

NON-RIGID REGISTRATION AND CORRESPONDENCE FINDING IN MEDICAL IMAGE ANALYSIS USING MULTIPLE-LAYER FLEXIBLE MESH TEMPLATE MATCHING

JIANHUA YAO

*Diagnostic Radiology Department, Clinical Center,
National Institutes of Health, Bethesda, MD 20892, USA
jyao@cc.nih.gov*

RUSSELL TAYLOR

*Computer Science Department,
The Johns Hopkins University, Baltimore, MD 21210, USA*

In this paper we present a novel technique for non-rigid medical image registration and correspondence finding based on a multiple-layer flexible mesh template matching technique. A statistical anatomical model is built in the form of a tetrahedral mesh, which incorporates both shape and density properties of the anatomical structure. After the affine transformation and global deformation of the model are computed by optimizing an energy function, a multiple-layer flexible mesh template matching is applied to find the vertex correspondence and achieve local deformation. The multiple-layer structure of the template can be used to describe different scale of anatomical features; furthermore, the template matching is flexible which makes the correspondence finding robust. A leave-one-out validation has been conducted to demonstrate the effectiveness and accuracy of our method.

Keywords: Non-rigid registration; statistical model; multiple-layer flexible mesh template; correspondence.

1. Background and Introduction

Non-rigid medical image registration is an essential step in many automated medical image analysis and has been widely investigated in recent years.^{3,5,7,10,11,15–17,19,23,33} Medical image registration can be categorized based on the subjects involved in the registration. One category is related to registration between two images, such as that between images from different modalities, images from different individuals, and images taken at different times.^{3,12,21,22} This category of registration usually relates the information in one image to information in another image by determining the one-to-one pixel correspondences. Another category of registration is between an anatomical model/atlas and a medical image.^{4,5,8–10,15–17,23,27,30,31} In this category, the anatomical model or

atlas is transformed and deformed to match the anatomic structure in the image. The ability to register an anatomical model to individual patient images provides the basis for solving several important problems in medical image interpretation. Once the model is registered to a particular image, structures of interest can be labeled and extracted for further analysis, and knowledge incorporated in the model can be transferred to the patient. The result of the registered model generates segmentation for the structure of interest, which can then be used for image analysis purposes such as measurement and visualization. Model-to-image registration also allows population studies to be analyzed in a common frame of reference.

In the model-to-image non-rigid registration, the correspondences between features on the model and those on the image need to be correctly established in order to obtain the desired transformation and deformation. Some methods are based on anatomical features (landmarks) such as points, curves and surfaces.^{1,2,8,10,14,23,25} Other methods depend on the intensity information in the image.^{3,5,24,29,32} Hybrid algorithms have also been proposed in which both geometric features and image intensities are used.^{11,15,19} In landmark-based methods, geometric features such as points, curves or surfaces are brought into alignment. The landmark points can be either intrinsic or extrinsic. Intrinsic points are derived from naturally occurring features, e.g. anatomic landmark points. Extrinsic points are derived from artificially applied markers or fiducials fixed to the patient and visible in both images and models. In surface-based methods, the geometric features (such as ridge curves) on the surface are used to establish the correspondence.²⁵ After the correspondences between landmarks are established, the image and/or model are warped to align the corresponding landmarks. Intensity-based registration stems from the observation that although images from different modalities or different subjects exhibit different data values, there is usually a large amount of shared information between images of the same structures. The correspondence between the model and the image is established based on the voxel intensity distribution.^{3,5,24,29,32}

An anatomical model is a tool to represent human anatomical structures and normal anatomical variability. Medical images are usually complex, noisy and possibly incomplete, which make their interpretation very difficult without prior knowledge of the anatomy. Therefore, many researchers have turned to prior statistical models for assistance in medical image registration. Shen *et al.*²³ proposed a statistical surface model using an affine invariant geometric attribute vector to find the vertex correspondences. Chen *et al.*³ built an average brain atlas based on statistical analysis of voxel intensity values. Cootes *et al.*⁵ proposed an Active Appearance Model (AAM) that incorporates both the shape variability and density variability. The AAM models were experimented on 2D MRI brain slices and 2D human face images. Cootes models were extended to 3D surface models by Fleute *et al.*¹⁰

Most existing model-to-image registration methods used either a surface model or an intensity grid model^{4,10,17,23,26,27} Hence the exterior shape and the internal density distribution cannot be matched at the same time. To address this

problem, our method is based on a statistical volumetric model incorporating both shape and density properties of the anatomical structure. As mentioned in some statistical model based registration methods,^{5,23} the prior model extracted from training models is not sufficient to characterize all the variations in a new image. To compensate this, a multiple-layer flexible mesh template matching technique is developed to find the feature correspondences and achieve local deformation. This technique takes advantage of our model topological structure (tetrahedral mesh) and properties incorporated in the model (shape and density).

The remainder of this paper is organized as follows. Section 2 briefly introduces the statistical bone density model and its construction from a set of training images. Section 3 presents our method for non-rigid registration between the statistical model and a CT image. Section 4 proposes a multiple-layer flexible mesh template matching method for correspondence finding and local deformation of the model. Finally, Secs. 5 and 6 conclude with validation experiments and discussions.

2. Statistical Bone Density Model

We proposed a unique model representation to characterize both the boundary surface and internal density distribution of the bone structures. The model is represented as a tetrahedral mesh equipped with embedded Bernstein polynomial density functions on the barycentric coordinates of each tetrahedron. Multiple level-of-details of the anatomical structure are characterized by a hierarchical representation. And prior information of both shape properties and density properties is incorporated in the model.

We developed an efficient tetrahedral mesh reconstruction from contours method to construct tetrahedral meshes for bone structures from a CT image. The method produces tetrahedral meshes with high flexibility and is able to accommodate any anatomical shape. The meshes are built from contours consistent with the cortical bone boundaries. The contours are extracted from the image slice-by-slice using an active contour technique.¹³ Then the contours are tiled into a tetrahedral mesh by solving a series of tiling, corresponding and branching problems.¹⁸ An analytical density function is assigned for every tetrahedron to minimize the residual errors in the density distribution. Currently, the density functions are written as n -degree Bernstein polynomials in barycentric coordinates of a tetrahedron:

$$D(\mu) = \sum_{i+j+k+l=n} C_{i,j,k,l} B_{i,j,k,l}^n(\mu) \tag{1}$$

where $C_{i,j,k,l}$ is the polynomial coefficient, and $\mu = (\mu_x, \mu_y, \mu_z, \mu_w)$.

$$B_{i,j,k,l}^n(\mu) = \frac{n!}{i!j!k!l!} \mu_x^i \mu_y^j \mu_z^k \mu_w^l \tag{2}$$

is a barycentric Bernstein basis function, and $(\mu_x, \mu_y, \mu_z, \mu_w)$ are the barycentric coordinates, with $\mu_x + \mu_y + \mu_z + \mu_w = 1$. The advantages of such a representation are:

(1) it is in an explicit form; and (2) it is a continuous function in 3D space. Therefore, it is convenient to integrate, to differentiate, and to interpolate. We also developed a tetrahedral mesh simplification algorithm based on edge collapsing operations²⁸ to build a multiple level-of-detail (LOD) model representation.

