Statistical Shape Model Generation Using Nonrigid Deformation of a Template Mesh

Geremy Heitz[a], Torsten Rohlfing[b], Calvin R. Maurer, Jr.[c]

[a]Department of Electrical Engineering, Stanford University, Stanford, CA 94305
[b]Neuroscience Program, SRI International, Menlo Park, CA 94025
[c]Department of Neurosurgery, Stanford University, Stanford, CA 94305

ABSTRACT

Active shape models (ASMs) have been studied extensively for the statistical analysis of three-dimensional shapes. These models can be used as prior information for segmentation and other image analysis tasks. In order to create an ASM, correspondence between surface points on the training shapes must be provided. Various groups have previously investigated methods that attempted to provide correspondences between points on pre-segmented shapes. This requires a time-consuming segmentation stage before the statistical analysis can be performed. This paper presents a method of ASM generation that requires as input only a single segmented template shape obtained from a mean grayscale image across the training set. The triangulated mesh representing this template shape is then propagated to the other shapes in the training set by a nonrigid transformation. The appropriate transformation is determined by intensity-based nonrigid registration of the corresponding grayscale images. Following the transformation of the template, the mesh is treated as an active surface, and evolves towards the image edges while preserving certain curvature constraints. This process results in automatic segmentation of each shape, but more importantly also provides an automatic correspondence between the points on each shape. The resulting meshes are aligned using Procrustes analysis, and a principal component analysis is performed to produce the statistical model. For demonstration, a model of the lower cervical vertebrae (C6 and C7) was created. The resulting model is evaluated for accuracy, compactness, and generalization ability.

Keywords: Statistical shape models, active shape models, correspondence problem, nonrigid registration, vertebral models, principal component analysis

1. INTRODUCTION

Many tasks in medical image analysis, including segmentation, recognition, and classification, are enhanced by the use of *a priori* information. Such information can inform processing tools about what they are looking for, providing accuracy, robustness, and speed improvements. Statistical models of shape, which capture the range of variations in a population, have become widespread in the medical image analysis community, and point distribution models (PDMs), in particular, are a commonly used tool.[1–4]

In order to produce an accurate model, a set of training shapes must be reliably and consistently labeled through some manual or automatic process. On top of the requirement that these landmarks accurately represent the underlying shape, such a method also requires the identification of correspondences between these points across every image in the training set. In order to learn the shape, we must know which parts of the various training shapes are the "same" part. This 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. In three dimensions (3D), manual selection of these landmarks is a slow and tedious process. An automatic mechanism for producing these correspondences is therefore desired. In addition, an automatic method can remove much of the observer dependence that limits the manual approach.

This paper presents a procedure for automatically generating statistical models of biological shapes from a set of training images. The procedure produces a shape model of a particular structure, requiring user assistance only in the semi-automatic segmentation of a single template image. After this initial segmentation, the method automatically segments the training images and *simultaneously produces the surface correspondences*. The method is an extension of our earlier work,[5] which is improved both to produce more accurate surface representations of the training set, and to reduce the model bias towards the shape of individual instances in

the training set. In Section 2 we review similar work on the correspondence problem performed by other groups. Section 3 provides a detailed description of our method, and Section 4 describes our experiments creating a model of the lower cervical vertebrae (C6 and C7). We conclude with a discussion in Section 5.

2. PRIOR WORK

The correspondence problem has been approached from a wide range of directions. Davies *et al.*[6] take an information-theoretic approach, where they attempt to optimize a minimum description length (MDL) cost function of the landmark positions used to create the model. They argue that the best model for such a task is the simplest and most general one available. They search for a parameterization of the surface that minimizes the information required to encode the model and the training set using the model.

Lorenz & Krahnstöver[7] tackle the problem by "coating" vertices from a sample mesh onto a set of training meshes produced from segmentation of the training image set. Given a set of training meshes, they choose one to be a template, and generate an affine transformation that aligns the template mesh to each training mesh. After this rigid alignment, they use an active surface technique described by McInerney & Terzopoulos[8] to bring the landmarks onto the surface of the target shape. In a similar vein, Paulsen & Hilger[9] deform a template mesh, attempting to find a deformation field that maximizes an *a posteriori* probability measure.

The most similar method to that presented in this paper is work by Frangi, Rueckert, & colleagues.[10] Their scheme performs nonrigid registration on a set of previously segmented training shapes. In addition, their work on active deformation models (ADMs)[11] foreshadows the approach of this paper. Within the ADM framework, a dense set of correspondences (a deformation field) between two images is established. They suggest using such a deformation field to transfer a surface representation into the images in a training set. They produce a small model using this method, but avoid a detailed evaluation. This paper seeks to explore further the possibility of leveraging image registration techniques to build shape models.

