Fitting Superellipses

Paul L. Rosin

Department of Computer Science

Cardiff University

UK

Paul.Rosin@cs.cf.ac.uk

Abstract

In the literature, methods for fitting superellipses to data tend to be computationally expensive due to the non-linear nature of the problem. This paper describes and tests several fitting techniques which provide different trade-offs between efficiency and accuracy. In addition, we describe various alternative error of fits (EOF) that can be applied by most superellipse fitting methods.

keywords: curve, superellipse, fitting, error measure

1 Introduction

A goal in low level vision is to represent image features such as edge lists by compact and expressive primitives. Although straight lines are commonly used they have shortcomings if the objects in the scene or the object models contain more complex curved parts. This has lead to higher order primitives such as circles, ellipses, and splines. This paper discusses the superellipse, which can be defined as

$$\left(\frac{x}{a}\right)^{\frac{2}{\epsilon}} + \left(\frac{y}{b}\right)^{\frac{2}{\epsilon}} = 1.$$

With only one additional parameter (ϵ) this extends the ellipse to cover a range of shapes including rectangles, ovals, ellipses, and diamonds. Moreover, a greater gamut of shapes can be achieved by allowing parameterised deformations such as tapering, bending, etc. [2]. However, there is one disadvantage, namely the introduction of the nonlinear parameter makes parameter estimation more difficult. In general a closed form solution is not possible. The literature contains a number of approaches to fitting superellipses such as gradient descent [5], Powell's direction set method [15], simulated annealing [20], exhaustive search [9], and point distribution model fitting [10]. Not only are they prone to finding suboptimal solutions, but they are all computationally expensive. In this paper we describe three aspects of fitting superellipses. First is the case where data covering the complete superellipse curve is available, which enables simpler more efficient fitting methods to be employed than otherwise possible. Second, we describe an approach to simplify the full 6D optimisation technique previously used to just a 1D optimisation. Next we survey, introduce, and compare nine error of fit (EOF) measures which can be used by fitting algorithms such as most of those listed above. Finally, we extensively test our methods on synthetic data to quantify their relative performances. In addition, the EOFs are compared by a set of quantitative assessment criteria.

2 Fitting to Complete Data

2.1 Area Based Method

If data covering the complete curve is available then most of the superellipse parameters can be obtained by geometric means rather than fitting functions, thereby avoiding selecting distance approximations which need to be iteratively minimised. There are various scenarios in which the complete curve data can be expected to be available. One is industrial inspection, in which the set-up is often controlled such that the object is not occluded. Another instance is when a larger figure has been decomposed into regions which are to be represented symbolically. Examples of applying superellipse/superquadric fitting to figure subparts are given in Pentland [9] and Bennamoun and Boashash [3].

Our approach is based on the minimum bounding rectangle (MBR) (i.e. the rectangle with minimum area that contains all the data). Determining the rectangle involves first finding the convex hull, which for an n-gon is $O(n)$ [11]. Then the MBR can be found using Toussaint's optimal linear algorithm [18]. The MBR provides all the superellipse parameters (axis lengths, centre, and orientation) except for squareness (ϵ). Similar to the method of moments approach [7] for estimating the parameters of ellipses and other features we estimate the value of ϵ by calculating the area of the superelliptical region in the image and comparing it against the theoretical area which is

$$A = 4 \int_0^a b \left(1 - \left(\frac{x}{a} \right)^{\frac{2}{e}} \right)^{\frac{e}{2}} dx = 4ab\,_2F_1(\frac{\epsilon}{2}, -\frac{\epsilon}{2}, 1 + \frac{\epsilon}{2}; 1) = 4ab\frac{\Gamma\left(1 + \frac{\epsilon}{2}\right)^2}{\Gamma\left(1 + \epsilon\right)}. \tag{1}$$

To circumvent difficulties in inverting this equation we approximate the complex hypergeometric series by its first two terms [1] $_2F_1(a, b, c; z) \approx 1 + \frac{abz}{c}$ which when substituted into (1) gives $\epsilon = 1 - t \pm \sqrt{t^2 - 6t + 5}$ where $t = \frac{A}{4ab}$. For $\epsilon > 0$ we know that $t < 1$ which results in the two solutions having opposite signs, from which we keep only the positive solution. The errors introduced by truncating $_2F_1$ are evident in figure 1 which plots the estimated values as a function of ϵ (the error is independent of a and b). For $\epsilon < 1.8$ the estimates could be easily improved by a linear correction. As an alternative, we have chosen to use a table lookup with linear interpolation to correct the estimates. We note that the recent approach by Voss and Süße [19] also uses moments for fitting all the parameters of a superellipse except for ϵ which is iteratively estimated afterwards.
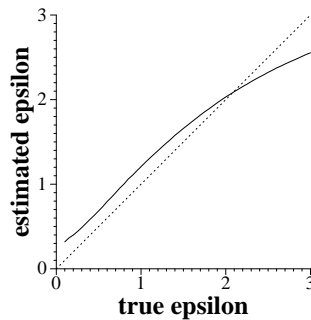

Figure 1: Estimated values of ϵ

2.2 Diagonal Based Method

We present another closed form method for estimating ϵ. The method just described has the disadvantage that any irregularities along its boundary are likely to distort the measurement of the area, and therefore affect the estimated value of ϵ. The second method only uses four points

along the boundary and is potentially less sensitive to such variations as the estimate will only be corrupted if these four points are incorrect. The position of the boundary points is a function of their orientation as well as the superellipse parameters. For the specific case of the intersection of the boundary with the diagonal of the superellipse's MBR the equation is $(x_i, y_i) = \frac{1}{2^{\frac{\epsilon}{2}}}(a, b)$ which can be easily inverted to provide an expression for squareness, $\epsilon = 2\frac{\log \frac{a}{x_i}}{\log 2}$. We determine the four intersection points of the boundary with the diagonals, obtaining four estimates of ϵ. The final estimate is taken as the median, providing some robustness to noise.

