High-level CAD Model Acquisition from Range Images*

Andrew W. Fitzgibbon David W. Eggert Robert B. Fisher

Department of Artificial Intelligence
University of Edinburgh
Edinburgh, Scotland EH1 2QL

eggertd, andrewfg or rbf@aifh.ed.ac.uk

Abstract

Automatic extraction of CAD descriptions which are ultimately intended for human manipulation requires the accurate inference of geometric and topological information. We present a system which applies segmentation techniques from computer vision to automatically extract CAD models from range images of parts with curved surfaces. The output of the system is a B-rep of the object which is suitable for further manipulation in a modelling system.

The segmentation process is an improvement upon Besl and Jain's variable-order surface fitting[1], extracting general quadric surfaces and planes from the data, with a postprocessing stage to identify surface intersections and to extract a B-rep from the segmented image.

We present results on a variety of machined objects, which illustrate the high-level nature of the acquired models, and discuss the numerical accuracy (feature sizes and separations) and the correctness of structural inferences of the system.

Keywords: Feature-based reverse engineering, Range image segmentation, Automatic model acquisition

Abbreviated title: CAD Models from Range Images

Address for correspondence: Andrew W. Fitzgibbon
Department of Artificial Intelligence
University of Edinburgh
5 Forrest Hill
Edinburgh, Scotland EH1 2QL
Phone: (44) (0131) 650 4504
Fax: (44) (0131) 650 6899

*This work was funded by UK EPSRC Grant GR/H/86905.

Introduction

Automatic extraction of CAD descriptions for reverse engineering, rather than for simple copy milling, demands models that are ultimately intended for human manipulation. This in turn implies the need for accurate inference of geometric and topological information, expressed in terms of component features and their interconnections. Such a description is conceptually at a higher level than that of approximating raw data with spline patches[2,3] for example.

Previous research into such "feature-based" reverse engineering has been limited either to interactive systems[4,?], which are not suitable when acquisition of many models (for legacy inventory modelling for example) is required. The noninteractive systems that have been described[5] are generally limited to polyhedra, which permits the reverse engineering of only a narrow class of objects.

The problem of feature segmentation is widely recognized in the field of computer vision, where extracted features are used to perform symbolic matching between images. This paper applies computer vision techniques to the automatic inference of geometric and topological structure for reverse engineering. We describe a system which automatically builds CAD models of objects which are bounded by arbitrary piecewise quadric surfaces, significantly expanding the range of applicability of the feature-based technique.

Our previous work[6] segments an image based on the signs of mean and Gaussian curvatures, producing a qualitative description of arbitrary curved surfaces suitable for object recognition. For CAD reconstruction, however, it has a number of inadequacies:

- The surface patches have no parametric description. While reconstruction up to a Euclidean transformation is theoretically possible from the curvatures, this is not a practical CAD representation.
- The patch boundaries are ragged due to the inherent amplification of errors in the extraction of second derivatives for curvature calculation.
- No topological information is provided.

The input data to the new algorithm are a $2\frac{1}{2}$D range image which has been coarsely segmented as described above, and the output is a B-rep model in Fisher's Suggestive Modelling System[7] language.

In the paper we briefly review previous approaches to range-data segmentation, and then describe our algorithm providing examples of its operation on a number of mechanical parts.

Survey of Range Image Segmentation

In this section we briefly survey strategies for range image segmentation, and indicate their relevance to the problem of CAD model acquisition. The surface models used range over planes[8,9], specific quadrics (for example cylinders, spheres and cones)[10,11,23,12,13], general quadrics[14,8], algebraic surfaces[15], superquadrics[16,17], and splines.[2,3] The published algorithms generally fall into three categories:

Split-and-merge algorithms[18,8,19,20,21,22,13] divide the image into many small regions, and then iteratively merge regions that are statistically likely to represent the same surface primitive. Merging stops when no two regions are sufficiently similar. Thresholding the similarity measure provides an easy way to determine the number of regions and scale of segmentation.

Algorithms based on *clustering*[23,24,25] estimate surface parameters on small patches and accumulate the parameters in a histogram. Large peaks in the histogram correspond to instances of the surface in the data. Flynn[23] and Han[24] both look at clusters of surface normals: Flynn uses the eigenvectors of the local normal covariance matrix to discriminate between planar, cylindrical and spherical patches; Han further histograms cross products of pairs of normals to test for and determine cylinder axes. Clustering techniques, like the Hough transform[26], are generally limited to surface types with a small number of free parameters.

Region growing[1,27,28] is a surface fitting technique which fits to each of several small "seed" regions a mathematical surface and then extends these surfaces to encompass pixels adjacent to the seed region. Besl and Jain's much-cited paper[1] describes a technique based on polynomial surfaces of orders up to

Diffusion-smooth range image $\mathbf{x}(i, j)$, preserving discontinuities
Calculate curvature images $H(i, j)$ and $K(i, j)$.
Morphologically smooth curvature images, producing $\tilde{H}(i, j)$ and $\tilde{K}(i, j)$.
Create label image $L(i, j)$ from connected components analysis of $\tilde{H}\tilde{K}(i, j)$.
Create seed regions $\{R_i\}_{i=1}^{n}$ from $L(i, j)$.

for each seed region R_i,
  repeat
    Calculate sample covariance matrix of region points to decide surface order.
    Robustly fit an algebraic surface $S_i(\mathbf{x}) = 0$ to the points of R_i
    Replace R_i with the largest overlapping connected region of points
      which are within a threshold τ of S_i, and are closer to S_i
      than to any other S_j.
  until R_i does not change.
end
Create 2-D adjacency graph from label image $L(i, j)$
Create 3-D Region-Sheet (RS) graph
Process RS graph into produce B-rep Region-Sheet-Curve-Vertex graph.
Optionally postprocess to replace general quadrics with specifics.

