Automatic Generation of Shape Models Using Nonrigid Registration with a Single Segmented Template Mesh

Geremy Heitz, Torsten Rohlfing, and Calvin R. Maurer, Jr.

Image Guidance Laboratories
Department of Neurosurgery
Stanford University
300 Pasteur Drive, Stanford, CA, 94305-5327, USA
Email: gaheitz@stanford.edu

Abstract

Statistical shape modeling using point distribution models (PDMs) has been studied extensively for segmentation and other image analysis tasks. Methods investigated in the literature begin with a set of segmented training images and attempt to find point correspondences between the segmented shapes before performing the statistical analysis. This requires a time-consuming preprocessing stage where each shape must be manually or semi-automatically segmented by an expert. In this paper we present a method for PDM generation requiring only one shape to be segmented prior to the training phase. The mesh representation generated from the single template shape is then propagated to the other training shapes using a nonrigid registration process. This automatically produces a set of meshes with correspondences between them. The resulting meshes are combined through Procrustes analysis and principal component analysis into a statistical model. A model of the C7 vertebra was created and evaluated for accuracy and compactness.

1 Introduction

Statistical models of shape in general, and point distribution models (PDMs) in particular, have seen wide use in the medical image analysis community [1]. These models contain *a priori* information about an object that can be used in a wide variety of tasks, including recognition, classification, and reconstruction.

As many authors have pointed out, however, accurate models require a reliable and consistent labeling of the training shapes used to build the model [2]. In addition to requiring that the landmarks accurately represent each object individually, there is a requirement of identifying which landmarks across the training set correspond to each other. The problem of landmark designation across the training set is called the *correspondence problem*, and is one of the primary directions of research in this area. Because manual landmark selection is a long and tedious process, particularly in 3D, an automatic mechanism of labeling the corresponding points on the various shapes is required. Additionally, manually determining landmark correspondences between two shapes is observer dependent; an automatic method can remove this extra source of variability.

In this paper we present a procedure for automatically generating a statistical shape model of biological structures from a set of training images. Our method produces a surface model for the shape in question, and requires hand segmentation of only one training image. In Section 2 we review work by other groups towards solving the correspondence problem. Section 3 gives a detailed description of our method, followed by an evaluation of our method applied to the C7 vertebra in Section 4. Section 5 discusses these results.

2 Prior Work

Several groups have approached the correspondence problem from a number of directions. Work by Davies *et al.* [3] optimizes an information theoretic cost function of the landmark positions used to create the shape model. The motivation for this approach comes from the hypothesis that the best

model is the one that is simplest and hence generalizes the best. Towards this end they find a parameterization of the shape surface that minimizes the information required to encode the model plus the information in the training set, given that model.

Another approach was championed by Lorenz and Krahnstöver [4]. They seek to solve the problem by creating a mesh from a template shape, and "coating" the mesh vertices onto the other shapes in the training set. The vertices are coated onto the target shape in a two-step process that involves an initial affine registration, followed by an active surface technique patterned after that described by McInerney and Terzopoulos [5].

Paulsen and Hilger [6] use a similar strategy. However, they cast the problem in a Bayesian framework, seeking a deformation field that maximizes an *a posteriori* probability measure.

Methods proposed by Frangi, Rueckert and colleagues [7] are the most similar to the one described in this paper. Their scheme relies on a nonrigid registration of the manually segmented training shapes. Work on so-called active deformation models (ADMs) [8] foreshadows the tools used in this paper, where a dense set of correspondences (a deformation field) between images is used to propagate surface landmarks from a template mesh to each sample in the training set. They suggest that the deformation fields can be used to propagate a shape representation to the shapes in a training set. They construct a model of the ventricles and caudate nuclei with a single surface segmentation, but provide no details or quantitative assessment of it. Our work explores this possibility further, creating a vertebral shape model in order to evaluate the viability of such a method.

