Abstract

A novel method that allows the development of surface point-based three-dimensional statistical shape models is presented. The method can be applied to shapes of arbitrary topology. Given a set of medical objects, a statistical shape model can be obtained by Principal Component Analysis. This technique requires that a set of complex shaped objects is represented as a set of vectors that on the one hand uniquely determine the shapes of the objects and on the other hand are suitable for a statistical analysis. The correspondence between the vector components and the respective shape features has to be the same in order for all shape parameter vectors to be considered. We present a novel approach to the correspondence problem for complex three-dimensional objects. The underlying idea is to develop a template shape and to fit this template to all objects to be analyzed. The method is successfully applied to obtain a statistical shape model for the lumbar vertebrae. The obtained shape model is well suited to support image segmentation tasks.

1. Introduction

In recent years, a rapidly increasing portion of research in the field of medicalimage analysis has begun to focus on shape as an anatomical object property. Shape representations and shape models are being used in connection with: three-dimensional visualization of anatomical objects; segmentation of 3D medical images [1]; diagnosis (e.g., based on deformable atlases) [2]; surgical simulation [3]; motion analysis (e.g., of the heart) [4] and radiotherapy treatment planning [5]. The desired properties of shape representations and shape models depend largely on the medical application to be supported. With reference to shape, one has to distinguish between shape classes (e.g., heart-shaped), shape instances (the shape of a certain heart), shape properties (such as local curvature, extension), shape representations (e.g., bi-cubic-interpolation) and shape parameterization (e.g., the control points and the (u,v) surface coordinates of Bézier patches).

Shape representation and parameterization are crucial for computerized processing and the manipulation of shapes. They determine flexibility, processing speed, possible user interactions and other issues important in an application. The shape (instance) of a sphere-like object can be described at one extreme using the set of surface points or at the other extreme using the four parameters of center and radius. The identification of a ‘good’ shape representation usually requires minimizing the number of necessary parameters to represent a shape of given complexity. Depending on the application of the shape model, it may also be required that the shape representation and parameterization are meaningful to the user, e.g, to ease user interaction.

In the domain of medical applications, the capture of object variability is of special importance. A shape model for a given anatomical object should be able to resemble a large portion of the shape instances that can be found either in different individuals (inter-patient variability) or within the same individual (intra-patient variability). This effect can be achieved by a statistical analysis of a set of shape instances of the object of interest [6,7]. The parameter vector of the shape parameterization is interpreted as a random vector. The parameter vectors of a set of shape instances define a multivariate distribution that can be described by the infinite series of moments of the random vector. If the distribution is Gaussian, it is already determined by the first (mean) and second (covariance) moments. In the case of a non-Gaussian distribution the first and second moments can still be used as an approximation.

The one-to-one correspondence between the elements of parameter vectors for different shape instances is a crucial requirement for a statistical analysis. All vector elements with the same index must represent corresponding shape information. For 3D shapes, this requirement is in general non-trivial. Solutions, based on
spherical coordinates and a shape representation using Fourier coefficients, have been presented [8] but are restricted to one-connected objects. Approaches based on stacks of 2D contours [9,10] depend on the orientation of the contour planes and can give rise to difficulties in case of object branching. The work presented here is based on a representation of the object shape by a set of surface points. The point set results from a thinning process of the object surface in the voxel image. A triangulation of the point set defines a surface mesh and can be used to cover the surface area in-between the points. The thinning procedure performed on two objects will in general produce non-corresponding point sets. Therefore, the thinning procedure is performed only for one object instance and the resulting point set is used subsequently as a template to be “coated” onto the surfaces of the remaining object instances. The coating procedure preserves the required one-to-one correspondence. Applied to all object instances, the coating procedure results in a set of parameter vectors, each of dimension three times the number of surface points. The final shape model consists of the mean shape parameter vector and variation modes in the form of the eigenvectors of the covariance matrix. The model can serve as a low-parametric organ specific shape model and can be used, for example for segmentation or registration of medical images. We used abdominal CT scans of lumbar vertebrae for a first application of the model generation procedure. The obtained model is based on 31 object instances and contains roughly 600 surface points and 1200 triangles.

