view and recovery is difficult if not impossible without additional constraints. There are numerous ad hoc assumptions one can add to alleviate this problem, e.g., both [Bajcsy-Solina-87b] and [Pentland-86a] consider finding only the superquadric with the smallest volume. While this is rather ad hoc reasoning, the results are surprisingly good, see figure 10. This reformulation can also be rationalized because it makes the inside-out function a better approximation of the true distance of a point to the SE. Consider the square of the distance of a point $(r + k, r + k, r + k)$ from a sphere of radius r. The true distance is $3k^2$. At that point, the inside-out function has the value

**There are at least two ways to represent any SE which has $\epsilon_2 \neq 1$, which follows from the fact that except for rotation and scaling, an SE with $\epsilon_1 = \delta, \epsilon_2 = \gamma$ is equivalent to one with $\epsilon_1 = \delta, \epsilon_2 = 2 - \gamma$.

††See [Barr-81] for those inside-out functions not presented here.

$3\frac{(k)^2}{r^3}$ which can (depending on r) be a gross underestimate of the distance for large objects. If only one view is available, the "error of fit" will decrease as the size of the object increases in the direction opposite the viewing direction. However, the distance of the surface to the data points may actually increase. Introducing the volume term into the inside-out function results in the value of $3r(k)^2$ which a overestimates the actual distance, but is still a better approximation than the standard inside-out function.

Our Current System

This section presents some of the details of our current system for the recovery of superquadrics from 3-D information. The basis of our system is the use of the inside-out function, see equation 1. Similar equations and constraints can be derived using the inside-out function of any of the superquadrics. Note that one of the advantages of minimizing the inside-out function is that it requires little extra effort to incorporate multiple views, assuming one knows the sensor position for each view (to convert points to a common coordinate system).

Given 3-D information about a SE in canonical position, one can use a nonlinear minimization technique to recover the 5 parameters needed to define it. Our system uses a Gauss-Newton iterative nonlinear least square minimization technique.[‡‡] That is, the system minimizes $\sum(1 - f(x, y, z))^2$ where the summation is over all known information points.

If the SE is not in canonical position, the system must also recover estimates of the translation and rotation necessary to put the information on the surface of a canonical SE. There are obviously many approaches to deal with the translation and rotation, the two most obvious are the use of a pair of transforms (one for translation and one for rotation) and the use of a homogeneous transform that combines both translation and rotation. Our system uses the first approach.

For the remainder of this section let $C_\theta = \cos\theta$ and $S_\theta = \sin\theta$. Thus given a canonical SE surface defined as $\vec{s} = \vec{s}(\eta, \omega)$ with an inside-out function $f = f(x, y, z)$, the translated and rotated SE solid $\vec{s}\,'$ is given by

$$\vec{s}\,' = R_\theta R_\eta R_\psi \vec{s} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix},$$

where θ, ψ, ϕ are the Euler angles expressing the rotation about the x, y, z axes respectively, $R_\theta R_\eta R_\psi$ are the rotation matrices about the x, y, z axes respectively and t_x, t_y, t_z are the translation in the $x, y,$ and z directions respectively.

Given the data for a general SE the system minimizes $\sum(1 - f'(x, y, z))^2$ where the summation is over all know information points[***], and where $f'(x, y, z) = f(x', y', z')$ and

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} C_\eta \cdot C_\psi & S_\theta \cdot S_\eta \cdot C_\psi - C_\theta \cdot S_\psi & C_\theta \cdot S_\eta \cdot C_\psi + S_\eta \cdot S_\psi \\ C_\eta \cdot S_\psi & S_\theta \cdot S_\eta \cdot S_\psi + C_\theta \cdot C_\psi & S_\theta \cdot S_\eta \cdot C_\psi - S_\theta \cdot S_\psi \\ -S_\eta & S_\theta \cdot C_\eta & C_\theta \cdot C_\eta \end{bmatrix} \begin{bmatrix} x - t_x \\ y - t_y \\ z - t_z \end{bmatrix}.$$

To employ the Gauss-Newton iteration, the system must compute the Jacobian of the transformation, and thus also needs the partial derivatives of $f'(x,y,z)$ with respect to the 11 parameters, (5 shape, 3 rotation and 3 translation), which were obtained symbolically.

The addition of bending is easily implemented as a simple angle preserving transform on a given axis before rotation and translation. Tapering is simply a linear scaling of the solid along some axis before the application of bending, translation or rotation. Thus in recovery, these transforms are applied first.

