A Hybrid Approach to the Recovery of Deformable Superquadric Models from 3D Data

James Sinnott and Toby Howard
Advanced Interfaces Group
Department of Computer Science
University of Manchester
Manchester M13 9PL
sinnottj@cs.man.ac.uk

Abstract

The problem of recovering the shape of objects from three-dimensional data is important to many areas of computer graphics and vision. We present here a method for the recovery of single-part objects from unstructured 3D points sets, based on the fitting of deformable superquadric models. The limitations of least-squares minimisation as a technique for fitting superquadric models are discussed. After investigating the possibility of using a genetic algorithm as an alternative, we propose a hybrid approach to the recovery of deformable superquadrics based on a two-stage fitting process that combines a genetic algorithm and nonlinear least-squares minimisation.

1. Introduction

The problem of recovering the shape of objects from unstructured 3D data is important in many areas of computer graphics and computer vision, including robotics, medical imaging and the automatic construction of virtual environments. In the last 25 years, much work has focussed on finding suitable models for the recovery of objects from 3D data. This work has largely proposed the use of some form of parametric model, most commonly the superquadrics.

Superquadrics are simple parametric models that can represent a large range of shapes with a small number of parameters, and have mathematical properties that make them particularity suited to efficient model recovery. The shape of an object is recovered by fitting a superquadric model as closely as possible to the 3D data. Pentland [1] was first to propose the use of superquadrics as a model for object recovery, and in recent years, many researchers [2], [3], [4], [5], [6], [7] have reported success in the recovery of superquadrics models, often combined with global or local deformations. The most commonly used method of fitting superquadric models to 3D data is through nonlinear least-squares minimisation of an error-of-fit function. However, least-squares minimisation approaches often perform poorly when searching complex parameter spaces and can only guarantee convergence to a local minimum. In this paper we present an alternative *hybrid approach* to the recovery of deformable superquadric models from 3D data. We propose a two-stage process for fitting a deformable superquadric to a set of points, based on a genetic algorithm and a nonlinear least-squares minimisation routine.

The rest of the paper is organised as follows. Section 2 provides a formal definition of our object model, superquadrics with global parametric deformations. Section 3 defines the problem of fitting deformable superquadric models to 3D points and describes the standard approach to solving this problem through the use of nonlinear least-squares minimisation. Section 4 investigates the possibility of using genetic algorithms as an alternative. Following this discussion, Section 5 proposes a novel hybrid approach to fitting deformable superquadrics to 3D points, and Section 6 presents the results of testing this hybrid approach. Finally, Section 7 concludes the paper and discusses areas for future work.

2. Model definition: superquadrics with global deformations

Superquadrics [8] are a family of parametric solids derived from the basic quadric surfaces and solids. Extra flexibility in shape representation is achieved by raising each trigonometric term in the quadric equations to an exponent. These exponents control the relative roundness and squareness along the major axes of the surface. By altering the value of the exponents, a wide range of forms may be generated: spheres, cylinders, parallelepipeds, pinched stars and the shapes in between. The following position vector $\underline{x}$ defines a superquadric surface:

$$\underline{x}(\eta,\omega) = \begin{bmatrix} a_1 \cos^{\varepsilon_1}\eta \cos^{\varepsilon_2}\omega \\ a_2 \cos^{\varepsilon_1}\eta \sin^{\varepsilon_2}\omega \\ a_3 \sin^{\varepsilon_1}\eta \end{bmatrix} \qquad \begin{matrix} -\pi/2 \le \eta \le \pi/2 \\ -\pi \le \omega < \pi \end{matrix}$$

The vector $\underline{x}$ sweeps out a closed surface as the two independent parameters, η and ω, change in the given intervals. Terms a_1, a_2 and a_3 define the superquadric scaling in the x, y and z directions, respectively. Exponents ε_1 and ε_2 control the curvature of the surface in the north-south and east-west directions respectively. ε_1 and ε_2 are usually referred to as the superquadric shape parameters.

Although superquadrics can model a relatively wide range of shapes, they are not flexible enough to model the complex forms that might occur in a real-world scene. Many researchers [3], [7], [9] have proposed the use of parametric deformations to improve the representational range of superquadrics. By deforming a superquadric in some way (for example bending or tapering it), it can be made to model shapes that would not otherwise be possible. Of particular interest are deformations that are used in the manufacture of objects, or that allow the modelling of complex natural forms [7].