3 Shape Model Generation

The goal of our work is to create a statistical shape model of the surface of an object given a set of three-dimensional (3D) images containing instances of the object. We do not assume that any of the images are previously segmented, and there is no requirement of uniform resolution or size across the training set. The method we developed is semi-automatic, in that it requires user initialization, but is fully automatic after this beginning phase. The primary advantage of this method over those previously described is the need to segment only one shape in the training set, as opposed to every one.

The process takes a set of N training images as input, and the technique for shape model generation described here involves five steps:

1. Selection of a template image and initial segmentation of the template shape.
2. Generation of a template surface mesh from this segmentation.
3. Intensity-based nonrigid registration of the template image to the remaining training set images.
4. Warping of the template mesh to the training set shapes using the transformation produced by the nonrigid registrations.
5. Alignment and statistical analysis of the training set meshes resulting from warping of the template mesh.

This model creation process is shown schematically in Fig. 1, and each step of the process is described in further detail below.

3.1 Template Selection and Segmentation

The first step in the creation pipeline is to select a template image from the training set, and segment the template shape in the selected image. In our case, we interactively segmented the template shape in each slice of the image using an adaptive boundary segmentation method known as "intelligent scissors" or "Live-wire" [9]. This produces a label image that describes pixels as either inside or outside the shape.

Choosing a good template shape is important for the results of this process, as will be discussed later. The quality of a template can be judged by the quality of the registrations obtained between the template and all the other images in the set.

3.2 Template Mesh Generation

From the template shape label image produced in the first step, the marching cubes algorithm is utilized to generate a triangulated surface mesh. Given this initial fine mesh, we smooth and decimate it to a desired mesh resolution. These operations were performed using the Visualization Toolkit (VTK) software [10], which provides many tools for mesh manipulation.

It is important to note that this process of segmentation followed by marching cubes with mesh refinement is performed *only for the template*

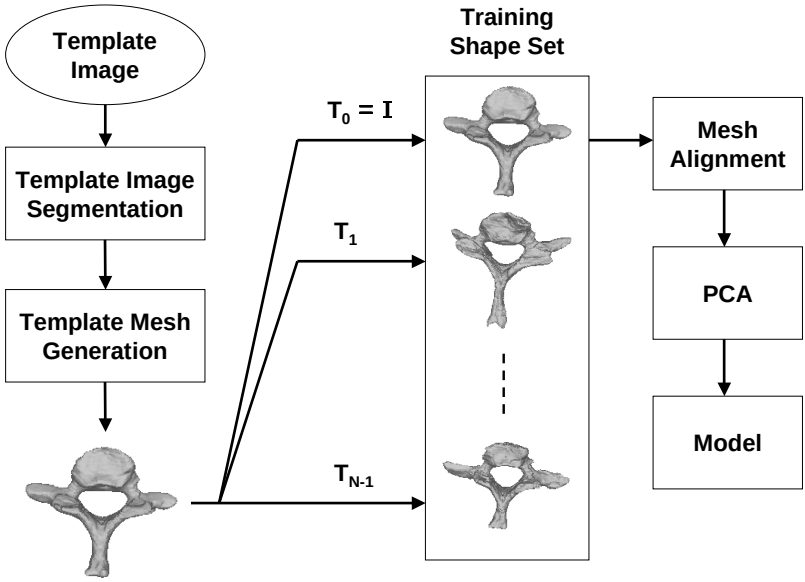

Figure 1: Statistical shape model creation process. First, a template image is selected from the training set. The template shape is segmented in the selected image, which produces a binary label image. A template mesh is generated from the label image using marching cubes and decimated and smoothed to a desired mesh resolution. The template mesh is warped using nonrigid transformations produced by intensity-based nonrigid registration of the template image to the training images. This produces a set of training set meshes, which are aligned using generalized Procrustes analysis. Finally, principal component analysis (PCA) is performed to obtain the statistical shape model.

shape. The number of vertices and triangles chosen for the template shape will then be fixed across the training set, with a one-to-one correspondence between vertices in each of the training surfaces.

