Theoretical Foundations of Computer Vision

Proceedings of the Vth Workshop 1992
Buckow (Märkische Schweiz)
March 30–April 3, 1992

edited by

Reinhard Klette and Walter G. Kropatsch

Akademie Verlag

Reconstruction of Superquadrics from 2-D Contours *

Andreja Vidmar and Franc Solina
Computer Vision Laboratory
Faculty of Electrical Engineering and Computer Science
University of Ljubljana,
Tržaška 25, 61001 Ljubljana, Slovenia

Abstract

A method for recognition of superquadrics from image contours is presented. The image contour considered here is a 2-D occluding contour assuming parallel projection. The method consists of two steps. In the first step the position of superquadric is determined. In the second step of the method the values of shape and orientation parameters are computed. Results of model recovery are given at the end and directions for further research are discussed.

Key words: superquadrics, occluding contours, model recovery, least-squares fit

1 Introduction

Recovery of scene descriptions that support intelligent action is one of the main goals of computer vision. Image information, especially in the form of position and shape of individual parts and objects, directly supports reasoning and action. This level of image description is hence a declared goal of many vision systems. Part-level shape models and their recovery play an important part in reaching this goal. The most wide-spread part-level models in computer vision have been so far generalyzed cylinders. But due to their very general form a robust and reliable recovery of generalyzed cylinders from any image cue is difficult. A more restricted form of volumetric models called superquadrics have been introduced to computer vision by Pentland (1986). Instead of the common way of building models of larger granularity by combining low level image models such as edges, surface patches and surface normals the simpler parametrization of superquadrics allows direct recovery of their parameters from low

*This research was supported by the Ministry of Research and Technology of the Republic of Slovenia under Project P2–1122

level image information. We have introduced a reliable method for recovery of superquadrics enhaced with a set of global deformations from range data, see Solina and Bajcsy (1990). This method has already been used in many computer vision applications, see Solina and Bajcsy (1989), Allen and Roberts (1989), and Choi et al. (1990). However, dense range data that can be obtained using a range scanner is not always available or sensible to use. So, other image cues must be used instead. We believe that the inherent parametrization of superquadric models provide enough constraints to recover possible 3-D interpretations of 2-D shapes. In other words, the internal constraints of superquadric models are the additional information that is needed to invert the projection of a 3-D shape onto a 2-D image. In this paper we present an attempt to recover superquadric models only from 2-D occluding contours of single objects.

Superquadric models are like lumps of clay that can be formed into different shapes that are allowed by the internal parametrization. Basic geometric solids such as spheres, cubes, parallelopipeds and ellipsoids and shapes inbetween are all included. Their granularity supports also some psychological theories of human perception (Biederman 1985) that assume an existence of a set of primitive blocks that can describe the wealth of differently shaped objects by combining them like phonemes in a language.

This paper is organized as follows. First, superquadric models are defined and assumptions used throughout this paper are given. Then a two step model recovery is described: position of a superquadric is determined first, then the actual shape and orientation of the superquadric is reconstructed using iterative least-squares fitting. The main issues of the reconstruction method are definition of an appropriate fitting function and finding of the initial estimates. At the end of the paper a few results are given and the guidelines for further work are discussed.

2 Definitions and assumptions

In this section basic definitions and terms which are used in this chapter are presented. Additionaly, we introduce assumptions used in reconstruction.

2.1 Superquadrics

Superquadrics are a family of parametric models. They are used as primitives for shape representation in computer graphics (Barr 1981, 1984) and in computer vision (Pentland 1986, Solina and Bajcsy 1990). In our paper we deal with a subset of superquadrics – superelliposids (see Sect. 2.3 for explanation).

A superellipsoid is defined with a 3-D vector:

$$\vec{s}(\eta,\omega) = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} = \begin{bmatrix} a_1 \cos^{\varepsilon_1}\eta \cos^{\varepsilon_2}\omega \\ a_2 \cos^{\varepsilon_1}\eta \sin^{\varepsilon_2}\omega \\ a_3 \sin^{\varepsilon_1}\eta \end{bmatrix}, \qquad \begin{array}{ccccc} -\frac{\pi}{2} & \leq & \eta & \leq & \frac{\pi}{2} \\ -\pi & \leq & \omega & < & \pi \end{array}$$

The *surface vector* $\vec{s}$ originates from the center of coordinate system. The parameter ω is the angle between x axis and the projection of vector $\vec{s}$ onto the x-y plane, while the parameter η is the angle between vector $\vec{s}$ and its projection onto the x-y plane. Parameters a_1, a_2, and a_3 determine the superellipsoid size along x, y, and z coordinate axes, respectively. The parameters ε_1 and ε_2 define the superellipsoid shape in direction of z axis and x-y plane, respectively.