We designed a training strategy to compute a statistical model from a collection of training models. A model aligning procedure is first performed to map all training models into a common mesh topological structure. Then the Principal Component Analysis (PCA) method is applied to compute the variability of both shape properties and density properties of the anatomical structure. Using the PCA method, the model can be approximated by a set of statistical mode parameters $\{b_i\}$:

$$Y = M(\bar{Y}, b) = \bar{Y} + Pb \tag{3}$$

where Y is a model instance, $M(\bar{Y}, b)$ is the instantiation operation, \bar{Y} is the average model representation, and P is the eigenvector matrix incorporating the prior information.

We have built a statistical density model for hemi-pelvis from eight training images. Figure 1 shows the exterior surface of a multiple-resolution hemi-pelvis model. Figure 2(a) shows the shape variation and Fig. 2(b) shows the density variation of the model by projecting the model to a 2D plane using a ray casting technique. Figure 2 demonstrates that a new model instance can be instantiated from the average model by setting the statistical mode parameters. Detailed description of our model and reconstruction method can be found in Refs. 34 and 36.



Fig. 1. Multiple resolution pelvis model.

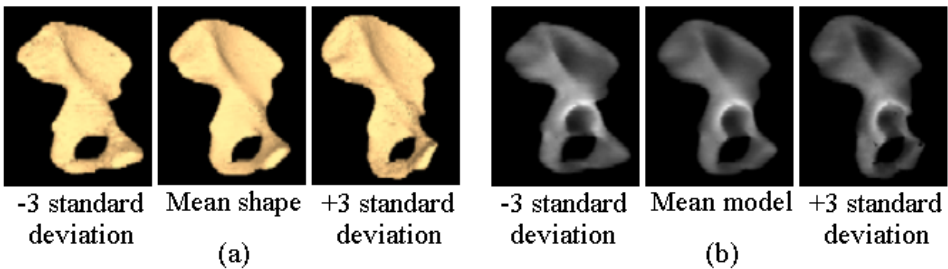


Fig. 2. Shape and density variation of the model.

3. Non-Rigid Registration Scheme

Non-rigid registration between our statistical bone density model and CT images is one of the purposes in our investigation. Our method utilizes both shape properties and density properties of the model, together with the statistical information extracted from a population of training models. The result from the registration can be used as a model-based segmentation of the bone structure.

The registration process is divided into three stages: affine transformation global deformation and local deformation. The result of preceding stage is used as initial values for the next stage. In the affine transformation stage, the translation, rotation and scale of the model are optimized to determine the location and orientation of the model. In the global deformation stage, the statistical mode parameters are optimized to match the model with the anatomical structure in the images. Due to the limited number of training models in the model training stage, the prior information in the statistical model does not include all the variability inherent in the anatomy. To compensate for this, a local deformation step is taken to build correspondences between vertices on the model and local features in the image and adaptively warp the model. The registration process is governed by a script file and is highly automated.

3.1. Optimization algorithm and energy function

Both the affine transformation stage and the global deformation stage are essentially multi-dimensional nonlinear optimization problems designed to minimize an energy function between the anatomical model and the CT image.

An energy function is defined as the objective function to evaluate the difference between the model and the CT image, written as:

$$E(mdl, img) = w_s E^{(s)}(mdl, img) + w_d E^{(d)}(mdl, img) \quad (4)$$

$$E^{(s)}(mdl, img) = \sum_{i=1}^{N(v)} (\mathbf{g}^{(mdl)}(v_i) \cdot \mathbf{g}^{(img)}(v_i)) \quad (5)$$

$$E^{(d)}(mdl, img) = \sum_{i=1}^{N(t)} \left(\oint_{\mu} \left(\left(\frac{d^{(mdl)}(t_i, \mu) - d^{(img)}(t_i, \mu)}{d^{(mdl)}(t_i, \mu)} \right)^2 \right) \right) \quad (6)$$

where mdl represents the model; img represents the image. The energy function consists of two parts: $E^{(s)}$ measures the shape difference; $E^{(d)}$ measures the density difference. v_i is a vertex on the model; $\mathbf{g}^{(mdl)}(v_i)$ is the surface normal at v_i on the model; $\mathbf{g}^{(img)}(v_i)$ is the image intensity gradient direction at location v_i . The surface normal of v_i is computed by fitting a quadratic surface of the neighboring vertices of $v_i \dots d^{(mdl)}(t_j, \mu)$ is the density value at a voxel μ within tetrahedron t_j in the model; $d^{(img)}(t_j, \mu)$ is the density value at the corresponding voxel coordinate (t_j, μ) in the image. $E^{(s)}$ is the sum of the dot product of $\mathbf{g}^{(mdl)}(v_i)$ and $\mathbf{g}^{(img)}(v_i)$ over

all vertices on the model; $N(v)$ is the total number of vertices in the model. $E^{(d)}$ is computed by integrating the density difference over all tetrahedra using the local barycentric coordinates of each tetrahedron; $N(t)$ is the total number of tetrahedra in the model. $E^{(s)}$ and $E^{(d)}$ are assigned weights w_s and w_d respectively, and $w_s + w_d = 1$. w_s and w_d are determined according to the images and applications. For instance, w_s should be large in images where edges are more prominent (such as images of bony structures). While for images where density distributions are more important (such as images of brains and soft tissues), $E^{(d)}$ has a larger weight w_d . In our pelvis registration problem, $w_s = 0.7$ and $w_d = 0.3$ were used.

Some optimization algorithms,²⁰ such as gradient descent methods, require computing derivatives of the energy function. In our method, the derivatives of the energy function with respect to both the affine transformation parameters and the statistical mode parameters are very difficult to compute. Therefore, Powell's method²⁰ was chosen as our optimization algorithm since it does not require the derivatives of the objective function.

To further improve the efficiency and robustness of the algorithm, the process is executed in a multiple-resolution framework. This involves first searching for the match in a coarser image using the lower resolution model, and then refining the solution in a series of higher resolution images and models. The multiple-resolution image space is implemented using a Gaussian image pyramid described in Ref. 6. The image on lower resolution level is formed by smoothing the image on its previous higher level, followed by subsampling to obtain an image with half the number of pixels in each dimension. The multiple-resolution model space is created based on edge collapsing techniques, where the number of vertices on models on subsequent levels is approximately halved. The algorithm is also implemented in a multiple step size manner, in which it starts with a large step size and gradually reduces the step size as getting closer to the optimal solution. The multiple-resolution and multiple step size scheme leads to a faster optimization process, and also makes the process less likely to fall into a local minimum.

