Computer Vision and Image Understanding **86,** 96–117 (2002)
doi:10.1006/cviu.2002.0969

Necklaces: Inhomogeneous and Point-Enhanced Deformable Models[1]

S. Ghebreab and A. W. M. Smeulders

Intelligent Sensory Information Systems Group, Informatics Institute, University of Amsterdam,
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
E-mail: ghebreab@science.uva.nl, smeulders@science.uva.nl

and

P. R. Pfluger

Department of Mathematics, Korteweg-De Vries Institute, University of Amsterdam,
Plantage Muidergracht 24, 1018 TV Amsterdam, The Netherlands
E-mail: pia@science.uva.nl

Received June 1, 2001; accepted July 17, 2002

In many advanced segmentation problems objects have inhomogeneous boundaries, hindering segmentation under uniform boundary assumption. We present a multifeature image segmentation method, called *necklaces*, that exploits local inhomogeneities to reduce the complexity of the segmentation problem. Multiple continuous boundary features, deduced from a set of training objects, are statistically analyzed and encoded into a deformable model. On the deformable model salient features are identified on the basis of the local differential geometric characteristics of the features, yielding a classification into point landmarks, curve landmarks, and sheet points. Salient features are exploited within a priority segmentation scheme that tries to find complete boundaries in an unknown image, first by landmarks and then by sheet points. The application of our method to segment vertebrae from CT data shows promising results despite their articulated morphology and despite the presence of interfering structures.

Key Words: multifeature object representation; landmark-based image segmentation; deformable models; shape classification.

---

[1] The authors thank Prof. Dr. Ir. F. Gerritsen from the Easy Vision Advanced Development, Philips Medical Systems, Best, for providing the CT images and the Easy Vision workstation. We are also thankful to Diana Bosch from the Academic Hospital of the Free University of Amsterdam for ground-truth segmentation.

1077-3142/02 $35.00

1. INTRODUCTION

Image segmentation, the partitioning of an image into distinct areas whose points can be characterized according to some specific property, is commonly performed on the basis of object boundaries. Image information that is associated with these boundaries is extracted locally and is then linked in one or another way by a computational method. When images contain complex ensembles of boundaries that are fractured, occluded, convoluted, or otherwise inhomogeneous, computation may be seriously hampered under the uniform boundary assumption. Segmentation under this premise only works well for a limited range of segmentation problems where objects are smooth and their edges are well defined along the entire boundary. It is therefore imperative to strive for segmentation that exploits inhomogeneities rather then apprehending them as obstacles. This is viable as inhomogeneities often form salient information and hence are suitable boundary landmarks for image segmentation.

The importance of boundary landmarks has been recognized in several works. Methods that profit from point landmarks are proposed in, e.g., [2, 17, 20, 21, 34]. In here, the user manually defines a set of point landmarks for computation to align it to the target object in the image. This is disadvantageous because manual definition is time-consuming and prone to failures. For this reason automatic landmark definition has been proposed in, e.g., [7, 8, 10, 15, 29, 33]. Point landmarks as well as curve landmarks are defined on geometrical or analytical grounds. Interactive detection is advocated in, e.g., [13, 27, 28], since automatic definition and detection of point landmarks are complex and prone to errors [26]. Interaction in this case means that either a region of interest or an approximate position of a specific landmark is given by the user to aid computation. We commemorate that existing methods define and employ landmarks manually, automatically, or interactively, the common denominator being the application of predefined features for landmark definition.

We conceive of landmark-based segmentation as a multifeature selection procedure aiming at automatically defining and gradually exploiting landmarks. In a multifeature approach object boundaries are defined using multiple image and shape features, often giving rise to a repertoire of landmarks. For example, when examining the bending along the heart's boundary in cardiac images, curvature feature values will give a peak at the lower tip of the outer wall. When, in addition, observing the edge structure along the heart's boundary, this might give a highlight at the position where the aorta enters the heart. The local deviations in multifeature observations can be seen as landmarks with differing degrees of freedom. We strive to learn such landmarks from a given set of training objects and to exploit them accordingly. The question addressed in this work is how to conveniently define landmarks and how to befittingly apply them for segmentation of volumetric image data.

The work presented in this paper is organized as follows. In Section 2 object boundary landmarks are discussed. Attention is devoted to object boundary representation, definition of landmarks, and their application. Section 3 focuses on landmark-based segmentation by necklaces. Necklaces are introduced in terms of model construction, model qualification, and model optimization. Experiments and results follow in Section 4. The paper concludes with a discussion in Section 5.

2. OBJECT BOUNDARY LANDMARKS

We discuss requirements for boundary landmarks, their definition and application. Before we do that we first briefly describe object representation, which plays an essential role in defining suitable landmarks.

2.1. Boundary Representation

In [14], the appearance of a two-dimensional object is learned from a repertoire of features. Multiple continuous features are extracted from M training images $I_m(\mathbf{x})$ for $m = 1, \ldots, M$ and $I_m : \mathbf{x} \in \Re^2 \rightarrow \Re$, for each of which the known segmentation is represented by smooth curves $\mathbf{S}_m(u)$, $\mathbf{S} : u \in \Re \rightarrow \Re^2$. For the mth learning example, the shape $\mathbf{S}_m(u)$, relates to the image at the points $I_m(\mathbf{S}_m(u))$. This is expressed in terms of N features derived from the shape as well as from the image. The mapping $\mathbf{F} : u \in \Re \rightarrow \mathcal{F}$ handles this, yielding *feature functions*,

$$\mathbf{F}_m(u) = \left[F_m^1(u), \ldots, F_m^N(u)\right]. \tag{1}$$

This way, features are integrally captured by space curves in the N-dimensional feature space $\mathcal{F}$, each dimension associated with one feature. This representation allows convenient localization of landmarks arising from multiple features of the object. The landmarks are typically maxima, minima, and zero-crossings of the space curves, identified at the level of derivatives by differential curve geometry.

Here, we take over the multifeature object representation in order to exploit landmarks that appear in feature functions for image segmentation. We do this for three-dimensional objects, extending two-dimensional concepts in [14] to three dimensions. In this case, training images $I_m(\mathbf{x})$ are three-dimensional; i.e., $I_m : \mathbf{x} \in \Re^3 \rightarrow \Re$. Object outlines are smooth surfaces represented by $\mathbf{S}_m(\mathbf{u})$, $\mathbf{S} : \mathbf{u} \in \Re^2 \rightarrow \Re^3$. The mapping $\mathbf{F} : \mathbf{u} \in \Re^2 \rightarrow \mathcal{F}$ yields feature functions

$$\mathbf{F}_m(\mathbf{u}) = \left[F_m^1(\mathbf{u}), \ldots, F_m^N(\mathbf{u})\right]. \tag{2}$$

Features are now conveniently captured by manifolds [22] in the N-dimensional feature space $\mathcal{F}$. In search for landmarks, we have to consider high curvature points in these manifolds. Differential geometric surface analysis then localizes landmarks.

We briefly summarize requirements for landmarks, conforming to the ones in [24].

- Landmarks should be *selective*. Landmarks that do not possess characteristics that discriminate them from other boundary points have no additional value for segmentation.
- Landmarks should be *sparse*. Landmarks that are abundantly present are less suitable for concise representation of the segmentation problem.
- Landmarks should be *detectable*. Landmarks the presence of which cannot be confirmed computationally may cause erroneous solutions.
- Landmarks should be *robust* in their appearance. Boundary landmarks that are inconsistent in their frequency of occurrence and that have highly variable appearance may hamper segmentation.

These requirements are general in nature; they hold for a large number of points on and off boundaries. We concentrate on boundary landmarks.

2.2. Definition of Landmarks

We take a differential geometric approach to find suitable landmarks from feature functions, reducing landmark definition to localization of surface landmarks. Surface landmarks are captured by the well-known first and second fundamental forms (see, e.g., [18, 31]). In this section we will discuss landmark definition for the case where the x, y, and z-coordinates of a surface are the only features to consider; i.e., $\mathbf{F} : \mathbf{u} \in \Re^2 \rightarrow \Re^3$.