3.3 Nonrigid Registration of Template Image to Training Images

In order to warp the template mesh onto the other shapes in the training set, we must determine the appropriate transformation to use. This deformation is computed using an initial affine registration step, followed by an independent implementation [11] of an intensity-based nonrigid registration technique developed by Rueckert *et al.* [12]. The 3D grayscale template image is registered to each of the remaining training set images containing the anatomical object of interest. The deformation is defined on a uniformly-spaced grid, and uses B-splines for interpolation between control points. The 3D control points $\phi_{i,j,k}$ can be positioned in the target image independently, and the transformed location of a source point (x, y, z) in the target image is determined by:

$$\mathbf{T}(x,y,z) = \sum_{l,m,n=0}^{3} B_l(u) B_m(v) B_n(w) \phi_{i+l,j+m,k+n} \quad (1)$$

In this equation, i, j, and k denote indices of the control point cell that contains (x, y, z), and (u, v, w) represents the relative position of (x, y, z) within the cell. These quantities can be computed using:

$$i = \lfloor \frac{x}{\delta_x} \rfloor - 1, j = \lfloor \frac{y}{\delta_y} \rfloor - 1, k = \lfloor \frac{z}{\delta_z} \rfloor - 1 \quad (2)$$

and

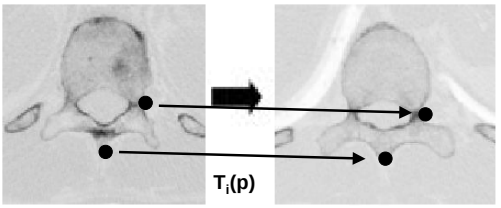

Figure 2: Mapping of landmarks from the source or template shape (left) onto the target or training shape (right). The mapping is the transformation produced by the intensity-based nonrigid registration of the template image to the training image.

$$u = \frac{x}{\delta_x} - (i+1), v = \frac{y}{\delta_y} - (j+1), w = \frac{z}{\delta_z} - (k+1) \quad (3)$$

where δ_x, δ_y, and δ_z are the fixed distances between the control points in each dimension in the source image. The B_0 through B_3 functions are the third-order approximating spline polynomials. The optimization step finds the control point displacements that maximize the image similarity measure, normalized mutual information (NMI) [13].

The output of this step provides a dense correspondence between points in the template and points in the training shape. This warp can be thought of as a lookup function that specifies where in the target (training) image to look to find the pixel that corresponds to a given pixel in the source (template) image.

3.4 Warping of the Template Mesh

We compute a non-rigid transformation for every shape in the training set. Note that the warp for the shape used as the template is the identity transformation. The transformation for the i^{th} training shape is denoted T_i, and is used to produce the mesh corresponding to the shape in the i^{th} image. The target mesh is produced by finding the warped location of every vertex in the template mesh. In particular, the n^{th} template surface vertex, $v_n = (x_n, y_n, z_n)$, is transformed onto the training surface to produce the vertex $v'_n = T_i(x_n, y_n, z_n)$. An example of this mapping from source to target is shown in Fig. 2.

The locations of the mesh vertices are now described in the local coordinates of the target image, and should lie on the surface of the structure if the nonrigid transformation is accurate. The application of the transformation corresponding to each training sample is shown in Fig. 1, resulting in a set of such surfaces with vertex correspondences.

3.5 Alignment and Statistical Analysis

Given a set of N surface meshes, one for each of the training shapes, the next step is to align the shapes into a common reference frame and remove the rotational, scaling, and translational components, which we do not wish to include in the model of shape. This is done with an iterative method known as generalized Procrustes analysis (GPA) [14].

Although our registration technique begins with an affine registration step, we still must perform the Procrustes analysis to determine the affine transformation that relates the training mesh to the template mesh for two reasons. The first is that the elastic registration phase may add some additional global affine components. This must be included in the final alignment of the training shapes. The second reason is that we are concerned with the optimal transformation between the object surfaces. While the global transformation is optimal across the entire volumetric image, a slightly different affine transformation may more accurately relate the surface representations. For these reasons, we perform the GPA step once all the surface meshes have been obtained. The result is a set of triangulated meshes with vertex correspondences that are aligned in a common reference frame with the affine components of their variations removed.