3   Fitting by 1D Optimisation

In an attempt to speed up the full 6D optimisation approach to fitting superellipses that we took previously [15] we describe here a 1D optimisation approach. It is based on defining the shape of the superellipse as a weighted average of an ellipse and a rectangle (for $0 < \epsilon \leq 1$). We fit an ellipse and a rectangle to the data to provide parameter estimates at these extremes. The simplifying assumption is that for superellipses at intermediate values of ϵ their remaining parameters will also lie at corresponding intermediate values. This enables a 1D search (Brent's method [12] is used) over $0 < \epsilon \leq 1$ where simultaneously the other parameter values are determined by linear interpolation of their values at the extremes of ϵ in order to minimise the error of fit.

The initial ellipse is determined using the recently developed ellipse-specific fitter [4]. The same method is used to initialise the 6D optimisation approach to fitting that will be tested in section 5. Fitting the initial rectangle is more problematic as we want a simple, reliable, and efficient method that can be applied to partial data. Although this rules out the minimum enclosed rectangle that we used for complete data we again make use of geometric methods. Like before we start by finding the convex hull. Next we determine the diameter (i.e. the most distant antipodal pair of points) of the convex hull which contains m points. This takes $O(m)$ time [11], so that the overall running time for processing the edge list of n points remains as $O(n)$. The diameter corresponds to the diagonal of the rectangle, and so the point of maximum deviation from the diagonal will be a corner of the rectangle. This directly provides three corners, and therefore the final corner can be found by symmetry.

With the rectangle fitting, in practise we found unacceptable errors were incurred for even moderate amounts of rounding of the rectangle's corners (i.e. for moderate values of ϵ). Some improvements were made by robustly fitting straight lines between the corners using the least median of squares (LMedS) method. Nevertheless, initial experiments with the overall optimisation revealed that it performed very poorly. It seemed that the parameter estimates of the ellipse and rectangle differed so much that the intermediate interpolated superellipses fitted the data extremely poorly. Therefore we simplified the above approach and set *all* the parameters using either the initial ellipse or rectangle except for the squareness which was found by a straightforward 1D optimisation. The two optimisations starting from both the ellipse and rectangle were carried out, and then the fit giving lower error was selected. Note that in order to compare the fits it is necessary to have an error function that is comparable over different values of ϵ – we used the central ray method (EOF_5).

4   Error Measures

Since many fitting techniques operate by minimising some error measure, the choice of this measure is of great importance. A function of the distance from the points along the normals to the superellipse would be a suitable measure except that it cannot be computed easily. Therefore

an approximation to this distance is required.[1] Below we describe some possible measures, both old and new. To visualise their effects their iso-distance contours are plotted[2] in figure 2 for a superellipse with $a = 400$, $b = 100$, $\epsilon = \frac{1}{2}$. To improve visualisation the distance values between contours are different for each plot – this does not affect the interpretation of their quality since multiplicative factors do not affect the fitting techniques. In addition, the assessment criteria developed in Rosin [13] are applied to quantify their linearity, curvature bias, asymmetry, and overall goodness. In the following we will assume that the superellipse has been transformed to the canonical position, i.e. centred at the origin and aligned with the co-ordinate axes. During the iterative fitting process this is performed using the previous estimate of the superellipse's parameters, while in the case of the complete data fitting methods the MBR provides the canonical frame.

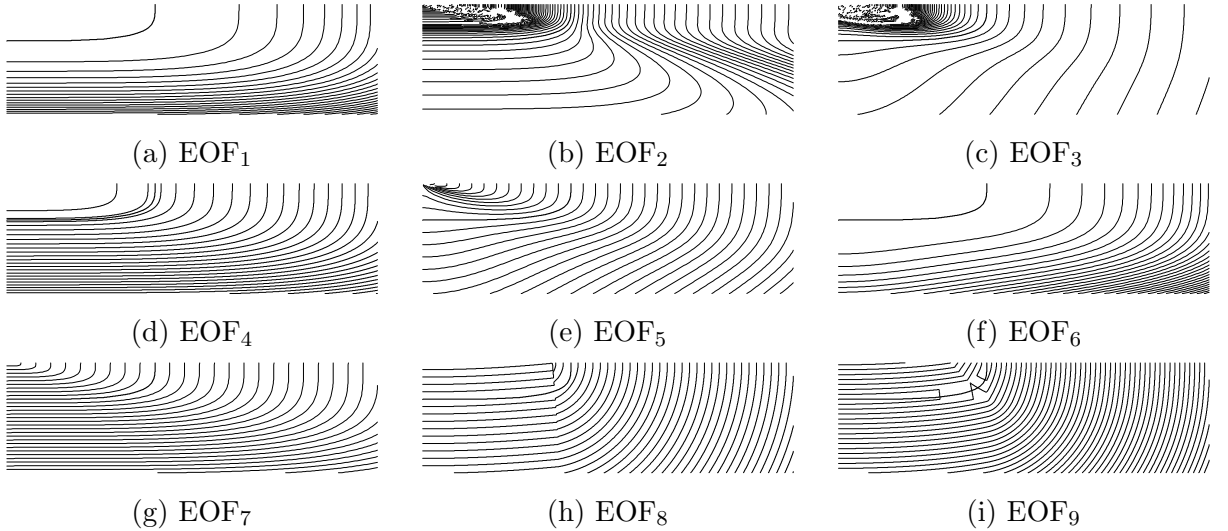|		
(a) EOF$_1$	(b) EOF$_2$	(c) EOF$_3$
(d) EOF$_4$	(e) EOF$_5$	(f) EOF$_6$
(g) EOF$_7$	(h) EOF$_8$	(i) EOF$_9$

