Constructing High-Precision Geometric Models from Sensed Position Data

William B. Thompson, H. James de St. Germain, Thomas C. Henderson, and **Jonathan C. Owen**
Department of Computer Science
University of Utah
Salt Lake City, UT 84112
`http://www.cs.utah.edu/projects/robot/`

Abstract

The construction of geometric scene models from sensed data is a fundamental task in image understanding. The creation of such models is essential in computer vision systems for applications ranging from reconnaissance to robot navigation. Automated model construction using visual techniques can also provide important support for systems in manufacturing and simulation. Vision-based geometric modeling has typically emphasized generality, with little explicit concern for the accuracy of the models produced. When high-precision is required, domain-specific representations and processing can provide significant advantages. We describe such a system for reverse engineering mechanical parts as part of maintenance and repair activities. Many of the key components of this system are also applicable to the generation of models for simulation and virtual environment applications.

1 Introduction

Methods for the construction of three-dimensional models of viewed objects and surfaces are a critical component of many image understanding systems. *Inverse optics* – the use of photometry and perspective to determine the scene properties likely to have generated a particular image – has been a dominate theme in computer vision since the early work of David Marr and Berthold Horn. With the advent of sensors capable of directly measuring distance to visible surface points, it has been possible to build models of surface and object geometry directly, without the confounding effects of photometry.

In this paper, we deal with the problem of constructing geometric models from sensed data about the 3–D position of surface points. While much prior work has been done in this area, few other researchers have specifically addressed the issue of modeling accuracy. Our approach is capable of achieving highly accurate descriptions of geometry through the use of two related techniques. Instead of using geometric representations based on generic modeling primitives, as is common in most computer vision methods, we use primitives natural to the task for which modeling is being done. Because these primitives are tailored to the task at hand, they can usually describe the relevant geometry with far fewer parameters than more general representations. This reduction in representational degrees-of-freedom translates directly into reduced sensitivity to sensor noise. In addition, the models that are produced are often in a form that is much easier to use by whatever domain-specific processing that follows. The second benefit of this approach is that domain-specific modeling primitives allow for the natural introduction of domain-specific geometric constraints describing the relationships that should hold between various aspects of the model. This provides further immunity to noise and in some circumstances can allow the reconstruction of key properties of the geometry in the absence of any relevant data.

The application addressed involves the construction of CAD models for existing, mechanical parts. This is important when existing parts must be reverse engineered to support the production of additional spares or to modify a design for which a working CAD model is not available. Similar techniques are likely to be relevant to the creation of geometric models to support a wide variety of simulation applications.

2 The Need for Reverse Engineering to Support DOD Maintenance and Repair Activities

DOD must maintain a large quantity of legacy hardware, much of it many decades old. The life cycle of many of these hardware systems will be further stretched as budgetary constraints force reductions in future appropriations. As a result, spare parts inventories are frequently exhausted well before de-commissioning of the relevant pieces of equipment. Additional spares are often difficult or impossible to obtain from the original suppliers of the equipment. Some of these suppliers are out of business. Others have migrated to non-defense businesses. Still others are unable or unwilling to provide custom manufacturing runs of spare parts at affordable prices and in a timely manner. A substantial portion of the contracts under which DOD hardware has been acquired have failed to re-

This work was supported by the Advanced Research Projects agency under Army Research Office grant number DAAH04-93-G-0420 and by the Office of Naval Research under grant N00014-93-1-0950.

quire documentation sufficient for another supplier to replicate needed parts.

The increasing shortage of spare parts has lead the DOD repair and maintenance community to identify reverse/re-engineering (RE) of existing parts as a pressing technical issue. Reverse/re-engineering allows the manufacture of new spare parts based on an analysis of existing parts, without the need for original CAD models or other documentation. While civilian applications of reverse/re-engineering exist, the civilian sector is unlikely to either provide or drive the development of RE systems sufficient to satisfy DOD needs. Far more civilian manufacturing is done in house than is the case with the military. When civilian manufacturers out-source the production of parts, they typically specify their own designs or require adequate documentation from their suppliers. For example, Caterpillar reportedly maintains sufficient documentation to be able to manufacture additional parts for any vehicle that it has ever delivered.