Figure 1: Algorithm overview

four. These systems work relatively well on simple industrial parts with sharp orientation and even curvature discontinuities but prove inadequate on more freeform surfaces.

The Algorithm

Our system is an improvement on the method of Besl and Jain[1], with postprocessing stages to recover the topological information and to convert the segmentation output to a B-rep model. Figure 1 gives an overview of the procedure, which comprises the following steps:

Curvature Classification.

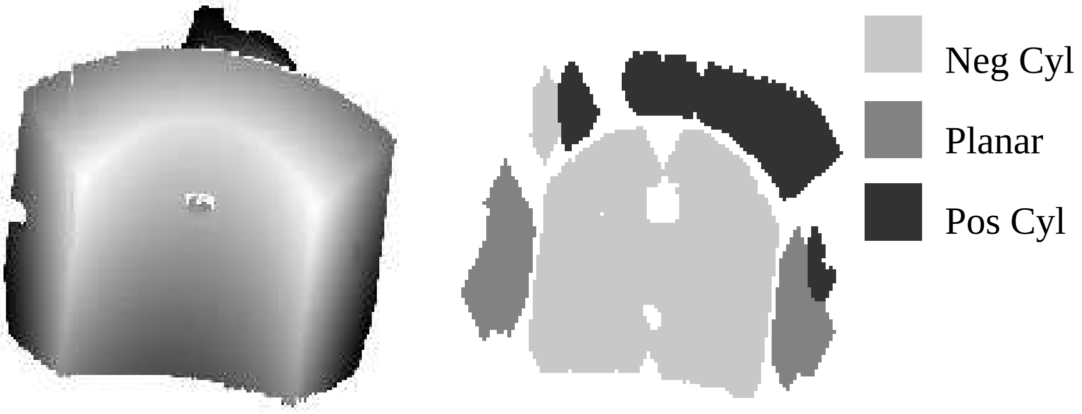
To summarize Trucco[6], we estimate principal curvatures at each pixel after spike noise removal and boundary-preserving smoothing have been applied to the raw data. From the principal curvatures κ_1 and κ_2, we calculate the Gaussian (K) and mean (H) curvatures from:

$$K = \kappa_1 \kappa_2 \qquad H = \frac{\kappa_1 + \kappa_2}{2}$$

From the signs of these curvatures, we can classify each image pixel into locally planar, cylindrical, spherical, or hyperbolic. A connected components analysis of this image then produces a set of regions of constant curvature class. Figure 2 illustrates the output of this stage.

Morphological Hysteresis Thresholding

In the current system the noisy curvature-sign images are morphologically smoothed using a new hysteresis-like inclusion criterion. Each curvature value is classified as Negative, Zero, Positive or Unknown based on the values of "inner" and "outer" thresholds. The inner threshold determines the range of values called Zero. The outer threshold determines the inner limit of the ranges of the Negative and Positive values. (Figure 3 provides a graphical depiction of these thresholds) Between these values the pixel is labelled as Unknown. These H and K sign maps are then morphologically dilated[29] using a



(a) (b)

Figure 2: (a) Raw range data (b) Curvature classification. The greylevels on the right label pixels as *locally* falling into one of three classes, based on the sign of the mean curvature, H, (on this object the Gaussian curvature is zero everywhere other than at edges).

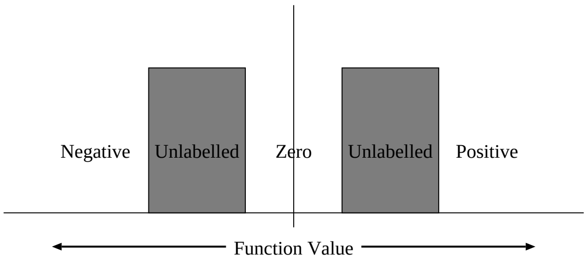


Figure 3: Ranges for hysteresis thresholding.

3×3 cross (“” shaped) structuring element to propagate the known labels into the Unknown regions.

Figure 4 illustrates the effectiveness of this technique on an image where the curvature thresholds have been set artificially low. The image on the left is a synthetic range image rendered using an artificial Lambertian lighting model to illustrate the noise level of the image. The central image shows the result of thresholding the mean curvature with an “inside” threshold of approximately the median curvature value observed in the planar region, and an “outside” threshold of approximately the minimum value in the ellipsoidal region.

Connected Components Analysis

In the last of the preprocessing stages, connected components analysis of the label images groups pixels which are 4-connected and share the same labels. This process produces a set of initial seed regions $\{R_i\}_{i=1}^{n}$, where a region is defined as a set of connected pixels $\{\mathbf{x}_k^i\}_{k=1}^{m_i}$ and an associated label.

Region growing: Initial pass

After the seed regions have been identified, the region growing stage refines the coarse segmentation in order to ensure intersection boundary consistency. Region growing is performed through an iterative expand/fit/contract cycle after the initial surface fitting.

Shaded (a) (b)

Figure 4: Morphological Hysteresis Thresholding. The object on the left has been processed with incorrect threshold values. (a) Mean curvature image with each pixel classified as Negative, Zero, Positive or Unknown. (b) Mean curvature image after application of morphological hysteresis.

Surface fitting

For each region in the initial segmentation above a minimal size a least squares surface fitting is performed, associating with each region R_i an algebraic surface $S_i(\mathbf{x}) = 0$. To perform the fitting, we use Taubin's generalized eigenvector fit[15] (GEVFIT), which minimizes the *approximate mean square distance*

$$\epsilon^2 = \frac{\sum_{k=1}^{m} F(\mathbf{x}_k)^2}{\sum_{k=1}^{m} \|\nabla_{\mathbf{x}} F(\mathbf{x}_k)\|^2}$$

to the surface defined by $F(\mathbf{x}) = 0$.

