

Modelling Objects Having Quadric Surfaces Incorporating Geometric Constraints

Naoufel Werghi, Robert Fisher, Craig Robertson, and Anthony Ashbrook

Department of Artificial Intelligence,
University of Edinburgh
5 Forrest Hill, EH1 2QL Edinburgh, UK
email:{naoufelw, rbf, craigr, anthonya}@dai.ed.ac.uk

Abstract. This paper deals with the constrained shape reconstruction of objects having quadric patches. The incorporation of geometric constraints in object reconstruction was used first by Porrill [10]. His approach combined the Kalman filter equations with linearized constraint equations. This technique was improved by De Geeter et al [5] to reduce the effects of linearization error. The nature and the specificity of this technique make it limited in scope and application.

In their approach for 3-D object pose estimation, Bolle et al [2] constrained some quadrics to have a certain shape (circular cylinder and sphere) by using a specific representation for these particular surfaces.

Our work uses a new approach to global shape improvement based on feature coincidence, position and shape constraints. The key idea is to incorporate user specific geometric constraints into the reconstruction process. The constraints are designed to fix some feature relationships (such as parallel surface separations, or cylindrical surface axis relationships) and then use least squares fitting to fix the remaining parameters. An optimization procedure is used to solve the reconstruction problem.

In this paper, constraints for planar and general quadric surface classes are given. Results with quadric surfaces show much improvement in shape reconstruction for both constrained and unconstrained relationships. The proposed approach avoids the drawbacks of linearization and allows a larger category of geometric constraints. To our knowledge this work is the first to give such a large framework for the integration of geometric relationships in object modelling.

The technique is expected to have a great impact in reverse engineering applications and manufactured object modelling where the majority of parts are designed with intended feature relationships.

1 Introduction

There has been a recent flurry of effort on reconstructing 3D geometric models of objects from single [3, 6, 8] or multiple [2, 4, 12, 11, 13] range images, in part motivated by improved range sensors, and in part by demand for geometric models in the CAD and Virtual Reality (VR) application areas. However, an important aspect which has not been fully investigated is the exploitation of the

geometric constraints defining the spatial or topological relationships between object features.

The work presented in this paper investigates reverse engineering, namely the combination of manufacturing knowledge of standard object shapes with the surface position information provided by range sensors.

The first motivation behind this work is that models needed by industry are generally designed with intended feature relationships so this aspect should be exploited rather than ignored. The consideration of these relationships is actually necessary because some attributes of the object would have no sense if the object modelling scheme did not take into account these constraints. For example, take the case when we want to estimate the distance between two parallel planes: if the plane fitting results gave two planes which are not parallel, then the distance measured between them would have no significance.

The second motivation is to see whether exploiting the available known relationships would be useful for reducing the effects of registration errors and mis-calibration. Thus improving the accuracy of estimated part features' parameters and consequently the quality of the modelling or the object localization.

In previous work [14] we have shown that this is quite possible for planar objects. A general incremental framework was presented whereby geometric relationships can be added and integrated in the model reconstruction process. The objects treated were polyhedral and the data was taken from single views only. An overview of the technique is given in Section 3.

In this paper we study the case of parts having quadric surfaces. Two types of quadric are treated here, cylinders and cones. Both single view data and registered multiple view data have been used.

Section.2 discuss the related work and the originality of our contribution. In Section.3, we summarize the technique. More details can be found in [14]. Section.4 gives some mathematical preliminaries about quadrics in general and cylinders and cones in particular. Section.5 demonstrates the process on several test objects.

2 Related work

The main problem encountered in the incorporation of geometric relationships in object modelling is how to integrate these constraints in the shape fitting process. The problem is particularly crucial in the case of geometric constraints many of which are non-linear. In his pioneering work, Porrill [10] suggested a linearization of the nonlinear constraints and their combination with a Kalman filter applied to wire frame model construction. Porrill's method takes advantage of the recursive linear estimation of the KF, but it guarantees satisfaction of the constraints only to linearized first order. Additional iterations are needed at each step if more accuracy is required. This last condition has been taken into account in the work of De Geeter et al [5] by defining a "Smoothly Constrained Kalman Filter". The key idea of their approach is to replace a nonlinear constraint by a set of linear constraints applied iteratively and updated by new measurements in

order to reduce the linearization error. However, the characteristics of Kalman filtering makes these methods essentially adapted for iteratively acquired data and many data samples. Moreover, there was no mechanism for determining how successfully the constraints have been satisfied. Besides, only lines and planes were considered in both of the above works.

The constraints considered by Bolle et al [2] in their approach to 3D object position covers only the shape of the surfaces. They chose a specific representation for the treated features: plane, cylinder and sphere.

Compared to Porrill's and De Geeter's work, our approach avoids the drawbacks of linearization, since the constraints are completely implemented. Besides our approach covers a larger category of feature shapes. Regarding the work of Bolles, the type of constraints which can be held by our approach go beyond the restricted set of surface shapes and cover also the geometric relationships between object features. To our knowledge the work appears the first to give such a large framework for the integration of geometric relationships in object modelling.

3 The optimization technique

Given sets of 3D measurement points representing surfaces belonging to a certain object, we want to estimate the different surfaces' parameters, taking into account the geometric constraints between these surfaces.

A state vector \mathbf{p} is associated to the object, which includes the set of parameters related to the patches. The vector \mathbf{p} has to best fit the data while satisfying the constraints. So, the problem that we are dealing with is a constrained optimization problem to which an optimal solution may be provided by minimizing the following function:

$$E(\mathbf{p}) = F(\mathbf{p}) + C(\mathbf{p}) \quad (1)$$

where $F(\mathbf{p})$ is the objective function defining the relationship between the set of data and the parameters and $C(\mathbf{p})$ is the constraint function. $F(\mathbf{p})$ could be the likelihood of the range data given the parameters (with a negative sign since we want to minimize) or the least squares error function. The likelihood function has the advantage of considering the statistical aspect of the measurements. In a first step, we have chosen the least squares function as the integration of the data noise characteristics into the LS function can be done afterwards with no particular difficulty, leading to the same estimation of the likelihood function in the case of the Gaussian distribution.