The Navy's Lifecycle in Service Repair Networking Center (LINC) / Repair Technology (REPTECH) Program has taken a lead in DOD activities relating to reverse/re-engineering. LINC/REPTECH is developing a statement of needs for reverse engineering within DOD, collecting information on efforts to date, and outlining technical project areas that require additional research and development. Many other groups within the Navy and the other services are involved in this effort or similar activities of their own.

RE needs to be based on sensed properties of existing parts. Most of the current work on RE systems, however, approaches the problem from a purely CAD/CAM perspective. IU techniques can provide better geometric model building tools and sensing strategies, improving the accuracy and utility of RE systems. This is particularly true for IU methods that are closely integrated with existing CAD/CAM technologies.

3 Reverse Engineering of Mechanical Parts

Reverse engineering techniques can be used to create CAD models of a part based on sensed data acquired using three-dimensional position digitization techniques. Part-to-CAD reverse engineering allows up to date NC fabrication plus easier modification of the design than would otherwise be possible. A number of vendors provide hardware and/or software to support this application [Broacha and Young, 1995]. Geometry is sensed using either contact Coordinate Measuring Machines (CMMs) or non-contact laser scanners produced by companies such as Cyberware, Digibotics, and Laser Designs. Several CAD vendors have tools for converting this data into their internal geometric representations. Manufacturers of position scanning devices and several third-party vendors have software that will convert scanned position data into either STL format or a collection of IGES splined surfaces for importation into any 3–D CAD package.

Two significant impediments are keeping current generation reverse engineering tools from more wide-spread usage:

- *Accuracy and tolerances.* Few if any of the commercially available reverse engineering tools are designed to optimize the accuracy of the models that are produced. Within the research community studying problems of geometric reconstruction, if accuracy is discussed at all it is in the context of the statistical validity of estimators and not their quantitative accuracy. No one has yet addressed the issue of determining appropriate tolerances to associate with a recovered model.

- *Usability.* Commercial reverse engineering packages provide models in a form that can be imported into CAD packages, but much of the theoretically available functionality is lost and substantial hand processing is usually required before the models can be used. Representations used for geometric reconstruction in computer vision are seldom appropriate for any end-user application.

The computer vision community is most familiar with sensed position information represented as a *range image*. As with more conventional optical images, range images are organized into two-dimensional arrays, in which each array element (pixel) corresponds to a ray in space defined by some projection model. The value of the pixel is the distance to the nearest surface point along that ray. Distance can be determined by triangulation, using a structured light approach, or by time-of-flight measurements [Jarvis, 1983, Besl, 1988].

Most often, range image sensors employ conventional imaging technology and are governed by the same projective models as the perspective transformations which describe that vast majority of optical imagers. Some use alternative projections, such as the cylindrical scanners made by Cyberware. In any event, the nature of projection that occurs is such that many of the low-level techniques used to analyze optical images have direct analogs for range images. In particular, there is a topological association between surface points corresponding to adjacent pixels. In particular, except in the case of surface boundaries, adjacent pixels provide information about the same surface. This property is used to help organize the information in range images, since fairly standard segmentation operations can be used to partition range images into regions corresponding to surfaces. Once this is accomplished, adjacent pixels correspond to adjacent surface points.

Occlusion effects appear in range images, just as they do in optical images. The detection of occlusion, however, is easier, since it is usually associated with discontinuities in range and need not be inferred from cues such as intensity discontinuities. Occlusion is still a critical problem when creating a geometric model based on a range image, since there is no way to know the geometric properties of occluded surfaces.

There are two approaches to dealing with occlusion when sensing position. The first involves combining multiple range images into a composite structure [Chen and Medioni, 1992, Chen and Medioni, 1993, Turk and Levoy, 1994], though registering multiple views can be a significant problem. In addition, the viewer-centered nature of range images can result in biased estimates of object shape unless care is taken [Bolle and Vemuri, 1991]. The second approach uses some sort of active sensing methodology to position sensors so as to avoid as much as possible one surface hiding another. For example, the DIGIBOT-II scanner moves a triangulation ranging device using an x-y transport, while the object being scanned is moved using a rotating table. The extra degree-of-freedom in positioning the sensor with respect to the object can be used to "look behind" some occlusions while also optimizing the position of the sensor with respect to the orientation of the surface point being measured. The position data points returned by such active sensing systems no longer have the same topological relationships to one another as is the case with range images, and are often described as *point cloud* data. Furthermore, occlusion still occurs, only now is harder to detect since there is no simple form of edge detection that can be applied. Fortunately, methods exist for organizing point cloud data in a manner that highlights adjacency relationships [Hoppe *et al.*, 1992, Hoppe *et al.*, 1993, Hoppe *et al.*, 1994].