One application of the vertebra model is the introduction of shape information into a vertebra segmentation procedure. The power of shape model guided segmentation has in previous work been demonstrated for the two dimensional case [6,7]. The segmentation of vertebrae in images from Computer Tomography (CT) and Computer Tomographic Angiography (CTA) is of importance e.g. in intervention planning procedures. The segmentation of a single vertebra in CT/CTA images is difficult due to the close connections between vertebrae in the spinal column. In addition, connections to contrasted vessels can potentially occur in the case of the CTA. These problems can to some extent be solved by means of morphological operations [11]. However, it is desirable to achieve solutions that are more robust and independent of the image resolution. The next section provides an overview of the model creation scheme. Section 3 describes the creation of the shape template and Section 4 describes the coating procedure.

The statistical analysis and the application of our method are presented in section 5 and 6, and results are given in section 7.

2. Shape-Model generation scheme

An overview of the shape-model generation scheme is given in figure 1. Input for the procedure is a set of 3D images. We assume the images are segmented with respect to the object of interest.

The first step is the definition of an object-related coordinate system. In case of the vertebrae, we chose the center of mass as the origin and the main axes of the second momentum matrix of the binarized objects as the coordinate axes. It turns out, that these axis are roughly aligned along the patient left/right, anterior/posterior and head/foot axes.

The second step is a re-sampling of the voxel images along the object coordinate axis. This step results in isotropic voxel images of the objects with predefined resolution, which are the basis for all further processing.

The third step is the generation of a point-distribution shape template. This template can be based on one of the object instances or on a derived object, such as the sum of all re-sampled input objects (e.g., as the result of a logical union). This step contains a surface curvature guided thinning and a triangulation procedure and is performed only once.

![Figure 1. Shape model generation scheme. Input of the procedure is a set of binary 3D objects. The surface point set of one of the objects is being thinned out and triangulated. The object shape is represented by a set of points (P1) and a face list (F) as a result of the triangulation. (P1,F) serves as a shape template for the rest of the objects. Each of the other objects are ‘coated’ with the template (P1,F). A statistical analysis of the point sets Pi can be used to create a statistical shape model of the set of objects.](image-url)

In step four, the shape template is “coated” to all remaining object instances. First, a set of landmarks is defined interactively on the template object and the object to be coated. The template is then deformed such that corresponding landmarks are in alignment. As a result, the template roughly matches the object to be coated.
Nevertheless, in general the template points will not be located on the object surface. To achieve this, the landmark based deformation is followed by a mesh relaxation that on the one hand, draws the template points to the object surface and, on the other hand, tries to retain mesh properties such as distance relations between neighboring surface points. For the relaxation, we use a spring-mass model based on the surface point set and the connecting edges, as defined by the triangulation.

*Step five* of the model generation scheme contains the statistical analysis of the shape parameter vectors being produced in step four. The mean shape is calculated as well as the eigenvectors and eigenvalues of the covariance matrix of the set of shape vectors.

### 3. Generation of a 3D point distribution template

One of the objectives of a shape model is the reduction of parameters needed to describe a shape instance. In a first step towards this goal, the set of surface voxels of an object in a volume data set is thinned out to obtain a lower number of surface points. The method and parameters of this thinning process determine the resolution of the shape representation. The thinning is guided by the local surface curvature because it is desirable that the surface point density at a certain surface patch correlates with the local “degree of detail” of the object. In addition, the selection of the actual points can be guided by the local surface curvature.

After the thinning procedure, a triangulation of the resulting point set is performed. A triangulation is necessary because it holds the information about the surface that the point distribution actually represents. Furthermore, the triangulation allows fast visualization of the shape model.

The thinning procedure is described in section 3.1 followed by the description of the surface triangulation in section 3.2.

#### 3.1. Thinning

For the generation of a point-distribution model of a surface, it is necessary to select a sub-set of surface voxels as *surface points*. This thinning algorithm works as follows: Initially all surface voxels are candidates and can be selected by the thinning algorithm. The algorithm iteratively selects surface points from the candidate list to be included in the point distribution model and eliminates all candidates that are within a certain neighborhood of the selected point. This selection and elimination of surface voxel is repeated until no candidates are left. The thinning is schematically shown in figure 2. Details about the procedure by which the surface voxels are eliminated and selected from the candidate set are two important aspects of the thinning algorithm. The selection of surface voxels can either be at random or curvature guided. It is of advantage to select voxels from those parts of the surface where the local curvature is high because those points lie on edges and tips of the surface. In order to achieve this, the curvature of each surface voxel has to be determined, which can be achieved by approximating local surface patches by second order polynomials [12]. Once a mean curvature $c = \sqrt{k_1^2 + k_2^2}$, where $k_1$ and $k_2$ are the main curvatures, is assigned to each surface voxel, the thinning procedure becomes curvature dependent by always choosing surface voxels with the largest mean curvature from the candidate set. The neighborhood around a selected voxel is defined as follows: Given a selected surface voxel and a thinning radius $r$, all surface voxels that have a surface distance less than $r$ are eliminated around the voxel. The thinning radius defines the resolution of the point distribution and can either be a constant or dependent on the surface curvature. If the thinning radius is dependent on the surface curvature, it can be chosen to be small in regions where the surface curvature is high. Making the thinning radius variable leads to an anisotropic point distribution. The thinning radius could be related linearly to the surface curvature $c$, however, our finding is that a stronger pronunciation of small thinning radii can better reproduce surface details [13].

