

# Efficient Nonlinear Finite Element Modeling of Nonrigid Objects via Optimization of Mesh Models<sup>1</sup>

Leonid V. Tsap, Dmitry B. Goldgof, and Sudeep Sarkar

*Department of Computer Science and Engineering, University of South Florida, Tampa, Florida 33620*  
E-mail: tsap@csee.usf.edu, goldgof@csee.usf.edu, sarkar@csee.usf.edu

and

Wen-Chen Huang

*Department of Health-Care and Hospital Management, Chia Nan College of Pharmacy and Science, Tainan, Taiwan, Republic of China*  
E-mail: wenh@chna.chna.edu.tw

Received October 15, 1996; accepted August 15, 1997

In this paper we propose a new general framework for the application of the *nonlinear finite element method* (FEM) to nonrigid motion analysis. We construct the models by integrating image data and prior knowledge, using well-established techniques from computer vision, structural mechanics, and computer-aided design (CAD). These techniques guide the process of optimization of mesh models.

Linear FEM proved to be a successful physically based modeling tool in solving limited types of nonrigid motion problems. However, linear FEM cannot handle nonlinear materials or large deformations. Application of nonlinear FEM to nonrigid motion analysis has been restricted by difficulties with high computational complexity and noise sensitivity.

We tackle the problems associated with nonlinear FEM by changing the parametric description of the object to allow easy automatic control of the model, using physically motivated analysis of the possible displacements to address the worst effects of the noise, applying mesh control strategies, and utilizing multiscale methods. The combination of these methods represents a new systematic approach to a class of nonrigid motion applications for which sufficiently precise and flexible FEM models can be built.

The results from the skin elasticity experiments demonstrate the success of the proposed method. The model allows us to objectively detect the differences in elasticity between normal and abnormal skin. Our work demonstrates the possibility of accurate computation of point correspondences and force recovery from range image sequences containing nonrigid objects and large motion. © 1998 Academic Press

## 1. INTRODUCTION

### 1.1. Motivation

Motion of physical objects is often *nonrigid*. Nonrigid motion can be represented as a transition between some extremal shapes. There is no elegant rule to describe this kind of motion. However, the demand for nonrigid motion estimation algorithms is great and coming mostly from sciences that are studying the behavior of human body parts such as medical imaging (including cardiac motion, hand and knee modeling, etc.), face and gesture recognition, and videoconferencing. One way to describe nonrigid motion is to employ the laws of rigid and nonrigid dynamics expressed through a set of Lagrangian equations that govern this motion. For this reason, *physically based models* are superior in terms of modeling complex local deformations. Using physically based modeling, the characteristics of these shapes can be incorporated into models and described in terms of forces and displacements. This approach defines physical principles and allows numerical simulation of the model behavior. In computer vision, the physically based modeling is used most often for 3-D shape fitting and motion analysis.

Prior work in deformable model-based nonlinear motion analysis was limited in a sense that it did not make use of nonlinear finite elements. It was justified partially because of the considerable computational complexity of the nonlinear method in its “pure” form. Some authors used linear elements, handling nonlinearities in different ways, e.g., global deformations, explicit rigid body modes, etc. Others made use of nonlinear spring-mass models. Since most real-world materials behave nonlinearly when they are subjected to large deformations, one cannot expect a linear FEM to be a good model in many situations. This restricts linear FEM application in many computer vision problems.

We consider the problem of tracking point correspondences on the surface of elastic deforming objects with known material

<sup>1</sup> This research was supported in part by the Whitaker Foundation Biomedical Engineering Research Grant and in part by the National Science Foundation Grant NSF IRI-9619240.

properties. *Nonlinear FEM* allows modeling of both material nonlinearities in the form of nonlinear properties and geometric nonlinearities in the form of the large deformations. The algorithm for point correspondence recovery in nonrigid motion uses assumptions of known material properties and multiple (but unknown) forces applied to the object. The algorithm builds a parametric description of the object's motion through point correspondence recovery on the surface of the deformable object and also recovers the forces that were exerted on the object (consistent with sensed object surface before and after the deformation).

A computer vision approach is different from a traditional physical approach in the sense that our knowledge of the object is limited and success very often can be determined by the possibility of recovering point correspondences between new and known shapes. This limitation is the main reason for the problems researchers face when using physically based modeling for nonrigid motion analysis. The problems include high computational complexity, inadequate quality of results, difficulties in controlling the model and noise sensitivity. We propose solutions to the above problems including changes in the parametric description of the object to allow easy automatic control of the model, analysis of the possible displacements to address the worst effects of the noise, application of the mesh control strategies, and utilization of multiscale methods.

## 1.2. Previous Work

In nonrigid motion analysis, dynamic shape modeling provides the mechanism for fitting and tracking visual data. Using deformable models, unstructured elastic motion can be represented compactly by a small number of parameters. The task of motion recovery is then reduced to the problem of parameter estimation.

Physically based modeling has shown excellent results. Terzopoulos *et al.* [1] used a *membrane-thin-plate elastic* model for 3-D object reconstruction. Wang and Wang [2] used elastic, deformable models and *Hamilton's principle* to reconstruct surfaces.

*Snakes* are a class of active contour models first proposed by Kass *et al.* [3]. Terzopoulos and Waters [4, 5] exploited a physically based and anatomical model of human facial animation. They developed a snake to track the nonrigid motions of facial features in video images. Leymarie and Levine [6] used a snake to track deformable (nonrigid) objects in a noisy intensity image. They proposed an improved terminating criterion for the minimization of the intensity energy. Kumar and Goldgof [7] used the active contour (snake) to track the tagged grid in Cardiac MR images automatically.

Furthermore, the snake model is easily generalized to deal with 3-D images, as shown by Cohen and Cohen in [8]. This newly formed class of deformable surfaces is called *balloons*. Cohen *et al.* have successfully applied this model to the segmentation of 3-D MRI images. McInerney and Terzopoluos [9] have addressed similar problems using a slightly different balloon model. Their focus was on the numerical solution with finite

elements. Cohen and Cohen [10, 11] presented a finite element method to solve a model of deformation of a balloon similar to the "snake" model. They used this model to track a series of 2-D slices of heart ventricles and make a 3-D reconstruction of the inside surface of the ventricles.

The snake model can be reduced to a spline function when external forces are removed. Bookstein [12] illustrated the potential applications of thin-plate splines, including the modeling of biological shape change, production of biomedical atlases, image feature extraction, etc.

Other developments on deformable models were done by DeCarlo and Metaxas [13], who proposed to incorporate blending into the model, forming an evolution from the initial to the final shape. Neuenschwander *et al.* [14] defined deformable surfaces using elastic model representation as triangulated meshes.

Terzopoulos and Vasilescu [15–17] used numerical grid generation to reconstruct the surface. They develop adaptive mesh models that are assembled from nodal points connected by adjustable springs. The adaptive mesh models have been implemented on the intensity data and range data. The external force moves the nodal point only along a specific direction. Furthermore, they develop techniques for adaptive hierarchical subdivision of adaptive meshes. Huang and Goldgof [18–20] built adaptive-size meshes to reconstruct and analyze nonrigid motion. In that approach, the mesh size increases or decreases dynamically during the surface reconstructing process to locate nodes near surface areas of interests (like high curvature points) and to optimize the fitting error. Among other related publications there are simplex meshes (Delingette [21]) for recovering 3-D objects.

Pentland and Sclaroff [22] presented a closed-form and efficient solution for physically based modeling and recognition. They generate the eigenvector transformation from the finite element method (FEM) to obtain the closed form solutions. Pentland [23] developed a system that is capable of automatically recovering deformable part models based on the finite element method. By limiting the number of deformation modes used in the representation, the analysis of nonrigid motion always can be transformed to an overconstrained problem. Later, the same model combined with an extended Kalman filter is applied by Pentland and Horowitz [24] to recover nonrigid motion and structure from contour. Their model allows overconstrained estimates of 3-D nonrigid motion from optical flow data [22]. To avoid extra parameterization, Sclaroff and Pentland [25] developed a new method that computes the object's vibration modes directly from the image data. New extensions of this work by Sclaroff and Pentland [26] use FEM to obtain a parametric description of nonrigid motion in terms of its similarity to known extremal views.

Young and Axel [27] built a finite element model of the left ventricle to fit material points tracked in biplanar views. They described a measure of deformation energy suitable for fitting deformable models to image data. However, their measurement of strain energy may not be optimal since the heart wall motion

is complex and may be nonlinear. Metaxas and Koh [28] used local adaptive finite elements to represent 3-D shapes efficiently. They subdivided the elements for the local deformation based on the distance between the given data points and the model, and calculate the forces. Gourret *et al.* [29] developed a finite element method to simulate the human skin deformations between objects and hands during a grasping process. They apply loads in terms of *displacements* instead of forces and/or moments—the same as the nonlinear FEM approach does in this work. Grasping of a ball subject to internal pressure and the animation of successive steps of finger flexing without contact are shown in their paper.

Nastar and Ayache [30] have attempted to unify the work of Terzopoulos *et al.* and Pentland *et al.* They followed similar physics principles and developed elastic models for nonrigid motion tracking. They use a mass spring mesh to segment a 3-D magnetic resonance image of a human head and track the mitral value of the left ventricle on a set of ultrasound 2-D images. Guccione and McCulloch [31] used the axisymmetric finite element to model the left ventricle with a realistic geometry and fibrous architecture, physiological boundary conditions, and a 3-D constitutive equation. They computed the distributions of stress and strain of the passive left ventricle with promising results.

This work directly develops the ideas presented by Huang and Goldgof [32] and later refined by Huang, Goldgof, and Tsap [33]. It includes many new methods that improve the results and FEM performance.

## 2. THEORETICAL BACKGROUND

### 2.1. Theory of Elasticity and Stress-Strain Equations

The theory of elasticity shows that the response of a solid body to external forces is influenced by the geometric configuration

of the body as well as the mechanical properties of the material. *Elastic materials* are materials in which the deformation and stress disappear with the removal of external forces.

The stress  $p$  in the element  $\Delta A$  of the area is defined as the average force per unit area when the area  $\Delta A$  approaches zero, or

$$p = \lim_{\Delta A \rightarrow 0} \frac{\Delta F}{\Delta A} = \frac{dF}{dA}. \quad (1)$$

The stress vector depends on the location of the point as well as the orientation of the surface through the point. Let  $\sigma$  represent the normal stress, i.e., the component of stress *perpendicular* to the plane in which it acts (see Fig. 1a). The shear stress is the component of stress which lies *in* the plane and is denoted by the symbol  $\tau$ . Thus, we have

$$p^2 = \sigma^2 + \tau^2. \quad (2)$$

The strain,  $\epsilon$ , is defined as the change in displacement,  $u$ , with the change in length,  $x$ , or

$$\epsilon = \frac{\partial u}{\partial x}. \quad (3)$$

The *modulus of elasticity*, or Young's modulus,  $E$ , can be defined as

$$E = \frac{\Delta \sigma}{\Delta \epsilon}, \quad (4)$$

where  $\Delta \sigma$  is the stress change and  $\Delta \epsilon$  is the strain change. The stress,  $\sigma$ , can be viewed as force per unit area and the strain,  $\epsilon$ , as change of length per unit length.