A wide variety of geometric primitives have been proposed for use in creating geometric models from sensed data. Surface primitives in existing modeling systems range from simple planes and cylinders [Chivate and Jablokow, 1993] to piecewise smooth surface parametric surface patches [Sarkar and Menq, 1991, Piegl, 1991, Lounsbery *et al.*, 1992]. Volumetric primitives include generalized cylinders [Binford, 1971, Nevatia and Binford, 1977, Brooks, 1977, Marr, 1982] and parametric shapes such as superquadrics [Pentland, 1986]. CAD models obtained from design systems have been used as a geometric representation for various computer vision algorithms [Ikeuchi and Kanade, 1988, Hansen and Henderson, 1989], but little work has been done within the computer vision community on the use of CAD models to represent the geometry implicit in sensed data.

Since geometric models for complex shapes are almost always described in terms of multiple primitives, a decision is required as to what data points should be considered to be part of each primitive. Most other approaches to dealing with position data use some form of bottom up segmentation procedure [Besl and Jain, 1988, Suk and Bhandarkar, 1992]. Faces on polyhedral objects are found with plane fitting techniques. Curved faces are found using grouping operations which combine collections of points into surfaces, followed by detection of lines of orientational discontinuities. However, few mechanical parts are polyhedra. For curved surfaces, segmentation based on orientational discontinuities is problematic due to noise effects in most range sensors, which produce substantial local variations in surface normals. This problem is particularly acute at surface boundaries, where reliable information is essential for bottom-up processing.

4 Feature-Based Modeling

The modeling primitives used to describe a geometric shape should be appropriate to the task for which the modeling is being performed. In our work, we are primarily concerned with man-made *artifacts* which are designed and constructed by people to serve some purpose. Modern design systems are starting to exploit the concept of *design features*, which are hierarchical structures closely related to design intent. Design features are desirable because they form the basis of a natural and compact representational language. In addition, because design features capture aspects of intent, they often facilitate the process planning that is needed to translate the design into a sequence of actions for constructing the object.

While originally developed to support creation of new entities, design features can also be used as geometric primitives for the construction of models of existing entities. Three advantages derive from this approach:

- *Appropriateness.*

  Models described using the same primitives as used in designing the object being modeled will clearly be adequate to describe its shape.

- *Ease of use.*

  The same tools originally intended to support design activities can operate on the constructed model.

- *Reduced need for complete, robust geometric computations.*

  Substantial effort is involved in converting a collection of surface patches obtained by fitting to scanned data into a form usable by a solid modeler. If a feature-based design system exists, it is far easier to generate a feature-based description and then let the design system generate a topologically correct B-rep solid model.

- *Accuracy.*

  Non-contact position digitizers are subject to errors which can exceed the tolerances needed in modeling many objects. The local smoothing that is implicit in methods based on fitting surface patches to position data may not be optimal for reducing this sensing noise. The use of design features as primitives can substantially increase the accuracy of the generated models.

We have described elsewhere a program called REFAB, which uses design features as modeling primitives in a reverse engineering system for mechanical parts. [Owen *et al.*, 1994, Thompson *et al.*, under review]. REFAB operates in an interactive manner. The user specifies the types of design features present and the approximate location of each feature in the object. REFAB deals with the determination of precise, quantitative parameterization of each feature. In REFAB, the use of design features tailored to NC machining provides a natural interface to the end user, significantly simplifies aspects of the model building, and leads to methods for obtaining models that are more accurate than would otherwise be possible.

Modeling accuracy depends on effective use of properties that distinguish the geometry of interest from effects due to sensor noise. The current version of REFAB uses three such types of information: geometric primitives that are capable of describing object geometry with a minimal number of degrees-of-freedom, information about the most common ways in which designers use these primitives, and constraints about how different primitives interact in a properly designed object.

