

Finite Element Techniques for Fitting a Deformable Model to 3D Data

Tim McInerney

Department of Computer Science
University of Toronto
Toronto, ON, Canada, M5S 1A4

Demetri Terzopoulos¹

Abstract

This paper presents a physics-based approach to surface reconstruction using an elastically deformable "sheet" model. The model is based on a thin-plate under tension spline which deforms to fit visual data according to internal forces stemming from the elastic properties of the surface and external forces which are produced from the data. We employ the finite element method to represent the model as a continuous surface. We implement two versions of the sheet using two different finite elements. The first is a triangular, quintic finite element whose nodal variables comprise the position of the surface plus its first and second partial derivatives. This element is "natural" in the sense that the nodal variables reflect each of the partial derivatives that occur in the spline's strain energy functional. The partial derivatives are useful in measuring the differential geometric properties of the fitted surface. The second element is a rectangular, bicubic finite element whose nodal variables also include some of the partial derivatives of the surface. We apply the sheet model to the reconstruction of various 3D data sets generated by several different sensing technologies related to CAGD and terrain mapping.

1 Introduction

General purpose shape reconstruction in low-level visual processing requires models with the ability to assume a wide range of shapes. The models must extract meaningful information from noisy sensor data while making the weakest possible assumptions about observed shapes. Deformable models can stretch, bend, and twist to assume complex freeform shapes; hence, they seem ideally suited to this task. These models integrate, regularize, and approximate the data to produce a globally consistent interpretation.

Surface reconstruction techniques based on splines have attracted significant interest for many years because of their versatility. However, the dynamic form of this type of surface reconstruction was introduced fairly recently by Terzopoulos *et al.* [13]. They proposed a dynamic deformable cylinder model constructed from generalized splines and developed force field techniques to fit the model to image data. The dynamic model fitting approach is being pursued by several researchers [6, 9, 2, 3, 14, 5, 15], as it is in this paper.

This paper presents a physics-based approach to surface reconstruction using an elastically deformable "sheet" model. The model is based on physically motivated multi-dimensional generalizations of classical

splines. The model is dynamic, and its deformation is governed by the laws of nonrigid motion. The formulation of the motion equations includes strain energies, simulated forces, and other physical quantities. The strain energy stems from a thin-plate under tension spline, and deformation results from the action of internal forces which describe continuity constraints and external forces which describe data compatibility constraints.

We employ the finite element method to discretize our deformable sheet into a set of connected element domains. The finite element method provides an analytic surface representation and it generally requires fewer discretization points than finite difference techniques for the same degree of accuracy. We implement two versions of the sheet model using two different finite elements. The first is a triangular, quintic finite element whose nodal variables reflect the derivative terms found in the thin-plate under tension energy expression. That is, the nodal degrees of freedom include not only the nodal positions, but all of the first and second order partial derivatives of the surface as well. These quantities are useful in the analysis of the fitted model surface. The second element is a rectangular, bicubic finite element whose nodal variables also include some of the partial derivatives of the surface.

Finite element representations for variational problems in vision were first explored in [11]. Our formulation applies the finite element method to the thin-plate under tension spline proposed in [12] in order to derive discrete nonrigid dynamics equations. Our work is related to that of [1] who also develop a deformable surface model which is based on the thin-plate under tension spline.

2 Dynamic Deformable Surface Model

The deformable sheet is a rectangular open surface represented as a single-valued scalar function $z(x, y)$ defined on a bounded domain in the (x, y) image plane, where z represents the lateral displacement of the sheet. We construct the sheet using a simulated thin-plate material under tension. The deformation energy of this material serves as a smoothness constraint which compels the sheet to vary smoothly almost everywhere. The data constraints apply forces which pull the sheet laterally from its nominally planar state to the data points.

The deformation energy is given by the functional

¹Fellow, Canadian Institute for Advanced Research

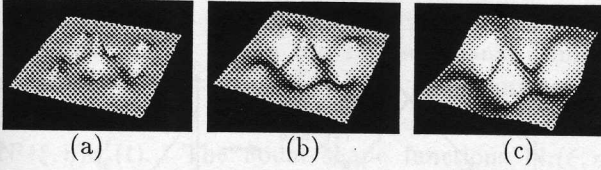


Figure 1: Sheet fitted to nine data points. (a) Sheet with $\alpha_{ij} = 0.8$, $\beta_{ij} = 0.0$. (b) $\alpha_{ij} = \beta_{ij} = 0.4$. (c) $\alpha_{ij} = 0.0$, $\beta_{ij} = 0.8$.