The superquadric model described here is augmented by three parametric deformations, based on work described by Barr [10]: global linear tapering, global axial twisting and global linear bending. Figure 1 shows examples of different superquadric models generated by varying the shape parameters, and superquadrics that have been subjected to tapering, twisting and bending deformations.

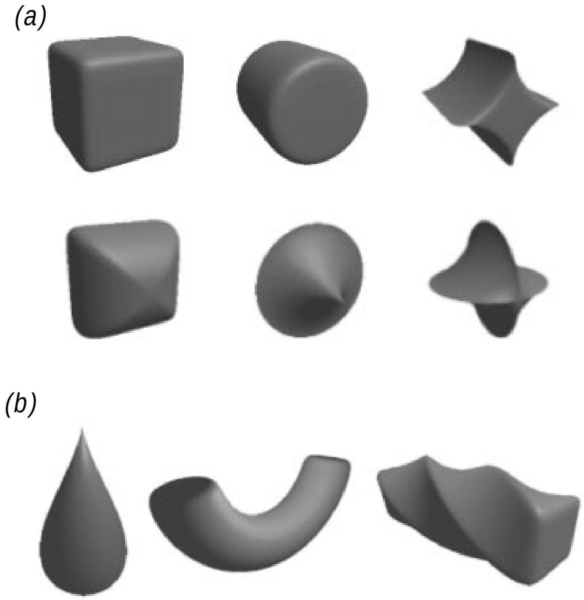

Figure 1. (a) Various superquadric forms generated by varying the shape parameters ε_1 and ε_2 and (b) superquadrics that have been subjected to tapering, bending and twisting deformations.

3. Fitting deformable superquadrics to 3D points

The problem we wish to solve is that of recovering a superquadric model from a set of 3D points. Given a set of N points (x_i, y_i, z_i), $i = 0,\ldots,N$, the task is to find the superquadric model that most closely fits those points i.e. such that average distance of the points from its surface is as small as possible.

A generally positioned deformable superquadric is defined here by 15 parameters, specifying its shape, size, position, orientation and any deformations that are applied. The problem is to find suitable values for each of these parameters, such that they describe the superquadric that best fits the points. In order to do this, we need a function F that can compute how close a given point is to the superquadric's surface. By evaluating this function over all the points, and taking the average, we get an error-of-fit (EOF) measure for the model. The problem then becomes that of finding the set of parameters that minimise this EOF measure:

$$\min \frac{1}{N} \sum_{i=1}^{N} F(x_i, y_i, z_i)$$

The work described here uses a version of the Euclidean distance EOF measure proposed by Gross and Boult [11], extended to take into account any deformations applied to the model. The EOF is computed as the mean Euclidean distance between each data point and the corresponding point on the surface of the superquadric, along the line connecting the data point with the centre of the superquadric.

3.1. Nonlinear least-squares minimisation

The most commonly used method for fitting a superquadric model to a set of 3D points is nonlinear least-squares minimisation (NLLSM) of the EOF measure, using an algorithm such as that proposed by Levenberg-Marquardt [12]. The Levenberg-Marquardt (LM) algorithm is supplied with a set of model parameters, corresponding to an initial point in the parameter space. The algorithm then proceeds iteratively, attempting to converge on a minimum in parameter space, i.e. a set of parameters that minimise the EOF measure.

3.2. Disadvantages of NLLSM

Least-squares minimisation techniques such as the LM algorithm follow a rigidly defined strategy for finding a minimum in parameter space and the minimisation process can, in general, only guarantee convergence to a *local* minimum. This can be a significant problem when the parameter space is of complicated topology, i.e. exhibits multiple local minima, as is the case when fitting superquadric models to 3D points. The initial set of model parameters supplied to the algorithm determines to *which* minimum it will converge. Unless minimisation is started in the vicinity of the global minimum it is unlikely to reach it, and will converge to a minimum local to the starting point instead. The success of the algorithm in reaching the global minimum depends greatly on its starting point in the parameter space, and this reliance on the initial model parameters represents the main weakness of the least-squares minimisation approach.

Most researchers attempt to maximise the effectiveness of the LM algorithm by providing it with a set of initial model parameters that predict in some way how the model will fit the data points. For example, by analysing the points it may be possible to estimate both the position and orientation of the model that most closely fits them, with reasonable accuracy. This "initial fit" of the model to the data will hopefully provide the algorithm with a reasonably direct route to the global minimum.

