A Triangulation-Based Object Reconstruction Method*

Fausto Bernardini
Chandrajit L. Bajaj
Jindong Chen†
Daniel R. Schikore

Department of Computer Sciences
Purdue University ‡

1 Introduction

Reconstructing the shape of a 3D object from a digital scan of its surface has a range of applications, such as reverse engineering, authoring 3D synthetic worlds, shape analysis, 3D faxing and tailor-fit modeling.

Input data might come in different forms, depending on the scanning device used. It is usually comprised of the location (x_i, y_i, z_i) of points on the surface of the object, and at times additional topological and geometric information, as well as measures of other physical properties. The sampling provided by recent scanning devices (such as the *laser range scanner*) is *dense*, in the sense that the resolution is much smaller than the size of shape features of interest. Often multiple scans are required to capture the entire object's surface. We make no assumptions on spatial relations among sample points, and assume that the input is a large, but unorganized, collection of measurements.

Our goal is to reconstruct a boundary representation of the object, based on implicit polynomial surface patches of low degree, that has the desired geometric continuity and approximates the data within a user-specified parameter ϵ. For a discussion of related prior work the reader is referred to [6].

In [1], we presented a method based on alpha-shapes, to build an initial piecewise-linear reconstruction, followed by an incremental, adaptive piecewise polynomial fitting of the signed distance function defined by the alpha-shape. The method relied on the user to select a good α-value. The final reconstructed model was represented as a collection of C^1-smooth implicit algebraic patches. A more up-to-date, detailed description of the algorithm can be found in [2].

In this paper, and the accompanying video presentation, we present a novel reconstruction technique (more details can be found in [6]), capable of handling piecewise-smooth objects. While we still use alpha-shapes and implicit algebraic patches in the new method, there are significant differences with our previous method. First, we devised an automatic selection method to find an optimal α-value. Second, we developed a heuristic-based approach, to improve the quality of the selected (regularized) alpha-shape. Third, instead of approximating the signed-distance function, we apply a mesh-reduction algorithm followed by an interpolatory A-patch fitting scheme.

*This research was partially supported by AFOSR grant F-49620-94-1-0080, NSF grant CCR 92-22467 and ONR grant N00014-94-1-0370

†Present address: Corporate Research & Technology, Xerox. Email: jchen@cp10.es.xerox.com.

‡West Lafayette, Indiana 47907-1398 USA

{bajaj,fxb,jdc,drs}@cs.purdue.edu
http://www.cs.purdue.edu/research/shastra

Computational Geometry 97 Nice France

2 Algorithm overview

An example of the reconstruction process is shown in Figure 1. Our algorithm is based on the following three phases:

1. Build an initial triangle mesh that interpolates all data points, approximating the object shape (Figure 1 (a)-(c)). Our approach is based on (regularized) alpha-shapes [8], and is capable of automatically selecting an optimal α-value and improving the resulting mesh in areas of insufficient sampling.

The resulting triangle mesh can be used to estimate normals at smooth vertices (by averaging the normals of incident triangles) and to detect sharp features (by looking at the dihedral angle formed by two adjacent triangles). Notice that for the dense surface sampling that we are interested in, these estimates are usually quite accurate. The use of more complex and accurate sharp-feature detection strategies will be investigated in the future.

2. Simplify the mesh to reduce the number of triangles, while guaranteeing good aspect-ratio of triangles, bounded distance of the data points from the reduced mesh, and feature preservation (Figure 1 (d)). The technique used in our paper has been extended from [4]. The edges and vertices of the reduced mesh are "tagged" as either *smooth* (the surface is C^1 continuous across it) or *sharp* (only C^0 continuity), and vertices are classified according to the type of incident edges and the number and type of estimated vertex normals (Figure 1 (e)).