*Poisson's ratio*,  $\mu$ , is defined as the ratio of the magnitude of the transverse strain to the magnitude of the axial strain. In other words, consider a parallelepiped element under a uniaxial

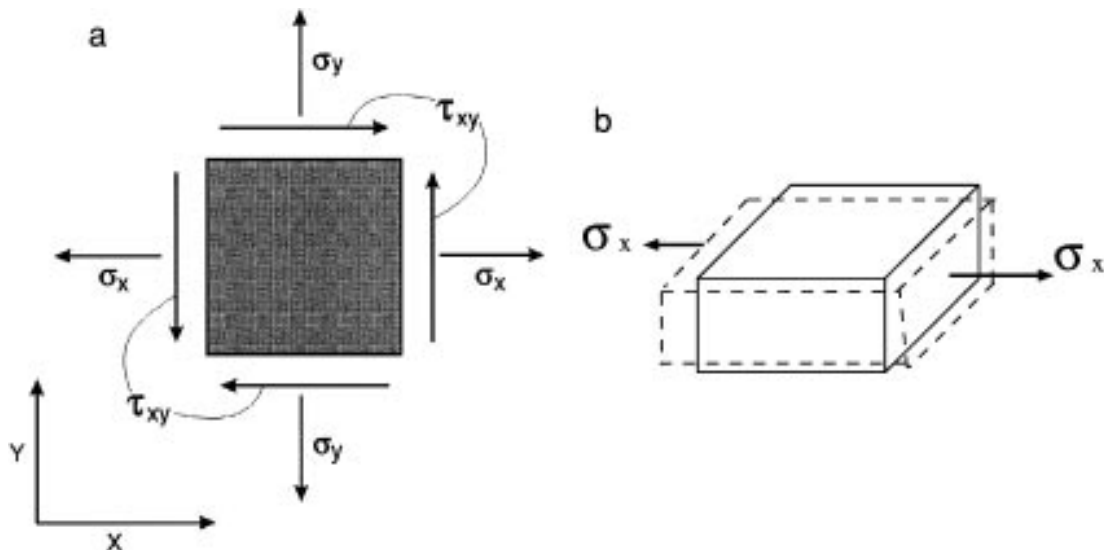


FIG. 1. Element stress. (a) Normal and shearing components. (b) Uniaxial Stress  $\sigma_x$ .

stress  $\sigma_x$  shown in Fig. 1b. When there is an elongation in the  $x$  direction, there will be contractions in the  $y$  and  $z$  directions. These are given by

$$\epsilon_y = \epsilon_z = -\mu(\sigma_x/E). \quad (5)$$

For a general state of stress the stress–strain relations, which are known as “generalized Hooke’s law,” consist of the equations

$$\begin{aligned} \epsilon_x &= \frac{1}{E}[\sigma_x - \mu(\sigma_y + \sigma_z)] & \lambda_{xy} &= \frac{1}{G}\tau_{xy} \\ \epsilon_y &= \frac{1}{E}[\sigma_y - \mu(\sigma_x + \sigma_z)] & \lambda_{yz} &= \frac{1}{G}\tau_{yz} \\ \epsilon_z &= \frac{1}{E}[\sigma_z - \mu(\sigma_x + \sigma_y)] & \lambda_{zx} &= \frac{1}{G}\tau_{zx}, \end{aligned} \quad (6)$$

where  $\lambda_{xy}$  is a shear strain component (of a shear stress component  $\tau_{xy}$ ) and  $G$  is called the modulus of elasticity in shear, or the modulus of rigidity:

$$G = \frac{E}{2(1 + \mu)}. \quad (7)$$

These relations complete the system of field equations necessary to formulate a problem in elasticity.

General *nonlinear* deformation theory defines the displacement field as a combination of rigid-body motions and pure deformations [34]. Rigid-body motions include translations and rotations. Their main property is that the distance between any pair of material points remains unchanged. Any quantity that measures the *change* in length between the neighboring points is a measure of pure deformation.

## 2.2. Finite Element Method

The FEM is a computer-aided mathematical technique for obtaining approximate numerical solutions to the abstract equations of calculus that predict the response of physical systems subjected to external influences [34]. Hence, general mathematical formulation is derived from a physical problem and then simplified to be solved with a model. FEM is one of the methods for finding an approximate solution for a simplified model.

FEA (finite element analysis) makes it possible to predict the response of the system (as a physical object composed of various materials) that is subjected to the external influences (forces, temperatures, voltages, etc.). The result generally represents a numerical solution to the governing equations (expressing conservation or balance of some physical property such as mass, energy, or momentum) and loading conditions (externally originating forces, fields, etc.) that together characterize and determine the behavior of the system.

FEA begins by making a finite element model of the object. The model is an assemblage of finite elements which are pieces of various sizes and shapes. For each element the governing equations, usually in differential or integral form, are

transformed into algebraic element equations which are an approximation of governing equations and much easier to solve. The resulting numbers are assembled into the system equations that characterize the response of the entire system.

The finite element model usually contains the following information about the object:

(1) geometry, subdivided into finite elements, (2) materials, (3) constraints, and (4) forces. Geometry is the only item on the list that we can have difficulty representing even for traditional engineering application. It becomes a more intricate task for computer vision problems, since we can have problems understanding the geometry (using available range data) before describing it.

The way FEA obtains the stresses, temperatures, fields or other desired unknowns in the model is by minimizing an energy functional, which consists of all energies associated with the particular model. *The law of energy conservation* states that the finite element energy functional must equal zero. The basic equation for FEA is

$$\frac{\partial F}{\partial p} = 0 \quad (8)$$

where  $F$  is the functional and  $p$  is the unknown grid point potential to be calculated, which varies with the type of problem [35].

## 2.3. Comparison between Linear and Nonlinear FEM

In general, linear FEM can be expressed as

$$[K]u = F, \quad (9)$$

where  $[K]$  is a matrix of stiffness coefficients,  $u$  is a displacement vector, and  $F$  is a force vector. In linear analysis,  $u$  is a linear function of  $F$ . The displacement of the structure is directly proportional to the force. The proportionality factor is the structural stiffness  $K$ , which is constant in a linear structure. The column vector  $u$  represents the unknown DOFs in the model. Typical FEA problems are large and  $u$  might contain more than a million unknowns [35]. Linear equations can be solved using standard solvers. One pass through the solver, called an *iteration*, is usually enough to reach an approximate solution.

The static nonlinear problem can be expressed as

$$[K(u)]u = F(u), \quad (10)$$

where both a matrix of stiffness coefficients and a force vector depend on  $u$ . Nonlinear structural behavior produces the response which is not proportional to the input.

“Linear” structures usually are truly nonlinear, but the degree of nonlinearity is small enough to be neglected. For example, in

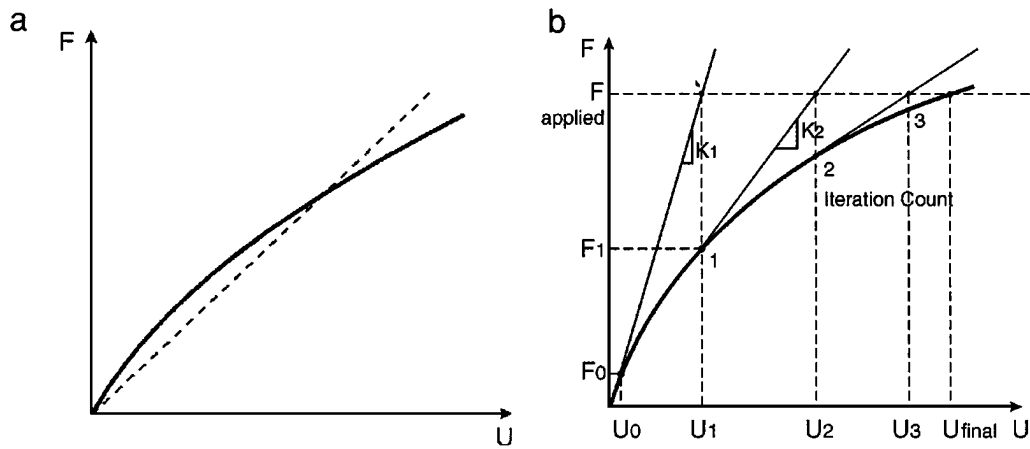


FIG. 2. Force-displacement relationships. (a) Material true "linearity." (b) Iterative solution.

Fig. 2a a solid line shows true, nonlinear behavior, and a dashed line shows assumed, linear behavior.

Nonlinear analysis must be considered if large displacements occur with linear materials (*geometric nonlinearity*), small displacements occur with nonlinear stress-strain relationships for structural materials (*material nonlinearity*), or a combination of both effects. The first kind of nonlinearity is the one predominantly used in this work.

Geometry can produce significant nonlinear effects. Linear elements are formulated under the assumption that forces and displacements are small and that any displacement of the grid points does not change the geometry of the element. This is the small strain assumption. Nonlinear analysis is needed whenever elements move a significant distance and/or experience large strain. Large deflection analysis assumes the displacements may be large enough to effect the stiffness significantly. Since the stiffness is affected by the displacements, and vice versa, an iterative solution is required.

The most important *nonlinear effect* is due to the behavior of *materials*. When the forces and responses are small, material

properties are constant, independent of forces. Material nonlinearities occur when material properties vary with the type and level of load. This variation causes the structural stiffness matrix to vary with the load as well, resulting in nonlinear behavior. Nonlinear stress-strain behaviors fall into two categories—elastic and plastic. In our experiments we consider *elastic* materials—materials that follow the same path in both loading and unloading.

Nonlinear behavior cannot be represented directly with a set of linear equations. However, a series of successive linear approximations with corrections can be used to solve nonlinear problems. Each linear approximation requires one pass, or iteration, through the equation solver (see Fig. 2b).

In the nonlinear analysis, the stiffness matrix  $[K]$  (and possibly the load vector  $\{F\}$ ) varies with the applied load; hence it is unknown. The procedure used to solve such a problem (Fig. 3) in general requires multiple iterations, where each iteration again is one pass through the solver.

The ANSYS program used in this work gives a choice of over 100 element types. Each element has properties corresponding

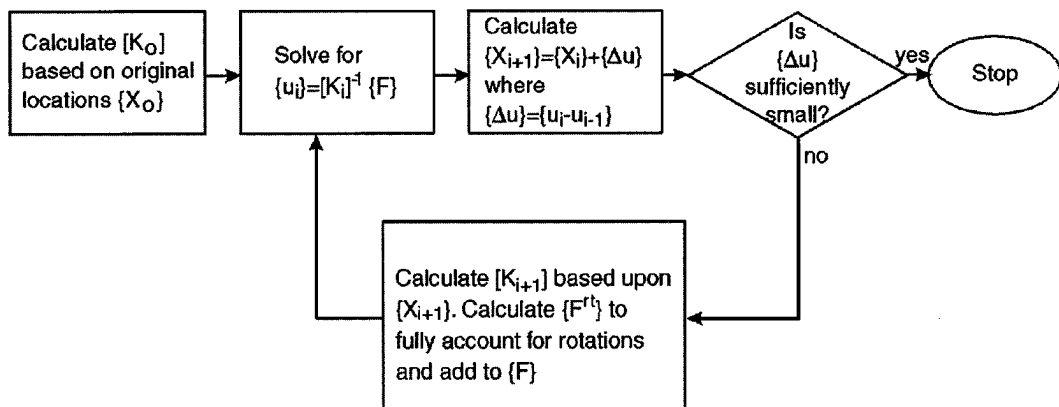


FIG. 3. An iterative solution algorithm.

to the analysis type, object structure, constraints, and loads. The element type used in this work is elastic shell which belongs to the class of structured 3-D shells.

#### 2.4. Large Strain Nonlinearities in Finite Element Method

The type of geometric nonlinearities we account for is *large strain*. It assumes that the strains are no longer infinitesimal. Shape changes and rotations are also accounted for. The applied loads on a body make it move (or deform) from the position  $\bar{u}_1$  to the position  $\bar{u}_2$ . Hence, the displacement vector is

$$\bar{u} = \bar{u}_2 - \bar{u}_1. \quad (11)$$

The deformation gradient can be defined as

$$[G] = \frac{\partial \bar{u}_2}{\partial \bar{u}_1}, \quad (12)$$

where deformation gradient includes the volume change, the rotation, and the shape change. The volume change at a point is

$$\frac{dV_2}{dV_1} = \det[G], \quad (13)$$

where *det* denotes determinant of the matrix. The deformation gradient can be separated into a rotation and a shape change using the right polar decomposition theorem,

$$[G] = [R][U_s], \quad (14)$$

where  $[R]$  is the rotation matrix and  $[U_s]$  is the right stretch (shape change) matrix. Once a stretch matrix is known, a logarithmic or Hencky strain measure is defined as

$$[\epsilon] = \ln[U_s], \quad (15)$$

or, equivalently, through the spectral decomposition of  $[U_s]$ ,

$$[\epsilon] = \sum_{i=1}^3 \ln(\lambda_i) e_i e_i^T, \quad (16)$$

where  $\lambda_i$  are eigenvalues of  $[U_s]$  (principal stretches) and  $e_i$  are eigenvectors of  $[U_s]$  (principal directions). Hence, from (14) we can calculate the average rotation at a point. Computationally, incremental approximation is used [36] and increment of the deformation gradient at the current time step  $n$  is defined through the previous time step  $n - 1$ :

$$[\Delta G_n] = [G_n][G_{n-1}]^{-1}. \quad (17)$$

### 3. DESCRIPTION OF THE ALGORITHM

#### 3.1. Nonlinear Finite Element Modeling

The main assumption of the regular FEM is that the forces (both force magnitudes and directions) are known. The displacements of the nodes can be obtained by applying the forces to the model. However, the force information is not available in most vision problems. On the other hand, it is sometimes possible to examine the displacements of points of the object as reactions to the external forces applied to this object. In our experiments, we called the points where these forces were applied *forced points*. Huang [37] assigned the displacements of *forced points* instead of applying the forces to the object.

This is an adaptation of the general *inverse problem* category used in nondestructive (electromagnetic, acoustic) testing [35]. We are given the resulting displacement and attempt to deduce the nature of the effect that caused the observed movement. The general inverse problem is not solved yet even for trivial engineering mechanics applications.