3.2. Affine transformation

An affine transformation includes translation $T(t_x, t_y, t_z)$, rotation $R(r_x, r_y, r_z)$, and scale $S(s_x, s_y, s_z)$. The affine transformation process is a *nine*-dimensional nonlinear optimization problem, which can be written as:

$$\arg \min_{R,S,T} (E(R \cdot S \cdot mdl + T, img)). \quad (7)$$

Here $E(\cdot)$ is the energy function defined in Eq. (4); R , S and T are translation, rotation and scale, respectively. The nine parameters in the affine transformation are divided into three subsets — translation, rotation, and scale; during each pass, only one subset of the parameters is optimized. The parameters are optimized back and forth several times to get the optimal parameters that minimize the energy function.

After the optimized affine transformation is achieved, the location, orientation and size of the model should roughly match with the anatomical structure.

3.3. Global deformation

As mentioned in Sec. 2, given a set of statistical mode parameters $b = \{b_i\}$, an instance of the model can be instantiated. The new instance Y of the model is equivalent to a warped version of the average model (see Fig. 2). The operation of changing the statistical mode parameters to obtain a new model instance is referred to as the global deformation of the model. The global deformation stage can also be treated as an optimization procedure in the statistical mode parameter space to minimize an energy function between the model and the image. For each statistical mode parameter being evaluated, an instance of the model is generated. This hypothesis is then compared with the image using the energy function defined in Eq. (4). As in the affine transformation stage, Powell's method is adopted to optimize the statistical mode parameter set. The optimization problem can be written as:

$$\arg \min_b (E(M(mdl, b), img)). \quad (8)$$

Here, $b = \{b_i\}$ is the statistical mode parameter of the anatomical model, $E(\cdot)$ is the energy function defined in Eq. (4), and $M(\cdot)$ [Eq. (3)] is the instantiation operation to apply the statistical mode parameters to the average model as defined. The set of parameters that minimizes the energy function is used to instantiate the registered model.

In our experiment on the hemi-pelvis model, first five most significant statistical modes are optimized. Our experiment showed that these five statistical modes can cover about 93% of variability in the training set. The global deformation brings the model very close to the anatomical structure in the image, but some discrepancies still exist due to the limited variability incorporated in the model. The global deformation stage provides an initial value and start point for further local deformation and correspondence finding.

4. Correspondence and Local Deformation

Human anatomical structures vary significantly among individuals, but the statistical model only characterizes the variability exhibited in the training set. Given a specific image which is not in the training set, there always exist variations that cannot be reproduced from the statistical model through model instantiation. Therefore, a local deformation stage is necessary after the global deformation stage to address small discrepancies. The purpose of the local deformation stage is to locally adjust the location of each vertex on the model to match local images features. We proposed a multiple-layer flexible mesh template matching method to first find the correspondences between model vertices and image features, and then perform the local deformation of the model.

<p><u>Input:</u> A model M before local deformation, and an image I</p> <p><u>Output:</u> A model M^* after local deformation</p> <p><u>Step 1:</u> Initialization</p> <p><u>Step 2:</u> for every vertex v_i on model M</p> <p style="padding-left: 2em;">2.1) a multiple-layer flexible mesh template $T(v_i)$ from model M is constructed (Section 4.1)</p> <p style="padding-left: 2em;">2.2) the attribute vector $A^{(m)}(v_i)$ for $T(v_i)$ is computed (Section 4.2)</p> <p style="padding-left: 2em;">2.3) the corresponding image coordinate c_i for v_i is located within a searching range (Section 4.3).</p> <p style="padding-left: 2em;">end for</p> <p><u>Step 3:</u> $\{v_i\}$ are adaptively deformed to corresponding image coordinate $\{c_i\}$. (Section 4.4)</p> <p><u>Step 4:</u> <u>Step 2</u> and <u>Step 3</u> are repeated until the difference between models in two iterations is below a threshold or the maximum number of iterations is reached.</p>

Fig. 3. Pseudo-code of local deformation and correspondence procedure.

Figure 3 provides the pseudo-code of the correspondence finding and local deformation procedure. In the procedure, a multiple-layer flexible mesh template is first constructed for each vertex on the model, and the attribute vector for each template is computed. Then the corresponding image coordinate for each vertex is located using flexible mesh template matching techniques. Finally, the vertices are adaptively deformed to their corresponding image coordinates. The above process is iterated several times until convergence or the maximum number of iterations is reached.

4.1. Multiple-layer flexible mesh template

Due to the complexity in human anatomical structures, it is not easy to find the correspondence between vertices on the model and image voxel locations. Shen and Davatzikos *et al.*⁶ presented an affine invariant attribute vector using the volumes of tetrahedra formed by the neighbors of vertices on the surface mesh to find the vertex correspondences. Inspired by their work, we proposed a robust template matching method to correlate each vertex on the model to its corresponding voxel location.

The tetrahedral mesh topological structure of the model is taken advantage of to construct a *multiple-layer flexible mesh template* for each vertex in the model. For each vertex on the model, its topological neighbors on the tetrahedral mesh and itself naturally form a mesh template centered at the vertex. The mesh template is retrieved directly from the model on the fly, so no extra storage is needed to keep the template. Figure 4 shows a flexible mesh template in a 2D case, which illustrates a two-layer mesh template with 13 nodes. The vertices and edges in the figure are vertices and edges in the tetrahedral mesh.

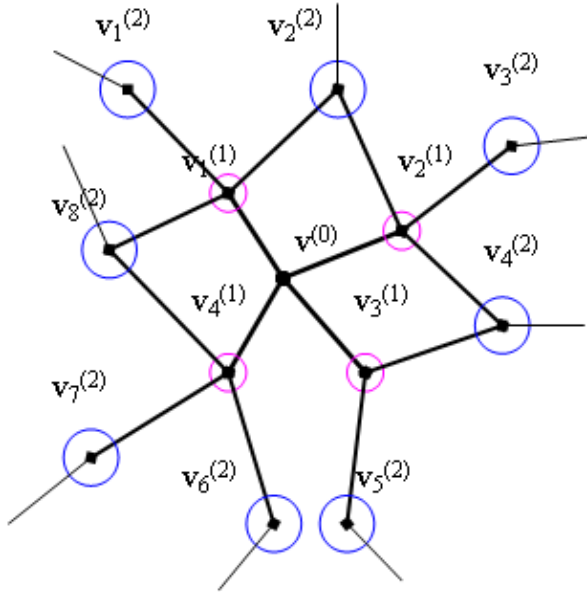


Fig. 4. Multiple-layer flexible mesh template.

The structure of the template centered at a vertex $v^{(0)}$ consists of an array of nodes $\{v_i^{(k)}\}$. For a node $v_i^{(k)}$, k is the topological distance between the node and $v^{(0)}$ in the tetrahedral mesh, k is also referred to as the layer number of node $v_i^{(k)}$, and i is the index number of the node in layer to k . Several attributes are assigned to a node $v_i^{(k)}$: (1) the relative position to the center, $p_i^{(k)} = v_i^{(k)} - v^{(0)}$; (2) a searching sphere $r_i^{(k)}$; (3) an attribute value and a weight.