For the surface $\mathbf{F}(\mathbf{u})$ the *first fundamental* form $I(\mathbf{u}, d\mathbf{u})$ measures the length of a small movement $d\mathbf{F}$ from the point $\mathbf{F}(\mathbf{u})$ in the direction given by $d\mathbf{u}$. The first fundamental form $I(\mathbf{u}, d\mathbf{u})$ is determined on the basis of the surface derivatives $\mathbf{F}_\mathbf{u}$ in the $\mathbf{u} = [u_1, u_2]^T$ direction. It is defined as

$$I(\mathbf{u}, d\mathbf{u}) = d\mathbf{u}^T \begin{bmatrix} \mathbf{F}_{u_1} \cdot \mathbf{F}_{u_1} & \mathbf{F}_{u_1} \cdot \mathbf{F}_{u_2} \\ \mathbf{F}_{u_2} \cdot \mathbf{F}_{u_1} & \mathbf{F}_{u_2} \cdot \mathbf{F}_{u_2} \end{bmatrix} d\mathbf{u}. \tag{3}$$

For the *second fundamental form* the unit normal vector is needed:

$$\mathbf{n}(\mathbf{u} \,|\, \mathbf{F}) = \frac{\mathbf{F}_{u_1} \times \mathbf{F}_{u_2}}{|\mathbf{F}_{u_1} \times \mathbf{F}_{u_2}|}. \tag{4}$$

It measures the change $d\mathbf{n}$ of the normal vector and the change in the surface position $d\mathbf{F}$ given by

$$II(\mathbf{u}, d\mathbf{u}) = d\mathbf{u}^T \begin{bmatrix} \mathbf{F}_{u_1u_1} \cdot \mathbf{n} & \mathbf{F}_{u_1u_2} \cdot \mathbf{n} \\ \mathbf{F}_{u_1u_2} \cdot \mathbf{n} & \mathbf{F}_{u_2u_2} \cdot \mathbf{n} \end{bmatrix} d\mathbf{u}. \tag{5}$$

A surface is essentially expressed by mapping these two matrices into the normal curvature, defined as

$$\kappa_0(\mathbf{u} \,|\, \mathbf{F}) = -\frac{d\mathbf{F} \cdot d\mathbf{n}}{d\mathbf{F} \cdot d\mathbf{F}}, \tag{6}$$

and the principal curvatures, which occur when $d\mathbf{n}$ and $d\mathbf{F}$ are aligned for a particular direction $d\mathbf{u}$. The maximum curvature κ_1 and the minimum curvature κ_2 are defined as

$$\kappa_1(\mathbf{u} \,|\, \mathbf{F}) = \frac{\mathbf{F}_{u_1u_1} \cdot \mathbf{n}}{\mathbf{F}_{u_1} \cdot \mathbf{F}_{u_1}}, \tag{7}$$

$$\kappa_2(\mathbf{u} \,|\, \mathbf{F}) = \frac{\mathbf{F}_{u_2u_2} \cdot \mathbf{n}}{\mathbf{F}_{u_2} \cdot \mathbf{F}_{u_2}}. \tag{8}$$

The directions $\mathbf{u}$ and $\mathbf{w}$ in which the principal curvatures occur, the *principal curvature directions*, are determined from $d\mathbf{u} = (du_1, du_2)$ by solving the quadratic equation

$$\begin{vmatrix} du_1du_1 & -du_1du_2 & du_2du_2 \\ \mathbf{F}_{u_1} \cdot \mathbf{F}_{u_1} & \mathbf{F}_{u_1} \cdot \mathbf{F}_{u_2} & \mathbf{F}_{u_2} \cdot \mathbf{F}_{u_2} \\ \mathbf{F}_{u_1u_1} \cdot \mathbf{n} & \mathbf{F}_{u_1u_2} \cdot \mathbf{n} & \mathbf{F}_{u_2u_2} \cdot \mathbf{n} \end{vmatrix} = 0. \tag{9}$$

An alternative to the principal curvatures is the direction independent *mean curvature*, κ_3, and *Gaussian curvature*, κ_4:

$$\kappa_3(\mathbf{u} \,|\, \mathbf{F}) = \kappa_1\kappa_2, \tag{10}$$

$$\kappa_4(\mathbf{u} \,|\, \mathbf{F}) = \frac{\kappa_1 + \kappa_2}{2}. \tag{11}$$

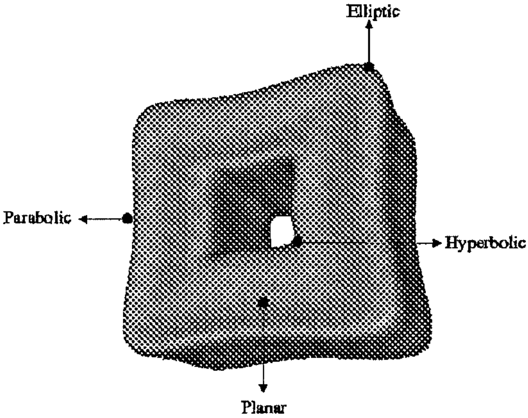

FIG. 1. An object with four types of boundary points: flat/planar, ridge/parabolic, peak/elliptic, saddle/hyperbolic.

For specific applications it is desirable to make these quantities scale-independent by multiplication with the surface area

$$\delta A(\mathbf{u} \mid \mathbf{F}) = \sqrt{(\mathbf{F}_{u_1})^2 + (\mathbf{F}_{u_2})^2}. \tag{12}$$

In this way two surfaces that are different in size but similar in shape can be considered the same with regard to curvature.

Two frequently used landmark localization approaches are discussed. They employ principal curvatures $\kappa_1(\mathbf{u} \mid \mathbf{F})$ and $\kappa_2(\mathbf{u} \mid \mathbf{F})$, the largest curvature $\kappa_1(\mathbf{u} \mid \mathbf{F})$ being the one with the highest value.

One way of classifying surface points is based on the principal curvatures $\kappa_1(\mathbf{u} \mid \mathbf{F})$ and $\kappa_2(\mathbf{u} \mid \mathbf{F})$ and associated principal directions $\mathbf{v}(\mathbf{u} \mid \mathbf{F})$ and $\mathbf{w}(\mathbf{u} \mid \mathbf{F})$ (e.g., [11, 33]). Two curvature functions, $\kappa_1(s \mid \mathbf{C})$ and $\kappa_2(s \mid \mathbf{C})$, which measure the curvature of the surface along a curve $\mathbf{C}(s)$, parameterized by s, that goes through the point of interest and has a tangent equal to the corresponding principal direction at that point are determined. Zero-crossings of the directional derivatives of these curvature functions are used to classify points as elliptic, parabolic, or planar (see Fig. 1 and Table 1). Elliptic points can be viewed as point landmarks, parabolic points as curve landmarks.

This approach provides a geometrically meaningful distinction between different types of surface points. The disadvantage is that on the basis of the zero-crossing only it is not possible to distinguish between "strong" and "weak" elliptic/parabolic points. Even almost flat structures and minor bumps in the surface are classified as elliptic or parabolic.

TABLE 1

Shape Classification on the Basis of Zero-Crossings of the Curvature Functions $\kappa_1(s \mid \mathbf{C})$ and $\kappa_2(s \mid \mathbf{C})$

	$\nabla\kappa_1(s \mid \mathbf{C}) = 0$	$\nabla\kappa_1(s \mid \mathbf{C}) \neq 0$
$\nabla\kappa_2(s \mid \mathbf{C}) = 0$	Elliptic	Parabolic
$\nabla\kappa_2(s \mid \mathbf{C}) \neq 0$	Parabolic	Planar

TABLE 2
Shape Classification on the Basis of the Signs of the Principal Curvatures $\kappa_1(\mathbf{u} \mid \mathbf{F})$ and $\kappa_2(\mathbf{u} \mid \mathbf{F})$

	$\kappa_1(\mathbf{u} \mid \mathbf{F}) < 0$	$\kappa_1(\mathbf{u} \mid \mathbf{F}) = 0$	$\kappa_1(\mathbf{u} \mid \mathbf{F}) > 0$
$\kappa_2(\mathbf{u} \mid \mathbf{F}) < 0$	Peak	Ridge	Saddle
$\kappa_2(\mathbf{u} \mid \mathbf{F}) = 0$	Ridge	Flat	Valley
$\kappa_2(\mathbf{u} \mid \mathbf{F}) > 0$	Saddle	Valley	Pit

Another common way of classifying surface points is based on the signs of principal surface curvatures (e.g., [1, 31]). The possible configurations are listed in Table 2. In this case, surface points are classified into one of six basic types: peak, pit, ridge, valley, saddle, and flat. An equivalent classification is reached by using the signs of the Gaussian and mean curvature. Saddle points are then resolved into saddle ridge, saddle valley, and minimal surface, yielding eight basic surface point types. Peaks pits and saddle points can be viewed as point landmarks, ridges as curve landmarks.

The advantage of this classification approach is that a distinction is also made between pits and peaks and between ridges and valleys which have identical shape but are embedded differently in three-dimensional space. However, as with the previous classification scheme, minor bumps in the surface are classified as elliptic or parabolic. The number of landmark points derived this way can be very large. In addition the distinction between different types of landmarks is not always clear, making them less suited for aiding segmentation.

The above discussion motivates the definition of landmarks on the basis of a *threshold for the principal curvature values*. We modify the classification scheme based on the signs of the principal curvature and classify surface points depending on whether the absolute values of the principal curvature $\kappa_1(\mathbf{u} \mid \mathbf{F})$ and $\kappa_2(\mathbf{u} \mid \mathbf{F})$ exceed a predefined threshold c_s, where the value of c_s is derived from a priori knowledge.

Three types of surface points are distinguished in this way, as illustrated in Fig. 2. We refer to them as point landmarks, curve landmarks, and sheet points. Point landmarks are uniquely localized due to the presence of a strong curvature in both principal directions. Curve landmarks are well defined in only one of the principal directions. Sheet points are only well defined in the normal direction. The points are listed in Table 3.

2.3. Application of Landmarks

When surface points are classified into point landmarks, curve landmarks, and sheet points as defined above, it is natural to adopt a priority scheme when objects are segmented.