$$\mathcal{E}_p(z) = \iint \alpha_{10} |z_x|^2 + \alpha_{01} |z_y|^2 + \beta_{20} |z_{xx}|^2 + \beta_{11} |z_{xy}|^2 + \beta_{02} |z_{yy}|^2 dx dy, \quad (1)$$

where the x and y subscripts denote derivatives in the x and y directions respectively. \mathcal{E}_p is a controlled-continuity spline defined in [12]. The nonnegative weighting functions $\alpha_{ij}(x, y)$ and $\beta_{ij}(x, y)$ control the elasticity of the material. The α_{10} and α_{01} functions control the tensions in the x and y directions, respectively. The β_{02} and β_{20} functions control the bending rigidities in the x and y directions, respectively. The β_{11} function controls the twisting rigidity. Increasing the α_{ij} has a tendency to decrease the surface area of the material, while increasing the β_{ij} tends to make it more rigid. In general, the weighting function may be used to introduce depth and orientation discontinuities in the material. In this paper, however, we do not pursue this topic and we set the functions to constant values $\alpha_{ij}(x, y) = \alpha_{ij}$ and $\beta_{ij}(x, y) = \beta_{ij}$.

The expression for the total energy of the deformable sheet is

$$\mathcal{E}_{sheet}(z) = \mathcal{E}_p(z) + \mathcal{E}_{ext}(z), \quad (2)$$

where

$$\mathcal{E}_{ext} = - \iint z f(x, y) dx dy \quad (3)$$

is a data energy derived from the data forces $f(x, y)$. We fit the model to the data by minimizing this energy.

Figure 1 illustrates the surface reconstruction problem using the sheet model. The figure shows the sheet fitted to several data points using different values for $\alpha_{ij}(x, y)$ and $\beta_{ij}(x, y)$.

In keeping with the physical nature of our reconstruction technique, we fit the deformable sheet model to data by making it dynamic. A dynamic formulation naturally supports model visualization during the data fitting process, and, if needed, allows a user to introduce constraint forces to pull the model out of local minima towards a good solution.

In a dynamic formulation, the positions of material points becomes a time-dependent function $z(x, y, t)$ and the simulated material is imbued with mass and damping densities. Using Lagrangian dynamics the energies yield forces and $\mathcal{E}_{sheet}(z)$ is minimized when the forces equilibrate and the model becomes stationary $\partial z / \partial t = \partial^2 z / \partial t^2 = 0$.

The dynamic behavior of the model during the fitting process is governed by

$$\mu \frac{\partial^2 z}{\partial t^2} + \gamma \frac{\partial z}{\partial t} + \delta_z \mathcal{E}_p = f(x, y, t). \quad (4)$$

where the first term represents the inertial forces due to the mass density $\mu(x, y)$, the second term represents the damping forces due to the damping density $\gamma(x, y)$, the third term represents the elastic force which resist deformation, and finally $f(x, y, t)$ represents the data forces.

3 Finite Element Representation

To apply the finite element method, we tessellate the continuous image domain (x, y) into a mesh of element subdomains E_j . We approximate the continuous z as a weighted sum of piecewise polynomial basis functions \mathbf{N}_i :

$$\hat{z}(x, y, t) = \sum_{i=1}^N \mathbf{N}_i(x, y) \mathbf{q}_i(t) \approx z(x, y, t), \quad (5)$$

where \mathbf{q}_i is a vector of nodal variables associated with mesh node i . We define $\mathbf{q} = [\mathbf{q}_1^T, \dots, \mathbf{q}_i^T, \dots, \mathbf{q}_N^T]^T$.

Substituting (5) into (4) yields the discrete equations of motion

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{f}_q, \quad (6)$$

where the mass matrix \mathbf{M} , damping matrix \mathbf{C} , and stiffness matrix \mathbf{K} are sparse, symmetric matrices and \mathbf{f}_q are nodal data forces. These *global* matrices and nodal data forces may be assembled from their associated *local* element matrices, \mathbf{M}^j , \mathbf{C}^j , \mathbf{K}^j , and nodal forces \mathbf{f}_q^j , by expanding each element matrix appropriately into a $q \times q$ matrix and then summing. For example:

$$\mathbf{K} = \sum_{j=1}^M \mathbf{K}_{q \times q}^j, \quad (7)$$

where \mathbf{K}^j is the element stiffness matrix associated with element E_j , $j = 1, \dots, M$.