The template associated with a vertex $v^{(0)}$ has multiple layers. $v^{(0)}$ is the center of the template, $\{v_i^{(1)}\}$ are nodes on the first layer neighbors of $v^{(0)}$, and $\{v_i^{(2)}\}$ are on the second layer neighbors of $v^{(0)}$, and so on. The more layers a template has, the larger and more global structure it is able to characterize. Conversely, the fewer layers in a template, the smaller and more local structure it describes. If too many layers are used, the structure may become too big to have any correspondence in a local neighborhood. On the other hand, if too few layers are used, many correspondences may be available and it is difficult to single out the correct correspondence. The example shown in Fig. 4 has two layers. The first layer has four nodes $\{v_1^{(1)}, v_2^{(1)}, v_3^{(1)}, v_4^{(1)}\}$, and the second layer has eight nodes $\{v_1^{(2)}, \dots, v_8^{(2)}\}$. In most cases, a two-layer mesh template is adequate for finding local correspondences.

The template is also flexible. This means the location of a node on the template is not fixed, instead it can be any location within a searching sphere $r_i^{(k)}$ associated with the node. The circles around each node in Fig. 4 represent the searching spheres. In the flexible template matching, the nodes on the template move within their searching spheres to find the best match. Due to the local deformation, the mesh template may not have exact matched features in the image. The searching

sphere makes the template flexible and able to match with similar features within a reasonable range. The center of the template has a searching sphere with zero radius, which means the center node of the template is the only node on the template with fixed location. The nodes farther from the template center have larger searching spheres. Currently in our implementation, the radius of the searching sphere is defined as 1/10 of the distance from a node to the center node of the template. This is a value based on experience, which may vary in different applications.

The mesh template has the following advantages over an ordinary fixed-size matrix template: (1) it is retrieved directly from the tetrahedral mesh topology; (2) the multiple-layer structure can be used to describe different scales of anatomical features; and (3) the flexible searching sphere makes the template matching robust to correlate similar features.

4.2. Template attribute vector and dynamic image attribute vector

An attribute vector is assigned to every mesh template. The attribute vector describes both the density properties and shape properties of the mesh template. Each node on the template is associated with an attribute value $(d; \mathbf{g})$ and a weight w , where d is the density value of the node and \mathbf{g} is the gradient direction at the node location. The attribute value of a node $v_i^{(k)}$ is directly computed from the density model. The density value d is computed from the Bernstein polynomial density function assigned to each tetrahedron (Sec. 2). The gradient orientation is computed from mesh layer surface. The template attribute vector is the concatenation of the attribute values of all nodes on the template, and can be written as

$$A^{(m)} = (w_0^{(0)} d_0^{(0)}, w_1^{(1)} d_1^{(1)}, w_2^{(1)} d_2^{(1)} \dots; w_0^{(0)} \mathbf{g}_0^{(0)}, w_1^{(1)} \mathbf{g}_1^{(1)}, w_2^{(1)} \mathbf{g}_2^{(1)} \dots) \quad (9)$$

here $(w_0^{(0)}, d_0^{(0)}, \mathbf{g}_0^{(0)})$ are the weight and attribute values of the center node of the template, and $(w_i^{(k)}, d_i^{(k)}, \mathbf{g}_i^{(k)})$ are the weight and attribute values of node i on layer k , i.e. node $v_i^{(k)}$. A template attribute vector $A^{(m)}$ is computed for every vertex on the model.

In the template matching and correspondence finding process (details in Sec. 4.3), the mesh template of a vertex v is moved in the image field and the best matched voxel location for vertex v is located. For any voxel location c in the image space, an attribute vector $A^{(c)}$ is defined for the given mesh template of vertex v . The image attribute vector has the same format as the template attribute vector $A^{(m)}$ of v , but the attribute values of nodes are obtained from corresponding voxel locations. Because a node $v_i^{(k)}$ on the mesh template is associated with a searching sphere $r_i^{(k)}$, when retrieving the attribute value at a voxel location, the region within the searching sphere $r_i^{(k)}$ of node $v_i^{(k)}$ is searched for the most alike attribute value. The attribute vector of a voxel is called the dynamic image attribute vector since it varies for different mesh templates. The goal of template matching is to find the corresponding voxel location for each vertex, where the template attribute vector $A^{(m)}$ of the vertex and the dynamic image attribute vector $A^{(c)}$ are best matched.

4.3. Flexible mesh template matching and correspondence finding

A template matching method is employed to find the vertex correspondence because of its simplicity and computational efficiency. Template matching is a simple filtering method to detect a particular shape or object in an image. An object can be detected if its appearance is known accurately in terms of a template. In our application, we want to detect the anatomical structure defined by the mesh structure associated with each vertex on the model. The idea of rigid template matching is straightforward. The template is moved in the image field, and the difference between the template and the image is calculated at each voxel location. An optimal match is reported at the location where the minimum difference between the template and the image is reached. A shortcoming of the rigid template matching is that it requires the template be very precise, so it is sensitive to shape and density variation. We proposed a flexible mesh template matching to overcome this problem. Each node on the template is associated with a searching sphere $r_i^{(k)}$. And the search for best matched voxel location for each node is conducted within its searching sphere. Essentially in flexible template matching, the process is broken down into two steps: first the best match for each node on the template is located within its searching sphere, and then the match of the entire template is computed using the best match of each node.

The goal of our template matching method is to find the corresponding voxel location in the image space for each vertex on the model, so that a vertex can be warped to its corresponding voxel location. Each vertex is associated with a mesh template and a template attribute vector $A^{(m)}$. The approach to find the corresponding image coordinate is to move the mesh template over the image space to find the most similar dynamic image attribute vector. Minimizing the difference between the template attribute vector and dynamic image attribute vectors, a best template matching can be found, and so the corresponding image coordinate for the vertex. The following equation is the way to compute the difference between two attribute vectors $A^{(m)}$ and $A^{(c)}$:

$$\begin{aligned} \text{Diff}(A^{(m)}, A^{(c)}) = & \frac{1}{\text{len}(A^{(m)})} \sum_{i=1}^{\text{len}(A^{(m)})} \left(w_i \left| \frac{d_i^{(m)} - d_i^{(c)}}{d_i^{(m)}} \right| \right) \\ & + \frac{1}{\text{len}(A^{(m)})} \sum_{i=1}^{\text{len}(A^{(m)})} \left(\frac{w_i(1 - \mathbf{g}_i^{(m)} \cdot \mathbf{g}_i^{(c)})}{2} \right) \end{aligned} \quad (10)$$

where $A^{(m)}$ is the attribute vector of the mesh template, and $A^{(c)}$ is the image attribute vector of a voxel location. It contains two parts, the density vector difference and the gradient vector difference, and each part has been normalized to range $(0 \dots 1)$. The density vector difference evaluates the average percentage difference of density values, and the gradient vector difference measures the average dot product of gradient directions. The two parts are weighed by their weights.