3. The reduced mesh is used as the starting point for a polynomial-patch data fitting. For every triangle, we build an implicit Bernstein-Bézier patch of low degree which interpolates the vertices and vertex normals (if defined) and least-squares approximates data points in its vicinity. The

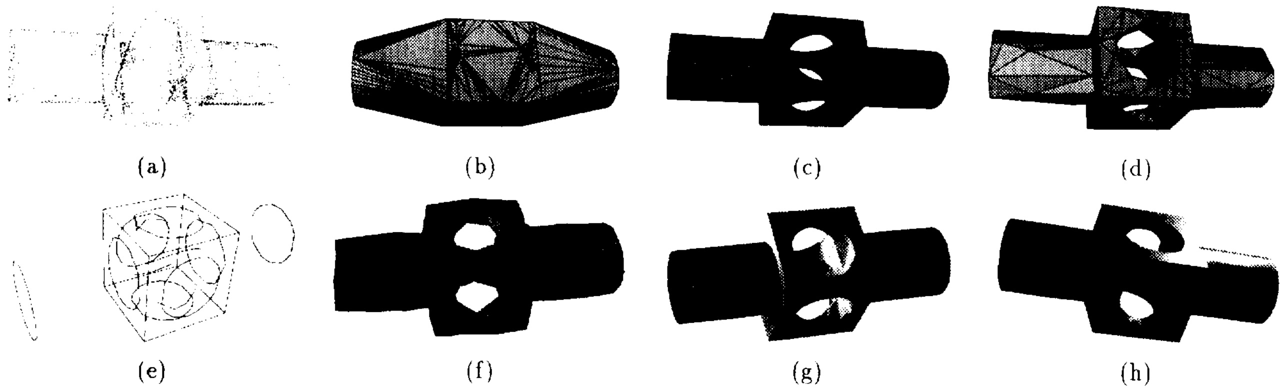

Figure 1: The complete reconstruction process. (a) Point sampling. (b) 3D Delaunay triangulation. (c) Alpha-solid. (d) Simplified mesh. (e) Sharp features. (f) Support mesh. (g) A-patches. (h) Reconstructed model.

algebraic patches used (cubic A-patches [3]) allow a simple formulation of C^1 continuity constraints between adjacent patches, and have been extended to allow the modeling of sharp features such as linear sharp edges, piecewise-planar curved creases and sharp corners (Figure 1 (f)-(h)).

3 Shape Reconstruction Using Alpha-Shapes

One of the most difficult problems of surface reconstruction from unorganized points is understanding how to connect the points so as to form a surface that has the same topological (e.g. number of handles) and geometric (e.g. depressions and protrusions) characteristics of the original. In particular, we look at the following problem ($D(K, B)$ is a suitably defined "distance" measure):

Let B be the boundary of a solid M, and $S \subset B$ a finite set of points (sampling). Construct a (geometric) simplicial complex K, such that $K^{(0)} = S$, K is homeomorphic to B, and $D(K, B) < \varepsilon$.

We call ρ-dense a sampling such that any sphere of radius ρ, centered on the surface of the object, contains at least one sample point. We prove in [5] that for a manifold whose radius of curvature and "feature-size" are larger than ρ, a ρ-dense sampling suffices to guarantee a homeomorphic and error-bounded reconstruction using alpha-shapes. While the theorems above give us sufficient conditions for a sampling to allow a faithful reconstruction using alpha-shapes, in practice one has to deal with less than ideal scans.

In general, i.e. when the conditions of the theorems above are not satisfied, an alpha-shape is a non-connected, mixed-dimension polytope. We define the alpha-solid to be the union of all tetrahedra in the 3D Delaunay triangulation T of S that are contained within a continuous envelope of alpha-shape triangles. The alpha-solid is clearly a homogeneously three-dimensional object. Observe also that it can be computed very efficiently from the underlying triangulation, by simply traversing the adjacency graph. Varying α, one obtains a finite collection of different alpha-solids, ranging from the empty set for sufficiently small values of α, to the convex hull of the set of points for α large enough. We can perform a binary search for the minimal α-value such that all points lie on the boundary or in the interior of the alpha-solid, and such that the alpha-solid is a closed, connected manifold. This search takes $O(\log n)$ time (where n is the size of S) because the number of possible α-values is bounded by the number of simplices in T, which is polynomial in n. Observe also that the convex hull satisfies the three properties above, so the algorithm always terminates successfully. For a sufficiently dense and uniform sampling of the object boundary the alpha-solid selected by this strategy is a good approximation of the object's shape. However, small, concave features may be occluded by unwanted tetrahedra, and some of the sampled points might lie in the interior of the alpha-solid.

