Reconstructing Surfaces By Volumetric Regularization Using Radial Basis Functions

Huong Quynh Dinh[1], Greg Turk[1], and Greg Slabaugh[2]

[1]Graphics, Visualization, and Usability Center
College of Computing
Georgia Institute of Technology

[2]Center for Signal and Image Processing
School of Electrical and Computer Engineering
Georgia Institute of Technology

Abstract— **We present a new method of surface reconstruction that generates smooth and seamless models from sparse, noisy, non-uniform, and low resolution range data. Data acquisition techniques from computer vision, such as stereo range images and space carving, produce 3D point sets that are imprecise and non-uniform when compared to laser or optical range scanners. Traditional reconstruction algorithms designed for dense and precise data do not produce smooth reconstructions when applied to vision-based data sets. Our method constructs a 3D implicit surface, formulated as a sum of weighted radial basis functions. We achieve three primary advantages over existing algorithms: (1) the implicit functions we construct estimate the surface well in regions where there is little data; (2) the reconstructed surface is insensitive to noise in data acquisition because we can allow the surface to approximate, rather than exactly interpolate, the data; and (3) the reconstructed surface is locally detailed, yet globally smooth, because we use radial basis functions that achieve multiple orders of smoothness.**

Index terms: **regularization, surface fitting, implicit functions, noisy range data**

I. Introduction

The computer vision community has developed numerous methods of acquiring three dimensional data from images. Some of these techniques include shape from shading, depth approximation from a pair of stereo images, and volumetric reconstruction from images at multiple viewpoints. The advantage of these techniques is that they use cameras, which are inexpensive resources when compared to laser and optical scanners. Because of the affordability of cameras, these vision-based techniques have the potential to enable the creation of digital models by home computer users who may not have professional CAD training. On the other hand, models in popular use in the entertainment industry (animation and gaming applications), video and image editing, and computer graphics research come from dense laser scans or medical scans, not from vision-based techniques. There are significant differences in terms of quality and accuracy between data sets obtained from active scanning technology (e.g. optical, laser, and time-of-flight range scanners) and passive scanning technology (e.g. shape from shading, voxel coloring) that use only images and camera calibration to obtain 3D point sets. Many of the well-known and often used reconstruction algorithms were designed to generate surfaces from dense and precise data such as those obtained from active scanners. These methods are not robust to the challenges posed by data obtained from passive scanning technology. The aim of our method is to be able to reconstruct smooth and continuous surfaces from the more challenging vision-based data sets.

The new approach presented in this paper constructs a 3D implicit function from vision-based range data. We use an analytical implicit representation that can smoothly interpolate the surface where there is little or no data, that is compact when compared to discrete volumetric distance functions, and that can either approximate or interpolate the data. The resulting surfaces are inherently manifold, smooth, and seamless. Implicit surfaces are well-suited for operations such as collision detection, morphing, blending, and modeling with constructive solid geometry because they are formulated as a single analytical function, as opposed to a piecewise representation such as a polygonal model or a dense volumetric data set. Implicit surfaces can also accurately model soft and organic objects and can easily be converted to a polygonal model by iso-surface extraction.

We construct an implicit surface using volumetric regularization. This approach is based on the variational implicit surfaces of Turk and O'Brien [48]. Our implicit function consists of a sum of weighted radial basis functions that are placed at surface and exterior constraint points defined by the data set. The weights of the basis functions are determined by solving a linear system of equations. We can approximate the data set by relaxing the linear system through volumetric regularization. The ability to choose whether to approximate or interpolate the data is especially advantageous in the presence of noise. Surface detail and smoothness are obtained by using basis functions that achieve multiple orders of smoothness.

Our main contributions are: (1) introducing the use of variational implicit surfaces for surface reconstruction from vision-based range data, (2) the application of a new radial basis function that achieves multiple orders of smoothness, (3) enhancement of fine detail and sharp features that are often smoothed-over by the variational implicit surfaces, and (4) construction of approximating, rather than interpolating surfaces to overcome noisy data.

The remainder of the paper is organized as follows: in Section II, we review related work in surface representation and reconstruction. We give an overview of our approach in III. In Section IV, we introduce volumetric regularization and describe our approach to constructing approximating surfaces using the variational implicit surface representation. In Section V, we introduce a radial basis function that achieves multiple orders of smoothness. In Section VI, we discuss sampling issues and the preservation of topology in

our framework. Results from synthetic range images and from real space carved data sets are shown in Section VII.

II. Related Work

Our approach to surface reconstruction can be compared to previous works in the areas of shape representation, reconstruction, smoothing, and surface regularization. The large number of published methods in these areas makes it nearly impossible to perform a comprehensive survey. Instead, we describe some of the more well-known approaches, with a bias towards those more closely related to our own approach. Table I summarizes the comparison between related reconstruction algorithms and our own.

A. Surface Representation

Three general classes of surface representations include discrete, parametric, and implicit approaches. Discrete forms, such as a collection of polygons and point samples, are the most widely used representations. The primary disadvantages associated with them are that they are verbose, that they can only approximate smooth surfaces, and that they have fixed resolution. In contrast, parametric surfaces, such as B-splines and Bezier patches, may be sampled at arbitrary resolution and can be used to represent smooth surfaces. The main drawback of paramteric surfaces is that several parametric patches need to be combined to form a closed surface, resulting in seams between the patches. Implicit representations, on the other hand, do not require seams to represent a closed surface. Implicit representations come in both analytical and discrete sampled forms. Analytical representations, such as our own, are more compact than sampled representations. Examples of sampled implicit functions include gridded volumes and octree representations such as those used by Szeliski et al. [39], Frisken et al. [18], and Curless and Levoy [12].

B. Surface Reconstruction

In this section, we discuss the more popular reconstruction algorithms. The shape reconstruction methods we describe include range data merging and mesh reconstruction, region growing, algorithms based on computational geometry, and algebraic fitting.

Although our work does not focus on reconstructing surfaces from dense and precise range data, methods that merge multiple range images and reconstruct smooth meshes address issues similar to our own. Issues that arise in such work include merging multiple range images, closing of gaps in the reconstruction, and handling of outliers. Curless and Levoy [12] and Hilton et al. [20] construct signed distance functions from the range images and obtain a manifold surface by iso-surface extraction. Soucy and Laurendeau [37] and Turk and Levoy [47] merge triangulations of the range points. Note that all of these methods require range data using structured light that is much more accurate than can be measured passively using photographs alone.

Another approach is region growing, and examples include Hoppe's work on surface reconstruction [21] and Lee,

Tang and Medioni's work on tensor voting [26,40]. Hoppe uses a plane is fitted to a neighborhood around each data point, providing an estimate of the surface normal for the point. The surface normals are propagated using a minimal spanning tree, and then a signed distance function is contructed in small neighborhoods around the data points. Lee and Medioni's tensor voting method is similar in that neighboring points are used to estimate the orientations of data points. The tensor is the covariance matrix of the normal vectors of a neighborhood of points. Each data point votes for the orientation of other points in its neighborhood using its tensor field. In [40], the surface is reconstructed by growing planar, edge, and point features until they encounter neighboring features. Both methods described above are sensitive to noise in the data because they rely on good estimates for the normal vector at each data point.

Several algorithms based on computational geometry construct a collection of simplexes that form the shape or surface from a set of unorganized points. These methods exactly interpolate the data — the vertices of the simplexes consist of the given data points. A consequence of this is that noise and aliasing in the data become embedded in the reconstructed surface. Of such methods, three of the most successful are Alpha Shapes [15], the Crust algorithm [1], and the Ball- Pivoting algorithm [4]. In Alpha shapes, the shape is carved out by removing simplexes of the Delaunay triangulation of the point set. A simplex is removed if its circumscribing sphere is larger than the alpha ball. In the Crust algorithm, Delaunay triangulation is performed on the original set of points along with Voronoi vertices that approximate the medial axis of the shape. The resulting triangulation distinguishes triangles that are part of the object surface from those that are on the interior because interior triangles have a Voronoi vertex as one of their vertices. Both the Alpha Shapes and Crust algorithms need no other information than the locations of the data points and perform well on dense and precise data sets. The object model that these approaches generate, however, consists of simplexes that occur close to the surface. The collection of simplexes is not a manifold surface, and extraction of such a surface is a non-trivial post-processing task. The Ball-Pivoting algorithm is a related method that avoids non-manifold constructions by growing a mesh from an initial seed triangle that is correctly oriented. Starting with the seed triangle, a ball of specified radius is pivoted across edges of each triangle bounding the growing mesh. If the pivoted ball hits vertices that are not yet part of the mesh, a new triangle is instantiated and added to the growing mesh. In Figure 1 (right panel), the Crust algorithm is applied to real range data obtained from the generalized voxel coloring method of [11]. Although the general shape of the toy dinosaur is recognizable, the surface is rough due to the noisy nature of the real range data.