However, if the material properties are known, we can assign the displacements so that the reconstructed surface is the same as the sensed surface after motion. In many cases we do not have to know all material properties or we do not have to know them exactly. For example, the exact value of the modulus of elasticity is not necessary for the inverse problem solution according to the experiments. Elastic finite elements and the knowledge of displacements determine the behavior of the object. Relative forces can be recovered this way. Only in order to recover exact values of forces we need to know the precise value of Young's modulus. However, precision of Poisson's ratio is more important and contributes to the total error.

In order to recover the correct displacements we have to establish the point correspondences between two images. This is the most time-consuming part in the direct approach described next [37].

Assume that the 3-D coordinate values of all surface points before and after nonrigid motion are known. The 3-D coordinate values of these surface points are obtained from the sensing system, such as a range scanner, or from stereo images. The initial nonlinear FEM model, which includes nodes and elements, is built from the 3-D coordinate values of surface points before motion by simply transferring chosen points of the object into the nodes of the model. After assigning the displacements of *forced points* (if known), displacements of all the nodes can be computed by nonlinear FEM. The estimated location of the nodes after motion can be determined by adding the displacement of each node from its position before motion. Clearly, the situation in which the positions of *forced points* are available before and after motion is rare.

Alternatively, the exact corresponding points can be found by creating a set of hypotheses about possible corresponding *forced points*. Then the surface error which is associated with each hypothesis can be computed by comparing the sensed surface after the motion and the resulting surface calculated by nonlinear

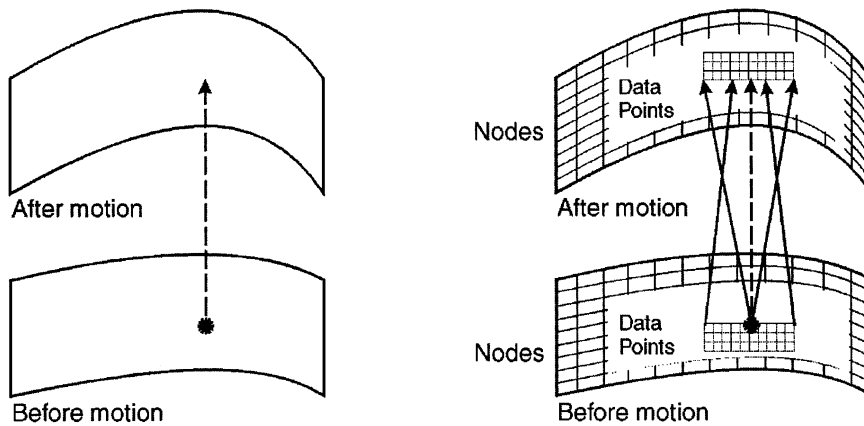


FIG. 4. Searching for the corresponding points using FEM.

FEM. Displacements of forced points are found by choosing the hypothesis with the smallest error. The error is defined as the distance between two shapes.

Figure 4 shows the possible search area for the hypotheses between two selected areas before and after the motion. From all the hypotheses, the one with the smallest surface error gives the best corresponding forced point before and after the motion. Figure 5 shows the flowchart of this implementation.

### 3.2. Changes in the Surface Description

The main limitations of the direct approach in its original form are high computational cost, relatively poor quality of results (estimated by the surface error and validation points error

as defined later in Section 4.1), and problems resulting from noise. In addition, transferring chosen points of the object into the nodes of the model and then entering the nodes limits us to simple objects because it does not allow us to properly describe either different volumes or different materials within the same object.

Both the quality of the results and the speed of the algorithm can be improved significantly by the “smarter” use of *a priori* knowledge. For example, one can observe (Section 4.2) that the sensed surface of the bending plate is not smooth (due to the acquisition noise). The reconstructed surface also is not smooth even though we know that the metal plate will remain smooth during bending (unless forces are quite large). This and the following sections present refinements of the direct approach.

The first step is defining the solid model through the keypoints. If we have some *a priori* knowledge of the object, this step provides an analytical characterization of the whole surface (or volume) and, as a result, drastically reduces problems caused by irregular sampling of the feature points. Some errors can be corrected by comparing the parts of the model against the list of geometric primitives:

- 2-D primitives such as circular, rectangular, and polygonal areas
- 3-D primitives such as cylindrical, hexahedral, prism, conical, spherical, and toroidal volumes.

More complex parts can be described using Boolean operations, including intersection, addition, and overlap. In the images with a significant amount of noise, when the planar objects cannot be represented as planar areas, then the procedure of fitting a nonplanar surface (Coons’ patch) is used.

Better analytical description of the object makes models more flexible, allowing us to change the partitioning into elements dynamically in the automatic mode based on the comparison of the actual and expected model responses. A uniform mesh, consisting of elements all about the same size and shape and repeated in a regular manner, is the easiest type to construct and modify later, if necessary.

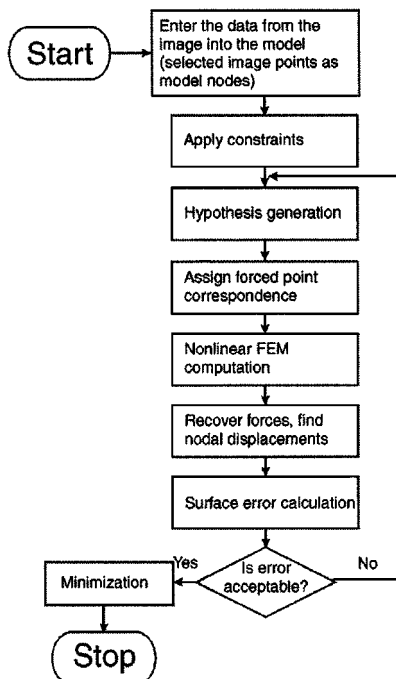
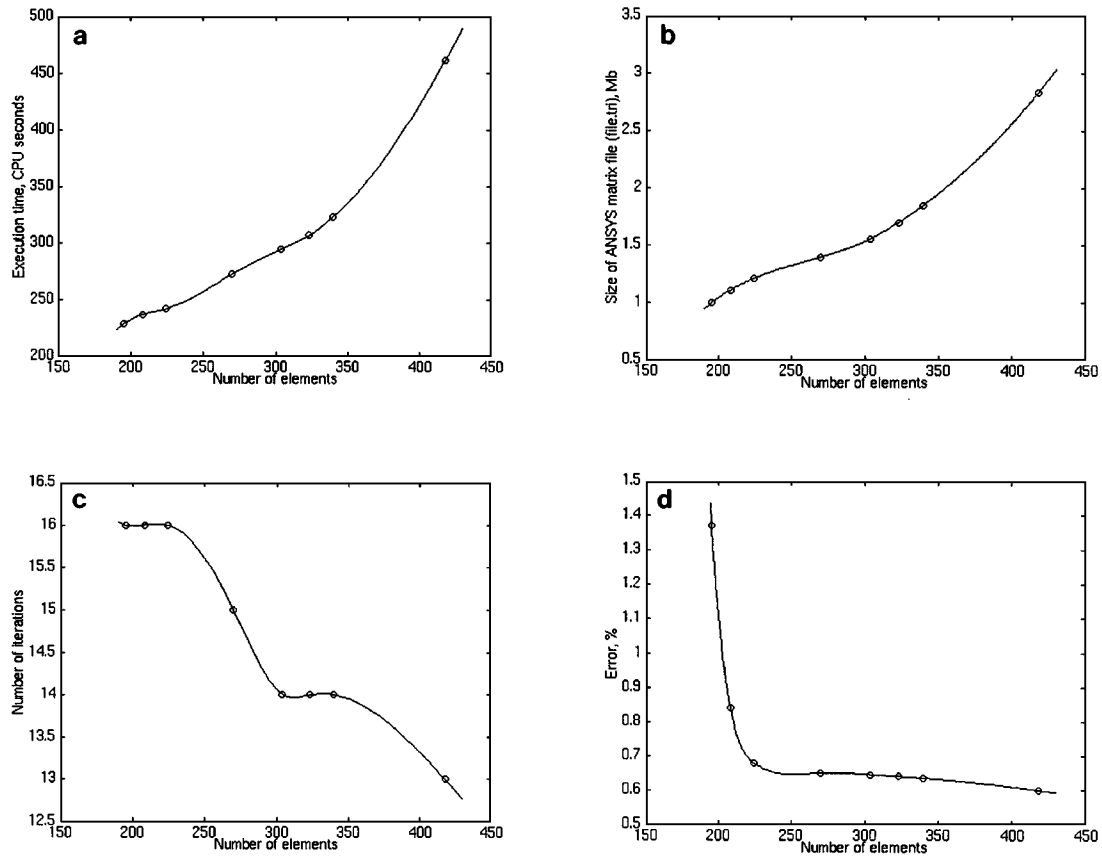


FIG. 5. Flowchart of the direct approach.



**FIG. 6.** (a) Execution time/number of elements relationship. (b) Size of one of the ANSYS data files/number of elements relationship. (c) Number of ANSYS iterations/number of elements relationship. (d) Calculated error/number of elements relationship.

### 3.3. Analyzing the Possible Displacements

Analyzing each hypothesis of the displacement is even more beneficial than the method described above. Blind incorporation of noise into the model affects the quality of results. However, trying to use noisy displacements with a physically based model is much worse. The model behavior becomes unpredictable (including no convergence within a reasonable time interval or convergence with unacceptable results). A local cartesian coordinate system is used to check the displacement in  $x$ ,  $y$ , and  $z$  directions. One heuristic solution is to use the biggest value as the scale and to round (approximate) two other (smaller) components of the displacement to this scale using an empirically determined threshold. This addresses the essential problem of noise in the displacement values. In the general case, displacement analysis is more complex and model dependent.

Further development of this idea would be some rule-based physical analyzer generating the list of viable displacements. In general, the success of this method will depend on *a priori* knowledge of a particular object. This procedure helps to avoid trying to move the model in a constrained direction. That not only improves the smoothness of the resulting surface, but also significantly improves the convergence of the FEM and makes the solution phase much faster.

### 3.4. Processing at Multiple Scales

An important limitation of the direct approach is the fact that the global search is computationally expensive, especially if we have many elements. However, a sufficiently large number of elements is necessary to achieve an accurate solution. Figure 6a shows that the execution time increases rapidly with the increase in the number of elements. The same happens to the file size (Fig. 6b). Since we need several data files of this size for each run, space requirements become almost as important as the computation time. However, reducing the number of elements increases the error (Fig. 6d). Multiscale processing was applied successfully to many computer vision problems (for example, edge detection [38]). The term *scale* denotes a level of resolution, or, equivalently, a level of detail. Most objects can be described usefully through a variety of scales. At the finest scale the texture becomes important; at the coarsest, the gross shape is primary. Then the outputs of operators at multiple scales are combined. This idea has found use most often as a mechanism to reduce computation: low-cost, low-resolution processing over a coarse grid (applied to the smoothed data) serving to guide high-cost, high-resolution processing over a finer grid.

FEM solutions obtained with coarse meshes are used in this work to guide solutions at finer scales. It allows us to achieve



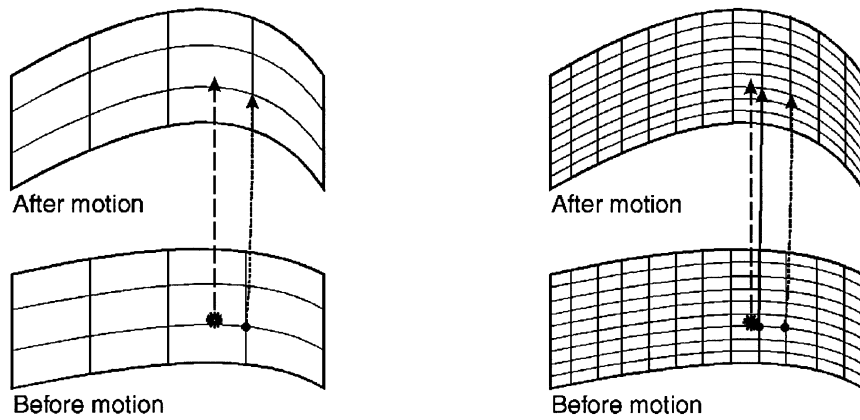


FIG. 7. Multiscale method.

further computational improvement. Initially, a coarse mesh is used to generate an approximate solution. Then the error is calculated for each hypothesis. The best result at a coarse scale is the starting point for the next scale. The process proceeds to an accurate solution by splitting up the existing elements and searching in the region where the solution (force position) was found. This is done in a completely automated manner without user intervention (Fig. 7).