The choice of fitting algorithm was a result of comprehensive tests on several 2D conic fitting algorithms[30] which found GEVFIT to provide the best tradeoff between speed and accuracy. While the same evaluation has not been formed in 3D, the very simple analogy between quadrics and conics allows us to be confident that the results will extend to higher dimensions.

Here we limit the types of surfaces to quadrics and planes, due to problems of instability when fitting higher-order surfaces. The decision about which type of surface to fit is made by examining the sample covariance matrix of the region points and fitting a plane if the ratio of its two smallest eigenvalues exceeds a preset threshold. Otherwise, a quadric surface is fitted.

Expansion

Next, each region in turn is grown. For expansion, a pixel is added to the current region if it meets the following requirements:

1. it is 2-D adjacent (defined as 4-connectivity on the label image) to the current region,

2. the corresponding 3-D point $\mathbf{p}$ is within a minimum perpendicular distance τ of the current surface. The minimum distance for quadrics is readily calculated if coordinates are transformed[†] so that the quadric is

   $$\mathbf{x}^T D \mathbf{x} = 1$$

   for diagonal D. Then the $\mathbf{x}$ closest to $\mathbf{p}$ satisfies

   $$\mathbf{x} + \lambda D \mathbf{x} = \mathbf{p}$$
   $$(I + \lambda D)\mathbf{x} = \mathbf{p}$$

[†]This transformation cannot be performed for paraboloids and certain degenerate quadrics, but a similar analysis can be easily applied if the paraboloid is transformed to the form $z = ax^2 + by^2$.

So that λ is a solution of $\mathbf{p}^T(I+\lambda D)^{-T}D(I+\lambda D)^{-1}\mathbf{p}=1$, or

$$\frac{d_1 p_1^2}{(1-d_1\lambda)^2}+\frac{d_2 p_2^2}{(1-d_2\lambda)^2}+\frac{d_3 p_3^2}{(1-d_3\lambda)^2}=1$$

Which, on multiplication by $||I-\lambda D||_F^2$ gives a 6^{th} degree polynomial in λ. Choosing the root which minimizes the distance yields the closest point.

3. the point is closer to the current surface than to any other surface for which it may have been labelled during the growing of a previous region,

4. the surface normal at the pixel (estimated by least-squares fitting af a plane to a 5×5 window about the pixel) is within an angular threshold θ of the current surface normal at that position,

5. the estimated pixel normal is in better agreement with the current surface than with any other surface for which it may be labelled.

Choice of minimum perpendicular distance threshold τ

The choice of threshold τ depends on the accuracy of the range sensor, and is chosen such that about 95% of range points are expected to lie within τ of their true surface. The value may be estimated by imaging a surface of known geometry, such as a sphere or a plane, performing a robust least-squares fit, and calculating the 95^{th} percentile residual value. Using the range data gathered by our in-house laser striper, values of $\tau = 0.6$mm and $\theta = 80°$ were used, while for other experiments[31] values of 1mm and 4mm were required.

Contraction

The boundary of the current region is extended in this manner as far as possible. Then the surface is refitted to this new data set. Finally, a contraction of the region boundary is performed. Each pixel is tested using the previous criteria against the new surface estimate. If it is not best accounted for by the new surface, the pixel is returned to the region from which it was originally taken. This expand/contract cycle is iterated until the region boundary stabilizes, or until a maximum iteration limit is reached.

Intersection boundary consistency

After a single pass has been made through the surfaces, the majority of pixels have been labelled. However, the intersection boundaries between surfaces may be very ragged, as there is often a significant overlap between regions because pixels on the boundary will be within τ of both the adjoining surfaces. The criterion above that assigns the label of the *closest* surface to border pixels often fails to give a clean intersection boundary due to the effects of noise in the data, as demonstrated in Figure 5.

To resolve this problem, the second and subsequent passes add a novel compatibility criterion which attempts to ensure that the boundaries of adjacent regions are consistent with the regions' intersection curve. The new criterion is applied only at *ambiguous* pixels (those that are within the distance threshold τ of more than one region), and replaces the "closest point" constraint with one that labels an ambiguous pixel based on its being on the same side of a *decision surface* as the region for which it will be labelled.

In the case of planes, this surface is another plane passing through the line of intersection between the current plane and the plane corresponding to the current labelling of the pixel. This dividing plane is also chosen to bisect the volume of space between the two planes in question. Thus, for two planes