4. Genetic algorithms for superquadric fitting

This section discuses the possibility of using a genetic algorithm as an alternative method for fitting deformable superquadric models to 3D points.

Genetic algorithms (GAs), first introduced by Holland [13], are class of robust search and optimisation procedures based on the ideas of genetics and the Darwinian theory of evolution. In contrast to least-squares minimisation techniques, a GA can explore many regions of a parameter space in parallel. This property allows a GA to avoid becoming trapped in a local minimum, even when searching a topologically complex parameter space. GAs have been successfully used in the recovery of other types of parametric model; Taylor *et al* have used them with a variety of 2D active shape models [14], and Vaerman & Thiran have used GAs for fitting Hyperquadrics to range data [15].

4.1. Basic principles of a GA

A GA works by attempting to *evolve* a population of potential solutions to a problem in order to find an optimal solution. A typical GA consists of the following components [16], [17]:

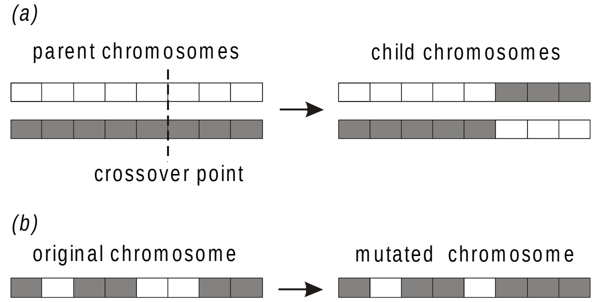

Figure 2. Examples of the genetic operators (a) crossover and (b) mutation.

- a way of encoding a potential solution to the problem, known as a *chromosome*, usually as a string of bits, integers or real numbers,
- a *population* of chromosomes,
- a *fitness function*,
- a set of *genetic operators*,
- and a method of *parent selection*.

The fitness function decodes a chromosome and applies the solution it represents to the problem. The chromosome is awarded a *fitness* corresponding to how 'good' a solution it represents. The two most common genetic operators used in a GA are *crossover* and *mutation*. Crossover takes two existing chromosomes and 'mates' them, producing two new chromosomes that combine the characteristics of each of their 'parents'. The mutation operator is applied to all newly created chromosomes; mutation randomly alters the value of one or more of a chromosome's component values. Figure 2 illustrates both the crossover and mutation operators, applied to simple bit-string chromosomes. Parent selection is the process by which chromosomes are chosen from the population for mating, in order to produce new chromosomes. Parent selection is generally a 'biased random' procedure, the likelihood that a particular individual will be chosen is proportional to its fitness in the population. In this way a 'fit' individual is more likely to be chosen for mating than an 'unfit' one.

Execution of a GA search begins with the creation of an initial population. This population is either random, or may be seeded with chromosomes that represent a potentially good solution, if available. The algorithm then enters it *generational loop*. A new population of chromosomes is created by selecting individuals from the existing population and mating them to produce children. The newly created population replaces the old one, and the process repeats. The search is terminated when either (a) at pre-set number of generations has been performed, or (b) a particular individual has been reproduced so often that it has taken over the population, which is then said to have *converged*.

Key to the success of a GA are the concepts of parent selection, crossover and mutation. When two chromosomes are mated, they pass on different combinations of their genes to each child. There is the possibility that one of these new combinations of genes will represent a *better* solution to the problem than either of the parent chromosomes. By biasing parent selection towards the fitter individuals in the population, it is more likely that those individuals will pass on their genes to the next generation. Each new generation should contain better solutions than the previous and, over several generations, the weaker solutions will be phased out as the better solutions take over the population. Effectively, crossover enables the search to move towards 'promising' regions of the parameter space. The mutation operator provides the GA with another tool to help it avoid becoming trapped in a local minimum. By allowing the possibility of new, untried, gene values being introduced into the population, mutation enables the GA to randomly sample new points in the parameter space throughout the search.

4.2. Applying a GA to superquadric fitting

The problem of fitting a superquadric to a set of 3D points, as described in Section 3.1, can be shown to fit neatly into the context of a genetic algorithm. There are two aspects of a GA that tie it to the problem domain: the method of encoding a solution as a chromosome and the method by which a chromosome's fitness is computed. Thus:

- A chromosome, is the set of parameter values that define a deformable superquadric model.
- The fitness function will generate the superquadric model defined by the parameter values stored in a chromosome and compute the EOF value for that model to the points. This EOF value is then used to compute the fitness of the chromosome.