The superellipsoid *normal vector* is a cross product of two surface tangent vectors: $\vec{n} = \frac{\partial \vec{s}}{\partial \eta} \times \frac{\partial \vec{s}}{\partial \omega}$, where $\times$ denotes cross product.

$$\vec{n}(\eta, \omega) = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \begin{bmatrix} \frac{1}{a_1} \cos^{2-\varepsilon_1}\eta \cos^{2-\varepsilon_2}\omega \\ \frac{1}{a_2} \cos^{2-\varepsilon_1}\eta \sin^{2-\varepsilon_2}\omega \\ \frac{1}{a_3} \sin^{2-\varepsilon_1}\eta \end{bmatrix}$$

The earlier definitions are expressed with the coordinates of the local coordinate system. If we want to determine the position and the orientation of a superellipsoid in space, we have to express the superellipsoid definitions in a world coordinate system (for details see Vidmar (1991):

$$\begin{aligned} \vec{s}\,' &= R\vec{s} + \vec{p} \\ \vec{n}\,' &= R\vec{n} \end{aligned}$$

where R is a *ZYZ Euler rotational matrix* and $\vec{p}$ is a *translation vector.*

To summarize, every superellipsoid in general position and orientation is completely determined with shape parameters, its orientation, and position:

$$\mathcal{P} = \langle a_1, a_2, a_3, \varepsilon_1, \varepsilon_2, \varphi, \vartheta, \psi, p_x, p_y, p_z \rangle$$

2.2 Image contour

The *image contour* considered in this work is a 2-D curve which occludes the projection of a 3-D object. Parallel or perspective projection can be used.

If the object is convex and parallel projection is used, the surface points which lie on the image contour can be easily determined. The scalar product of the normal vector $\vec{n}$ of the surface point and the viewing direction vector $\vec{v}$ must be equal to zero: $\vec{n} \cdot \vec{v} = 0$.

If we use perspective projection instead of parallel, the upper condition is somewhat more complex: $\vec{n} \cdot (\vec{s} - \vec{w}) = 0$. The vector $\vec{n}$ is the normal vector, $\vec{s}$ is the surface vector, and $\vec{w}$ is the position vector of the view point. The difference $(\vec{s} - \vec{w})$ represents the vector from the view point to the surface point.

2.3 Assumptions

As we mentioned before, we use a subset of superquadrics – superellipsoids. Also, we use superellipsoids without any deformations. We believe, that once we have a method for reconstruction of superellipsoids, the method can be extended to include deformed superquadrics also. To simplify the condition for determining surface points which maps to the image contour, we use convex superellipsoids, that is $\varepsilon_1, \varepsilon_2 \leq 2$. It is also true that superellipsoids generated with ε_1 or $\varepsilon_2 \gg 2$ perceptually consist of several parts.

Two projections can be used – parallel or perspective. If we want to use perspective projection, the accurate position of camera (the viewpoint) must be known. To avoid that, we have decided to assume parallel projection. We use *weak parallel projection* – parallel projection with constant view direction vector $\vec{v} = [0, 0, 1]^T$. This does not restrict our problem since any orientation of a superellipsoid can be obtained with the rotational matrix R.

We also presume that there is an image contour of only one superellipsoid in the given picture. In this way, we can focus only on the problem of recognition and avoid the segmentation problem which in general probably has to be solved simultaneously with shape recognition, see Bajcsy et al. (1990).

3 Determining the position in space

If we want to determine the position of a superellipsoid in space we have to find the value of the translation vector $\vec{p} = [p_x, p_y, p_z]^T$.

The coordinate p_z can not be found when parallel projection is used and z axis is the viewing direction. Its determination would be possible, if the distances from the view point to the given contour points were known. This means that the value of p_z is optional. However, when a particular superellipsoid is recognized, the value of p_z coordinate can be determined by measuring the distance at only one or in worst case at few contour points.

