Wrapping 3D Scanning Data

Herbert Edelsbrunner[*†], Michael A. Facello[†], Ping Fu[†], Jiang Qian[†] and Dmitry V. Nekhayev[†]

[*]Department of Computer Science, University of Illinois at Urbana-Champaign
[†]Raindrop Geomagic, Inc., Champaign, Illinois

Abstract

geomagic Wrap™ is a commercially available software for reconstructing shapes and surfaces from 3D scanning data. The data can be any arbitrary finite point set in 3D, and there are no requirements on local density or organization in slices etc. The software contains components for surface reconstruction, improvement, and analysis, and it supports a variety of input and output formats that make it compatible with scanning hardware and CAD and graphics software.

Keywords. Surface reconstruction, geometric algorithms, computer aided geometric design, physical design, 3D scanning, reverse engineering, rapid prototyping.

1 Introduction

The purpose of this paper is to describe geomagic Wrap™, for short Wrap, which is a commercial software package for converting scanning data to computerized models of shapes and surfaces. Given a scanned set of points in 3D, Wrap reconstructs the surface in an automatic fashion. It also provides high-level geometric editing functions that allow the user to modify the surface and to improve its quality in terms of smoothness and other criteria.

Physical versus conceptual design. The lack of software for the automatic reconstruction of surfaces from scanning data has long been a stumbling block in the development of commercially viable reverse engineering solutions to customized manufacturing. With the new technology, it will be economically feasible to use physical design in industries where uniqueness and faithful representation are of critical importance. Examples are reverse engineering of human body parts, such as teeth and blood arteries in medical CAD applications, and the customized manufacturing of high-end consumer goods, such as ski boots and artificial hair-pieces. *Physical design* gathers data from a physical object with the goal to generate a precise computerized model of that object. This process is sometimes also referred to as reverse engineering. Traditional *conceptual design* starts with an idea and often with a hand-drawing and a computerized model is generated from scratch using modeling software. With the availability of cost-effective physical design software coupled with scanning hardware, computer aided geometric design can be extended to industries that are not amenable to the conceptual design paradigm that dominates the CAD market of today.

Historical development. Before beginning the technical sections of this paper we say a few words about the historical background and the evolution of Wrap. The intellectual roots can be found in combinatorial studies of algorithms, geometry, and topology. Wrap shares some of the mathematical ideas with the Alpha Shapes software developed under the guidance of Edelsbrunner and Fu at the University of Illinois. Alpha Shapes has been freely distributed for more than five years and it is used all over the world by scientists and engineers. Alpha Shapes is virtually bug-free but it is not a solution to the reverse engineering problem and was never designed as one. Wrap was developed by Raindrop Geomagic and it addresses some of the shortcomings of Alpha Shapes if used for surface reconstruction. Wrap differs from Alpha Shapes as follows:

1. Wrap constructs one initial shape while Alpha Shapes generates an entire family of shapes, all at once.
2. Wrap adapts to local and possibly varying point density while Alpha Shapes relies on uniform density to produce a surface without holes.
3. The shape constructed by Wrap can be edited while Alpha Shapes provides no control for the user.
4. The Wrap software uses object oriented software design and is written in `C++` while Alpha Shapes is written in `C`.

In summary, Wrap has been raised in the commercial world while Alpha Shapes grew up in a University environment. For converting scanning data to computerized models, Wrap is vastly superior to Alpha Shapes, which is designed for molecular modeling applications.

2 Related Work

Surface fitting is an extensively researched topic in computer graphics, CAD/CAM, approximation theory, and statistics. We limit the literature review to work directly related to Wrap. Additional references can be found in survey papers such as Lodha [1].

3D Delaunay complexes. The central idea of Wrap is to reconstruct the surface as a subcomplex of the 3D Delaunay complex uniquely defined by the data set. This idea appeared earlier in the computational geometry literature. In 1983 α-shapes have been defined in 2D by Edelsbrunner, Kirkpatrick and Seidel [2]. They were later generalized to 3D by Edelsbrunner and Mücke [3] and used for surface reconstruction by Bajaj, Bernardini and Xu [4]. While α-shapes and Wrap subcomplexes are different, they share common intellectual roots. In 1984, Boissonnat [5] described how to sculpt a shape from a 3D Delaunay complex by removing simplices at or next to the boundary. The weakness of that method is the lack of an effective rule for deciding which simplices to remove and in what sequence, and this weakness has been remedied by Wrap.

Parallel slices. The reconstruction of a surface is considerably easier than in the general case if the points represent a sequence of parallel slices, such as in CAT scans. Fuchs, Kedem and Uselton [6] show how to connect two polygons in parallel planes with a cylindrical surface of minimum area. The problem is more difficult if there are more than one polygon per slice. Various heuristics for determining which polygons to connect and how to draw the surface have been investigated. For example Meyers, Skinner and Sloan [7] use the minimum spanning tree and Boissonnat and Geiger [8] use the 3D Delaunay complex to decide where to draw the connecting surface. In spite of the effort reflected by a large body of literature there is no algorithm that produces branching surfaces from sliced data in a generally satisfactory manner.

