Feature-Based Reverse Engineering of Mechanical Parts

William B. Thompson, Jonathan C. Owen,
and H. James de St. Germain

UUCS-95-010

Department of Computer Science
University of Utah
Salt Lake City, UT 84112 USA

August 1, 1995

Abstract

Reverse engineering of mechanical parts requires extraction of information about an instance of a particular part sufficient to replicate the part using appropriate manufacturing techniques. This is important in a wide variety of situations, since functioning CAD models are often unavailable or unusable for parts which must be duplicated or modified. Computer vision techniques applied to 3–D data acquired using non-contact, three-dimensional position digitizers have the potential for significantly aiding the process. Serious challenges must be overcome, however, if sufficient accuracy is to be obtained and if models produced from sensed data are truly useful for manufacturing operations. This paper describes a prototype of a reverse engineering system which uses geometric representations natural to the manufacturing process. The system is interactive, which improves performance and allows for human entry of information that cannot be acquired from sensed data alone.

Feature-Based Reverse Engineering of Mechanical Parts

William B. Thompson, Jonathan C. Owen, and H. James de St. Germain

Department of Computer Science
University of Utah
Salt Lake City, UT 84112

University of Utah Dept. of Computer Science Tech. Report UUCS-95-010, August 1, 1995.

Abstract

Reverse engineering of mechanical parts requires extraction of information about an instance of a particular part sufficient to replicate the part using appropriate manufacturing techniques. This is important in a wide variety of situations, since functional CAD models are often unavailable or unusable for parts which must be duplicated or modified. Computer vision techniques applied to 3–D data acquired using non-contact, three-dimensional position digitizers have the potential for significantly aiding the process. Serious challenges must be overcome, however, if sufficient accuracy is to be obtained and if models produced from sensed data are truly useful for manufacturing operations. This paper describes a prototype of a reverse engineering system which uses geometric representations natural to the manufacturing process. The system is interactive, which improves performance and allows for human entry of information that cannot be acquired from sensed data alone.

1 Introduction.

There is increasing interest in producing geometric models suitable for computer processing through the use of three-dimensional position digitization techniques. Three application areas are currently commercially significant: computer graphics, medicine, and CAD/CAM. Three-dimensional digitization is being used to create renderable models from actual objects for applications ranging from advertising to feature film animation. Medical uses include prosthetics, orthotics, and planning for reconstructive and cosmetic surgery. In manufacturing, 3–D digitization allows for a form of reverse CAD in which CAD models are created from existing parts, rather than creating the CAD models on a design system and then using the models to fabricate a part.

Three-dimensional digitization for CAD/CAM is used for two rather distinct types of tasks. Many designers prefer to work in clay rather than on CAD workstations when dealing with objects having complex, sculptured surfaces. Such objects range from automobile body panels to telephone hand sets. Once the clay model is built, it must be digitized for analysis, simulation, design of mechanical detail, and production. The second type of manufacturing problem involving models produced by 3–D digitization is the *reverse engineering* of existing parts. CAD models of existing parts may be unavailable because other design techniques

---

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.

were used to create the part, the original CAD model is unusable for whatever reason, or use of the original CAD model is prohibited due to legal, proprietary, or technical reasons. This is a particular problem for long life cycle systems for which spare part inventories have been exhausted and original suppliers are unable or unwilling to provide custom manufacturing runs of spare parts at affordable prices and in a timely manner. Part-to-CAD reverse engineering allows up to date NC fabrication plus easier modification of the design than would otherwise be possible. Successful instances include everything from sporting goods to aircraft parts.

Reverse engineering of solid objects traces its roots back to the pantograph, which uses a mechanical linkage to duplicate arbitrary geometric shapes at any predetermined scale. Copy lathes and mills are more contemporary and automated versions of the pantograph. In a copy lathe, a mechanical stylus is moved along a template specifying a 1–D profile. The position of the cutter is adjusted based on this template, producing a revolute object with the same profile. A copy mill typically moves a stylus over a surface, using the height of the surface to set the z-axis in a 3-axis mill, thus making a copy of the original object.[1] Several vendors have produced copy mills which use non-contact sensors. These systems have the added advantage of storing the sensed profile, so that an object can be duplicated many times without repeated scanning.