The parameters that describe a deformable superquadric model are encoded as a string of floating point numbers, whose values belong to a convex domain. Each parameter has a set of constraints, specifying upper and lower bounds on its value. Certain aspects of the GA must be adapted to take into account these constraints, in particular population initialisation and the genetic operators of crossover and mutation. The rest of the GA is problem domain-independent and would be implemented as described in Section 4.1.

4.3. Potential problem with using a GA

Given the discussion above, use of a GA would appear to be a viable alternative to NLLSM, avoiding the drawbacks associated with that approach. However, there is a significant obstacle to applying a GA to a complex nonlinear optimisation problem such as that described here, specifically the high computational cost due to the slow convergence rate of the GA [18]. In general, a GA will take significantly longer to converge than a least-squares minimisation routine applied to the same problem. This is primarily because a GA does not make direct use of local parameter-space information to determine the most promising search direction. Rather, the GA explores the parameter space in a less directional fashion, considering many different regions in parallel.

In our own comparisons of the performance of a GA with that of NLLSM when applied to the problem of superquadric fitting, we have found that, for the same data set, the GA takes significantly longer to converge. This problem of slow convergence made the use of a GA a considerably less attractive prospect.

4.4. Combining a GA with NLLSM

A popular strategy for alleviating the problem of slow convergence is to combine the GA with another, more 'local' search technique [18] [19] [20] [21]. The idea is to combine the advantages of the GA with those of the local search technique, whilst avoiding their respective disadvantages. Through the use of the GA, a combined approach should be less prone to converging on local optima than the local search technique alone. By passing control of the search over to the local search technique at an appropriate point, the combined approach should converge more quickly than if a standard GA was used. We therefore suggest that if the two techniques, GA and NLLSM, were combined, a fitting system could be produced that:

1. Does not require 'good' initial fit in order to converge on an optimal fit
2. Converges quickly and accurately to the optimal fit.

The basic principle of this approach is that we use the GA to find a 'good' fit to the points, which can then be used as a starting point for the NLLSM routine. The next section discusses this *hybrid approach* in more detail.

5. The hybrid approach

5.1. Fitting strategy

In this section we propose a novel *hybrid approach* to fitting deformable superquadric models to 3D points, based on the combination of a genetic algorithm (GA) and NLLSM. The hybrid approach is based on a two-stage fitting process:

1. The GA is used to locate a model that represents a 'good' fit to the points.
2. NLLSM is used to refine the fit, extracting the parameters of the model that most closely fits the points.

The first stage exploits the main advantage of the GA: finding the region of parameter space containing the global minimum, while avoiding becoming trapped in a local minimum. The second stage exploits the advantages of NLLSM: fast and accurate convergence to the global minimum.

Figure 3 illustrates the strategy used to fit a deformable superquadric model to a set of 3D points. Two inputs are provided to the system: the set of 3D points and a superquadric model that represents an initial guess fit to those points. Generation of the initial model is discussed in section 5.2. The points and the initial model are sent to the GA subsystem for the first stage of the fitting process. The GA executes until either a pre-set number of generations are completed, or a 'good' chromosome is found. In this case a 'good' chromosome represents a model that can be used as a suitable starting point for the NLLSM routine. The user must decide on an appropriate fitness value for a 'good' chromosome beforehand. If at any point the user decides that the GA has found a model that fits the data points sufficiently, he can stop it manually. Once execution of the GA has stopped, the best-fit model is passed to the NLLSM subsystem, for which it now represents the input model. When minimisation has finished, the final set of optimised model parameters is returned. These parameters are used to generate an output model, representing the best fit to the data points that the system could find.

5.2. Implementation

The hybrid superquadric fitting system consists of two subsystems, which control the GA and the NLLSM routines respectively. The GA subsystem was implemented as described in section 4.2. The NLLSM subsystem was based on the DN2FB routine, an implementation of the Levenberg-Marquardt algorithm freely available from the netlib repository [22].

The hybrid fitting system must be supplied with an initial 'guess fit' model, which provides a starting point in parameter space, from which it can attempt to locate the minimum. To generate an the initial model the 3D points are analysed and the information extracted is used to compute estimates of some of the model parameters:

- The initial position of the model in space is estimated from the centre of gravity of the points.
- The initial orientation of the model is estimated from the eigenvectors of the matrix of central moments of the points [7]. The z-axis of the model is aligned along the eigenvector with the least moment of inertia.
- The scale of the initial model is estimated from the dimensions of the convex hull of the points.