Many algebraic methods avoid creating noisy surfaces by fitting a smooth function to the data points, and by not requiring that the function pass through all data points. The reconstructed surface may consist of a single global

TABLE I

COMPARISON OF RELATED WORKS

Methods	Shape Representation	Arbitrary Topology	Complex Models	Robust to Noise	Fills Gaps
Distance Fields	discrete	yes	yes	no	no
Region Growing	piecewise continuous	yes	yes	no	no
Computational Geometry	piecewise continuous	yes	yes	no	no
Algebraic Methods	analytical	yes	no	yes	yes
Deformable Superquadrics	analytical	no	no	yes	yes
Volumetric Regularization	analytical	yes	yes	yes	yes

function or many functions that are pieced together. Examples of reconstruction by global algebraic fitting are the works of Taubin [41, 42], Gotsman and Keren [22, 23], and Blane et al. [5]. Taubin fits a polynomial implicit function to a point set by minimizing the distance between the point set and the implicit surface. In [41], Taubin develops a first order approximation of the Euclidean distance and improves the approximation in [42]. Gotsman and Keren create parameterized families of polynomials that satisfy desirable properties, such as fitness to the data or continuity preservation. Such a family must be large so that it can include as many functions as possible. This technique leads to an over- representation of the subset, in that the resulting polynomial will often have more coefficients for which to solve than the simpler polynomials included in the subset, thus requiring additional computation. Blane et al. performs polynomial fitting of points on a zero level set and (for stability) fits points on two additional level sets close to the zero level set — one internal and one external level set. The primary limitation of global algebraic methods is their inability to reconstruct complex models. The highest degree polynomials that have been demonstrated are around degree 12, and this is far too small to represent complex shapes.

In [3], Bajaj overcomes the complexity limitation by constructing piecewise polynomial patches (called A-patches) that combine to form one surface. Bajaj uses Delaunay triangulation to divide the point set into groups delineated by tetrahedrons. An A-patch is formed by fitting a Bernstein polynomial to the data points within each tetrahedron. By constructing a piecewise surface, Bajaj's approach loses the compact characteristic of a global representation, and operations such as collision detection, morphing, blending, and modeling with constructive solid geometry become more difficult to perform since the representation is no longer a single analytical function.

Examples of algebraic methods developed earlier in the vision community that provide both smooth global fitting and accurate local refinement include the works of Terzopoulos and Metaxas on deformable superquadrics [46] and Pentland and Sclaroff on generalized implicit functions [32,34]. Both methods use superquadric ellipsoids as the global shape and add local deformations to fit the data points. Terzopoulos and Metaxas separate the re-

constructed model into global parameters defined by the superquadric coefficients, and local displacements defined as a linear combination of basis functions. The global and local deformation parameters are solved using dynamics. Pentland and Sclaroff define a generalized implicit model that consists of a superquadric ellipsoid and a modal deformation matrix. The modal deformation parameters are found by iteratively finding the minimum RMS error to the data points. The residual error after the deformation parameters have been found are incorporated into a displacement map to better fit the data. As with most algebraic methods, the drawback of these techniques is their inability to handle arbitrary topology.

Our approach is similar to global algebraic fitting in that we construct one global implicit function, although our basis functions are not polynomials. Previous work that is most closely related to our own are methods based on *regularization* which we describe next.

C. Surface Regularization

Surface reconstruction is an ill-posed inverse problem because there are infinitely many surfaces which may pass through a given set of points. *Surface regularization* restricts the class of permissible surfaces to those which minimize a given energy functional. Terzopoulos pioneered finite-differencing techniques to compute approximate derivatives used in minimizing the thin-plate energy functional of a height-field. He developed computational molecules from the discrete formulations of the partial derivatives and uses a multi-resolution method to solve for the surface. Boult and Kender compare classes of permissible functions and discuss the use of basis functions to minimize the energy functional associated with each class. Using synthetic data, they show examples of overshooting surfaces that are often encountered in surface regularization. As exemplified by these two methods, many approaches based on surface regularization are restricted to height fields.

In [16], Fang and Gossard reconstruct piecewise continuous parametric curves. The advantage of parametric curves and surfaces over height-fields is the ability to represent closed curves and surfaces. Each curve in their piecewise reconstruction minimizes a combination of first, second, and third order energies. Unlike previous examples, the deriva-

tive of the curve in this method is evaluated with respect to the parametric variable. Each curve is formulated as a sum of weighted basis functions. Fang and Gossard show examples using Hermite basis. The approach we present in this paper has similar elements. We also use basis functions to reconstruct a closed surface which minimizes a combination of first, second, and third order energies. We differ from the previous work in that we reconstruct complex 3D objects using a single implicit function; we perform volumetric rather than surface regularization; and we use energy-minimizing basis functions as primitives.

Because our method of reconstruction applies regularization, comparisons can also be made to other classes of stabilizers (or priors) and other energy-minimizing basis functions. We postpone the discussion of other prior assumptions and resulting basis functions to Section V where we introduce the multi-order basis function that we use to reconstruct implicit surfaces. The use of radial basis functions for graphical modelling was introduced by Blinn[6]. Since then, methods have been published that use this surface representation for surface reconstruction, including Muraki[29] and Savchenko[33]. Our work differs from these methods in that we use a basis function that minimizes multiple energies in 3D, including thin-plate and membrane. Comparison with reconstructions using Gaussian and thin-plate basis functions will be addressed in Section V-A.

D. Surface Smoothing

A closely related topic is that of mesh smoothing, where a low-pass filter is applied to a mesh to reduce noise. Examples of this method include the works of Taubin et al. [43] and Desbrun et al. [13]. The primary drawback of mesh smoothing methods is that they require an initial mesh. Our approach creates and smoothes a surface in one step.

Regularization and smoothing are closely tied. The relationship between regularization and smoothing has been studied by many, including Girosi et al. [19], Terzopoulos [44], and Nielson et al [30]. In Section V-A, we use a volumetric data set to demonstrate the similarity between regularization and spatial smoothing. Our reconstruction of the data set (which uses no information about the gridded structure of the volume) comes very close to a model obtained by spatially smoothing the 3D data set prior to iso-surface extraction. The advantage of our reconstruction algorithm is that it may be applied to data sets that are unstructured and non-uniform. Spatial smoothing cannot easily be applied to such data.

E. Active versus Passive Scanning Technology

Many of the methods described above reconstruct surfaces from dense and precise data obtained from active scanning. In this paper, we address the problem of reconstructing smooth and seamless surfaces using data obtained from passive scanning. In passive scanning, only images and camera calibration information are used to obtain 3D point sets. Active scanning technology (e.g. light stripe and time-of-flight range scanners) differ from passive
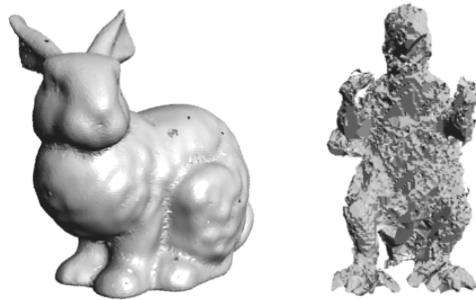

Fig. 1. Left: Stanford Bunny data set from cyberware scanner. Right: The toy dinosaur data set from voxel coloring. Both reconstructions were generated using the Crust algorithm. The dinosaur data set obtained from passive scanning is noisier and lower in resolution.

scanning technology (e.g. shape from shading, voxel coloring) in terms of quality, accuracy, and cost. The typical scanning resolution of cyberware scanners is 0.5 mm, while that of the voxel coloring data sets we use as examples in this paper are approximately 1.25 mm. Data from passive scanning is comparatively more noisy, more non-uniform, and more sparse than data from active scanners. In particular, surface reconstruction methods such as [12, 20, 47, 37] are not suited for creating models from data captured using passive scanning techniques.

Figure 1 is a comparison between data sets obtained from laser scanners and that obtained from voxel coloring. Both data sets were reconstructed using the Crust algorithm of Amenta et al. which exactly interpolates all data points. The toy dinosaur data set obtained from voxel coloring is significantly lower in resolution and accuracy than the Stanford Bunny obtained using a cyberware scanner. The primary advantage of passive scanning methods is the low cost of digital cameras (less than $1000) that are used to capture the images. Camera calibration is obtained using a calibration grid that is captured in the images. In contrast, the current cost of active range scanners is from $10,000 to over $100,000.

III. Overview of the Approach

Our approach to surface reconstruction is based on creating a single implicit function $f(\mathbf{x})$ by summing together a collection of weighted radial basis functions. We adopt the convention that the implicit function is positive inside the surface, zero on the surface, and negative outside the surface. The nature of the radial basis functions that are used is important to the quality of the reconstructions, and we discuss the basis function selection in detail in Section V. As input to implicit function creation, our method requires a collection of constraint points $\mathbf{c}_i$ that specify where the function should take on particular values. Most of the constraint points come directly from the input data, and these are points where the implicit function should take on the value zero. We call these 3D locations *surface constraints*. In addition, our method requires that some 3D

points be explicitly identified as being outside the surface, and we call these *exterior constraints*. Scattered data approximation of the surface and exterior constraints is then used to construct the implicit function. In Section IV-B we describe the details of the implicit formulation, and in Section VI we discuss the sampling of surface and exterior constraints from the measured data of an object.

IV. Volumetric Regularization

The surface reconstruction technique that we present in this paper is an extension of the variational implicit surfaces of [48]. This approach is based on the calculus of variation and is similar to surface regularization in that it minimizes an energy functional to obtain the desired surface. Unlike surface regularization, however, the energy functional is defined in R^3 rather than R^2. Hence, the functional does **not** act on the space of surfaces, but rather, on the space of 3D functions. We call this *volumetric regularization*. We use volumetric regularization to obtain a smooth 3D implicit function whose zero level set is our reconstructed surface. By Sard's theorem [8, 17], the set of nonregular values of such a smooth implicit function is a null set. Hence, the surface described by the zero level set of our implicit function does not contain pathological, or non-differentiable, points. In this section, we describe how we construct an approximating surface and obtain the implicit function representing the surface using volumetric regularization.