![Figure 2. A two-dimensional schematic visualization of the thinning process.](image)

#### 3.2. Delaunay Triangulation

To obtain a complete parameterization of the object surface and to ease visualization, the surface point set resulting from the thinning process is triangulated. To do this, a Voronoi graph of the surface point set is used to obtain the Delaunay triangulation [14]. An elementary multiple source shortest path graph algorithm is used to simultaneously expand regions of surface voxels around each surface point selected in the thinning process. As the regions increase in size, the borders of the regions start to collide. The points where the fringes collide represent the...
points that have equal surface distance from two or more surface points and represent approximately the edges and nodes of the Voronoi graph on the surface of the object.

The points where more than two regions collide (i.e., the nodes of the Voronoi graph), represent the faces of the triangulation. Figure 3 shows three triangulations of a cube. Figure 3 (a) shows clearly that a random selection of surface voxels is ignorant of edges and corners while the curvature dependent triangulation manages to capture the shape of the cube while using as little as 26 vertices for the triangulation (figure 3 (b)).

**Figure 3.** Curvature guided triangulation of a cube. Left: Isotropic unguided triangulation. Middle: Isotropic triangulation, point selection curvature guided. Right: Point density and point selection curvature guided.

4. Adaptation of the template to a set of objects

For a statistical shape analysis, it is necessary to obtain parameter vectors for a set of objects. The procedures described in the previous section resulted in a triangulation \( T = (V,F) \) with vertices

\[
V = \{p_1, p_2, p_3, ..., p_n\} \subset \mathbb{R}^3 \quad \text{and faces} \quad F \subset \mathbb{V}^3.
\]

The set of vertices constitutes the point distribution model. The set of vertices can be used to obtain a 3n-dimensional parameterization

\[
p = (p_{1x}, p_{1y}, p_{1z}, p_{2x}, p_{2y}, p_{2z}, ..., p_{nx}, p_{ny}, p_{nz}) \in \mathbb{R}^{3n}
\]

of the object. Given a set of object samples, performing a surface thinning and a triangulation of the objects would result in a set of parameter vectors. This set of parameter vectors cannot, however, be used for a statistical shape analysis because the parameter vectors would not correspond to each other. The number of surface points and triangles (and thus the dimension of the parameter vector) will in general be different for each individual object. In addition, the i-th surface point of one object will in general not correspond to the i-th surface point of another object. For a statistical analysis, however, it is necessary that a given component of a set of parameter vectors represents a well-defined surface property, for example a well-defined part of the surface in the case of a point distribution (see figure 4). To overcome this problem, we proposed to generate, for one object sample, a shape template and to “coat” this template onto the remaining set of object samples.

**Figure 4.** Correspondence problem of shape parameterizations. The number of parameters used to represent different shape instances as well as the type of information contained in a given parameter should be the same.

The coating should preserve the one-to-one correspondence of the components of the parameter vectors:

2. Deform the surface point distribution of the source object in such a way that corresponding landmarks of the source and destination object match. The surface points are now positioned close to the target surface, but generally not exactly on it.
3. Project the surface points onto the surface of the destination object.
4. Perform a mesh relaxation, that moves the surface points on the target object in such a way that the point distribution resembles, as much as possible, the source mesh, while keeping the points on the target surface.