$$\mathbf{n}_1\mathbf{x} = d_1$$
$$\mathbf{n}_2\mathbf{x} = d_2$$

the equation of the decision plane is simply

$$(\mathbf{n}_1-\mathbf{n}_2)\mathbf{x} = d_1 - d_2$$

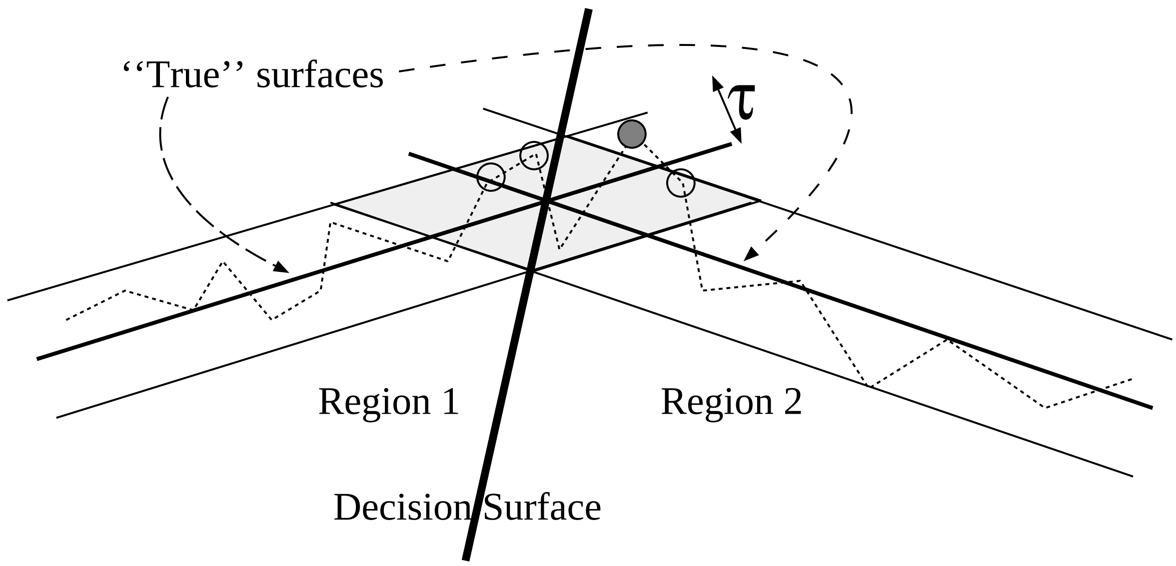


Figure 5: An example showing where assigning pixels to their closest region fails. The thick sloped lines represent two true planes, while the jagged dotted line represents the noisy image of the planes. The closest region criterion will classify the circled pixels into the wrong regions. Applying the decision surface criterion to the pixels in the grey area (where points are within τ of both regions) will correctly classify all but the grey circled pixel, which is outside the threshold range for its "correct" region.

Figure 5 illustrates the concept in one dimension, and shows how this decision surface criterion will produce more consistent labellings in a simple case. The figure also demonstrates that the criterion will fail when a pixel is sufficiently far from its "correct" plane that it falls outside the threshold bands. In this case the connected components labelling will discover this isolated pixel and reject it as being too small to be a single region, but in general if the initial phase produces a labelling that is grossly wrong, this criterion will not be able to correct the error.

The same technique may be applied to curved surface intersections by considering the intersections of the local tangent planes, because the decision surface is calculated separately for each pixel, and is based on the local surface normal.

Topology extraction: Building the B-rep model

Having performed the segmentation step, production of the B-rep consists of extracting 3D adjacency and surface intersection curves and output of the SMS format model.

The topological information in which we are interested comprises the adjacency graph of the scene,

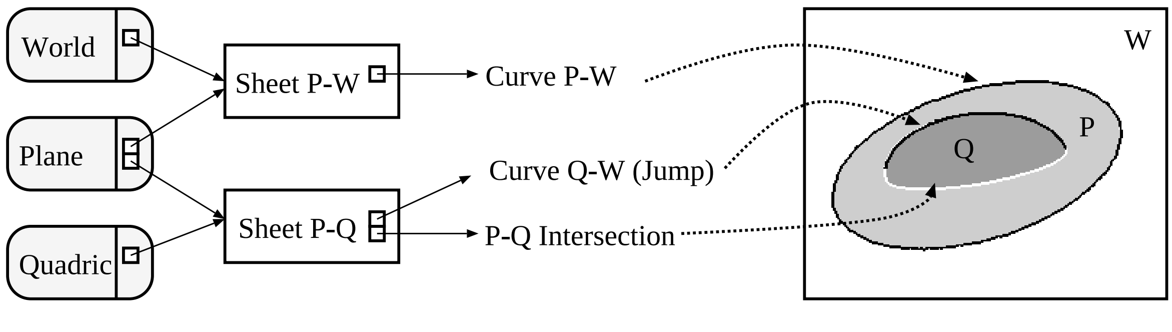


Figure 6: Illustration of the B-rep graph for a simple object

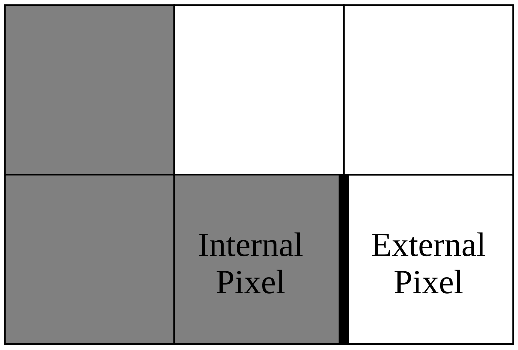


Figure 7: Example dual-lattice edgel. The edgel is defined by the position of its midpoint, which in turn defines its orientation — a fractional X and integral Y imply the edgel is vertical. With each edgel we associate an internal and external pixel.

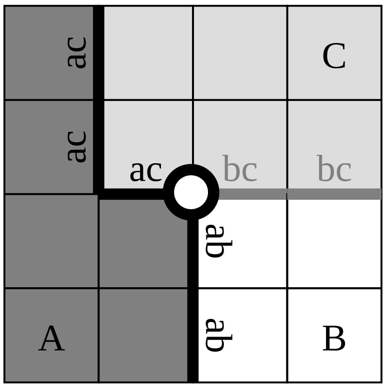


Figure 8: Example of vertex identification. The solid black edgel sequence is the exterior boundary of region A, with edges labelled with the indexes of their adjoining regions. The three-way vertex is simply indicated by the labelling change from "ab" to "ac" as the boundary is tracked.

where two patches are deemed adjacent if their *finite* intersection is a space curve lying on the sensed surfaces. This definition therefore incorporates the identification of intersection curves into the adjacency determination process.

To assist in this process, we do have a certain amount of topological information available initially. The segmented $2\frac{1}{2}$D range image gives us a 2-D adjacency map, and we can make the following observation:

> If two 3-D region boundaries are within τ of each other, they will be 2-D adjacent in the label map, because each boundary point is within τ of both surfaces.

This allows us to extract 2D adjacency during the execution of a dual-lattice boundary tracker, and to mark as 3D-adjacent boundary points which are within τ of both the internal and external surfaces.