A. Approximation vs. Interpolation

Scattered data interpolation is the process of estimating previously unknown data values using neighboring data values that are known. In the case of surface reconstruction, the surface passes exactly through the known data points and is interpolated between the data points. Data interpolation is appropriate when the data values are precise. In vision-based data, however, there is some uncertainty in the validity of the data points. Using data interpolation to construct the surface is no longer ideal because the surface may not actually pass exactly through the given data points. This is precisely the problem with algorithms from computational geometry that generate polygonal meshes using data points as the vertices of the mesh. If the uncertainty of the data points is known, a surface that better represents the data would pass close to the data points rather than through them. Constructing such a surface is known as *data approximation*. Many vision-based techniques for capturing 3D surface points have an associated error distribution for the data points. In this section, we discuss how data approximation is achieved in our framework using volumetric regularization.

In regularization, the unknown function is found by minimizing a cost functional, H, of the following form:

$$H[f] = \beta[f] + \frac{1}{\lambda} \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i))^2 \qquad (1)$$

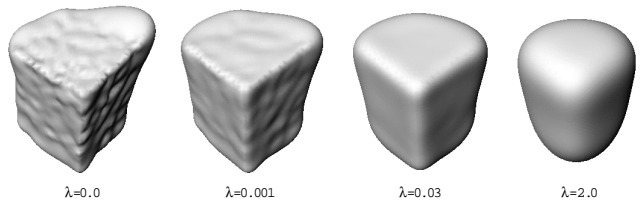In the above equation, f is the unknown implicit surface

Fig. 2. Reconstruction of a synthetic range image of a cube corner using various values of λ.

function; $\beta[f]$ is the smoothness functional, such as thin-plate; n is the number of constraints, or observed data points; y_i are the observed values of the data points at locations $\mathbf{x}_i$; and λ is a parameter (often called the *regularization parameter*) to weigh between fitness to the data points and smoothness of the surface. We can allow the surface to pass close to, but not necessarily through, the known data points by setting $\lambda > 0$. When $\lambda = 0$, the function interpolates the data points. The λ values may be assigned according to the noise distribution of the data acquisition technique. Figure 2 shows the results of applying different λ values on the same data set. As λ approaches zero, the surface becomes rougher because it is constrained to pass closer to the data points. At $\lambda = 0$, the surface interpolates the data, and overshoots are much more evident. At larger values of λ, the reconstructed model is smoother and approaches an amorphous bubble.

B. A Solution to the Regularizing Cost Functional

Derivations presented in [19, 49] show that the cost functional given in Equation 1 is minimized by a sum of weighted radial basis functions as shown below:

$$f(\mathbf{x}) = P(\mathbf{x}) + \sum_{i=1}^{n} w_i \phi(|\mathbf{x} - \mathbf{c}_i|) \qquad (2)$$

In the above equation, $f(\mathbf{x})$ is an implicit function that evaluates to zero on the surface, negatively outside, and positively inside; ϕ is the radially symmetric basis function; n is the number of basis; $\mathbf{c}_i$ are the locations of the centers of the basis; and w_i are the weights for the basis. In [48], Turk and O'Brien center a basis function at each constraint point. We do the same in this work. The constraints may be points on the surface of the object to be reconstructed or points external to the object. The polynomial term, $P(\mathbf{x})$, spans the null space of the basis function. For thin-plate energy, the polynomial term consists of linear and constant terms because thin-plate energy consists of second order derivatives. In 3D where $\mathbf{x} = (x, y, z)$, the polynomial term for thin-plate is $P(\mathbf{x}) = p_0 + p_1 x + p_2 y + p_3 z$. The unique implicit function is found by solving for the weights, w_i, of the radial basis functions and for the coefficients, $p_0, p_1, p_2,$ and p_3, of $P(\mathbf{x})$. The unknowns are solved by constructing the following linear system, formed by applying Equation 2 to each constraint, $\mathbf{c}_i$.

$$
\begin{bmatrix}
\phi(r_{11}) + \lambda_1 & \cdot & \phi(r_{1n}) & 1 & \mathbf{c}_1 \\
\vdots & & \vdots & 1 & \vdots \\
\phi(r_{n1}) & \cdot & \phi(r_{nn}) + \lambda_n & 1 & \mathbf{c}_n \\
1 & 1 & 1 & 0 & 0 \\
\mathbf{c}_1 & \cdot & \mathbf{c}_n & 0 & 0
\end{bmatrix}
\begin{bmatrix}
w_1 \\ \vdots \\ w_n \\ p_0 \\ \mathbf{p}
\end{bmatrix}
=
\begin{bmatrix}
f(\mathbf{c}_1) \\ \vdots \\ f(\mathbf{c}_n) \\ 0 \\ 0
\end{bmatrix}
$$
$$
r_{ij} = |\mathbf{c}_i - \mathbf{c}_j| \tag{3}
$$

In the above equation, p_0 and $\mathbf{p} = (p_1, p_2, p_3)$ are coefficients of $P(\mathbf{x})$. The function value, $f(\mathbf{c}_i)$, at each constraint point is known since we have defined the constraint points to be on the surface or external to the object. $f(\mathbf{c}_i) = 0$ for all $\mathbf{c}_i$ on the surface. All exterior constraints are placed at the same distance away from the surface constraints and are assigned a function value of -1.0 (more details will be given in Section VI-A on selection of exterior constraints). Notice that in the above system matrix, λ appears on the diagonal. By increasing the value of λ, the system matrix becomes better conditioned because it becomes more diagonally dominant. The addition of λ does not invalidate Equation 2 because $\lambda \sum_{i=1}^{n} w_i = 0$ (as seen in row $n+1$ of the matrix). The use of λ for trading off interpolation and approximation is found in numerous other publications, including those of Girosi et al. [19], Yuille et al. [51], and Wahba [49] where a detailed derivation can be found.

It is possible to assign distinct λ values to individual constraints. In this case, $\sum_{i=1}^{n} \lambda_i w_i \neq 0$, but instead, becomes part of the constant in the null space term, $P(\mathbf{x})$. This flexibility is especially important when we use exterior constraints because they are added only to provide orientation to the surface but do not represent real data. In practice, we have found that λ works well as a semi-global regularizing parameter, where one λ value is used for all surface constraints, and another for all exterior constraints. Using one λ value for all surface constraints is appropriate when the spatial distribution of noise is isotropic. This is a reasonable model for many vision-based data sets including the voxel-coloring data set that we later use as examples. With other noise models, it may be more appropriate to use λ as a local fitting parameter by assigning a λ value for each surface constraint based on the confidence measurement of the point. A large λ value such as 2.0 is often used for exterior constraints, while small values such as 0.001 is often used for surface constraints. This choice of λ for surface constraints was found through measures of fitness and curvature applied to the voxel coloring data set of a toy dinosaur. We found that a practical upper bound for λ for surface constraints from these types of data sets is 0.003. A detailed description of the fitness measures and results for various values of λ can be found in our technical report [14].

The implicit formulation described by Equation 2 has been used in a number of previous work, including those [6, 9, 28, 29, 31, 33, 48, 50, 51]. In [6, 29, 51], the basis function, ϕ, was a Gaussian, while in [9, 31, 33, 48, 50], ϕ inherently minimized thin-plate energy. In [6, 29], the basis functions were not centered at surface data points and

regularization was not applied to obtain the weights for the implicit function. Instead, Muraki iteratively added Gaussian basis functions until a sufficiently close fit is obtained. In [28, 48, 50], reconstructions were performed on accurate, dense cyberware scanned data. Hence, regularization was not necessary and simply using basis functions which minimize a desired energy was sufficient. In the next section, we compare the various choices of ϕ and discuss our selection of a basis function that minimizes multiple orders of energy.

Figure 3 is a comparison of reconstructions of a toy dinosaur. The Crust algorithm was used to reconstruct the surface shown in (a) which exactly interpolates all 20,120 data points; thin-plate basis functions were used to construct the interpolating implicit surface shown in (b); and in (c), thin-plate basis functions were used to construct the approximating implicit surface with λ set to 0.001. Only 3000 surface and 264 exterior constraints were used to reconstruct the implicit models. The approximating thin-plate surface is much smoother than either of the other two surfaces. The overshoots are less apparent, and there are fewer protruding bumps and fewer small pockets embedded in the surface. Unfortunately, the toy dinosaur's features are blobby and amorphous, especially at the feet and hands. Distinct limbs, such as the feet and tail, are fused together. It is apparent from this result that the thin-plate basis function used by Turk and O'Brien generates models which are too blobby.

V. A Radial Basis Function for Multiple Orders of Smoothness

The results in Figures 3(a), (b), and (c) show that a balance is needed between a tightly fitting, or *shrink-wrapped*, surface, and a smooth surface. A tightly fitting surface separates the features of the model but is prone to jagged artifacts. For example, the Crust reconstruction, shown in Figure 3(a), is an exact fit to the data with no smoothness constraint. On the other hand, a smooth surface may become too blobby as seen in Figures 3(b) and (c), which show that minimizing the thin-plate energy alone is not sufficient to produce a surface that separates features well and is locally detailed.