5.3. Expected benefits of the hybrid approach

The aim of the hybrid approach was to produce a fitting system that, like the GA, did not require a 'good' initial fit in order to converge on, or close to, an optimal fit, and like NLLSM, was able to converge quickly and accurately to that optimal fit. We anticipate that the hybrid approach will, for the same data, be slower to converge than NLLSM used alone. We suggest, however, that the increase in convergence time will be an acceptable trade-off, given the expected improvement in the fits obtained when both approaches are started from the same initial fit.

6. Testing and results

The hybrid approach was tested on a number of example data sets and the results were compared to those

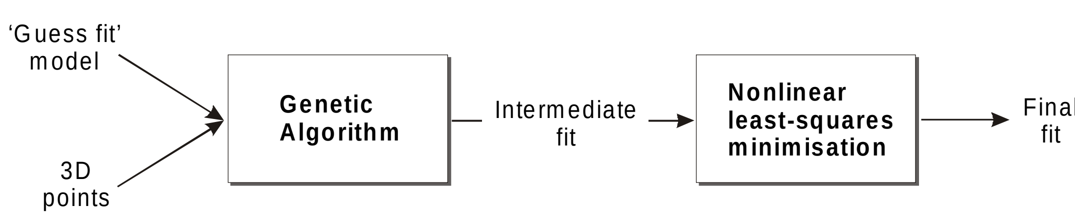

Figure 3. The hybrid approach fitting strategy.

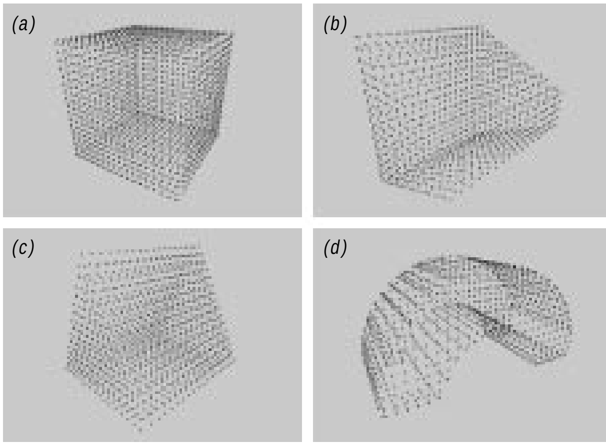

Figure 4. The data sets: (a) BOX, (b) TRUNCATED WEDGE, (c) TRIANGULAR PRISM and (d) ARCH.

obtained with an equivalent fitting system that uses the NLLSM approach alone. The data used consist of 3D point sets, sampled from the surface of simple, single-part objects. We use three objects taken from the Washington Sate University 3D image/model database [23], BOX, TRIANGULAR PRISM and TRUNCATED WEDGE, and ARCH, a model of a simple children's building block. Each of these data sets is shown in Figure 4. These data sets were chosen as, in each case, a set of model parameters exists that represent an optimal (or near-optimal) fit to the points.

In each test, both the hybrid and the NLLSM-based fitting system were supplied with the default initial 'guess-fit' models, generated as described in section 5.2. In the case of the hybrid fitting system, the GA subsystem was allowed to search until either a model was found with an EOF value of less than 0.1, or 100 generations had been completed. The GA used a population size of 200 for each of the tests.

The results of the tests were generated in terms of (a) the EOF value of the final fit, (b) the time (in seconds) taken to converge on the final fit and (c) the visual appearance of the final fit. Figures 5, 6, 7 and 8 give the results of the tests. Figure 5 gives the results in terms of the EOF value of the final fit, while Figure 6 compares the time taken by each approach to converge on this fit. Figure 7 illustrates the final fits obtained by the NLLSM-based fitting system for each of the data sets. Figure 8 illustrates the fits obtained by the hybrid approach, showing both the intermediate fit, produced by the GA and supplied as input to the NLLSM routine, and the final fit obtained for each data set.

Firstly, we consider the results in terms of the visual appearance and EOF values of the final fits obtained. Comparing the visual appearance of the final fits, we find that the results for data set BOX are comparable, but in all other cases, the fits obtained by the hybrid approach appear much better than those obtained with NLLSM. In each case, the EOF values of the final fits confirm these findings. Secondly, we consider the results in terms of the time taken to converge on a final fit. As expected, we find the hybrid approach takes longer to converge than NLLSM. Specifically, the hybrid approach required approximately 10 times as much time to converge for each of the data sets.