The B-rep model structure

The specific B-rep which our system generates is that defined in Fisher's Suggestive Modelling System.[7] The SMS defines surfaces in terms of *shape*, *extent* and *position*[32] where the shape parameters (such as the 3 radii of an ellipsoid) determine the parametric or implicit surface in a canonical position. Surface extent is defined by a collection of space curves which lie on the surface and by a point defining the interior of the surface. This limits surfaces to being singly connected components, so that a surface representation comprises one external sheet, and zero or more "hole" sheets. Each sheet may in turn be defined by an assembly of space curves. Figure 6 illustrates the structure of a simple model.

Boundary tracking

The segmentation process gives us collections of regions whose boundaries are implicit in the label map. By tracking around the pixels using a dual lattice boundary tracker we can build a list of boundary *edgels* around each region. Each edgel contains (in addition to its position and orientation) the label of the external pixel and the distance from the external pixel to the region's surface, as shown in Figure 7.

Using the external distance information, 3-D adjacency can be determined at each edgel as observed above. Clearly this scheme will erroneously report adjacencies along chamfers of radius less than τ. However, as τ has been chosen to reflect the level of noise in the range data, such chamfers are outside the resolution of the system in any case (see Figure 9).

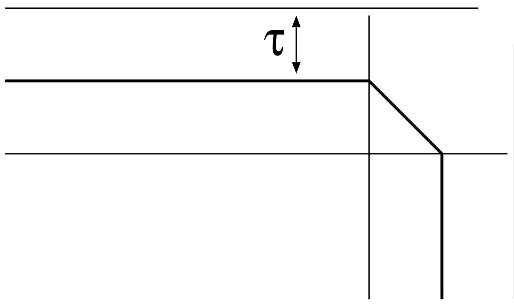


Figure 9: A chamfer that is outside the resolution of our system. Such structures will generally be modelled as two intersecting planes, ignoring the chamfer.

Having identified pixel adjacencies, we can segment the region boundaries into sequences of edgels which share the same internal/external region labels. This gives us a Region-Sheet-Curve decomposition where the curves are still represented as raw sequences of edgels. For example region Q in Figure 6 will be represented as having one sheet, comprising two curves: the 3D adjacency curve between P and Q (labelled QP), and the non-adjacency curve corresponding to Q's occluding contour (labelled QW) indicating the boundary between Q and the World.

Surface intersection: Curves and vertices

We can now replace sequences which denote 3D adjacency between surface patches (curve PQ in the example) with the intersection curves of the adjoining patches.[‡]

On transitions between sections (where region A changes from being adjacent to region B to region C for example — see Figure 8), we know that there must be a vertex found by the intersection of the three surfaces and can therefore define it thus.

Curve description and hole extraction

Finally, in the current case where not all of the object is observed, some boundary segments will be adjacent to the World. Currently these segments are simply approximated by lines and conic sections using a segmentation algorithm based on the run-distribution test.[33] Note that this frequently applies near surface holes, so that this process produces a description of the circular rim of the hole even in the absence of range data from the cylindrical inner surface.

As holes are generally considered an important feature, the system specifically attempts to model internal boundaries as a single circle by the following process:

1. A plane is least-squares fitted to the points $\{\mathbf{x}_i\}_{i=1}^n$ by extracting the centroid and shortest eigenvector of their sample covariance matrix.

2. If the residuals after this fit exceed τ, the process is terminated.

3. The points are transformed to lie in the X-Y plane, where they are expressed as $\{(x_i, y_i)\}_{i=1}^n$.

4. A circle is least-squares fitted using GEVFIT and the linear parameterization $a_1 + a_2 x_i + a_3 y_i + a_4(x_i^2 + y_i^2) = 0$, producing a center (c_x, c_y) and estimated radius R.

5. The boundary is tracked again, and the sample radii $R_i = \|(x_i, y_i) - (c_x, c_y)\|$ are extracted.

6. Application of the run distribution test to $\{R_i\}_{i=1}^n$ determines whether or not the boundary is a circle.

In the case when a modelled intersection curve reaches such background curves, we link them together using a line segment which connects the last point on the background curve to its closest point on the intersection curve.

‡This paper does not discuss the mathematics of the surface intersections, as the key interest is in the identification of the finite extent of the intersection.

Specialization of quadric surfaces

To be useful in CAD systems, it is often preferable to deal with specific quadric surfaces such as cones, cylinders and spheres rather than with the general 10-parameter form. Our system includes an optional postprocessing stage that applies some simple heuristics to determine whether a generic quadric instance is that of a specific subclass. The process transforms the quadric $\mathbf{x}^T A\mathbf{x} + B\mathbf{x} + C = 0$ into its central form by extracting the eigensystem of A:

$$R^T D R = A$$

rotating the coordinate system:

$$\begin{aligned} A' &\leftarrow R^T A R \\ B' &\leftarrow R^T B \\ C' &\leftarrow C \end{aligned}$$

translating B to the origin: (note that if D is noninvertible, the tests are not performed)

$$\begin{aligned} \mathbf{t} &\leftarrow -\frac{1}{2} D^{-1} B' \\ A'' &\leftarrow A' \\ C'' &\leftarrow C' + \frac{1}{2}\mathbf{t} \cdot B' \\ B'' &\leftarrow B' + 2A'\mathbf{t} \end{aligned}$$

and finally extracting the shape parameters:

$$\begin{aligned} R_x &= \frac{\mathrm{sgn}(A''_{11})}{\sqrt{|A''_{11}|}} \\ R_y &= \frac{\mathrm{sgn}(A''_{22})}{\sqrt{|A''_{22}|}} \\ R_z &= \frac{\mathrm{sgn}(A''_{33})}{\sqrt{|A''_{33}|}} \\ K &= C'' \end{aligned}$$

The following tests are applied to the shape parameters in order to identify specific subclasses:

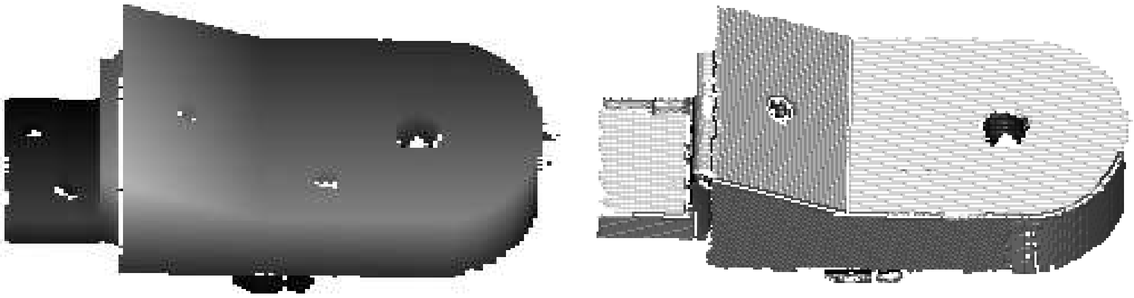
1. if R_x and R_y have the same sign and $|R_z|$ exceeds a threshold R_∞, the surface is a cylinder.
2. if $|K|$ is less than a threshold ϵ_c, the surface is a cone.
3. if R_x, R_y and R_z have the same sign and their absolute values are under R_∞, the surface is an ellipsoid.

In the cases where a specific subclass has been determined, a nonlinear least squares algorithm[34] is applied to fit the appropriate model, which replaces the general quadric description.

Experimental results

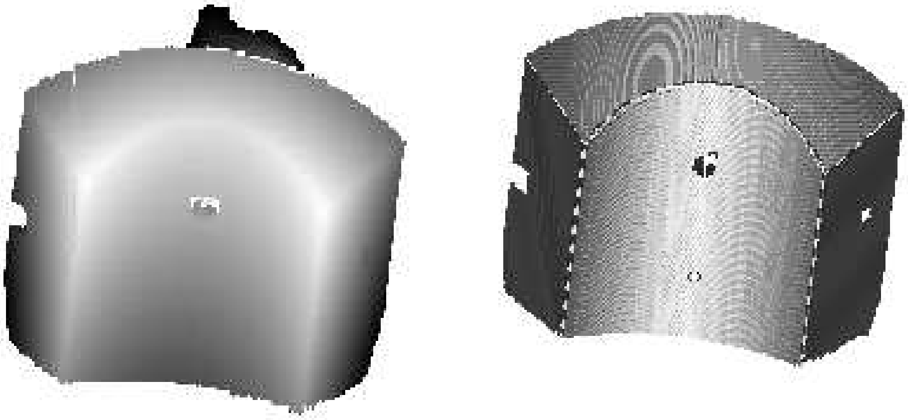
We have implemented the above system in C++ and tested it on range data gathered by the laser striper in our laboratory.[35] In addition, the same algorithm restricted to polyhedra has been tested by Hoover *et. al.*[31] in their comparison of range segmentation systems.

Figures 10, 11 and 12 show the operation of the system on three objects which are bounded by planes and quadric surfaces. The figures show the raw range data and the renderings of the final models in the SMS object viewer.



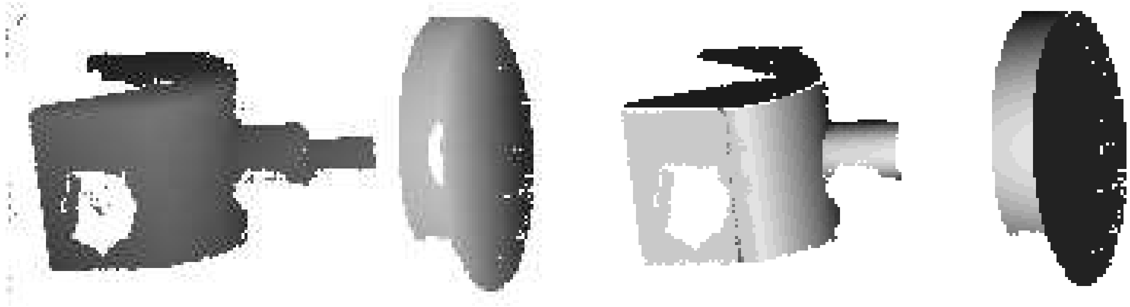
(a) (b)

Figure 10: British Aerospace Widget: (a) Raw range data (b) Automatically acquired model.



(a) (b)

Figure 11: Manufactured part: (a) Raw range data (b) Automatically acquired model



(a) (b)

Figure 12: Optical Stand: (a) Raw range data (b) Automatically acquired model

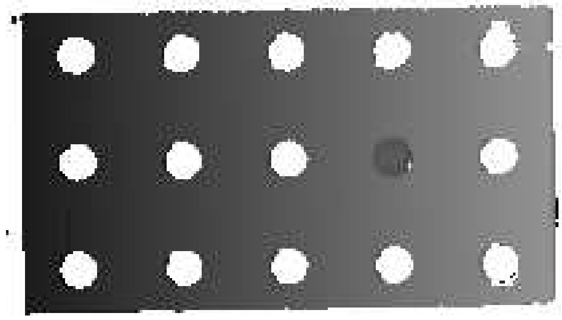


Figure 13: Hole accuracy: Test object. The object is sampled at 1mm intervals in X and Y. The true hole radius is 7.5mm, with a separation of 40mm.