4.1 Task-Specific Geometric Primitives

When fitting models to noisy sensor sensor data, the best noise immunity is usually obtained by using the modeling primitives with the fewest degrees-of-freedom required to describe the shapes of interest. To take a simple example, consider the problem of fitting a closed contour to a set of 2–D points. If it is known that the original shape is a conic section, then it is far better to fit an ellipse (DOF $= 5$) than to use a spline function (DOF $\gg 5$). Noise is averaged out much more effectively when fitting the ellipse than with a spline, which will wiggle around to fit the observed data in ways that can't correspond to the actual shape.

REFAB uses the same feature set as employed in the Alpha_1 CAD/CAM system [Drake and Sela, 1989]. This allows DOF reduction in two ways. The most obvious advantage is that most of the Alpha_1 features are highly structured, consisting of various types of holes, pockets, and the like. The second advantage is that the most common features are $2\frac{1}{2}$–D in nature. Effectively, they are extrusions of planar curves. Such features can be modeled in two steps. The first determines the orientation of the feature and then projects all data points relevant to modeling the feature into a plane defined by that orientation. The second step determines the parameters describing the profile of the feature using 2–D analysis techniques.

While we have demonstrated the advantages of domain-specific geometric primitives in reverse engineering applications, they are relevant to many other sensor-based modeling tasks as well. In particular, man-made structures are almost always explicitly or implicitly designed using a limited set of primitives. Using the same geometric representations as the basis of recovering models from sensed data simultaneously provides a natural user interface and improved modeling accuracy.

4.2 Task-Specific Modeling "Hints"

Human designers, for reasons of both habit and conveniency, seldom use the full descriptive power of the representational languages available to them. For example, while most CAD systems allow the specification of profiles in terms of arbitrary closed contours, the profiles associated with the inside pocket in Figure 2 and the outer profile side in Figure 5 are made up only of line and arc segments. This simplification is pervasive in machined parts.

Effects such as this can provide "hints" that further reduce the degrees-of-freedom needed to adequately model the underlying geometry. When REFAB creates geometric models from points lying along a 2–D profile, sequences of points which can accurately be approximated by line segments [Nevatia and Babu, 1980] are identified first. The remaining points correspond to curved portions of the profile. An at-

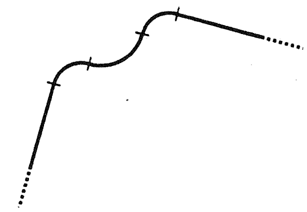

Figure 1: Tangent matching triple arc.

tempt is made to fit each of these segments using one, two, or three constant radius arcs of alternating curvature. Multiple arcs are constrained to maintain tangent continuity from one to the next. The triple arc case is further constrained such that the outer two arcs have the same curvature and arc length (Figure 1, corresponding to a boss-like sub-feature often found in parts such as shown in Figure 2. If a curved segment cannot be accurately fit with one of the arc constructs, a general Bezier curve is used instead.

Design practices which underlie constraints of this sort are highly domain-specific. Their discovery often requires a detailed understanding of what skilled designers actually do on real problems. The potential benefits are significant, however. Constraints arising out of design practice provide additional immunity to sensing noise. Often equally important, they provide domain-specific improvements in modeling accuracy. In applications such as manufacturing and the simulation of indoor environments, it is important that models be accurate not only in terms of geometric position but also in terms of local surface shape [Thompson *et al.*, 1996]: Flat surfaces should be modeled as flat. Smoothly curving surfaces should be modeled as smoothly curving. Right angles should be preserved as exactly as possible.

4.3 Task-Specific Modeling Constraints

For many man-made objects, common relationships exist between design features. In the object shown in Figure 2, for example, it is likely that linear portions of the profile defining the pocket are parallel with the outer profile. Less certain, but still plausible, is that the three triple arcs near each hole have a common radius for their central and surrounding arcs. In the object shown in Figure 5, it is reasonable to presume that the large hole and the circular arc at the upper left are coaxial. These relationships provide further constraints on the possible geometry of the object. The result is a further reduction in modeling degrees-of-freedom and thus additional improvements in modeling accuracy in the presence of sensing noise.

While some relationships such as these can be inferred from observation of the geometry alone, most involve a sophisticated understanding of the task for which the object was designed and the intended function of the various shapes involved. As with the choice of which design features to use in a given modeling operation, we adopt an interactive approach. An initial model is first generated. The end user of the system is then able to suggest possible constraints between parameters of different design features making up the description of a complete part or assembly. A new model, consistent with these constraints, is then generated. The error in fitting the data will increase due to the added constraints in the optimization process. However, if that increase is small there is good reason to believe that the constraints are valid. The result is that while there will be a small decrease in the accuracy with which the generated model fits the data, there is likely to be an *increase* in the accuracy with which the generated model fits the model used initially to create the part.