In [10], Chen and Suter derive radial basis functions for the family of Laplacian splines. The basis functions are comprised of $|r|^k$, $|r|^k log|r|$, exponential, and Bessel function terms, where r is the distance from the center of the radially symmetric basis. The value of k depends on the dimension and order of smoothness. Turk and O'Brien use $\phi(r) = |r|^2 log|r|$ for 2D thin-plate interpolation, and $\phi(r) = |r|^3$ for 3D thin-plate interpolation. Figure 4(a) shows that these functions exhibit global influence because the value of the function tends toward infinity as the distance from its center increases. The system matrix, which consists of the evaluation of the basis function at distances between pairs of constraints, is dense because constraint points are uniformly spread across the region of interest.

First, second, and third order energy-minimizing splines are also members of the family of Laplacian splines. Thin-
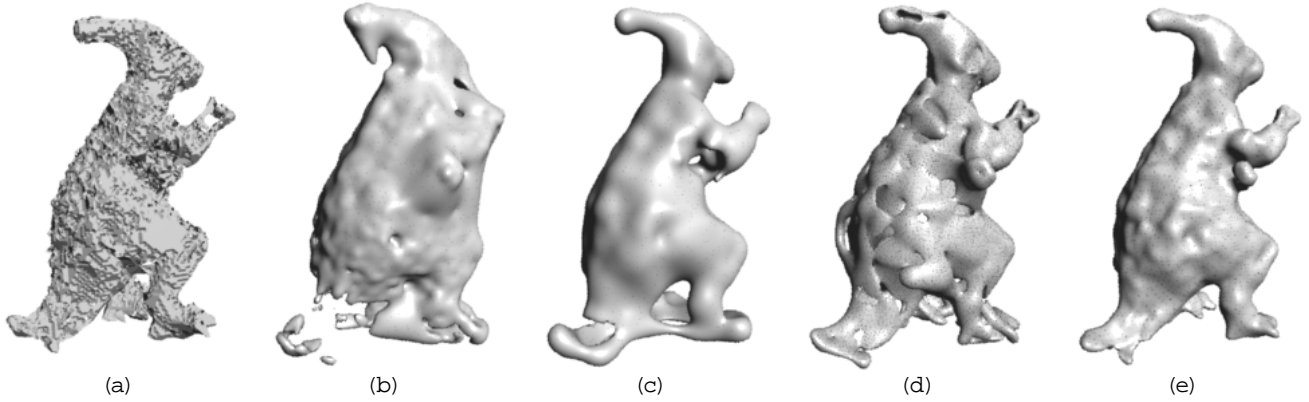
Fig. 3. Reconstructions of the toy dinosaur. (a) Crust reconstruction. (b) Exact interpolation using thin-plate basis function. (c) Surface approximation using thin-plate basis function. (d) Surface approximation using Gaussian basis function. (e) Surface approximation using multi-order basis function.
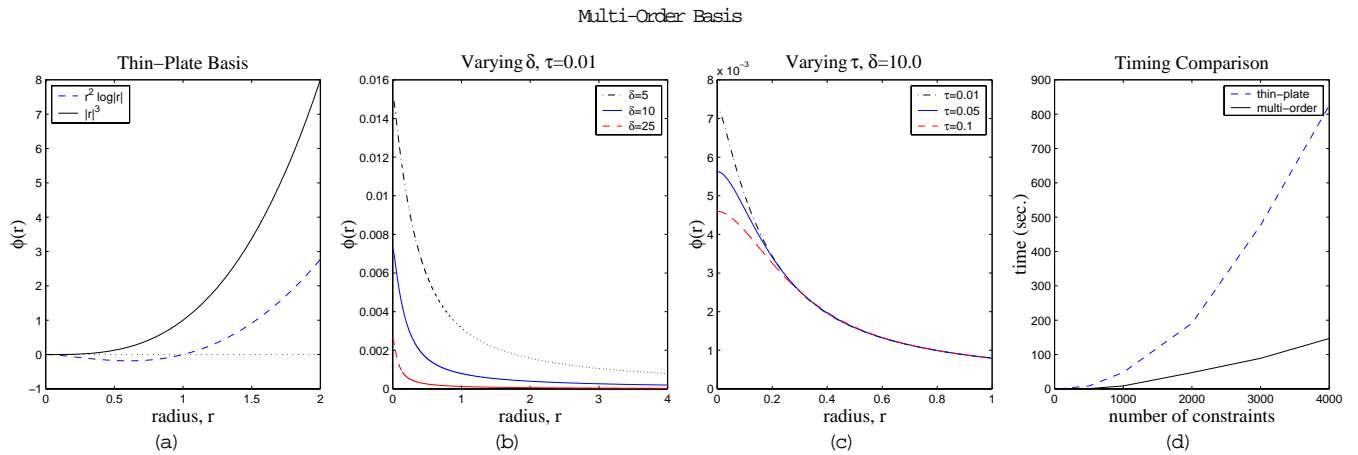

Fig. 4. (a) Cross sections of radially symmetric basis functions for $|r|^2 log|r|$ and $|r|^3$. (b) and (c) Cross sections of basis functions for a combination of first, second, and third order smoothness for various values of δ and τ. (d) Comparison of running times to solve for weights for the thin-plate and and for the multi-order basis functions.

plate energy is equivalent to second order energy, and membrane to first order energy. Surprisingly, a radial basis function that minimizes a combination of first, second, and third order energies quickly falls toward zero, yielding a better conditioned system matrix than one that minimizes thin-plate energy alone. In [38], Suter and Chen used basis functions that minimize multiple orders of smoothness (beyond the first and second order) to reconstruct human cardiac motion. They found that a model minimizing third and fourth order energy resulted in the smallest RMS error. They concluded that basis functions that minimize more than just the first and/or second order energy generate more accurate reconstructions. In addition, as the space dimension increases, the order of continuity of the thin-plate spline at data points decrease. Suter and Chen show that in 3D, the thin-plate spline basis has discontinuous first order derivatives at the data points. We chose to use a basis that achieves first, second, and third order smoothness because, unlike motion, object surfaces may contain sharp features that are C^1 discontinuous. The resulting implicit function has continuous derivatives due to

the additional third order smoothness (although, the isosurface may not have continuous derivatives). The geometric analogy to minimizing third order energy is curvature continuity. It has been shown in previous work by Fang and Gossard [16] that including curvature continuity results in improved curve and surface fitting. Terzopoulos also speculates on the use of curvature continuous stabilizers in [44].

In [10], Chen and Suter derive such a basis, using a smoothness functional comprised of the first, second, and third order Laplacian operator. The associated partial differential equation is similar to Laplace's equation $-\Delta f = 0$, but also has higher order terms:

$$-\delta\Delta f + \Delta^2 f - \tau\Delta^3 f = 0 \qquad (4)$$

In the above equation the Laplacian operator Δ in 3D is:

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} \qquad (5)$$

In Equation 4, δ controls the amount of first order smoothness, and τ controls the amount of third order smoothness. The balance between δ and τ controls the

amount of second order smoothness. The radial basis function that inherently minimizes the above energy functional in 3D as derived in [10] is:

$$\phi(r) = \frac{1}{4\pi\delta^2 r}(1 + \frac{we^{-\sqrt{v}r}}{v-w} - \frac{ve^{-\sqrt{w}r}}{v-w})$$
$$v = \frac{1+\sqrt{1-4\tau^2\delta^2}}{2\tau^2} \qquad w = \frac{1-\sqrt{1-4\tau^2\delta^2}}{2\tau^2} \tag{6}$$

In the above equations, r is the distance from the center of the radial basis function. The polynomial term spanning the null space of the multi-order basis function is simply a constant, $P(\mathbf{x}) = p_0$. Figures 4(b) and (c) show plots of the above function for various values of δ and τ. Unlike the plot for $\phi(r) = |r|^3$, these plots show that the value of the basis function quickly falls toward zero as the distance from its center increases.

A. Comparison with Gaussian, Thin-Plate Radial Basis Functions, and Spatial Smoothing

The multi-order basis function described by Equation 6 has several advantages over the thin-plate and Gaussian basis functions used by Blinn, Muraki, Yuille, and others [6, 51, 29]. The system matrix formed by the thin-plate basis function is dense, and non-zero values grow larger away from the diagonal. Computation time increases significantly as more constraints are specified. In contrast, the system matrix formed by the multi-order basis function is diagonally dominant and is especially amenable to the biconjugate gradient method of solving linear equations. Even though the matrix formed by the multi-order basis is dense, non-zero values diminish away from the diagonal. Timing results show that the unknown weights of Equation 2 were solved in 1.5 minutes using the multi-order basis function with $\delta = 10$ and $\tau = 0.01$, while the system matrix generated for the same set of 3264 constraints using the thin-plate basis function required 7.9 minutes to solve on an SGI Origin with 195 MHz MIPS R10000 processor. Figure 4(d) is a comparison of running times versus number of constraints for the thin-plate and multi-order basis functions. The increase in running time as the number of constraints increase is fairly linear for the multi-order basis function as opposed to the thin-plate basis. The system matrix formed using Gaussian basis functions (with $\sigma = 0.01$) is sparse, requiring only 2.6 minutes to solve. The system matrix is solved even more quickly with smaller values of σ, but at the cost of worse reconstructions.