Signed distance functions. A method for surface reconstruction that has become popular in computer graphics uses the data to construct a continuous function $f : \mathbb{R}^3 \to \mathbb{R}$; the surface is the zero-set, $f^{-1}(0)$, which can be constructed e.g. with the marching cube algorithm of Lorensen and Cline [9]. Hoppe et al. [10] construct f to approximate the signed Euclidean distance. A necessary assumption in their work is that the data is uniformly dense over the entire surface and that the density exceeds the smallest feature size. Curless and Levoy [11] use information about rays available from some types of 3D scanners to extrapolate f over gaps in the scanning data.

Manual and semi-manual patch fitting. Patches of spline surfaces can be fit manually over scanning data. This is still the preferred method in the animation industry where looks are more important that accuracy. Semi-manual methods are common in the mechanical CAD industry where parametrized patches are fit over subsets of the data identified by a user-guided process. The commercial software CopyCAD by Delcam and Surfacer by Imageware are examples of this strategy. The fitting process is easier but not straightforward if the points are already connected to a surface by a collection of triangles. Krishnamurthy and Levoy [12] describe how they manually choose and automatically fit spline surfaces over an already triangulated surface.

Automatic patch fitting. Methods for smoothing a piecewise linear surface are abundant in the splines literature. For example, Loop [13] and Peters [14] construct a C^1-continuous surface by decomposing triangles into smaller pieces and replacing those by low-degree Bézier triangles and quadrilaterals. Both methods result in a large number of patches. This drawback has been addressed by Eck and Hoppe [15] who fit tensor product B-spline patches over a decimated set of quadrilaterals and use regression to approximate the original data. Similarly, Bajaj, Chen and Xu [16] use regression to fit A-patches that approximate the signed-distance function near a triangulated surface. Each A-patch is a map $\mathbb{R}^3 \to \mathbb{R}$ defined within a tetrahedron, and it contributes its zero-set clipped within that tetrahedron to the surface description.

3 Overview of Wrap Software

The current version 1.2 of the Wrap software consists of six components. The three components referred to as phases reconstruct the surface from the data, one component provides analysis functionality, and the two remaining components handle the input and output. During the surface reconstruction process, Wrap maintains a history of all operations performed. The user can return to any point during the editing process and continue from there. It is thus easy to experiment with the software and find operation sequences that produce the best results. The remainder of this paper devotes one section to each component and illustrates features with examples and pictures. The components are as follows:

Input. Scanning data is either read from files or received directly from scanning devices.

Point Phase. The scanning data can be edited for reduction of density in selected sections, removal of noise, etc.

Wrap Phase. The surface is automatically reconstructed from the given point data.

Surface Phase. The surface can be manipulated to produce representations that are smoother, denser, sparser, etc.

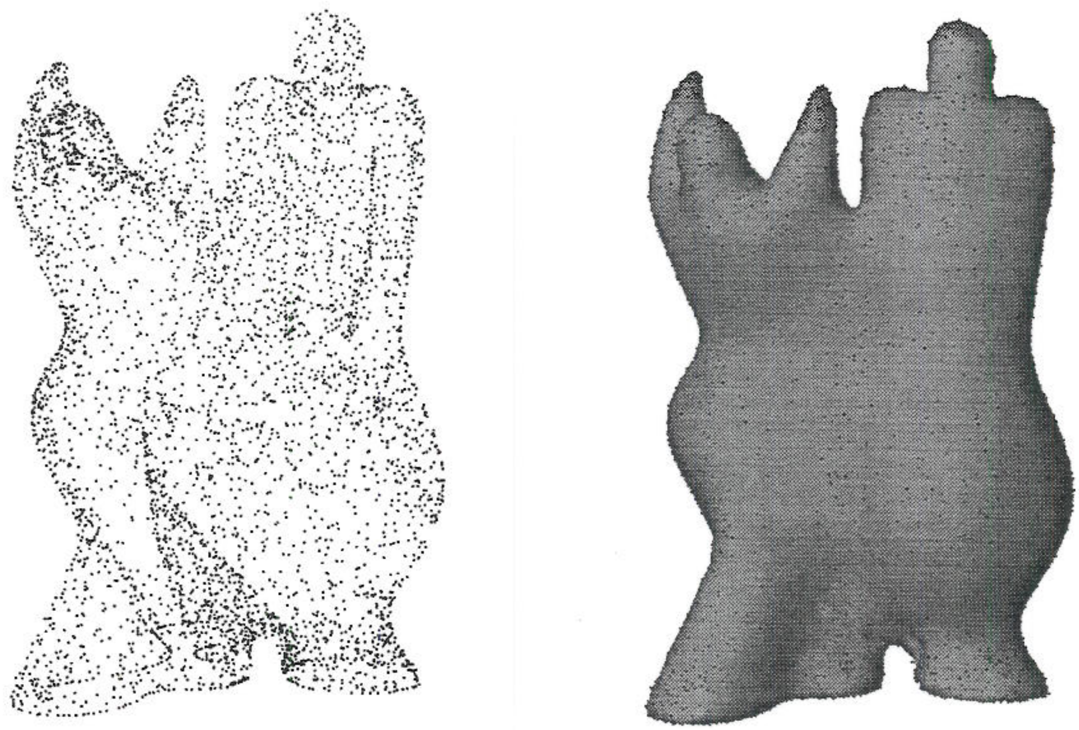
Analysis. Properties of the surface and its relationship to the original model or data can be measured.