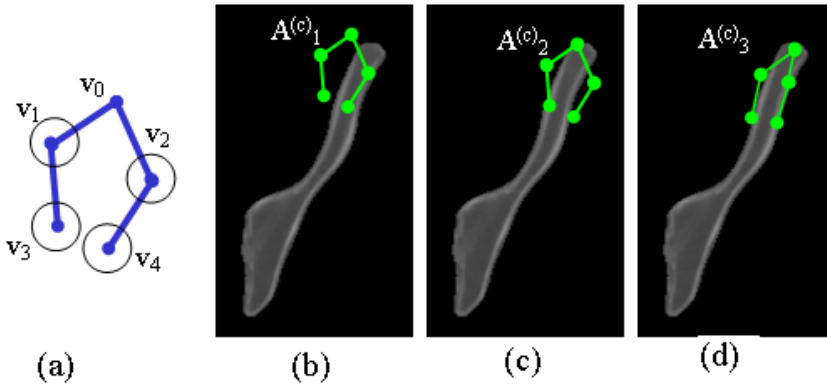


Fig. 5. Example of flexible template matching (see text for explanations).

In order to demonstrate the flexible mesh template matching, Fig. 5 shows a simple example in 2D space. The mesh template in Fig 5(a) is a simplified 2D two-layer mesh template with five nodes. v_0 is the center node of the mesh template; v_1 and v_2 are the first layer neighbors; v_3 and v_4 are the second layer neighbors. Its template attribute vector can be written as:

$$A^{(m)} = (w_0d_0, w_1d_1, w_2d_2, w_3d_3, w_4d_4; w_0\mathbf{g}_0, w_1\mathbf{g}_1, w_2\mathbf{g}_2, w_3\mathbf{g}_3, w_4\mathbf{g}_4). \quad (11)$$

By moving this mesh template over the image space, dynamic image attribute vectors are evaluated at different voxel locations, denoted as $A_1^{(c)}, A_2^{(c)}, A_3^{(c)}, \dots$. Figures 5(b)–5(d) demonstrate some dynamic image attribute vectors at different voxel locations. Figure 5(b) shows a totally unmatched voxel location. Figure 5(c) shows a location with partial matching by applying the rigid template matching. After searching for the best match of each node within its searching sphere, a perfect match is found in Fig. 5(d). The template difference between Figs. 5(c) and 5(d) (the difference is exaggerated for the purpose of visualization) demonstrates the idea of the flexible template matching. Hence, the voxel location of the image attribute vector $A_3^{(c)}$ in Fig. 5(d) is the corresponding voxel location of the center vertex v_0 of the mesh template. At the end, vertex v_0 can be locally and adaptively warped to its corresponding voxel location.

4.4. Adaptive deformation and constraints

If any deformability is allowed, one can always deform one object into any other objects (e.g. morphing a head into a teapot). Some constraints are necessary for local deformation, especially in anatomical structures. In order to make the local deformation smooth and keep the tetrahedral mesh structure valid, several strategies were adopted, including Gaussian morphing, adaptive deformation focus, and maximum deformation range. Among these techniques, adaptive focusing and Gaussian morphing are inspired by Shen’s work.²³

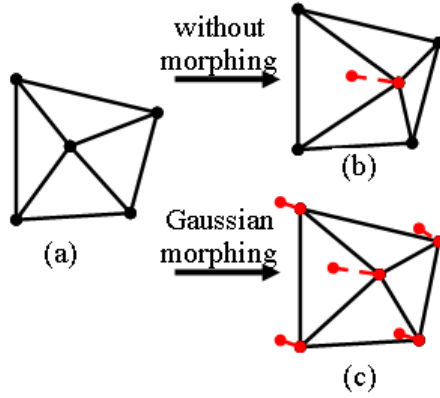


Fig. 6. Gaussian morphing.

To keep the deformation continuous and smooth, a Gaussian morphing strategy is adopted. In this strategy, when one vertex is moved, its neighbors will also be morphed accordingly. The following equation describes the Gaussian morphing operation,

$$\begin{aligned} \Delta v_0 &= v'_0 - v_0 \\ \Delta v_l &= \Delta v_0 \cdot e^{-\frac{l^2}{2\sigma^2}} \end{aligned} \tag{12}$$

where v_0 is the vertex location before deformation, v'_0 is the corresponding location obtained from the template matching, Δv_0 is the movement of the deformed vertex, Δv_l is the movement of its l th layer neighbors, σ is the morphing parameter. Figure 6 illustrates the effect of Gaussian morphing in 2D. Figure 6(a) shows a vertex and its neighbors. In Fig. 6(b), the vertex deforms without the morphing of its neighbors. In Fig. 6(c), the vertex deforms with Gaussian morphing of its neighbors. The transition from Figs. 6(a) to 6(c) is much more continuous and smoother than that from Figs. 6(a) to 6(b).

In addition, an adaptive deformation focus strategy is applied, i.e. the vertex with highest matched image attribute vector will deform first and drag its neighbors to morph. Then, the focus will move to the next highest matched vertex. The adaptive focusing can prevent the deformation from running randomly by restricting the deformation to the most prominent matches first.

The third strategy to prevent invalid deformation is to assign a maximum deformation range to each vertex v_i , which essentially restricts the possible deformation range of a vertex within a sphere in one iteration. The radius of the maximum deformation range is defined as half of the distance between a vertex and its closest neighbor. This range is adaptively updated at the end of each iteration.

During the local deformation stage, each vertex is labeled with a status value. There are four types of status values: “Free”, “Intermediate”, “Restricted” and “Stable”. All vertices are initialized as “Free” vertices. When the template

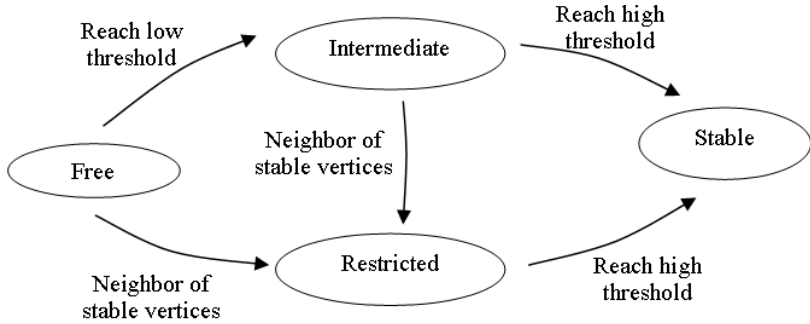


Fig. 7. Status transition of vertices in local deformation.

matching of a vertex (i.e. the difference between its template attribute vector and its best matched image attribute vector) is higher than a low threshold, it becomes an “Intermediate” vertex. When the template matching of a vertex reaches a high threshold, it becomes a “Stable” vertex. “Stable” vertices cannot be deformed any more. The first layer neighbors of “Stable” vertices are defined as “Restricted” vertices. “Free”, “Intermediate” and “Restricted” vertices have different deformation ranges. “Intermediate” vertices have less freedom in deformation than “Free” vertices. And “Restricted” vertices have even restrictive deformation ranges. Figure 7 illustrates the status transition of vertices. Only “Intermediate” and “Restricted” vertices can be deformation focus. “Free” vertices cannot be deformation focus and can only be morphed by its neighbors. The low threshold and the high threshold used in our investigation are 0.5 and 0.9, where 0 indicates totally unmatched and 1 indicates perfectly matched. The status transition restricts the deformation of certain vertices to guarantee the smoothness and continuity of the deformation.