In terms of reconstruction quality, the multi-order basis function is able to reconstruct more locally detailed models while still retaining global smoothness. Both the thin-plate and the Gaussian basis functions result in models with overshooting surfaces. The Gaussian basis actually forms holes embedded in the surface. The thin-plate basis creates poorer reconstructions than the multi-order basis because the thin-plate basis forces the surface to be too smooth, resulting in blobby models. The Gaussian basis is an infinite mixture of Tikhonov stabilizers, also resulting in surfaces that are too smooth. Figure 3 is a comparison of reconstructions of the toy dinosaur using the thin-plate (c), the Gaussian (d), and the multi-order (e) basis functions. Note that the round protrusion beneath the arm is the wind-up key for the toy and that the bumps on the back are the scales and spines of the actual toy dinosaur (see Figure 9 for two of the original images).

Another difference between reconstruction using the multi-order and the thin-plate basis is in use of non-zero interior and exterior constraints. Reconstruction using the thin-plate basis is much more dependent on the dense placement of exterior constraints to prevent the surface from overshooting into regions where the model should not exist and on the placement of interior constraints to define the orientation of the surface. In [48], Turk and O'Brien pair each surface constraint with a *normal constraint* that is interior to the surface and has a function value of 1.0. The multi-order basis does not overshoot as much as the thin-plate basis. Hence, a sparse, uniform spread of exterior constraints are enough to orient the implicit surface. We have found in practice, that approximately one exterior constraint for every ten surface constraints is sufficient and that interior constraints are unnecessary. More details are provided in Section VI-A on how exterior constraints are obtained.

The real voxel coloring data sets we use, described in Section VII, are embedded in a global grid structure. In such cases, it is possible to spatially smooth the data in 3D and obtain a smooth reconstruction through iso-surface extraction. Note that this is not true in the general case where the input data set may be unstructured. As it turns out, the multi-order prior we use can give reconstructions that are very similar to spatial smoothing when δ and τ are appropriately set to be smooth. Figure 5 compares the reconstruction of the toy dinosaur using spatial smoothing and using the multi-order basis. The similarity of these reconstructions show that the multi-order basis is indeed closely related to spatial smoothing. As noted in [43], spatial smoothing tends to shrink features (such as the paws of the dinosaur), while volumentric regularization does not. An added advantage of using energy-minimizing basis functions is that it can create smooth reconstructions of unstructured and non-uniform data, to which spatially smoothing cannot easily be applied. Uniform spatial smoothing of unstructured data would require a resampling step to integrate all data points into a structured grid, as was done in [12]. In addition, the parameters, δ and τ, associated with the multi-order basis allows finer control over how much smoothing is applied. For example, in Figure 3(e), δ and τ were set to preserve the scales and spines on the back of the toy dinosaur which is lost by too much smoothing in Figure 5.

VI. CONSTRAINT SPECIFICATION

As described in Section IV-B, the implicit function we reconstruct evaluates to zero on the surface, positively inside the surface, and negatively outside. The data sets we use to perform the reconstruction is from passive range scanning. Such data sets are noisy, low in resolution, and more sparse than data sets from active range scanning. We describe the data sets in more detail in Section VII. In this
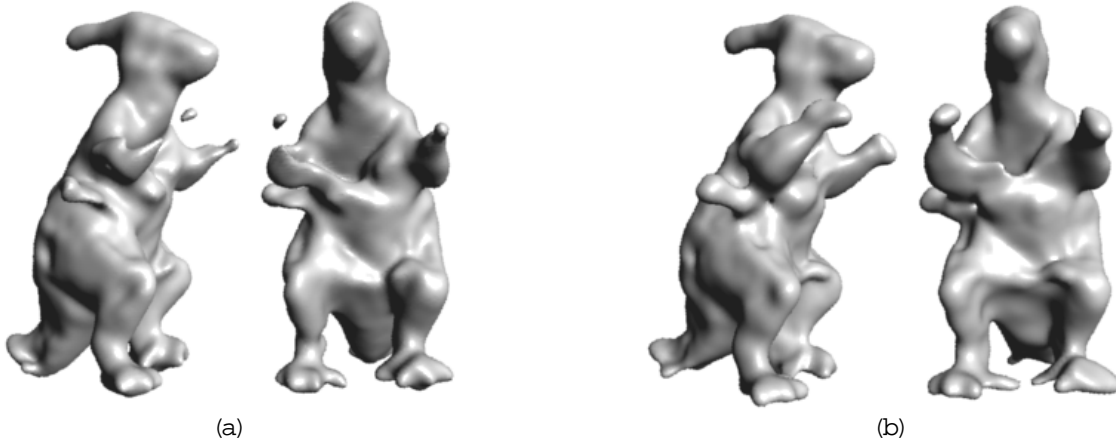
Fig. 5. (a) Iso-surface extraction of volumetric data after spatial smoothing using a Gaussian filter with a radius of four voxels. (b) Reconstruction using the multi-order basis function with 3000 surface constraints and $\delta = 10.0$ and $\tau = 0.025$.

section, we describe the method by which we obtain surface and exterior (negative) constraints used in the reconstruction. We also address the sampling required to guarantee that the topology of the object is correctly reconstructed and how this sampling density is mapped to the selected values for the parameters, δ and τ, controlling the amount of first and third order smoothness respectively.

A. Exterior Constraints

The computer vision community has developed many methods to acquire 3D positional information from photographic images taken by cameras. The goal of all these methods is to determine a collection of 3D points that lie on a given object's surface. When such a collection of points is acquired using cameras, the camera position and direction provide additional information that can be used for surface reconstruction. If a surface point is seen from a particular camera, there are no other surfaces between the camera and the point. We call the region between the camera and the surface *free space*. Other approaches to surface reconstruction make use of this information as well [12]. We can use this a priori knowledge about the object surface locations and the free space to define constraints that lie on or outside of the object, as seen in Figure 6.

Recall that the exterior constraints are those locations where we want our implicit function to be negative, and the surface constraints are where the implicit function should evaluate to zero. In practice, we place exterior constraints at the same distance away from the surface constraints towards the camera viewpoints and assign them a function value of -1.0. As mentioned in Section IV-B, exterior constraints do not represent actual data, but rather, are hints to the surface orientation. Hence, a sparse sampling of exterior constraints is sufficient to properly orient the surface, and a large value of λ, such as 2.0, indicates that the negative data point should be highly approximated. We have found that one exterior constraint per ten surface constraints works well in practice. An additional sparse set (about 16 points) of exterior constraints on a bounding
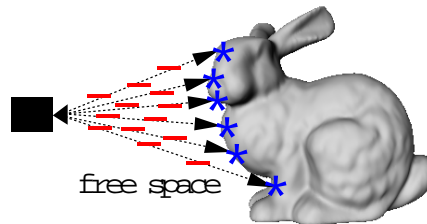

Fig. 6. Free space is carved out by rays projecting from the camera to the object surface. Surface (*) and exterior (-) constraints are defined by the free space.

sphere around the object helps to constrain the surface, and alone, is often sufficient to define the surface orientation. Next, we discuss how we subsample both the exterior and surface constraints.

B. Subsampling Surface Constraints

Because our method of reconstruction requires the solution of a linear system, it is computationally limited in the number of constraints that can be used to construct the surface. Examples shown in this paper have used around 3000 surface points, sampled from a set of around 20,000 surface points. Using the entire data set would not only be intractable, but would also result in an implicit function that is equal in size to the original discrete data set. In this case, the representation would no longer be compact.

The sampling density of a reduced data set must be such that the features in the data are well sampled. Since this information is not known a priori, our approach is to uniformly sample the data and then map this sampling density to appropriate δ and τ parameters. Surface points from the full data set are randomly selected. Each time a sample is selected, the neighboring samples within a small radius are eliminated from possible selection in the next round. The elimination process prevents clusters of closely placed constraint points, and resembles a 3D version of Poisson disc sampling. We have applied this method to uniformly
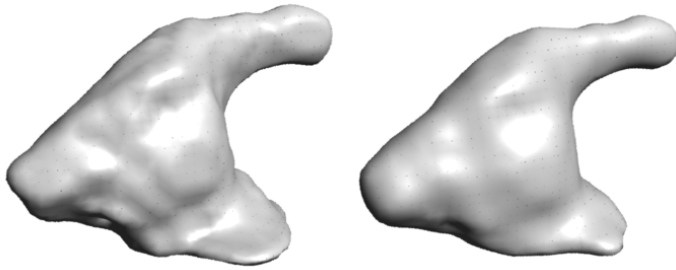
Fig. 7. Reconstructions of the head of the toy dinosaur. Left: reconstruction using full data set (3173 surface points). Right: reconstruction using a subset of available data (477 surface points).

subsample the voxel coloring data and exterior constraints previously described.

Experiments show that the reduced data set is sufficient to capture the details present in the noisy data. Figure 7 is a comparison between reconstructions from the entire data set and from a sampled subset. The full data set consists of 3173 surface points, while the reduced set consists of 477 points. The total distance, or error, between the original 3173 surface points and the surface reconstructed from the full data set was 0.008, while the total error between the 3173 surface points and the surface reconstructed from the reduced set was 0.009. The model itself was constrained to be within a $2 \times 2 \times 2$ box.