Figure 2: Iso-value contours; the thick line shows the superellipse and the thin lines show points of constant distance from the superellipse according to various distance approximations

4.1 Algebraic Distances

Due to its simplicity the most commonly used error measure is the algebraic distance which is defined as

$$\text{EOF}_1 = Q(x, y) = \left(\frac{x}{a}\right)^{\frac{2}{\epsilon}} + \left(\frac{y}{b}\right)^{\frac{2}{\epsilon}} - 1.$$

For ellipse fitting the algebraic distance has the added advantage that its minimisation has a closed form solution, although unfortunately this does not hold for the superellipse. A standard approach for improving the algebraic distance is to inversely weight it by its gradient, which is equivalent to a first order Taylor's expansion of the true distance

$$\text{EOF}_2 = \frac{Q(x, y)}{|\nabla Q(x, y)|} = \frac{\left(\frac{x}{a}\right)^{\frac{2}{\epsilon}} + \left(\frac{y}{b}\right)^{\frac{2}{\epsilon}} - 1}{\sqrt{\left[\frac{2}{\epsilon x}\left(\frac{x}{a}\right)^{\frac{2}{\epsilon}}\right]^2 + \left[\frac{2}{\epsilon y}\left(\frac{y}{b}\right)^{\frac{2}{\epsilon}}\right]^2}}.$$

Although the linearity and curvature bias of the algebraic distance have been much improved there is still substantial error along the corner. In an effort to avoid these anomalies we have

---

[1]Even for simpler curves such as ellipses approximate distances are generally used for fitting [13].

[2]Artifacts from the plotting process have caused some contours to be missed near the centre of some superellipses.

4

considered replacing the derivative by the directional derivative along the ray from the point to the superellipse centre instead

$$\text{EOF}_3 = \frac{Q(x,y)}{|\nabla_R Q(x,y)|} = \frac{\left(\frac{x}{a}\right)^{\frac{2}{\epsilon}} + \left(\frac{y}{b}\right)^{\frac{2}{\epsilon}} - 1}{\frac{2}{\epsilon x}\left(\frac{x}{a}\right)^{\frac{2}{\epsilon}}\cos\theta + \frac{2}{\epsilon y}\left(\frac{y}{b}\right)^{\frac{2}{\epsilon}}\sin\theta}.$$

Again the linearity and curvature bias of the algebraic distance have been improved, but anomalies are still present. As an alternative correction to the algebraic distance for superquadrics Gross and Boult [5] suggested taking the ϵ'th power of the algebraic distance $\text{EOF}_4 = Q(x,y)^\epsilon$. We see that it does indeed improve linearity although the curvature bias is still evident.

4.2   Ray to Centre

Several approaches are based around the ray **OP** passing through the origin **O** (i.e. the centre of the superellipse) and the data point $\mathbf{P} = (x,y)$. The ray intersects the superellipse at $\mathbf{I} = (x_i, y_i)$, where

$$x_i = \left(\frac{1}{a^{\frac{2}{\epsilon}}} + \left(\frac{y}{xb}\right)^{\frac{2}{\epsilon}}\right)^{-\frac{\epsilon}{2}} ; \quad y_i = \frac{y}{x}x_i.$$

Denote the distances along the ray as $n = \overline{\mathbf{IP}}$ and $m = \overline{\mathbf{OI}}$. Then the distance approximation $\text{EOF}_5 = n$ has been used for ellipse [8] and superellipse [5] fitting, while for ellipse fitting Safaee-Rad *et al.* [16] derived the following weighted algebraic distance

$$\text{EOF}_6 = m\frac{1 + \frac{n}{2a}}{1 + \frac{n}{2m}}Q(x,y).$$

4.3   Similar Superellipse

Related to Stricker's approach [17] to estimating the distance to an ellipse we consider scaling the superellipse axes to find the "similar" superellipse which passes through the point. Solving the equation

$$\left(\frac{x}{sa}\right)^{\frac{2}{\epsilon}} + \left(\frac{y}{sa}\right)^{\frac{2}{\epsilon}} = 1$$

for the scale factor s and measuring distance as the difference in axis lengths between the two superellipses (e.g. $sa - a$) we can neglect the multiplicative factor a, yielding

$$\text{EOF}_7 = s - 1 = \left(\left(\frac{x}{a}\right)^{\frac{2}{\epsilon}} + \left(\frac{y}{b}\right)^{\frac{2}{\epsilon}}\right)^{\frac{\epsilon}{2}} - 1.$$

A weakness of the similar superellipse is that it is just as prone to the curvature bias as the algebraic distance. In other words, near the pointed end the superellipse does not need to be stretched out as much as near the flat end.