What about coordinates p_x and p_y? If we prove that an image contour is symmetrical about its center, that is the projection of the center of the superquadric onto the image plane, the center $T(t_x, t_y)$ of the contour can be computed with the following algorithm. So, we can determine values of p_x and p_y, for it holds: $t_x = p_x$ and $t_y = p_y$.

Algorithm 1 (Center Determination)

1. Construct a circumcircle of the 2-D curve which is symmetrical about its center.
2. There are pairs of points with the greatest mutual distance lying on the circumference of the circumcircle.
3. Connect points of one pair.
4. The center of the curve is at the midpoint of the connection line.

The proof, that the image contour of a superellipsoid is symmetrical about its center, can be easily done with the help of two propositions. The first is that every superellipsoid is symmetrical about its center. The second proposition states that if

$$\begin{aligned} g_1(s'_x, s'_y\,;\,\omega, \vec{P}) &= r_{11}s_x + r_{12}s_y + r_{13}s_z - s'_x \\ g_2(s'_x, s'_y\,;\,\omega, \vec{P}) &= r_{21}s_x + r_{22}s_y + r_{23}s_z - s'_y \end{aligned}$$

where $\vec{P}$ is the vector of the values of the superellipsoid parameters $\mathcal{P}$. We try to obtaine the following minimum:

$$\min_{\vec{\omega},\vec{P}} \sum_{i=1}^{n} \left(g_1^2(x_i, y_i\,;\,\omega_i, \vec{P}) + g_2^2(x_i, y_i\,;\,\omega_i, \vec{P}) \right) \tag{5}$$

where x_i and y_i are coordinates of the contour point T_i, ω_i is the angle where point T_i is computed, $\vec{\omega}$ is the vector of angles ω_i, and n is the number of contour points.

4.2 Algorithm for LSM

Because the obtained fitting function is nonlinear (eq. 5), we have chosen the Levenberg-Marquardt method (Press et al. 1986) for the minimization algorithm, a common approach for such problems. We have slightly modified the method.

The original method is appropriate when the number of parameters is much smaller than the number of given points (the multiplication of Jacobian matrix with its transpose decreases the order of the matrix of linear system). Because, in our case, in the set of parameters also includes angles ω (the angles, where contour points are computed), the method is not that effective (multiplication with transpose matrix J^T decreases the order of the system matrix at most for a half). Because of that, we have modified the system of linear equations (for details see Vidmar (1991)). To solve the system of linear equations we use the singular value decomposition (SVD) method (Press et al. 1986). The reason is, that the method gives resonable results even in cases like ours—where the system is predefined, the order of the system matrix is high, and the matrix can be numerically close to a singular matrix. In addition, we change the strategy of modifying the factor μ in the Levenberg-Marquardt algorithm.

Algorithm 2 (Modified Levenberg-Marquardt)

1. calculate $G(\vec{u}_0)$
2. set $\mu = 0.01$, $\mu_0 = 100$, $\nu_1 = 2$, $\nu_2 = 10$
3. set $\vec{u} = \vec{u}_0$
4. repeat
5.   perform matrix decomposition (with SVD method) $J = UWV^T$
6.   solve system of equations $U\left(W + \mu W^{-1}\right) V^T\,\vec{d} = -\vec{b}$
7.   calculate $G(\vec{u} + \vec{d})$
8.   if $G(\vec{u} + \vec{d}) \geq G(\vec{u})$
9.   if $\mu \leq \mu_0$ set $\mu = \mu\nu_1$

        else set $\mu = \mu\nu_2$
    else
        set $\vec{u} = \vec{u} + \vec{d}$
        if $\mu < \mu_0$ set $\mu = \frac{\mu}{\nu_1}$
        else set $\mu = \frac{\mu}{\nu_2}$
until $|G(\vec{u} + \vec{d}) - G(\vec{u})| \leq stat$

Let us explain the meaning of the symbols. The vector of parameters $\vec{u}$ is as follows:

$$\vec{u} = [a_1, a_2, a_3, \varepsilon_1, \varepsilon_2, \varphi, \vartheta, \psi, \omega_1, \omega_2, \ldots, \omega_n]^T$$