5 Segmentation and Refinement

Model generation from sensed data requires that the data be partitioned into subsets corresponding to individual geometric primitives before parameters of each primitive are determined. An interactive approach, together with the use of design features as primitives, can avoid most of the difficulties associated with traditional data partitioning methods by using a robust, top-down segmentation technique.

In REFAB, the user starts out by specifying a type and approximate location for each feature. Given this rough feature parameterization, the system selects those position points that are close to the surface of the approximated feature in *both* distance and orientation. The combination gives a much better indication of points that are really part of the feature than would either property alone. For example, consider the problem of finding those sensed points on the wall of a drilled hole. Clearly, we want to consider only those points near the expected location of the hole. Using only a distance check, however, will inevitably include some points on the surface through which the hole was drilled, near the rim of the hole. An orientation check quickly discards these points. Additional improvements are obtained by further restricting the distance check, based on per-feature information about where position data is most likely to be accurate. In the case of the hole, data near the rim and deep within the hole is most suspect.

The initial segmentation is done using a large tolerance for distance and orientation, but only using those parts of the user-specified model which are expected to yield the best sensed data. The feature is then re-fit to the segmented data, resulting in a refinement to that part of the model. As the estimates of feature parameters are improved, the position data can be re-segmented using tighter tolerances on distance and orientation, while reducing or eliminating the restrictions on which parts of the feature surface to consider. Additional improvements come from the use of robust fitting techniques [Rousseeuw and Leroy, 1987] which are able to ignore extraneous data points arising from whatever segmentation errors do occur.

6 Experimental Results

In order to quantitatively evaluate the accuracy of the models obtainable using the feature-based modeling approach, we started with parts from the "Hard-Copy Benchmark" [Thompson and Owen, 1994, Owen *et al.*, 1994, Owen *et al.*, 1996] for which we had access to the original CAD models. Instances of these parts were carefully machined out of aluminum using a 3-axis NC mill. Surface points on the parts were measured using a non-contact laser digitizer. New CAD models for each part were generated using the REFAB system. Finally, the geometric differences between the original and recovered models were computed [Thompson *et al.*, 1996].

Position data was acquired with a DIGIBOT II laser position digitizer. The DIGIBOT II has a nominal measurement accuracy of ± 50 microns ($1\ \sigma$) under optimal conditions. In practice we have observed accuracies on the order of $\pm$50-300 microns, depending on the nature and shape of the surface at that point. The manufacturing processes used to produce the test objects can achieve precisions on the order of $\pm$2-10 microns for hole and bore spacings. Cutting accuracy, which is more relevant here, is typically on the order of $\pm$50-250 microns depending on the feature being cut and the tool being used. Thus, the overall variability of the sensed data relative to the original model of the parts is on the order of $\pm$100-550 microns.

To remove specularities that cause problems for most current range finding systems, parts were sprayed with a white powder, which left a thin, talcum-like coating. Multiple scans were taken of each part and transformed into common point-cloud data sets, using a registration procedure similar to [Shum *et al.*, 1994]. The reverse engineering of the shock plate involved the use of 143,140 3–D points. 44,578 points were used for the steering arm.

Figures 2 and 5 show two of the actual parts used for testing. Figures 3 and 6 are wire frame drawings generated from the reverse engineered CAD models produced by REFAB for the two parts, shown as exploded views to emphasize the feature-based nature of the representation. Figures 4 and 7 are new parts made from these models, demonstrating an end-to-end reverse engineering process.

Table 1 shows the quantitative deviation between reconstructed models and the original CAD models for the outer and inner profiles of the object shown in Figure 2 and the outer profile and large hole for the object shown in Figure 5. Due to their shape and the sensor noise associated with inter-reflections in the pocket and hole, these contours are the hardest parts of the objects to model accurately. As a result, they provide the best test of the advantages of domain-specific modeling primitives and geometric constraints. Pairs of average (RMS) and maximum deviation are shown. The first pair of numbers shows the accuracy of a simple spline fit to the data, using standard smoothing and resampling techniques before fitting a Bezier curve. The second set of RMS/max deviations are for a reconstructed model produced by using design features as geometric primitives and employing when possible modeling hints taken from common design practice. For the third set of values, global modeling constraints were added to the reconstruction process.