The initial implementation of the system recovered only the 5 shape parameters. Using synthetic data with up to 10% uniformly distributed noise, the system could start the minimization procedure from a canonical position (all 5 parameters = 1) and in most cases was still able to recover the underlying surface within the error of the data. Moreover, the convergence was generally quick, requiring < 15 iterations.

‡‡ While it is often argued that the error in depth measurements is gaussian distributed in a direction parallel to the line of measurement, e.g., see [Bolle-Cooper-86], one cannot assume that the direction of measurement is constant for all data points. Thus in our system the use of least squares is an ad hoc assumption rather than an attempt to minimize the error of reconstruction with respect to noise in the data.

*** One example presented introduced an extra factor of $a_1 a_2 a_3$ to make the inside-out function a better approximation to the true distance.

Next the system was extended to deal with translations in addition to the 5 shape parameters. Again the system was generally robust with respect to recovery of the underlying surface. The authors believe that this robustness is partially due to the fact that many of the symmetries which cause the general SEs to be nonorthogonal, require some amount of rotation.

When the system was extended to handle 11 parameters, things became more complicated. For most of the examples presented, even when presented with very poor estimates for starting values, the system was quickly able to find an SE that had low error. Unfortunately, the solutions proposed by the system often seemed to be nonintuitive. However, when examined closely, these solutions proved themselves to be reasonable interpretations of the data. For example, when presented with the range data (from the Utah range database, [utah-85]) for a coke can minus the concave portion of the bottom end, the system initially proposed a solution with error = .0078; the parameters were $\epsilon_1 = 2.0, \epsilon_2 = .88, a_1 = 1, a_2 = 1.3, a_3 = 19.59, t_x = -.19, t_y = -1.69$, and $t_z = -.834$.[*] These parameters describe an object that is beveled along its long axis, and rather round in the other direction. The length of the object is entirely incorrect but then it is being intersected with a negative ellipsoid, and the bottom is cut by a ground plane. When examined closely, the proposed object (when intersected with the other objects known to be in the scene) does seem to be a reasonable fit to the data, but still a cylinder seems intuitively to be a better fit. If forced to look for a cylindrical object, the system finds an SE with error .17.[†]

To help the system avoid local minima, Poisson noise was added to the residual[‡] The system also derived estimated bounds on the possible values for each parameter. These estimates were derived from knowledge of "sensor" and the data values. If during the minimization, any parameter attempts to stray beyond its allowed boundary, the system stochastically pushed it back toward its initial value.

Currently, the system obtaines initial estimates of the translation parameters from the centroids of the original data and derives bounds from sensor information and overall data (maximum variation in any data). However there are many problems with these estimates especially if the number of data points is small or if the system is only given a partial view of an object. These estimates are are obviously much better if multiple views of the object are available. The estimates also assume that the data is segmented, an assumption with which the authors feel particularly uncomfortable.

The system derives estimates of rotation angles and length scales from moments of inertia and bounds on the length parameters from sensor information and overall data. Because the moments of inertia require second order moments the estimates are plagued with more difficulties than the estimates of the translation parameters. The estimates of the both rotation and length scales are very poor if an object (after segmentation) is the result of boolean combinations.

A Few Examples from Our Current System

The first example in figure 3, is a synthetic SE with noisy synthetic data from multiple views. The actual parameters of the superquadric are $\epsilon_1 = 1.59, \epsilon_2 = .39, a_1 = 1, a_2 = 2, a_3 = 3, t_x = 1.5, t_y = 2.5, t_z = 3.5, \theta = .1, \eta = .1, \psi = .1$. The noise in the underlying object was uniformly distributed over the interval [-.15, +.15] and then added to the z (depth) value of a point. The 1000 data points were randomly distributed on the surface before noise was added. The system recovered an SE with parameters $\epsilon_1 = 1.8, \epsilon_2 = .3, a_1 = 1, a_2 = 2, a_3 = 3.58, t_x = 1.49, t_y = 2.5, t_z = 3.47, \theta = .079, \eta = .09, \psi = .101$. The error of the reconstruction was .079. The system required 7 iterations from the initial values to find the solution.

Figure 4, is the recovery of the same synthetic SE as in example 1. However, this time this system was given 1000 data points from one view of the object. Under these condition the recovered parameters were: $\epsilon_1 = 2.09, \epsilon_2 = .67, a_1 = 1, a_2 = 1.94, a_3 = 3.36, t_x = 1.43, t_y = 2.53, t_z = 4.3, \theta = .075, \eta = .07, \psi = .08$. The error of the reconstruction was .169. Obviously the reconstruction from multiple views is superior.