Output. The reconstructed surface can be output in a variety of industry-standard formats.

4 Input

Wrap accepts various input file formats and direct input from digitizers. The simplest format is a vertex file, which is a list of x-, y-, z-coordinate triplets. This format allows the user to import virtually any file describing a set of 3D points. Among others Wrap accepts OBJ and STL files as well as the 3D scanner formats of Digibotics and Cyberware. For each of these file types, Wrap reads 3D points that are manipulated in the later components of the software. In the future, Wrap will also be able to read the geometry from OBJ and STL files and allow the user to directly apply the surface improvement and analysis operations discussed in Sections 7 and 8. Wrap has its own WRP format used to save models which can be loaded at a later time for further processing.

In addition to file input, Wrap accepts direct input from a Microscribe digitizing arm. This digitizer collects points on the surface of a physical object while touching it with the tip of the arm. Other software packages that work with the arm require advance planning such as drawing a rectangular grid of points on the object that are then digitized. Since Wrap does not rely on any particular point distribution, it allows the user to freely scribble by moving the tip continuously over the surface. This technique requires no advance planning and decreases the amount of time it takes to digitize. The point set in Figure 1 (a) was collected in this way from a ceramic sculpture reconstructed in (b).



(a) (b)

Figure 1: (a) Data set obtained from ceramic sculpture with a hand-held digitizer. (b) Reconstructed surface after applying some of the surface improvement operations described in Section 7.

5 Point Phase

The point phase provides tools to manipulate the input data by local reduction of point density. Often the user will skip this phase and reconstruct the surface directly from the input set. However, there are many reasons why one would want to have the option of preprocessing the data, e.g. if the density is too high or there is obvious noise that can be removed manually.

The main operation in this phase reduces the density within a local region to a percentage specified by the user. The region is defined by a rectangle or polygon with a depth plane truncating the corresponding 3D cylinder normal to the computer screen. The reduction of density is achieved by random sampling, which is not arbitrary but rather gives each point the same chance to remain and that chance is the specified percentage. Two additional operations are erasing, which removes *all* points inside the region, and cropping, which removes all points *outside* the region. It is also possible to add new points whose coordinates are determined by the depth plane and the screen location of the mouse. To support automation of point processing, Wrap provides point templates which are files that record sequences of operations. The user can define such a template by stepping through one example and saving the sequence. The template can then be used in batch mode to preprocess many or maybe just a few but large input sets.

6 Wrap Phase

The basic surface reconstruction happens in the wrap phase. This is a difficult problem in particular because we insist the algorithm works for any set of points, no matter how they are distributed in 3D. We make an attempt to explain the basic logic of the algorithm, enough to make the results plausible.

Theoretical background. As mentioned earlier, Wrap produces shapes and surfaces by taking subcomplexes of the Delaunay complex of the data set, S. This is a particularly elegant and useful decomposition of the convex hull described in a seminal paper by the Russian mathematician Boris Delaunay [17]. In the generic case it consists of tetrahedra spanned by quadruplets of points in S. Specifically, if a, b, c, d are four points in S then $abcd$ is a tetrahedron in the Delaunay complex if and only if all other points lie outside the unique sphere that passes through a, b, c, and d. The set of tetrahedra together with their triangles, edges, and vertices make up the *Delaunay complex*. Wrap uses an incremental algorithm for constructing the complex [18, 19] and it uses symbolic perturbation to reduce any degenerate configuration to the generic case [20].

A shape is obtained by selecting a *subcomplex*, that is, a subset of simplices that are closed under the face relation. For example the helix surface in Figure 2 (c) is a collection of triangles from the Delaunay complex, together with their edges and vertices. It is a smaller subcomplex than the one in (b). Taking a subcomplex is a very general operation that can result in volumetric shapes, in surfaces, and even in curve-like objects. Commonly, Wrap is used to reconstruct surfaces, but even here there are different types. In a *2-manifold* every edge belongs to exactly two triangles. In a *2-manifold with boundary* every edge belongs to either two triangles or to one triangle, and the latter edges form the *boundary*. Finally, there are surfaces with *branching* or *non-manifold* edges that belong to three or more triangles. Wrap makes a choice among the various types that best fits the input data set. It is also possible that it reconstructs a surface in some parts and a volume attached to it in another part, as for example in Figure 2 (b). In summary, Wrap is dimension-independent and it is able to automatically and freely mix different dimensions as required by the data set.

Block operations. A difficult question that remains is *which* subcomplex should be selected, and this is a critical point where Wrap differs from the sculpting and the α-shapes algorithms described in [5, 2, 3]. All we can say in this paper is that the simplices in the Delaunay complex are organized in *blocks*. In the first step, Wrap removes the outer block and returns the boundary of what remains as the reconstructed surface. This is a little bit like burning point holes in the interior of a raw piece of marble with a laser ray device. If these holes are arranged along the intended surface it will be sufficient to touch the piece and all surplus material outside that surface will automatically fall off and reveal the shape in an instance.