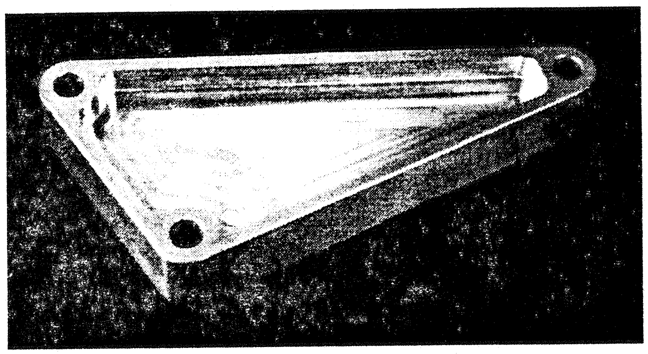

Figure 2: Shock plate: original part.

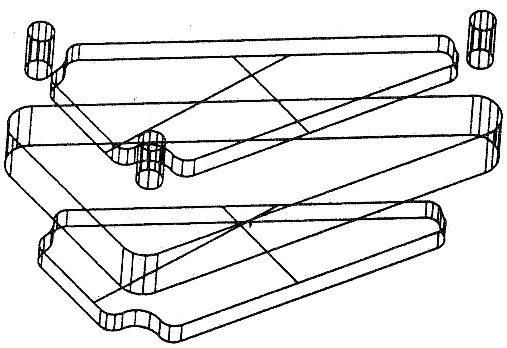

Figure 3: Exploded view of the features making up the reverse engineered shock plate.

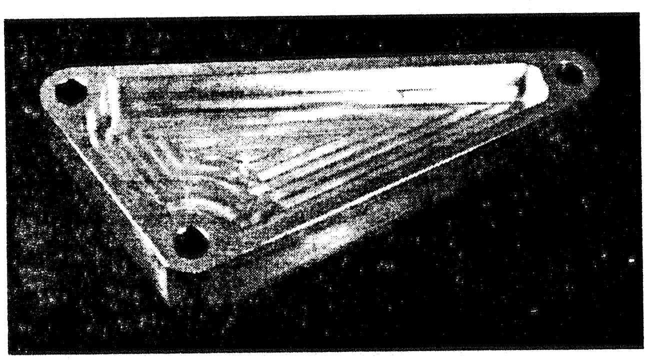

Figure 4: Shock plate: reverse engineered part.

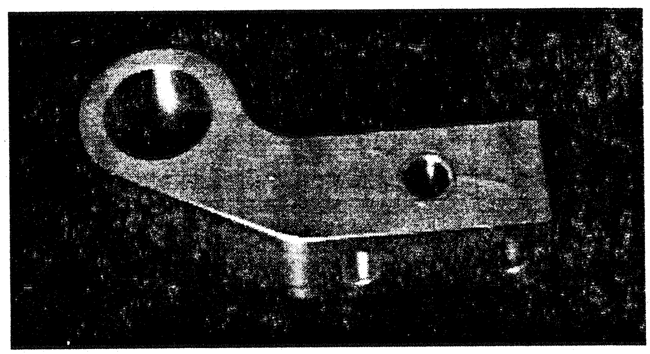

Figure 5: Steering arm: original part.

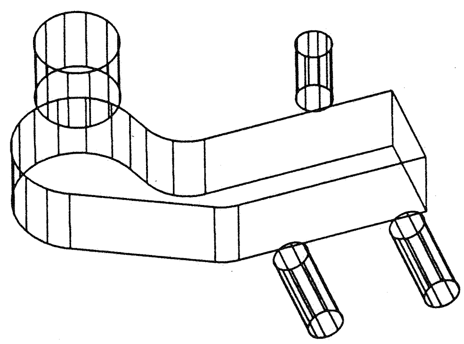

Figure 6: Exploded view of the features making up the reverse engineered steering arm.