Copy lathes and mills duplicate a physical part without producing any intermediate model of the geometry of the part, other than stylus position or 3–D points acquired with a non-contact sensor. While some can produce NC code capable of driving other lathes and mills, none can produce a CAD model of an existing part. Such models are desirable for a number of reasons. Modifications to the part cannot easily be done at the level of NC code. Even if the part is to be duplicated as is, refixturing and hidden concavities often lead to situations in which multiple scans of an object's shape must be combined into a single, consistent representation. Some shape properties such as deep holes will not be accurately measured by either mechanical styli or non-contact sensors. Finally, there are situations such as occur in the die and mold industry where, depending on production requirements, designs are often modified on the shop floor. The original design must then be updated to match the manufactured part.

The most straightforward approach to generating a reverse engineered model of a mechanical part involves a designer or engineer making measurements using traditional devices such as calipers and gauges and entering the results into a standard CAD system. When high precision is required, contact coordinate measuring machines (CMMs) are often used. Positional accuracy on the order of ±3 microns locally and ±14 microns corner to corner is possible, but sensing of a large number of points is extremely slow and expensive damage can be done if the probe is not maneuvered towards the object along an appropriate path. More recently, non-contact CMMs produced by companies such as Cyberware, Digibotics, and Laser Designs have significantly increased the speed with which data can be collected. These devices project a spot or line of light and use triangulation to determine range. While less accurate than contact CMMs, the best are capable of positional accuracy exceeding ±50 microns. Non-optimal surface properties can degrade this, while deep concavities, discontinuous surface orientation, surface geometries forcing oblique viewing angles, or outright occlusion will cause data to be missing entirely. For comparison, commonly available NC milling machines can achieve precisions of ±2-10 microns for hole and bore spacings and can produce cutting accuracies on the order of ±50-250 microns depending on the feature being cut and the tool being used, though special measures can be used to obtain higher precision.

---

[1] We are all familiar with a ubiquitous 2-axis copy mill – the key duplicating machine.

To date, reverse engineering of mechanical parts using automatically acquired three-dimensional position data has used rather unsophisticated geometric models. Often, a digitizer is moved along parallel scanning paths and NC code generated to move a cutter along the same 3–D path. In effect, no model other than the raw scan data is used, though preprocessing to remove noisy data points, align scan lines from multiple scans, etc., is usually necessary. Several software *surfacing* packages, including Imageware Surfacer™, Parametric Pro/SCAN-TOOLS™, and Cyberware Cyserf™, have recently become available. These packages fit spline patches to raw data points, allowing the importation of of the surfaces into CAD systems. Tool paths can then be generated to move a cutter over this surface.

The current practice of recovering only low-level geometric models and then using these to generate NC code is most appropriate for parts consisting largely of sculptured surfaces. Representing geometry in terms of surface points or collections of parametric surface patches is adequate to describe positional information, but cannot capture any of the higher level structure of the object. It is thus nearly impossible to make modifications or to generate efficient and effective process plans automatically. For example, these representations might be able to capture the shape of a hole, but the fact that it is actually a true, cylindrical hole is not made explicit. As a result, it can be difficult for a designer to do something as simple as change the diameter of the hole. Modification of more complex manufacturing features is even more difficult.

In this paper, we describe an alternate approach for efficiently creating a CAD model of a part with a significant number of specialized manufacturing features. The system is interactive, since some aspects of the reverse engineering process cannot be done based on the part alone and other aspects of the process can benefit significantly from a small amount of human intervention. In a sense, we provide a set of *electronic calipers* to be used as a smart measuring tool, specialized to the job of creating CAD/CAM models. The system is effective because it analyses 3–D sensor data using knowledge of manufacturing processes and modeling techniques.

2 Geometric Representations.

Low-level geometric object models can be produced from sets of 3–D surface point locations using triangulated meshes [1, 2, 3, 4, 5, 6] or by fitting piecewise smooth surface parametric surface patches [7, 8]. Such models suffer from two important deficiencies, however, when used for reverse engineering. First of all, they do not provide an organization of the object into meaningful subparts. Secondly, the local smoothing that is implicit in these methods may not be optimal for reducing sensing noise in the raw data.

Many of the standard three-dimensional representation tools used in computer vision have similar problems. Wire frame models [9] and surface-edge-vertex networks [10, 11] don't adequately describe organizational structure. Generalized cylinders [12, 13, 14, 15, 16] and superquadrics [17, 18, 19] have been used to create hierarchical shape descriptions, but are poorly matched to the shapes that commonly occur in manufacturing. Full function CAD modelers have been used in computer vision systems [20, 21], but examples to date have presumed a priori availability of the models.