4.4   Quadrarc

An alternative approach to approximating the distance directly is to approximate the superellipse, and then find the distance to the approximate curve. A good candidate is the quadrarc which has previously been extensively used to generate reasonably accurate and simple approximations of the ellipse [14]. The quadrarc consists of four circular arcs with centres $(\pm h, 0)$ and $(0, \pm k)$ and radii $a - h$ and $b + k$ respectively such that they pass through the extremal points of the ellipse.
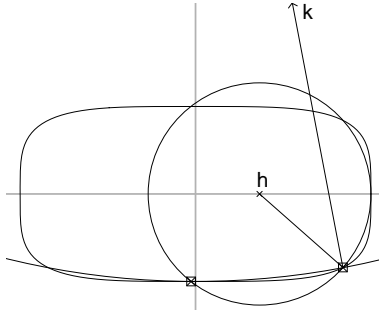
Figure 3: Quadrarc approximations of a superellipse; the intersections of the two circles are marked and the normals to the arc joints are drawn.

We adapt Knowlton's [6] method for constructing elliptical quadrarcs to the superellipse. Given the symmetry we only need describe results for the first quadrant. The intersection point of the diagonal of the superellipse's minimal enclosing rectangle and the superellipse itself is taken as the joint between the arcs, and constrains the circular arcs to

$$h = \frac{a^2 - x_i^2 - y_i^2}{2(a - x_i)}; \;\; k = -\frac{b^2 - x_i^2 - y_i^2}{2(b - y_i)}.$$

An example for $a = 200$, $b = 100$, $\epsilon = 0.5$ is shown in figure 3. To determine the distances to the curve we must choose the appropriate arc to calculate the normal to. We take the bisector of the lines joining the circles and the arc joint as the dividing line between the two sets of normals. The distance can then be simply calculated as the distance from the point to the circle centre less the circle radius

$$\text{EOF}_8 = \begin{cases} \sqrt{(h-x)^2 + y^2} - (a - h) & \text{if } y(a - b) + x\frac{by_i}{x_i - h} - \frac{a(y_i + k)}{x_i} + ak - bh < 0 \\ \sqrt{x^2 + (y + k)^2} - (b + k) & \text{otherwise.} \end{cases}$$
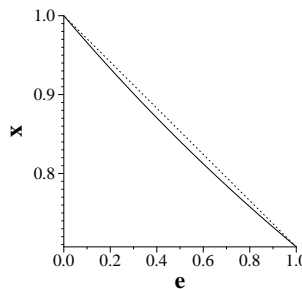
4.5 Weighted Average

Figure 4: Position of diagonal intercept against squareness

For $0 < \epsilon \leq 1$ we can consider the superellipse as a cross between an ellipse and a rectangle. Moreover, on plotting out the position of the "corner" of the superellipse (i.e. its intersection with the diagonal) as a function of ϵ we find that their relation is fairly linear, see figure 4 in which the straight line $(1, \frac{1}{\sqrt{2}}) \rightarrow (0, 1)$ is shown dotted. Therefore it seems reasonable to construct a superellipse distance measure as a linear combination of the two distances to the ellipse and rectangle which form the bases of the superellipse $\text{EOF}_9 = d_e\epsilon + d_r(1 - \epsilon)$. The weights are chosen since the curve is an ellipse at $\epsilon = 1$ and approaches a rectangle as ϵ tends to

zero. The distance to the minimum enclosing rectangle (for the first quadrant) is calculated as

$$
d_r = \begin{cases} \sqrt{(x-a)^2 + (y-b)^2} & \text{if } x > a \text{ and } y > b \\ \min(|y-b|, |x-a|) & \text{if } x < a \text{ and } y < b \\ |y-b| & \text{if } x \le a \\ |x-a| & \text{if } y \le b, \end{cases}
$$

while we use the confocal conic method to accurately estimate the distance d_e to an ellipse [14]. We can apply the same strategy to cover $1 \le \epsilon \le 2$ by combining a diamond with the ellipse. Figure 2i shows the combination of d_e and d_r. The result has excellent linearity, but shows discrepancies at the corner.

4.6 Assessment of EOFs

One approach to comparing the various EOFs is to apply the set of assessment measures developed by Rosin [13]. These enable us to quantify the linearity (L), curvature bias (C), asymmetry (A), and overall goodness (G) as well as overall goodness excluding the interior of the superellipse (G'). An EOF rates well if it has a high linearity while the remainder of the measures should be low. All measures have been applied to the $a = 400$, $b = 100$, $\epsilon = \frac{1}{2}$ superellipse used previously, and are normalised with respect to the algebraic distance (EOF$_1$). To aid interpretation the best third of the entries in each column have been highlighted in boldface and the worst third are in italics. Also, in two instances asymmetry was so severe that it was not measured properly, and is indicated by a dash in the table. The EOFs may display different behaviour close to or distant from the superellipse boundary, and so we have evaluated the measures simulating this as the displacement of data points from the true superellipse due to two levels of Gaussian noise, see tables 1 and 2. We see that for low levels of noise EOFs such as EOF$_2$ and EOF$_6$ do well while EOF$_8$ and EOF$_9$ do poorly. At greater levels of noise however, EOF$_8$ and EOF$_9$ do much better, EOF$_5$ performs quite well, and EOF$_{1-3}$ do badly.