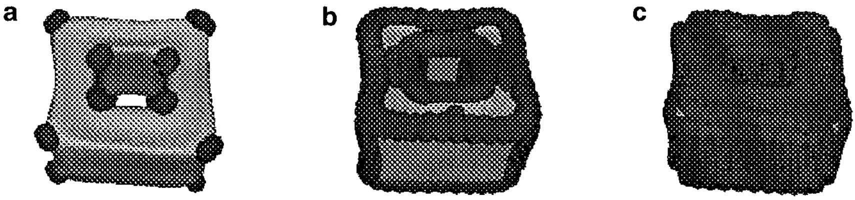

FIG. 2. Classification of surface points (indicated by spheres) on the basis of a threshold for the principal curvatures: (a) point landmarks, (b) curve landmarks, and (c) sheet points.

TABLE 3
Shape Classification on the Basis of a Threshold for the Principal Curvatures $\kappa_1(\mathbf{u} \mid \mathbf{F})$ and $\kappa_2(\mathbf{u} \mid \mathbf{F})$

	$\kappa_1(\mathbf{u} \mid \mathbf{F}) < -c_s$	$-c_s \leq \kappa_1(\mathbf{u} \mid \mathbf{F}) \leq c_s$	$c_s < \kappa_1(\mathbf{u} \mid \mathbf{F})$
$\kappa_2(\mathbf{u} \mid \mathbf{F}) < -c_s$	Point landmark	Curve landmark	Sheet points
$-c_s \leq \kappa_2(\mathbf{u} \mid \mathbf{F}) \leq c_s$	Curve landmark	Sheet point	Curve landmark
$c_s < \kappa_2(\mathbf{u} \mid \mathbf{F})$	Point landmark	Curve landmark	Point landmark

We employ the following order in finding objects:

1. Detect point landmarks. The result is a rough estimate of the position of the object by its point landmarks.
2. Given 1, localize curve landmarks. The result is an outline of the object.
3. Detect sheet points departing from the solution of 2. The result is the location of all boundary points.

This priority scheme makes it possible to search for curve landmarks and sheet points once point landmarks are detected. The incremental approach has two advantages. In the first place it makes it possible to handle complex image scenes by exploiting well-defined information first, reducing disturbance from unrelevant image data and diminishing the search space. Another advantage is that point landmarks reduce user interaction to point-to-point correspondence when dealing with erroneous solutions, e.g., due to propagation of errors made in point landmark localization.

3. LANDMARK-BASED SEGMENTATION BY NECKLACES

Landmark-based segmentation can be applied adequately within the deformable model platform. Deformable model methods have been extensively used for image segmentation (e.g., [4, 5, 30, 32]). To accommodate deformable models for segmentation of inhomogeneous boundaries, we have developed one that uses multiple features to define and exploit boundary landmarks. As its inhomogeneous and point-specific properties remind of a *necklace* with different kind of beads we call it as such.

The essence of necklaces is formulated in terms of the following aspects: how the boundary model is constructed, how the objective function is formulated and how optimization is performed. Figure 3 shows the main components of the necklace segmentation technique. They will be addressed in the following sections.

3.1. Model Construction

As in [14], we aim at learning multiple boundary features rather than defining them on the basis of a priori geometrical or analytical knowledge. However, in contrast to [14], where image and shape features are weighted according to the variation seen in the training feature functions, here we select them according to saliency seen in these functions. We are aware of the fact that this may introduce shortcomings related to neglecting variational information. However, this has inconsequential effects on the concepts proposed here.

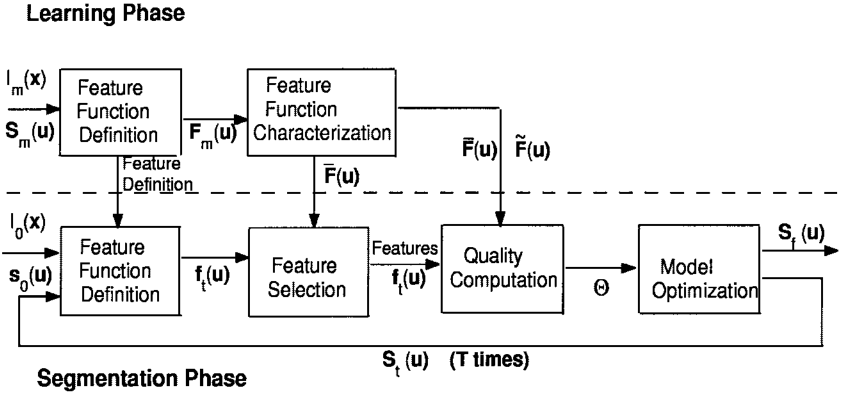

FIG. 3. Overview of components of the necklace segmentation technique. Note that at the start of the learning phase, we need a set of images $I_m(\mathbf{x}), m = 1, \ldots, M$ with corresponding correct ground-truth segmentation represented by $\mathbf{S}_m(\mathbf{u})$.

3.1.1. Shape Features

Object shape is learned from a training set of M surfaces $\mathbf{S}_m(\mathbf{u}), m = 1, \ldots, M$, embedded in example images $I_m(\mathbf{x})$. A number of shape features (e.g., curvature) are recorded along $\mathbf{S}_m(\mathbf{u})$ to obtain a population of features. The population average for the nth feature is

$$\bar{F}^n(\mathbf{u}) = \frac{1}{M} \sum_{m=1}^{M} F_m^n(\mathbf{u} \mid \mathbf{S}_m), \tag{13}$$

where $F_m^n(\mathbf{u}), n \in \{1, \ldots, N\}$ is the feature function emanating from fitting a surface through the feature samples taken along $\mathbf{S}_m(\mathbf{u})$. The feature functions are smoothed and aligned to properly compare features at same positions along the boundary. The variation in shape feature values is

$$\breve{F}^n(\mathbf{u}) = \frac{1}{M} \sum_{m=1}^{M} \left\| F_m^n(\mathbf{u} \mid \mathbf{S}_m) - \bar{F}^n(\mathbf{u})^2 \right\|. \tag{14}$$

As shape feature we select the translation and rotation invariant *mean curvature* made scale independent by multiplication with the local surface area $\delta A(\mathbf{u} \mid \mathbf{S}_m)$. Apart from this, we also learn the principal curvature values $\kappa_1(\mathbf{u} \mid \mathbf{S}_m)$ and $\kappa_2(\mathbf{u} \mid \mathbf{S}_m)$ to relieve landmark localization in subsequent steps. Scalar and vector features form dimensions of the feature space. Table 4 lists the features and their definition.

3.1.2. Image Features

As the gray-level patterns about a boundary point will often be similar in different images, we also statistically capture the image data along the object boundary. The statistics of image features along $I_m(\mathbf{S}_m(\mathbf{u}))$ are computed analogously to the shape statistics,

$$\bar{F}^n(\mathbf{u}) = \frac{1}{M} \sum_{m=1}^{M} F_m^n(\mathbf{u} \mid I_m, \mathbf{S}_m), \tag{15}$$

TABLE 4
Shape Features in Our Implementation

Feature	Dimension	Definition
1st principal curvature	$F_m^1(\mathbf{u}\mid\mathbf{S}_m)$	$k_1(\mathbf{u}\mid\mathbf{S}_m)$
2nd principal curvature	$F_m^2(\mathbf{u}\mid\mathbf{S}_m)$	$k_2(\mathbf{u}\mid\mathbf{S}_m)$
Mean curvature	$F_m^3(\mathbf{u}\mid\mathbf{S}_m)$	$(k_1(\mathbf{u}\mid\mathbf{S}_m) * k_2(\mathbf{u}\mid\mathbf{S}_m))/\delta A(\mathbf{u}\mid\mathbf{S}_m)$

Note. First principal curvature values $F_m^1(\mathbf{u}\mid\mathbf{S}_m)$ form the first dimension of $\mathcal{F}$ and second principal curvature values $F_m^2(\mathbf{u}\mid\mathbf{S}_m)$ form the second dimension. The third dimension is formed by scale independent *mean curvature* values $F_m^3(\mathbf{u}\mid\mathbf{S}_m)$.

and corresponding variance,

$$\breve{F}^n(\mathbf{u}) = \frac{1}{M}\sum_{m=1}^{M} \left\| F_m^n(\mathbf{u}\mid I_m, \mathbf{S}_m) - \bar{F}^n(\mathbf{u})^2 \right\|. \tag{16}$$

To highlight specific structures in the image while suppressing unrelevant ones, we use filter information obtained from the structure tensor. The structure tensor is commonly used to capture the local three-dimensional structure of the image (e.g., [7, 16, 23, 35]), as it contains more information about the image than gradient information only. For an image position $I(\mathbf{x})$ it is defined by

$$\mathbf{M}(\mathbf{x};\sigma_d) = \begin{pmatrix} \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_1}(\mathbf{x};\sigma_d)} & \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_3}(\mathbf{x};\sigma_d)} \\ \overline{I_{x_1}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_2}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_2}(\mathbf{x};\sigma_d)I_{x_3}(\mathbf{x};\sigma_d)} \\ \overline{I_{x_3}(\mathbf{x};\sigma_d)I_{x_1}(\mathbf{x};\sigma_d)} & \overline{I_{x_3}(\mathbf{x};\sigma_d)I_{x_2}(\mathbf{x};\sigma_d)} & \overline{I_{x_3}(\mathbf{x};\sigma_d)I_{x_3}(\mathbf{x};\sigma_d)} \end{pmatrix}, \tag{17}$$

where σ_d is the differentiation scale and $(\bar{\cdot})$ denotes smoothing with scale σ_s to obtain the dyadic product. The matrix elements I_{x_i} denote partial image derivatives, defined as the convolution of the image with derivatives of a Gaussian function,

$$I_{x_i}(\mathbf{x};\sigma_d) = I(\mathbf{x}) * G_{x_i}(\mathbf{x};\sigma_d).$$

The eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ of $\mathbf{M}(\mathbf{x};\sigma_d)$ are used to define image feature functions. When all eigenvalues are sufficiently large this indicates a pointlike structure. The feature function $F_m^4(\mathbf{u}\mid I_m, \mathbf{S}_m)$ records a boundary in an image filtered to highlight such point-like structures. Two eigenvalues, λ_1, λ_2, much larger than the smallest eigenvalue λ_3 indicate a point on a curve-like structure. The feature function $F_m^5(\mathbf{u}\mid I_m, \mathbf{S}_m)$ records a boundary in an image filtered to highlight such tubular structures. Similarly, $F_m^6(\mathbf{u}\mid I_m, \mathbf{S}_m)$ measures intensity values in an image which has been processed to highlight sheet-like structures. The image features are listed in Table 5.