### 3.5. Heuristic Search and Multiforce Strategy

Nonlinear FEM computation is the most time-consuming part of the algorithm. Since it is done for each hypothesis, reducing the number of hypotheses will drastically reduce the total computational time. Often a proportional relation exists between the force location and the difference between the surfaces before and after motion. Hence, it is logical to estimate this difference *before* choosing the initial hypothesis. “Estimate” means that, without point correspondences, precise calculation is impossible. However, even approximate surface differences can give us an idea about the relative value of the stress at a point. The direction in which this difference increases may help us estimate the location of the node where the force was applied. In general

this direction,  $\theta$ , and the optimization step,  $d$ , can be computed for the current step by

$$\theta = \tan^{-1} \frac{\partial S / \partial y}{\partial S / \partial x} \quad d = \sqrt{(\partial S / \partial y)^2 + (\partial S / \partial x)^2}, \quad (18)$$

where  $S$  is a distance along  $z$  direction between two surfaces. Then we move  $d$  away in the direction  $\theta$  and repeat the calculation. When this difference is no longer increasing, we can say that the forced point is found. As an alternative, this difference can be compared for all model nodes. This way we cannot locate the forced point exactly, but at least we can limit the number of hypotheses by limiting the search to some small surface region.

Hence, additional improvements in the quality can be achieved by replacing global (brute-force) search with smarter strategies of locating forced points where the surface error is bigger. Even after reaching a satisfactory solution, we can locate another force in the direction from the current force hypothesis where the error between the surfaces before and after motion is still increasing (Fig. 8). The methods of locating another component of the force again represent the trade-off between accuracy and

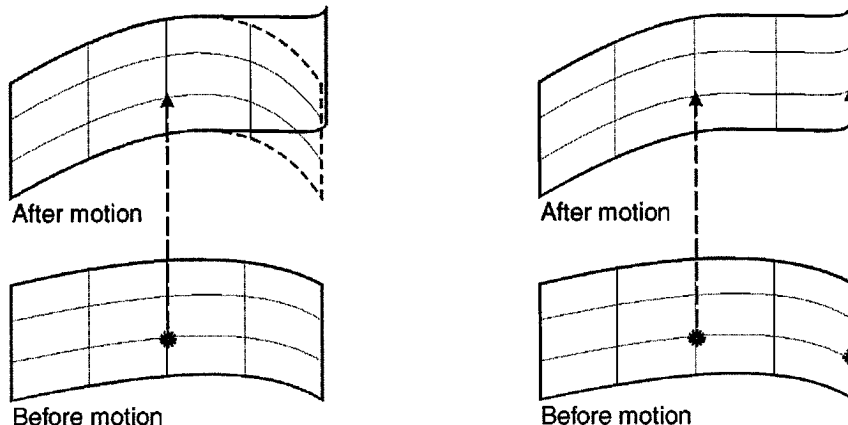


FIG. 8. Multiforce method.

speed—from exhaustive search to choosing one of the median nodes in that direction.

### 3.6. Mesh Control Strategies

In order to further improve the results in some applications (like the stretching rubber experiment in the next section) we need to specify mesh controls in the addition to the above steps. A finer mesh is necessary around the areas of high stress concentration. Many effects of nonrigid motion on the objects are *directional*, which can be shown by the error distribution. The element size can be uniformly reduced along some empirically determined direction after processing the results from the first quick solution. We define a smaller element size where significant displacements are expected. Complex objects usually consist of several regions (areas or volumes). *Local mesh refinement*, a common practice described by Burnett [34], provides additional degrees of freedom in some regions to increase the local accuracy. Based on the automated strategy, the areas with relatively high local error are identified as candidates for remeshing. It is hard to achieve completely automatic meshing based on the given accuracy since the error estimates are difficult and the automatic meshing problem has not been completely solved [35]. When the areas are found, their meshing is refined using smaller elements. In general, all domains of the model where the solution tends to be more complicated (e.g., near sharp changes in the shape or concentrated loads) must have meshing of greater density to maintain a given level of accuracy.

Figure 9 shows the combined flowchart of all variations of the algorithm. It includes the branches that are chosen depending on *a priori* knowledge of the object and forces applied to it.

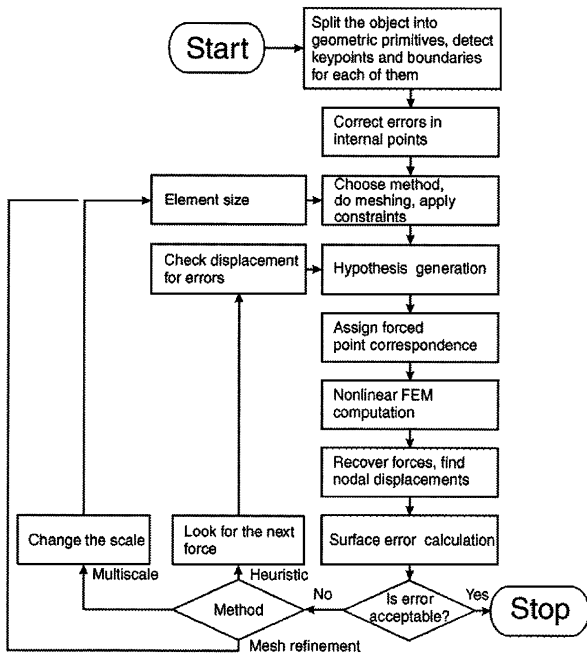


FIG. 9. Flowchart of the proposed algorithm.

## 4. EXPERIMENTAL RESULTS

This section presents the experimental results of applying nonlinear FEM to real data sequences. In order to test our algorithm, we utilize sequences of range images of a bending metal plate, a stretching rubber membrane, and a forearm skin. The range images are acquired by an active triangulation sensor based on the projection of binary coded patterns [39].

### 4.1. Error Computation

For this work we need to define two kinds of errors: the surface error and the validation points error.

We use the surface error to compare the results for each point correspondence hypothesis. The surface error can be defined as the Euclidean distance between nodes of the model and the object surface after motion and approximated as

$$E = \sum_{i=1}^n |\bar{n}_i - \bar{s}_i|/n, \quad (19)$$

when  $\bar{n}_i$  is the vector of 3-D coordinates of the  $i$ th node and  $\bar{s}_i$  is the vector of 3-D coordinates of the nearest surface point to  $\bar{n}_i$ . The displacements of *forced points* can be found by choosing the hypothesis which leads to the smallest surface error.

The dark points (i.e., validation points) in the intensity images are used only for validation. The error is computed by comparing validation points after the motion with a predicted position of validation points. The error is computed relative to the displacement of validation points. In other words, let  $\bar{b}_i$  be the  $i$ th validation point before motion, let  $\bar{a}_i$  be the  $i$ th validation point after motion, and let  $\bar{e}_i$  be the predicted position of the  $i$ th validation point after the deformation. The error is, therefore, defined as

$$\frac{1}{n} \sum_{i=1}^n \frac{|\bar{e}_i - \bar{a}_i|}{|\bar{a}_i - \bar{b}_i|}, \quad (20)$$

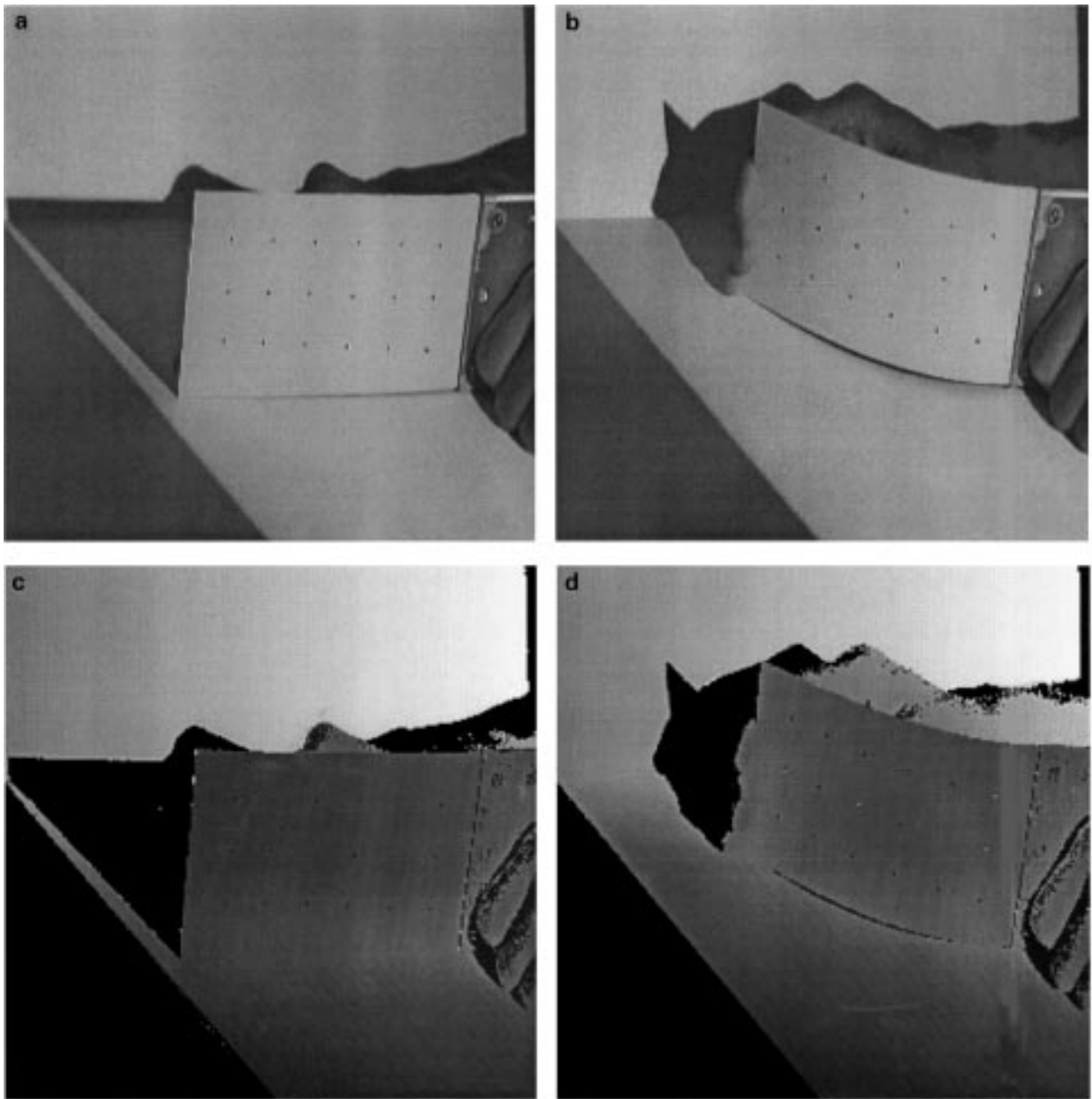
where  $n$  is the number of validation points and  $|\bar{e}_i - \bar{a}_i|$  and  $|\bar{a}_i - \bar{b}_i|$  are distances between points.

### 4.2. Bending Metal Plate

Figures 10a and 10b show the gray level intensity images of the bending metal plate before and after motion, respectively. The corresponding range images are shown in Figs. 10c and 10d.

Application of the direct approach (using nonlinear FEM) explained briefly in Section 3.1 was described in [33]. Linear FEM did not work at all because it could not find a satisfactory solution for such relatively big displacement values.

To prevent irregular feature sampling, keypoints and shape fitting are used (defining an area and its corresponding lines through the keypoints as explained in 3.2) to minimize an error in the location of internal nodes caused by the noise (if we know the geometry of the object before the motion). For both



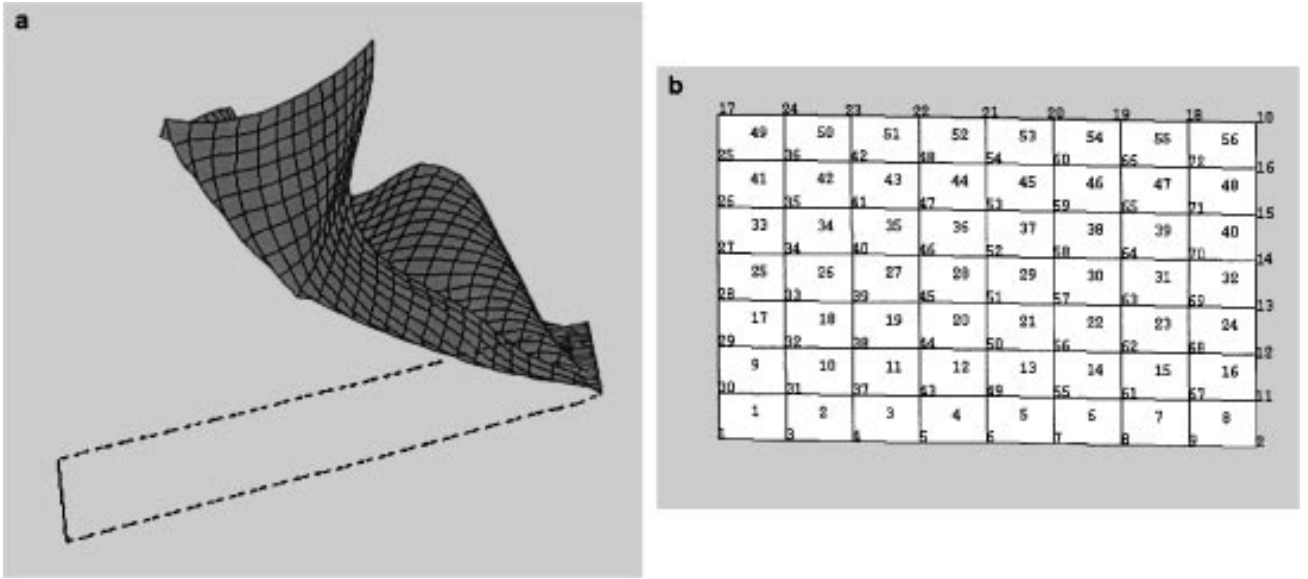
**FIG. 10.** (a) The intensity image of the bending metal plate before motion. (b) The intensity image of the bending metal plate after motion. (c) The range image of the bending metal plate before motion. (d) The range image of the bending metal plate after motion.

plate and membrane (Section 4.3) experiments keypoints are easy to locate along the boundary since we have simple surfaces before the motion. Then the resulting shape is split up into the elements. This gives us the flexibility to experiment with the element size and the number of nodes, which is important. Moreover, this description of the object allows easy automatic control.