Our criterion for improving the initial alpha-solid is based on the search for a subset of tetrahedra whose boundary interpolates all the points, and that maximizes the "smoothness" of the triangle mesh, defined as $\sum_{e_i} |\pi - \gamma_i|$, over all edges of the triangle mesh, where γ_i is the dihedral angle formed by the two triangles incident on edge e_i.

Searching for the global optimum for this optimization problem would be clearly computationally expensive, but in practice our alpha-solid is already a good approximation of the optimal polyhedron, and we only need to modify it where concave, high curvature features are present. We therefore resort to a simple greedy strategy, similar to the "sculpturing" approach proposed by Boissonnat [7]. However, we apply the iterative removal of tetrahedra only to locally improve the alpha-solid, rather than as a global strategy to extract an interpolating mesh from the 3D Delaunay triangulation. Figure 2 and Table 1 illustrate some examples of alpha-solids computed with the technique described above.

4 Mesh Simplification

Surface mesh simplification refers to a general category of techniques designed to generate compact, adaptive approximations of dense tessellated surfaces. A discussion of previous work in the field can be found in [4].

In this work, we adapt and improve the method of [4] to handle explicit edge feature detection and preservation. The resulting algorithm is able to maintain a strict bound on the distance between the original mesh and the surface mesh, in addition to maintaining sharp features in the reconstructed triangulation.

The simplification algorithm follows the basic strategy of other "vertex deletion" schemes, and is based on accumulated error bounds which are propagated from the original surface mesh through the successive simplified meshes produced by point deletion. A compact error representation

Object	Number of Points	Alpha-solid Time	Number of tetrahedra	Removed tetrahedra	Number of Triangles
Femur	9807	1.5	36182	3704	19610
Tibia	9200	1.4	33232	2172	18396
Fibula	8146	1.1	30876	2896	16288
Patella	2050	0.3	7536	683	4096
CSG Solid	13040	2.5	42507	2473	26088
Club	16864	4.1	58657	754	33142
Bunny	33123	19.6	127607	3761	66224
Mannequin	10392	2.1	35383	2077	19802

Table 1: Results of alpha-solid reconstruction. The table shows for each object, from left to right: (1) The number of points in the sampling; (2) The time, in minutes, required by the alpha-solid computation (including 3D Delaunay triangulation, computation of family of alpha-shapes, automatic selection of α-value, improvement by local sculpturing). All computations were carried out on a SGI Indigo2, with a 250MHz MIPS 4400 CPU; (3) The number of tetrahedra in the initial alpha-solid; (4) The number of tetrahedra removed by the heuristic; (5) The number of triangles in the boundary of the final reconstructed model.