We now derive expressions for \mathbf{M}^j , \mathbf{C}^j , \mathbf{K}^j , and \mathbf{f}_q^j from element kinetic and potential energy functionals. Let $z^j(x, y, t)$ be the position of material point x, y within E_j , and let \mathbf{q}^j denote the concatenation of nodal variables for all the nodes of E_j . Following equation (5), we write the element trial function

$$\hat{z}^j(x, y, t) = \mathbf{N}^j(x, y) \mathbf{q}^j(t) \approx z^j(x, y, t), \quad (8)$$

where \mathbf{N}^j are known as element shape functions. Note that the basis functions \mathbf{N}_i are obtained by superposing the shape functions associated with node i . The element velocity is $\partial \hat{z}^j / \partial t = \mathbf{N}^j \dot{\mathbf{q}}^j$, where $\dot{\mathbf{q}}^j(t)$ is the rate of change of the nodal variables.

The kinetic energy associated with element E_j be expressed as

$$\frac{1}{2} \iint_{E_j} \mu \frac{\partial \dot{z}^j}{\partial t} \frac{\partial \dot{z}^j}{\partial t} dx dy = \frac{1}{2} \dot{\mathbf{q}}^j \mathbf{M}^j \dot{\mathbf{q}}^j, \quad (9)$$

where the element mass matrix is given by

$$\mathbf{M}^j = \iint_{E_j} \mu \mathbf{N}^j \mathbf{N}^j \mathbf{T} dx dy. \quad (10)$$

We introduce simple velocity-proportional kinetic energy dissipation according to the dissipation functional

$$\frac{1}{2} \iint_{E_j} \gamma \frac{\partial \dot{z}^j}{\partial t} \frac{\partial \dot{z}^j}{\partial t} dx dy = \frac{1}{2} \dot{\mathbf{q}}^j \mathbf{C}^j \dot{\mathbf{q}}^j, \quad (11)$$

where the element damping matrix is given by

$$\mathbf{C}^j = \iint_{E_j} \gamma \mathbf{N}^j \mathbf{N}^j \mathbf{T} dx dy. \quad (12)$$

According to (1) the element deformation matrix may be expressed as

$$\mathcal{E}_p^j(z) = \iint_{E_j} \boldsymbol{\sigma}^j \mathbf{T} \boldsymbol{\epsilon}^j dx dy \quad (13)$$

where the strain vector is

$$\boldsymbol{\epsilon}^j = \left[\frac{\partial z^j}{\partial x}, \frac{\partial z^j}{\partial y}, \frac{\partial^2 z^j}{\partial x^2}, \frac{\partial^2 z^j}{\partial x \partial y}, \frac{\partial^2 z^j}{\partial y^2} \right]^T \quad (14)$$

and the stress vector is

$$\boldsymbol{\sigma}^j = \begin{bmatrix} \alpha_{10}^j & 0 & 0 & 0 & 0 \\ 0 & \alpha_{01}^j & 0 & 0 & 0 \\ 0 & 0 & \beta_{20}^j & 0 & 0 \\ 0 & 0 & 0 & \beta_{11}^j & 0 \\ 0 & 0 & 0 & 0 & \beta_{02}^j \end{bmatrix} \boldsymbol{\epsilon}^j = \mathbf{D}^j \boldsymbol{\epsilon}^j, \quad (15)$$

Using (8), we can write

$$\boldsymbol{\epsilon}^j = \mathbf{B}^j \mathbf{q}^j, \quad (16)$$

where \mathbf{B}^j is the element strain matrix. Inserting the expressions for $\boldsymbol{\epsilon}^j$ and $\boldsymbol{\sigma}^j$ into (13) yields

$$\mathcal{E}_p^j(z) = \mathbf{q}^j \mathbf{T} \mathbf{K}^j \mathbf{q}^j \quad (17)$$

where the element stiffness matrix is given by

$$\mathbf{K}^j = \iint_{E_j} \mathbf{B}^j \mathbf{T} \mathbf{D}^j \mathbf{B}^j dx dy. \quad (18)$$

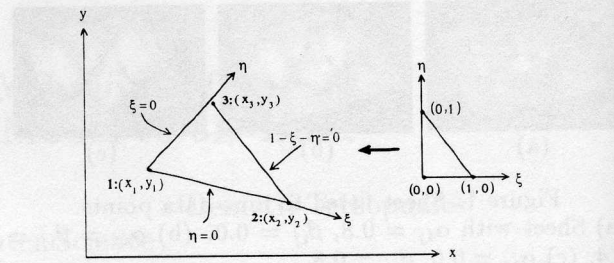


Figure 2: C^1 continuous triangular element. The three nodes are numbered.