Given M geometric constraints, the constraint function is represented by the following equation:

$$C(\mathbf{p}) = \sum_{k=1}^M \lambda_k C_k(\mathbf{p}) \quad (2)$$

where $C_k(\mathbf{p})$ is a vector function associated to constraint k . λ_k are weighting coefficients used to control the contribution of the constraints in the parameters' estimation. Each function is required to be convex since many robust techniques

for minimizing convex functions are available in the literature. The objective function $F(\mathbf{p})$ is convex by definition, so therefore $E(\mathbf{p})$ is also convex.

Figure 1 shows the optimization algorithm that we have used, which has been simplified so that a single λ is associated to all the constraints. The algorithm starts with an initial parameter vector $\mathbf{p}^{[0]}$ that satisfies the least squares function. Then we iteratively increase λ and solve for a new optimal parameter $\mathbf{p}^{[n+1]}$ using the previous $\mathbf{p}^{[n]}$. The new optimal vector is found by means of the standard Levenberg-Marquardt algorithm. The algorithm stops when the constraints are satisfied to the desired degree or when the parameter vector remains stable for a certain number of iterations. The initial value λ_0 has to be large enough to avoid the trivial null solution and to give the constraints a certain initial weight. A convenient value for the initial λ is : $\lambda_0 = F(\mathbf{p}^{[0]})/C(\mathbf{p}^{[0]})$

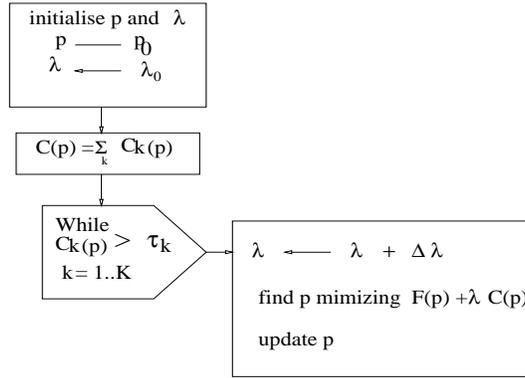


Fig. 1. The optimization algorithm *optim*

4 Preliminaries

This section give a brief overview about constraining quadrics and some particular shapes. A full treatment of these surfaces can be found in [1]. While the material contained here is largely elementary geometry, we present it in order to make clear how the set of constraints used for each surface type and relationship relate to the parameters of the generic quadric. The generic quadric form is used because it is easy to generate a least squares surface fit using the algebraic distance.

A general quadric surface is represented by the following quadratic equation:

$$f(x, y, z) = ax^2 + by^2 + cz^2 + 2hxy + 2gxz + 2fyz + 2ux + 2vy + 2wz + d = 0 \quad (3)$$

which can be written : $X^T A X + 2X^T B + C = 0$

where

$$A = \begin{bmatrix} a & h & g \\ h & b & f \\ g & f & c \end{bmatrix}, \quad B = [u, v, w]^T, \quad C = d; \quad X = [x, y, z]^T \quad (4)$$

The type of the quadric depends on the discriminant of the quadric Δ and the cubic discriminant \mathcal{D} :

$$\Delta = \begin{vmatrix} a & h & g & u \\ h & b & f & v \\ g & f & c & w \\ u & v & w & d \end{vmatrix} \quad \mathcal{D} = \begin{vmatrix} a & h & g \\ h & b & f \\ g & f & c \end{vmatrix} \quad (5)$$

and the cofactors of \mathcal{D} :

$$\begin{aligned} \mathcal{A} &= bc - f^2 \\ \mathcal{B} &= ac - g^2 \\ \mathcal{C} &= ab - h^2 \\ \mathcal{F} &= gh - af \\ \mathcal{G} &= hf - bg \\ \mathcal{H} &= gf - ch \end{aligned} \quad (6)$$

4.1 The cylinder

The quadric is a cylinder when $\Delta = \mathcal{D} = 0$, $u\mathcal{A} + v\mathcal{H} + w\mathcal{G} = 0$ and $\mathcal{A} + \mathcal{B} + \mathcal{C} > 0$. The equation of the cylinder axis is

$$\frac{x - \frac{uf}{\mathcal{F}}}{1/\mathcal{F}} = \frac{y - \frac{vg}{\mathcal{G}}}{1/\mathcal{G}} = \frac{z - \frac{wh}{\mathcal{H}}}{1/\mathcal{H}} = 0 \quad (7)$$

this means that the cylinder axis has as direction the vector $(1/\mathcal{F}, 1/\mathcal{G}, 1/\mathcal{H})$ and passes through the point $X_o(\frac{uf}{\mathcal{F}}, \frac{vg}{\mathcal{G}}, \frac{wh}{\mathcal{H}})$. The axis orientation corresponds to the eigenvector of the matrix A related to the null eigenvalue. The two other eigenvalues are positive.

The circular cylinder For a circular cylinder, we can show that the parameters of the quadric should also satisfy the following conditions:

$$\begin{aligned} agh + f(g^2 + h^2) &= 0 \\ bhf + g(h^2 + f^2) &= 0 \\ cfg + h(f^2 + g^2) &= 0 \\ \frac{u}{f} + \frac{v}{g} + \frac{w}{h} &= 0 \end{aligned} \quad (8)$$

A circular cylinder may be also represented by the canonical form:

$$(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2 - (n_x(x-x_0) + n_y(y-y_0) + n_z(z-z_0))^2 - r^2 = 0 \quad (9)$$

where $X_o = [x_o, y_o, z_o]^T$ is an arbitrary point on the axis, $\mathbf{n} = [n_x, n_z, n_y]^T$ is a unit vector along the axis and r is the radius of the cylinder.