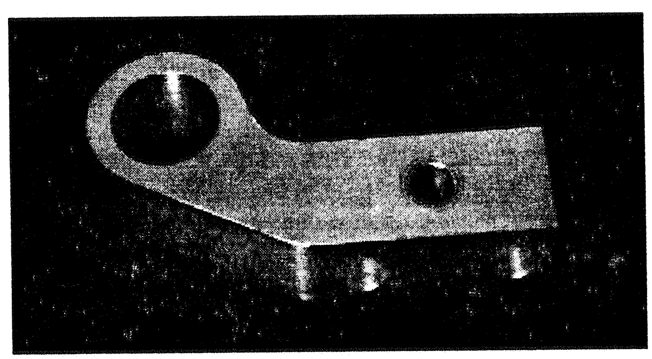

Figure 7: Steering arm: reverse engineered part.

Error wrt. model	spline fit		local constraints		global constraints	
	RMS	max	RMS	max	RMS	max
Shock plate						
outer profile	97	243	46	121	27	81
inner profile	185	784	97	283	59	199
Steering arm						
outer profile	95	828	70	125	55	117
large hole			115	164	61	98

Table 1: Difference between recovered model and original model (in microns).

Error wrt. data	spline fit		local constraints		global constraints	
	RMS	max	RMS	max	RMS	max
Shock plate						
outer profile	74	511	82	513	76	466
inner profile	101	613	110	573	105	600
Steering arm						
outer profile	93	743	93	741	89	783
large hole			160	1,263	171	1,345

Table 2: Difference between recovered model and data (in microns).

Shock plate parametric errors		local constraints		global constraints	
		radius	center	radius	center
triple 1	arc 1	316	533	162	305
	arc 2	101	56	81	23
	arc 3	316	515	162	299
triple 2	arc 1	50	96	98	50
	arc 2	697	692	81	10
	arc 3	50	74	98	61
triple 3	arc 1	548	530	0	199
	arc 2	32,626	32,500	81	116
	arc 3	548	404	0	199

Table 3: Parametric difference between recovered model and original model (in microns).

The accuracy results in Table 1 for the spline fit are typical of what can be achieved with standard modeling techniques which do not take advantage of any special information about the possible shapes. Substantial increases in both average and worst-case precision are achieved through the use of appropriate geometric primitives and local, domain-specific constraints on those primitives (second pair of numbers in the table). Adding global constraints specifying how different geometric primitives relate to one another further improves accuracy.

The results shown in Table 1 were obtainable because we tested the modeling system with objects of known geometry. Table 2 shows the effect of the different modeling techniques on how well the data itself is approximated. In general, the stronger the constraints, the worse the precision with which the reconstructed model represents the data. This points out the important difference between evaluating modeling techniques based only on how well they approximate data versus evaluating them with respect to the underlying geometry which it is desirable to reconstruct [Thompson *et al.*, 1996].

Table 3 shows the parametric accuracy of the two constrained fitting methods applied to each of the arcs making up the triple arcs in the corners of the pocket of the object shown in Figure 2. These are the hardest sub-features in the test objects to model accurately, due both to inter-reflection effects and the fact that the arc lengths are quite small. The use of global constraints effectively reduces the degrees-of-freedom by a factor of three, resulting in substantial improvements in parametric accuracy.

7 Conclusions

Automated vision systems can aid in the construction of geometric models for a wide variety of important tasks. Many of these tasks, however, require modeling precision well beyond that obtainable with conventional computer vision methods. Our approach uses a combination of domain-specific modeling primitives and domain-specific constraints to achieve accuracies well within the intrinsic noise of the sensor used to acquire the position data on which the models are based. The approach has been demonstrated using a quantitative comparison between the recovered models and the true geometry.

References

[Besl and Jain, 1988] P. Besl and R. Jain. Segmentation through variable-order surface fitting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 167–192, March 1988.

[Besl, 1988] P. J. Besl. Active, optical range imageing sensors. *Machine Vision and Applications*, pages 127–152, 1988.

[Binford, 1971] T. O. Binford. Visual perception by a computer. In *Proc. IEEE Conference on Systems and Controls*, 1971.

[Bolle and Vemuri, 1991] R. M. Bolle and B. C. Vemuri. On three-dimensional surface reconstruction methods. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 1–13, January 1991.

[Broacha and Young, 1995] R. Broacha and M. Young. Getting from points to products. *Computer-Aided Engineering*, July 1995.

[Brooks, 1977] R. A. Brooks. Symbolic reasoning among 3D models and 2D images. *Artificial Intelligence*, 8:77–98, 1977.