Many modern CAD/CAM systems support some form of *feature-based design*, allowing designers to

Stock

Hole
- *simple hole*
- *counter bore*
- *counter sink*
- *counter drill*
- *tapped hole*
- *counter drilled tapped*
- *back counter bore*
- *back counter sink*
- *step bore*

Slot

Pocket
- *rectangular pocket*
- *profile pocket*

Facing Features
- *straight step*
- *profile face*

Groove
- *internal groove*
- *external groove*
- *face groove*
- *profile groove*

Boss
- *circular boss*
- *profile boss*

Profile Features
- *profile chamfer*
- *profile round*
- *profile side*

Figure 1: Manufacturing features in the Alpha_1 CAD system.

specify a shape in terms of complex primitives [22]. For example, Figure 1 shows the manufacturing features available in the University of Utah's Alpha_1 CAD system (see [23]). Each of these feature types has associated with it the appropriate geometric information plus manufacturing specifications such as fillets and chamfers. Design systems of this sort have two clear advantages over modeling solely at the level of detailed geometry. They provide a more natural interface for machinists and they allow much more sophisticated automated process planning, since the intent of the designer is clearer.

Several methods have been proposed for automatically extracting a high-level, feature-based description from lower-level models of part geometry. [24] presents a comprehensive generate-and-test approach for recognizing features in solid models and handling their interactions. The method relies on extracting "hints" from a solid model and using them to complete a feature frame within a frame-based reasoning system. [25] presents an algorithm for taking a CAD model and extracting a set of machinable features suitable for generating all alternative interpretations of the model. [26] presents a method for encoding and recognition of machining features in a boundary representation. All of these systems start with an exact representation of surface shape. While they provide useful ideas applicable to creating high-level models from sensed data, none begin to deal with the error and variability present in such data.

3 Feature-Based Reverse Engineering.

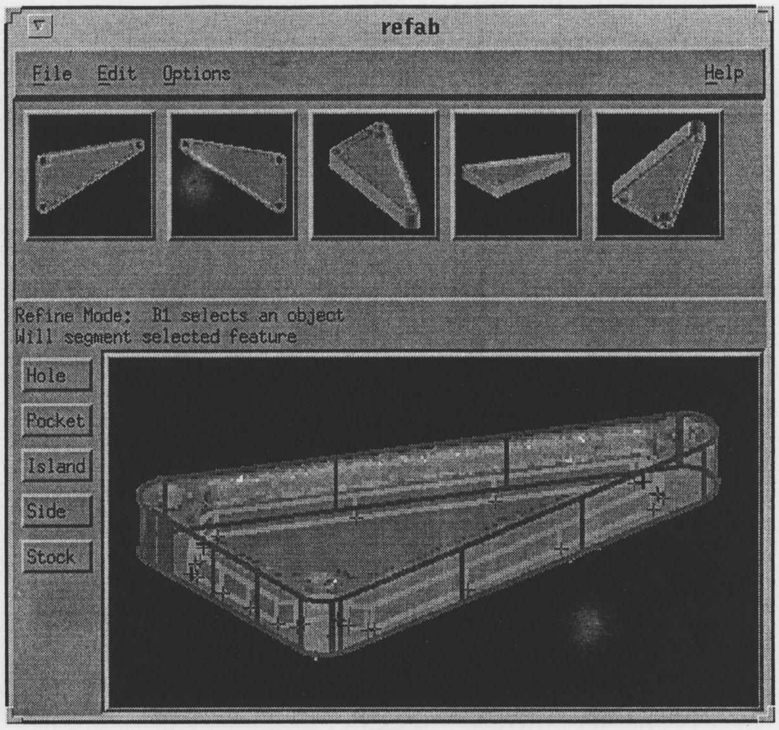
Sensor-based reverse engineering of mechanical parts must yield accurate object models appropriate for computer-aided manufacturing. Current commercial practice, which represents geometry in terms of scan lines or meshes of scan points, is inflexible and requires careful coordination between scanning patterns, tool selection, and tool paths. Parametric model fitting techniques proposed to date do not use geometric primitives that are natural to most manufacturing operations. Methods for extracting manufacturing features from lower-level geometric representations are intended to work with existing CAD models, not imperfect sensed data.

Improvements can be made by specializing the recovery of object models to the manufacturing environment. Most machined parts are made using a relatively small number of manufacturing operations, each of a constrained form (Figure 1). Reverse engineering can be done using parametric model fitting, where the primitives correspond to these features. This avoids inconsistencies between actual object shape and what the models are capable of representing, while leading in a natural and obvious way to representations usable in full-function CAD/CAM systems. The approach we describe here is interactive, which improves performance and allows for human entry of information that cannot be acquired from sensed data alone.