The number of parameters m is $m = k+n$, where $k = 8$ (the number of superellipsoid parameters $\mathcal{P}$) and n is the number of the contour points. G represents the sum of square values of the fitting functions:

$$G(\vec{u}) = \sum_{i=1}^{n} \left(g_1^2(x_i, y_i\,; \vec{u}) + g_2^2(x_i, y_i\,; \vec{u}) \right) \tag{6}$$

The elements of the Jacobian matrix J (matrix of partial derivatives of the fitting functions) are defined as follows:

$$(J_k)_{ij} = \begin{cases} \left.\frac{\partial g_1(x_i, y_i\,; \vec{u})}{\partial u_j}\right|_{\vec{u}=\vec{u}_k}, & \begin{array}{l} i = 1, \ldots, n \\ j = 1, \ldots, k+n \end{array} \\ \\ \left.\frac{\partial g_2(x_i, y_i\,; \vec{u})}{\partial u_j}\right|_{\vec{u}=\vec{u}_k}, & \begin{array}{l} i = n+1, \ldots, 2n \\ j = 1, \ldots, k+n \end{array} \end{cases}$$

The vector $\vec{b}$ is the vector of values of the fitting function at a particular point:

$$\left(\vec{b}_k\right)_i = \begin{cases} g_1(x_i, y_i\,; \vec{u}_k), & i = 1, \ldots, n \\ \\ g_2(x_i, y_i\,; \vec{u}_k), & i = n+1, \ldots, 2n \end{cases}$$

5 Initial estimates

The fitting functions have several minima. Which minimum will be found by the LSM depends mostly on the initial estimate for parameters (superellipsoid parameters $\mathcal{P}$ and angles ω). To find such a minimum that the contour of the resulting model will closely fit the input contour, we select such initial estimates for the superellipsoid parameters that the shape of initial model resembles the shape of the input contour. To explain how we get these estimates we have to introduce the notion of a *transformational graph*.

5.1 Transformational graphs

We introduce two tranformational graphs. The first graph $r(\tau)$ shows the distance of a contour point $T(x, y)$ from the center point as a function of the angle τ (see Fig. 1 (a) and (b)). The transformational graph $r(\tau)$ can be simply computed from the contour:

$$\begin{aligned} r &= \sqrt{x^2 + y^2} \\ \tau &= \arctan \tfrac{y}{x} \end{aligned}$$

The second transformational graph is the function $\omega(\tau)$. This function shows the dependence of angle ω where the contour point $T(x, y)$ was computed from angle τ. An example is shown in Fig. 1 (c). To compute the graph we need to know not only the contour but also the values of angle ω where the individual contour points were computed.

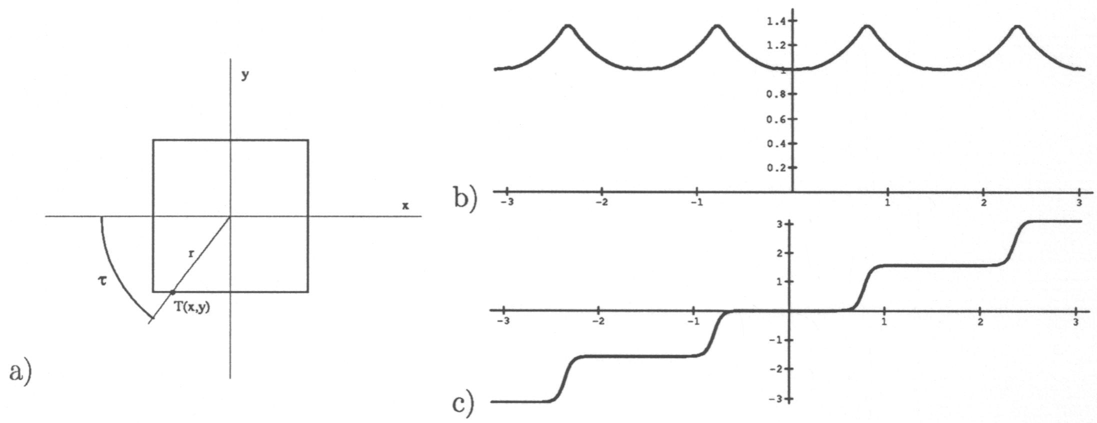


Figure 1: (a) The distance r of contour point $T(x, y)$ from the coordinate center is a function of angle τ ($\tau \in [-\pi, \pi]$). (b) Transformational graph $r(\tau)$ for the contour. (c) Transformational graph $\omega(\tau)$ for the contour.