Brute force attempts to use noisy displacements causes many problems in the model behavior, such as refusing to converge without a thousandfold increase in convergence values (from the default automatically calculated by ANSYS) and giving poor results (for example, Fig. 11a). The problems were cor-

rected using the method of displacement analysis described in 3.3. This procedure helps to avoid trying to move the model in the constrained direction. Our solution converges with a validation error of 0.68% after 16 iterations and the execution takes an average of 27 times less time than the basic algorithm implementation.

To use the multiscale method (defined in Section 3.4) for the plate we start with the coarse mesh—56 rectangular elements (72 nodes). Figure 11b shows the element and node structure for this initial model. After hypothesis generation the surface error is computed for every hypothesis. Figure 12 shows examples of results for various hypotheses.



**FIG. 11.** (a) Using noisy displacements leads to undesirable model behavior. (b) The element and node structure for the initial model (multiscale method).

The hypothesis with the smallest surface error is shown in Fig. 13a. An average time of evaluation for every hypothesis is 67.41 CPU seconds (s) on a SUN SPARC 10 with 32 MB of memory.

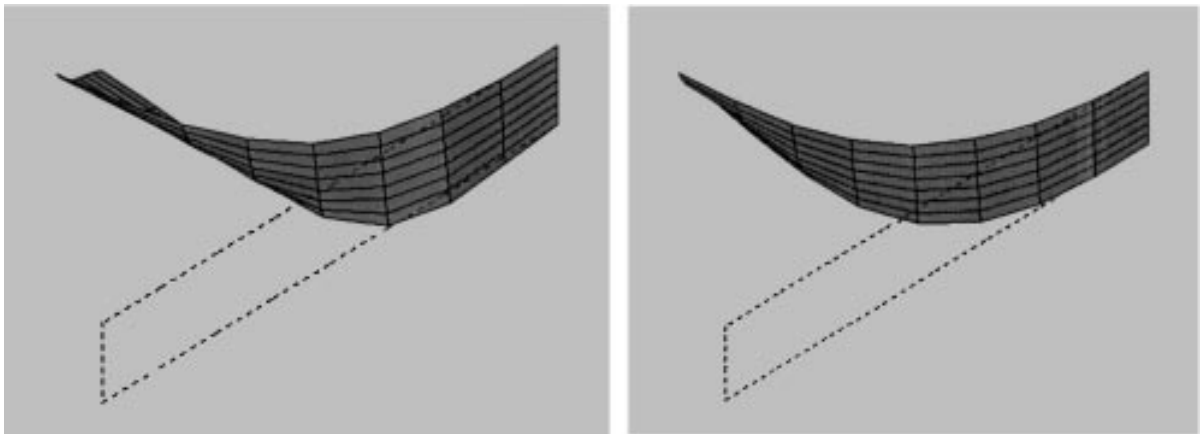
The true forced point may be located between the found node and its neighbors. The process proceeds to an accurate solution by remeshing the model using new size (length) of the element three times smaller than the original meshing and searching in the region where the solution was found at the previous scale. As a result of using the smaller element we now have 418 elements and 460 nodes (Fig. 13b). An average time of evaluation for every hypothesis is 376.54 CPU s on a SUN SPARC 10. We are decreasing the computational complexity in two ways: having almost nine times fewer cases to consider and spending 5.59 times less time on every case. The second step does not consume a significant amount of time, since remeshing

takes 1.2 CPU s and we have only nine new cases to go through. Figure 14a shows the final result.

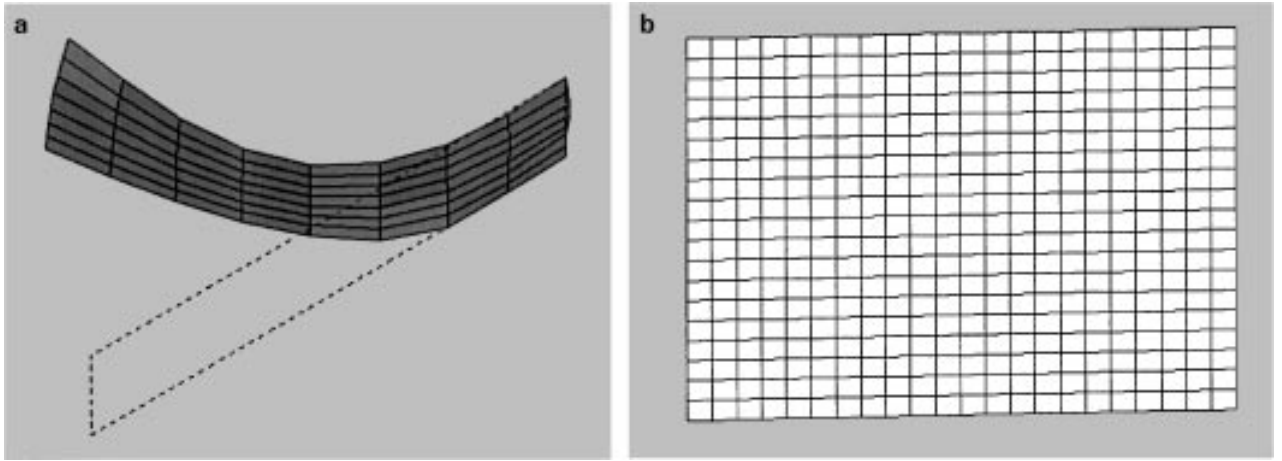
Additional improvements in the quality are achieved by replacing global search with the strategy of locating forced points where the surface difference (defined in Eq. (19)) is large. Even after reaching a satisfactory solution, we can locate another force in the direction of the current force hypothesis where the local error is increasing. The validation error in Fig. 14b is 0.42%. Figure 15 shows motion tracking of validation points.

#### 4.3. Stretching Rubber Membrane

Figures 16a and 16b show images of the stretching rubber before and after motion, respectively. Grid points painted on the rubber are used as the validation points. The corresponding range images are shown in Figs. 16c and 16d.



**FIG. 12.** Results for various hypotheses.



**FIG. 13.** (a) Hypothesis with the smallest surface error for the processing over a coarse scale. (b) New model at a fine scale.

Application of the direct approach to this experiment was also described in [33]. The model could not deal correctly with the geometry of the object and noise in the range data. Computational complexity was not addressed either.

In order to improve the results of the stretching rubber experiment, we need to employ directional mesh refinement (as explained in Section 3.6). The stress concentration area is easier to locate for this case compared with the bending plate experiment. Figure 17 shows both initial and refined meshing with constraints applied. Proposed directional refinement helps to avoid abrupt changes in the element sizes from very fine to very coarse, which may lead to ill-conditioning problems. We use the multiscale method again and start with only 96 elements.

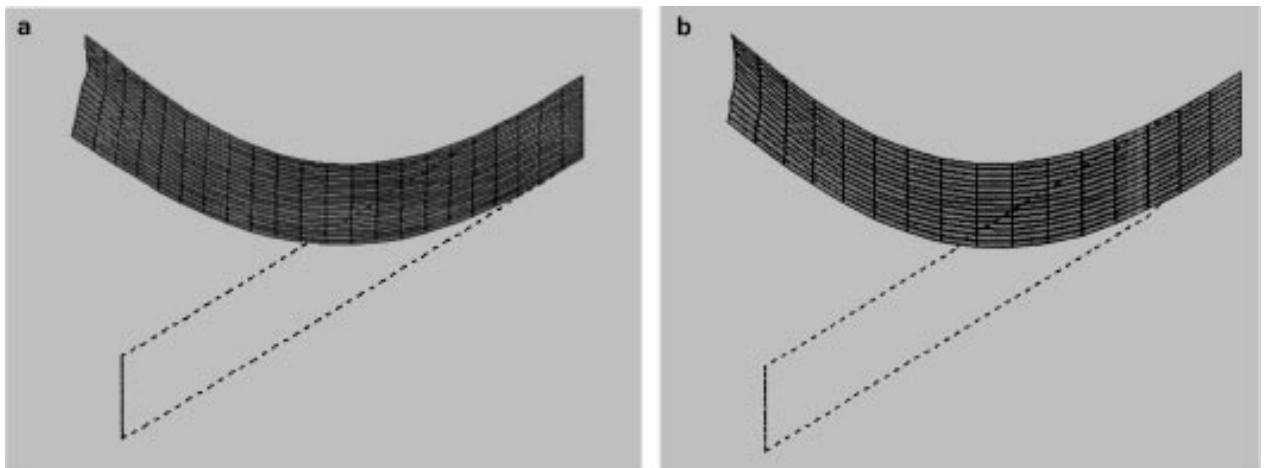
As a result, we are able to find the hypothesis with minimal surface error (Fig. 18a). To look for an accurate solution at a fine scale, we must do the remeshing. We do not split the existing elements since we have to keep our directional mesh generation.

Instead, we specify ten elements instead of five to be built along the line between two keypoints and change the ratio between the largest and smallest segments from 5 : 1 to 10 : 1. Remeshing gives us 312 elements (Fig. 18b). Again, we saved computation time because we would have had to look during the first step at almost three times fewer cases (by the number of nodes). Also, in each case calculation took 2.79 times less time compared to the fine scale computing. Using the fine scale resolution, we look at a small region only.

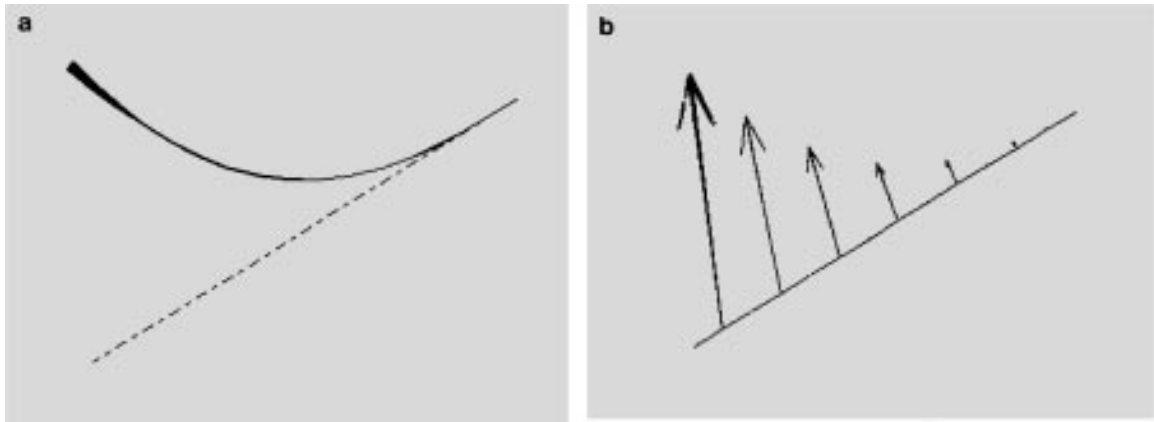
Figure 19 shows the reconstructed surface as a mesh. The validation error is 4.29% and the smoothness of the surface is very close to that expected. Heuristic search works even better than for the first experiment since the surface error is changing gradually from the boundaries to the center.

#### 4.4. Experiments with Sparse Data Sets

Figure 20 shows the results in the terms of validation error for sparse data sets. We are randomly omitting sensed points and



**FIG. 14.** (a) The result of multiscale method. (b) Improved results by multforce algorithm.



**FIG. 15.** (a) Initial and reconstructed surfaces. (b) Motion tracking of validation points.

calculating the validation error to see how the results deteriorate. The dashed line shows relationships when we interpolate the surface before looking for forced points (in the hypothesis generation stage). Even with more than 50% of the points not sensed, the results are still within the acceptable range (the error is less than 10%).