Figure 5 shows the system recovering 5 parameters (i.e. an SE without translation or rotation) using only one view, and very sparse data. In figure 5 one can see the initial data (small dots) and recovered

[*] The error for all examples is defined to be the residual value $(\frac{1}{n}\sum_{i=1}^{i=n}[1 - f(x_i, y_i, z_i)]^2)^{.5}$ where f is the inside-out function of the superquadric. Note that this is not the same as the least square distance to the surface.

[†] This problem can be alleviated by multiplying by $a_1 a_2 a_3$ in the minimization, in which case the system converges to a reasonable cylinder. Oddly enough, the system converged to a can more readily if data from the negative ellipsoid was not removed.

[‡] This was suggested by A. Pentland.

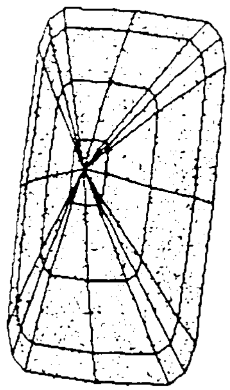

Figure 3: Example recovering 11 parameters using noisy synthetic data from multiple views

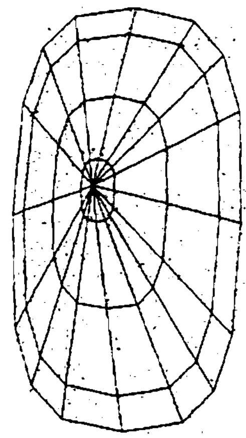

Figure 4: Example recovering 11 parameters using noisy synthetic data from single view

SE. The parameters of the initial data were: $\epsilon_1 = 1.234, \epsilon_2 = .2345, a_1 = 1.2, a_2 = 2.3, a_3 = 3.4$. The system was given only 32 data points which had uniform error in the range [-.17,+.17], and was able to reconstruct a SE with parameters $\epsilon_1 = 1.317, \epsilon_2 = .2745, a_1 = 1.222, a_2 = 2.333, a_3 = 3.53$ and an error of .178.

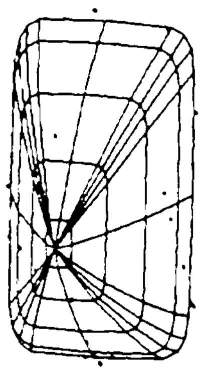

Figure 5: Recovery of 5 parameters from 32 noisy points

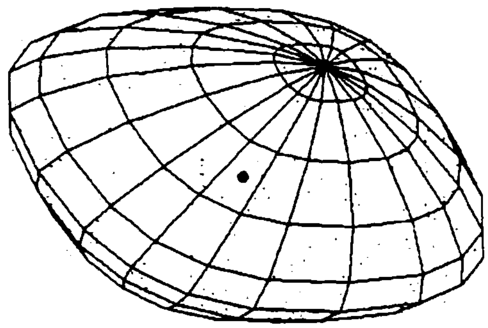

Figure 6: Synthetic example recovering 8 parameters with no initial estimates. Small dark circle is initial approximation.

As an example of the robustness of the algorithm, figure 6 shows an example of the system recovering 8 parameters using multiple views of synthetic data. Moreover, the system makes absolutely no calculated starting approximation to the parameters (i.e. starts the minimization with inital values that are independent of data). The figure shows the inital data (cloud of dots) computed solution (line figure) and the initial estimate (the small dark line figure). The actual parameters were: $\epsilon_1 = 1.39, \epsilon_2 = .795, a_1 = 50, a_2 = 35, a_3 = 25, t_x = 3, t_y = 2, t_z = 1, \theta = .9, \eta = .6, \psi = .6$. and the recovered parameters were $\epsilon_1 = 1.387, \epsilon_2 = .773, a_1 = 50.5, a_2 = 35, a_3 = 27.5, t_x = 3.011, t_y = 1.98, t_z = 1, \theta = .9, \eta = .6, \psi = .6$. One component of the initial data (x, y, or z with equal likelihood) was perturbed by a random amount in the range [-2.5,2.5]. This noise was meant to simulate error in depth from multiple views.

The final three examples presented, show the fitting of actual range data from the Utah range database.