5. Results and Validation

We have tested the non-rigid registration algorithm using a statistical density model of hemi-pelvis. The statistical model is computed from eight training images (Sec. 2). An additional pelvis CT image was then acquired, and the non-rigid registration was applied between the statistical model and the CT image. Figure 8 shows some visual results of the registration process at different stages. The first row is a 3D visualization of the process in a volume rendering mode, and the second row shows one cross-section of the volume. The model is superimposed on the CT image to illustrate the match between the model structure and the anatomical structure. Figure 8(a) is the initial state of the model. Figure 8(b) is the result after the affine transformation. Figure 8(c) is the result after the global deformation by optimizing five statistical mode parameters. After this stage, the shape of the model roughly matches the pelvis boundary, but some small discrepancies still exist due to limited variability in the model. Finally, Fig. 8(d) is the result after the local correspondence finding and deformation stage. The superimposing of the model and

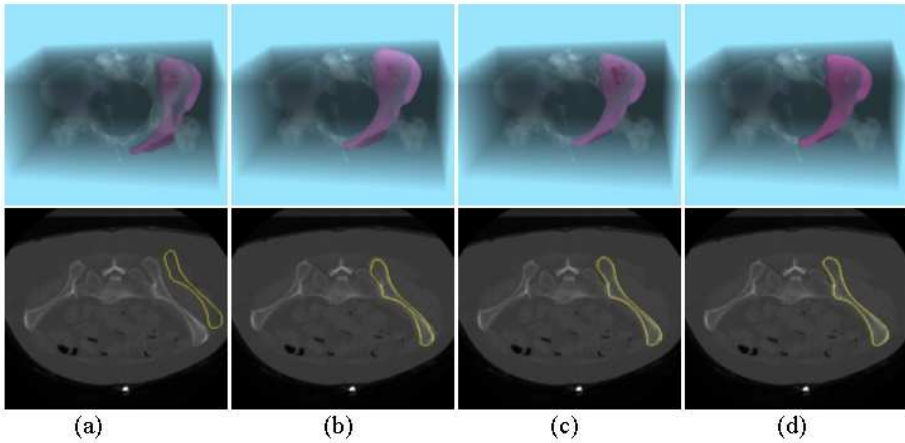


Fig. 8. Visual results of non-rigid registration between the statistical model and CT images (a) Initial state; (b) after affine transformation; (c) after global deformation; (d) after local deformation.

the image demonstrates a very close match between the pelvis boundary and the deformed model.

Figure 9 plots the value of the energy function between the statistical model and the CT image during the registration procedure. X-axis denotes the iteration index, and Y-axis is the value of the energy function [Eq. (4)]. The energy function is normalized to the range of $[0 \dots 100]$, where 0 means the model and the image are perfectly matched, and 100 means that they are totally unmatched. From Fig. 9, the initial value of the energy function is about 90. After the affine transformation stage, it drops to around 20. And at the end of the global deformation stage, it decreases to around 6. Finally, the local deformation stage further improves the energy function to about 3. There are several “jumps” in the plot; the jump at first iteration is the result of moving the centroid of the model to the center of the image volume; other jumps are the result of changing image resolution in the multi-resolution scheme. The total process takes about 10 minutes on a Pentium III 850 PC. Among these, the initialization stage takes about 2 minutes, the affine transformation stage takes about 1 minute, the global deformation stage takes about 2 minutes, and the local deformation stage takes about 5 minutes.

Currently there are eight pelvis CT images in the training set. A “leave-one-out” validation was conducted on the training data sets. One data set in the training set was selected as the testing data, and the other seven data sets were used to build a statistical model. Then the statistical model was registered with the testing data set, and the registration results were validated with a ground truth model. The ground truth model was obtained by a semi-automatic segmentation followed by manual verification and adjustment. Volume and surface-based error metrics have been implemented for the validation. The error metrics measure the overlapped volume of two volumetric models and the surface distance. First a scan of voxels

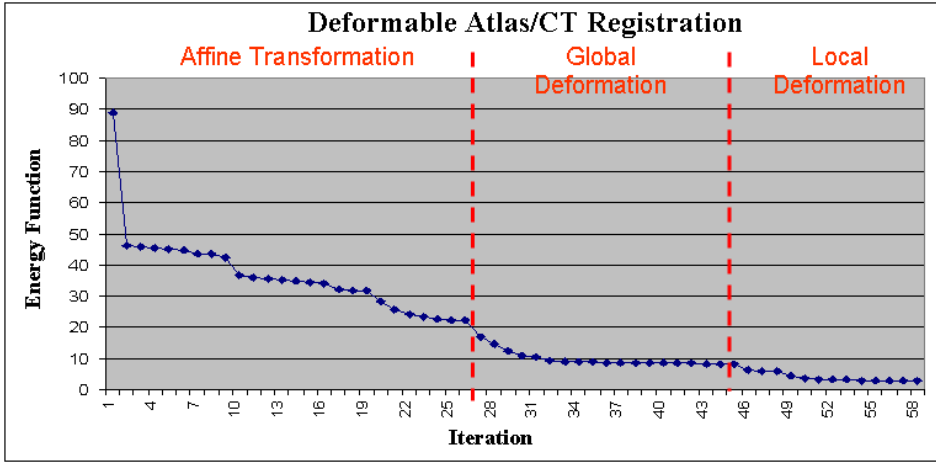


Fig. 9. Energy function in non-rigid model/CT registration.

inside the ground truth model was obtained. And a scan of voxels inside the model produced by the non-rigid registration process was also generated. Assume model 1 is the model produced by the registration procedure, and model 2 is the ground truth model. The percentage of the number of overlapping voxels in models 1 and 2 is computed, written as

$$\text{Overlap} = \|V_1 \cap V_2\| / \|V_1\| \times 100\% \tag{13}$$

where V_1 is the set of voxels in model 1, V_2 is the set of voxels in model 2, and $\|\cdot\|$ represents the size of a set; and the distance between exterior surfaces of the two models is also evaluated, including

$$\begin{aligned} \text{Avg}D &= \text{avg}_{\mu_i \in V_1} (d(\mu_i, p(\mu_i))) \\ \text{Max}D &= \max_{\mu_i \in V_1} (d(\mu_i, p(\mu_i))) \end{aligned} \tag{14}$$

where i_i is the vertex on the exterior surface of V_1 , and $p(\mu_i)$ is closest point of μ_i on the exterior surface of V_2 , $d(*, *)$ is the distance between two points. $\text{Avg}D$ measures the mean distance, $\text{Max}D$ measures the maximum distance. The unit is mm.