Adaptively increasing the sampling in highly detailed regions is not appropriate in many vision-based data sets. Detailed regions are often synonymous with areas of high curvature and small area. In a vision-based system, these small areas map to few pixels in the acquired images, resulting in low confidence for such regions. Increasing the sampling density in these small, detailed regions would taint the reduced data set with many low confidence points.

It is possible to partition the data set, construct a separate implicit surface for each partition, and then combine the surfaces. However, the resulting representation would not be compact. We opted not to take this approach since the difference in the fitness errors between the full and the reduced data sets was minimal. Yngve and Turk [50] and Carr et al. [9] have also shown that it is unnecessary to have a basis function for each surface point. Their approach was to iteratively add basis functions until the fitness error was sufficiently low. We avoid an iterative solution by uniformly sampling the data set. One drawback of the uniform sampling approach is that noise at the scale of features cannot be removed. Some examples of this effect are shown in the toy dinosaur's chest area.

C. Mapping Surface Sampling Density to δ and τ Values

Recall from Section V that δ controls the amount of first order smoothness, while τ controls the amount of third order smoothness. The values of δ and τ that correspond to the best reconstruction of a surface is dependent on the sampling density of the surface and the desired smoothing. In our work, we maintain consistent average sampling density across all models by constraining the size of the model

and by using nearly the same number of surface constraints to cover the data set. We scale all the models to lie within a $2 \times 2 \times 2$ box. By applying this normalization, the feature size, average sampling density, and choice of δ and τ are consistent across all models. This normalization is appropriate because all our input data sets have approximately the same resolution. One measure of this normalization is the average minimal distance between sample points. We compute this distance by averaging the distances between each sample point and its closest neighboring sample point. We show later in Section VII where we discuss the data sets in more detail that this average minimal distance is similar across all data sets after normalization and sampling.

We chose appropriate values for δ and τ by comparing models that have been reconstructed at various values of δ and τ. We have two methods of validation and comparison between the reconstructed models. These methods are a measure of fitness error and a measure of average curvature. We define fitness error to be the aggregate distance between the original data points and the reconstructed surface. To measure the average curvature of a surface, we first extract a polygonal model from the implicit function. We measure curvature at each vertex of the polygonal model using an approximation that was developed for the smoothing operator in [13]. The average curvature is obtained by dividing the aggregate curvature by the number of vertices in the polygonal model. High curvature is associated with sharp features in the surface, while low curvature is associated with overshoots and blobby surfaces.

We applied the measures of fitness and curvature to the toy dinosaur data set to guide selection of appropriate values for λ, δ, and τ. For details on the selection of these values, see our technical report [14]. We have found in practice that values of λ between 0.001 to 0.003, δ between 5.0 to 40.0 and τ between 0.005 to 0.025 can be used to produce locally detailed, yet globally smooth, reconstructions with minimal error on a variety of data sets.

D. Handling Outliers

Outliers are handled by a preprocessing step that finds the largest connected component in the data set. For the voxel coloring data set, we traverse the volume of surface points and group together voxels that are within the 26-neighborhood of each other. The single, largest connected component is kept, and all other surface points are eliminated. If n components exist (where $n > 1$), then we can sort the components in the data set according to their size, and keep only the first n largest components.

E. Topology Adaptation

One of the main advantages of the variational implicit surface technique is its ability to reconstruct models of arbitrary topology without explicit knowledge of the topology of the model beforehand. The resulting topology is, however, dependent on the data samples used to reconstruct the model. It is necessary to sufficiently specify surface and exterior constraints to define the topology. For example, if a torus is to be reconstructed, then at least one

exterior constraint is needed near the torus hole to force the existence of the hole in the middle of the torus. As long as the surface and exterior space are uniformly sampled, the topology is correctly reconstructed. However, since we are using data from vision-based methods, view occlusion or a lack of reference views may prevent correct sampling of the space. For example, if no views of the torus showing the hole in the center are available, then the hole may not be correctly reconstructed. We argue, that in such a case, the topology of the reconstructed model is consistent with the ambiguity of the topology in the data set.

VII. Results

We now show that volumetric regularization generates globally smooth, yet detailed, surfaces and discuss the addition of color to the models. We reconstructed surfaces using the multi-order basis on two types of data − synthetic range data and real voxel coloring data. Our method of reconstruction can generate smooth surfaces from data sets that are globally unstructured and noisy. Although neither types of data sets we use have both these features, each is an example of one feature. The synthetic range data is not embedded in a global grid, while the voxel coloring data is quite noisy in comparison to active range scanning data.

A. Synthetic Range Data

We use a modified ray-tracer [24] to generate synthetic range images as one test of our approach. We used the Stanford Bunny as our test model, and created three synthetic range images from positions separated by 120 degrees on a circle surrounding the model. For each range image, surface constraints are created by uniformly downsampling the range image to reduce the size of the data set. For every ten surface constraints, one exterior negative constraint is created within the free space described in Section VI. Additional exterior constraints are defined on a sphere surrounding the bounding box of the object at a distance farther away from the object. Figure 8(a) shows the original Stanford Bunny model consisting of 69,451 triangles, while (b), (c), and (d) show the implicit surface reconstructed from 2168 surface and 193 exterior constraints using the multi-order basis function. Figures 8(c) and (d) also show the distribution of the constraints overlayed on top of the reconstruction. The average minimum distance between surface samples used in the reconstruction is 0.051. Values of $\lambda = 0.001$, $\delta = 10$, and $\tau = 0.01$ were used to reconstruct the surface. The implicit surface is quite similar to the ground truth. Our method of reconstruction produces plausible surfaces even in locations where the data is sparse. The model is closed on the top and bottom of the Bunny even though few constraint points were placed there. The model is closed at these places due to the inherently manifold nature of implicit surfaces, and it is smooth at these locations by virtue of minimizing the cost functional.

B. Real Volume-Carved Data

Synthetic data does not have the noisy characteristic of real data. We now describe the real space carved data

that we use and how we define the surface and exterior constraints. We use three data sets of real objects obtained through voxel coloring [35, 11] − a toy dinosaur (from Steve Seitz [35]), a broccoli stalk, and a stack of toy tori (from Bruce Culbertson and Tom Malzbender [36] and referred to as the *towers* data set). Both data sets were obtained by taking about 20 images approximately on a circle around each object. Thin-shelled, voxelized surfaces were then constructed using the generalized voxel coloring algorithm [11]. The space is carved by splatting each visible voxel towards each calibrated camera and determining the consistency of the color across the images. If the variance in color intensity is below a specified threshold, the voxel is kept as part of the object surface. Otherwise, it is cast out and assigned a zero opacity value. The data consists of red, green, and blue channels. Non-empty voxels represent the presence of a surface, as deduced by the voxel coloring algorithm. Figure 10 shows the real voxel coloring data sets.

We apply the technique described in Section VI to obtain surface and exterior constraints for the voxel coloring data set. Non-empty voxels are surface locations. Exterior constraints are found by projecting each surface voxel in the volume to the image plane of each camera. If the ray from the surface voxel to a camera intersects other surface voxels, then the view of the voxel is blocked. Otherwise, the camera has an unobscured view, and an exterior constraint can be placed at a small distance away from the surface voxel along the ray towards the camera, as depicted in Figure 6. Note that for each surface voxel, an exterior constraint is created for each camera that has an unobscured view of the surface voxel. Again, only a subset of the surface and exterior constraints are selected by the Poisson disc sampling technique in Section VI-B. Once a specified number of constraints have been collected, they are given to the reconstruction algorithm. In this paper, we have used from 2000 to 3000 surface constraints. We have found that 100 to 300 exterior constraints suffice to define the orientation of the surface. Figure 10 shows examples of our reconstructions from space carved data. The average minimum distance between surface samples used in the reconstruction for the toy dinosaur, broccoli, and towers data sets are 0.035, 0.041, and 0.042, respectively. Note that the bumps on the back of the dinosaur are the scales and spines of the actual toy. The small protrusion near the base of the broccoli stalk is an actual leaf that has been accurately detected by the voxel coloring algorithm and has been correctly sampled and reconstructed by the method we describe in this paper. The running time for Marching Cubes [27] to extract an iso-surface is dependent on the desired resolution of the model and the number of terms (or constraints) in the implicit function. Surface extraction of the toy dinosaur at the resolution shown in Figure 10 took 14.5 minutes.

C. Model Coloring

In order to create a color version of the surface, we begin with a polygonal model that was obtained through iso-
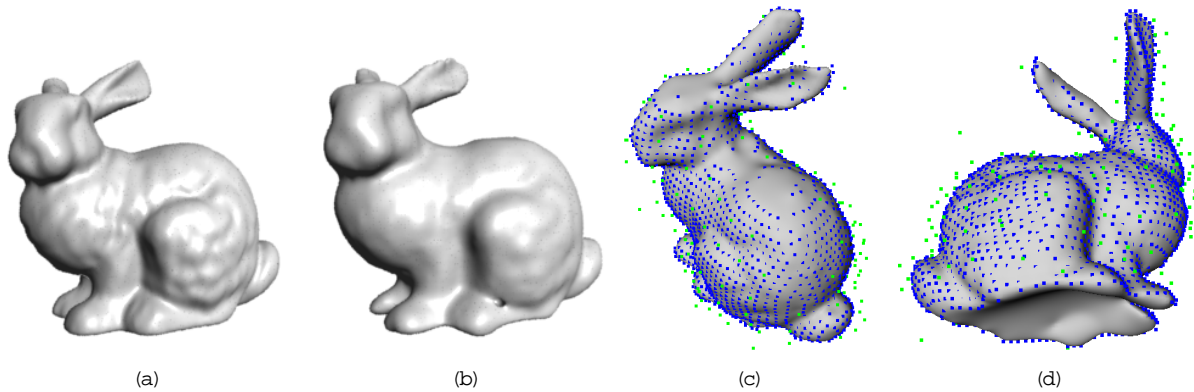
(a)   (b)   (c)   (d)

Fig. 8. Part (a) is the original Stanford Bunny consisting of 69,451 triangles. Parts (b), (c), and (d) show the reconstructed surface using the multi-order basis reconstruction method of this paper. Parts (c) and (d) show the surface constraints (blue squares) and the exterior constraints (green squares) used in the reconstruction overlayed on top of the reconstructed surface. Note that the reconstructed surface is closed on the top and bottom even though few constraints are present.