Figure 7 shows the elliptical indentation on the bottom of a soda can. The object which is defined by 590 data points. The system recovered the parameters $\epsilon_1 = 1.29, \epsilon_2 = .955, a_1 = .939, a_2 = .930, a_3 =$

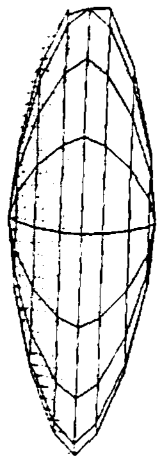

Figure 7: Reconstruction of a negative ellipsoid from real range data

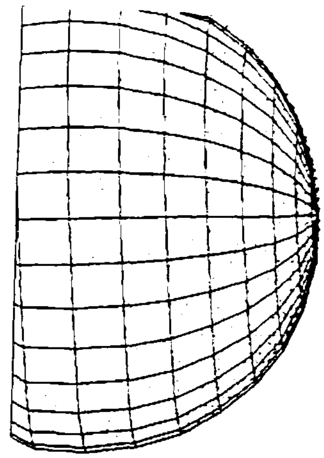

Figure 8: Reconstructed sphere from actual range data

$.277, t_x = -.096, t_y = -1.55, t_z = 2.07, \theta = -.05, \eta = 0, \psi = 0$. and the error was .0293.

Figure 8 shows a quasi-spherical object which is defined by 859 data points. The system recovered the parameters $\epsilon_1 = .994, \epsilon_2 = .951, a_1 = 1.19, a_2 = 1.13, a_3 = 1.13, t_x = .4729, t_y = 1.437, t_z = -1.457, \theta = -.05, \eta = -.04, \psi = 0$ and the error was .021.

Figure 9 shows the cylindrical portion of a soda can defined by 1645 data points. When using the above described estimations techniques, the system recovered the parameters $\epsilon_1 = 2.0, \epsilon_2 = .88, a_1 = 1, a_2 = 1.3, a_3 = 19.59, t_x = -.19, t_y = -1.69, t_z = -.834, \theta = -.04, \eta = .05, \psi = 0.0$. and the error was .0079. When the inside-out function was modified to include an extra multiplicative factor of $a_1a_2a_3$, the result was the object in figure 10. Unfortunately, the residual errors cannot be compared.

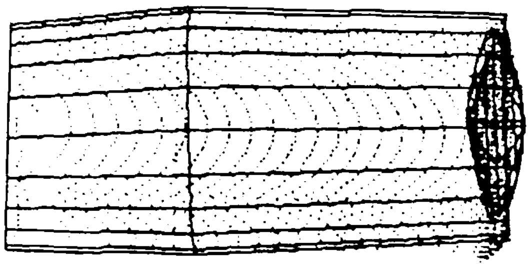

Figure 9: Example using real range data example of soda can, best fit surface with standard inside-out function.

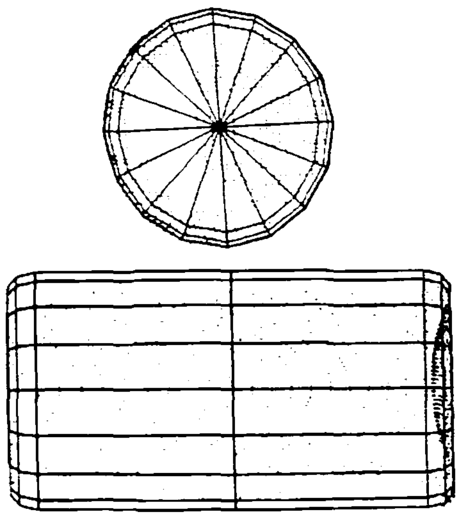

Figure 10: Example using real range data example of soda can, best fit with inside-out function multiplied by $a_1a_2a_3$.

Advantages and Drawbacks of Our Approach

The approach has a number of advantages. This section presents them and discusses them with respect to the two other systems for recovery of SEs known to these authors. The other systems are described in [Pentland-86a], [Bajcsy-Solina-87a], and [Bajcsy-Solina-87b]. The advantages of our approach are:

+ The system has demonstrated the ability to recover negative superquadrics. This is particularly important if we wish to use SEs combined with boolean operations.

+ The system has shown itself to be quite robust, even for small data sets, if given good estimates of rotation and reasonable bounds on translation and size parameters.

± The system demonstrated (using synthetic data) that multiple views (i.e. data on more than one side of the SE) result in much better reconstruction. This turned out be be quite important because it helps remove some ambiguity about the size of the object. (The authors believe this approach is better than assuming minimal volume as in both [Bajcsy-Solina-87b] and [Pentland-86a]. Unfortunately, it is often difficult to obtain multiple views from known positions.)