Table 1 lists the results of the leave-one-out validation experiment. The results were evaluated at the beginning of the registration and at the end of each stage. The overlap percentage [Eq. (13)] and the distance between exterior surfaces [Eq. (14)] of the transformed model and the ground truth model were reported. In the table, “AvgD” is the mean surface distance, and “MaxD” is the maximum surface distance. “Overlap” is the percentage of overlapping voxels to all voxels within the model. In the average case, about 94% overlap between the ground truth model and the registered model was achieved. The leave-one-out validation result in Table 1

Table 1. Leave-one-out validation of non-rigid model/CT registration.

Data set	Initial Stage	Affine Transformation		Global Deformation		Local Deformation (Final Result)		
	Overlap	AvgD	Overlap	AvgD	Overlap	AvgD	MaxD	Overlap
1	0.0%	3.34	84.5%	1.22	91.3%	0.58	3.52	96.4%
2	5.4%	4.11	82.4%	1.07	89.1%	0.79	4.52	94.3%
3	0.0%	3.05	82.4%	1.34	88.6%	0.75	4.17	94.1%
4	0.0%	3.19	78.2%	1.29	89.1%	0.6	3.66	93.2%
5	0.0%	4.39	81.1%	1.25	89.3%	0.81	4.61	94.1%
6	1.3%	3.12	84.6%	1.15	91.6%	0.55	3.41	95.6%
7	3.8%	4.81	90.2%	1.81	94.4%	0.75	3.7	97.4%
8	0.0%	3.65	79.2%	1.03	88.4%	0.69	4.36	92.2%
Avg	1.31%	3.71	82.81%	1.27	90.21%	0.69	3.99	94.67%
Std Dev	2.1%	0.65	3.7%	0.24	2.1%	0.1	0.48	1.7%

Table 2. Control parameters versus registration results.

Control Parameters	Trial 1	Overlap	Trial 2	Overlap	Trial 3	Overlap
w_s and w_d	0.3, 0.7	93.2%	0.7, 0.3	96.4%	0.8, 0.2	96.1%
Template layer	1	94.4%	2	96.4%	3	95.9%
Flexible search range	0.05	96.2%	0.1	96.4%	0.2	95.3%
Gaussian morphing	0.5	96.3%	1.0	96.4%	2.0	96.0%
Transit_low, transit_hi	0.3, 0.8	94.7%	0.5, 0.9	96.4%	0.6, 0.95	96.2%

and the visual result in Fig. 8 proved that our method was based on statistical models, and flexible mesh templates is effective on non-rigid registration of bony structures.

The effect of several control parameters in the local deformation algorithm was also evaluated. The parameters versus the registration accuracy (volume overlap) are listed in Table 2. Among those parameters, “ w_s and w_d ” are the weights for shape properties and density properties; “Template layer” is the number of layers in the mesh template; “Flexible search range” is the search range associated with each node on the template; “Gaussian morphing” is the morphing parameter; and “Transit_low and transit_hi” are thresholds used in the status transition. The detailed descriptions of the parameters can be found in Sec. 4. For each parameter, we evaluated one value used in our current algorithm (in bold font, Trial 2) and two other values. We have roughly optimized the parameters for our application. However, the comparison also showed that using other parameters does not degrade the results too much.

6. Discussion

We have presented a new method for non-rigid medical image registration and model-based segmentation using a statistical bone density model. The model is represented as a tetrahedral mesh and contains both shape and density properties and

their variability. We also proposed a multiple-layer flexible mesh template to find the correspondence between vertices on the model and voxel locations in the image.

The statistical model is computed from a population of training models, so it characterizes the variability inherent in the training set. To compensate for the variation not exhibited in the training models, further local deformation is performed using a multiple-layer flexible mesh template associated with each vertex on the model. Allowing local deformation leads to refined local matches, and is more effective than simply adding more models to the training set. The template itself is flexible and non-rigid so that it can be used to correlate approximate features. Its multiple-layer structure allows the potential to characterize different sizes of structures. 0-layer template describes the feature of a single point. If sufficient layers are used, the mesh template can be used to characterize the entire model. The mesh template also provides a way to combine the shape feature and density feature of an anatomical structure in a single data structure. The experiments of the mesh template matching on bony anatomies have been proven successful. Since the idea of the statistical model and mesh template is generic, this technique can be extended to other anatomical structures including soft tissues. In the experiment with pelvis images, more weights were put on shape properties than on density properties since the bone boundary is more prominent than density distribution inside the bone. In other applications, such as brain registration where the boundary is not as clear as bones and the information in density distribution is relatively rich, more emphasis can be put on the density properties.

The registration process has three stages: affine transformation, global deformation and local deformation. Among these, global deformation requires a statistical model built from a training set and provides a start point for further local deformation. If the statistics is not available, it is possible to skip the global deformation stage. However, since the local deformation stage only has a limited deformation range and the template matching tends to match local features, if the variability of the structure is too big, it may not be able to find matches for some vertices in their neighborhoods. We may increase the flexibility of the template and the searching range, but it would introduce a lot of ambiguity and it is difficult to determine the correct correspondence.

We adopted a multiple-resolution and multiple step size scheme in the registration. We tried an experiment using just highest resolution image and model. The optimization process was stuck in the early stage of the registration. It is still possible that registration converges to a local minimum in the multiple-resolution scheme. But we have not encountered this problem in the application of pelvis, partially because pelvis structure is not symmetric and has distinguishable local features.

In future works, we plan to build a bone density atlas based on the statistical bone density model to incorporate more information. In this atlas, anatomical landmarks are labeled and certain surgical procedures are defined. Once the atlas is registered to a specific patient image, the information stored in the atlas can be

transferred to the patient. We also plan to further extend this model to other anatomies such as knees and vertebrae. In our other work,^{35,37} we have demonstrated 2D/3D non-rigid registration between the statistical model and a set of X-ray images. In general, the 2D/3D technique can be used to perform 3D patient specific modeling and analysis without a patient specific CT image. Clinical tasks such as planning from X-rays, intra-operative guidance, post-operative analysis and retrospective studies are also potential applications of anatomical atlas and registration techniques.

Acknowledgments

This work was partially funded by NSF Engineering Research Center grant EEC9731478. We thank Shadyside Hospital for providing the pelvis data. We specially thank Dinggang Shen for his suggestion on deformable registration algorithm. We also thank Jerry Prince, Christos Davatzikos, and Chengyang Xu for their useful discussions.