This form has the advantage of having a minimal number of parameters. However its implementation in the optimization algorithm may cause some complexity, indeed it is not possible with this form to get separate terms for the data and the parameters as in (3) (which allows the data terms to be computed off line). Consequently this may increase the computational cost dramatically.

The expansion of (9) and the identification with (3) yields

$$\begin{aligned} a &= 1 - n_x^2 \\ b &= 1 - n_y^2 \\ c &= 1 - n_z^2 \\ h &= -n_x n_y \\ g &= -n_x n_z \\ f &= -n_y n_z \end{aligned} \tag{10}$$

These equations have the advantage of imposing implicitly the circularity constraints of the cylinder and avoid the problem when one of the parameters (f, g, h) vanishes. Besides, they make concrete the geometric relationships between the cylinder and other object features as we will see in Section 5.2.

4.2 The cone

A cone surface satisfies $\Delta \neq 0, \mathcal{D} = 0$. The summit of the cone is given by:

$$X_o = A^{-1}B \tag{11}$$

The axis of the cone corresponds to the eigenvector related to the negative eigenvalue of the matrix A . The two other eigenvalues are both positive.

Circular cone For a circular cone the parameters of the quadric equation have to satisfy the following conditions

$$\frac{af - gh}{f} = \frac{bg - hf}{g} = \frac{ch - fg}{h} \tag{12}$$

As for the cylinder case, a circular cone equation has a more compact form:

$$[(x - x_o)^2 + (y - y_o)^2 + (z - z_o)^2] \cos^2(\alpha) - [n_x(x - x_o) + n_y(y - y_o) + n_z(z - z_o)]^2 = 0 \tag{13}$$

where $[x_o, y_o, z_o]^T$ is the summit of the cone, $[n_x, n_y, n_z]^T$ is the unit vector defining the orientation of the cone axis and α is the semi-vertical angle. The quadric equation parameters can thus be expressed explicitly as a function of the above terms by :

$$a = n_x^2 - \cos^2\alpha \tag{14}$$

$$\begin{aligned}
b &= n_y^2 - \cos^2\alpha \\
c &= n_z^2 - \cos^2\alpha \\
h &= n_x n_y \\
g &= n_x n_z \\
f &= n_y n_z
\end{aligned}$$

For the same reasons mentioned in the cylinder case the compact form of the cone equation is not adequate for the optimization algorithm. Nevertheless it is useful to implicitly impose the conic constraints by means of equations (14) Instead of (12). Indeed in this form all the parameters (f, g, h) need to be different of zero.

4.3 Planes

A plane surface can be represented by this following equation:

$$n_x x + n_y y + n_z z + d = 0; \quad (15)$$

where $\mathbf{n} = [n_x, n_y, n_z]^T$ is unit normal vector ($\|\mathbf{n}\| = 1$) to the plane and d is the distance to the origin.

5 Application on some test objects

The objects treated in this section are real parts. The data was acquired with a 3D triangulation range sensor. The range measurements were already segmented into groups associated with features by means of the *rangeseg* [7] program.

5.1 Notation

For the rest of the paper we need the following notations:

\mathbf{i}_r is a vector which all the elements are null except the r^{th} element which is equal to 1.

$\mathbf{j}_{(r,s)}$ is a vector which all the elements are null except the r^{th} and the s^{th} elements which are equal to 1 and -1 respectively.

$M_{(r,s)}$ is a diagonal matrix which all the elements are null except the r^{th} and the s^{th} elements which are equal to 1 and -1 respectively.

$U_{(r,s)}$ is a diagonal matrix defined by

$$U_{(r,s)} = \begin{cases} U(i, i) = 1 & \text{if } r \leq i \leq s \\ U(i, i) = 0 & \text{otherwise} \end{cases}$$

$L_{(r,s,p)}$ a symmetric matrix defined by

$$L_{(r,s,p)} = \begin{cases} L(i, j) = L(j, i) = 1/2 & \text{if } i = r + t, j = s + t \quad 0 \leq t \leq p \\ L(i, j) = L(j, i) = 0 & \text{otherwise} \end{cases}$$

5.2 The half cylinder

This object is composed of four surfaces. Three patches S_1 , S_2 and S_3 have been extracted from two views represented in Figure 2(a,c). These surfaces correspond respectively to the base plane S_2 , lateral plane S_1 and the cylindrical surface S_3 (Figure 2.b). The parameter vector is $\mathbf{p} = [\mathbf{p}_1^T, \mathbf{p}_2^T, \mathbf{p}_3^T]^T$, where $\mathbf{p}_1 = [\mathbf{n}_1^T, d_1]^T$, $\mathbf{p}_2 = [\mathbf{n}_2^T, d_2]^T$ and $\mathbf{p}_3 = [a, b, c, h, g, f, u, v, w, d]^T$. The least squares error function is given by:

$$F(\mathbf{p}) = \mathbf{p}^T H \mathbf{p}, \quad H = \begin{bmatrix} H_1 & O_{(4,4)} & O_{(4,10)} \\ O_{(4,4)} & H_2 & O_{(4,10)} \\ O_{(4,10)}^T & O_{(4,10)}^T & H_3 \end{bmatrix} \quad (16)$$

where H_1, H_2, H_3 are the data matrices related respectively to S_1, S_2, S_3 :

$$H_i = \sum_j (X_j^i)(X_j^i)^T \text{ for } X_j^i \text{ belonging to surface } S_i$$

This object has the following constraints

1. S_1 and S_2 are perpendicular,
2. the cylinder axis is parallel to S_1 's normal,
3. the cylinder axis lies on the surface S_2 ,
4. the cylinder is circular.