To demonstrate the effectiveness of feature-based reverse engineering, we have created a prototype system called REFAB (**R**everse **E**ngineering – **F**e**A**ture-**B**ased). REFAB allows a user to interactively define a model composed of mechanical features from a set of 3–D surface points. The user specifies the types of manufacturing features present and the approximate location of each feature in the object. REFAB deals with the determination of precise, quantitative parameterization of each feature. The final output is a fully specified model usable by the Alpha_1 CAD/CAM system.

Figure 2 shows the user interface for the REFAB system. Though all modeling computations are done on 3–D point cloud data obtained from position digitizers, user interaction is facilitated by generating a triangulated mesh from the points [1], and then using standard rendering techniques to create synthetic views from a variety of vantage points. While the triangulated mesh lacks the accuracy and structure of a high-quality CAD model, it generates rendered views sufficiently realistically to allow users to easily indicate features of interest. The series of small images along the top corresponds to alternate views of the same object and allows the user to specify a current working view. REFAB maintains a single, internal coordinate system and views can be switched at any time to provide a better perspective on whatever feature the user is currently interested in. The set of buttons at the lower left corresponds to the set of features the system is able to model. To model a feature, the user selects a feature type and a view in which the feature can be seen on the object. The panel on the lower right will show the selected view and all previously modeled features. The mouse is used to specify enough points on the displayed image to indicate the approximate location of the feature. REFAB will then analyze the 3–D position data to provide an optimal parameterization of the feature, render the feature on the display, and then prompt for the next feature to be modeled.

While a fully automated system might seem desirable, there are two aspects of modeling for manufacturing that are infeasible based on automatic processing of sensed data alone. Figure 3 shows a downward-looking view of a plate with an opening in the middle. The opening can be represented exactly using either two holes or a single profile pocket. To choose the preferable representation requires a rather complex un-



Refine Mode: B1 selects an object
Will segment selected feature

Figure 2: REFAB user interface.

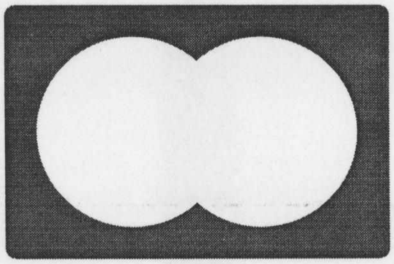


Figure 3: Two interacting holes or one pocket?

derstanding of dimensions, tolerances, and manufacturing costs. Next, consider a part which contains seven through holes of identical diameter, four of which mate with locating pins on another part, two of which stack with holes on parts to either side to form a conduit for oil, with the remaining hole providing access for a flexible cable that runs from one side of the part to the other. The tolerances and finishes required vary enormously. Cost effective fabrication requires that this information be understood and accounted for in the manufacturing process plan. The REFAB system acknowledges the need for human intervention, but frees the user from most of the tedious, quantitative analysis that can be done faster, easier, and more accurately by automated tools.

The current version of REFAB is limited to five common types of features: *stocks*, *simple holes*, *profile pockets*, *profile islands*, and *profile sides.*[2] The features are typical of those in parts machined using 3-axis mills for simple drilling and parallel sided cutting. Features can have different orientations, as would occur with refixturing with a 3-axis milling. Stock and island are additive features, holes, pockets, and sides are subtractive.

4 Segmentation, Fitting, and Refinement.

in REFAB, the first step in reverse engineering a machined part is to define the stock from which the part is to be cut. Currently, we support only block stock and determine the dimensions using a straightforward bounding box computation on the position data. Extensions to standard stock sizes and other stock shapes would be straightforward.

The remaining features require a more careful fit to the position data. Three interrelated problems must be solved in order to accurately model a particular manufacturing feature: determination of feature type, segmentation of relevant 3–D points, and model fitting. The user specifies the feature type and approximate location using REFAB's control panel. The segmentation and fitting operations proceed automatically, using an iterative refinement process.

Fitting a parameterized feature model to sensed data requires a decision as to what data points should be considered to lie on the feature and which values are parts of other features. Most other approaches to dealing with position data use some form of bottom up segmentation procedure (e.g., [27, 28]). 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 orientation 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.

Since in our case the user has specified an approximate feature type and location, we can use a much

---

[2] Profile features are extrusions of arbitrary planar curves. A profile island is a special kind of boss. It is defined only within the context of a pocket and specifies a volume to be "skipped" when the pocket is milled. A profile side represents a simple side cut (no plunging), and is typically used to trim stock down to the outside shape of a part.