3. MODEL GENERATION PROCESS

In our work, we aim to construct a statistical model of the surface of an object given a set of 3D images containing instances of the object. The input is the set of images with no pre-processing or prior segmentation required. The method we developed is semi-automatic, requiring the initial segmentation of a template, followed by the fully automatic segmentation of the training set and the statistical model creation. In addition, our procedure obtains the surface representations and the correspondences between these surfaces at the same time. This provides a significant gain over the methods previously described, many of which require a prior segmentation of every image in the training set, followed by a separate correspondence step.

Given the set of N training images, our generation technique proceeds with the following seven steps:

1. Creation of a population average template image from the training images.
2. Segmentation of the structure of interest from this image.
3. Generation of a template surface mesh from this segmentation.
4. Intensity-based nonrigid registration of the mean image to each image in the training set.
5. Warping of the template mesh using the transformations produced by the registration, resulting in surface meshes for each training example.
6. Mesh refinement using an active surface technique to increase the accuracy of the resulting meshes.
7. Alignment of the training meshes followed by a statistical analysis to produce the model.

The process is depicted as a block diagram in Fig. 1 and the steps are described in more detail below.

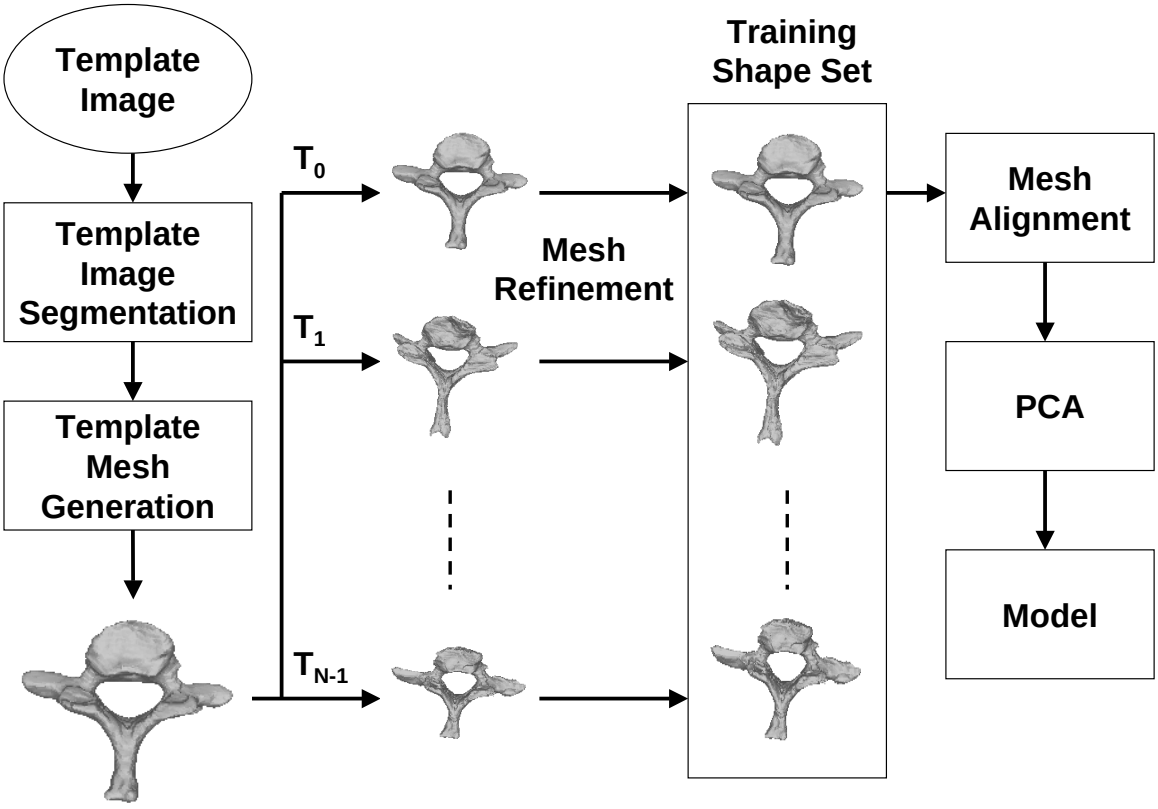

Figure 1. Model creation process. In the first stage the template image is created as described below. This image is then segmented and a template mesh is produced. This mesh is warped using a set of nonrigid transformations produced by intensity-based nonrigid registration. These meshes are then refined using a geodesic active surface model. The set of training meshes are finally aligned and principal component analysis is performed to produce the model.

3.1. Computing the Template Image

The first step in constructing the model is to create a template image. Because we do not want to bias the model structure towards a specific member of the training set, we create a population average grayscale image to serve as the template, through an iterative process using all of the training set images. We first select an arbitrary image from the N training set images as the initial template. We then register this template to each image in the training set, producing N nonrigid transformations. From these, we compute a mean transformation, and warp the initial image using the mean transformation to produce a new template image. We then register each training set image to this template, warp each image using the corresponding transformation, and compute average grayscale values at each pixel across the warped training set images. The resulting image is the output template for this iteration, which then becomes the input for the next iteration. This process is repeated for a chosen number of iterations, or until no change in the mean image is produced. The template image creation algorithm is given in pseudo-code in Fig. 2.