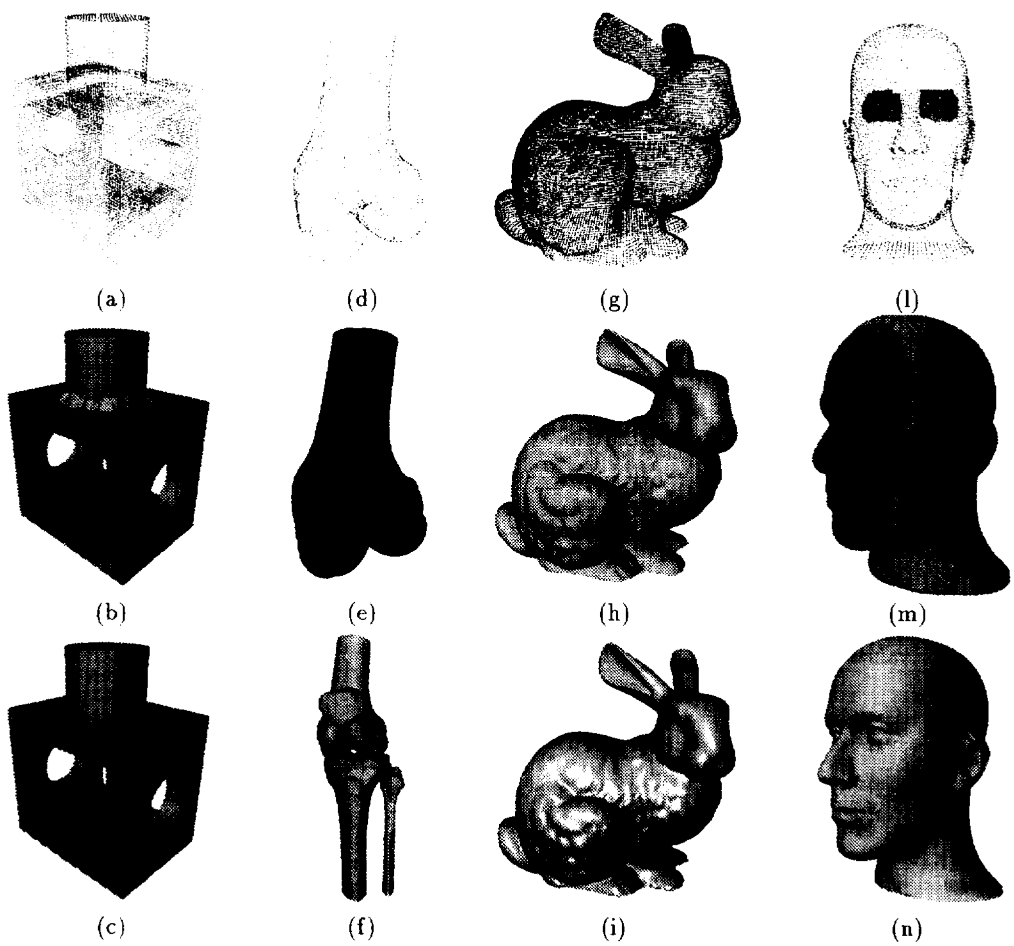

Figure 2: Example of reconstruction using alpha-shapes. (a)-(c) Random sampling of a solid generated with a commercial modeler. (a) Point sampling. (b) Selected alpha-shape. (c) Improved via local smoothness heuristic. (d)-(f) Reconstruction from isocontours of a CT scan (Visible Human Project). (d) Points. (e) Alpha-solid. (f) Reconstructed knee model. (g)-(i) Reconstruction from laser range data. (g) Data points. (h) Improved alpha-solid. (i) Phong rendering of the alpha-solid. (l)-(m) Use of weighted alpha-shapes for reconstruction from multiresolution scans. (l) Sampling. (m) Weighted points. (n) Alpha-solid.

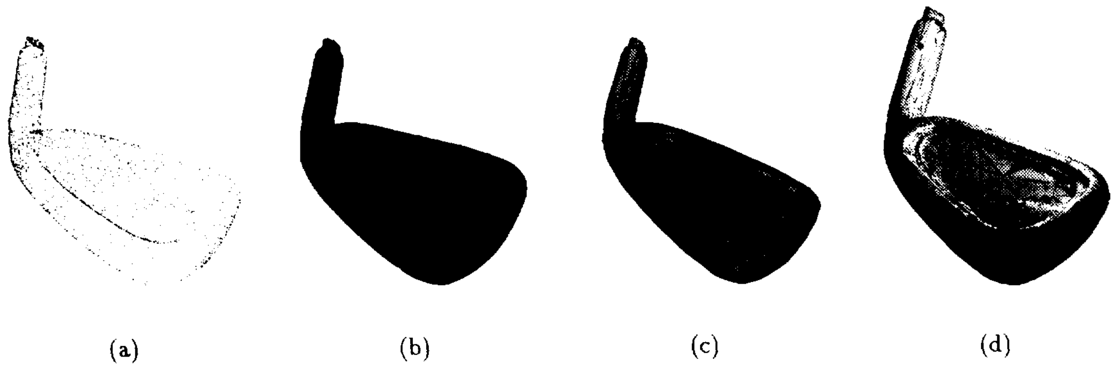

Figure 3: Reconstruction and piecewise-smooth A-patch fitting. (a) Scanned points. (b) alpha-solid. (c) Decimated mesh. (d) Reconstructed model (different colors identify different surface patches).