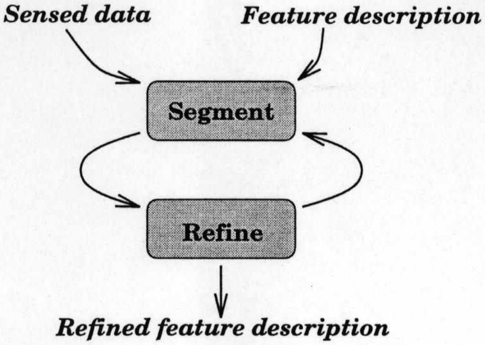


Figure 4: Iterating the segment/refine cycle.

more reliable top-down segmentation approach. Given an approximate feature parameterization, we select those position points that are close to the surface of the estimated 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. An 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. As the estimate of feature parameters is refined, the position data can be resegmented using tighter tolerances on distance and orientation, while reducing or eliminating the restrictions on which parts of the feature surface to consider.

Figure 4 illustrates the interaction between the top-down segmentation procedure and model fitting. A preliminary segmentation of the position data is done based on the user-provided initial feature description. A least-squares optimization procedure is then used to refine the parameters of the feature description, based on minimizing the distance from segmented points to some surface point on the feature. Next, the position data is resegmented by scanning the entire set of 3–D points for those that are consistent with the revised model, given a tighter set of tolerances than used in the previous iteration. Segmentation and refinement alternate until preset tolerance bounds are met for the segmentation process.

Except in the case of plane fitting, optimization is non-linear. Fixed degree-of-freedom parametric features such as holes are fit using a generalized simplex search algorithm [29]. Since we start with a good initial estimate, this can be done without significant worry about undesired local minima or unreasonable computational cost. Profile features are modeled by first determining the direction of sweep for the feature. A contour is then fit the points on the sides of the feature, projected into a plane normal to this direction. The fit is done using line segments and constant radii curves when possible, and splines elsewhere.

Many of the manufacturing features used in machined parts are effectively extrusions of planar curves. The geometry of such features can be represented by an *anchor* specifying the orientation of the extrusion and a local coordinate system for other dimensional information, a closed contour specifying the curve to be swept, and the start and stop locations for the sweep. Holes, slots, pockets, and sides are commonly drilled or milled on planar faces at an orientation perpendicular to the face. When a REFAB user indicates that this is the case for a particular feature, the system is able to use a simplified approach to model fitting. Once the sweep orientation has been determined, the coordinates of points on the walls of the feature can be projected along this direction and the remainder to the analysis done in 2–D.

5 Results.

We have tested the REFAB system on several machined parts originally designed for the Utah mini-Baja and formula SAE racing vehicles. Results from two of these parts are presented here. While the parts are relatively simple, they provide an adequate test of the accuracy and usability of our system. Figure 5 is a wire frame drawing generated from the CAD model used to create the shock mounting plate that forms a linkage in one version of the rear suspension of the vehicle. To better fit our workspace requirements, a special plate was made that was three quarters the size of the one used on the vehicle itself, yielding a part that was approximately 17.75 cm x 7.5 cm x 2 cm. Figure 6 shows one of these scaled shock plates, produced using a 3-axis mill. The second object is part of the vehicle's steering arm assembly and is approximately 10 cm x 5 cm x 2 cm (Figures 9 and 10). Figures 5 and 9 are included for information only. All reverse engineering operations were done without using any aspect of the CAD models from which the parts were originally constructed. Testing REFAB on objects of known geometry and design allows a comparison of the recovered model with the true shape of the part [30].

Position data was acquired with a DIGIBOT II laser position digitizer. The DIGIBOT II has a nominal measurement accuracy of ±50 microns (1 σ) under optimal conditions. In practice we have observed accuracies on the order of ±50-300 microns, depending on the nature and shape of the surface at that point. For evaluation purposed, we produced special versions of both parts without chamfers and threads, which were too small to be accurately measured with the DIGIBOT system. To remove specularities that cause problems for most current range finding systems, parts were sprayed with a penetrant process developer (Sherwin DUBL-CHEK D-100), which leaves a thin, talcum-like coating. Multiple views were taken of each part and registered into common point-cloud data sets, using a registration procedure similar to that in [31]. 143,140 3–D points were used in reverse engineering the engineering the shock plate. 44,578 were used for the steering arm. Figures 7 and 11 show samplings of both point sets, rendered so that nearer points are brighter.