References

1. A. Bauer, *Primary THR Using the ROBODOC System*, CAOS/USA, Pittsburgh, Pennsylvania, USA, 1999.
2. F. L. Bookstein, *Morphometric Tools for Landmark Data, Geometry and Biology*, Cambridge University Press, 1991.
3. M. Chen *et al.*, *3-D Deformable Registration of Medical Images Using a Statistical Atlas*, Carnegie Mellon University, Pittsburgh, PA, 1998.
4. D. Collins *et al.*, "Automatic 3D intersubject registration of MR volumetric data into standardized Talairach space," *J. Comput. Assist. Tomogr.* **18**, 2 (1994) 192–205.
5. T. F. Cootes *et al.*, *A Unified Framework for Atlas Matching Using Active Appearance Models*, IPMI, Springer 1999.
6. T. F. Cootes and C. J. Taylor, *Statistical Models of Appearance for Computer Vision*, 2000.
7. A. Evans *et al.*, "MRI-PET correlation in three dimensions using a volume-of-interest (VOI) atlas," *J. Cereb. Blood Flow and Metabolism* **11**, 2 (1991) A69–78.
8. J. Feldmar and N. Ayache, *Rigid, Affine and Locally Affine Registration of Free-Form Surfaces*, INRIA, 1994.
9. M. Ferrant *et al.*, *3D Image Matching Using a Finite Element Based Elastic Deformation Model*, Radiology, Brigham and Women Hospital, Harvard Medical School, Boston, USA, 1999.
10. M. Fleute and S. Lavalée, *Nonrigid 3-D/2-D Registration of Images Using Statistical Models*, MICCAI, Cambridge, UK, 1999.
11. J. Haller *et al.*, "Hippocampal MR imaging morphometry by means of general pattern matching," *Radiology* **199** (1996) 787–791.
12. D. J. Hawkes, "Review: algorithms for radiological image registration and their clinical application," *J. Anat.* **193** (1998) 347–361.
13. M. Kass, A. Witkin and D. Terzopoulos, "Snakes: active contour models," *Int. J. Comput. Vis.* (1988) 321–331.
14. C. Krettek *et al.*, "Computer assisted femoral nailing," *4th Int. Symp. CAOS*, Davos, Switzerland, 1999.

15. M. E. Leventon, *Statistical Models in Medical Image Analysis*, Electrical Engineering and Computer Science, MIT 2000, p. 156.
 16. D. G. Lowe, "Fitting parameterized three-dimensional models to images," *IEEE Trans. Patt. Anal. Mach. Intell.* **13**, 5 (1991) 441–450.
 17. C. R. Maurer *et al.*, "Registration of 3-D images using weighted geometrical features," *IEEE Trans. Med. Imag.* **15**, 6 (1996) 836–849.
 18. D. Meyers, S. Skinner and K. Sloan, "Surfaces from contours," *ACM Trans. Graph.* **11**, 3 (1992) 228–258.
 19. M. I. Miller *et al.*, "Mathematical textbook of deformable neuroanatomies," *Proc. Natl. Acad. Sci.* **90**, 24 (1993) 11944–11948.
 20. W. H. Press, *et al.*, *Numerical Recipes in C.*, 2nd edition, Cambridge University Press, 1992.
 21. G. K. Rohde, A. Aldroubi and B. M. Dawant, "Adaptive free-form deformation for inter-patient medical image registration," *SPIE Med. Imag.*, San Diego, CA, 2001.
 22. D. Rueckert *et al.*, "Nonrigid registration using free-form deformations: Application to breast MR Images," *IEEE Trans. Med. Imag.* **18**, 8 (1999).
 23. D. Shen and C. Davatzikos, *Adaptive-Focus Statistical Shape Model for Segmentation of 3D MR Structures*, MICCAI, Pittsburgh, PA, 2000.
 24. C. Studholme, D. Hill and D. Hawkes, *Multiresolution Voxel Similarity Measures for MR-PET Registration*, IPMI, 1995.
 25. G. Subsol, J.-P. Thirion and N. Ayache, *A General Scheme for Automatically Building 3D Morphometric Anatomical Atlas: Application to a Skull Atlas*, INRIA, France, 1995.
 26. J.-P. Thirion, *Fast Non-Rigid Matching of 3D Medical Images*, INRIA, 1995.
 27. P. Thompson and A. W. Toga, "A surface-based technique for warping three dimensional atlas to match anatomical brain images," *IEEE Trans. Med. Imag.* **15** (1996) 402–417.
 28. I. J. Trotts, B. Hamann and K. I. Joy, "Simplification of tetrahedral meshes with error bounds," *IEEE Trans. Visual. Comput. Graph.* **5**, 3 (1999) 224–237.
 29. P. Van Den Elsen *et al.*, "Grey value correlation techniques used for automatic matching of CT and MR volume images of the head," *SPIE Med. Imag.*, 1994.
 30. S. Warfield *et al.*, *Automatic Identification of Gray Matter Structures from MRI to Improve the Segmentation of White Matter Lesions*, Medical Robotics & Computer Assisted Surgery (MRCAS), 1995.
 31. S. Warfield *et al.*, "Adaptive, template moderated, spatially varying statistical classification," *Med. Imag. Anal.* **4**, 1 (2000) 43–55.
 32. W. M. Wells *et al.*, "Adaptive segmentation of MRI data," *IEEE Trans. Med. Imag.* **15**, 4 (1996) 429–442.
 33. R. Woods *et al.*, "Automated image registration: II. Intersubject validation of linear and nonlinear models," *J. Comput. Assist. Tomogr.* **22**, 1 (1998) 153–165.
 34. J. Yao and R. Taylor, *Tetrahedral Mesh Modeling of Density Data for Anatomical Atlases and Intensity-Based Registration*, MICCAI, Pittsburgh, PA, Springer, 2000.
 35. J. Yao, "A statistical bone density atlas and deformable medical image registration," *Computer Science*, The Johns Hopkins University, Baltimore, MD, 2001, p. 264.
 36. J. Yao and R. Taylor, "Construction and simplification of bone density models," *SPIE Med. Imag.*, San Diego, CA, 2001.
 37. J. Yao and R. Taylor, "Deformable registration between a statistical bone density atlas and X-ray images," *International CAOS*, 2002.
-



Jianhua Yao received B.S., M.Sc and Ph.D. degrees in computer science from Tianjin University (China), Tsinghua University (China), and The Johns Hopkins University (USA) in 1993, 1996 and 2001, respectively. Currently,

he is a staff scientist at National Institute of Health.

His research interests include clinical image processing and medical image registration.



Russell H. Taylor received a B.E.S. degree from The Johns Hopkins University in 1970 and a Ph.D. in computer science from Stanford in 1976. He joined IBM Research in 1976. In 1995, Dr. Taylor moved to The Johns

Hopkins University as a Professor of computer science, with joint appointments in radiology and mechanical engineering. He is also Director of the NSF Engineering Research Center for Computer-Integrated Surgical Systems and Technology. He has worked on all aspects of CIS systems, including modeling, registration, and robotics in areas including percutaneous local therapy, microsurgery, and computer-assisted bone cancer surgery. He is Editor Emeritus of the *IEEE Trans. Robotics and Automation*, a Fellow of the IEEE and the AIMBE.