5.2 Estimates for superellipsoid parameters $\mathcal{P}$

To determine a good initial estimate of parameters $\mathcal{P}$ we made a classification of all possible superellipsoid contours. We limited the values of individual parameters (for parameters ε_1 and ε_2 we selected values 0.1, 1, and 1.9; for size parameters we chose the examples when individual parameters are equal or different) and took representative examples. We examined contours of chosen superellipsoids in different orientations. In fact, we tried to identify the nodes of the corresponding *aspect graph*, see Koendering and van Doorn (1979). Fig. 2 shows a set of results.

This analysis clearly shows that different superellipsoids can produce identical contours. We grouped the contours according to similarity into 12 groups. The similarity

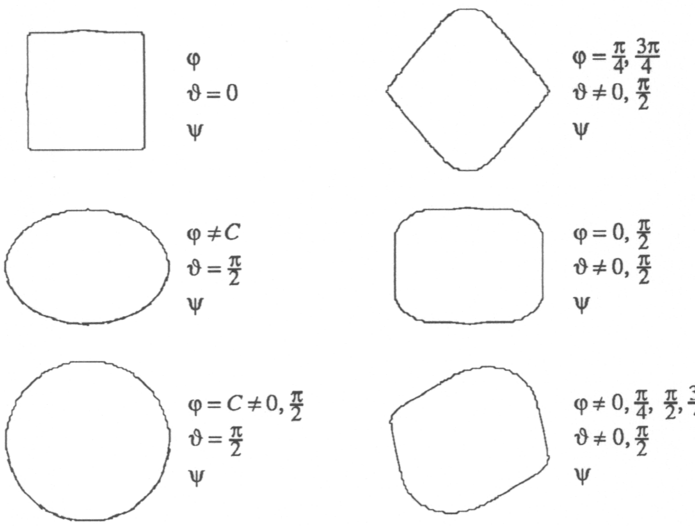


Figure 2: Possible contours for the superellipsoid with parameters: $\varepsilon_1 = 1$, $\varepsilon_2 = 0.1$, $a_1 = a_2 \neq a_3$. If no value is given for an angle, the value is arbitrary. C is a constant that depends on the ratio of parameters a_1, a_2, and a_3.

criterion is based on transformational graph $r(\tau)$. A set of descriptions for superellipsoids, which give the characteristic contour, belongs to each group.

To get initial estimates, the given contour is first classified into one of the contour groups using the shape of the transformational graph $r(\tau)$. From the set of superellipsoid group descriptions we get initial estimates for all possible superellipsoids that could corespond to the given contour. The estimates that correspond to a group description are derived in the following way:

- For ε_1 in ε_2 take the values from the group description.
- Set the values for a_1, a_2, and a_3 to 1.
- If angle φ is not arbitrary then take the given value. Else take two estimates: 0 and $\frac{\pi}{2}$.
- For angle ϑ the same rule is used as for φ. However, we must also consider the restrictions given in Sect. 6 so that if the estimate for ϑ is equal 0 or $\frac{\pi}{2}$ we have to add a small number to the estimate.
- Set angle ψ to 0.

5.3 Estimates for ω

An analysis of transformational graphs $\omega(\tau)$ derived for different superellipsoids gives the following results:

1. The graph of function $\omega(\tau)$ is a curve that has symmetrical "steps" along lines with direction $k = \pm 1$.

2. The shift of the function along the horizontal axis depends on the combination of values of angles φ, ϑ and ψ. If $\varphi = \vartheta = \psi = 0$ the shift is 0.

3. If $0 \leq \vartheta < \frac{\pi}{2}$ the curve ascends. If $\frac{\pi}{2} \leq \vartheta < \pi$ the curve descends.

The initial estimate for ω for point $T(x, y)$ is then:

$$\omega = \begin{cases} \arctan \frac{y}{x}, & 0 \leq \vartheta < \frac{\pi}{2} \\ -\arctan \frac{y}{x}, & \frac{\pi}{2} \leq \vartheta < \pi \end{cases}$$

In both cases the estimates for angle ω form a line with direction $k = +1$ or $k = -1$.