#### 4.5. Skin Deformation Reconstruction and Elasticity Estimation

In the following experiments we have addressed the problem of objectively accessing skin abnormalities. Our method determines characteristics of the tissue by evaluating distortions of the skin which occur when the skin is pulled or stretched.

The experiments involve stretching of the skin on the ventral side of the forearm. Figures 21a and 21b show the gray level intensity images of the forearm skin before and after stretching, respectively. The corresponding range images are shown in Figs. 21c and 21d.

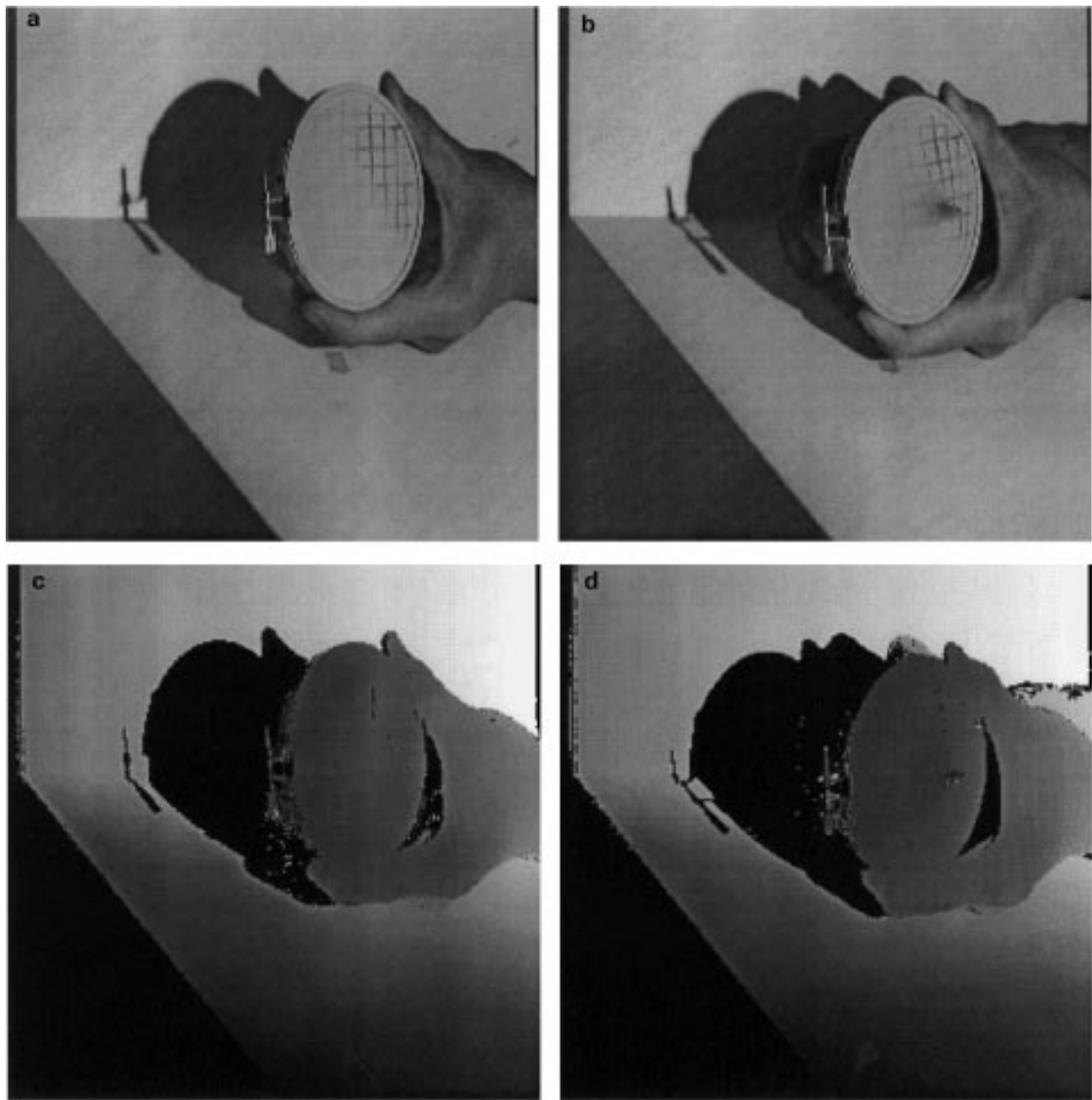
Skin exhibits a highly nonlinear stress-strain relationship and has been shown to be anisotropic. Since the loading in our experiments did not induce strains above the breaking point from stretchy to tough ( $\approx 8\%$ ), a material model that only captures the stretchy response is appropriate. To be effective the solution requires an energy-based material model, hyperelastic shell elements, a large strain integration, and analysis.

In these experiments our model is *local*. We consider only part of the forearm with the grid painted on the skin. The grid gives us necessary information about the keypoints (points used to build a model's geometry) and the displacements. The program reads intensity images and allows the user to select intersection points on the grid. In these experiments the selection process was manual (ongoing work includes implementation of snakes to detect and track grid intersection points). Coordinates of these points are converted into 3-D coordinates using range information. Keypoint information goes directly into a FEM model. The program generates areas of the model that approximate forearm geometry. Similarly, from the images after the motion we compute model displacements. The rest of the model, i.e., material

properties and number of areas, is case-independent. All model building heuristics described in previous chapter are applicable. However, this time change in scale (from 81 to 243 elements) was primarily motivated by the accuracy requirements (Fig. 21d). Both grids used in multiscale method are finer than the keypoint grid.

Figure 22a shows the reconstructed surface as a mesh. Displacement fields of grid points are shown in Fig. 22b. Despite the nonlinear nature of the problem, the solution takes on average about 1 min (34 s on a SUN SPARC 10 with 32 MB of memory for a model at a coarse scale (Fig. 21d) and 76 s for a model at a fine scale (Fig. 22a)) because of the relatively small number of elements. The surface error is 5.74% which means that the skin in the grid area is *normal* and *uniform*. When the error exceeds the threshold for this skin type and location, it demonstrates the existence of some skin abnormality such as a *scar*. Locating areas where the error exceeds the threshold and changing the value of Young's modulus for these areas reduces the error. In the experiment with simulated scars this procedure reduced the error from 11.62 to 5.97%. These changes are consistent with the elastic properties of this scar. This procedure shows promise in estimating the progress of scar healing by comparing its elastic properties with the elastic properties of the normal skin.

Another (quantitative) way to locate the abnormality is to compute the strain level throughout the skin surface. Figure 23a shows strains in the normal skin area: relatively even distribution in the middle of the area. The legend column on the right shows (top to bottom): maximum displacement, minimum strain (displayed only if it is different from zero), and maximum strain. The abnormality is simulated by having a tape attached to the forearm (Fig. 23b). Variations in the skin strain levels also indicate the existence of the area with the different material properties (Fig. 23c and 23d). Arrows point to the actual abnormal areas (different in every experiment). In both cases the area with the lowest strain coincides with the true location of the abnormalities. Other minimum peaks are identified in the boundary regions. Hence, the observation is: skin



**FIG. 16.** (a) The intensity image of the stretching rubber membrane before motion. (b) The intensity image of the stretching rubber membrane after motion. (c) The range image of the stretching rubber membrane before motion. (d) The range image of the stretching rubber membrane after motion.

regions with low strains other than the boundary are predominantly abnormal. We are currently working with the patients. The method is able to objectively assess characteristics of the burn scars and surrounding tissues. One of the typical results corresponding to the actual burn scar (Fig. 23e) is shown in the Fig. 23f.

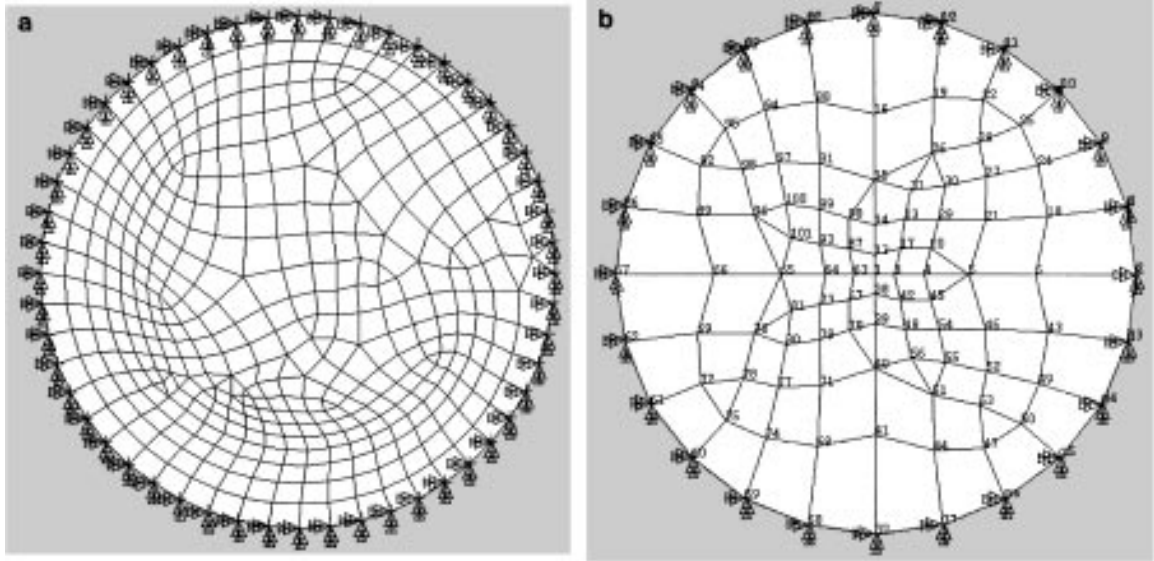
## 5. DISCUSSION AND CONCLUSIONS

### 5.1. Multiscale Method vs Mesh Refinement

Mesh refinement is an established recommended technique to increase the accuracy of FEM solution [34, 40]. To date, this is the first application of the multiscale method utilized in computer

vision to physically based modeling of nonrigid motion using FEM. Despite the fact that both methods deal with the meshing of the model, they are very different.

The multiscale method can be applied to more nonrigid motion cases since it needs fewer assumptions; i.e., we do not need to know the region of stress concentration. Since our algorithms differ from traditional FEM problems in mechanical engineering, we cannot use general mesh refinement unless we have sufficient *a priori* knowledge not only about the object (which is an assumption of this work) but also about the forces applied to it. In most engineering applications, the purpose of FEM analysis is to provide quantitative precision to an *a priori* qualitative understanding. Such *a priori* knowledge is used to help design a more efficient mesh.

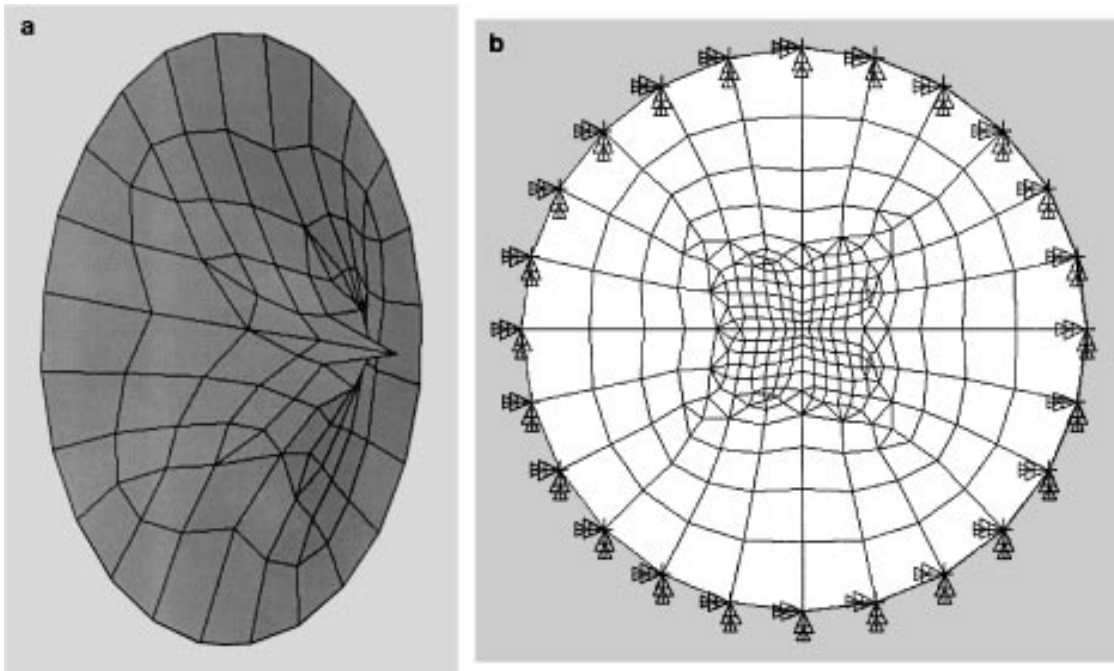


**FIG. 17.** (a) The initial node structure for rubber stretching modeling. (b) New node structure as the initial model for multiscale method.