3.2. Template Shape Segmentation and Mesh Creation

To create the initial surface representation, we interactively segment the structure of interest from the mean image. This segmentation is performed using a method known as "live-wire"[12] or "intelligent scissors,"[13] which adaptively detects the object boundary and aids the user. By using this method, the boundary of the structure is described in each slice, allowing the production of a label image in which each pixel is labeled as either inside or outside the structure.

Next the marching cubes algorithm is utilized with this label image as input, to produce a triangulated mesh representation of the surface. We then decimate this mesh to the desired resolution. These operations were

1	Set initial template $\mathbf{I}_0$ = arbitrary training image
2	Repeat for $t = 0...N_{ITERS}$
3	Register $\mathbf{I}_t$ to each image in the training set, producing transformations $\mathbf{T}_1...\mathbf{T}_N$
4	Compute average transformation $\bar{\mathbf{T}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{T}_n$
5	Set temporary template $\tilde{\mathbf{I}}_t = \bar{\mathbf{T}}(\mathbf{I}_t)$
6	Register each training image to $\tilde{\mathbf{I}}_t$
7	Compute $\mathbf{I}_{t+1}$ = grayscale mean of transformed training images

Figure 2. Iterative algorithm to compute the mean image that will be segmented to produce a template mesh.

performed using the Visualization Toolkit (VTK) software,[14] which provides a variety of mesh manipulation operations.

In our process, these steps of mean image creation, segmentation from this image and production of a template mesh are performed *only once*. Once we have this mean image and the mesh representing the object in this image, the rest of the process if fully automated. The resolution of this template mesh will become the resolution for each training shape and for the final resulting model.

3.3. Registration of Mean Image to Training Set Images

At this point, we wish to find transformations that will allow us to warp the template mesh to the shapes given in each of the training set images. We note that an *image* transformation mapping the mean image to a given training set image provides a function that can then map the mesh *vertices*. We compute this transformation using an initial affine registration of the images, followed by an intensity-based nonrigid registration developed by Rueckert *et al.*[15] We use an independent implementation of this algorithm[16,17] to compute transformations between the mean image and each training set image.

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 of each other, and the transformed location of a source point (x, y, z) in the target image is determined by:

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

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:

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

and

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

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. For this measure, we use normalized mutual information (NMI).[18]

This step of the model generation process produces a dense set of correspondences mapping points in the mean image to points in the training image. The warp can be thought of as a giant lookup table, specifying where in the training image to look for a pixel corresponding to a given pixel in the mean image. We compute a nonrigid warp from the mean image to every image in the training set, and for training image i, we call the resulting warp $\mathbf{T}_i$.

3.4. Template Mesh Warping

We can now use these transformations to propagate the mesh vertices to each training shape. For each template vertex point $v_n = (x_n, y_n, z_n)$ in the template mesh, we can compute the location of the corresponding point in mesh i using $v_n^i = (x_n^i, y_n^i, z_n^i) = \mathbf{T}_i(x_n, y_n, z_n)$.

This process of propagating the points from the template mesh to create a mesh corresponding to each training shape is shown in Fig. 3. $\mathbf{T}_i$ represents the nonrigid warp transformation corresponding to training image i. After this step, we have a template mesh that represents the surface of the structure of interest for each training shape, given in the local coordinate system of the corresponding training image. Ideally, this mesh corresponds exactly to the surface in the image. However, because the registration is often inexact, we follow this step with a mesh refinement step.