Finally, according to (3), the potential energy in element E_j due to data forces $f^j(x, y, t)$ is

$$- \iint_{E_j} z^j f^j(x, y) dx dy = -\mathbf{q}^j \mathbf{T} \mathbf{f}_q^j, \quad (19)$$

where the nodal data forces are given by

$$\mathbf{f}_q^j = \iint_{E_j} \mathbf{N}^j f^j dx dy. \quad (20)$$

3.1 Triangular C^1 Finite Element

The first version of the sheet model uses a fifth-order triangular finite element [4]. The nodal variables of this element are z , along with its first and second partial derivatives evaluated at each node i . The nodal variable vector for the sheet is therefore

$$\mathbf{q}_i(t) = [z_i, (z_x)_i, (z_y)_i, (z_{xx})_i, (z_{xy})_i, (z_{yy})_i]^T. \quad (21)$$

These nodal variables reflect each of the partial derivatives that occur in the thin-plate under tension energy (1) functional.

Figure 2 shows the C^1 continuous element defined locally in the dimensionless oblique coordinates (ξ, η) . In this local coordinate system, the global coordinates (x, y) can be expressed as

$$x = (1 - \xi - \eta)x_3 + \xi x_1 + \eta x_2 \quad (22)$$

$$y = (1 - \xi - \eta)y_3 + \xi y_1 + \eta y_2$$

where (x_i, y_i) are the global coordinates at the nodes of the triangular element (as numbered in the figure), and the local nodal variable vector becomes

$$\mathbf{q}_{i\xi}(t) = [z_i, (z_\xi)_i, (z_\eta)_i, (z_{\xi\xi})_i, (z_{\xi\eta})_i, (z_{\eta\eta})_i]^T. \quad (23)$$

The transformation from global to local coordinates is

$$\mathbf{q}_i = \mathbf{T}_i \mathbf{q}_{i\xi} \quad (24)$$

where the transformation matrix \mathbf{T}_i is specified in [4] (pp. 100-101).

Concatenating the $\mathbf{q}_{i\xi}$ at each of the three nodes of element j , we obtain the 18-dimensional element nodal vector $\mathbf{q}_{\xi}^j = [\mathbf{q}_{1\xi}^T, \mathbf{q}_{2\xi}^T, \mathbf{q}_{3\xi}^T]^T$. According to (8), we can write the local trial function as $\hat{z}^j(\xi, \eta, t) = \mathbf{N}^j(\xi, \eta)\mathbf{q}_{\xi}^j(t)$. The nodal shape functions $N_i(\xi, \eta)$ which are contained in the 18×18 matrix \mathbf{N}^j are specified in [4] (pp. 100–101). Note that the polynomial basis of the element is complete up to fourth-order terms and contains three fifth-order terms. The trial functions are C^∞ within elements and they ensure C^1 continuity between elements. Since (1) contains up to second order derivatives, the element is conforming.

The shape functions are expressed in terms of the local coordinates (ξ, η) and it is convenient to work with these coordinates. Thus, the required derivatives of the shape functions in the strain matrix \mathbf{B} are computed using repeated applications of the chain rule and equation (23). Also, a function $f(x, y)$ may be integrated over E_j by transforming to the local coordinate system:

$$\iint_{E_j} f(x, y) dx dy = \iint_{E_j} f(x(\xi, \eta), y(\xi, \eta)) \det \mathbf{J} d\xi d\eta. \quad (25)$$

where

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \quad (26)$$

is the Jacobian matrix. These integrals are approximated using Gauss-Legendre quadrature rules.

3.2 Rectangular C^1 Finite Element

Since the image domain is rectangular, it is natural to use rectangular finite elements. Consequently, we also implemented the sheet model using a four-noded rectangular bicubic Hermitian element with nodal variables z_i , along with its first derivatives and second cross-derivative evaluated at each node i . The nodal variable vector for this version of the sheet is

$$\mathbf{q}_i(t) = \left[z_i, \left(\frac{\partial z}{\partial x} \right)_i, \left(\frac{\partial z}{\partial y} \right)_i, \left(\frac{\partial^2 z}{\partial x \partial y} \right)_i \right]^T. \quad (27)$$

Thus there are sixteen degrees of freedom per element. Although the nodal variables of this element do not contain the *full* set of nodal derivatives, the uniform tessellation of the image domain along with the element nodal first derivatives can be used to compute the remaining second derivatives.