The normalization constant c_t in Table 5 reflects the minimum required image variation for an image point to be highlighted as one of the three boundary point types. This way we can focus on strong point landmarks and curve landmarks while disregarding image variations caused by minor bumps in the object boundary or by noise.

At this point we have a statistical description of the object of interest. The relevant information is contained in the $N = 6$ dimensional population average feature function $\bar{\mathbf{F}}(\mathbf{u}) = [\bar{F}^1(\mathbf{u}), \ldots, \bar{F}^6(\mathbf{u})]$ and variation $\breve{\mathbf{F}}(\mathbf{u}) = [\breve{F}^1(\mathbf{u}), \ldots, \breve{F}^6(\mathbf{u})]$. We transfer this information to the segmentation phase.

TABLE 5
Image Features in Our Implementation

Feature	Dimension	Definition
Image point landmarks	$F_m^4(\mathbf{u} \mid I_m, \mathbf{S}_m)$	$1 - e^{-\frac{\lambda_3}{c_t}}$
Image curve landmarks	$F_m^5(\mathbf{u} \mid I_m, \mathbf{S}_m)$	$1 - e^{-\frac{\lambda_2}{c_t}}$
Image sheet landmarks	$F_m^6(\mathbf{u} \mid I_m, \mathbf{S}_m)$	$1 - e^{-\frac{\lambda_1}{c_t}}$

Note. $F_m^4(\mathbf{u} \mid I_m, \mathbf{S}_m)$ highlights point landmarks, $F_m^5(\mathbf{u} \mid I_m, \mathbf{S}_m)$ highlights curve landmarks, and $F_m^6(\mathbf{u} \mid I_m, \mathbf{S}_m)$ highlights sheet points.

3.2. Model Qualification

For segmentation of an unknown image we exploit landmarks found in $\bar{\mathbf{F}}(\mathbf{u})$.

3.2.1. *Deformable Surface*

The *deformable surface* $\mathbf{s}_t(\mathbf{u})$, $\mathbf{s}_t : \mathbf{u} \in U \subset \Re^2 \to \Re^3$ is used to actively find an object in the unknown image $I_0(\mathbf{x})$. It deforms in time t to suggest new features extracted from it and the image it lives in, to be compared to $\bar{\mathbf{F}}(\mathbf{u})$ and $\breve{\mathbf{F}}(\mathbf{u})$ as the statistics from learning. The initial deformable surface $\mathbf{s}_0(\mathbf{u})$ is the average of the aligned training shapes,

$$\mathbf{s}_0(\mathbf{u}) = \frac{1}{M} \sum_{m=1}^{M} \mathbf{S}_m(\mathbf{u}). \tag{18}$$

On the deformable surface, point landmarks, curve landmarks, and sheet points are localized. This is done by investigating at path positions $\mathbf{u}$ whether the principal curvatures of the population average $\bar{\mathbf{F}}(\mathbf{u})$ exceed the predefined threshold c_s. For instance, if at $\bar{\mathbf{F}}^n(\mathbf{u})$ the statistical average of both principal curvatures exceeds the predefined threshold, than $\mathbf{s}_t(\mathbf{u})$ is considered a point landmark. Hence,

$$\mathcal{U}_A = \{\mathbf{u} \mid -c_s > \kappa_1(\mathbf{u} \mid \bar{\mathbf{F}}) > c_s, -c_s > \kappa_2(\mathbf{u} \mid \bar{\mathbf{F}}) > c_s\}, \tag{19}$$

$$\mathcal{U}_B = \{\mathbf{u} \mid -c_s > \kappa_1(\mathbf{u} \mid \bar{\mathbf{F}}) > c_s, -c_s < \kappa_2(\mathbf{u} \mid \bar{\mathbf{F}}) < c_s\}, \tag{20}$$

$$\mathcal{U}_C = \{\mathbf{u} \mid -c_s < \kappa_1(\mathbf{u} \mid \bar{\mathbf{F}}) < c_s, -c_s < \kappa_2(\mathbf{u} \mid \bar{\mathbf{F}}) < c_s\}. \tag{21}$$

The disjoint sets $\mathcal{U}_A, \mathcal{U}_B, \mathcal{U}_C$ together contain all path positions, with $\mathbf{s}_t(\mathbf{u})$ for $\mathbf{u} \in \mathcal{U}_A$ being point landmarks, $\mathbf{s}_t(\mathbf{u})$ for $\mathbf{u} \in \mathcal{U}_B$ curve landmarks, and $\mathbf{s}_t(\mathbf{u})$ for $\mathbf{u} \in \mathcal{U}_C$ sheets points.

3.2.2. *Objective Function*

To qualify the deformable surface when it is in an unknown image $I_0(\mathbf{x})$, an objective function is defined that measures how much a recorded boundary deviates from the population average. The deviation is calculated in terms of the above defined local shape and image features. An integration is carried to compute a global measure of deviation for all N features measured along $I_0(\mathbf{s}_t(\mathbf{u}))$. Denoting the N feature functions emanating from $I_0(\mathbf{s}_t(\mathbf{u}))$ by $\mathbf{f}_t(\mathbf{u}) = [f^1(\mathbf{u}), \ldots, f^N(\mathbf{u})]$, the global objective function summarizes the fit as follows

$$\Theta(\mathbf{s}_t \mid \bar{\mathbf{F}}, \breve{\mathbf{F}}, \mathbf{f}_t) = \int_{\mathbf{u}} \frac{1}{N} \sum_{n=1}^{N} \omega^n(\mathbf{u}) \theta^n(\mathbf{u} \mid \bar{F}^n, \breve{F}^n, f^n)\, d\mathbf{u}. \tag{22}$$

The weight function $\omega^n(\mathbf{u})$ is the actual feature selector. For each feature n it indicates the relative importance of that feature based on the *mean curvature* of the population average for feature n, denoted by $\kappa_3(\mathbf{u} \mid \bar{F}^n)$. Currently, $\omega^n(\mathbf{u})$ selects one feature and turns off others:

$$\omega^n(\mathbf{u}) = \begin{cases} 1 & \text{if } \kappa_3(\mathbf{u} \mid \bar{F}^n) > \kappa_3(\mathbf{u} \mid \bar{F}^i), i \in \{1, \ldots, N\}, i \neq n \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

The local objective function $\theta^n(\mathbf{u} \mid \bar{F}^n, \breve{F}^n, f^n)$ measures the distance between expected and recorded values for feature n. To ensure a controllable distance measure, the Mahalanobis distance [9] is computed using population average and variation information obtained from learning. For the nth feature this means

$$\theta^n(\mathbf{u} \mid \bar{F}^n, \breve{F}^n, f^n) = v(\mathbf{u}) \left(\frac{\bar{F}^n(\mathbf{u}) - f^n(\mathbf{u})}{\breve{F}^n(\mathbf{u})} \right)^2. \tag{24}$$

The local objective function is controlled by means of weighting function $v(\mathbf{u})$. It regulates the contribution of point landmarks, curve landmarks, and sheet points to the objective function value using prespecified weights v_A, v_B, and v_C:

$$v(\mathbf{u}) = \begin{cases} v_A & \text{if } \mathbf{u} \in \mathcal{U}_A \\ v_B & \text{if } \mathbf{u} \in \mathcal{U}_B \\ v_C & \text{if } \mathbf{u} \in \mathcal{U}_C. \end{cases} \tag{25}$$

The weights are positive and subjected to the following constraint:

$$v_A + v_B + v_C = 1. \tag{26}$$

In the priority scheme discussed in Section 2.3, the first step is performed using the weight settings $v_A = 1, v_B = 0, v_C = 0$, step two using $v_A = 0, v_B = 1, v_C = 0$, and step three using $v_A = 0, v_B = 0, v_C = 1$. The weights may also be set such that features along the entire surface contribute to the definition of the object, constrained according to Eq. (26).