The final step in construction of the statistical model is to perform principal component analysis (PCA) on the shapes. We represent each shape as a vector of the vertex positions, i.e., a vector $\mathbf{v}_i = (x_1, y_1, z_1, x_2, y_2, z_2, ..., x_n, y_n, z_n)$. The training vectors create a point cloud in $3n$ dimensions, where n is the number of vertices in each mesh. Modeling this point cloud as a multivariate Gaussian distribution, we can then compute the mean and covariance of the distribution:

$$\bar{\mathbf{v}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{v}_i \quad (4)$$

and

$$\mathbf{C} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{v}_i - \bar{\mathbf{v}})(\mathbf{v}_i - \bar{\mathbf{v}})^T. \quad (5)$$

The principal components (or principal axes) of this distribution are then given by the eigenvectors of the covariance matrix $\mathbf{C}$, i.e., vectors $\mathbf{p}_k$ such that, for some eigenvalue λ_k:

$$\mathbf{C}\mathbf{p}_k = \lambda_k \mathbf{p}_k. \quad (6)$$

Creating a matrix $\mathbf{P}$ with the m eigenvectors corresponding to the largest eigenvalues as columns, we can approximate any shape in the training set using the mean and these m modes of variation as:

$$\mathbf{v} \approx \bar{\mathbf{v}} + \mathbf{P}\mathbf{b}. \quad (7)$$

In addition, we hope that this decomposition of the surfaces generalizes well, i.e., that any viable instance of the shape is described using appropriate values in the vector of mode coefficients, $\mathbf{b}$.

4 Results

The method described in the preceding section was used to create a surface model for the C7 vertebra. As training data, we used ten CT image volumes of the neck and upper spine. The in-slice image resolution varied from 0.5 to 1.0 mm; the slice thickness was 1.25 mm for all images. To isolate the C7 vertebra, each image was manually cropped to include the vertebra in question and approximately half of each vertebra on either side.

4.1 Registration Accuracy

As mentioned above, the accuracy of the training mesh as a representation for the surface in the training image is directly determined by the quality of the nonrigid registration relating the template image to the training image. If the surfaces of the anatomical structures can be put in one-to-one correspondence, and the registration is perfect (i.e., the mapping is correct and exact), then the vertices transformed by the warp will lie on the surface, and the resulting mesh will be accurate.

In order to measure this accuracy, we manually segmented each vertebra (Section 3.1). This produced a label image where the foreground label is applied to pixels inside the vertebra, and the background label is applied to all other pixels. Running the marching cubes algorithm on this segmentation produced a mesh that we then used as a ground truth with which to compare our automatic results.

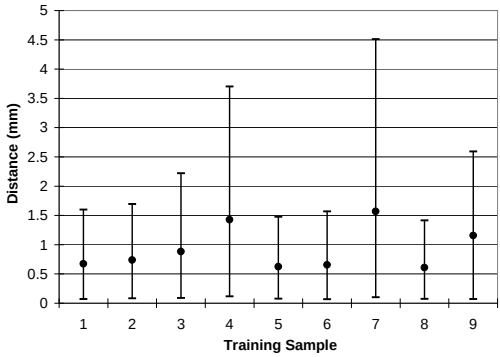

Figure 3: Distance between the vertices of the transformed (automatically generated) mesh used to represent the training sample surface and the ground truth surface mesh (manually segmented). This distance is computed for each vertex on the mesh, and the mean distance is denoted by the black circle. For each training sample, the graph also shows the range between the 10^{th} and 90^{th} percentiles. In other words, 10% of the vertices had distances smaller than the bottom hash mark, and 10% had distances larger than the top hash mark.