For the rectangular Hermitian bicubic element, the local coordinates

$$\xi = \frac{2}{a}(x - x_c), \quad \eta = \frac{2}{b}(y - y_c), \quad (28)$$

are used ((x_c, y_c) are the coordinates of the element center). Using this element, the trial function \hat{z} can be written as

$$\hat{z}(x, y) = \sum_{i=1}^4 \mathcal{N}_{1i} z_i + \mathcal{N}_{2i} \frac{\partial z_i}{\partial x} + \mathcal{N}_{3i} \frac{\partial z_i}{\partial y} + \mathcal{N}_{4i} \frac{\partial^2 z_i}{\partial x \partial y}, \quad (29)$$

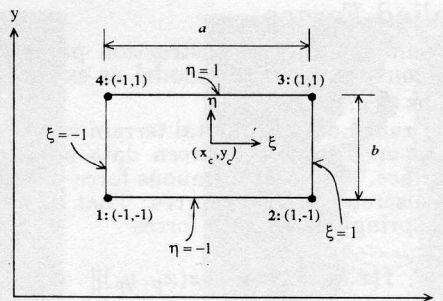


Figure 3: Hermitian rectangular element. The four nodes are numbered.

where

$$\begin{aligned} \mathcal{N}_{1i} &= N_{0i}(\xi)N_{0i}(\eta), & \mathcal{N}_{2i} &= N_{0i}(\xi)N_{1i}(\eta) \\ \mathcal{N}_{3i} &= N_{1i}(\xi)N_{0i}(\eta), & \mathcal{N}_{4i} &= N_{1i}(\xi)N_{1i}(\eta) \end{aligned} \quad (30)$$

The nodal shape functions N_{ij} are specified in [8] (pp. 206). Note that the trial function (29) is an incomplete sixth-order polynomial in terms of ξ and η and possesses continuity in both the function and its first derivatives. As the energy functional \mathcal{E}_p contains 2nd order partial derivatives, the Hermitian bicubic element ensures continuity of the trial function and its first derivatives at element interfaces, hence this element is also conforming.

3.3 Numerical Integration

In computer vision and geometric design applications involving the fitting of models to data, we can simplify the equations of motion (6) while preserving useful dynamics by setting the mass density $\mu(x, y)$ to zero to obtain $\mathbf{C}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{f}_q$. This first-order dynamic system governs a model which has no inertia and comes to rest as soon as all the applied forces vanish or equilibrate.

We integrate equation (3.3) forward through time using an explicit first-order Euler method. This method approximates the temporal derivatives with forward finite differences. It updates the degrees of freedom \mathbf{q} of the model from time t to time $t + \Delta t$ according to the formula

$$\mathbf{q}^{(t+\Delta t)} = \mathbf{q}^{(t)} + \Delta t(\mathbf{C}^{(t)})^{-1} (\mathbf{f}_q^{(t)} - \mathbf{K}\mathbf{q}^{(t)}). \quad (31)$$

The time step Δt must be selected carefully, since small time steps slow down the computation while excessively large time steps result in numerical instability.

In our implementation, we do not explicitly assemble and factorize a global stiffness matrix \mathbf{K} as is common practice in applied finite element analysis. Instead, we update the nodal vectors $\mathbf{q}_i^{(t+\Delta t)}$ iteratively by computing the product $\mathbf{K}^j \mathbf{q}^j$ on an element-by-element basis using the element stiffness matrices \mathbf{K}^j . This approach makes the model fitting process easily parallelizable.

4 Applied Forces

Our dynamic surface reconstruction paradigm applies data constraints to the model as external force distributions $f(x, y, t)$.

For laser range data or digital terrain maps, we use forces based on distances between data points and a model's surface, and the continuous force distribution becomes a discrete, vector of forces. That is, we define long-range spring-like "point" forces

$$\mathbf{f}(x, y) = \kappa \|\mathbf{p} - z(x_p, y_p)\| \quad (32)$$

proportional to the separation between a data point \mathbf{p} in space and the point of influence $z(x_p, y_p)$ of the force on the model's surface.

For the sheet model, the data point $p(x, y)$ is projected onto the surface parallel to the z axis. Equation (20) can then be applied to transform this force into an equivalent set of nodal forces \mathbf{f}_q^j .

5 3D Generalization

The deformable sheet model is represented as a *single-valued* scalar function $z(x, y)$ defined on a bounded domain in the (x, y) plane as described above. We have also implemented a deformable "balloon" model, using a more general, *multi-valued* parametric representation

$$\mathbf{x}(u, v) = [x(u, v), y(u, v), z(u, v)]^T \quad (33)$$

where vector \mathbf{x} represents the positions of material points (u, v) relative to a reference frame Φ in Euclidean 3-space. Note that the single-valued sheet model representation can be considered as a special case of the multi-valued representation: $\mathbf{x}(u, v) = [u, v, z(u, v)]^T$.