± The system combines the advantages of a fast nonlinear estimation technique (Gauss-Newton least square) with a Poisson randomness to avoid local minima and the use of stochastic resets if parameters stray beyond their estimated bounds. If the bounds are correct, this helps the system to avoid some local minima. Unfortunately, if the bounds are incorrect, we have condemned the system to search in a space where no solution can exist.

- The system is based on the minimization of the inside-out function. Unfortunately, this function is not simply related to distance of a point to the surface. This leads to minima that are nonintuitive.

- If there are many more points on one portion of the surface than another, this surface will dominate the search because, as far as least squares minimization is concerned, the small number of points on other sides are "noise". This is partially alleviated by using multiple views of an object (assuming we can). Even then we may overly constrain the "top" as compared to the sides. (We share this problem with [Bajcsy-Solina-87b].)

- Currently, if the estimates of the parameters are poor, the system may converge to a point far from the true minima. This can be greatly alleviated by using data from multiple views which makes the estimates significantly more accurate.

Conclusions and Future Direction

This paper has shown that there is some promise in the use of a least square minimization of the inside-out function as an approach to recovering superquadrics from depth data, but the system is currently dependent on reasonable estimates of the moments of the object. Thus, our research investigated the use of multiple views and found that they greatly increase the reliability of the system.

This paper also demonstrated that the system can recognize negative superquadrics, an important consideration if we are going to attempt to segment scenes made with CSG operations.

In the near future, there are a number of extensions the authors plan to make to the system. The first is to include bending and tapering parameters so we can model more objects, and then begin experimenting with possible changes to the function to be minimized (currently the inside-out function) to give the system better error and convergence properties.

Future plans for the system also include extensions to incorporate surface derivative information. This will be accomplished by minimizing a sum with (some variant of) both the inside-out function and a differentiated form of the inside-out function. This is different from the approach suggested in [Bajcsy-Solina-87a] where normals are considered. Our approach has advantages over directly using surface normals because it is applicable when only one derivative can be accurately measured and if normals are available, they provide two pieces of derivative information. Conditions when only one derivative can be accurately measured arise on occluding contours of nearly "square" corners. While theoretically one knows the actual normal at the occluding contour, small errors in the measurement of the location of the contour can greatly affect the normal.

Of course one of the most important avenues for future research will be attacking the segmentation problem, our current plans are to attempt at least two approaches: pure growing of superquadrics from small data patches, and a skeletonization (to find axis) followed by both growing and splitting of superquadric solids.

References

[Bajcsy-Solina-87a] R. Bajcsy and F. Solina. *Three dimensional object representation revisted.* Technical Report MS-CIS-87-19, GRASP LAB 99, University of Pennsylvania, March 1987.

[Bajcsy-Solina-87b] R. Bajcsy and F. Solina. Three dimensional object representation revisted. In *Proceedings of the IEEE Computer Society International Conference on Computer Vision*, pages 231–240, IEEE, June 1987.

[Barr-81] A. H. Barr. Superquadrics and angle preserving transformations. *IEEE Computer Graphics and Applications*, 1:11–23, January 1981.

[Bolle-Cooper-86] Rueud M. Bolle and D.B. Cooper. On optimally combining peices of information, with application to estimating 3-d complex-object position from range data. *PAMI*, ():, September 1986.

[Brooks-85] R.A. Brooks. Model based 3-d interpretation of 2-d images. In A. Pentland, editor, *From Pixels to Predicated*, Ablex Publishing Co., Norwood, N.J., 1985.

[Brooks-Binford-80] R.A. Brooks and T. O. Binford. Representing and reasoning about specific scenes. In *Proceedings of the DARPA Image Understanding Workshop*, pages 95–103, DARPA, April 1980.

[Gardiner-65] M. Gardiner. The superellipse: a curve that lies between the ellipse and the rectangle. *Scientific American*, September 1965.

[Pentland-86a] A. Pentland. *Recognition by Parts.* Technical Report 406, SRI International, December 1986.

[Pentland-86b] A. Pentland. Twoards an ideal 3-d cas system. In *Proceedings of the SPIE Conference on Machine Vision and the Man-Machine Interface*, SPIE, Janurary 1986. Order # 758-20.

[Rao-Nevatia-86] K. Rao and R. Nevatia. Generalized cone descriptions from sparse 3-d data. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 256–263, IEEE, 1986.

[Turner-74] K.J. Turner. *Computer perception of curved objects using a televesion camera.* PhD thesis, University of Edinburgh, 1974.

[utah-85] utah. *The University of Utah range database manual.* Technical Report , University of Utah, 1985.