We chose to measure the distance between the mesh vertices and the closest point on the ground truth mesh surface as a quantitative description of mesh accuracy. This is a measure of how closely the automatically generated mesh surface approximates the true surface according to the manual segmentation. Figure 3 shows the results of applying this measurement to each mesh in the training set. For each training shape, the mean surface-to-surface distance is reported. In addition, the graph also shows the range over which the distances vary, between the 10^{th} and 90^{th} percentiles.

The mean distance was typically about 1 mm, which is comparable to the in-plane resolution of the images. The mean error is thus on the order of one pixel dimension, and less than the slice thickness.

4.2 Model Evaluation

To further evaluate the scheme described above, we performed the alignment step and the statisti-

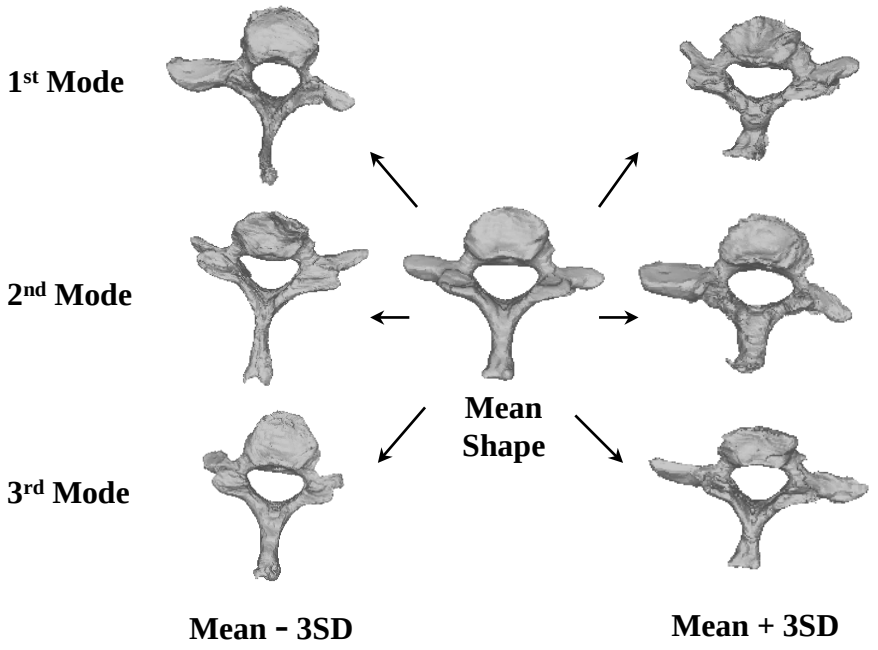

Figure 4: Instances generated from the model of the C7 vertebra. The first three modes are varied across three standard deviations. The modes are organized in rows, with the left vertebra for each pair being the mean shape minus three standard deviations (SD), and the right vertebra being the mean plus three SD. The mean shape appears in the middle of the figure.

cal analysis on the ten resulting meshes, and investigated the model that was created from this training set. Figure 4 shows some instances generated from the model. Here the first three modes are varied across three standard deviations. The modes are organized in rows, showing the mean shape minus three standard deviations (SD), and the mean plus three SD. The mean shape appears in the middle of the figure.

The modes correspond to physical aspects of the vertebra. The third mode, for instance, corresponds to a variation in length of the transverse processes. This fits well with our intuition that the variation across the training set should show variations in widths, lengths, etc. In addition, because the structures are symmetric, we expect the variations to also be symmetric, which we observe in this figure. There is not a noticeable twisting effect among the modes, which demonstrates that the vertices on the meshes did not slide across the surfaces. Sliding would not be desirable, as it would mean that the correspondences are incorrect, and would lead to non-viable instances for certain values of the mode coefficient vector.

Because one of the primary reasons for using a PDM is the ability of this model to describe the data with a small number of degrees of freedom, we investigated how much variability was described by each mode of variation. Figure 5 shows the percentage of shape variance described by the first n modes. Ideally, if there is little variation in the training set, we should be able to explain most of it with a smaller number of modes. As can be seen, we can describe 88% of the variance using only five modes.