Table 1: Normalised assessment results with low amounts of noise: $N(0, 2)$

EOF	L	C	A	G	G'
1	1.000	*1.000*	*1.000*	1.000	1.000
2	*0.994*	**0.190**	**0.419**	**0.699**	**0.746**
3	*0.923*	0.287	*1.213*	*1.077*	0.964
4	*0.999*	0.919	**0.248**	**0.981**	*1.122*
5	1.000	**0.280**	0.669	1.018	**0.935**
6	1.000	**0.283**	0.433	**0.968**	**0.913**
7	1.000	*1.006*	*1.193*	1.057	1.014
8	1.000	*1.481*	**0.147**	*2.030*	*1.499*
9	1.000	0.337	0.487	*2.709*	*2.626*

5 Experiments

In order to test out the various methods 1320 synthetic curves were generated with a variety of characteristics. Their parameters systematically ranged over $a = [150, 300]$, $b = 100$, $\epsilon = [0.1, 1.0]$, $\theta = [0, 3.0]$, $x_c = y_c = 400$. To improve the realism of the data the following processes were carried out. A binary image of each curve was formed (containing values 50 and 200), which was then blurred by a Gaussian filter ($\sigma = 2$). Gaussian noise was added ($\sigma = 20$), the

Table 2: Normalised assessment results with high amounts of noise: $N(0, 64)$

EOF	L	C	A	G	G'
1	*1.000*	*1.000*	1.000	*1.000*	*1.000*
2	*0.526*	0.109	–	*2.222*	0.166
3	*0.562*	0.108	–	*2.229*	0.247
4	1.153	*1.000*	1.430	0.329	0.307
5	**1.158**	**0.067**	**0.938**	**0.260**	**0.144**
6	1.071	0.172	**0.257**	0.565	*0.497*
7	1.145	*0.997*	*1.892*	0.262	*0.347*
8	**1.160**	**0.004**	0.518	**0.006**	**0.005**
9	**1.160**	**0.010**	1.030	**0.018**	**0.015**

image thresholded, and the boundary extracted. Each of the three methods was tested on the data; the 1D optimisation using EOF_5, and the 6D optimisation technique using EOF_1.

The absolute mean errors of the fitting techniques for each of the estimated parameters is plotted against the curve parameters in figure 5. As expected, since the quality of the data is fairly high (e.g. the complete curve is presented), the error rates are low. Many of the estimates are insensitive to the parameters of the data. Notable exceptions are:

- the error in the estimated axes length by the 1D method increases linearly with the size of the superellipse

- the error in the estimated orientation by the complete data (i.e. area/diagonal) and 1D methods decreases with the size of the superellipse

- due to the combined ellipse and rectangle fitting of the 1D method many variations are found in the parameter estimates as a function of ϵ

- for the complete data method orientation errors increase and squareness errors decrease with increasing ϵ; the estimation of ϵ by the area method (labelled complete) is more sensitive to squareness then the diagonal method (labelled complete2)

Overall, the full 6D optimisation technique performs best, although the geometric component of the 1D optimisation technique provides a slightly more accurate centre estimate. The complete data and 1D optimisation techniques are comparable since the former fares better on axis length and squareness while the latter performs better on centre and orientation estimates.

The second experiment looks at performance on noisy, complete data. As we would expect, figure 6 shows that performance degrades with increasing noise although the methods are affected by varying degrees. The severity of degradation is in the order of complete data methods, 1D optimisation, and 6D optimisation. We see that for the estimation of ϵ the diagonal based method provides improved robustness compared to the area based method. This makes it competitive with the 1D optimisation technique for estimating both axis length and squareness.

The third experiment tests the ability of the 1D and 6D optimisation techniques to cope with partial data. Since the breakdown of the methods can involve enormous inaccuracies (e.g. centre and axis values increased by a hundred orders of magnitude) we present graphs of α-trimmed mean errors ($\alpha = 0.05$). We see in figure 7 that the breakdown point is around 0.5–0.7. Although the 6D optimisation technique performs best for fairly complete data it breaks down earlier and more severely than the 1D optimisation technique which outperforms it when less than 60% of the data is available.