Whether or not the entire shape is revealed in this one automatic step depends on the data set. For example, the data in Figure 2 turns too much to suggest a clear enough surface that could be reconstructed in a single step. As shown in (a), the data set is generated by moving and at the same time rotating a line in space. Points are sampled in regular intervals along seventeen positions of the line during that motion. The surface reconstructed in a single step is shown in Figure 2 (b). The final surface in (c) is obtained by removing two blocks of simplices, one on each side of the surface. In Wrap, the two blocks are removed in a single step: all it takes is to mark each block at a triangle and to issue a push deep operation. To be more accurate, the simplices on each side of the surface are organized in several blocks, which can be removed one by one in a sequence of push shallow operations. However, neighboring blocks are organized in an asymmetric relation defined by a notion of size, and the push deep operation is able to remove an entire conglomerate of blocks at once using the rule that a block is removed recursively if it is adjacent to a removed block of smaller size.

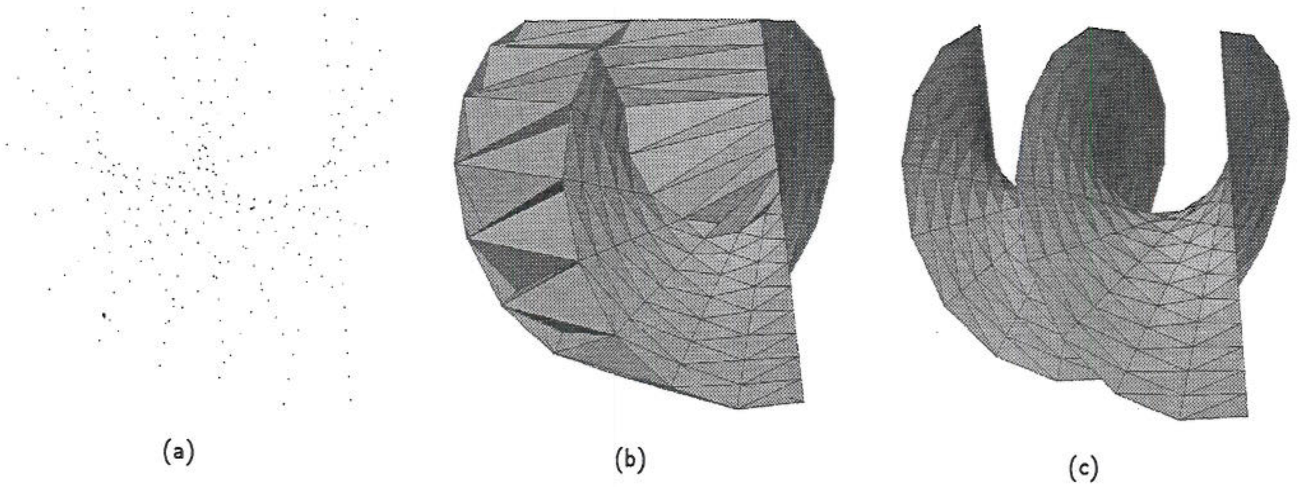


Figure 2: (a) Data set sampled from a line moving in space. (b) Initial reconstruction of the helix. (c) Helix surface obtained from the initial reconstruction by issuing a single push deep operation.

Fine-tune operations. Sometimes the organization into blocks is too coarse and the user will want to manipulate the surface on a finer level. Typical cases where this occurs are mechanical shapes with clear feature lines, such as the ring boundary in Figure 3, which is part of a steering knuckle used in a car design. The initial reconstruction at the left keeps the holes closed. Each hole consists of blocks that are removed in a single push deep operation. The result is shown in (b). Now we notice several imperfections along the feature lines. The boundary of the ring shows several bites in particular at the top and the bottom, and these bites are repaired in one step by the global angle repair operation. This operation uses a heuristic to look for and repair interruptions along otherwise clear feature lines. The concave feature line at the lower left of the knuckle piece is interrupted by remaining material that is removed with a fine-tune push operation.