3.3. Model Optimization

The optimization of the deformable surface corresponds to the minimization of the objective function. Optimization only effects the geometry of the initial surface $\mathbf{s}_{t=0}$ by changing the position of surface points while keeping the same uniform parameterization. The aim is to find the optimal deformable surface $\mathbf{s}_f(\mathbf{u})$, such that

$$\mathbf{s}_f = \underset{\mathbf{s}_t}{\operatorname{argmin}}\, \Theta(\mathbf{s}_t \mid \bar{\mathbf{F}}, \breve{\mathbf{F}}, \mathbf{f}_t). \tag{27}$$

Optimization involves two main steps for each surface point. In the first step a new suggested position is calculated based on the fit quality of the deformable surface. This is followed by movement of the deformable surface in order to reposition each point as close as possible to the newly preferred position.

3.3.1. Search Scheme

We adopt a priority scheme when optimizing the deformable surface; different surface points are fitted to the image data at different times and in different number of dimensions. The aim is to first search for well-defined points in the image and to exploit solutions thereof for obtaining an optimal global fit.

We consider optimization a function of scalars α, β, and γ. These parameters are associated at each surface point with the normal vector $\mathbf{n}(\mathbf{u} \mid \mathbf{s}_t)$, the maximum principal direction $\mathbf{v}(\mathbf{u} \mid \mathbf{s}_t)$, and the minimum principal direction $\mathbf{w}(\mathbf{u} \mid \mathbf{s}_t)$, respectively. These local surface properties make it possible to steer segmentation in ways that take into account physical object properties.

The suggested movement, or drive $\mathbf{d}(\mathbf{u} \mid \mathbf{s}_t)$, that works on a surface point $\mathbf{s}_t(\mathbf{u})$ to move it to the newly preferred position is formulated as a linear combination:

$$\mathbf{d}(\mathbf{u} \mid \mathbf{s}_t) = \alpha \mathbf{n}(\mathbf{u} \mid \mathbf{s}_t) + \beta \mathbf{v}(\mathbf{u} \mid \mathbf{s}_t) + \gamma \mathbf{w}(\mathbf{u} \mid \mathbf{s}_t). \tag{28}$$

In this way, movement is restricted to well-defined directions, permitting search spaces of different dimensions for different types of points as illustrated in Fig. 4. Moreover, the magnitude of the movement can be easily controlled by predefining the range of acceptable values for the optimization parameters.

Fitting the deformable surface to the image data is performed in four steps, following the priority scheme discussed in Section 2.3. In the first step deformable surface point landmarks are fitted to corresponding image point landmarks. The fit is performed in three dimensions where the search space is naturally spanned by the normal vector $\mathbf{n}(\mathbf{u} \mid \mathbf{s}_t)$ and the principal curvature directions $\mathbf{v}(\mathbf{u} \mid \mathbf{s}_t)$ and $\mathbf{w}(\mathbf{u} \mid \mathbf{s}_t)$. Accordingly, optimization reduces to finding the optimal values for α, β, and γ, defining the the quantity in Eq. (28).

The user may fit a restricted number of selected point landmarks by pointing and clicking in the image to steer the segmentation. User interaction yields initial values for parameters α, β, and γ. That is, interaction with image point $\mathbf{x}_p$ yields parameter values for which holds

$$\mathbf{x}_p = \mathbf{s}_t(\mathbf{u}) + \mathbf{d}(\mathbf{u} \mid \mathbf{s}_t), \quad \mathbf{u} \in \mathcal{U}_A. \tag{29}$$

The solution obtained by solving this linear system of equations forms the departure point for the computational method. The optimal values for α, β, and γ are automatically searched for in the vicinity of the point of interaction.

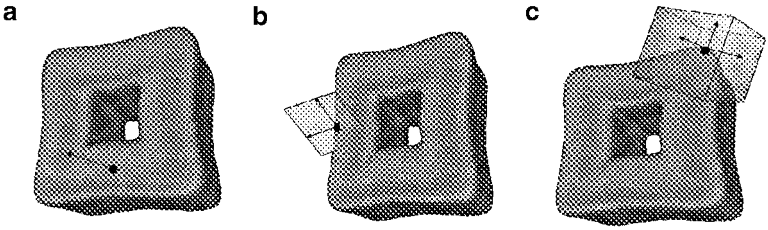

FIG. 4. Three types of search spaces: (a) one-dimensional for sheet points, (b) two-dimensional for curve landmarks, and (c) three-dimensional for point landmarks.

In the next step, curve landmarks are fitted to the image data departing from the previous result. Optimization is done away from point landmarks; i.e., first curve landmarks close to point landmarks are optimized, then curve landmarks at greater distances. In this way we further exploit earlier found solutions. The location of curve landmarks in the image is searched for in a two-dimensional space spanned by the normal vector $\mathbf{n}(\mathbf{u} \mid \mathbf{s}_t)$ and the maximum principal curvature $\mathbf{v}(\mathbf{u} \mid \mathbf{s}_t)$. Optimization corresponds to finding the optimal values for α and β, while γ is set to zero. Consequently, the driving force working on curve landmark $\mathbf{s}_t(\mathbf{u})$, $\mathbf{u} \in \mathcal{U}_B$, is fixed by

$$\mathbf{d}(\mathbf{u} \mid \mathbf{s}_t) = \alpha\mathbf{n}(\mathbf{u} \mid \mathbf{s}_t) + \beta\mathbf{v}(\mathbf{u} \mid \mathbf{s}_t). \tag{30}$$

Sheet points are then fitted to the image data to find all remaining boundary points. First those sheet points are optimized that are close to curve landmarks than sheet points at greater distance for the same reason as mentioned above. The location of sheet points in the image is obtained by fitting surface points $\mathbf{s}_t(\mathbf{u})$, $\mathbf{u} \in \mathcal{U}_C$ to the image data in the normal direction only:

$$\mathbf{d}(\mathbf{u} \mid \mathbf{s}_t) = \alpha\mathbf{n}(\mathbf{u} \mid \mathbf{s}_t). \tag{31}$$

So far point landmarks, curve landmarks, and sheets points have been optimized separately. In the final step all surface points are optimized once again in their respective dimensions to obtain a global solution and to fine tune results. This is especially important in light of the step-by-step segmentation scheme where errors made in early steps may be propagated. A final optimization accounts for this.

Furthermore, as movement of one surface point almost always suggests a similar movement of neighboring surface points, we choose to distribute the force working on a single surface point along the entire deformable surface. That is, we preserve the shape of the deformable surface as much as possible when fitting a specific surface point by simultaneously estimating the correct position for deformable surface points that have not yet been optimized. This means that, given drive $\mathbf{d}(\mathbf{u}_i \mid \mathbf{s}_t)$ working on surface point $\mathbf{s}_t(\mathbf{u}_j)$, the following movement of $\mathbf{s}_t(\mathbf{u}_j)$, $\forall\mathbf{u}_j \in U$, is performed to obtain the next preferred position,

$$\mathbf{s}_{t+1}(\mathbf{u}_j) = \mathbf{s}_t(\mathbf{u}_j) + \mathbf{d}(\mathbf{u}_i \mid \mathbf{s}_t)e^{-(\delta/c_d)}, \tag{32}$$

where $\delta = D(\mathbf{s}_t(\mathbf{u}_i), \mathbf{s}_t(\mathbf{u}_j))$ denotes the Euclidean distance between surface points $\mathbf{s}_t(\mathbf{u}_i)$ and $\mathbf{s}_t(\mathbf{u}_j)$. The constant $c_d > 0$ is a predefined value controlling the magnitude of the distribution. A small value for the distribution constant influences the shape of the surface in the immediate neighborhood of the point under optimization, while a large value also effects the shape of the deformable surface at large distances. This is illustrated in Fig. 5.

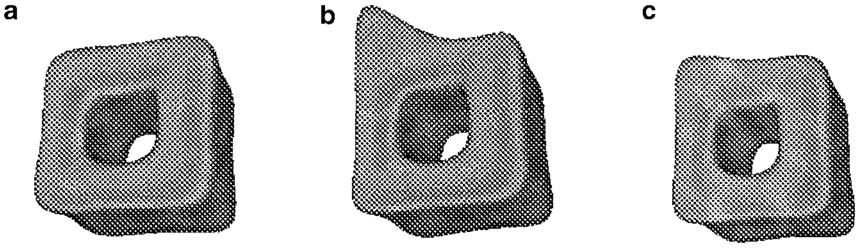

FIG. 5. The left upper point-landmark is repositioned causing deformation of the entire surface: (a) initial surface, (b) deformation with a low distribution constant, (c) deformation with a large distribution constant.

3.3.2. *Initialization*

The inherent difficulty in deformable model methods is that searching for a minimum over a nonconvex function is possible only under predefined conditions that lead to the desired solution [6]. As the objective function is typically nonconvex, it might have many local minima. As a consequence the use of a local optimization method will usually find only local minima, and therefore it becomes necessary to place the initial surface close to the true boundary to obtain acceptable results.

We allow the user to specify an initial guess that is close to a local minimum. The initial surface for the object at hand is placed in the image and aligned with the sought object boundary by transformation. The user translates, rotates, and scales the a priori deformable surface until an acceptable first guess is obtained. Starting from the user initialization the optimization method refines the initial guess to fit to the image data. In this context, the purpose of initialization here is to bootstrap optimization.