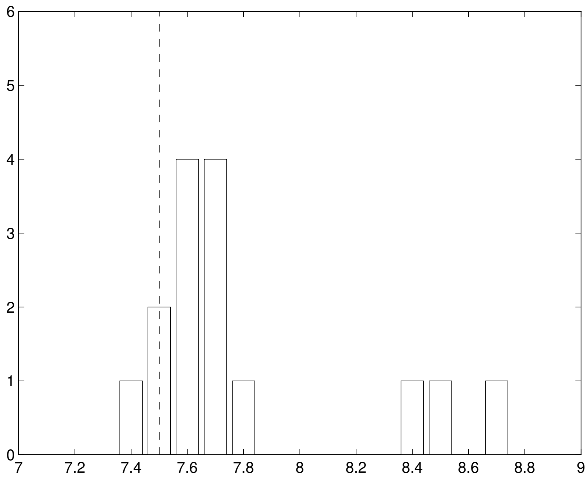


Figure 14: Hole accuracy: Radii. The graph shows the distribution of radii for the 15 holes. Median error is 0.16mm, mean error is 0.32mm.

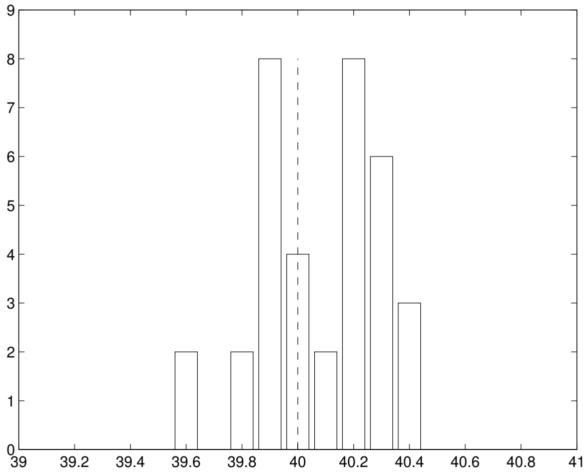


Figure 15: Hole accuracy: Separation. The graph shows the distribution of hole separations. Median error is 0.19mm, mean is 0.20mm.

Object	True	Measured	Error
Widget	45.00	44.93	0.07
Manufactured	60.00	59.88	0.12
Optical Stand	40.00	39.06	0.94
Optical Stand	50.00	49.61	0.39
Optical Stand	90.00	90.34	0.34

Table 1: True versus estimated cylinder radii on sampled objects. All measurements are in mm.

Object	Metric	True	Meas.	Error
Widget	PP	90.00	90.10	0.10
Widget	PP	90.00	80.85	0.15
Widget	PP	26.57	26.74	0.17
Manuf.	PP	120.00	119.95	0.05
Manuf.	PA	90.00	90.04	0.04
Manuf.	PA	90.00	89.91	0.09
OptStand	PP	90.00	90.50	0.50
OptStand	PP	90.00	90.07	0.07

Table 2: True versus estimated angles. The PP measures are between surface normals of the largest planes on the sampled objects. The PA measures are between plane normals and cylinder axes. All measurements are in degrees.

Accuracy

In order to evaluate the accuracy of the hole extraction stage, we commissioned the object shown in Figure 13 to be built. Taking an image of the object, with X and Y sampling of 1mm, and approximating each of the holes in the plane by a circle, we compare the distribution of returned radii against the true value of 7.5mm. Figures 14 and 15 show the error histograms for the radius and separation estimates.

Note that while the model error is on the order of 0.1mm, an order of magnitude less than the sampling rate, there are three holes where surface reflectance problems give a radius error of 1mm. We are investigating the use of combined intensity and range data to solve this problem while retaining the basic accuracy.

Additional accuracy measurements have been less rigorously studied, but Table reftbl:angles shows the true versus measured angles between plane normals and between normals and cylinder axes for the illustrated objects. Table 1 shows true versus estimated cylinder radius for the objects. The accuracy depends largely on the size of the surface patch and the quality of the range data, but the results on the images from our laboratory (the Widget and Manufactured parts) indicate that angular errors of approximately $0.1°$ are representative.

Discussion

Our current implementation of the system uses planar and quadric surfaces, and calculates intersections only for the plane-quadric case. However, both the segmentation and intersection detection depend only on the ability to measure distance from a point to a surface, and so may be readily applied to other surface types. The key to the intersection calculations is primarily in the image processing operations that are applied to the data rather than in the calculations required to perform the intersections themselves.

Contributions

The contributions of this research are in the extension of automatic feature-based reverse-engineering systems to curved surfaces, and in the incorporation of topological constraints into the segmentation process directly. The survey of Hoover *et. al.*[31] found the planar version of this algorithm to produce the most accurate structural results (in terms of pixels correctly labelled) with the second fastest running time, of the systems tested.[23,36,31]. We believe that this is largely attributable to the incorporation of topological constraints.

In comparison with other quadric-based segmentation systems, it is difficult to offer quantitative comparisons as few authors include quantitative results. In addition, because quantitative measures such as angles and hole radii are very dependent on the quality of the range data, they do not distinguish the qualitative differences in accuracy of labellings and topology extraction.

Conclusions

We have presented a system that applies techniques used in computer vision to the problem of CAD model acquisition for reverse engineering. This system does not require operator intervention and is therefore suitable for large projects where many models are to be acquired, such as in legacy inventory modelling. The system produces models which are very close in structure to those produced by a human operator given the same task, and as such are useful in environments where the models are to be further edited after acquisition.

Future work

- We are currently extending the system to operate on full 3D data from view merging. This will allow a more accurate identification of surface adjacency as the problems of self-occlusion will be greatly reduced.
- The system is being extended to output models in AutoCad DXF format so that they can be directly imported into this standard CAD system.
- Extension of the range of surface types allowed to include NURBS surfaces is being investigated, again with the objective of widening the range of applicability of the system.
- Incorporation of intensity information into the edge detection process.
- Incorporation of more domain knowledge to the acquisition system, so that for example planes of "approximately" 90° are converted to right angle joins by the system.

REFERENCES

1 P. J. Besl and R. C. Jain. Segmentation through variable-order surface fitting. *IEEE T-PAMI*, 10(2):167–192, March 1988.