4.1. Landmark-based mesh deformation

A user has to define pairs of landmarks \((p_i, p_j)\) in the source and the destination object space. This set of landmarks is used to obtain a mapping from the source object space to the destination object space. We used the scattered point thin-plate spline interpolation technique, which was introduced into the medical image processing domain by Bookstein [15]. The obtained spline function smoothly maps the source landmarks onto the destination (see figure 5). The idea is that points in the neighborhood of a landmark are moved in a similar way as the
landmark that matches the corresponding target landmark. Denoting $\mathbf{F}(\mathbf{P})$ as a coordinate in the source object space, $\mathbf{F}(\mathbf{P})$ as the transformed coordinate and $\mathbf{F}_i$ as the i-th landmark position, the resulting coordinate transformation can be written as \[ \mathbf{F}(\mathbf{P}) = \mathbf{F}_0 + \mathbf{A} \cdot \mathbf{P} + \sum_i \mathbf{W}_i \cdot U(|\mathbf{P} - \mathbf{F}_i|) \] \hspace{1cm} (1)

The first part of this equation is a translation and the second part an affine transformation (defined by a matrix $\mathbf{A}$). The third part sums all transformations of all landmarks and consists of the radial basis function $U(r)$ and the weight vectors $\mathbf{W}_i$.

The above transformation must map all landmarks from their source to their destination position $\mathbf{F}(\mathbf{P}_i) = \mathbf{S}_i$. These equations can be used to obtain the parameters in the above mapping. The appropriate radial basis functions, solving the thin-plate problem are the Duchon functions $U(r) = r^2 \ln r$ and $U(r) = r \ln r$ in the 2D and 3D case, respectively.

However, in principle, another function could be chosen, for example, the often used functions of type $U^H_i(r) = \left(|r - r_i|^2 + s_i^2\right)^\alpha$ with $\alpha > 0$ and $s_i$ being a “stiffness-radius”, introduced by Hardy [17].

We have implemented both Duchon and Hardy type radial basis functions. The Duchon-type function has a larger long-distance influence and seems to work better for our purpose, since the aim is to influence the position of many vertices with just a few landmarks. For an efficient user-guided development of a shape model it is desirable that the number of landmarks to be defined is as low as possible. It is not yet clear what the optimal number of landmarks is. In the case of the vertebra example, we used 15 landmarks. It is imaginable that some landmarks could be detected automatically, (e.g. based on surface curvature information in the binary image or based on gray-value curvature in the gray-value image [18]).

4.2. Projection of the deformed mesh onto the target surface

The landmark-based deformation leads to a rough adaptation of the template to the target object. However, the vertices of the triangulation are not necessarily located on the object surface. To achieve this, we use a distance map [19] to project every vertex to the nearest surface point (see figure 6). The projection might be omitted in the future, since the mesh relaxation step that follows the projection includes a surface term, pulling the vertices to the surface of the target object.

Figure 5. Top row: Based on a set of corresponding landmarks (left), a smooth deformation of the space (and the triangulation) can be achieved (right). The landmarks in the left figure are shown schematically, the actual landmarks are defined in 3D.

Bottom row: The triangulated template is deformed to fit onto a destination object based on 15 landmarks. For comparison, the rightmost image shows an independent triangulation of the destination object that has no correspondence to the other two. The deformed triangulation corresponds to the template, while resembling the shape of the destination object. The coloring of the triangles is chosen in order to show the correspondence between triangulations.

Figure 6. Projection of the vertices of the deformed triangulation onto the target surface.
4.3. Mesh relaxation

It can happen that the projection of the vertices to the destination surface causes folds of the triangular mesh. Also irregular large or small triangles or misplaced vertices may appear. We introduced, as a last step, a mesh-relaxation based on a mass-spring mode to “unfold” the surface and to regularize the mesh (see figure 7). Each vertex is associated with a certain mass and the triangle-edges with springs. Denoting the current distance between vertex i and j with \( d_{ij} \), the distance between vertex i and j in the template mesh with \( d_{ij}^0 \), the unit vector pointing from vertex i to j with \( \mathbf{u}_{ij} \), and the current distance vector between vertex i and the target surface with \( \mathbf{d}_{DM} \) (resulting from the distance map), the force acting on the i-th vertex can be written as:

\[
F_i = \sum_j k_1 (d_{ij} - d_{ij}^0) \mathbf{u}_{ij} + k_2 d_{DM}^r (\mathbf{R}_i)
\]

The equation of motion writes \( \mathbf{F}_i = \mathbf{m} \mathbf{a}_i + \gamma \mathbf{v}_i \), with \( \mathbf{a}_i \) being the acceleration, \( \mathbf{v}_i \) the velocity, and \( \gamma \) the damping coefficient. Folds of the mesh can be detected using the difference between the triangle normals and the surface normals of the binary target object. If a fold is detected, the surface force is switched off for the vertices of that particular triangle, to allow the necessary distance from the surface during unfolding.

The parameters \( k_1 \) and \( k_2 \) steer the interplay between surface and mesh force. For the time being, we chose them such that the surface force exceeds the mesh force.