3.4. Implementation

In our implementation we use B-spline surfaces. B-spline surfaces have been chosen because they allow analytic computation [12, 25] and hence an easy derivation of local second order surface properties. Apart from this, the influence of control points is limited to well-defined surface intervals providing local control of surface manipulation. This simplifies user-steered adjustments of locally undesirable results.

The B-spline surface is a collection of B-spline curves [25]. Taking $\mathbf{s}_t(\mathbf{u})$ as an example, the surface is defined as the set of all points given by the following expression for all parameter values of $\mathbf{u} = [u_1, u_2]^T$,

$$\mathbf{s}_t(\mathbf{u}) = \sum_{j=1}^{J} \sum_{l=1}^{L} B_p^j(u_1) B_q^l(u_2) \mathbf{b}^{j,l}, \tag{33}$$

where $\mathbf{b}^{j,l}$ is the array of $J \times L$ control points. The $B_p^j(u_1)$ are B-spline basis functions of degree $p-1$ in u_1 direction, which are $p-2$ times continuously differentiable. The $B_q^l(u_2)$ are the basis functions of degree $q-1$ in u_2, which are $q-2$ times continuously differentiable. A set of knots in a path parameter interval relating to the control points is used to define the basis functions. For an analytic expression of B-spline basis functions see [25].

For the calculation of new shapes during the optimization process we need the points $\mathbf{P}^{j,\ell}$, $j = 0, \ldots, J$, $\ell = 0, \ldots, L$, which are the interpolation points of $\mathbf{s}_t(\mathbf{u})$ at $\mathbf{u} = [u_1^j, u_2^\ell]^T$. They are defined as

$$\mathbf{P}^{j,\ell} = \mathbf{s}_t\left(u_1^j, u_2^\ell\right) = \sum_{j=1}^{J} \sum_{l=1}^{L} B_p^j\left(u_1^j\right) B_q^l\left(u_2^\ell\right) \mathbf{b}^{j,l}. \tag{34}$$

Powell's method is used in combination with Brent's line minimization method [3] to calculate new suggested positions for the control points. This variant of coordinate descent optimization minimizes each parameter in turn using Brents line search minimization. The method cycles repeatedly through all parameters. It discards one coordinate dimension in favor of another one if this is a better candidate. The process is repeated after each cycle of minimization, until a stable solution is found for the objective function. Although being slower than gradientdescent approaches, Powell's method has the distinct benefit that no derivatives for the objective function need to be known.

4. EXPERIMENTS AND RESULTS

The application to evaluate the proposed necklace method deals with 20 CT images of vertebrae. The CT images are acquired on a Philips SR 700 CT at 140 kV (Philips Medical Systems, Best, The Netherlands). They were scanned at a resolution of approximately 225 × 225 $\mathcal{W}$m and slice thickness of 0.5 mm. The images concern the lumbar part of the human spine, exhibiting considerable variation in shape and gray-level appearance even among normal patients.

Four steps are involved in the evaluation procedure: (1) manually outlining and aligning vertebral boundaries for composing the training set, (2) statistically analyzing features in the training set for construction of the vertebra model, (3) segmenting vertebral boundaries in new unknown images, and (4) comparing segmentation results with the known solutions. These steps are performed in two distinct phases: learning and segmentation.

In the learning phase the medical expert provides ground-truth segmentations by indicating three-dimensional points in three two-dimensional orthogonal slices of the volume data. This is done only one time and within one day. A total of 144 points are positioned along a single vertebra boundary, forming the interpolation points for B-spline surfaces. The resulting B-spline surfaces and corresponding images are added to the training set. Subsequently, image and shape features are taken at 400 samples along each surface in the training set, then statistically analyzed.

The testing phase involves segmentation of the vertebra images with the help of the necklace model. The segmentations are then compared with the ground truth delineations. This is done using a cross-validation technique [19], as splitting the limited amount of 20 vertebra images into separate fixed learning and testing categories would result in an insufficient number of test and learning cases. Figure 6 shows the population average shape of the vertebra in our training set.

4.1. Experiments

The experiments investigate the benefit of landmark-based segmentation with help of necklaces over segmentation under uniform and smooth boundary assumption. We concentrate on landmarks that emanate from the curvature of the vertebra surface; i. e., landmarks

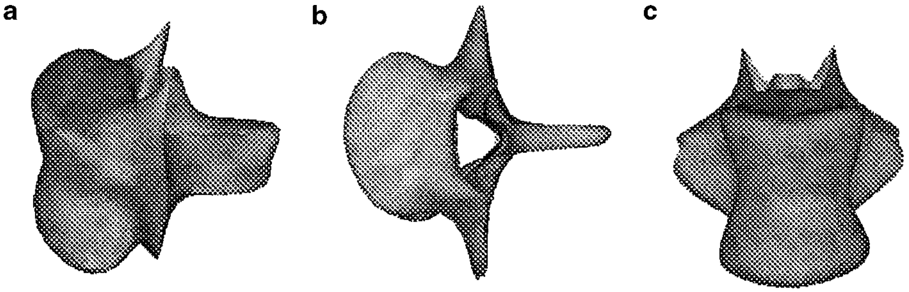

FIG. 6. The population average vertebra surface $\mathbf{s}_{t=0}(\mathbf{u})$: (a) side view, (b) upper view, and (c) frontal view. Its shape is controlled by 144 interpolation points.

are defined only on the basis of high value positions in $\bar{F}^1(\mathbf{u})$ and $\bar{F}^2(\mathbf{u})$ under the assumption that in this application saliency is predominantly due to vertebra shape. Apart from this, feature selection is done according to the type of surface point: at point landmarks features $F^3(\mathbf{u})$ and $F^4(\mathbf{u})$ are evaluated, at curve landmarks features $F^3(\mathbf{u})$ and $F^5(\mathbf{u})$, and at sheet points $F^3(\mathbf{u})$ and $F^6(\mathbf{u})$. In this way we always take into consideration shape features while exploiting saliency in the image data.

To compare landmark-based segmentation using necklaces with conventional segmentation we also study segmentation under uniform boundary assumption. To this end we reduce the necklace method to one without landmarks (a) by learning and applying $F^3(\mathbf{u})$ and $F^4(\mathbf{u})$ along the entire boundary and (b) by optimizing each point of the deformable surface in three dimensions, regardless of the dimensionality of the features at the surface points.

We perform three segmentation experiments both for landmark-based segmentation and for the conventional way of segmentation. The ground-truth shape model is transformed to the initial deformable surface by translating it with $-15 \leq \chi \leq 15$ voxels, rotating it by $-10^\circ \leq \psi \leq 10^\circ$ along the z coordinate, or scaling it by $0.9 \leq \sigma_t \leq 1.1$ times with respect to a center point. The deformable surface is then fitted to the image data using the Powell optimization procedure, with equilibrium defined as a 10% or smaller change in maximal 50 iterations for line search and a maximum of 50 calls to line search. The curvature threshold is fixed at $c_s = 0.03$. In the first three steps of the priority scheme the weights v_A, v_B, and v_C are turned either on or off depending on the segmentation aim, whereas in the global optimization step they are set to $v_A = 0.5$, $v_B = 0.3$, and $v_C = 0.2$. The image features are obtained using differentiation scale $\sigma_d = 3.0$ and smoothing scale $\sigma_s = 4.0$. The normalization constant c_t is set to 100.

4.2. Results

Evaluating the spatial accuracy of the necklace method amounts to comparing the extracted vertebral segments with ground truth, which is not predetermined, but rather is a function of the test data and the accuracy of the human expert. The distance between the segmented surface and the target surface is measured using the root squared metric error, which computes the average Euclidean distance between points in the proposed solution and points in the ground truth. The Euclidean distance is computed using a discrete error formula,

$$\epsilon_d = \min_{0 \leq \delta < L} \frac{1}{L} \sum_{l=0}^{L-1} \|\mathbf{S}_m(\mathbf{u}_l) - \mathbf{s}_f(\mathbf{u}_l + \delta)\|, \tag{35}$$

where $\mathbf{S}_m(\mathbf{u})$ is the ground-truth solution and $\mathbf{s}_f(\mathbf{u})$ is a point on the optimized deformable surface. This error metric is used commonly for measuring the accuracy of a point sample distribution.