The final aspect of the model that we investigated was the accuracy of the mesh representation of the training set shapes using various numbers of modes. We computed the coefficient vector $\mathbf{b}$ required to fully reconstruct each training sample, and then reconstructed the mesh by truncating $\mathbf{b}$. In order to

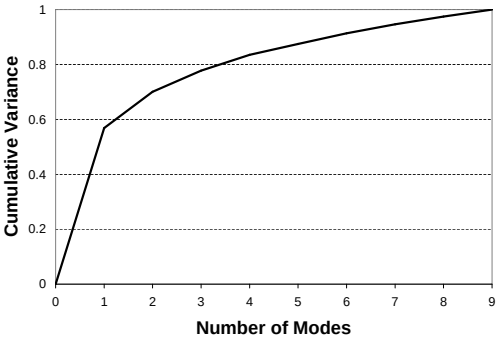

Figure 5: The cumulative shape variance described by using the first n modes of variation. Note that with zero modes we are using the mean mesh to represent each surface, so none of the variance is explained. With all modes, however, the entire training set can be described perfectly. With five modes, we can explain 88% of the variation.

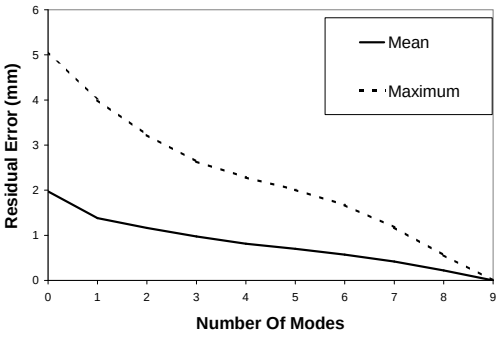

Figure 6: The residual error using the first n modes of variation to describe the training set. The error here measures the distance from each vertex reconstructed with n modes to the corresponding vertex in the true mesh. These distances are then averaged across the entire training set.

represent the mesh with $m \leq n$ modes, we retain the first m values in the vector $\mathbf{b}$ and set the rest to zero. Truncating $\mathbf{b}$ in this manner is the optimal method for compressing the shape representation, i.e., causes the least loss in accuracy. The metric used to measure accuracy was the residual error between the reconstructed mesh vertex and the true mesh vertex (the mesh produced by warping the template mesh), measured in millimeters.

Figure 6 shows the mean and maximum residual error using the first n modes, and averaged across each shape in the training set. Notice that even if we use only the mean shape (no modes of variation) to reconstruct each mesh, we still see only 2 mm of mean residual error. This means that most vertices in a given mesh do not deviate very far from their mean values. We can also see that, as expected, the error goes to zero when we use all nine modes. It is a mathematical consequence of PCA that we can reconstruct all ten training vectors perfectly if we use all nine modes.

5 Discussion and Conclusion

In this paper we have presented an automatic procedure for the construction of 3D statistical shape models. The method requires the manual segmentation of only one image in the training set, and generates the surfaces and provides correspondences for the other images. This is an improvement over similar existing methods, which require a presegmentation of every shape in the training set.

Results from the creation of a C7 vertebral model showed that an accurate representation can be achieved (with a mean surface distance of about 1 mm) by warping the template mesh to correspond to the appropriate training image. In addition, the shape model can describe the mesh to an accuracy of about 1 mm using only five modes of variation.

The performance of our method is comparable to a study done by Kaus *et al.* [2]. They used images with a slice thickness of 2.0 mm, and a similar range of in-plane resolutions. They were able to obtain a mean surface distance of 0.8 mm, and a mesh accuracy of 1.6 mm with ten modes (they used a set of 32 vertebrae, including L1-L4). Their results required a prior segmentation of every shape, however, and we believe the savings in time using our method are worth the slight decrease in surface representation accuracy. In addition, for most applications an accuracy of 1 mm should be sufficient.