5. Statistical Analysis

The output of the procedures described above, is a set of \( n \) shape parameter vectors \( p_i = (p_{i,1}, p_{i,2}, ..., p_{i,3N}) \in \mathbb{R}^{3N} \), with \( i = 1, 2, ..., n \). Each shape vector contains the coordinates of \( N \) surface points. All vectors are derived from one shape template and have thus the same dimension \( 3N \). The coordinates of the surface points are given in the object-related coordinate system as defined in section 2. No other normalization, such as the ‘Procrustes Analysis’ [7], has been applied. Therefore, scaling is expected to be part of the natural variability and will be part of the model variability. Given that the coating procedure worked properly, there is a one-to-one correspondence between the vector elements of a given index, in the sense that for all instances this vector element represents corresponding shape information. If the object instances used so far are representative for the class of objects under consideration, it should be possible to extract statistical information about the object variability. More specifically, we are interested in the mean shape vector

\[
\bar{p} = \frac{1}{n} \sum_{i=1}^{n} p_i
\]

and in the correlation between vector elements, as given by the empirical covariance matrix

\[
S = \frac{1}{n} \sum_{i=1}^{n} (p_i - \bar{p})(p_i - \bar{p})^T.
\]

If we write \( P \) for the \( 3N \times n \) matrix of centered parameter vectors, \( S \) can be written as \( S = PP^T \). The principal component analysis of \( S \) yields the eigenvalues \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{3N} \geq 0 \) and the corresponding eigenvectors \( q_i \), \( i = 1, 2, ..., 3N \). The eigenvectors corresponding to the largest eigenvalues describe the most significant modes of variation in the parameter space. Here, it is assumed, that a small number of modes account for a large portion of the total variability. If the number of shape samples \( n \) is smaller than the shape vector dimension \( 3N \), only the first \( n \) eigenvalues of \( S \) are non-zero. In this case instead of the \( 3N \times 3N \) eigenvalue problem, the associated \( n \times n \) eigenvalue problem of the implicit covariance matrix \( \tilde{S} = P^TP \) can be solved [20]. The first \( n \) eigenvalues and eigenvectors of \( S \) can be determined by the ones of \( \tilde{S} \):
The model allows to generate new shape samples by adding a weighted sum of modes to the mean parameter vector:

\[ p = \bar{p} + Qb \]

with \( Q = (q_1, ..., q_m), \ m \leq n \) being the matrix of the first \( m \) modes and \( b = (b_1, ..., b_m) \) the weight vector. This can be used to explore the object shape space with a low parametric representation of the object, for example during a segmentation procedure. The function

\[
\nu_{\sigma^2}(k) = \sum_{i=1}^{k} \frac{\sigma_i^2}{\sum_{j=1}^{N} \sigma_j^2}
\]

evaluated for the variances \( \sigma_i^2 \) of the coordinates of the surface points and for the eigenvalues \( \lambda_i \) of the covariance matrix can be used to determine the concentration of variability that has been achieved by the eigenmode analysis (see also figure 10).

6. Model application

The generated shape model can be used in order to support the segmentation of lumbar vertebrae in CT/CTA images. The first step is a manual rough initial positioning of the mean shape in the 3D image. If the mean orientation of the object in the patient coordinate system (left/right, anterior/posterior, head/foot) is known, the initial orientation can be selected automatically, since the relation between the patient coordinate system and the image voxel coordinate system is an attribute of the CT image data set. The next step is an automatic refinement of the rotational and translational parameters as well as an adaptation of the weights of the first most important eigenmodes in (6). For this, a standard downhill-simplex optimization procedure was used [21].

Since the object under consideration is bright (large gray-value) on dark background (small gray-value) we minimized an objective function that was based on the sum of first derivatives directed outwards along the triangular face normals. The first derivatives are approximated by finite differences in the tri-linearly interpolated voxel image and are calculated just-in-time.

The model adaptation scheme is depicted in figure 8.

7. Results

7.1. Model Generation

Segmented abdominal CT scans of lumbar vertebrae have been used for a first application of the model generation procedure. The model is based on 31 object instances (9xL1, 11xL2, 9xL3 and 2xL4) and contains roughly 600 surface points and 1200 triangles. For the landmark-based mesh deformation, 15 landmarks distributed over the vertebra have been used. The manual definition of the 15 landmarks took approximately 2 to 3 minutes per vertebra.