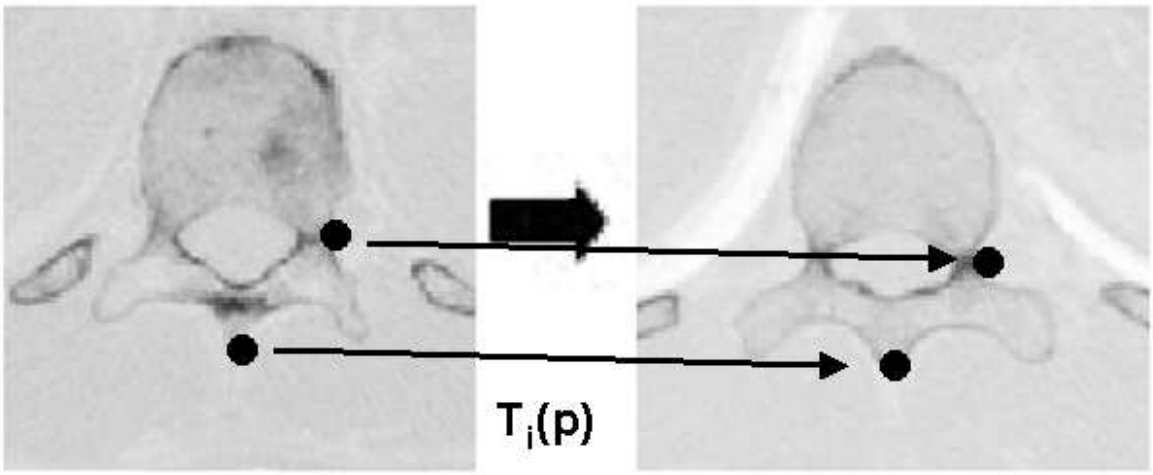

Figure 3. Propagation of correspondences. *Left:* template image, *right:* training image i. To propagate the landmarks from the template mesh to the training meshes, we transform each vertex using the nonrigid spline-based warp produced by the registration step. The resulting vertices should lie near or on the structure boundary in the corresponding training image.

3.5. Training Mesh Refinement

At this point, we refine the meshes produced in the previous step, hoping to increase the accuracy of the surface representation. We accomplish this by allowing each vertex on the mesh to adjust its position closer to the points of high gradient in the image. These points likely represent the boundary between the object of interest and the background.

The mesh refinement step involves treating the mesh surface as an active surface, using the geodesic-based formulation given by Caselles *et al.*[19] This formulation involves an update of the positions of the vertex locations in order to optimize a function that trades off a term minimizing the local curvature of the shape (maintaining its smoothness) with a term maximizing the gradient magnitude at the final location (improving alignment of the mesh with the object boundary). We iteratively update the vertex position until the shape converges to an equilibrium point. The update for vertex v at iteration t is given by

$$v_t = v_{t-1} + \alpha \left[g(\mathbf{I})\kappa - (\nabla g \cdot \vec{\mathcal{N}}) \right] \vec{\mathcal{N}} \tag{4}$$

where κ and $\vec{\mathcal{N}}$ are, respectively, the estimated Euclidean curvature and surface normal of the mesh at vertex v, and $g(\mathbf{I})$ is a function of the image gradient at the vertex position that drives the vertex towards higher gradient magnitudes. The parameter α is an update rate that determines how aggressively we update the vertex position. There are two terms at work here. Note that the κ term seeks to decrease the mesh curvature, while the ∇g term seeks to move toward higher gradient magnitude.

At convergence, a proper balance between these two forces is struck, and we therefore hope to have produced a final mesh that more accurately represents the underlying surface. As an additional advantage of performing this refinement step, we also obtain a smoother surface.

3.6. Statistical Analysis and Model Creation

We now have a full set of N surface meshes, with explicit correspondences between the vertices in each mesh. The next step is to align these surfaces into a common reference frame for analysis. The meshes are aligned with respect to rotation, translation, and scaling using a form of generalized Procrustes analysis (GPA).[20] This removes view and size dependent considerations to isolate variations in shape alone, which is the goal of our analysis.

Although we have already found an affine transformation that aligns the images in Section 3.3, an additional affine alignment of the surface meshes is needed for two reasons. First, the initial affine registration optimized a function over the *entire image*, and may have been slightly inaccurate in the local object neighborhood. However, because an accurate correspondence of the *surfaces* is required, the alignment may benefit from an operation that focuses on just this area. The second reason for computing a new affine alignment is that the nonrigid transformation may have introduced a small affine offset that was not captured in the initial affine registration. For both of these reasons, we choose to recompute the alignment at this point.

At this point, we are finally ready to perform the statistical analysis as in Cootes *et al.*[1] Because the structure of the mesh (i.e., connectivity) is fully determined by that of the template, we only need to represent the vertex locations, which vary across the instances. Vertex locations are represented as a vector $v_i = (x_1, y_1, z_1, x_2, y_2, z_2, ..., x_n, y_n, z_n)$. The training vectors create a cloud of points in $3n$ dimensions, which are modeled as a multivariate Gaussian distribution. We can therefore compute the mean of the distribution (centroid of the point cloud) as

$$\bar{\mathbf{v}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{v}_i, \tag{5}$$

and the covariance is given by

$$\mathbf{C} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{v}_i - \bar{\mathbf{v}})(\mathbf{v}_i - \bar{\mathbf{v}})^T. \tag{6}$$

Our goal at this point is to construct a method for approximating any instance of this shape using only a small number of parameters. In order to accomplish this, we determine the principal components of the Gaussian distribution, which represent the axes that contain the most variation in the set. This principal component analysis (PCA) produces modes of variation given by the eigenvectors of the covariance matrix. These are vectors $\mathbf{p}_k$ such that for some eigenvalue λ_k, we have

$$\mathbf{C}\mathbf{p}_k = \lambda_k\mathbf{p}_k. \tag{7}$$

By creating a matrix $\mathbf{P}$ with the m eigenvectors corresponding to the largest eigenvalues as the matrix columns, we can now compactly approximate each training instance using the mean these m modes of variation as

$$\mathbf{v} \approx \bar{\mathbf{v}} + \mathbf{P}\mathbf{b}, \tag{8}$$

for some set of mode coefficients $\mathbf{b}$.