Figures 8 and 12 are wire frame drawings generated from the reverse engineered CAD models produced by REFAB for the two parts. To emphasize that the recovered CAD models are feature-based and not just arbitrary surface representations, Figures 13 and 14 show exploded views of the two models indicating the separate features making up each object. The shock plate is a fairly simple object with an outer contour defined by a profile side, two symmetric profile pockets that serve to lighten the part, and three mounting holes. The steering arm has an outer profile side with both smooth contours and sharp corners, one large hole and one smaller hole drilled normal to the stock, and two small holes drilled in a perpendicular orientation.

Since we have available for testing purposes the original CAD models for both parts, it is possible to quantitatively analyze the performance of the REFAB system. Tables 1 and 2 show the deviation between

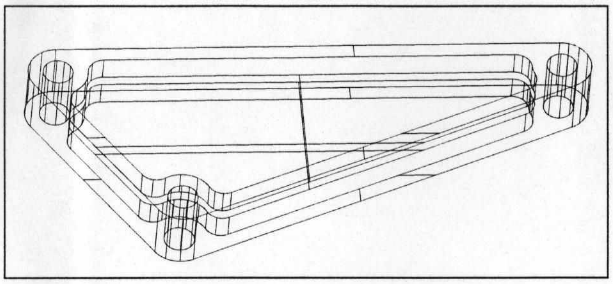


Figure 5: CAD model used to make shock plate.

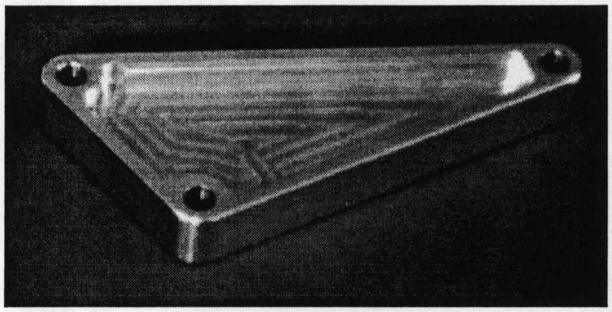


Figure 6: Shock plate: actual part.

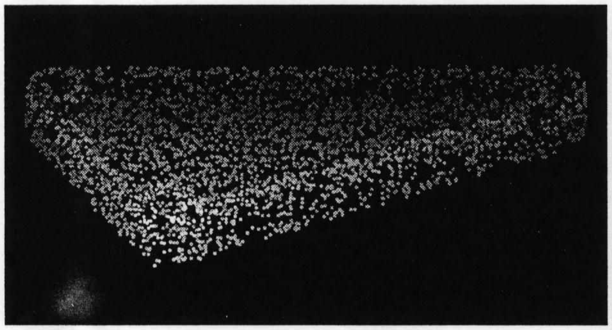


Figure 7: Shock plate: sensed 3–D position points.

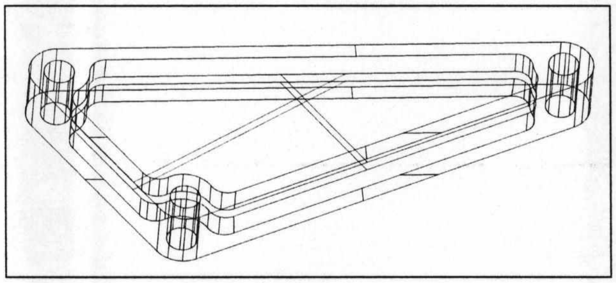


Figure 8: Wire frame rendering of reverse engineered shock plate.

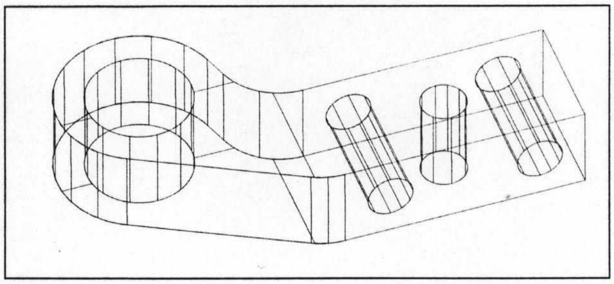


Figure 9: CAD model used to make steering arm.

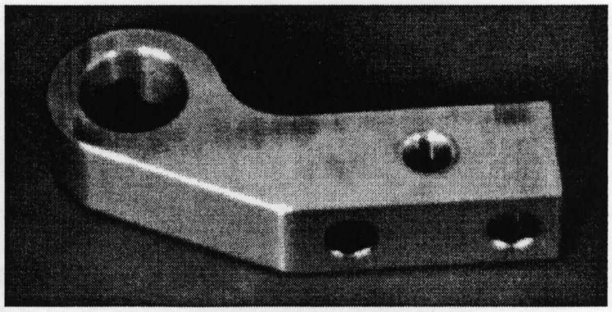


Figure 10: Steering arm: actual part.