2 M. Potmesil. Generating models of solid objects by matching 3D surface segments. In *Proceedings, IJCAI.* Karlsruhe, Germany, pp 1089-1093, August 1983.

3 B. W. York, A. R. Hanson, and E. M. Riseman. 3D object recognition and matching with B-splines and surface patches. In *Proceedings, IJCAI*, pages 648–651, 1981.

4 J. C. Owen, P. P. J. Sloan, and W. B. Thompson. Interactive feature-based reverse engineering of mechanical parts. In *Proceedings, Image Understanding Workshop*, pages 1115–1124, 1994.

5 H.-Y. Shum, K. Ikeuchi, and R. Reddy. Virtual reality modelling from a sequence of range images. In *Proceedings, DARPA Image Understanding Workshop*, 1995.

6 E. Trucco and R. B. Fisher. Experiments in curvature-based segmentation of range data. *IEEE T-PAMI*, 17(2):177–182, 1995.

7 R. B. Fisher. SMS: A suggestive modelling system for computer vision. *Image and Vision Computing*, 5(2):98–104, May 1986.

8 O. Faugeras and M. Hebert. A 3-d recognition and positioning algorithm using geometric matching between primitive surfaces. *International Journal of Robotics Research*, 5(3):27–52, 1986.

9 W. E. L. Grimson and T. Lozano Pérez. Model-based recognition and localization from sparse range or tactile data. *International Journal of Robotics Research*, 3(3), 1984.

10 G. J. Agin. *Representation and description of curved objects.* PhD thesis, AIM-173, Stanford AI Lab, 1972.

11 R. C. Bolles and M. A. Fischler. A RANSAC-based approach to model fitting and its application to finding cylinders in range data. In *Proceedings, IJCAI*, pages 637–643, 1981.

12 T. Lozano Pérez and W. E. L. Grimson. Finding cylinders in range data. In *Proceedings, ICCV*, pages 202–207, 1987.

13 M. Oshima and Y. Shirai. Object recogntion using three dimensional information. In *Proceedings, IJCAI*, pages 601–606, 1981.

14 T.-J. Fan. *Describing and recognizing 3-D objects using surface properties.* Springer-Verlag, 1990.

15 G. Taubin. Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE T-PAMI*, 13(11):1115–1138, November 1991.

16 A. P. Pentland. Perceptual organisation and the representation of natural form. *Artificial Intelligence*, 28:293–331, 1986.

17 F. Solina and R. Bajcsy. Recovery of parametric models from range images - the case for superquadrics with global deformations. *IEEE T-PAMI*, 12(2):131–147, 1990.

18 P. Boulanger and G. Godin. Multiresolution segmentation of range images based on Bayesian decision-theory. In D. P. Casasent, editor, *Proceedings, Intelligent Robots and Computer Vision XI*, volume 1825, pages 338–350. SPIE, 1992.

19 M. Gay. Segmentation using region merging with edges. In *Proceedings, Alvey Vision Conference*, pages 115–119, 1989.

20 A. Hanson and E. Riseman. The VISIONS image-understanding system. In C. Brown, editor, *Advances in Computer Vision.* Lawrence Erlbaum Assoc, NJ 07642, 1988.

21 G. A. Jones, J. Princen, J. Illingworth, and J. Kittler. Depth data segmentation using robust estimation in a hierarchical data structure. In *Visual Form*, pages 313–321. Plenum Publishing Corp, NY 10013, 1992.

22 G. Medioni and B. Parvin. Segmentation of range images into planar surfaces by split and merge. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 415–417, 1986.

23 P. Flynn. *CAD-Based Computer Vision: Modelling and Recognition Strategies.* PhD thesis, Michigan State University, 1990.

24 J. Han, R. Volz, and T. N. Mudge. Range image segmentation and surface parameter extraction for 3D object recognition of industrial parts. In *Proceedings, ICCV*, pages 380–386, 1987.

25 A. K. Jain and R. Hoffman. Evidence-based recognition of 3-D objects. *IEEE T-PAMI*, 10(6):783ff, November 1988.

26 P. V. C. Hough. *Method and means for recognising complex patterns.*, Dec 1962. U.S. Patent No. 3,069,654.

27 D. S. Chen. A data-driven intermediate level feature extraction algorithm. *IEEE T-PAMI*, 11(7):749–758, July 1989.

28 M. A. Fischler and R. C. Bolles. Perceptual organisation and curve partitioning. *IEEE T-PAMI*, 8(1):100–105, January 1986.

29 R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*, volume 1. Addison-Wesley, 1993.

30 A. W. Fitzgibbon and R. B. Fisher. A buyer's guide to conic fitting. In *Proceedings, BMVC*, pages 513–522, 1995.

31 A. Hoover, G. J. Jean-Baptiste, Xiaoyi Jiang, et al. A comparison of range segmentation algorithms. *(submitted to IEEE T-PAMI)*, 1995.

32 A. W. Fitzgibbon and R. B. Fisher. Suggestive modeling for machine vision. In *Proceedings, Curves and Surfaces in Computer Vision and Graphics III*, Boston, MA, November 1992. SPIE.

33 A. W. Fitzgibbon and R. B. Fisher. Lack-of-fit detection using the run-distribution test. In *Proceedings, ECCV*, page xx, Stockholm, 1994.

34 J. E. Dennis, D. M. Gay, and R. E. Welsch. An adaptive nonlinear least-squares algorithm. *ACM Trans. Math. Software*, 7(3):348–368, 1981.

35 D. K. Naidu and R. B. Fisher. A comparative analysis of algorithms for determining the position of a stripe to sub-pixel accuracy. In *Proceedings, BMVC*, pages 217–225, 1991.

36 X. Y. Jiang and H. Bunke. Fast segmentation of range images by scan line grouping. *Machine Vision and Applications*, 7(2):115–122, 1994.