The first row in Fig. 7 shows the average Euclidean distance between the segmentation result (with and without landmarks) and the ground-truth solution for different transformations. The flatness of the graphs is an indication of the capture range of the deformable surface. It can be seen that even small rotations negatively influence segmentation, while translations and scaling up to ±5 pixels and scaling with ±0.04 produce acceptable results with Euclidean distance less than 5. In most cases landmark-based segmentation achieves a higher reduction in Euclidean distance; i.e., the result is closer to the true solution. Sometimes the result is worse. We attribute this to the fact that when point landmarks

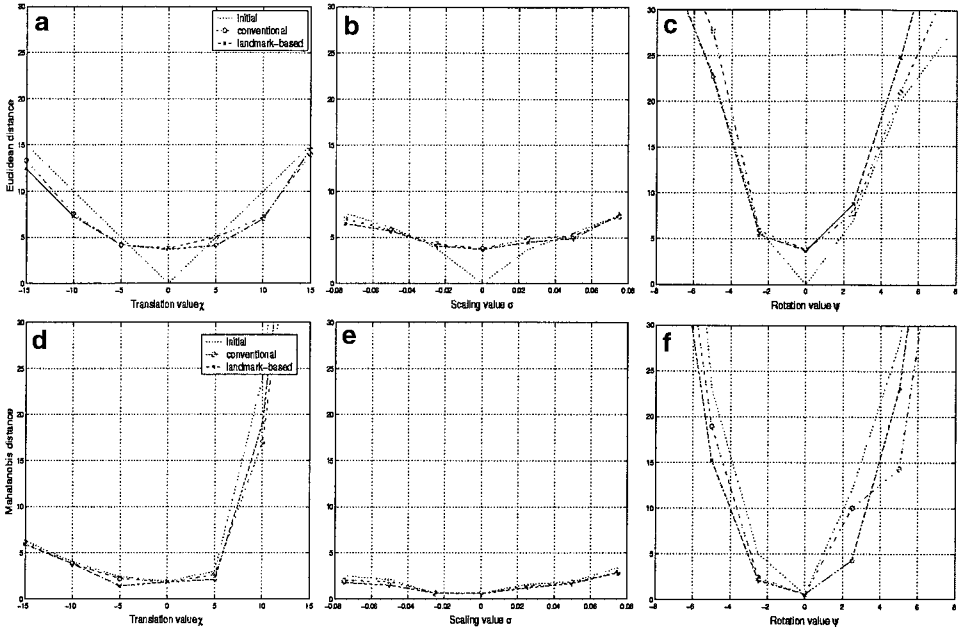

FIG. 7. Comparison of segmentation results with and without application of landmarks. Top row shows the average Euclidean distance of results to ground-truth shape for (a) translation, (b) scaling, (c) rotation. Bottom row shows the average Mahalanobis distance of result to population average for, (d) translation, (e) scaling, (f) rotation.

are localized and fixed wrongly; e.g., because the sought image landmark point is out of the capture range of the deformable surface, it has a considerable negative affect on the subsequent localization of other points. We conclude that landmark-based segmentation using necklaces outperforms conventional segmentation when point landmarks are localized well.

The second row in Fig. 7 indicates how the average Euclidean distance corresponds to the average Mahalanobis distance of the segmentation result to the population average. It can be seen that the Mahalanobis distance after optimization is lower then at the beginning, meaning that the final deformed surface has feature values closer to population average than the initial deformable surface. Landmark-based segmentation obtains a larger reduction in Mahalanobis distance when starting from a scaled version of the ground-truth shape. For translation and rotation landmark-based segmentation yields higher reduction in some cases, and lower in other ones. Note that for the vertebra application the objective function is much more sensitive for translation and rotation than for scaling, presumably due to the dense context of the segmentation scene with ribs and other organs easily disturbing the fit of the vertebra model.

From Fig. 7 we observe that even when the initial deformable surface is the ground-truth, the optimization brings the deformable surface to rest at an average of a little under a 4-voxel distance. The minimal voxel distance of 4 is largely due to variation in assignment of ground truth by the expert. For our stochastic model such a variation is the lower bound in accuracy of what the algorithm is capable of finding back. In a four-time repetition of the ground-truth assignment under similar conditions we found a variation of 4.4, 4.9, and 6.7 voxels for point-landmarks, curve landmarks, and sheet points respectively. We expect an improvement of performance proportional to the accuracy of the ground-truth segmentation, either by more precise individual assignment or by using larger amounts of salient points per vertebra.

Furthermore, we partially ascribe the minimal voxel distance of 4 to the fact that the model was built from different types of lumbar vertebrae (L1, L2, L3, L4, and L5), which may have led to a population average that does not sufficiently resemble any of the segmented vertebra. This causes the deformable surface sometimes to shift away from its correct position in order to comply to the population average. We note that it is essential to use as much invariant features as possible to reduce the effect of the population average features to the accuracy of the segmentation result, e.g., considering image features after histogram equalization.

To obtain an impression of the computational gains of our algorithm, we also observe how fast the initial model converges to a stable solution. We perform 10 segmentations where we observe the number of calls to the objective function, the reduction in Mahalanobis distance, and the reduction in Euclidean distance. For landmark-based segmentation a total of 4104 calls are required on average, taking on average a little over 1 min to reach equilibrium. The average reduction in Mahalanobis distance is 0.7, corresponding with an average reduction in Euclidean distance of approximately 1.4. Conventional segmentation requires on average 9055 calls, with an average reduction in Mahalanobis distance of 0.43 and in Euclidean distance of 1.1. We attribute the computational efficiency to the reduction of search space of solutions for curve landmarks (2D) and sheet points (1D). The fact that landmark solutions are propagated to estimate adjacent boundary points on the same and other vertebrae, also contributes to an increase in computational efficiency.

We give a segmentation illustration of the fourth lumbar vertebra. Segmentation departs from the population average vertebra shape in contrast to the above experiments which have been conducted starting from the ground-truth shape. To get an impression of the tree-dimensional context of the image data and the deformable surface in it, Fig. 8 gives three different views of the scene. Figure 8a shows a volume rendering of the image data on the basis of a threshold for gray-value intensity, providing a visual means of comparing results from the deformable surface approach with those based on thresholding. The three-orthogonal slices through the image data show intensity values of the original image. Figure 8b shows how the deformable surface and the volume rendering relate by superimposing the surface on the rendering. Figure 8c shows another view of the deformable

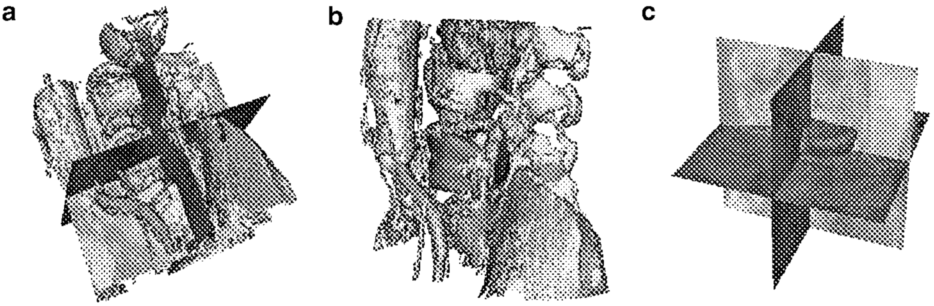

FIG. 8. Three views of the segmentation scene: (a) volume rendering of the 3D data block with orthogonal intensity slices, (b) volume rendering of the 3D data block with initial model superimposed, (c) initial model embedded in 3D data block visualized by orthogonal intensity planes.

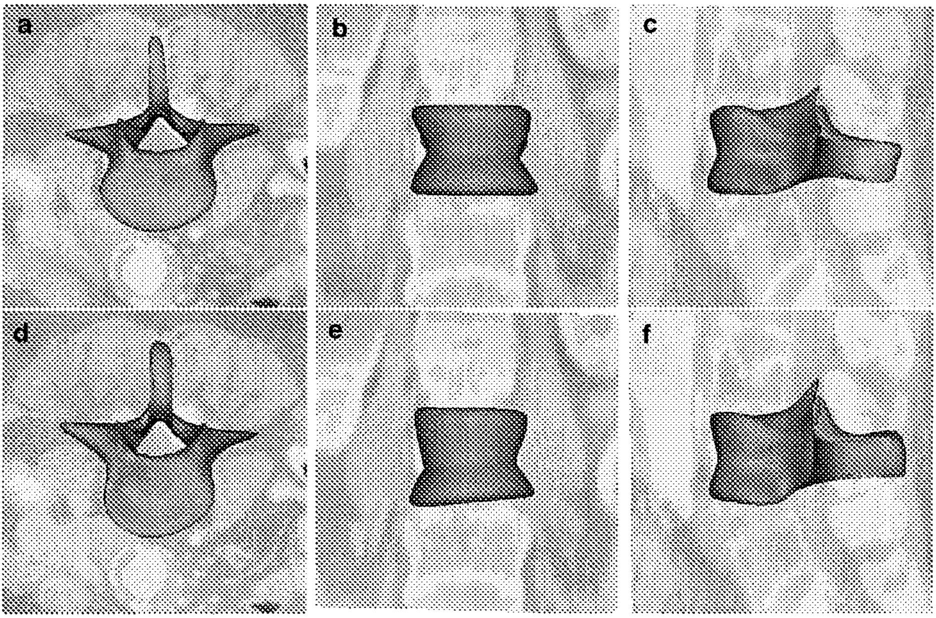

FIG. 9. An example segmentation with initial and final deformable surface superimposed on the intensity slices through the volume data. Top row shows the initial situation: (a) upper view, (b) frontal view, (c) side view. Bottom row shows result after deformation: (d) upper view, (e) frontal view, (f) side view.

model. As the planes cut the deformable surface, only parts of it in front of the plane are visible.

The top row in Fig. 9 illustrates the initial deformable surface from three different perspectives (see Fig 8 for context). The bottom row illustrates the segmentation result. The images show how the deformable surface correctly moves to the upper and lower surfaces of the vertebral body. At some parts the model moves only slightly toward the target boundary or even moves away from it to comply with the population statistics.