Constraint 1 is expressed by the following condition

$$C_{ang}(\mathbf{p}) = (\mathbf{n}_1^T \mathbf{n}_2)^2 = (\mathbf{p}^T L_{(1,5,2)} \mathbf{p})^2 = 0; \quad (17)$$

Constraint 2 is satisfied by equating the unit vector \mathbf{n} in (9) to S_1 's normal \mathbf{n}_1 . Constraint 3 is represented by two conditions: axis vector \mathbf{n} is orthogonal to S_2 's normal \mathbf{n}_2 , and one point of the axis satisfies S_2 's equation. The first

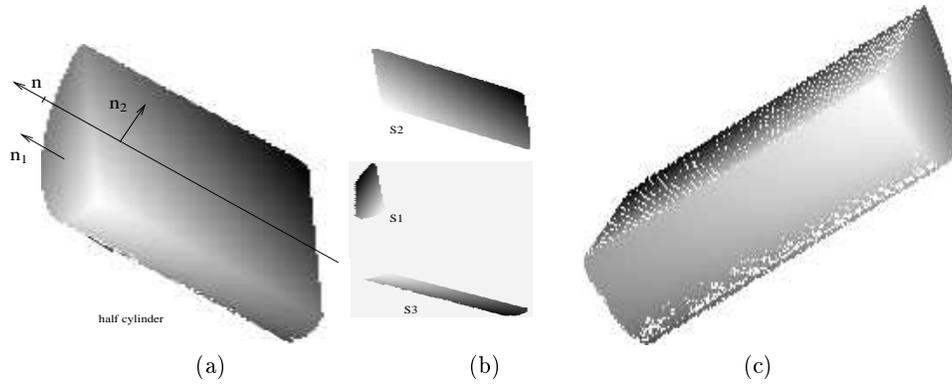


Fig. 2. Two views of the half cylinder and the extracted surfaces

condition is guaranteed by constraint 2 since \mathbf{n}_2 is orthogonal to \mathbf{n}_1 . For the second condition the point X_o in Section 4.1 has to satisfy the equation:

$$C_{axe}(\mathbf{p}) = (X_o^T \mathbf{n}_2 + d_2)^2 = (-[u, v, w]^T \mathbf{n}_2 + d_2)^2 = (\mathbf{i}_8^T \mathbf{p} - \mathbf{p}^T L_{(5,15,2)} \mathbf{p})^2 = 0 \quad (18)$$

using equations (6) and (10).

The cylinder circularity constraint is implicitly defined by the equations (10). From these equations we extract the following constraints on the parameter vector \mathbf{p} :

$$\begin{aligned} C_{circ_1}(\mathbf{p}) &= (\mathbf{i}_9^T \mathbf{p} + \mathbf{p}^T U_{(1,1)} \mathbf{p} - 1)^2 = 0 & (19) \\ C_{circ_2}(\mathbf{p}) &= (\mathbf{i}_{10}^T \mathbf{p} + \mathbf{p}^T U_{(2,2)} \mathbf{p} - 1)^2 = 0 \\ C_{circ_3}(\mathbf{p}) &= (\mathbf{i}_{11}^T \mathbf{p} + \mathbf{p}^T U_{(3,3)} \mathbf{p} - 1)^2 = 0 \\ C_{circ_4}(\mathbf{p}) &= (\mathbf{i}_{12}^T \mathbf{p} + \mathbf{p}^T L_{(1,2,0)} \mathbf{p})^2 = 0 \\ C_{circ_5}(\mathbf{p}) &= (\mathbf{i}_{13}^T \mathbf{p} + \mathbf{p}^T L_{(1,3,0)} \mathbf{p})^2 = 0 \\ C_{circ_6}(\mathbf{p}) &= (\mathbf{i}_{14}^T \mathbf{p} + \mathbf{p}^T L_{(2,3,0)} \mathbf{p})^2 = 0 \end{aligned}$$

We group then all the above constraints in a single one

$$C_{circ}(\mathbf{p}) = \sum_{k=1}^6 C_{circ_k}(\mathbf{p}) = 0 \quad (20)$$

Finally the normals \mathbf{n}_1 and \mathbf{n}_2 have to be unit. This is represented by:

$$C_{unit}(\mathbf{p}) = (\mathbf{p}^T U_{(1,3)} \mathbf{p} - 1)^2 + (\mathbf{p}^T U_{(5,7)} \mathbf{p} - 1)^2 = 0 \quad (21)$$

The constraint function is then

$$C(\mathbf{p}) = C_{unit}(\mathbf{p}) + C_{ang}(\mathbf{p}) + C_{axe}(\mathbf{p}) + C_{circ}(\mathbf{p}) \quad (22)$$

and optimisation function is

$$E(\mathbf{p}) = \mathbf{p}^T \mathbf{H} \mathbf{p} + \lambda (C_{unit}(\mathbf{p}) + C_{ang}(\mathbf{p}) + C_{axe}(\mathbf{p}) + C_{circ}(\mathbf{p})) \quad (23)$$

Experiments In the first test, the algorithm *optim* has been applied to data extracted from a single view (Figure 2.c). The behaviour of the constraints (17),(18),(20) and (21) during the optimization have been mapped as a function of λ as well as the least squares residual (16) and the constraint function (22). The figures show a linear logarithmic decrease of the constraints with respect to λ . It is also noticed that at the end of the optimization all the constraints are highly satisfied. The least squares error converges to a stable value and the constraint function vanishes at the end of the optimization. The figures also show that it is possible to continue the optimization further until a higher tolerance is reached, however this is limited by the computing capacity of the machine. We have noticed that beyond a certain value of λ some numerical instabilities occurred.