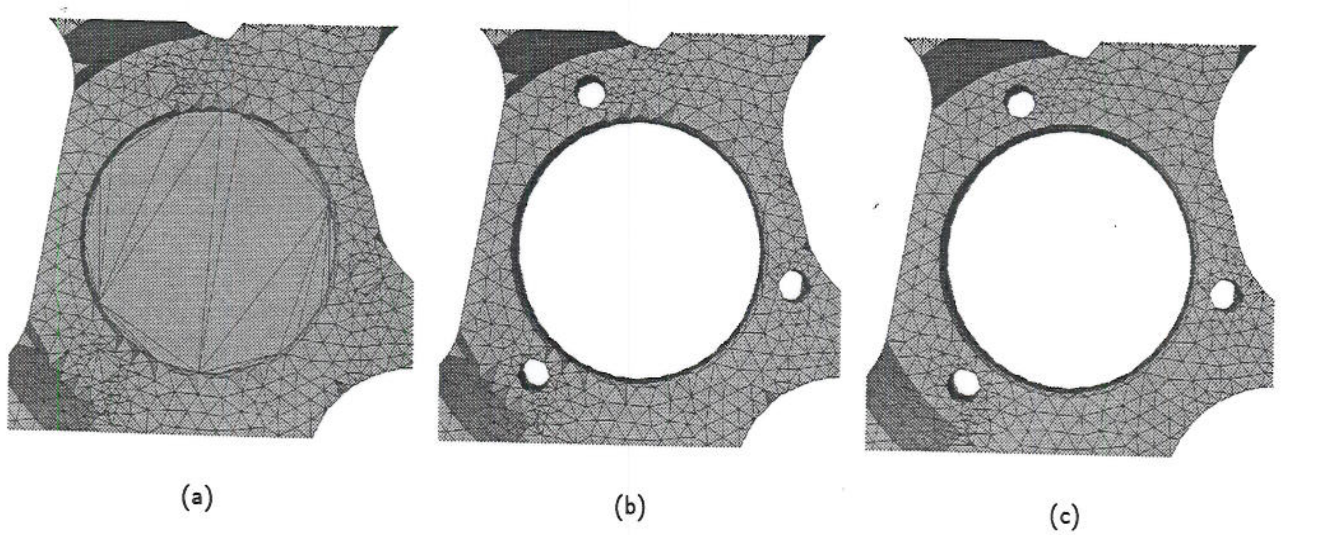
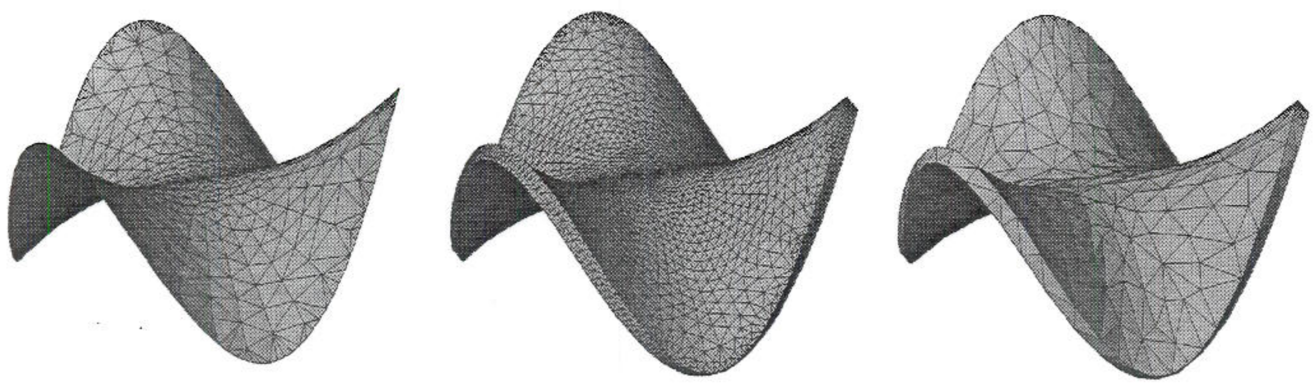


Figure 3: Original data courtesy of Mary Pickett at General Motors. (a) Part of a steering knuckle after initial reconstruction. (b) Knuckle after opening holes by aplying a push deep operation. (c) Knuckle after repairing feature lines with angle repair operations.

7 Surface Phase

All operations in the wrap phase manipulate the surface without changing the input data. There are modeling effects that can only be achieved by adjusting the position of points. This section discusses four such operations: thickening, refining, relaxing, and decimating. Each operation can be applied either globally or restricted to one or several interactively defined surface regions. In most cases the default is the global application, but we also use intelligent *automatic* region selection to expedite the execution of common operation sequences.

Thickening. Mathematical surfaces have zero thickness whereas their physical counterparts, such as sheets of metal or paper, have small but definitely non-zero thickness. The thickening operation provides the modeling step that turns a surface into a sheet of user-specified thickness. An example can be seen in Figure 4 (a) and (b) where a piece of a so-called monkey saddle is thickened by creating two copies off-set from each other by the specified amount and connected along the boundary by a thin strip.



(a) (b) (c)

Figure 4: (a) Monkey saddle reconstructed automatically in one step in the wrap phase. (b) Surface in (a) refined once and then thickened by a small amount. (c) Surface in (b) decimated to 10% the number of triangles.