6 Parameter constraints

All parameter constraints except those for a_1, a_2, and a_3 are used to prevent numerical instability. We limited all size parameters to values between 0 and 1: $0 < a_1, a_2, a_3 \leq 1$. The constraint on the size was taken that it is easier to estimate the initial values for a_1, a_2, and a_3. Secondly, the value of the fitting functions is then also constrained to $[-2, 2]$. Hence the stoping condition for iteration in Algorithm 2 is equal for all sizes of superellipsoids. To enforce this constraint the given contour must be normalized.

As stated in Sect. 2.3 we deal only with convex superellipsoids ($0 \leq \varepsilon_1, \varepsilon_2 \leq 2$). We have made the constraint even stricter: $0.1 \leq \varepsilon_1, \varepsilon_2 \leq 1.9$, so that the fitting function would be everywhere well behaved. There are some points where the value of the fitting function is defined only as its limit and the function is numerically unstable in their vicinity. The constraint is not restrictive since the perceptual difference between the shapes of superellipsoids obtained at $\varepsilon_1, \varepsilon_2 = 0\,(2)$ and shapes obtained at $\varepsilon_1, \varepsilon_2 = 0.1\,(1.9)$ is not substantial.

The constraints for angles ω and η are as follows: $\omega \neq 0, \pm\frac{\pi}{2}, -\pi$, $\eta \neq 0, \pm\frac{\pi}{2}$. At these values some derivatives of fitting functions have value defined only in the limit and they are numerically unstable in their viccinity. To prevent possible numerical problems, the data points with angles η in ω close to those values should be removed from the contour. Experiments show that these are the points corresponding to the minima of the transformational graph $r(\tau)$.

Angle ϑ is constrained by $\vartheta \neq 0, \frac{\pi}{2}$. This constraint is due to the way angle η is computed, constraints for η, and its derivatatives. When $\vartheta = 0$ then η is 0. When $\vartheta = \frac{\pi}{2}$ then $\eta = \pm\frac{\pi}{2}$. Functions are numerically unstable at these values of η.

7 Implementation and results

In this section some results of model recovery on occluding contours are given. The input contours are synthetic and generated with a program. The program generates contour points of a superellipsoid. The distance between points can be determined. Those points which don't comply with constraints for angles η and ω (see Sect. 6) are eliminated. It is clear from Sect. 3 that superellipsoid position can be easily computed from the complete occluding contour. So, we have implemented only the method for determining parameters of shape and orientation (Algorithm 2). The initial estimates were obtained with the procedure described in Sect. 5.

7.1 Results

By having different sets of initial estimates for the same input contour we wanted to derive all possible interpretations for it. However, when experimenting we have sometimes obtained the same solution for different initial estimates. On the other hand, we have not succeeded to determine parameters for some possible superellipsoid interpretations regardless of the choice of initial estimates.

Due to the constraints on ω and η (see Sect. 6) we could not reconstruct contours which are entirely generated at $\eta = 0$, $\omega = \pm\frac{\pi}{2}$, or $\omega = 0, -\pi$. Those orientations correspond to visual singularities (nodes in the aspect graph) and would have to be handled individually anyway.

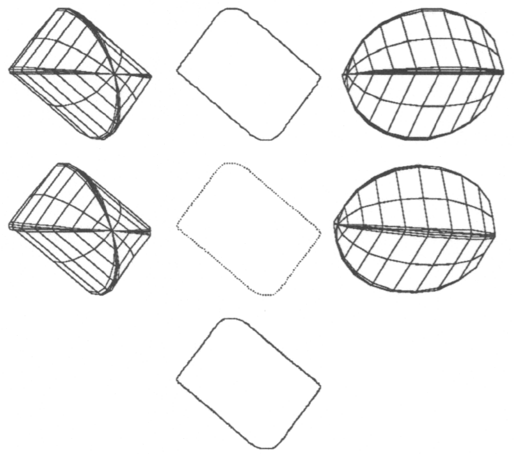
The selection of contour points also influences the results. The method converges faster if those input contour points are selected that correspond to the maxima of the transformational graph $r(\tau)$ of the given contour. This means that the method is better if the input contour is dense at the corners and has few points on the straight parts. This is not surprising since psychological experiments have demonstrated that for human perception the information about corners and joints is more salient than the connecting straight edges, see Biederman (1985). Such selection of points is also not in conflict with the constraints on angles ω and η (see Sect. 6).