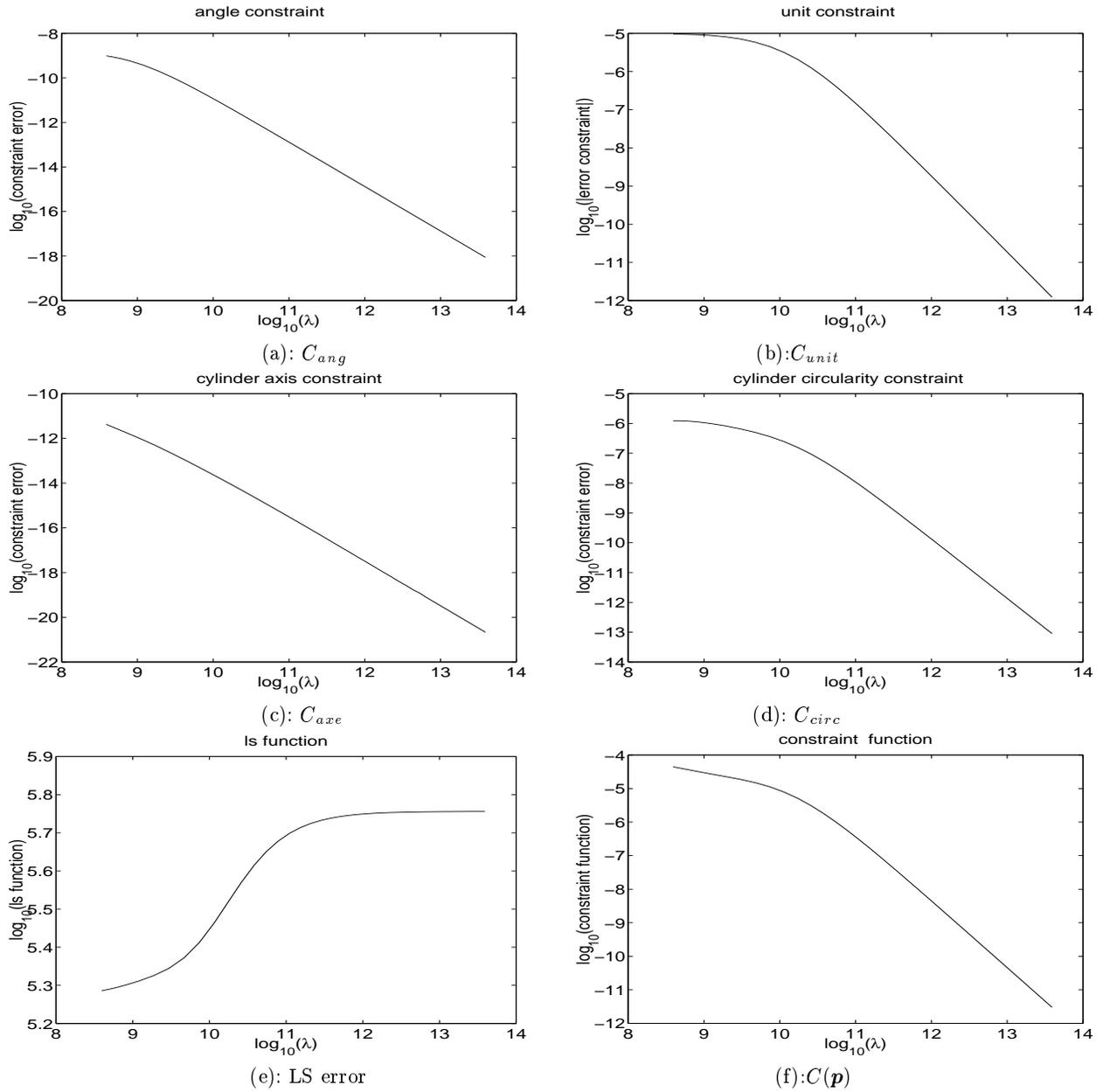


Fig. 3. (a),(b),(c),(d): decrease of the different constraints with respect to λ . (e),(f): variation of least squares function and the constraint function with respect to λ .

In the second test, registered data from view1 (Figure 2.a) and view2 (Figure 2.c) have been used. The registration was carried out by hand. Results similar to the first test have been obtained for the constraints.

Tables 1 and 2 represent the values of some object characteristics obtained from an estimation without considering the constraints and from the presented optimization algorithm. These are shown for the first and second test respectively.

The characteristics examined are the angle between plane S_1 and plane S_2 , the distance between the cylinder axis's point X_o (see (4.1) and (9)) and the plane S_2 and the radius of the cylinder. The comparison of the tables' values for the two approaches show the clear improvement carried by the proposed technique. This is noticed in particular for the radius which the actual value is $30mm$, although the extracted surface covers considerably less than a half of a cylinder. As we constrained the angle and distance relations, we expect these to be satisfied, as they are to almost an arbitrarily high tolerance, as seen in Fig.3. The radius was not constrained, but the other constraints on the cylinder have allowed the least squares fitting of the unconstrained parameters to achieve a much more accurate estimation of the cylinder radius in both cases.

view2	angle(S_1, S_2)(degree)	distance(X_o, S_2)(mm)	radius(mm)
without constraints	90.84	6.32	26.98
with constraints	90	0	29.68
actual values	90	0	30

Table 1. Improvement in shape and placement parameters with and without constraints from data from single view.

registered view1 and view2	angle(S_1, S_2)(degree)	distance(X_o, S_2)(mm)	radius(mm)
without constraints	89.28	2.23	30.81
with constraints	90	0	30.06
actual values	90	0	30

Table 2. Improvement in shape and placement parameters with and without constraints from data merged from two views.