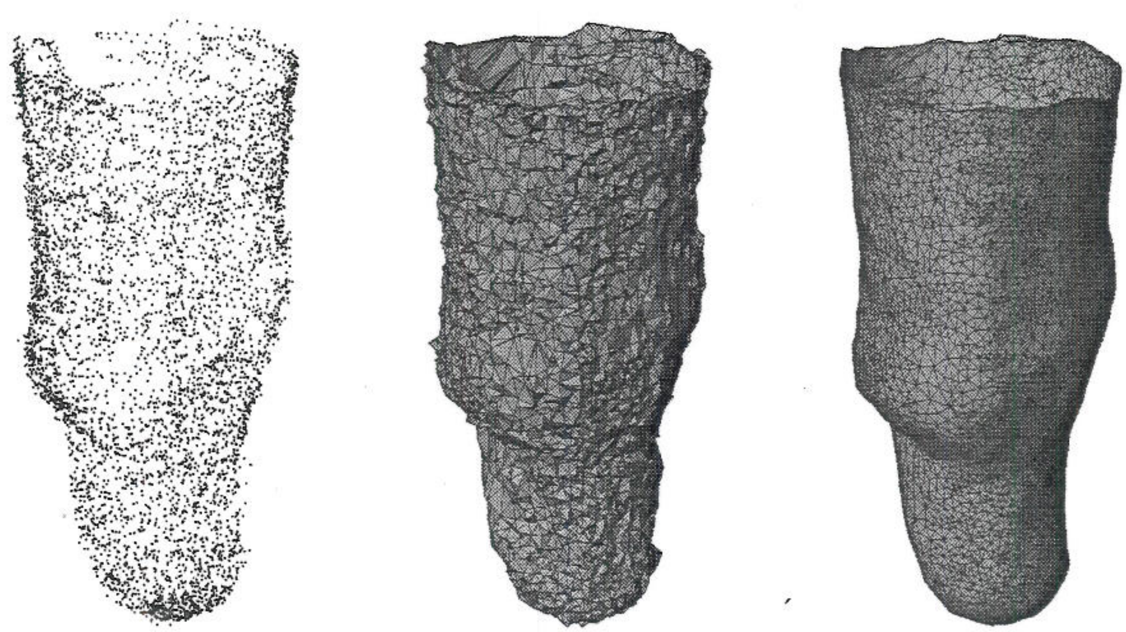
Observe that the original monkey saddle in (a) is a 2-manifold with boundary: it is a disk warped up and down three times with a similarly warped boundary circle. The thickening operation produces two slightly off-set copies of the saddle by locally moving each vertex a small amount along its estimated normal directions. Since off-setting is done locally, the achieved thickness is uniformly the same everywhere. The two copies of the saddle are connected by a cyclic strip. The final result is a 2-manifold (without boundary). As a word of caution we note that thickening by too large an amount can cause the locally off-set surfaces fold upon themselves. Section 8 discusses how to detect resulting self-intersections and how to repair them.

Refining. A subdivision of the surface into more and smaller triangles is achieved by the refinement operation. If applied globally it cuts each edge at the midpoint and it decomposes each triangle into four similar triangles. If applied locally, the triangles in the chosen regions are cut into four, triangles adjacent to these regions are cut into two, and all other triangles remain as they are. An interesting version of the operation adjusts vertex positions during a refinement step to improve smoothness. The result is referred to as a ***subdivision surface*** in the computer graphics literature where many different schemes are described. The simplest and most popular one is due to Loop [21]. If p is an old vertex with neighbors $q_1, q_2, \ldots, q_m$ its location is adjusted to

$$p' = (1-\mu) \cdot p + \frac{\mu}{m} \cdot \sum_{i=1}^{m} q_i,$$

where $\mu = \frac{1}{64} \cdot (40 - (3 + 2\cos\frac{2\pi}{m})^2)$. Instead of the midpoint between p and q, the position of the new vertex cutting the edge into two is chosen to be $\frac{1}{8} \cdot (3p+3q+r+s)$, where pqr and pqs are the two triangles sharing the edge. The subdivision surface approaches a smooth limit if the refinement and position adjustment operations are applied repeatedly. The effect of a single application is illustrated in Figure 4 (b).

Relaxing. Similar to the second version of the refinement operation, a relaxation improves the smoothness of the surface. The difference between the two operations is that relaxation achieves the smoothing effect solely by adjusting vertex positions. Thus, refinement is appropriate where non-smoothness is due to a lack of detail while relaxation is appropriate where it is a result of noisy data. Wrap implements the method of Taubin [22] that interprets shape characteristics as sums of waves with frequencies. To *relax* means to eliminate high frequencies. The basic idea precipitates to a formula of the same type as Loop's: the old point p is moved to a new location at $p' = p + \mu \cdot \Delta p$, where $\Delta p = \frac{1}{m} \cdot \sum(q_i - p)$ is the average difference vector to the m neighbors. The straightforward application of this idea has the side effect of global shrinking, which is counteracted by alternating two types of iterations: $p'_1 = p + \mu_1 \cdot \Delta p$ and $p'_2 = p + \mu_2 \cdot \Delta p$, where $\mu_1 > 0$ and $\mu_2 < 0$ are small constants whose sum is ever so slightly negative. The first step has a shrinking effect while the second counteracts with a slight expansion. The values of μ_1 and μ_2 used in relaxation are significantly smaller than that of μ in the subdivision scheme. As a consequence, relaxation steps are iterated, maybe between 5 and 50 times before they reach the desired effect. As an example consider the reconstruction of the surface of a person's



(a) (b) (c)