The deformation energy of the thin plate material under tension for the multi-valued case is an extension of (1)

$$\mathcal{E}_p(\mathbf{x}) = \iint \alpha_{10} |\mathbf{x}_u|^2 + \alpha_{01} |\mathbf{x}_v|^2 + \beta_{20} |\mathbf{x}_{uu}|^2 + \beta_{11} |\mathbf{x}_{uv}|^2 + \beta_{02} |\mathbf{x}_{vv}|^2 du dv, \quad (34)$$

where the u and v subscripts denote parametric derivatives.

6 Surface Analysis

Once a surface has been extracted and reconstructed from the data, a logical next step is to characterize the surface structure for use in analysis, visualization, and recognition tasks. For example, surface curvature extrema often correspond to significant intrinsic features of an object [7]. Consequently, it is desirable for a surface model to not only extract accurate and meaningful information from the data, but to provide the extracted information in a convenient and useful form.

From differential geometry, 3D smooth surfaces are uniquely characterized by their first and second fundamental forms. The parametric surface form of the sheet model (i.e. $\mathbf{x}(u, v) = [u, v, z(u, v)]^T$), along with the finite element approximation, provides us

with an analytical description of the reconstructed surface that is directly amenable to a differential analysis. Furthermore, since the nodal variables of our finite elements contain not only the nodal position but the nodal derivatives (i.e. $\mathbf{x}_u, \mathbf{x}_v, \mathbf{x}_{uu}, \mathbf{x}_{uv}, \mathbf{x}_{vv}$ where $\mathbf{x}_u = (1, 0, \frac{\partial z}{\partial x})$ and $\mathbf{x}_v = (0, 1, \frac{\partial z}{\partial y})$ etc.) as well, all the information needed to compute the first and second fundamental forms of the surface is generated automatically. The intrinsic differential characteristics of the surface, such as the unit normal and the principal curvatures, can be conveniently computed from this information.

7 Applications

We have applied our surface fitting technique using the deformable sheet model on sampled laser range data and a 3D digital terrain map. Our experiments run at interactive rates on a Silicon Graphics Indigo workstation.

7.1 Fitting Surfaces to 3D Range Data

This section demonstrates the reconstruction of surfaces from regularly sampled laser range data. Range data based surface reconstruction has practical applications in the areas of visualization and CAD based manufacturing. The range data used in the experiments are from the NRCC 3D image database [10].

In the first set of experiments we fit a 50×50 element deformable sheet to the following range images

- 5000 range data points randomly sampled from a 128×128 pixel range image of a statuette (Fig. 4(a)) (NRCC image database CAT #155).
- 2599 range data points randomly sampled from a range image of the upper "hemisphere" of an egg (Fig. 4(d)) (NRCC image database CAT #233).
- 7696 range data points randomly sampled from a range image of the upper part of a mug with a pitted surface (Fig. 4(g)) (NRCC image database CAT #251).

For this set of experiments, the weights α_{ij} and β_{ij} were set to 0.8 and 0.01 respectively, the force gain parameter κ was set to 30.0 and the time step Δt was set to 0.002. The fitting process required approximately 80 seconds, 40 seconds, and 80 seconds to complete for each of the three data sets respectively on a Silicon Graphics Indigo workstation. Both implementations of the sheet model were fitted to the range images listed above, with similar results. Figure 4(c) shows the result of the rectangular finite element version of the sheet, while figures 4(f) and 4(i) show the result of the triangular element version.

7.2 Digital Terrain Mapping

We used the deformable sheet to reconstruct 3D terrain models from cartographic data. For this experiment we extracted isoelevation contours from a 256×256 digital terrain map. Figure 5(a) shows a rendering of the digital terrain map. We subsampled the terrain map using 8 contour lines at 200m intervals (Fig. 5(b)) and then fit the sheet to the resulting sparse contour data. Figure 5(c) shows the final fitted