5.3 The cone object

This object contain two surfaces: a plane (S_1) and a cone patch (S_2) (Figure4(c,d)). Two views have been taken for this object (Figure4(a,b))

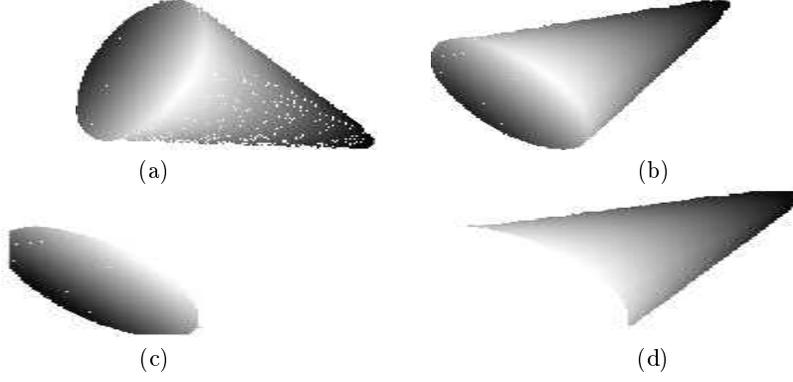


Fig. 4. (a,d): Two views of part , (c,d): extracted patches

The parameter vector is $\mathbf{p} = [\mathbf{p}_1^T, \mathbf{p}_2^T]^T$, where $\mathbf{p}_1 = [\mathbf{n}_1^T, d_1]^T$ and $\mathbf{p}_2 = [a, b, c, h, g, f, u, v, w, d]^T$. The least squares error function is given by:

$$F(\mathbf{p}) = \mathbf{p}^T H \mathbf{p}, \quad H = \begin{bmatrix} H_1 & O_{(4,10)} \\ O_{(4,10)}^T & H_2 \end{bmatrix} \quad (24)$$

where H_1 and H_2 are the data matrices related to S_1 and S_2 . This object involves the following constraints: 1) the cone axis is parallel to S_1 , 2) the cone is circular. Constraint 1 is imposed if S_1 's normal is equated to the unit vector \mathbf{n} of the cone axis. Eliminating $\cos^2 \alpha$ from the cone circularity equations (14) and taking into consideration constraint 1, the circularity constraints are formulated as :

$$\begin{aligned} a - b &= n_{1_x}^2 - n_{1_y}^2 & (25) \\ a - c &= n_{1_x}^2 - n_{1_z}^2 \\ b - c &= n_{1_y}^2 - n_{1_z}^2 \\ h &= n_{1_x} n_{1_y} \\ g &= n_{1_x} n_{1_z} \\ f &= n_{1_y} n_{1_z} \end{aligned}$$

A matrix formulation of these equations as a function of the parameter vector \mathbf{p} is:

$$\begin{aligned} C_{circ_1}(\mathbf{p}) &= (\mathbf{j}_{(5,6)}^T \mathbf{p} - \mathbf{p}^T M_{(1,2)} \mathbf{p})^2 = 0 & (26) \\ C_{circ_2}(\mathbf{p}) &= (\mathbf{j}_{(5,7)}^T \mathbf{p} - \mathbf{p}^T M_{(1,3)} \mathbf{p})^2 = 0 \\ C_{circ_3}(\mathbf{p}) &= (\mathbf{j}_{(6,7)}^T \mathbf{p} - \mathbf{p}^T M_{(2,3)} \mathbf{p})^2 = 0 \\ C_{circ_4}(\mathbf{p}) &= (\mathbf{i}_8^T \mathbf{p} - \mathbf{p}^T L_{(1,2,0)} \mathbf{p})^2 = 0 \\ C_{circ_5}(\mathbf{p}) &= (\mathbf{i}_9^T \mathbf{p} - \mathbf{p}^T L_{(1,3,0)} \mathbf{p})^2 = 0 \\ C_{circ_6}(\mathbf{p}) &= (\mathbf{i}_{10}^T \mathbf{p} - \mathbf{p}^T L_{(2,3,0)} \mathbf{p})^2 = 0 \end{aligned}$$

which are grouped into a single constraint $C_{circ}(\mathbf{p}) = \sum_{k=1}^6 C_{circ_k}(\mathbf{p}) = 0$. Considering as well the unit constraint related to the normal \mathbf{n}_1 $C_{unit}(\mathbf{p}) = (\mathbf{p}^T U_{(1,3)} \mathbf{p} - 1)^2 = 0$. The whole constraint function is : $C(\mathbf{p}) = C_{unit}(\mathbf{p}) + C_{circ}(\mathbf{p})$.

Experiments The optimization algorithm was applied to data from a single view and to registered data. The behaviour of the unit constraint, the circularity constraint, the least squares function and the constraint function during the optimization are qualitatively identical to that shown in Fig.3. As with the cylinder object, the constraints are satisfied up to a high tolerance.

For both the single view and the registered data, the angle between the cone axis and the plane normal, the distance from the cone summit to the plane and the semi vertical angle α have been computed. Table 3 and Table 4 show the estimated values.

We notice that the orthogonality of the plane and the cone axis is almost perfectly satisfied with the optimization. The actual values of the distance and the angle α are not known with high accuracy. Nevertheless the estimated values are within the errors' tolerances.

view1	angle(cone axis , \mathbf{n}_1)(degree)	distance(X_o, S_1)(mm)	α (degree)
without constraints	2.61	72.74	20.78
with constraints	0.00	72.55	19.68
True values	0	70	20

Table 3. Improvement in shape and placement parameters with and without constraints from data from a single view.

registered view1 and view2	angle(cone axis , \mathbf{n}_1)(degree)	distance(X_o, S_1)(mm)	α (degree)
without constraints	0.79	71.73	19.20
with constraints	0.00	70.09	19.59
True values	0	70	20

Table 4. Improvement in shape and placement parameters with and without constraints from data merged from two views.