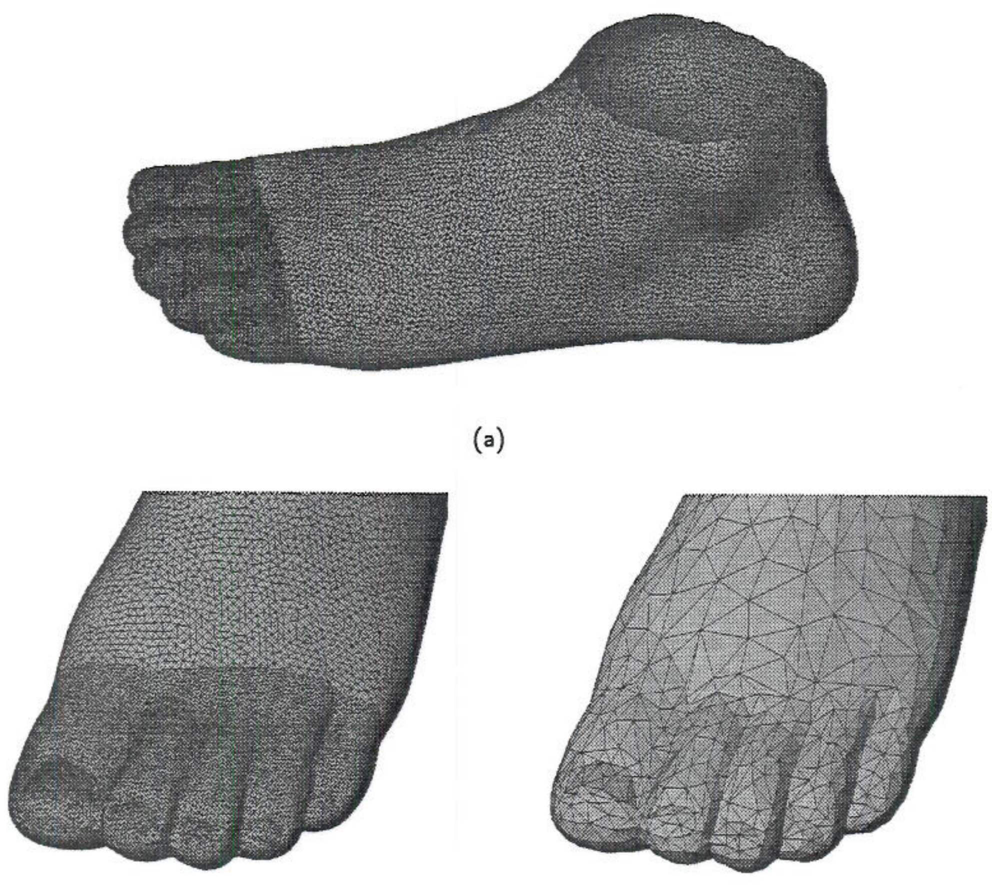
Figure 5: Original data courtesy of Guy Houser at Seattle Limb Systems. (a) Residual limb scanned with hand-held laser scanner. (b) Surface reconstructed from noisy data. (c) Surface after 50 iterations of global relaxation.

residual limb illustrated in Figure 5. The data in (a) is obtained with a prototype hand-hold laser range scanner. The data is exceedingly noisy as is obvious from the automatically reconstructed surface in (b). The smooth surface in (c) is obtained by repeated relaxation.

Decimating. While a finer subdivision of the surface is generated by refinement, the opposite effect is achieved with decimation. The user can specify a percentage and the decimation operation will reduce the number of triangles describing the same overall shape to that percentage, if possible. The effect of decimation is illustrated in Figure 4 (c). The method of choice is repeated edge contraction as described by Garland and Heckbert [23]. At each step an edge ab is chosen and contracted to a new point c. The triangles that contain ab are contracted to edges that contain c. The quality of the resulting surface depends on the prioritization of the edges for contraction and on the choice of the new points c. Wrap uses the quadratic error measure suggested in [23] for both purposes. The effectiveness of the method is illustrated in Figure 6 which shows the reconstruction of a foot surface from about 20 thousand scanned points. To highlight the preservation of shape detail, Figures 6 (b) and (c) zoom into the toe region of the model.

8 Analysis

The analysis component of Wrap is planned as a comprehensive suite of tools for analyzing and visualizing quality criteria of single models and deviations between different models of maybe the same shape. At the moment the only mature tool is a self-intersection test. Surface self-intersections may be created by thickening, and less likely by refining, relaxing, and decimating. The test checks every triangle against every other one and reports the number of improper intersections. Since Wrap deals with large numbers of triangles it would be painfully slow to explicitly check every pair. Wrap uses the multi-level divide-and-conquer algorithm of Edelsbrunner and Overmars [24] to first find intersecting pairs of smallest



(a) (b) (c)

Figure 6: Original data courtesy of Hugues Hoppe at Microsoft. (a) The reconstruction of a foot from about 20 thousand points. (b) Zoom into the toe region whose details require a higher variation of local curvature than the rest of the foot. (c) Decimation to 5% the original number of triangles preserves the shape of toes and nail lines.

bounding boxes and check triangles only if their boxes overlap. The triangles involved in improper intersections form surface patches, and the test reports the number of such patches, how many intersect themselves, and how many pairs intersect. Other than these numbers, the test returns with the detected patches as chosen regions. The rationale is that the user might want to eliminate the intersections, and probably the easiest way to do that is to apply a local relaxation that untwists and separates the patches by stretching.