Armed with this approximation method for the training instance, we generalize this model to describe the whole range of instances of the shape in question. In order for the model to work, it must satisfy two properties. The first is that any viable instance can be represented by some $\mathbf{b}$, up to a desired accuracy. Secondly, we hope

that any vector **b**, within some defined set, generates a viable instance of the shape using Eq. 8. To determine such a set of permissable **b** vectors, we will compute the variance of each component of **b**, and restrict that component to lie within three standard deviations above and below the mean.

4. EXPERIMENTS AND RESULTS

To test the method described in the previous section, we created a shape model of the lower cervical vertebrae, including training samples consisting of eleven images of the C7 vertebra, and five of the C6 vertebra. The slice thickness for each image is 1.25 mm, and the in-slice resolution varies between 0.5–1.0 mm. These images were first manually cropped to include the vertebra in question and parts of the vertebrae above and below, allowing easier isolation of the correct vertebra.

4.1. Training Shape Mesh Accuracy

In this process, the primary determinant of model accuracy is the quality of the nonrigid image registrations. If the surfaces of the structure of interest in each image can be put into one-to-one correspondence, and the calculated registrations are perfect (i.e., they correctly capture the underlying correspondences), then the transformed vertices will lie exactly on the surface of the training shape. However, because the registrations are not perfect, we perform the mesh refinement step, as discussed above, which should further improve the accuracy of our training set meshes.

To quantitatively determine this accuracy, we manually segment the surface of interest from each training image, and perform marching cubes on this segmentation to obtain a ground truth mesh (as in Section 3.2). The automatically generated mesh is now compared to the ground truth by finding the distance between the respective surfaces, which is considered the mesh error. This distance is sampled across the surface for a number of samples proportional to the number of vertices in the highest resolution mesh.

After the registration step, we achieve a mean surface-to-surface distance of 0.7 mm across the entire training set. Following the mesh refinement step, we are able to reduce this distance down to 0.6 mm, for an 11% improvement in mean error. The mean error before and after the refinement step is shown in Fig. 4. In addition, a confidence interval is shown, indicating an error that 90% of the distance samples are below. We note that the mean errors are on the order of the in-slice resolution and generally less than the slice thickness. These errors are therefore less than or equal to one pixel.

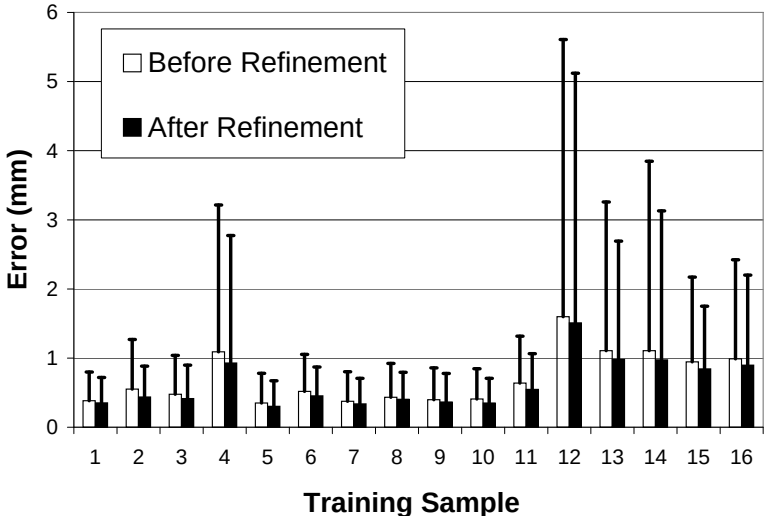

Figure 4. Mean model approximation error for each mesh in the training set. This error is the sampled distance between the automatically generated surface and the manually generated one. The bar heights indicate the mean error and the lines on top indicate the 90% confidence intervals, or the values that 90% of the distance samples are below.