The final experiment again goes through a range of noise levels, and tests the different distance measures. Since we would expect all the measures to work reasonably for complete data, this time only 70% complete data sets are used. Again, some extremely bad estimates produce confusing mean errors, and so results of median errors are plotted instead. Experiments showed that optimisation using Gross and Boult's measure (EOF_4) failed. This comes about because many of the residuals are less than one and so the error function tended to be minimised by just increasing squareness. The median errors of all the remaining measures are shown in figure 8a–d. To improve readability of the graphs they have been redrawn in figure 8e–h, excluding the high error measures EOF_8, and EOF_9 and going over a smaller range. It can be seen that roughly ordering the measures in decreasing merit gives: $EOF_3/EOF_5/EOF_7$, EOF_2, EOF_6, EOF_1, EOF_8, EOF_9.
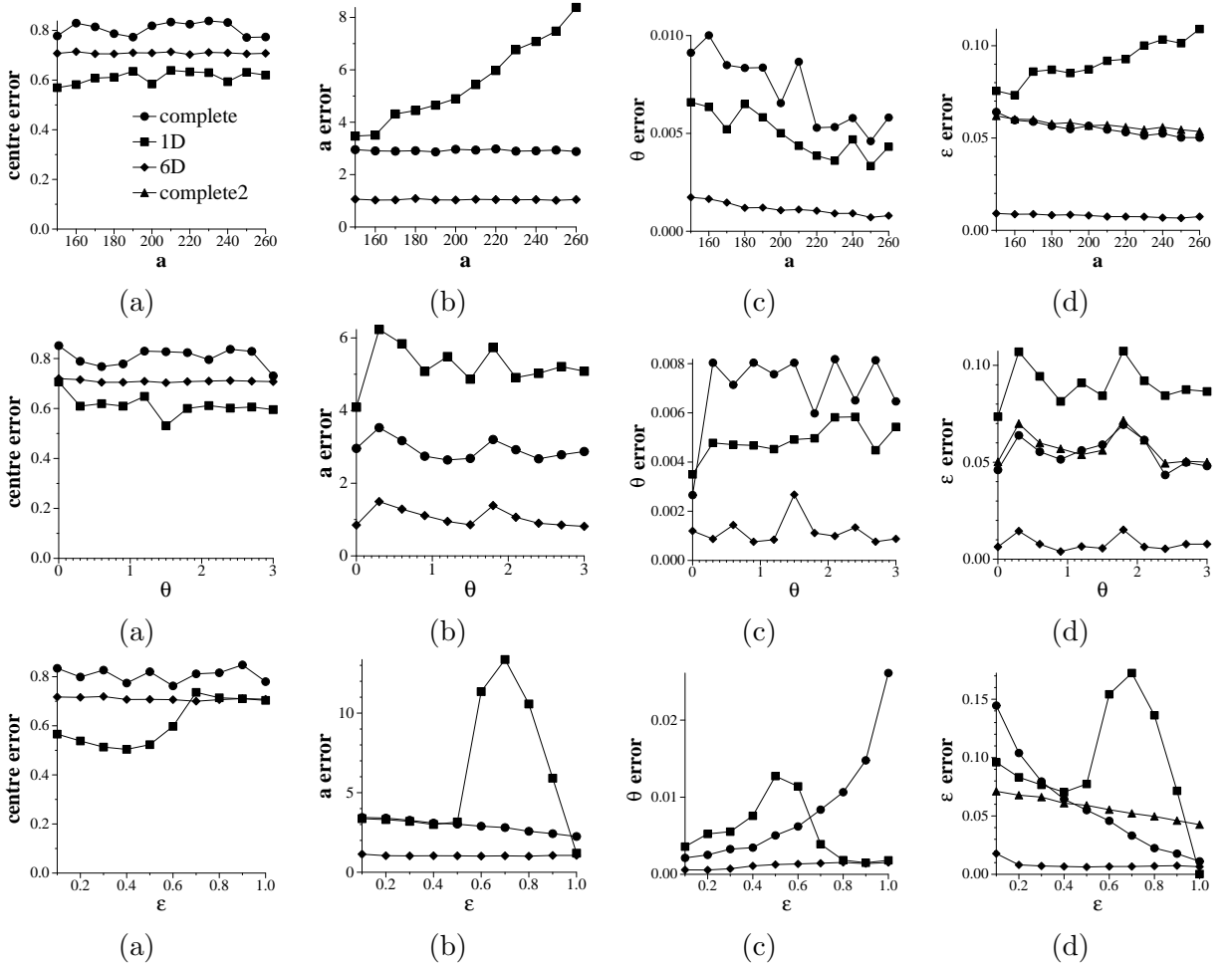