9 Output

Wrap saves models in various industry-standard formats for engineering, architecture, and animation. The formats include 3DS DXF, IGES, IV, OBJ, STL, and VRML. The output is guaranteed to be topologically consistent, or in other words there are no cracks of gaps in the model description. This is a crucial requirement for rapid prototyping applications. In addition, Wrap has its own WRP format to save models which can be loaded at a later time for further processing.

Another format available from Wrap is spline output. The reconstructed model is sliced by planes normal to a specified direction, and each slice is saved as one or more closed spline curves. The model surface is then described by a sequence of such slices saved as an IGES file. This format can be read by other software packages joining the curves to form a spline surfaces. The slicing technique is also useful in turning an unorganized point set into one that is organized in regular intervals along curves in parallel planes.

10 Conclusion

This paper describes a new software package, Wrap, for automatic surface reconstruction from scanning data. It offers a cost-effective solution to physical design where mass customization of existing physical parts is essential. Wrap bings exceptional precision and versatility to 3D modeling. It automatically converts raw point data into accurate computerized surface and solid models — an industry first. This means that complicated and labor-intensive computations are now only a mouse-click away. Engineers, designers, animators, and soon physicians and dentists can quickly generate precise complex 3D models, measure them, and simulate forces to deform them through time. Automation, precision, and versatility set Wrap apart from its competitors in the market place. Further information and free evaluation software is available at `www.geomagic.com`.

References

[1] S. K. Lodha. Scattered data techniques for surfaces. *In* "Proc. Dagstuhl Conf. Scientific Visualization". Springer-Verlag, to appear.

[2] H. Edelsbrunner, D. G. Kirkpatrick and R. Seidel. On the shape of a set of points in the plane. *IEEE Trans. Inform. Theory* **IT-29** (1983), 551–559.

[3] H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Trans. Graphics* **13** (1994), 43–72.

[4] C. L. Bajaj, F. Bernardini and G. Xu. Automatic reconstruction of surfaces and scalar fields from 3D scans. *Comput. Graphics*, Proc. SIGGRAPH 1995, 109–118.

[5] J.-D. Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Trans. Graphics* **3** (1984), 266–286.

[6] H. Fuchs, Z. Kedem and S. P. Uselton. Optimal surface reconstruction from planar contours. *Comm. ACM* **20** (1977), 693–702.

[7] D. Meyers, S. Skinner and K. Sloan. Surfaces from contours. *ACM Trans. Graphics* **11** (1992), 228–258.

[8] J.-D. Boissonnat and B. Geiger. Three-dimensional reconstruction of complex shapes based on the Delaunay triangulation. *In* "Proc. Biomedical Image Process. Biomed. Visualization 1993", 964–975.

[9] W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. *Comput. Graphics* **21**, Proc. SIGGRAPH 1987, 163–169.

[10] H. Hoppe, T. de Rose, T. Duchamp, J. McDonald, and W. Stützle. Surface reconstruction from unorganized points. *Comput. Graphics*, Proc. SIGGRAPH 1992, 71–78.

[11] B. Curless and M. Levoy. A volumetric method for building complex models from range images. *Comput. Graphics*, Proc. SIGGRAPH 1996, 303–312.

[12] V. Krishnamurthy and M. Levoy. Fitting smooth surfaces to dense polygonal meshes. *Comput. Graphics*, Proc. SIGGRAPH 1996, 313–324.

[13] C. T. Loop. Smooth spline surfaces over irregular meshes. *Comput. Graphics*, Proc. SIGGRAPH 1994, 303–310.

[14] J. Peters. C^1-surface splines. *SIAM J. Numer. Anal.* **32** (1995), 645–666.

[15] M. Eck and H. Hoppe. Automatic reconstruction of B-spline surfaces of arbitrary topological type. *Comput. Graphics*, Proc. SIGGRAPH 1996, 325–334.

[16] C. L. Bajaj, J. Chen and G. Xu. Modeling with cubic A-patches. *ACM Trans. Graphics* **14** (1995), 103–133.

[17] B. Delaunay. Sur la sphère vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk* **7** (1934), 793–800.

[18] H. Edelsbrunner and N. R. Shah. Incremental topological flipping works for regular triangulations. *Algorithmica* **15** (1996), 223-241.

[19] M. A. Facello. Implementation of a randomized algorithm for Delaunay and regular triangulations in three dimensions. *Comput. Aided Geom. Design* **12** (1995), 349–370.

[20] H. Edelsbrunner and E. P. Mücke. Simulation of Simplicity: a technique to cope with degenerate cases in geometric algorithms. *ACM Trans. Graphics* **9** (1990), 66–104.

[21] C. T. Loop. Smooth subdivision surfaces based on triangles. M. S. thesis, Dept. Math., Univ. Utah, Salt Lake City, 1987.

[22] G. Taubin. A signal processing approach to fair surface design. *Comput. Graphics*, Proc. SIGGRAPH 1995, 351–358.

[23] M. Garland and P. S. Heckbert. Surface simplification using quadratic error metrics. *Comput. Graphics*, Proc. SIGGRAPH 1997, 209–216.

[24] H. Edelsbrunner and M. H. Overmars. Batched dynamic solutions to decomposable searching problems. *J. Algorithms* **6** (1985), 515–542.