In the following figures (Fig. 3–4) some results are shown. In each figure the following information is given: (a) At the top is a table with superellipsoid parameter values. The left column gives the actual values for generating the input contour. The middle column shows the initial estimates and in the right column the results of model recovery. (b) The figures in the top row show the superellipsoid for generating the input contour (left the superellipsoid, middle the corresponding contour, right the same model from a view point perpendicular to the original). (c) The figures in the middle row show the results of the recognition. (d) For comparison the bottom row shows the overlapping input and resulting contours.

8 Conclusions and further research

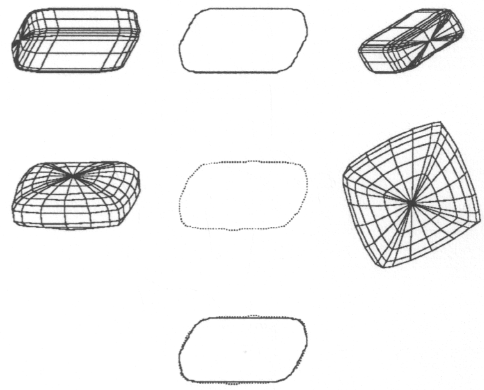
A method for model recovery with which all possible interpretations of a given contour should be derived was presented. The recognition process consists of two steps. First, the symmetry of superellipsoid on its center is used to determine its position. Then parameters of shape and orientation are obtained with least square minimization method.

a_1	0.5	1	0.70
a_2	0.7	1	0.51
a_3	1	1	1.00
ε_1	1	1	0.98
ε_2	0.1	0.1	0.12
φ	0.9	0	−0.60
ϑ	0.4	0.01	0.37
ψ	0	0	−0.07



a)

a_1	0.2	1	1.00
a_2	0.5	1	0.88
a_3	0.7	1	0.26
ε_1	0.3	0.1	0.70
ε_2	0.3	0.1	1.63
φ	1	0	-0.50
ϑ	1.9	1.58	-1.94
ψ	0	0	1.71



b)

Figure 3: (a) In this case a very close match between the original and the recovered model was obtained. (b) We recovered a model with a contour that closely matches the input contour, however, the actual model is quite different.

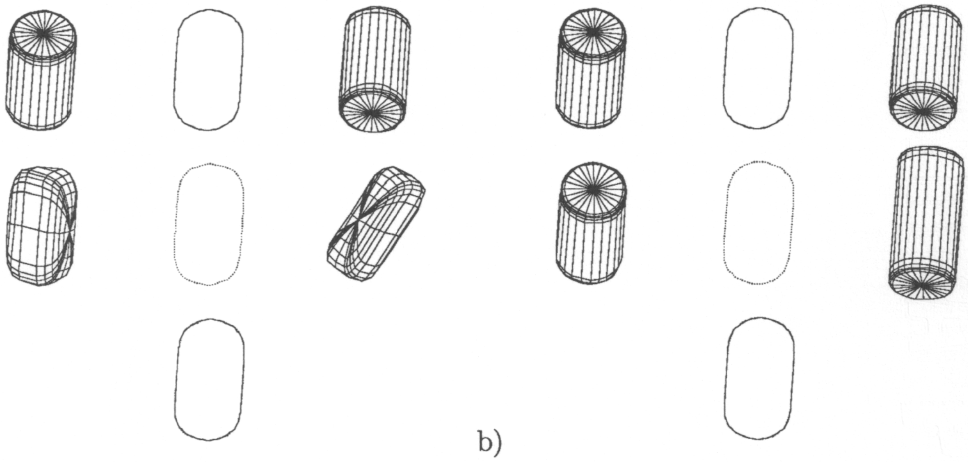
A fitting function based on surface vector was constructed. For the LSM a slightly modified Levenberg-Marquardt method was used. A procedure for establishing initial estimates based on aspect graphs was given. With the analysis of these graphs typical contour shapes and data about corresponding superellipsoid parameters were obtained. The second part of the method was implemented and results of this implementation were presented.