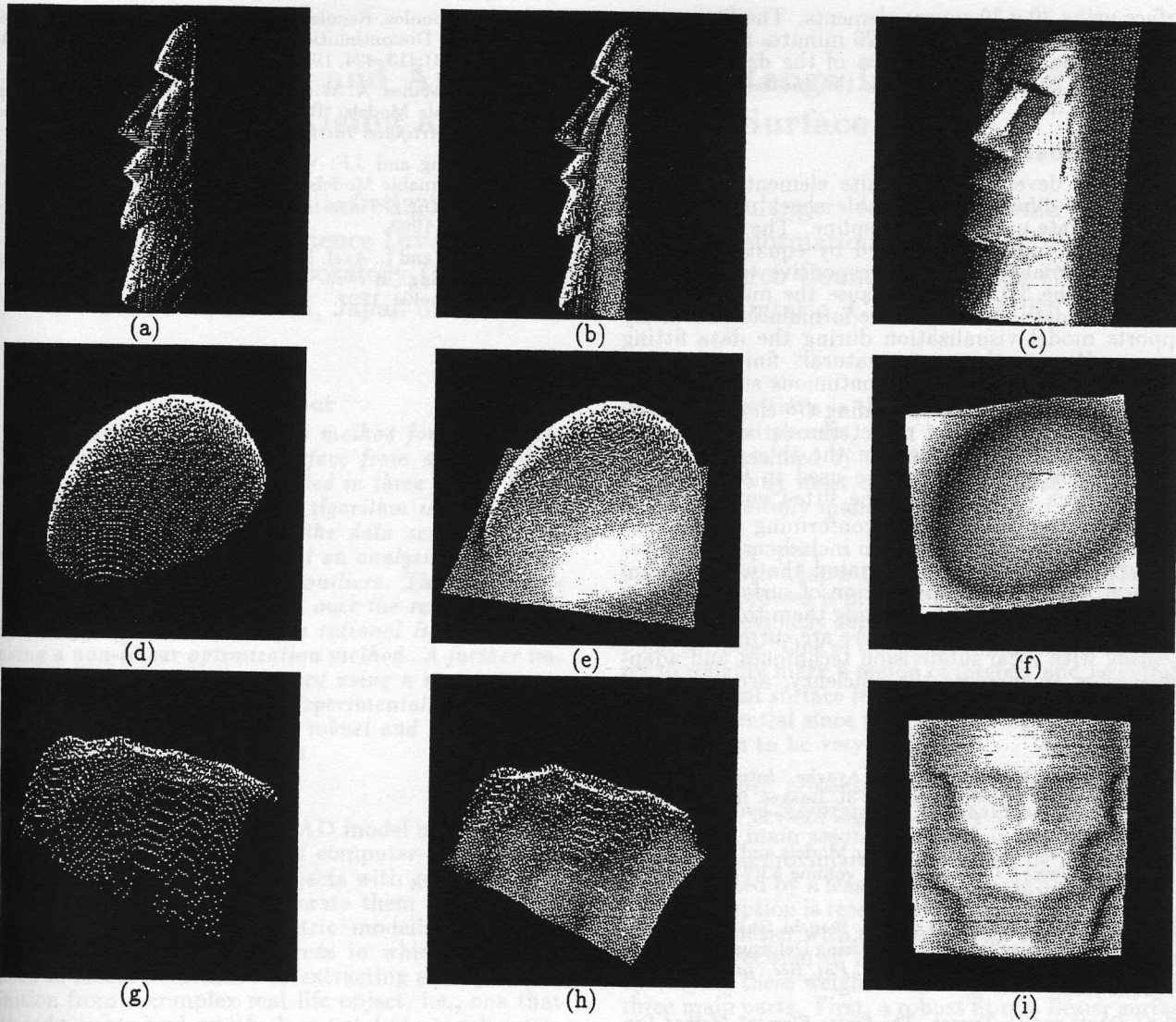


Figure 4: (a)(d)(g) Range data images. (b)(e)(h) Sheet deforming to data. (c)(f)(i) Final reconstructions.

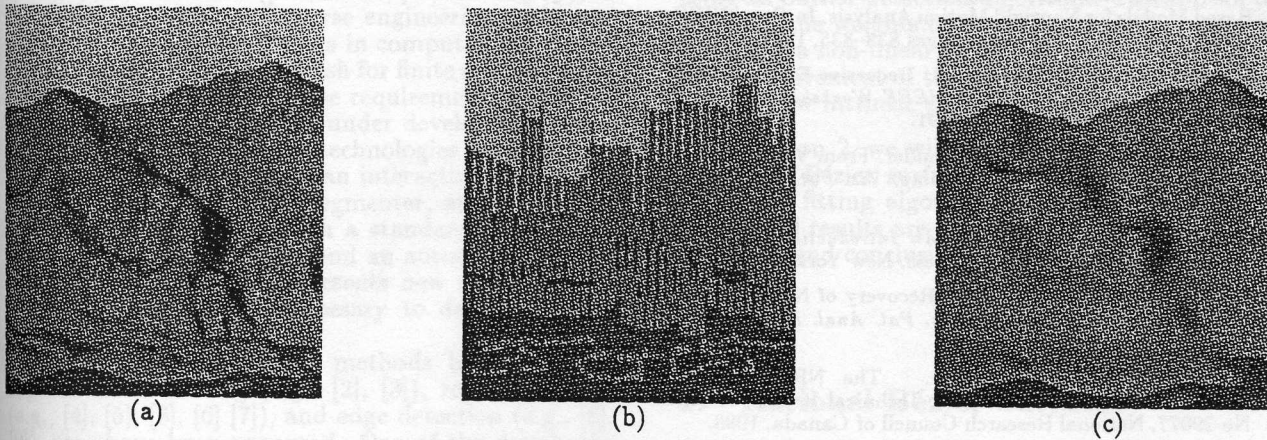


Figure 5: (a) Original digital terrain map. (b) Rendered contour data. (c) Reconstructed terrain.