consisting of two scalar error bounds per triangle is used. The error values correspond to a bound on the error (geometric displacement) toward the outside (inside) of the object. These bounds effectively form an envelope surrounding the simplified mesh which is guaranteed to contain the original surface, thus maintaining a bound on the total amount of accumulated error through successive deletion of vertices. The algorithm can be summarized as follows:

1. Initialize errors on all triangles to 0
2. Initialize priority queue P of candidate vertices v_i
   (a) Classify v_i according to number of incident "sharp" edges
   (b) Compute an initial triangulation of the neighbors of v_i
   (c) Perform edge flipping to lower the error in the new triangulation
   (d) Assign priority based on introduced error associated with v_i
3. While next candidate vertex v from P does not violate error constraints
   (a) Delete v and incident triangles
   (b) Add new triangles
   (c) Update error values for new triangles
   (d) Update P

5 A-patch Fitting

Our A-patch fitting scheme interpolates the vertices (and estimated surface normals) of the simplified mesh computed as described above, and approximates the remaining data points. Features tagged as *sharp* during mesh simplification are retained in the resulting piecewise-smooth model. The fitting process begins with the construction of a tetrahedral mesh to act as support for the A-patches. Then, weights for each patch are set to interpolate vertices and sharp features, and least-squares approximate the remaining point. Finally, a fairing and fitting optimization can be applied to improve the quality of the reconstructed model.

The scheme used here follows Bajaj, Chen and Xu [3], and consists basically in building one tetrahedron for each side of every triangle of the mesh (*face-tetrahedra*) and four edge-tetrahedra to close the gaps.. Clough-Tocher splitting is only used in special cases. This scheme can be extended to accommodate sharp features of the following type:

1. Sharp corners (multiple normals defined, for example the corner of a cube).
2. Singular vertices (no normal defined, for example the apex of a cone).
3. Straight edges (two normals, one for each side, defined at each endpoint).
4. Planar or piecewise-planar curved ridges (two normals at each endpoint).

Some of the weights of each patch are only constrained in sign by the single-sheeted condition. Their value can be set by solving a least-squares problem that minimizes the local error-of-fit to the data points. An additional optimization step to improve the fitting and obtain a better fairing can be applied as a postprocessing to the piecewise surface.

Figure 3 shows an example of reconstruction using this algorithm.

References

[1] Bajaj, C., Bernardini, F., and Xu, G. Automatic reconstruction of surfaces and scalar fields from 3D scans. In *Computer Graphics Proceedings* (1995), Annual Conference Series. Proceedings of SIGGRAPH 95, pp. 109–118.

[2] Bajaj, C., Bernardini, F., and Xu, G. Reconstructing surfaces and surfaces on surfaces from 3D scans. *Algorithmica* (1997). To appear.

[3] Bajaj, C., Chen, J., and Xu, G. Modeling with cubic A-patches. *ACM Trans Graph. 14*, 2 (1995), 103–133.

[4] Bajaj, C., and Schikore, D. Error-bounded reduction of triangle meshes with multivariate data. In *Proceedings of SPIE Symposium on Visual Data Exploration and Analysis III* (Jan. 1996), SPIE, pp. 34–45.

[5] Bernardini, F., and Bajaj, C. Sampling and reconstructing manifolds using alpha-shapes. Tech. Rep. CSD-TR-97-013, Department of Computer Sciences, Purdue University, 1997. Revised version submitted for publication.

[6] Bernardini, F., Bajaj, C., Chen, J., and Schikore, D. Automatic reconstruction of 3D CAD models from digital scans. Tech. Rep. CSD-TR-97-012, Department of Computer Sciences, Purdue University, 1997. Submitted for publication.

[7] Boissonnat, J.-D. Geometric structures for three-dimensional shape representation. *ACM Trans. Graph. 3*, 4 (1984), 266–286.

[8] Edelsbrunner, H., and Mücke, E. P. Three-dimensional alpha shapes. *ACM Trans. Graph. 13*, 1 (Jan. 1994), 43–72.