As shown, the initial estimates have a great impact on the method's success. To improve the initial estimates for superellipsoid parameters, an algorithm for generation of aspect graphs should be constructed. Then the analysis of possible contour shapes can be done more precisely. We did not expect that estimates for angles ω will have such effect on results of recognition, so we didn't pay much attention to them. An analysis of how transformational graphs $\omega(\tau)$ depend on orientation and shape of superellipsoid should be done.

For a given contour the number of possible superellipsoid interpretations can be very large or even infinite. It would be sensible to somehow limit the possibilities. One possible way of limiting the number of different interpretations would be to give

a_1	0.4	1	0.25
a_2	0.4	1	0.72
a_3	0.7	1	0.41
ε_1	0.1	0.1	0.63
ε_2	1	1	0.34
φ	0	0	0.35
ϑ	0.7	0.01	1.04
ψ	1.5	0	-0.26

a_1	0.4	1	0.39
a_2	0.4	1	0.40
a_3	0.7	1	0.87
ε_1	0.1	0.1	0.10
ε_2	1	1	0.93
φ	0	0	-3.34
ϑ	0.7	0.01	-0.50
ψ	1.5	0	4.64



a) b)

Figure 4: (a) Although the initial estimates for ε_1 and ε_2 favoured a cylindrical model, the recovered model is not a cylinder. (b) The input and initial estimates for parameters $\mathcal{P}$ of the superellipsoid are the same as in example (a). But for estimates of ω we used their actual values and succeded in recovering a cylinder. This shows that improving the estimates of ω a perceptually better solution could be achieved.

different perceptual weights to different interpretations. To a human observer some interpretations are obviously more "natural" than others (i.e. Fig. 4). Another way of limiting the number of possible interpretations is to get additional information about the object. Experiments have shown that by adding just very few range points the model recovery is much better. The use of shading information or two contours from two different view points should also be explored.

So far an interpretation of a single isolated contour was studied. In a scene one would typically have several such contours. After segmentation not all contours would necessarily be complete. Hence, we would have to interpret incomplete contours. Another possible extension is to include deformations of superquadric models as in Solina and Bajcsy (1990).

Bibliography

Allen, P. K., and K. S. Roberts. 1989. Haptic object recognition using a multi-fingered dextrous hand. *Proceedings of IEEE Conference on Robotics and Automation.*

Bajcsy, R., F. Solina and A. Gupta. 1990. Segmentation versus object representation – are they separable?, in R. C. Jain and A. K. Jain (Eds.), *Analysis and Interpretation of Range Images.* New York: Springer.

Barr, A. H. 1981. Superquadrics and angle-preserving transformations. *IEEE Computer Graphics and Aplications* 1, 11–23.

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

Biederman, I. 1985. Human image understanding: Recent research and a theory. *Computer Vision, Graphics, and Image Processing.* 32 (1), 29–73.

Choi, T., H. Delingette, M. DeLuise, Y. Hsin, M. Hebert and K. Ikeuchi. 1990. A perception and manipulation system for collecting rock samples. *Proceedings NASA Symposium on Space Operations, Applications and Research,* Albuquerque, New Mexico.

Kriegman, D. J. and J. Ponce, 1990. On Recognizing and Positioning Curved 3-D Objects from Image Contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12 (12), 1127–1137.

Koendering, J. J. and A. J. van Doorn. 1979. The internal representation of solid shape with respect to vision. *Biological Cybernetics* 32, 211–216.

Pentland, A. P. 1986. Perceptual Organization and the Representation of Natural Form. *Artificial Intelligence* 28 (3), 293–331.

Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. 1986. *Numerical recipes.* Cambridge: Cambridge University Press, 277–282 and 523–528.

Solina F. and R. Bajcsy. 1989. Recovery of mail piece shape from range images using 3-D deformable models. *International Journal of Research & Engineering, Postal Applications.* Inaugural Issue, 125–131.

Solina, F. and R. Bajcsy. 1990. Recovery of parametric models from range images: The case for superquadrics with global deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-12(2), 131–147.

Vidmar, A. 1991. Recognition of Superellipsoids from Image Contours, Technical report LRV-91-9, Computer Vision Laboratory, Faculty of Electrical Engineering and Computer Science, University of Ljubljana, Ljubljana, Slovenia (in Slovenian).