Fig. 9. Each pair of images is a comparison of the original input image used to generate the voxel coloring data set (left) and the reconstructed implicit model rendered from the same camera viewpoint (right). A novel viewpoint of the implicit model is shown in Figure 10

surface extraction using Marching Cubes [27]. We assign a color to each triangle of the polygonal model by reprojecting the triangles back to the original input images. Each triangle in the polygonal model is subdivided until its projected footprint in the images is subpixel in size, so that it can simply take on the color of the pixel to which it projects. In most cases, a triangle is visible in several of the original images. We combine the colors from the different images using a weighted average. The weight of each color contribution is calculated by taking the dot product between the triangle normal and the view direction of the camera that captured the particular image. Cameras with viewing directions that are nearly perpendicular to the triangle normal contribute less than those with viewing directions that are nearly parallel to the triangle normal. We use z-buffering to ensure that only cameras with an unobscured view of the triangle can contribute to the triangle color. Figure 10 shows the final models of the toy dinosaur, broccoli, and towers from novel viewpoints after color has been applied. Figure 9 is a comparison of two of the original input images of the toy dinosaur with rendered images

of the reconstructed implicit surface from the same camera viewpoints.

D. Limitations of Volumetric Regularization

Surface reconstruction using volumetric regularization does not generate surfaces with boundaries. Instead, our method closes over gaps in the data set to construct a manifold surface. Open surfaces can be generated by placing limits on the iso-surface extraction.

As noted in Section VI, the features and topology of the reconstructed model is dependent on the density of the input data set. Features that are not inherent in the data will not be reconstructed. Conversely, noise that is the size of features will become embedded in the reconstruction. This limitation is common to most methods of reconstruction and smoothing.

Our method of reconstruction requires the solution of a matrix system. This requirement constrains the size of the data sets that we can reconstruct due to speed and memory limitations. Recently published work by Carr et al. [9] on reconstructing surfaces from dense, precise data

20120 voxels $\delta = 25, \tau = 0.01$

29882 voxels $\delta = 5, \tau = 0.025$
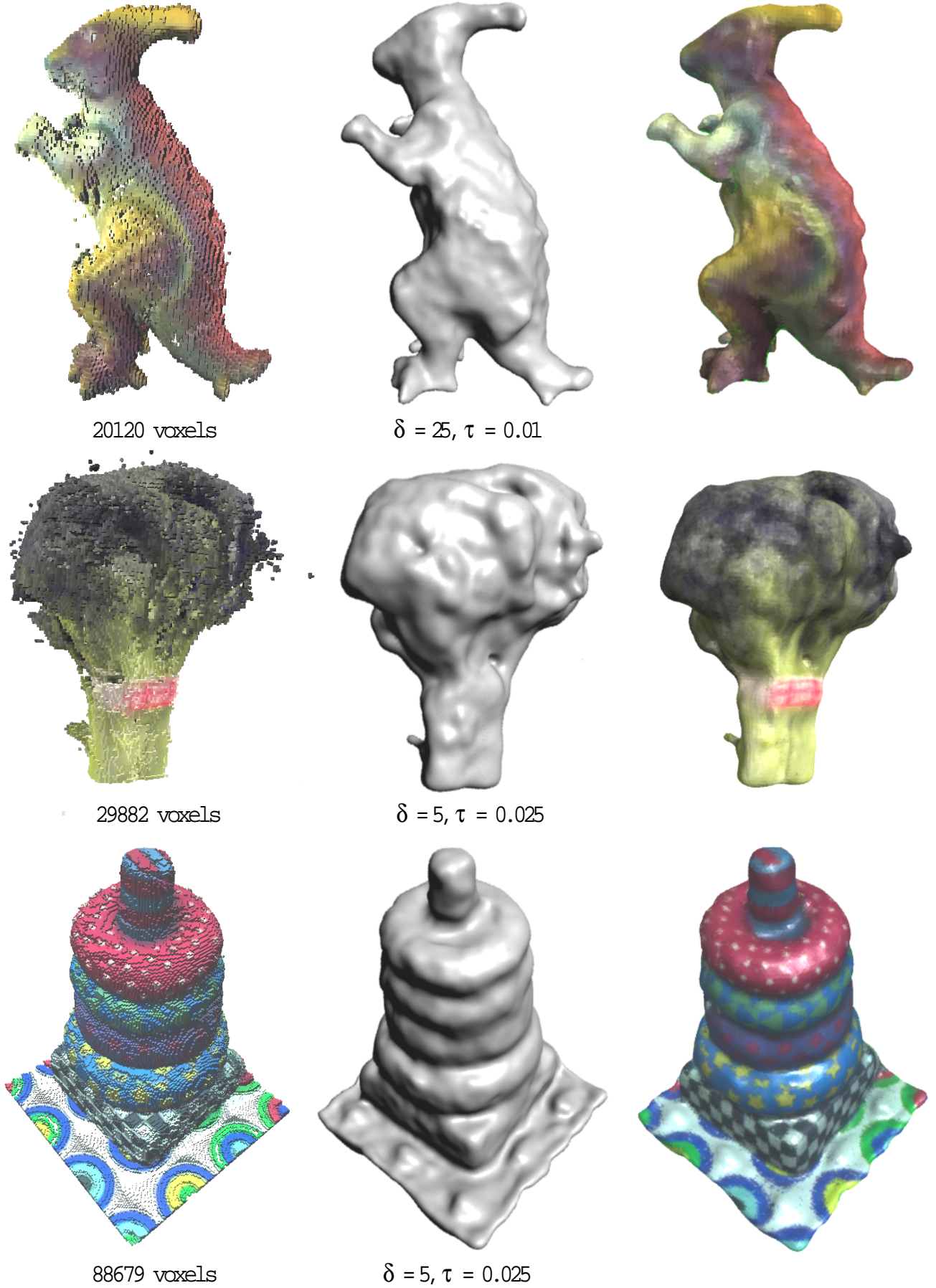
88679 voxels $\delta = 5, \tau = 0.025$

Fig. 10. From left to right: original voxel data sets from voxel coloring, our new implicit surface reconstructions using the multi-order radial basis function, and textured versions of our reconstructions. From top to bottom: toy dinosaur, broccoli, and towers data sets. 3000 surface constraints were used to construct the implicit surfaces.

sets using the thin-plate spline offers an efficient solution to the variational implicit method. We believe that their work using the Fast Multipole Method can also be applied here with the multi-order basis.

VIII. CONCLUSION AND FUTURE WORK

The reconstruction algorithm we have presented in this paper generates models that are smooth, seamless, and manifold. Our method is able to address challenges found in real data sets, including noise, non-uniformity, low resolution, and gaps in the data set. We have compared our technique to an exact interpolation algorithm (Crust), to thin-plate and Gaussian variational implicits, and to the original volumetric reconstruction using the toy dinosaur as a running example. Obvious advantages to the models generated by volumetric regularization are that there are no discretization artifacts as found in volumetric models, and the surface is not jagged as in the Crust reconstruction. Volumetric regularization can generate approximating, rather than interpolating, surfaces, and is most closely related to the thin-plate variational implicit surfaces. It compares favorably to the thin-plate variational implicit surfaces in computation time as well as in the surfaces that are generated. Using the multi-order radial basis function, volumetric regularization generates locally detailed, yet globally smooth surfaces that properly separate the features of the model.

We have adapted the variational implicit surfaces approach to real range data by developing methods to define surface and exterior constraints. Although surface points are directly supplied by the range data, we have introduced new methods for creating exterior constraints using information about the camera positions used in capturing the data. We have applied this technique to space carved volumetric data and synthetic range images.

We plan to look at several potential improvements to our approach, including use of confidence measurements and modifying the basis functions locally. For each 3D surface point obtained from the generalized voxel coloring algorithm, the regularization parameter, λ, can be assigned based on the variance of the colors to which the surface voxel projects in the input images. Another alternative is to assign different δ and τ values for the multi-order basis according to the curvature measure at constraint points. These future directions hold promise of further refining the sharp features of reconstructed surfaces of real world objects.

IX. ACKNOWLEDGEMENTS

REFERENCES

[1] Amenta, N., M. Bern, and M.Kamvysselis. "A New Voronoi-Based Surface Reconstruction Algorithm", *SIGGRAPH Proceedings*, Aug. 1998, pp.415–420.