Conversely, in computer vision applications we seldom have enough information to apply traditional methods. From the images before and after the motion we cannot tell what the general qualitative nature of the solution will be. This makes it impossible to locate small regions for the refinement in the bending plate experiment. This could be because the entire surface of the plate has almost the same need for refinement. The bending plate is more or less uniform, whereas the stretching of the rubber membrane is localized. With relative ease, we can deter-

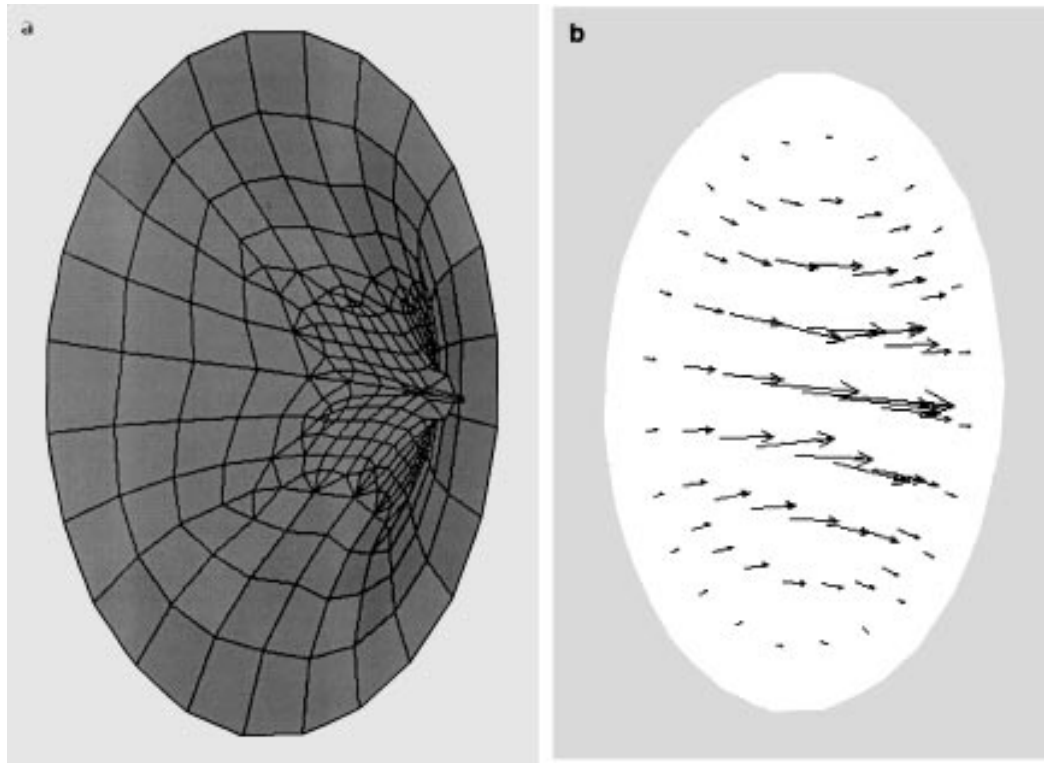
mine this region in the membrane experiment with our heuristic search for the largest local error. However, this case is more an exception than the rule. There are no requirements for additional knowledge in order to apply the proposed multiscale method.

The minimal number of the elements to reach a solution defines our first step. Then, we change the scale to look for an accurate solution in the local region determined during the first step.



**FIG. 18.** (a) The best result at a coarse scale processing. (b) New fine scale node structure.



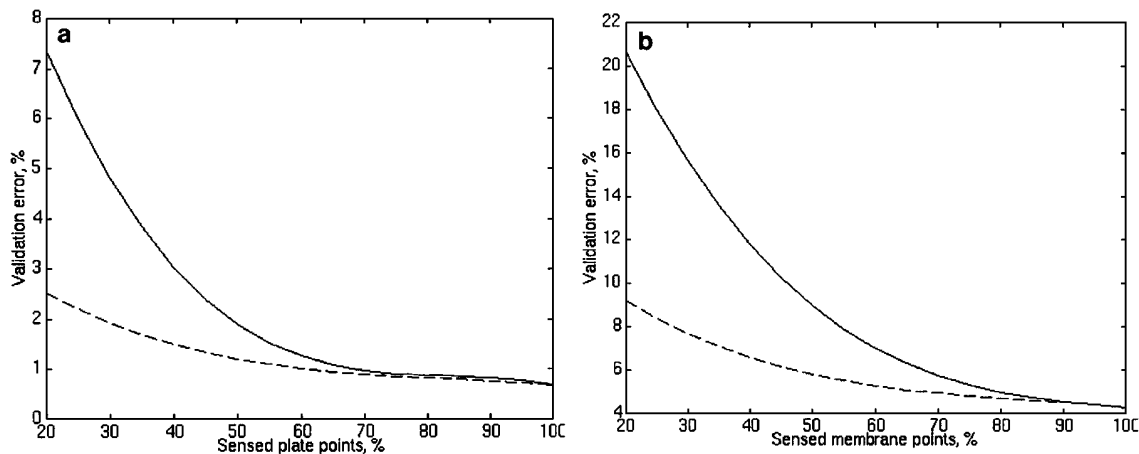


**FIG. 19.** (a) Reconstructed membrane by the multiscale method (at the fine scale). (b) Motion tracking of validation points.

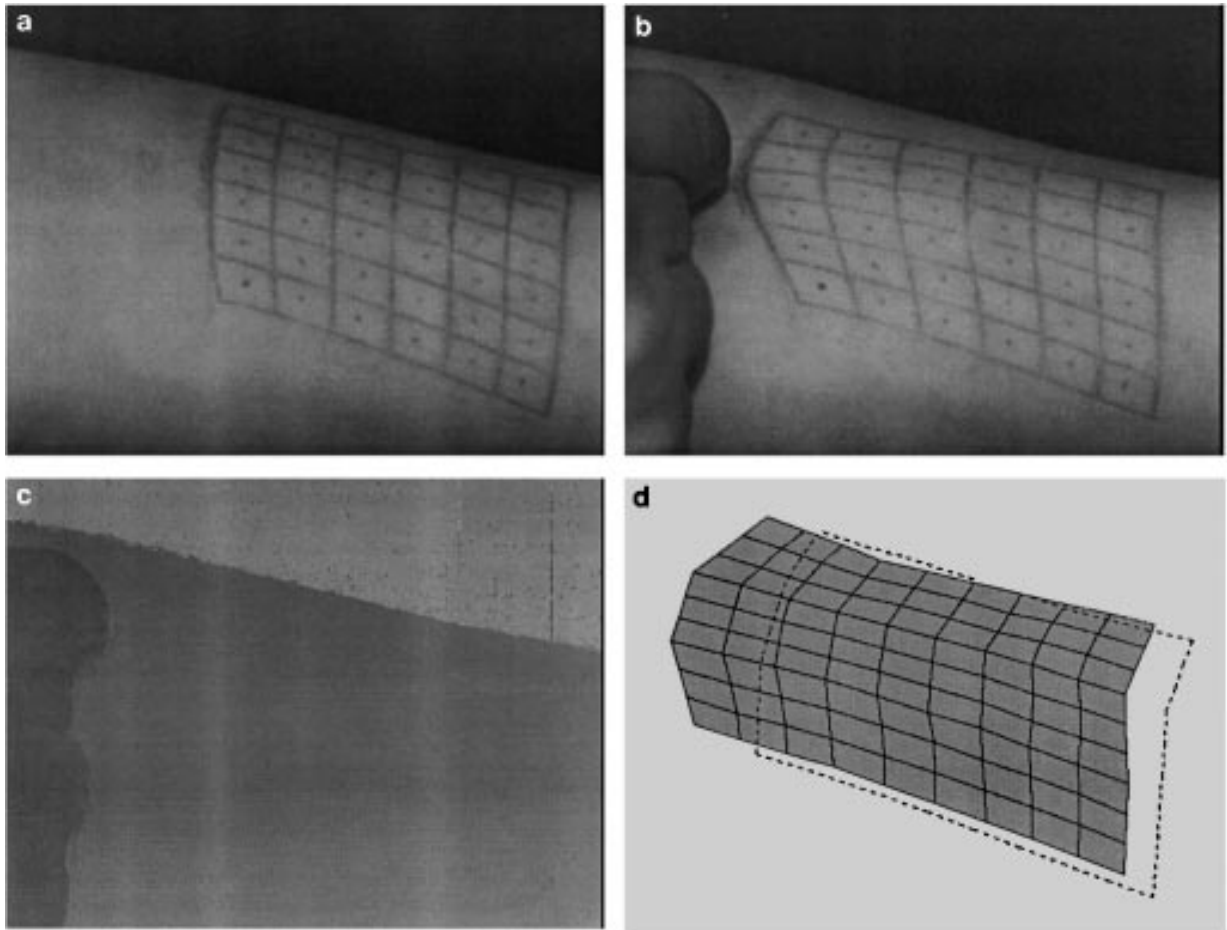
*5.2. Conclusions and Future Research*

In this work we have presented new methods for more efficient utilization and increased performance of nonlinear FEM for nonrigid motion analysis. Experimental results demonstrate the success of proposed algorithms. Some highlights include:

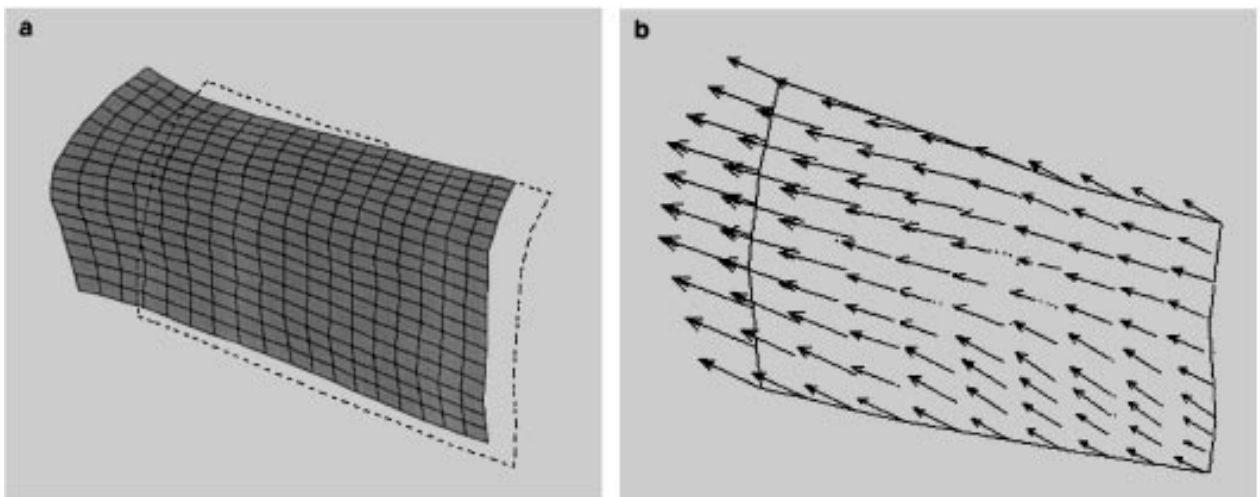
- Relatively stable, noise insensitive algorithm
- Significant speedup (over the direct approach)
- per hypothesis—better solution time (up to 30 times faster)
- per hypothesis—more effective mesh (3–9 times faster)
- smaller number of hypotheses (3–5 (or more) times faster)
- Dynamic performance of the model. For example, updating the number of nodes dynamically as the data (or our knowledge about it) changes
  - High quality of results in terms of both surface error and validation points error.