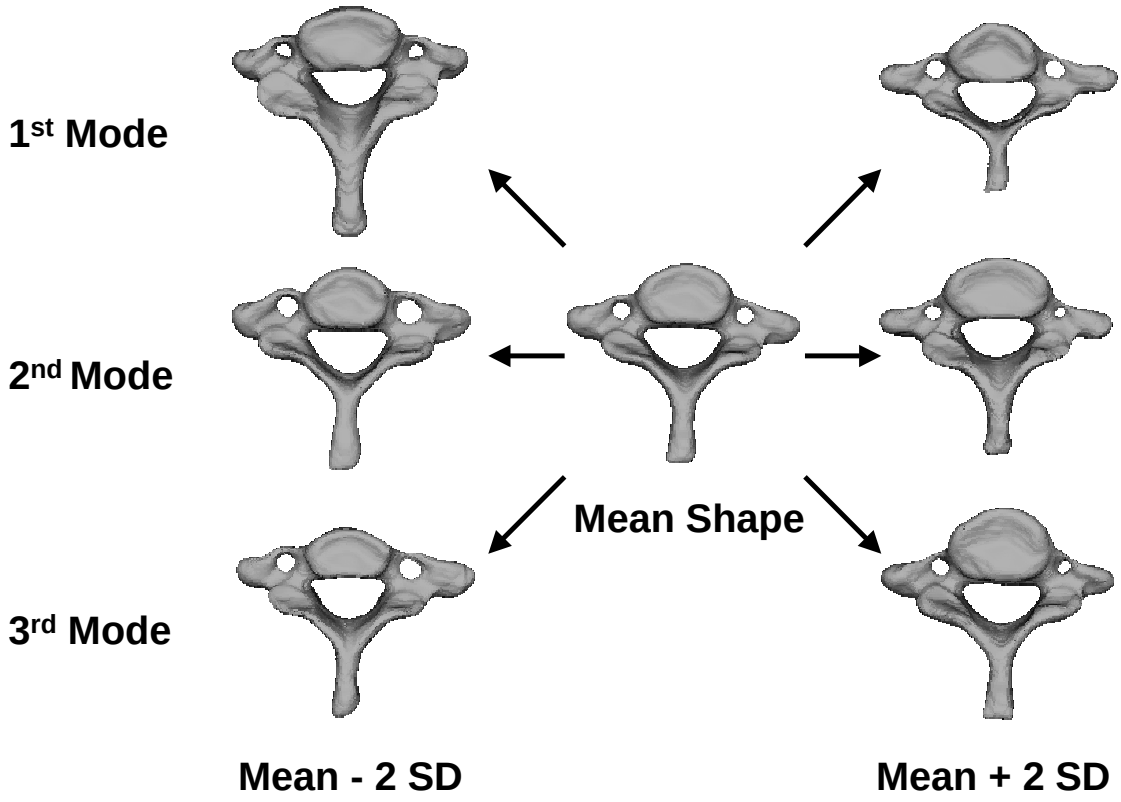

Figure 5. Instances of the cervical vertebra model. The first three modes of variation are shown in the rows, where the left shape in each pair shows the mean shape minus two standard deviations, and the right one shows the mean plus two SD. The modes of variation are generally symmetric and represent variations that are relatively intuitive.

4.2. Model Evaluation

After obtaining the aligned meshes from the training set images, we aligned them, computed the PCA, and created a statistical shape model. A few example instances of the model are shown in Fig. 5, where the first three modes in the model were varied across two standard deviations. Each row in the figure represents a mode of variation, and the surface to the left in each pair shows the mean shape minus two standard deviations (SD) in the given mode, while the one on the right shows the mean plus two standard deviations. In addition, the mean surface appears in the center.

From a quick inspection of the model instances, we see some intuitively nice results. First we note that the variations are generally symmetric, and correspond to the relative widths, lengths, etc. of the various parts of the shape. The first mode seems to represent length of the spinous process and the thickness of the processes in general. As another example, the third mode appears to describe the relative size of the vertebral body compared to the rest of the shape. We also note that there is no discernable twisting effect among the modes, which would indicate that the correspondences are incorrect.

Next we quantitatively analyzed the quality of this model. There are two factors that contribute to this. The first is the compactness of the model, or the ability to express the variation in the training set with as little information as possible. As a second consideration, we would like the model to represent new instances of the object that were not in the training set as accurately as possible.

To quantitatively analyze the ability of the model to represent the variation in the shape compactly, we look at the amount of variation in the training set explained by each mode. Ideally, if the actual variation in the shape is relatively simple and consistent, then a few degrees of freedom should be able to explain most of the

variation. Fig. 6 shows that 85% of the variation can be accounted for using only the first eight modes. While the compactness of this model is certainly meaningful, we would like to see a steeper slope in Fig. 6. It is likely that we would be able to represent the variation of the training with a smaller fraction of the available modes if we had a larger training set, because we would see more of the variation in the population.