[Chen and Medioni, 1992] Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *International Journal of Image and Vision Computing*, 10(3):145–155, April 1992.

[Chen and Medioni, 1993] Y. Chen and G. Medioni. Integrating multiple range images using triangulation. In *Proc. ARPA Image Understanding Workshop*, pages 951–958, April 1993.

[Chivate and Jablokow, 1993] P. N. Chivate and A. G. Jablokow. Solid-model generation from measured point data. *Computer-Aided Design*, September 1993.

[Drake and Sela, 1989] S. Drake and S. Sela. A foundation for features. *Mechanical Engineering*, 111(1), January 1989.

[Hansen and Henderson, 1989] C. Hansen and T. C. Henderson. CAGD-based computer vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 1181–1193, 1989.

[Hoppe *et al.*, 1992] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. In *Computer Graphics, SIGGRAPH '92*, volume 26, July 1992.

[Hoppe *et al.*, 1993] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimization. In *Computer Graphics, SIGGRAPH '93*, volume 27, August 1993.

[Hoppe *et al.*, 1994] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Piecewise smooth surface reconstruction. In *Computer Graphics, SIGGRAPH '94*, volume 28, July 1994.

[Ikeuchi and Kanade, 1988] K. Ikeuchi and T. Kanade. Towards automatic generation of object recognition programs. *Proc. IEEE*, 76:1016–1035, 1988.

[Jarvis, 1983] R. A. Jarvis. A perspective on range finding techniques for computer vision. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 122–139, March 1983.

[Lounsbery *et al.*, 1992] M. Lounsbery, S. Mann, and T. DeRose. Parametric surface interpolation. *IEEE Computer Graphics and Applications*, September 1992.

[Marr, 1982] D. A. Marr. *Vision*. W. H. Freeman, San Francisco, 1982.

[Nevatia and Babu, 1980] R. Nevatia and K. Babu. Linear feature extraction and description. *Computer Graphics and Image Processing*, 13:257–269, 1980.

[Nevatia and Binford, 1977] R. Nevatia and T. O. Binford. Description and recognition of curved objects. *Artificial Intelligence*, 8:77–98, 1977.

[Owen *et al.*, 1994] J. C. Owen, P. P. J. Sloan, and W. B. Thompson. Interactive feature-based reverse engineering of mechanical parts. In *Proc. ARPA Image Understanding Workshop*, pages 1115–1124, November 1994.

[Owen *et al.*, 1996] J. C. Owen, H. J. de St. Germain, S. Stark, and W. B. Thompson. Calibrated imagery for quantitative evaluation of IU pose-estimation and stereo algorithms. In *Proc. ARPA Image Understanding Workshop*, February 1996.

[Pentland, 1986] A. P. Pentland. Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28:29–73, 1986.

[Piegl, 1991] L. Piegl. On NURBS: A survey. *IEEE Computer Graphics & Applications*, pages 55–70, January 1991.

[Rousseeuw and Leroy, 1987] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, New York, 1987.

[Sarkar and Menq, 1991] B. Sarkar and C. H. Menq. Smooth-surface approximation and reverse engineering. *Computer Aided Design*, 23(9):623–628, November 1991.

[Shum *et al.*, 1994] H. Y. Shum, K. Ikeuchi, and R. Reddy. Virtual reality modeling from a sequence of range images. In *Proc. ARPA Image Understanding Workshop*, pages 1189–1198, November 1994.

[Suk and Bhandarkar, 1992] M. Suk and S. Bhandarkar. *Three Dimensional Object Recognition from Range Images*. Springer-Verlag, New York, 1992.

[Thompson and Owen, 1994] W. B. Thompson and J. C. Owen. "Hard-copy" benchmark suite for image understanding in manufacturing. In *Proc. ARPA Image Understanding Workshop*, pages 221–227, November 1994.

[Thompson *et al.*, 1996] W. B. Thompson, R. F. Riesenfeld, and J. C. Owen. Determining the similarity of geometric models. In *Proc. ARPA Image Understanding Workshop*, February 1996.

[Thompson *et al.*, under review] W. B. Thompson, J. C. Owen, and H. J. de St. Germain. Feature-based reverse engineering of mechanical parts. *Computer-Aided Design*, (under review).

[Turk and Levoy, 1994] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *Computer Graphics, SIGGRAPH '94*, volume 28, pages 311–318, July 1994.