Their method treats the affine-transformed mesh as an active surface, and allows the mesh to deform so as to push the mesh vertices towards the boundaries in the volumetric image. While their input images consist of white foreground pixels on a black background, this process would likely work with the original grayscale image. As a result, we believe that this technique could also be applied to our warped meshes, and work is underway to refine the

mesh after the non-rigid warping step to improve its accuracy. Additional refinement of the mesh should help to push the vertices closer to the boundaries, creating a more accurate surface representation.

The experiments described above were performed on a small set of ten CT images, and we believe that it is important to evaluate the method on a larger set. We hope to obtain a larger set of images and build a model from these in the future. Because registration with NMI has been shown to work across imaging modalities, we do not require a training set confined to one modality. Thus, we may be able to build larger training sets by combining imaging data from a variety of sources. The quality of the model should improve with the size of the training set.

While our model results are comparable to models generated with other methods, the true test of the model is how well it works when applied to various segmentation and image analysis tasks. This technique should work for any shape and is sufficiently general to provide many future opportunities.

ACKNOWLEDGEMENT

The authors thank Daniel Rueckert, Department of Computing, Imperial College London, for extended discussions and valuable feedback about this work.

References

[1] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models – their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[2] M. R. Kaus, V. Pekar, C. Lorenz, R. Truyen, S. Lobregt, and J. Weese. Automated 3-D PDM construction from segmented images using deformable models. *IEEE Transactions on Medical Imaging*, 22(8):1005–1013, 2003.

[3] R. H. Davies, T. F. Cootes, and C. J. Taylor. A minimum description length approach to statistical shape modelling. In *Information Processing in Medical Imaging*, volume 2082 of *LNCS*, pages 50–63. Springer-Verlag, 2001.

[4] C. Lorenz and N. Krahnstöver. Generation of point-based 3D statistical shape models for anatomical objects. *Computer Vision and Image Understanding*, 77(2):175–191, 2000.

[5] T. McInerney and D. Terzopoulos. Deformable models in medical images analysis: a survey. *Medical Image Analysis*, 1(2):91–108, 1996.

[6] R. R. Paulsen and K. B. Hilger. Shape modelling using markov random field restoration of point correspondences. In *Information Processing in Medical Imaging*, volume 2732 of *LNCS*, pages 1–12. Springer-Verlag, 2003.

[7] A. F. Frangi, D. Rueckert, J. A. Schnabel, and W. J. Niessen. Automatic 3D ASM construction via atlas-based landmarking and volumetric elastic registration. In *Information Processing in Medical Imaging*, volume 2082 of *LNCS*, pages 78–91. Springer-Verlag, 2001.

[8] D. Rueckert, A. Frangi, and J. A. Schnabel. Automatic construction of 3-D statistical deformation models of the brain using nonrigid registration. *IEEE Transactions on Medical Imaging*, 22(8):1014 –1025, 2003.

[9] E. N. Mortensen, B. S. Morse, W. A. Barrett, and J. K. Udupa. Adaptive boundary detection using 'live-wire' two-dimensional dynamic programming. In *IEEE Proceedings of Computers in Cardiology*, pages 635–638, 1992.

[10] W. Schroeder, K. M. Martin, and W. E. Lorensen. *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics (2nd ed.)*. Prentice-Hall, 1998.

[11] T. Rohlfing and Jr C. R. Maurer. Non-rigid image registration in shared-memory multiprocessor environments with application to brains, breasts, and bees. *IEEE Transactions on Information Technology in Biomedicine*, 7(1):16–25, 2003.

[12] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes. Non-rigid registration using free-form deformations: Application to breast MR images. *IEEE Transactions on Medical Imaging*, 18(8):712–721, 1999.

[13] Colin Studholme, Derek L. G. Hill, and David J. Hawkes. An overlap invariant entropy measure of 3D medical image alignment. *Pattern Recognition*, 32(1):71–86, 1999.

[14] J. C. Gower. Generalised Procrustes analysis. *Psychometrika*, 40:33–51, 1975.