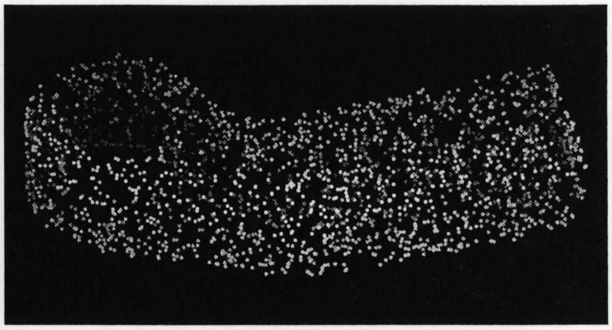


Figure 11: Steering arm: sensed 3–D position points.

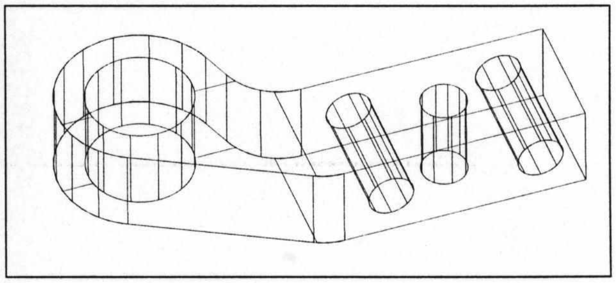


Figure 12: Wire frame rendering of reverse engineered steering arm.

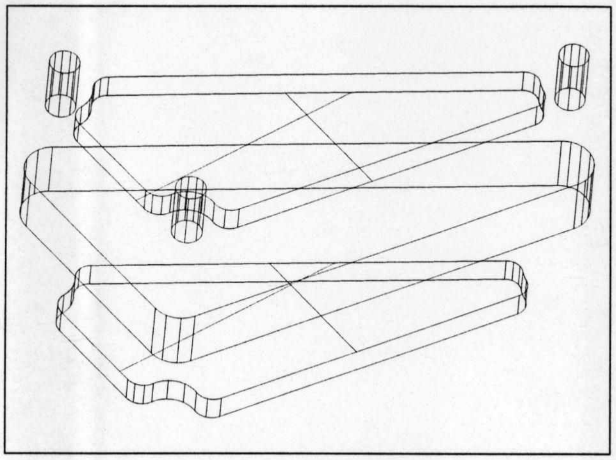


Figure 13: Exploded view of the features making up the reverse engineered shock plate.

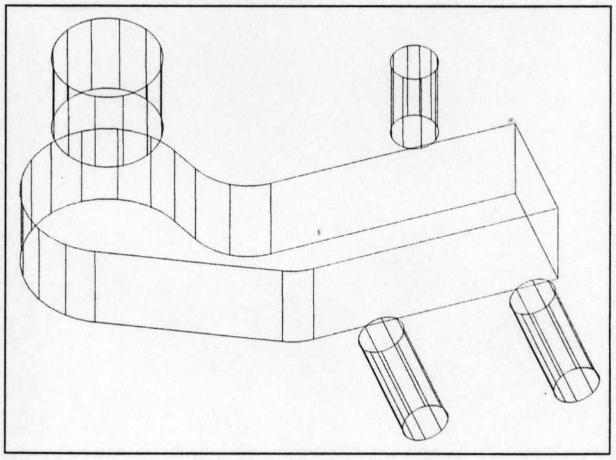


Figure 14: Exploded view of the features making up the reverse engineered steering arm.

Shock plate	RMS	worst-case
overall	61	350
outer profile	48	118
top cap	6	6
bottom cap	29	29
pocket 1 bottom	6	6
pocket 1 side	114	350
pocket 2 bottom	43	43
pocket 2 side	141	350
hole 1	86	119
hole 2	72	104
hole 3	103	151

Table 1: Modeling error – reverse engineered shock plate (microns).

Steering arm	RMS	worst-case
overall	59	127
top	66	66
bottom	66	66
outer profile	63	127
large hole	35	57
small top hole	22	57
side hole 1	48	88
side hole 2	35	80

Table 2: Modeling error – reverse engineered steering arm (microns).

the reconstructions and the original CAD model. The values were determined by first registering the original and reconstructed models and then taking a uniform sampling of points on the reconstructed surfaces and determining the distance to the nearest point on some surface in the original model. Both RMS and worst-case values are given.