In summary, the results establish that the hybrid approach successfully addresses the problems observed with using NLLSM alone. For the same initial point in

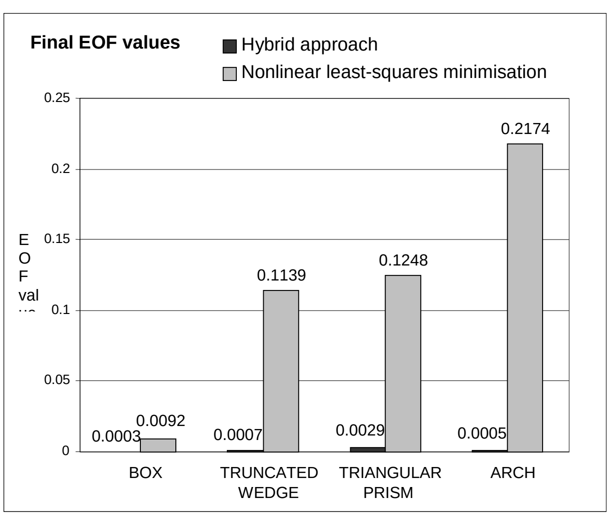

Figure 5. The EOF values of the final fits obtained with the hybrid approach compared to those for the nonlinear least-squares minimisation.

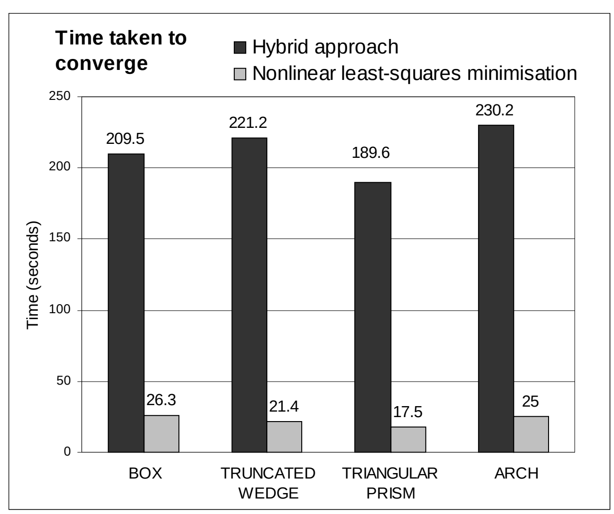

Figure 6. The time taken to converge on a final fit by the hybrid approach compared to those for the nonlinear least-squares minimisation.

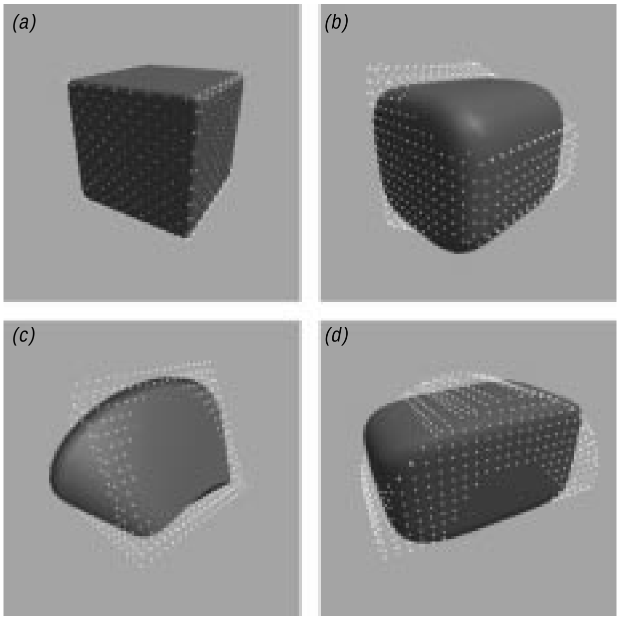

Figure 7. Final fits obtained with the NLLSM fitting system on data sets (a) BOX, (b) TRUNCATED WEDGE, (c) TRIANGULAR prism and (d) ARCH.

parameter space, the hybrid approach was found to converge on a fit that was better than that obtained by NLLSM, both visually and in terms of EOF value. However, the hybrid approach was found to require longer to converge than NLLSM used alone.

7. Conclusion

In this paper, we have presented a novel hybrid approach to solving the problem of fitting deformable superquadric models to 3D point sets.