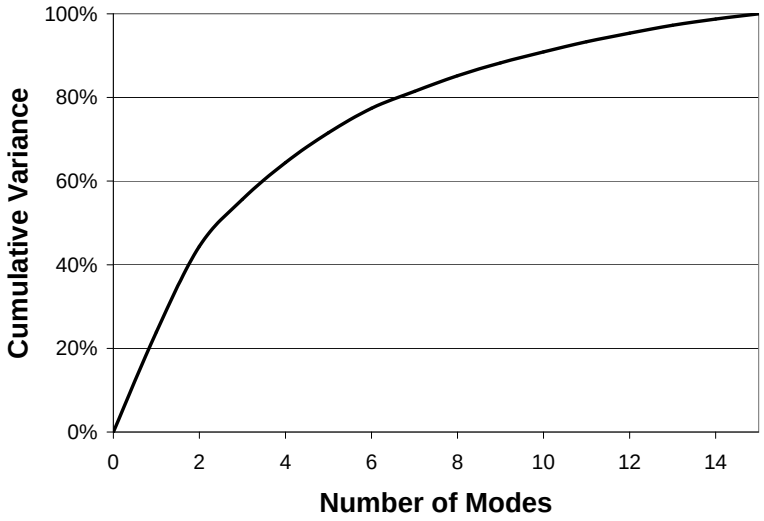

Figure 6. Cumulative variance explained by the first n modes of variation of the shape model. The more variance that can be explained by a given number of modes, the more compact the model representation of the shape.

To test the generalization ability of the models produced by this process, we next performed a leave-one-out cross-validation study. In this experiment, we created a model using all of the training samples except one as the input, and attempted to reconstruct the left-out sample using the resulting model.

In order to reconstruct a particular mesh from a given model, we use an iterative reconstruction approach. In the first step, we perform a rigid (Procrustes) alignment between the input mesh and the current estimate (initially the model mean). In the second step, we calculate the error between these two meshes, and find the least squares model coefficients to minimize the error. This process is repeated until alignment and least squares minimization produce no improvement in reconstruction accuracy.

The results of this experiment confirm our suspicion that our training set is not large enough. The mesh error for various numbers of modes, averaged across all training sample reconstructions, is shown in Fig. 7. We first note that by merely aligning the mean shape to the left-out surface, we are able to achieve a mean error of 0.9 mm and a 90% confidence interval of 1.9 mm. This is somewhat surprising, and indicates that the total variation of the shape is not very large. Using all 14 modes, however, we are only able to decrease the error down to a mean of 0.6 mm and confidence interval of 1.3 mm. We see a clear improvement by using the variation modes, but in general these models do not represent enough of the true variation to produce near perfect reconstructions of other instances of the shape.

5. DISCUSSION AND CONCLUSION

In this paper, we have presented a procedure to automatically generate a statistical shape model that requires only a single user segmentation. Such a process scales to a training set of any size, and it works for any shape for which we can find accurate and precise image registrations. As mentioned above, the quality of these registrations is one accuracy bottleneck, and continuing research in this area will increase the usefulness and trustworthiness of the models derived through this method. In addition, better image resolution and more accurate initial segmentation methods should also improve the final results.

Our results on a set of 16 images of the lower cervical vertebrae (C6 and C7) demonstrated that accurate surface representations (with a mean surface error of 0.6 mm) can be achieved by warping a template mesh

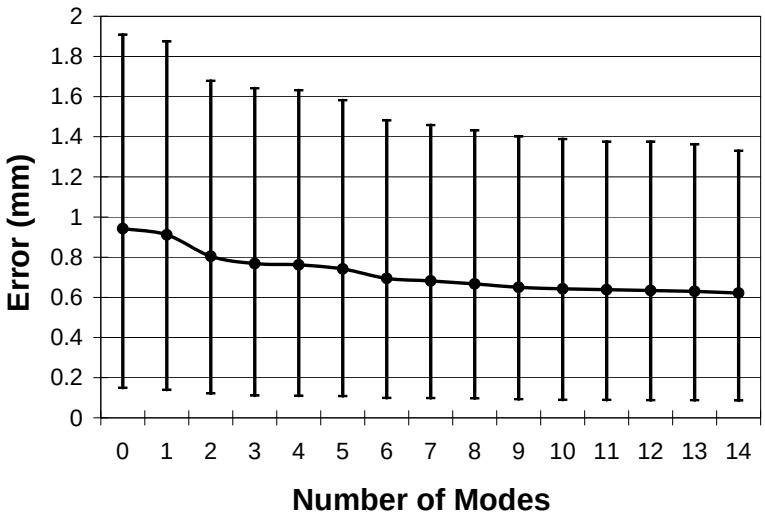

Figure 7. Results of the leave-one-out experiments. The error for reconstructing the left-out shape using the generated model is shown for the various numbers of modes. We performed this experiment by leaving out each training image, and averaged the errors across all resulting models.

according to registration transformations followed by an active surface refinement step. These mean errors are smaller than the in-slice pixel resolution, and thus may be near the limits of achievable accuracy. There is still room for improvement, however, as the confidence intervals in Fig. 4 could be lowered. These points of larger error indicate the existence of areas of local inaccuracy in the registrations.

The results achieved using our method are comparable to those of Kaus *et al.*,[21] who used images with 2.0 mm slice thickness, and achieved a training mesh accuracy of 0.8 mm. Their results, however, required a prior segmentation of every training shape. Our method therefore produces a huge saving in interactive time with no apparent loss of accuracy.