5.4 The multi-quadric object

The third series of tests have been carried out on a more complicated object (Fig.5). It has two lateral planes S_1 and S_2 , a back plane S_3 , a bottom plane S_4 , a cylindrical surface S_5 and a conic surface S_6 . The cylindrical patch is less than

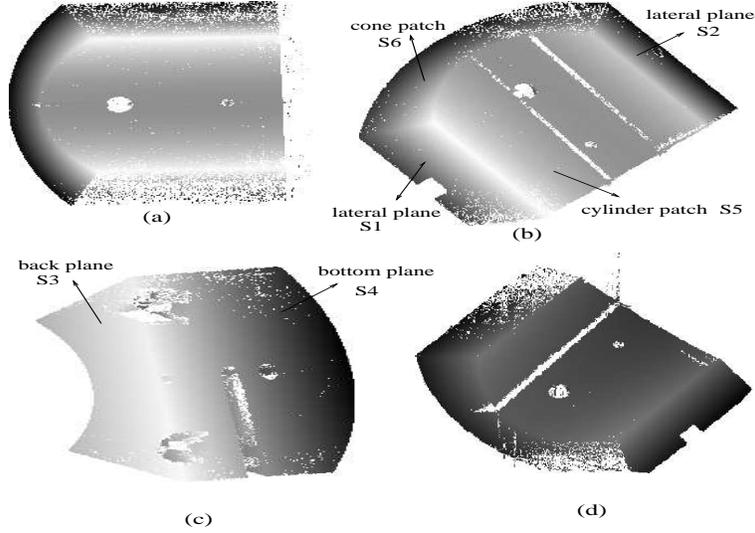


Fig. 5. four views of the multi-quadric object

a half cylinder (40% arc), the conic patch occupies a small area of the whole cone (less than 30%)

The vector parameter associated to this object is then $\mathbf{p}^T = [\mathbf{p}_1^T, \mathbf{p}_2^T, \mathbf{p}_3^T, \mathbf{p}_4^T, \mathbf{p}_5^T, \mathbf{p}_6^T]$ where \mathbf{p}_i is the parameter vector associated to the surface S_i .

The considered surfaces of the object have the following constraints

1. S_1 makes an angle of 120° ¹ with S_2
2. S_1 and S_2 are perpendicular to S_3
3. S_1 and S_2 make an angle of 120° with S_4
4. S_3 is perpendicular to S_4
5. the axis of the cylindrical patch S_5 is parallel to S_3 's normal
6. the axis of the cone patch S_6 is parallel to S_4 's normal
7. the cylindrical patch is circular
8. the cone patch is circular

These constraints are then represented in the same manner as in Section 5.2 and Section 5.3.

Experiments Since the surfaces can not be recovered from a single view, four views (Fig.5) have been registered by hand. The results regarding the algorithm convergence are qualitatively identical to those shown in Fig.3. All the constraint functions vanish and are highly satisfied.

In order to check the robustness and the stability of the technique, we have carried out 100 optimizations, in each of them 50% of the surfaces' points are

¹ We consider the angle between normals.

angle	(S_1, S_2)	(S_1, S_3)	(S_1, S_4)	(S_2, S_3)	(S_2, S_4)	(S_3, S_4)
without constraints	119.76	92.08	121.01	87.45	119.20	90.39
with constraints	120.00	90.00	120.00	90.00	120.00	90.00
actual values	120	90	120	90	120	90

Table 5. Improvement of the surface's angle estimation.

selected randomly. The results shown below are the average of this tests. Our first intention was to compare the constrained approach with an object estimation method which does not consider constraints, in this case the least squares technique applied to each surface separately.

In Table 5 the angles between the different planes are mapped, we notice that all the angles converge to the actual values. Table 6 and Table 7 contain the estimated values of some attributes of the cylinder and the cone. The values show that each of the axis constraints are perfectly satisfied, the estimated radius and the cone half angle θ are quite close to the actual ones. We notice the good shape improvement of improvement, relative to the unconstrained least squares method, given by reduction of bias of about $12mm$ and 3° respectively in the radius and the half angle estimation. The standard deviation of the estimations have also been reduced.

The radius estimation is within the hoped tolerances, a systematic error of about $0.5mm$ is quite nice. However the cone half angle estimation seems to involve a larger systematic error (about 1.8°). Two factors may contribute to this fact: 1) the registration error may be too large since it was made by hand and 2) the area of the cone patch covers less than 30 % of the whole cone. It is known that when a quadric patch does not contain enough information concerning the curvature, the estimation is very biased, even when robust techniques are applied, because it is not possible to predict the variation of the surface curvature.

cylinder parameters	angle(axis , S_3 's normal)	radius	standard deviation
without constraints	2.34	37.81	0.63
with constraints	0.00	59.65	0.08
actual values	0	60	0

Table 6. Improvement of the cylinder characteristic estimates.

cone parameters	angle(axis , S_4 's normal)	α	standard deviation
without constraints	6.0866	26.0108	0.3024
with constraints	0	31.8389	0.1337
actual values	0	30	0

Table 7. Improvement of the cone characteristic estimates.

angle	(S_1, S_2)	(S_1, S_3)	(S_1, S_4)	(S_2, S_3)	(S_2, S_4)	(S_3, S_4)
without constraints	119.76	92.08	121.48	87.45	119.20	90.39
with constraints	119.99	90.33	120.00	90.00	120.00	90.00
actual values	120	90	120	90	120	90

Table 8. Improvement of non-constrained angle estimates.