We have discussed the standard approach to solving this problem, NLLSM, and highlighted the main drawbacks associated with this approach. It was found that, in general, NLLSM required a good-quality initial fit to the data, in order to converge to an optimal fit. Without such an initial fit, NLLSM could only guarantee convergence to a local minimum in the parameter space, most likely corresponding to a sub-optimal fit.

The use of a Genetic Algorithm a potential alternative to NLLSM was discussed. It was established that a GA would be less prone to becoming trapped in local minima, and that the success of a GA in finding an optimal fit would not be so dependent on the initial ‘guess’ fit, i.e. the starting point in parameter space. However, it was also established that GAs often suffer long convergence times when solving complex optimisation problems and as such, the use of a GA alone was not considered a viable alternative. The possibility of combining a GA with a

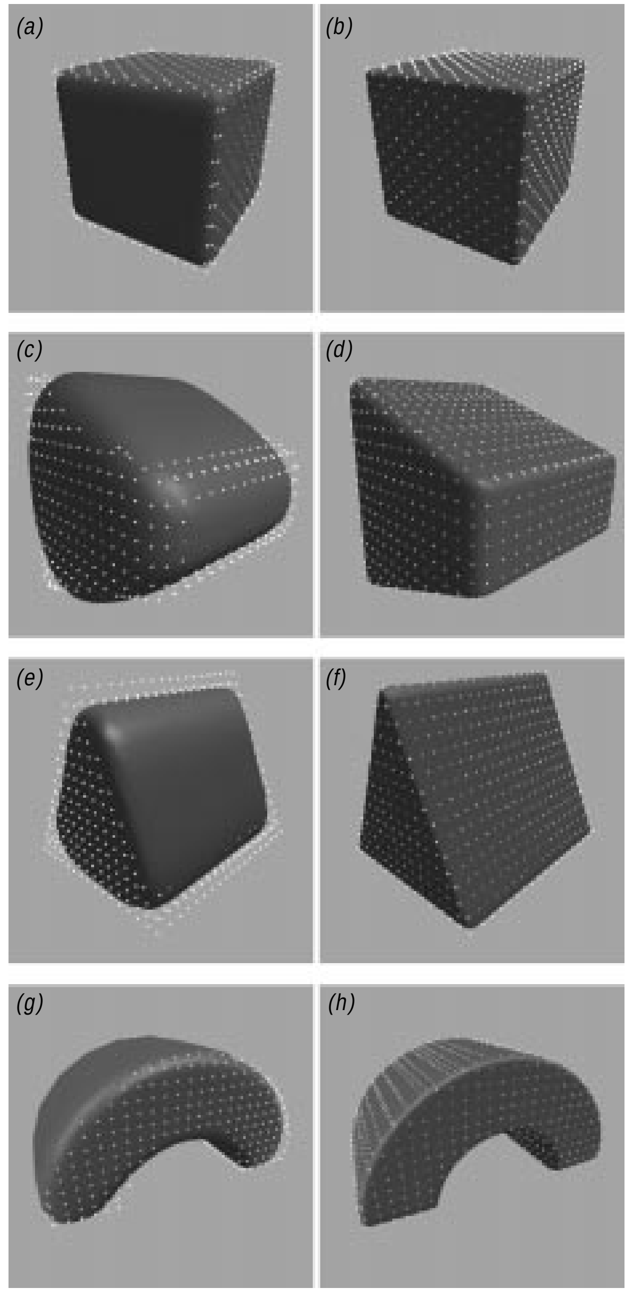

Figure 8. Results for the hybrid approach. (a) & (b) show intermediate and final fits to the data set BOX, (c) & (d) show intermediate and final fits to the data set TRUNCATED WEDGE, (e) & (f) show intermediate and final fits to the data set TRIANGULAR PRISM, and (g) & (h) show intermediate and final fits to the data set ARCH.

local search technique, such as NLLSM, to overcome the problems of poor convergence rates, while maintaining the advantage of a GA was then considered.

This discussion lead to the proposal of a hybrid approach to fitting superquadric-based models to 3D point sets. This hybrid approach combines both a GA and NLLSM in a two-stage fitting process, the aim being to overcome the problems associated with each technique. After testing, we were able to show that, for the same data, the hybrid approach successfully fulfilled its aims.

At the time of writing, the hybrid approach can only fit single deformable superquadric models to points corresponding to simple, single-part objects. This represents a major restriction on the form of data that can be used, and we intend to address this restriction in our future work.