In our previous work[5] we used a training set sample image to produce the template mesh. In addition, there was no mesh refinement step, which was used in the newer process to produce an additional improvement in accuracy. These two improvements decreased the mean surface distance for the training meshes from 1.0 mm to 0.6 mm. The improvement is therefore theoretically more sound, because it does not bias the resulting registrations toward any single member of the training set. In addition, it is easier in practice (no need to choose a "good" template from the training set), and, finally, it also produces better results than our previous technique.

As described above, our relatively small training set reduces the quality of our model by limiting the range of variation seen in the model learning process. We hope that as the training set grows to 50 or 100 instances, there comes a point where adding more instances does not increase the number of salient modes (modes with variance above some noise threshold). Such modes should then represent the true degrees of freedom in the shape of interest.

The primary contributions of this paper are the following: 1) the creation and use as a template of a mean grayscale image by the averaging of nonrigid deformations, 2) the simultaneous computation of training surfaces and correspondences through nonrigid deformations, and 3) a post-transformation refinement stage where the mesh evolves as a geodesic active surface.

While the quality of our resulting models appear to be comparable to those of other methods in the literature, and have an accuracy that is comparable to the image resolution, the true test of these models will be in their usefulness in the wide range of image processing tasks that can benefit from them. This technique should work for many biological shapes, and is general enough to provide many opportunities for future use and extensions.

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), pp. 38–59, 1995.
2. Y. Wang and L. H. Staib, "Boundary finding with correspondence using statistical shape models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 338–345, (Santa Barbara, CA), June 1998.
3. S. Benameur, M. Mignotte, S. Parent, H. Labelle, W. Skalli, and J. D. Guise, "3D/2D registration and segmentation of scoliotic vertebrae using statistical models," *Computerized Medical Imaging and Graphics* **27**(5), pp. 321–337, 2003.
4. M. Fleute, S. Lavallée, and L. Desbat, "Integrated approach for matching statistical shape models with intra-operative 2d and 3d data," in *MICCAI '02: Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part II*, pp. 364–372, Springer-Verlag, 2002.
5. G. Heitz, T. Rohlfing, and C. R. Maurer, Jr., "Automatic generation of shape models using nonrigid registration with a single segmented template mesh," in *Proceedings 9th International Fall Workshop, Vision, Modeling and Visualization (VMV 2004)*, B. Girod, M. Magnor, and H.-P. Seidel, eds., pp. 73–80, Akademische Verlagsgesellschaft, Berlin, 2004.
6. 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*, *LNCS* **2082**, pp. 50–63, Springer-Verlag, 2001.
7. C. Lorenz and N. Krahnstöver, "Generation of point-based 3D statistical shape models for anatomical objects," *Computer Vision and Image Understanding* **77**(2), pp. 175–191, 2000.
8. T. McInerney and D. Terzopoulos, "Deformable models in medical images analysis: a survey," *Medical Image Analysis* **1**(2), pp. 91–108, 1996.
9. R. R. Paulsen and K. B. Hilger, "Shape modelling using Markov random field restoration of point correspondences," in *Information Processing in Medical Imaging*, *LNCS* **2732**, pp. 1–12, Springer-Verlag, 2003.
10. 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*, *LNCS* **2082**, pp. 78–91, Springer-Verlag, 2001.
11. 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), pp. 1014–1025, 2003.
12. 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*, pp. 635–638, 1992.
13. E. N. Mortensen and W. A. Barrett, "Interactive segmentation with intelligent scissors," *Graphical Models and Image Processing* **60**(5), pp. 349–384, 1998.
14. W. Schroeder, K. M. Martin, and W. E. Lorensen, *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics (2nd ed.)*, Prentice-Hall, 1998.
15. D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, "Nonrigid registration using free-form deformations: Application to breast MR images," *IEEE Transactions on Medical Imaging* **18**(8), pp. 712–721, 1999.
16. T. Rohlfing and C. R. Maurer, Jr., "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), pp. 16–25, 2003.
17. T. Rohlfing, C. R. Maurer, Jr., D. A. Bluemke, and M. A. Jacobs, "Volume-preserving nonrigid registration of mr breast images using free-form deformation with an incompressibility constraint," *IEEE Transactions on Medical Imaging* **22**(6), pp. 730–741, 2003.
18. C. Studholme, D. L. G. Hill, and D. J. Hawkes, "An overlap invariant entropy measure of 3D medical image alignment," *Pattern Recognition* **32**(1), pp. 71–86, 1999.
19. V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision* **22**(1), pp. 61–79, 1997.
20. J. C. Gower, "Generalised Procrustes analysis," *Psychometrika* **40**, pp. 33–51, 1975.

21. 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), pp. 1005–1013, 2003.