[2] Bajaj, C. L., J. Chen, and G. Xu. "Free Form Surface Design with A-Patches", *Graphics Interface Proceedings*, 1994, pp.174–181.

[3] Bajaj, C. L., F. Bernardini, and G. Xu. "Automatic Reconstruction of Surfaces and Scalar Fields from 3D Scans", *SIGGRAPH Proceedings*, Aug. 1995, pp.109–118.

[4] Bernardini, F., J. Mittleman, H. Rushmeier, and C. Silva. "The Ball-Pivoting Algorithm for Surface Reconstruction", *IEEE Transactions on Visualization and Computer Graphics*, Oct.-Dec. 1999, 5(4), pp.349–359.

[5] Blane, M.M., Z. Lei, H. Civi, and D.B. Cooper. "The 3L Algorithm for Fitting Implicit Polynomial Curves and Surfaces to Data", *IEEE Transactions on Visualization and Computer Graphics*, Mar. 2000.

[6] Blinn, J. "A Generalization of Algebraic Surface Drawing", *ACM Transactions on Graphics*, July 1982, 1(3).

[7] Boult, T.E. and J.R.Kender. "Visual Surface Reconstruction Using Sparse Depth Data", *Computer Vision and Pattern Recognition Proceedings*, 1986, pp.68–76.

[8] Bruce, J.W. and P.J. Giblin. *Curves and Singularities, 2nd ed.*, Cambridge University Press, 1992, pp.59–98.

[9] Carr, J., R.K. Beatson, J.B. Cherrie, T.J. Mitchell, W.R. Fright, and B.C. McCallum. *SIGGRAPH Proceedings*, Aug. 2001, pp.67–76.

[10] Chen, F. and D. Suter. "Multiple Order Laplacian Spline - Including Splines with Tension", *MECSE 1996-5, Dept. of Electrical and Computer Systems Engineering Technical Report, Monash University*, July, 1996.

[11] Culbertson, W. B., T. Malzbender, and G. G. Slabaugh. "Generalized Voxel Coloring", *Vision Algorithms: Theory and Practice*. Part of Series: *Lecture Notes in Computer Science*, Vol. 1883 Sep. 1999, pp.67–74.

[12] Curless, B. and M. Levoy. "A Volumetric Method for Building Complex Models from Range Images", *SIGGRAPH Proceedings*, Aug. 1996, pp.303–312.

[13] Desbrun, M., M. Meyer, P. Schroder, and A.H. Barr. "Implicit Fairing of Irregular Meshes Using Diffusion and Curvature Flow", *SIGGRAPH Proceedings*, Aug. 1999, pp.317–324.

[14] Dinh, H. and G. Turk. "Reconstructing Surfaces by Volumetric Regularization", *GVU-00-26, College of Computing, Georgia Tech*, December 2000.

[15] Edelsbrunner H. and E.P. Mucke. "Three-Dimensional Alpha Shapes", *ACM Transactions on Graphics*, Jan. 1994, 13(1), pp. 43–72.

[16] Fang, L. and D. Gossard. "Multidimensional Curve Fitting to Unorganized Data Points by Nonlinear Minimization", *Computer-Aided Design*, Jan. 1995, 27(1), pp.48–58.

[17] Fomenko, A. and T. Kunii. *Topological Modeling for Visualization*, Springer, 1997.

[18] Frisken, S.F., R.N. Perry, A.P. Rockwood, and T.R. Jones. "Adaptively Sampled Distance Fields: A General Representation of Shape for Computer Graphids", *SIGGRAPH Proceedings*, Aug. 2000, pp.249–254.

[19] Girosi, F., M. Jones, and T. Poggio. "Priors, Stabilizers and Basis Functions: From Regularization to Radial, Tensor and Additive Splines", *A.I. Memo No. 1430, C.B.C.L. Paper No.75, Massachusetts Institute of Technology Artificial Intelligence Lab*, June, 1993.

[20] Hilton, A., A.J. Stoddart, J. Illingworth, and T. Windeatt. "Reliable Surface Reconstruction from Multiple Range Images", *Proceedings of Fourth European Conference on Computer Vision*, 1996.

[21] Hoppe, H., T. DeRose, and T. Duchamp. "Surface Reconstruction from Unorganized Points", *SIGGRAPH Proceedings*, July 1992, pp.71–78.

[22] Keren D. and C. Gotsman. Tight Fitting of Convex Polyhedral Shapes. *International Journal of Shape Modeling (special issue on Reverse Engineering Techniques)*, pp.111–126, 1998.

[23] Keren, D. and C. Gotsman. "Fitting Curve and Surfaces with

Constrained Implicit Polynomials", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(1), pp.21–31, 1999.

[24] Kolb, C., The Rayshade Homepage. *http://www-graphics.stanford.edu/ cek/rayshade/info.html*.

[25] Kutulakos, K.N. and S.M.Seitz. "A Theory of Shape by Space Carving", *International Journal of Computer Vision*, Aug. 1999.

[26] Lee, M. and G. Medioni. "Inferring Segmented Surface Description from Stereo Data", *Computer Vision and Pattern Recognition Proceedings*, 1998, pp.346–352.

[27] Lorensen, W.E. and H.E. Cline. "Marching Cubes", *SIGGRAPH Proceedings*, July 1987, pp.163–169.

[28] Morse, B.S., T.S. Yoo, P. Rheingans, D.T. Chen, and K.R. Subramanian. "Interpolating Implicit Surfaces from Scattered Data Using Compactly Supported Radial Basis Functions" *Proceedings of Shape Modeling International*, May, 2001, pp.89–98.

[29] Muraki S. "Volumetric Shape Description of Range Data using Blobby Model", *SIGGRAPH Proceedings*, July 1991, pp.227–235.

[30] Nielsen, M., L. Florack, and R. Deriche. "Regularization, Scale-Space, and Edge Detection Filters", *Journal on Mathematical Images and Vision*, 1997.

[31] Nielson, G.M. "Scattered Data Modeling", *Computer Graphics and Applications*, 1993, pp.60–70.

[32] Pentland, A. "Automatic Extraction of Deformable Part Models", *International Journal of Computer Vision*, 1990, Vol.4, pp.107–126.

[33] Savchenko, V.V., A.A. Pasko, O.G.Okunev, and T.L.Kunii. Function Representation of Solids Reconstructed from Scattered Surface Points and Contours. *Computer Graphics Forum*, Vol. 14, No. 4, 1995, pp.181– 188.

[34] Sclaroff, S. and A. Pentland. "Generalized Implicit Functions for Computer Graphics", *SIGGRAPH Proceedings*, July 1991, pp.247–250.

[35] Seitz, S.M. and C.R. Dyer. "Photorealistic Scene Reconstruction by Voxel Coloring", *International Journal on Computer Vision*, 1999, 35(2), pp.151–173.

[36] Slabaugh, G., B. Culbertson, T. Malzbender, and R. Schafer. "A Survey of Methods for Volumetric Scene Reconstruction from Photographs", *International Workshop on Volume Graphics*, June 2001, pp.51–62.

[37] Soucy, M. and D. Laurendeau "A General Surface Approach to the Integration of a Set of Range Views", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, April 1995, 17(4), pp.344–358.

[38] Suter, D. and F, Chen. "Left Ventricular Motion Reconstruction Based on Elastic Vector Splines", To appear in *IEEE Transactions on Medical Imaging*, 2000.

[39] Szeliski R. and S. Lavallee. "Matching 3D Anatomical Surfaces with Non-Rigid Deformations using Octree-Splines", *International Journal of Computer Vision*, 1996, 18(2), pp.171–186.

[40] Tang, C. and G. Medioni. "Inference of Integrated Surface, Curve, and Junction Descriptions From Sparse 3D Data", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Nov. 1998., 20(11).

[41] Taubin, G. "Estimation of Planar Curves, Surfaces, and Nonplanar Spaces Curves Defined by Implicit Equations with Applications to Edge and Range Image Segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Nov. 1991, 13(11).

[42] Taubin, G. "An Improved Algorithm for Algebraic Curve and Surface Fitting", *Proceedings Fourth International Conference on Computer Vision*, May 1993, pp.658–665.

[43] Taubin, G. "A Signal Processing Approach to Fair Surface Design", *SIGGRAPH Proceedings*, 1995, pp.351–358.

[44] Terzopoulos, D. "Regularization of Inverse Visual Problems Involving Discontinuities", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 1986, 8(4).

[45] Terzopoulos, D. "The Computation of Visible Surface Representations", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 1988, 10(4).

[46] Terzopoulos, D. and D. Metaxas. "Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July 1991, 13(7).

[47] Turk, G. and M. Levoy. "Zippered Polygon Meshes from Range Images", *SIGGRAPH Proceedings*, 1994., pp.311–318.

[48] Turk, G. and J.F. O'Brien. "Shape Transformation Using Variational Implicit Functions", *SIGGRAPH Proceedings*, Aug. 1999, pp.335–342.

[49] Wahba, G. "Spline Models for Observational Data", *CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM*, 1990. pp.11–14, pp.45–65.

[50] Yngve, G. and G. Turk. "Robust Creation of Implicit Surfaces from Polygonal Meshes", To appear in *ACM Transactions on Visualization and Computer Graphics*.

[51] Yuille, A.L. and N.M.Grzywacz. "The Motion Coherence Theory", *Proceedings of the International Conference on Computer Vision*, 1988, pp.344–353.