Figure 5: Errors in parameter estimates as a function of data parameters

Finally, in figure 9 we show an example of applying the fitting techniques to some real data – the computer mouse from [15]. The data shows various discrepancies, e.g. protrusions, and straight rather than curved sides. We can see that the 6D optimisation method has produced good fits while the other methods are less satisfactory. In particular they have incorrectly estimated the major axis of the outer curve. For the area and 1D methods this leads to the incorrect estimate of ϵ, but the diagonal method is relatively unaffected, and has managed to produce estimates of ϵ only a little worse than the 6D optimisation method. The quality of the fits can be seen to improve as the fitting methods progress from closed form (area), to 1D fitting, closed form (diagonal), to 6D fitting. Along with this improvement is a concurrent increase in computation (with the exception of the diagonal method which is both efficient and accurate
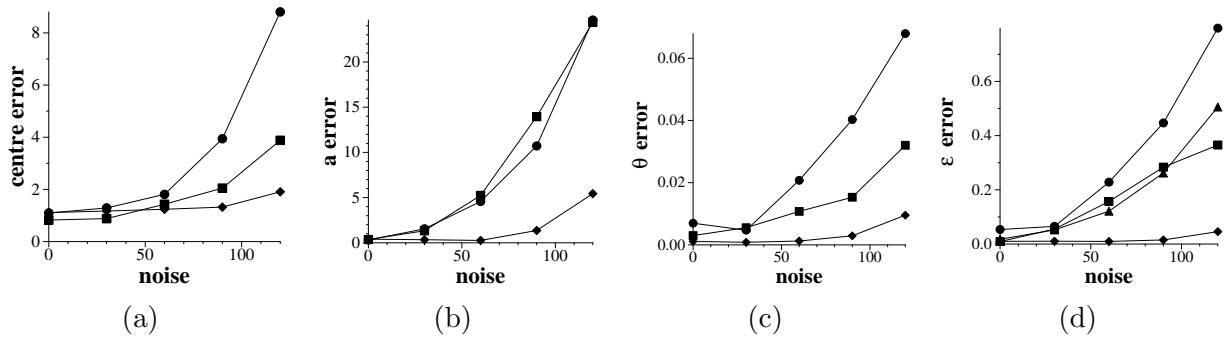
Figure 6: Errors in parameter estimates as a function of noise
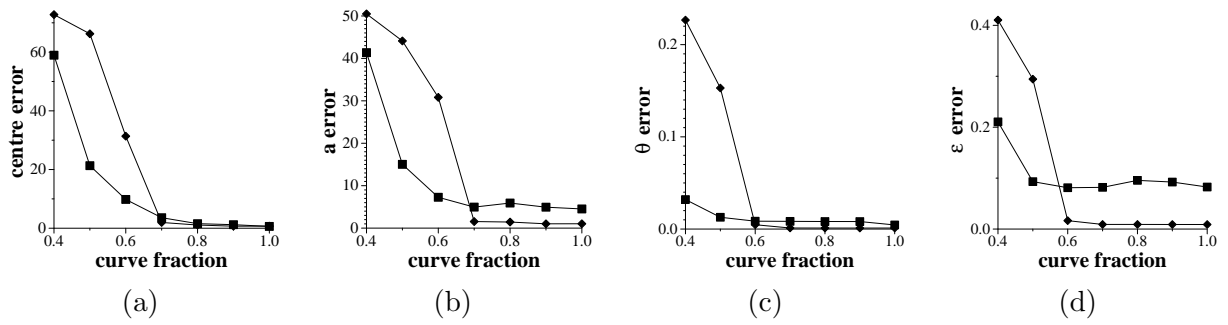

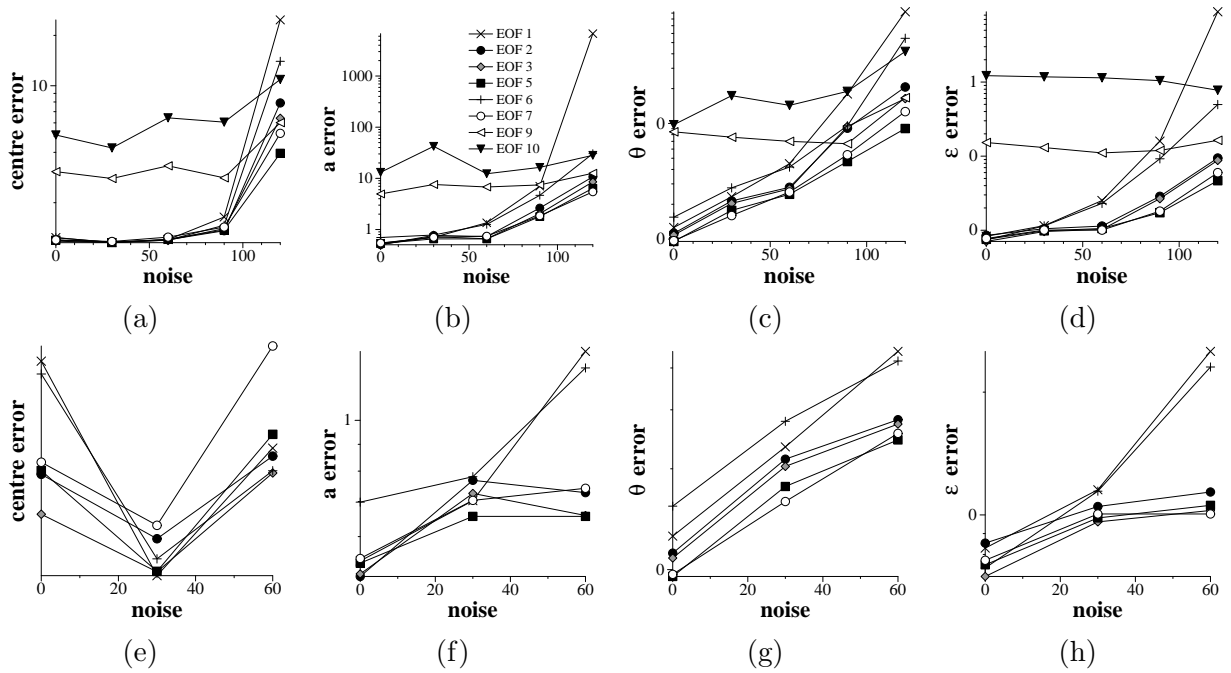Figure 7: Errors in parameter estimates as a function of completeness of curve