surface using 30×30 square elements. The fitting process takes on the order of 20 minutes to complete, primarily due to the sparseness of the data; in areas where there is little or no data the model takes more time to "fill in".

8 Conclusions

We have developed two finite element implementations of a dynamic deformable sheet model based on a thin-plate under tension spline. The behavior of the dynamic model is governed by equations of motion, which make the model responsive to forces derived from the 3D data and cause the model to conform to the data. The dynamic formulation naturally supports model visualization during the data fitting process. We use the most "natural" finite elements to represent the model as a continuous surface. The triangular element is a conforming C^1 element whose nodal degrees of freedom reflect the derivative terms found in the energy expression the thin-plate under tension spline and they can be used in the analysis of the surface properties of the fitted surface. The rectangular element is also a conforming C^1 element whose nodal degrees of freedom include useful derivative terms. We have demonstrated the usefulness of our sheet model for reconstruction of surfaces with irregular shape features by applying them to laser range data and digital terrain data. We are currently experimenting with local subdivision techniques and adaptive meshes to increase the efficiency, accuracy, and flexibility of our model.

References

- [1] I. Cohen, L.D. Cohen, and N. Ayache. Introducing New Deformable Surfaces to Segment 3D Images. In *Proc. IEEE Conf. Comp. Vis. Pat. Rec.*, pages 738-739, June 1991.
- [2] L.D. Cohen. On Active Contour Models and Balloons. In *CVGIP: Image Understanding*, volume 53(2), pages 211-218, March 1991.
- [3] H. Delingette, M. Hebert, and K. Ikeuchi. Shape Representation and Image Segmentation Using Deformable Surfaces. In *Proc. IEEE Conf. Comp. Vis. Pat. Rec.*, pages 467-472, June 1991.
- [4] G. Dhatt and G. Touzot. *The Finite Element Method Displayed*. Wiley, New York, 1984.
- [5] W.C. Huang and D.B. Goldof. Adaptive-Size Physically-Based Models for Nonrigid Motion Analysis. In *Proc. IEEE Conf. Comp. Vis. Pat. Rec.*, pages 833-835, 1992.
- [6] D. Metaxas and D. Terzopoulos. Recursive Estimation of Shape and Nonrigid Motion. In *IEEE Workshop on Visual Motion*, pages 306-311, Oct. 1991.
- [7] O. Monga, N. Ayache, and P. Sander. From Voxel to Curvature. In *Proc. IEEE Conf. Comp. Vis. Pat. Rec.*, pages 644-649, June 1991.
- [8] D.H. Norrie and G. deVries. *An Introduction to Finite Element Analysis*. Academic Press, New York, 1978.
- [9] A. Pentland and B. Horowitz. Recovery of Nonrigid Motion and Structure. *IEEE Trans. Pat. Anal. Mach. Intel.*, 13(7):730-742, July 1991.
- [10] M. Rioux and L. Cournoyer. The NRCC Three-Dimensional Image Data Files. Technical Report CNRC No 29077, National Research Council of Canada, 1988.
- [11] D. Terzopoulos. Multilevel computational processes for visual surface reconstruction. *Computer Vision, Graphics, and Image Processing*, 24:52-96, 1983.
- [12] D. Terzopoulos. Regularization of Inverse Visual Problems Involving Discontinuities. *IEEE Trans. Pat. Anal. Mach. Intel.*, 8(4):413-424, 1986.
- [13] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on Deformable Models: Recovering 3D Shape and Nonrigid motion. *Artificial Intelligence*, 36(1):91-123, 1988.
- [14] Y.F. Wang and J.F. Wang. Surface Reconstruction using Deformable Models with Interior and Boundary Constraints. *IEEE Trans. Pat. Anal. Mach. Intel.*, 14(5):572-579, May 1992.
- [15] A. Young and L. Axel. Non-Rigid Heart Wall Motion using MR Tagging. In *Proc. IEEE Conf. Comp. Vis. Pat. Rec.*, pages 399-404, 1992.