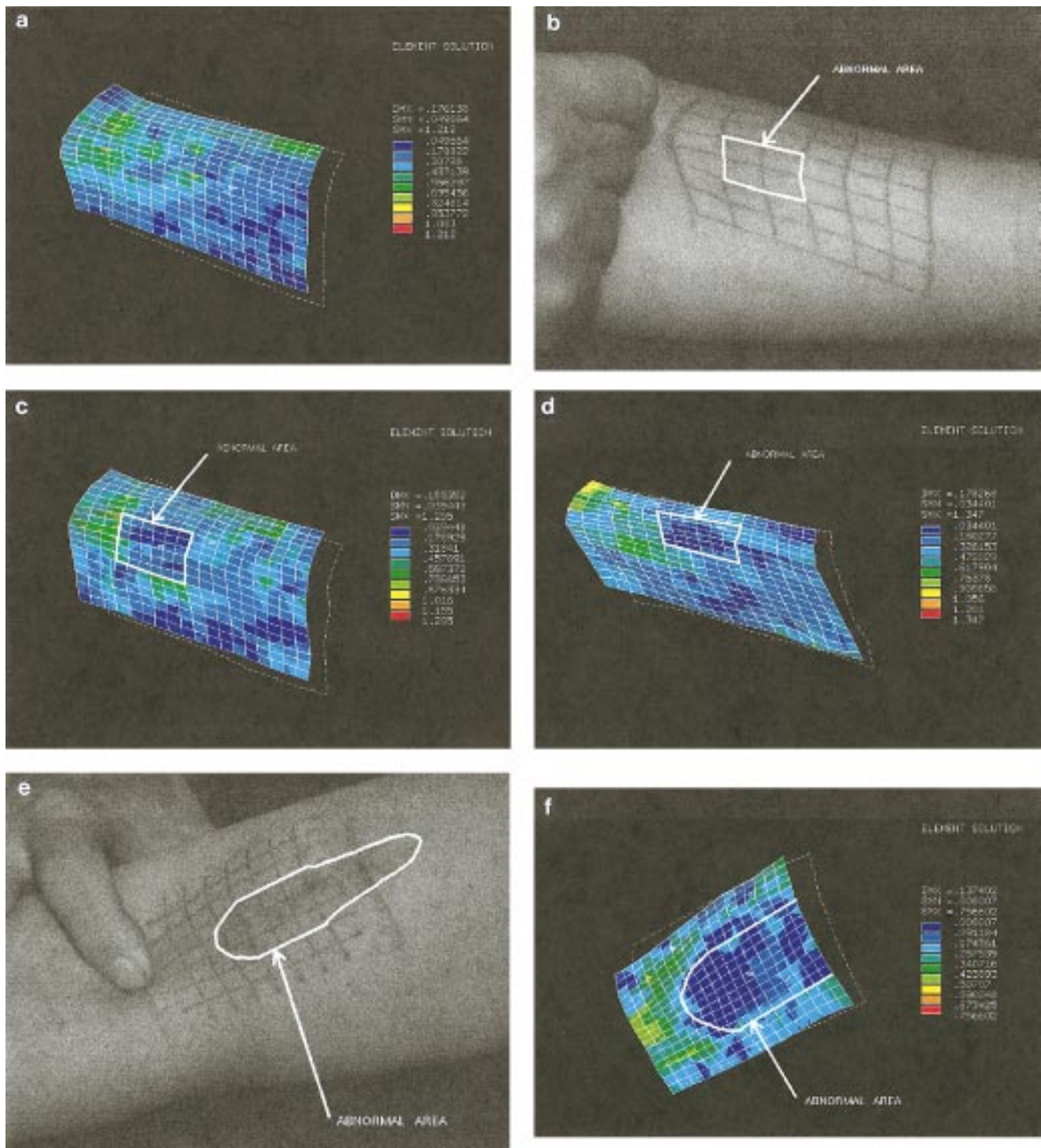
**FIG. 20.** (a) Results for the plate. (b) Results for the membrane.



**FIG. 21.** (a) The intensity image of the forearm skin before stretching. The grid painted on the skin is used for tracking; the dark dots are validation points. (b) The intensity image of the forearm skin after stretching. (c) The range image of the forearm skin after stretching. (d) Results at a coarse scale.



**FIG. 22.** (a) Reconstructed skin surface in the grid area. (b) Motion tracking of validation and grid points.



**FIG. 23.** (a) Strain distribution for the normal skin. (b) The intensity image of the forearm skin with the abnormality after stretching. (c–d) Strain distributions for the skin containing abnormal areas. (e) The intensity image of the skin (with the burn scar boundary outlined). (f) The corresponding strain distribution for the patient data (the scales are different).

We also demonstrate the success of our systematic approach to a practical application of nonlinear FEM for nonrigid motion analysis. Skin stretching experiments indicate promise in applying nonlinear FEM to the detection of abnormal skin areas based on the evaluation of their elastic properties relative to the surrounding areas. It will be utilized for quantitative assessment and comparison of burn scar treatment results. The method is still relatively computationally expensive for large problems. This can be addressed by the recent research [35] which reported that many parts of FEA are amenable to parallelization on a massive scale. All processors can work on the same problem simultaneously. These machines may extend the feasible problem size up into 10 million DOF range and make all of the described as well as any future strategies practical for everyday analysis of complex motion problems.

Nonlinear FEM provides the mechanism for fitting and tracking the visual data. It can handle both material nonlinearity and large deformation. It demonstrates the possibility of accurate computation of point correspondences in image sequences containing nonrigid shapes. The necessary assumptions include the knowledge of the material properties of the object (as precise as possible) and the availability of range data sensed before and after deformation.

Future work in this direction may include automatic matching with geometric primitives for complex objects and automatic tracking of grid lines in the skin stretching experiments.

The use of nonlinear FEM in the analysis of nonrigid motion is currently being extended by our research group to other biomedical applications like facial expression analysis, cardiac motion, and hand [41] and knee modeling.

## ACKNOWLEDGMENTS

We thank Dr. Bill Carpenter at the Civil Engineering Department for his valuable suggestions on ANSYS. Special thanks to Dr. Xiaoyi Jiang and Dr. Horst Bunke at the Institute of Informatics and Applied Mathematics, University of Bern, Switzerland for providing the range images of deforming materials.

## REFERENCES

1. D. Terzopoulos, A. Witkin, and M. Kass, Symmetry-seeking models and 3d object reconstruction, *Int. J. Comput. Vision*, 1987, 211–221.
2. Y. F. Wang and J. F. Wang, Surface reconstruction using deformable models with interior and boundary constraints, *Proceedings of ICCV, December 4–7, 1990*, pp. 300–303.
3. M. Kass, A. Witkin, and D. Terzopoulos, Snakes: active contour models, *Int. J. Comput. Vision* 1(4), 1988, 321–331.
4. D. Terzopoulos and K. Waters, Analysis of facial images using physical and anatomical models, *Proceedings of ICCV, 1990*, pp. 727–732.
5. D. Terzopoulos and K. Waters, Analysis and synthesis of facial images using physical and anatomical models, *IEEE Trans. Pattern Anal. Mach. Intell.* 14(6), 1993, 569–579.
6. F. Leymarie and M. D. Levine, Tracking deformable objects in the plane using an active contour model, *IEEE Trans. Pattern Anal. Mach. Intell.* 14(6), 1993, 617–634.
7. S. Kumar and D. B. Goldgof, Automatic tracking of SPAMM grid and the estimation parameters from cardiac MR images, *IEEE Trans. Medical Imaging* 13(1), 1994, 122–132.
8. L. D. Cohen and I. Cohen, Deformable models for 3d medical images using finite elements and balloons, *Proc. CVPR, 1992*, pp. 592–598.
9. T. McInerney and D. Terzopoulos, A finite element model for 3d shape reconstruction and nonrigid motion tracking, *Proc. of 12th ICPR, 1993*, pp. 518–523.
10. L. D. Cohen and I. Cohen, A finite element method applied to new active contour models and 3d reconstruction from cross sections, *Proceedings of ICCV, 1990*, pp. 587–591.
11. L. D. Cohen and I. Cohen, Finite-element methods for active contour models and balloons for 2-d and 3-d images, *IEEE Trans. Pattern Anal. Mach. Intell.* 15(11), 1993, 1131–1147.
12. F. L. Bookstein, Thin-plate splines and the atlas problem for biomedical images, *Information Processing in Medical Images, 12th International Conference, 1991*, pp. 326–342.
13. D. DeCarlo and D. Metaxas, Adaptive shape evolution using blending, in *Proc. Fifth International Conference on Computer Vision, Boston, MA, June 1995*, pp. 834–839.
14. W. Neuenschwander, P. Fua, G. Szekely, and O. Kubler, Deformable velcro surfaces, in *Proc. Fifth International Conference on Computer Vision, Boston, MA, June 1995*, pp. 828–833.
15. D. Terzopoulos and M. Vasilescu, Adaptive surface reconstruction, *Proceedings of SPIE, 1990*, Vol. 1383, No. 24.
16. D. Terzopoulos and M. Vasilescu, Sampling and reconstruction with adaptive meshes, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1991*, pp. 70–75.
17. M. Vasilescu and D. Terzopoulos, Adaptive meshes and shells, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1992*, pp. 829–832.
18. W. C. Huang and D. B. Goldgof, Analysis of intensity and range image sequences using adaptive-size meshes, *J. Vis. Commun. Image Representation* 4(4), 1993, 364–381.
19. W. C. Huang and D. B. Goldgof, Adaptive-size physically-based models for nonrigid motion analysis, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, June 15–18, 1992*, pp. 833–835.
20. W. C. Huang and D. B. Goldgof, Adaptive-size meshes for rigid and non-rigid shape analysis and synthesis, *IEEE Trans. Pattern Anal. Mach. Intell.* 15(6), 1993, 611–616.
21. H. Delingette, Adaptive and deformable models based on simplex meshes, *Proceedings of the 1994 IEEE Workshop on Motion of Non-Rigid and Articulated Objects, November 11–12, 1994*, pp. 152–157.
22. A. Pentland and S. Sclaroff, Closed-form solutions for physically based shape modeling and recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 12(7), 1991, 715–729.
23. A. Pentland, Fast surface estimation using wavelet bases, Technical Report 142, M.I.T. Media Lab Vision and Modeling Group, June 1990.
24. A. Pentland and B. Horowitz, Recovery of nonrigid motion and structure, *IEEE Trans. Pattern Anal. Mach. Intell.* 12(7), 1991, 730–742.
25. S. Sclaroff and A. Pentland, A modal framework for correspondence and description, *Proc. of 12th ICPR, 1993*, pp. 308–313.
26. S. Sclaroff and A. P. Pentland, Physically-based combinations of views: Representing rigid and nonrigid motion, in *Proc. of the 1994 IEEE Workshop on Motion of Non-Rigid and Articulated Objects, Austin, TX, November 11–12, 1994*, pp. 158–164. IEEE, Press, New York.
27. A. Young and L. Axel, Non-rigid wall motion using mr tagging, in *Proc. of Computer Vision and Pattern Recognition, Champaign, IL, June 1992*, pp. 399–404.

28. D. Metaxas and E. Koh, Efficient shape representation using deformable models with locally adaptive finite elements, in *Geometric Methods in Computer Vision II, San Diego, CA, July 1993*, SPIE Vol. 2031, pp. 160–171.
29. J.-P. Gourret, N. M. Thalmann, and D. Thalmann, Simulation of object and human skin deformations in a grasping task, *Comput. Graphics* **23**(3), 1989, 21–30.
30. C. Nastar and N. Ayache, A physically based analysis of deformations in 3d images, in *Geometric Methods in Computer Vision II, San Diego, California, 11–16 July 1993*, SPIE Vol. 2031, pp. 182–192.
31. J. M. Guccione and A. D. McCulloch, Finite element modeling of ventricular mechanics, in *Theory of Heart* (L. Glass, P. Hunter, and A. McCulloch, Eds.), pp. 121–144. Springer-Verlag, New York, 1991.
32. W.-C. Huang and D. B. Goldgof, Point correspondence recovery in non-rigid motion using nonlinear finite element modeling, in *Proceedings of Asian Conference on Computer Vision, Osaka, Japan, 23–25 November 1993*, pp. 256–259.
33. W.-C. Huang, D. B. Goldgof, and L. Tsap, Nonlinear finite element methods for nonrigid motion analysis, in *Proc. IEEE Workshop on Physics-Based Modeling in Computer Vision, Cambridge, MA, June 1995*, pp. 85–91.
34. D. S. Burnett, *Finite Element Analysis*, Addison-Wesley, Reading, MA, 1988.
35. J. R. Brauer, *What Every Engineer Should Know about Finite Element Analysis*, Marcel Dekker, New York, 1993.
36. Swanson Analysis System, *ANSYS User's Manual for Revision 5.0*, Swanson Analysis System, Inc., Houston, PA, 1993.
37. W. C. Huang, *Physically-Based Modeling in Nonrigid Motion Analysis*, Ph.D. thesis, University of South Florida, 1994.
38. V. S. Nalwa, *A Guided Tour of Computer Vision*, Addison-Wesley Reading, MA, 1993.
39. T. G. stahs and F. M. Wahl, Fast and robust range data acquisition in a low-cost environment, in *Proceedings of ISPRS-Conference, Zurich, 1990*, SPIE Vol. 1395, pp. 496–503.
40. O. C. Zienkiewicz, J. Z. Zhu, Y. C. Liu, K. Morgan, and J. Peraire, Error estimates and adaptivity from elasticity to high speed compressible flow, in *The mathematics of Finite Elements and Applications VI* (J. R. Whiteman, Ed.), pp. 473–511. Academic Press Limited, London, 1988.
41. L. V. Tsap, D. B. Goldgof, and S. Sarkar, Human skin and hand motion analysis from range image sequences using nonlinear fem, in *IEEE Nonrigid and Articulated Motion Workshop, San Juan, Puerto Rico, June 1997*, pp. 80–89.