We have also investigated whether leaving some features unconstrained will affect the estimation since one can say that the satisfaction of the other constraints may push the unconstrained surfaces away from their actual positions. To test this, we have left the angles between the pair of planes (S_1, S_2) and (S_1, S_3) unconstrained. The results in Table 8 show that the estimated unconstrained angles are still close to the actual ones and the accuracy is improved compared to the non-constrained method. The computation time for this object in Matlab was 48s on a 200Mhz sun Ultrasparc workstation.

6 Conclusion

If we consider the objectives stated in the introduction which are : object shape reconstruction which satisfies the constraints and improves the estimation accuracy, we can say that these objectives have been reached. The experiments show that parameter optimization search does produce shape fitting that almost perfectly satisfied the constraints. The comparison of the results with the non-constrained fitting confirms that the proposed approach improves the quality of the fitting accuracy to a high degree. For the two objects having cylinder patches, the radius error is less then 0.5mm. Results for the cone are reasonable but less satisfactory. This is mainly due to the relatively small area of the conic patch. Actually, we intentionally chose to work with small patches because it is the case when non-constrained fitting surface techniques fail to give reasonable estimation even with the robust algorithms. This is due to the “poorness” of the information embodied in the patch. However we intend to investigate a more robust form for the objective function which involves the data noise statistics.

Regarding the constraint representation, it is noticed that some constraints involve a large number of equations, in particular for the circularity constraint. One solution is to implicitly impose this constraints through the representation of the quadric equation $((X - X_o)^T(I - nn^T)(X - X_o) - r^2 = 0)$ for the cylinder and $((X - X_o)^T(nn^T - \cos^2(\alpha))(X - X_o) = 0)$ for the cone. The main problem encountered with this representation is the complexity of the related objective function and the difficulty of separating the data terms from the parameter terms, but we are working on this issue. It will be also worthwhile to investigate some topological constraints between surfaces which have a common intersection. The adequate formulation of this type of constraints is the main problem to solve.

We are starting to investigate is how one might identify inter-surface relationships that can have a constraint applied. In manufacturing objects, simple

angular and spatial relationships are given by design. It should be straightforward to define Mahalanobis distance tests for standard feature relationships, subject to the feature's statistical position distribution. With this analysis, a computer program could propose a variety of constraints that a human could either accept or reject, after which shape reconstruction could occur.

We have also investigated [14] an approach where constraints are incrementally added, for example by a human reverse engineer, but have found no essential difference in results. The batch satisfaction of all constraints as presented here takes very little computing time, so we no longer use the incremental algorithm.²

References

1. R.J.Bell *An elementary treatise on coordinate geometry.* McMillan and Co, London, 1910.
2. R.M.Bolle, D.B.Cooper *On Optimally Combining Pieces of Information, with Application to Estimating 3-D Complex-Object Position from Range Data.* IEEE Trans. PAMI, Vol.8, No.5, pp.619-638, September 1986.
3. K.L.Boyer, M.J.Mirza, G.Ganguly *The Robust Sequential Estimator.* IEEE Trans. PAMI, Vol.16, No.10, pp.987-1001 October 1994.
4. Y.Chen, G.Medioni *Object Modelling by Registration of Multiple Range Images.* Proc. IEEE Int. Conf. Robotics and Automation, Vol.2 pp.724-729, April, 1991.
5. J.De Geeter, H.V.Brussel, J.De Schutter, M. Decreton *A Smoothly Constrained Kalman Filter.* IEEE Trans. PAMI pp.1171-1177, No.10, Vol.19, October 1997.
6. P.J.Flynn, A.K.Jain *Surface Classification: Hypothesizing and Parameter Estimation.* Proc. IEEE Comp. Soc. CVPR, pp. 261-267. June 1988.
7. A. Hoover, G. Jean-Baptiste, X. Jiang, P. J. Flynn, H. Bunke, D. Goldgof, K. Bowyer, D. Eggert, A. Fitzgibbon, R. Fisher *An Experimental Comparison of Range Segmentation Algorithms.* IEEE Trans. PAMI, Vol.18, No.7, pp.673-689, July 1996.
8. S.Kumar, S.Han, D.Goldgof, K.Boyer *On Recovering Hyperquadrics from Range data.* IEEE Trans. PAMI, Vol.17, No.11, pp.1079-1083, November 1995.
9. S.L.S. Jacoby, J.S Kowalik, J.T.Pizzo *Iterative Methods for Nonlinear Optimization Problems.* Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 1972.
10. J.Porrill *Optimal Combination and Constraints for Geometrical Sensor Data.* International Journal of Robotics Research, Vol.7, No.6, pp.66-78, 1988.
11. M.Soucy, D.Laurendo *Surface Modelling from Dynamic Integration of Multiple Range Views.* Proc 11th Int. Conf. Pattern Recognition, pp.449-452, 1992.
12. H.Y.Shun, K.Ikeuchi, R.Reddy *Principal Component Analysis with Missing Data and its Application to Polyhedral Object Modelling.* IEEE Trans. PAMI, Vol.17, No.9, pp.855-867.
13. B.C.Vemuri, J.K Aggrawal *3D Model Construction from Multiple Views Using Range and Intensity Data.* Proc. CVPR, pp.435-437, 1986.
14. N.Werghi, R.B.Fisher, A.Ashbrook, C.Robertson *Improving Model Shape Acquisition by Incorporating Geometric Constraints.* Proc. BMVC, pp.530-539 Essex, September 1997.

² Acknowledgements: the work presented in this paper was funded by UK EPSRC grant GR /L25110.