5. DISCUSSION AND CONCLUSION

In conclusion, in this work we present an inhomogeneous and point-enhanced deformable model called a necklace. The necklace aims at exploiting salient information in images containing sophisticated objects and constellations. Saliency is determined on the basis of multiple features observed along object boundaries in a given training set. Multiple features from multiple objects, forming a collection of manifolds in multidimensional feature space, are averaged and inspected for landmarks on the basis of the second order structure of the population average. A distinction is made between point landmarks, curve landmarks, and sheets points depending on the number of dimensions with extreme values for the curvature of the manifolds. Along the entire manifold features are selected that give saliency. They are subsequently employed in a priority scheme to search for boundaries in new unknown images. Point landmarks are detected in three-dimensional image areas, plausibly bootstrapped by interaction. Starting from detected point landmarks, curve and sheet landmarks are detected in respectively two-dimensional and one-dimensional areas. A final detection of all points provides a global segmentation solution.

The method has in common with other landmark-based methods such as [8, 21] that it exploits salient image and shape information for segmentation. However, in contrast to mainstream landmark-based segmentation, we focus not on a single basis for a landmark but rather on trying to learn landmarks emanating from multiple features. What makes our approach different from methods with the same aim such as [27] is that we cast more than just one feature in a multivariate functional feature space to identify a repertoire of landmarks automatically. Apart from this, feature selection is carried out to exploit the most discriminative feature at each point along the object boundary. This way the uniqueness of features is exploited as much as possible along the entire boundary. Furthermore, to make segmentation less distracted and more efficient the search for boundaries is performed in relevant dimensions only, specified by the directions in which points are uniquely defined.

The necklace method is suited for interactive segmentation. Point landmarks are sparse and have a unique zero-dimensional character forming excellent conditions for steering purposes in three dimensions. They permit the user to conveniently perform segmentation by easily making a one-to-one correspondence between point landmarks on the deformable surface and point landmarks in the image, enhancing indigenous integration of interaction and computation to avoid graphically editing solutions in a post mortem sense. Where the experiment results suggest that a better localization of point-landmarks improves the performance of the necklace method it may give the operator a natural place in accrediting the whole segmentation process.

In the experiments described in the previous section we determine landmarks on the basis of a single feature, the curvature, and apply two features for fitting the deformable surface to the image data. We did not investigate how the necklaces behave with landmark selection from a large number of features. It is straightforward to extend Eq. (23) to the case where more features are selected at each point. We expect that more accurate results are obtained by (a) tuning v_A, v_B, v_C locally such that features along the entire surface contribute to the definition of the object at all steps of the priority scheme and (b) reestimating the positions of deformable surface points at all steps of the priority scheme rather than fixing them. In the definition of features a number of other issues remain unexplored when the sign of the principal curvatures is included in the definition of landmarks. Also the influence of the path parameterization and the number and distribution of the sample points over the surface requires more study. We leave these aside for future work.

We arrive at the conclusion that the necklace method, particularly when used in combination with user interaction, may help segmentation of complex scenes where a multidimensional feature set is needed to properly capture a boundary locally. Necklaces work well when object segmentation using one or two straightforward features fails due to lack of a simple definition of the object boundary.

REFERENCES

1. P. J. Besl and R. C. Jain, Invariant surface characteristics for 3D object recognition in range images, *Comput. Vision Graphics Image Process.* **33**(1), 1986, 33–80.
2. F. L. Bookstein, Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.* **11**(6), 1989, 567–585.
3. R. P. Brent, *Algorithms for Minimizationwithout Derivatives*, Prentice–Hall, Englewood Cliffs, NJ, 1973.

4. I. Cohen, L. D. Cohen, and N. J. Ayache, Using deformable surfaces to segment 3-d images and infer differential structures, in *Proceedings of European Conference on Computer Vision*, 1992, pp. 648–652.
5. L. D. Cohen, On active contour models and balloons. *Comput. Vision Graphics and Image Process.* **53**(2), 1991, 211–218.
6. L. D. Cohen and R. Kimmel, Global minimum for active contour models: A minimal path approach, *Int. J. Comput. Vision* **24**(1), 1997, 57–78.
7. G. de Vries and P. W. Verbeek, Scale-adaptive landmark detection, classification and size estimation in 3d object-background images, in *Proceedings of International Conference on Pattern Recognition*, *2000*, Vol. III, pp. 1026–1029.
8. J. Declerck, G. Subsol, J. P. Thirion, and N. Ayache, Automatic retrieval of anatomical structures in three-dimensional medical images, in *Proceedings of Computer Vision, Virtual Reality and Robotics in Medicine, 1995*.
9. B. S. Duran and P. K. Odell, *Cluster Analysis—A Survey*, Springer-Verlag, Berlin/New York, 1974.
10. D. Eberly, R. Gardner, B. Morse, S. Pizer, and C. Scharlach, Ridges for image analysis, *Math. Imaging Vision* **4**, 1994, 353–373.
11. T.-J. Fan, G. Medioni, and R. Nevatia, Segmented description of 3-d surfaces. *IEEE Trans. Robot. Automat.* **32**, 1987, 527–538.
12. G. Farin, *Curves and Surfaces for Computer Aided Geometric Design*, Academic Press, Boston, 1993.
13. S. Frantz, K. Rohr, and H. S. Stiehl, Multi-step procedures for the localization of 2-d and 3-d point landmarks and automatic roi size selection, in *Proceedings of European Conference on Computer Vision*, *1998*.
14. S. Ghebreab and A. W. M. Smeulders, Strings: Variational deformable models of multivariate ordered features, *IEEE Trans. Pattern Anal. Mach. Intell.*, to appear.
15. P. Giblin and B. B. Kimia, A formal classification of three-dimensional medial axis points and their local geometry, in *Proceedings of IEEE Computer Vision Patern Recognition Conference*, *2000*, Vol. I, pp. 560–573.
16. L. Haglund, *Adaptive Multi-dimensional Filtering*, Ph.D. thesis, 1992.
17. Kenneth M. Hanson, Ed., *Shape-Based Interactive Three-Dimensional Medical Image Segmentation*, Vol. 3034, 1997.
18. J. J. Koenderink, *Solid Shape*. MIT Press, Cambridge, MA, 1989.
19. R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in *Proceedings of International Joint Conference on Artificial Intelligence*, *1995*.
20. A. Lanitis, C. J. Taylor, and T. F. Cootes, Automatic Interpretation and coding of face images using flexible models. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**(7), 1997, 743–756.
21. C. Lorenz and N. Krahnstver, Generation of point-based three-dimensional statistical shape models for anatomical objects. *Comput. Vision Image Understanding* **77**(2), 2000, 175–191.
22. M. Nakahara, *Geometry, Topology and Physics*, Adam Hilger, Bristol/New York, 1998.
23. W. Niessen, *Multiscale Medical Image Analysis*, Ph.D. thesis, Utrecht University, 1997.
24. R. Pichumani, *Construction of a Three-Dimensional Geometric Model for Segmentation and Visualization of Cervical Spine Images*, Ph.D. thesis, Stanford University, 1997.
25. L. Piegl and W. Tiller, *The NURBS Book*, Springer-Verlag, Berlin, 1997.
26. K. Rohr, Landmark-based elastic matching of three-dimensional medical images, in *Proceedings of Second Germany–Korea Joint Conf. on Advanced Medical Image Processing, 1997*, pp. 60–63.
27. K. Rohr, *Landmark-Based Image Analysis: Using Geometric and Intensity Models*, Kluwer, Academic, Dondrecht, 2001.
28. K. Rohr, H. S. Stiehl, R. Sprengel, W. Beil, T. M. Buzug, J. Weese, and M. H. Kuhn, *Point-Based Elastic Registration of Medical Image Data Using Approximating Thin-Plate Splines*, pp. 297–306, Springer-Verlag, Berlin/Heidelberg, 1996.
29. O. Shinagawa, T. T. Kunii, A. G. Belyaev, and T. Tsukioka, Shape modeling and shape analysis based on singularities. *Int. J. Shape Modeling* **2**, 1996, 85–102.
30. L. H. Staib and J. S. Duncan, Boundary finding with parametrically deformable models, *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(11), 1992, 1061–1075.

31. D. J. Struik, *Lectures on Classical Differential Geometry*, Addison–Wesley, Reading, MA, 1950.

32. D. Terzopoulos and D. Metaxas, Constraining deformable superquadrics and nonrigid motion tracking, in *Proceedings of IEEE Computer Vision Patern Recognition Conference, 1991*, pp. 337–343.

33. J. P. Thirion, Extremal points: Definition and application to three-dimensional image registration, in *Proceedings of IEEE Computer Vision Patern Recognition Conference, 1994*, pp. 587–592.

34. Y. M. Wang and L. H. Staib, Boundary finding with prior shape and smoothness models. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(7), 2000, 738–743.

35. C. F. Westin, A. Bhalerao, H. Knutsson, and R. Kikinis, Using local three-dimensional structure for segmentation of bone from computer tomography images, in *Proceedings of IEEE Computer Vision Patern Recognition, 1997*, pp. 794–800.