6 Conclusions.

The use of manufacturing features as geometric primitives in part-to-CAD reverse engineering systems provides substantial advantages in accuracy and usability. In a prototype system, we were able to reverse engineer CAD models with an accuracy often exceeding that of the precision of the sensor used to acquire raw data about part shape. The models which are produced are feature-based, providing a higher level description of part geometry and allowing easy importation into feature-based CAD systems.

Our system does not yet deal with secondary feature properties such as taps, chamfers, fillets, and rounds. Each of these involves small scale geometry that requires specialized gauges for accurate measurement. Once measured, however, the feature based representation allows for easy addition of this information to the model without the tedious surface blending that would be necessary if large-scale geometry were represented only as an unorganized set of surface patches.

Acknowledgments.

Thomas C. Henderson suggested to us the idea of interactive feature specification. Peter-Pike J. Sloan created the user interface. Alyosha Efros and Brian Morris provided software support. Samuel Drake gave us important insights into the manufacturing process and assisted in the creation of test objects. The Alpha_1 group provided technical support on geometric modeling and manufacturing.

References

[1] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," in *Computer Graphics, SIGGRAPH '92*, vol. 26, July 1992.

[2] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Mesh optimization," in *Computer Graphics, SIGGRAPH '93*, vol. 27, August 1993.

[3] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Piecewise smooth surface reconstruction," in *Computer Graphics, SIGGRAPH '94*, vol. 28, August 1994.

[4] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *International Journal of Image and Vision Computing*, vol. 10, pp. 145–155, April 1992.

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

[6] G. Turk and M. Levoy, "Zippered polygon meshes from range images," Tech. Rep. CSL-TR-94-609, Stanford University, February 1994.

[7] B. Sarkar and C. H. Menq, "Smooth-surface approximation and reverse engineering," *Computer Aided Design*, vol. 23, pp. 623–628, November 1991.

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

[9] L. Roberts, "Machine perception of three-dimensional solids," in *Optical and Electro-Optical Information Processing* (J. Tippett *et al.*, eds.), Cambridge, MA: MIT Press, 1965.

[10] A. Hanson and E. Riseman, "VISIONS: A computer system for interpreting scenes," in *Computer Vision Systems* (A. Hanson and E. Riseman, eds.), pp. 303–333, Academic Press, 1978.

[11] W. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints.* The MIT Press, 1990.

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

[13] R. Nevatia and T. Binford, "Description and recognition of curved objects," *Artificial Intelligence*, vol. 8, pp. 77–98, 1977.

[14] R. Brooks, "Symbolic reasoning among 3D models and 2D images," *Artificial Intelligence*, vol. 8, pp. 77–98, 1977.

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

[16] I. Biederman, "Recognition-by-components: A theory of human image understanding," *Psychological Review*, vol. 94, pp. 115–157, 1987.

[17] A. Pentland, "Perceptual organization and the representation of natural form," *Artificial Intelligence*, vol. 28, pp. 29–73, 1986.

[18] A. Gupta and R. Bajcsy, "Part description and segmentation using contour, surface and volumetric primitives," in *Proc. Conf. Sensing and Reconstruction of 3D Objects and Scenes*, pp. 203–214, February 1990.

[19] V. Koivunen and R. Bajcsy, "Rapid prototyping using three-dimensional computer vision," Computer and Information Science Dept. MS-CIS-92-70, University of Pennsylvania, August 1992.

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

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

[22] J. J. Shah, "Assessment of features technology," *Computer-Aided Design*, June 1991.

[23] S. Drake and S. Sela, "A foundation for features," *Mechanical Engineering*, vol. 111, January 1989.

[24] J. Vandenbrande and A. Requicha, "Spatial reasoning for the automatic recognition of machinable features in solid models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pp. 1269–1285, December 1993.

[25] W. Regli, S. Gupta, and D. Nau, "Feature recognition for manufacturability analysis," Tech. Rep. ISR TR94-10, University of Maryland, February 1994.

[26] A. Fischer and M. Shpitalni, "Encoding and recognition of features by applying curvatures and torsion criteria to boundary representation," in *ASME 1992 Winter Annual Meeting – Symposium on Concurrent Engineering*, pp. 69–84, November 1992.

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

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

[29] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes in C*. Cambridge, MA: Cambridge, 1991.

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

[31] H. Y. Shum, K. Ikeuchi, and R. Reddy, "Virutal reality modeling from a sequence of range images," in *Proc. ARPA Image Understanding Workshop*, November 1994.