In figure 9, the magnitude of the eigenvalues and in figure 10 the concentration of variability in the first \( n \) eigenvalues according to (7) is shown. A variability of 90% is already captured by the first 10 eigenvectors. Figure 11 and 12 shows axial and sagittal views of the first two eigenmodes of the generated model. The shapes are generated according to (6). For illustration, the first and second entry of the weight vector are set to the natural standard deviation: \( b_{1,2} = \sqrt{\lambda_{1,2}} \), respectively.

![Figure 8. Shape model to image adaptation scheme. A manually defined initial position is the starting point for an optimization routine that adapts translational and rotational parameters as well as the weights of the first most important deformation modes.](image)

![Figure 9. Magnitude of eigenvalues.](image)
Figure 10. Portion of variability, captured by the first n parameters. Already the first 10 eigenvectors account for about 90% of the total variability. Note the logarithmic scale of the diagrams.

Figure 11. Axial view of first and second eigenmode of the generated model visualized by an overlay of the mean shape and two deformed shapes according to the first and sec. eigenvector of the covariance matrix: 
\[ p = \bar{p} \pm \sqrt{\lambda_i} \cdot q_i \]

Figure 12. Sagittal view of first and second eigenmode of the generated model visualized by an overlay of the mean shape and two deformed shapes according to the first and sec. eigenvector of the covariance matrix: 
\[ p = \bar{p} \pm \sqrt{\lambda_i} \cdot q_i \]

The first eigenmode is to some degree a size variation mode. This is due to the fact that size variation was not removed by a normalization procedure, but included in the natural variability of the anatomical object. Both modes are more or less symmetric about the mid-sagittal plane. Other modes show also asymmetric properties.

7.2. Model Application

Figure 13a,b shows the overlay of mesh cut-lines of a roughly positioned mean mesh onto orthogonal cuts of a 3D CTA image, containing an L2 vertebra, which was not part of the training set during model generation. Optimizing the weights of the first most important eigenmodes in (6), and the translational and rotational parameters, the model mesh further adapts itself to the vertebrae in the gray valued 3D image (figure 13c-e).
According to figure 10, about 80% of the total shape variability is captured by the first 6 eigenmodes. Figure 13 shows that the overall fit of the adapted model to the image data is quite reasonable. The imperfection of the adapted model that is visible in detail, is due to the limited number of deformation modes taken into account and the limited number of shape samples being available during model generation.

Nevertheless, the model is intended as a low parametric shape representation. It is not meant to follow the object shape in full detail. The computation time for shape adaptation results, such as shown in figure 13 amounts only to a few seconds on an SUN Ultra Sparc computer.

In order to translate the result of the mesh adaptation back to the 3D voxel image (since the voxel image is to be segmented), it is necessary to generate a voxel volume that corresponds to the object volume enclosed by the surface mesh. Figure 13(f) shows a surface rendering of such a voxel volume obtained from the mean mesh.

The described model generation and adaptation scheme is quite generic and applicable to various 3D imaging modalities and various anatomic objects. However, the definition of an appropriate objective function largely depends on the target object and modality.

8. Conclusions

In this article we presented a new approach for the generation of 3D statistical shape models. The method is valid for objects of arbitrary topology and is based on a triangulation of a thinned set of object surface points. From one object of a set of objects, a shape template is generated and adapted to the rest of the set. The adaptation (‘coating’) procedure consists of a rough landmark-based shape morphing and a mesh relaxation. This procedure results in a set of shape parameter vectors. From this set, the mean parameter vector and the covariance matrix, as well as the eigenvector and eigenvalues of the matrix, are calculated. The eigenvectors are interpreted as modes of shape variation. The method has been successfully applied to a set of 31 lumbar vertebrae. As expected, a large portion of the total shape variability is already captured within the first few eigenvectors. However, the number of objects used so far is very small and not representative in any sense. The model has been used as input for a simple automatic shape adaptation procedure, with automatic determination of translational and rotational parameters as well as variation mode weights.

Future directions include the generation of a shape model based on a larger set of objects and the application of the model in a segmentation or registration procedure for clinical applications. An improvement of the model generation procedure would be a reduction of the necessary number of landmarks to be identified interactively, e.g. by developing an automatic landmark detection scheme.

Acknowledgments

The authors would like to thank Prof. Dr. W. P. Th. M. Mali, Prof. Dr. B. C. Eikelboom and Dr. J. D.
The algorithm was implemented on an Easy Vision workstation from Philips Medical Systems and we would like to thank Easy Vision Modules Advanced Development, Philips Medical Systems, Best, for helpful discussions and their support.

References