Figure 8: Errors in parameter estimates as a function of noise for various error measures using the 6D optimisation method

for this example); the elapsed time for the fitting algorithms is: 0.14, 2.14, 0.15, and 85.64 seconds respectively. This demonstrates the large difference in the computational loads of the algorithms.
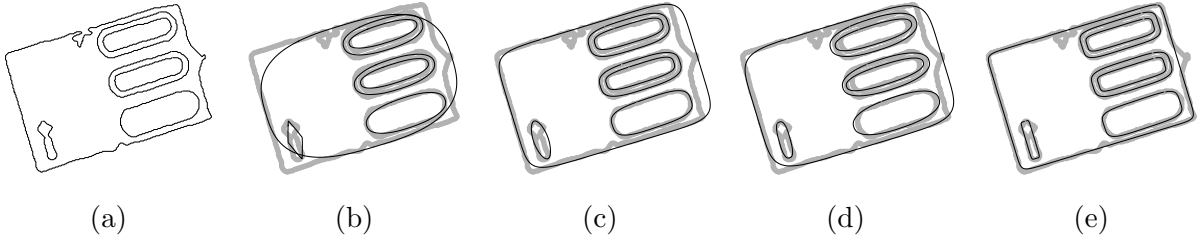

$$\text{(a)} \qquad \text{(b)} \qquad \text{(c)} \qquad \text{(d)} \qquad \text{(e)}$$

Figure 9: Fitting to real data of mouse; (a) pixel data, (b) area based method, (c) diagonal based method, (d) 1D optimisation, (e) 6D optimisation

6 Conclusions

Several new methods for superellipse fitting have been described and tested. On good data, i.e. unoccluded with low noise, all methods performed well. Since the two complete data methods performed several orders of magnitude faster than the 6D optimisation method they are obviously more appropriate in this setting; the diagonal based method appearing more robust than the area based method while requiring little additional computation time. However, when substantial amounts of noise are added the complete methods quickly degrades, and the 6D optimisation method gives the best results. Testing the 1D and 6D methods on incomplete data shows that both are little affected by small amounts of occlusion. With increasing amounts of occlusion both break down sharply, first the 6D method and then the 1D method, making the latter more suitable for incomplete data that has not much noise. For this latter case the 6D optimisation method performs best.

The results of comparing the various EOFs is more confused since the results of the tests based on fitting do not always agree with the indications provided by the assessment criteria. On some points they concur, for instance EOF_8 and EOF_9 perform relatively poorly when there is little noise while, when there is a greater amount of noise EOF_1 and EOF_6 perform worst. However, whereas the assessment criteria predict that EOF_8 and EOF_9 should perform well in the presence of substantial noise this was not borne out when the fitting was carried out. This suggests that the assessment criteria need to be further refined to ensure that they capture the qualities of EOFs that have most effect on the subsequent fitting process.

References

[1] M. Abramowitz and I.A. Stegun. *Handbook of Mathematical Functions.* U.S. Government, 1964.

[2] A.H. Barr. Global and local deformations of solid primitives. *ACM Computer Graphics*, 18(3):21–30, 1984.

[3] M. Bennamoun and B. Boashash. A structural description based vision system for automatic object recognition. *IEEE Trans. SMC B*, 27(6):893–906, 1997.

[4] A.W. Fitzgibbon, M. Pilu, and R.B. Fisher. Direct least square fitting of ellipses. *IEEE Trans. PAMI*, 21(5):476–480, 1999.

[5] A.D. Gross and T.E. Boult. Error of fit for recovering parametric solids. In *Proc. Int. Conf. Computer Vision*, pages 690–694, 1988.

[6] W. Knowlton, R.A. Beauchemin, and P.J. Quinn. *Technical Freehand Drawing and Sketching*. McGraw-Hill, 1977.

[7] R. Lee, P.C. Lu, and W.H. Tsai. Moment preserving detection of elliptical shapes in grayscale images. *Pattern Recognition*, 11:405–414, 1990.

[8] Y. Nakagawa and A. Rosenfeld. A note on polygonal and elliptical approximation of mechanical parts. *Pattern Recognition*, 11:133–142, 1979.

[9] A.P. Pentland. Automatic extraction of deformable part models. *Int. J. Computer Vision*, 4(2):107–126, 1990.

[10] M. Pilu and R.B. Fisher. Training PDMs on models: the case of deformable superellipses. *Pattern Recognition Letters*, 20(5):463–474, 1999.

[11] F.P. Preparata and M.I. Shamos. *Computational Geometry*. Springer-Verlag, 1985.

[12] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vettering. *Numerical Recipes in C*. Cambridge University Press, 1990.

[13] P.L. Rosin. Assessing error of fit functions for ellipses. *Graphical Models and Image Processing*, 58:494–502, 1996.

[14] P.L. Rosin. Ellipse fitting using orthogonal hyperbolae and Stirling's oval. *Graphical Models and Image Processing*, 60:209–213, 1998.

[15] P.L. Rosin and G.A.W. West. Curve segmentation and representation by superellipses. *Proc. IEE: Vision, Image, and Signal Processing*, 142:280–288, 1995.

[16] R. Safaee-Rad, I. Tchoukanov, B. Benhabib, and K.C. Smith. Accurate parameter estimation of quadratic curves from grey level images. *CVGIP: IU*, 54:259–274, 1991.

[17] M. Stricker. A new approach for robust ellipse fitting. In *Int. Conf. Automation, Robotics, and Computer Vision*, pages 940–945, 1994.

[18] G.T. Toussaint. Solving geometric problems with the rotating calipers. In *Proc. IEEE MELECON '83*, pages A10.02/1–4, 1983.

[19] K. Voss and H. Süße. A new one-parametric fitting method for planar objects. *IEEE Trans. PAMI*, 21(7):646–651, 1999.

[20] N. Yokoya, M. Kaneta, and K. Yamamoto. Recovery of superquadric primitives from a range image using simulated annealing. In *Int. Conf. Pattern Recognition*, pages 168–172, 1992.