It will be necessary to improve the system to allow for the recovery of much more complex, multi-part objects. This will involve adapting it to allow the fitting of part-models, constructed from structured arrangements of deformable superquadric primitives.

8. References

[1] A. Pentland, "Perceptual Organisation and the Representation of Natural Form", *Artificial Intelligence*, Vol. 28, 1986, pp. 293-331.

[2] T.E. Boult, A.D. Gross, "Recovery of Superquadrics from Depth Information", in *Proc. of 1987 Workshop on Spatial Reasoning & Multi Sensor Fusion*, October 1987, pp. 128-137.

[3] E. Bardinet, L.D. Cohen, N. Ayache, "A Parametric Deformable Model to Fit Unstructured 3D Data", *Research Report 2617*, INRIA, Sophia-Antipolis, July 1995.

[4] F.P. Ferrie, J. Lagarde, P. Whaite, "Darboux Frames, Snakes, and Superquadrics: Geometry from the Bottom Up", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 8, August 1993, pp. 771-784.

[5] A. Gupta, R. Bajcsy, "Volumetric Segmentation of Range Images of 3D Objects Using Superquadric Models", *Computer Vision, Graphics and Image Processing: Image Understanding*, Vol. 58, November 1993, pp. 302-326.

[6] A. Leonardis, A. Jaklic, F. Solina, "Superquadrics for Segmenting and Modelling Range Data", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 11, November 1997, pp. 1289-1295.

[7] F. Solina, R. Bajcsy, "Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 2, February 1990, pp. 131-147.

[8] A.H. Barr, "Superquadrics and Angle Preserving Transformations", *IEEE Transactions on Computer Graphics and Applications*, Vol. 1, No. 1, January 1981, pp. 11-23.

[9] D. Terzopoulos, D. Metaxas, "Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 7, July 1991, pp. 703-714.

[10] A.H. Barr, "Global and Local Deformations of Solid Primitives", *Computer Graphics*, Vol. 18, No. 3, 1984, pp. 21-30.

[11] A.D. Gross, T.E. Boult, "Error of Fit Measures for Recovering Parametric Solids", *Second international Conference on Computer Vision*, Tampa, Fl., December 1988, pp. 690-694.

[12] W. Press, B. Flannery, S. Teukolsky, W. Vatterling, *Numerical Recipes in C - The Art of Scientific Computing*, Cambridge University Press, 1992.

[13] J.H. Holland, *Adaption in Natural and Artificial Systems*, University of Michigan Press, 1975.

[14] A. Hill, C.J. Taylor, "Model-Based Image Interpretation Using Genetic Algorithms", *Image and Vision Computing*, Vol. 10, No. 5, June 1992, pp 295-300.

[15] V. Vaerman, J.P. Thiran, "Multidimensional Object Representation by a Parametric Deformable Model", *Internal Report LTS 98.04*, Signal processing Lab (LTS), Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, September 1998.

[16] D. E. Goldberg, *Genetic Algorithms in Search, Optimisation and Machine Learning*, Addison-Wesley Publishing Company, Inc., 1989.

[17] L. Davis, *Handbook of genetic algorithms*, Van Nostrand Reinhold, 1991.

[18] J. Yen, D. Liao, D. Randolph, B. Lee, "A Hybrid Approach to Modelling Metabolic Systems using Genetic Algorithm and Simplex Method", *Proc. 11th IEEE conference on Artificial Intelligence for Applications (CAIA95)*, Los Angeles, CA, February 1995, pp. 277-283.

[19] H. Ishibuchi, N. Yamamoto, T. Murata, H, Tanaka, "Genetic Algorithms and Neighbourhood Search Algorithms for Fuzzy Flowshop Scheduling Problems", *Fuzzy Sets and systems*, Vol. 67, No. 1, 1994, pp 81-100.

[20] K.E. Mathais, L.D. Whitley, C. Stork, T. Kusama, "Staged Hybrid Genetic Search for Seismic Data Imaging", *Proc. 1st IEEE Conference on Evolutionary Computation*, Orlando, Florida, June 1994, pp 356-361.

[21] T.Y. Park, G.F. Froment, "A Hybrid Genetic Algorithm for the Estimation of Parameters in Detailed Kinetic Models", *Computers and Chemical Engineering*, Vol. 22, 1998, pp 103-110.

[22] www.netlib.org

[23] *The Washington State University 3D Image/Object Database*, http://sampl.eng.ohio-state.edu/~sampl/data/