

Shape Deformation: SVM Regression and Application to Medical Image Segmentation

Song Wang, Weiyu Zhu and Zhi-Pei Liang
Department of ECE and Beckman Institute

University of Illinois at Urbana-Champaign, IL 61801, U.S.A.

songwang@uiuc.edu, weiyuzhu@uiuc.edu, z-liang@uiuc.edu

Abstract

This paper presents a novel landmark-based shape deformation method. This method effectively solves two problems inherent in landmark-based shape deformation: (a) identification of landmark points from a given input image, and (b) regularized deformation of the shape of an object defined in a template. The second problem is solved using a new constrained support vector machine (SVM) regression technique, in which a thin-plate kernel is utilized to provide non-rigid shape deformations. This method offers several advantages over existing landmark-based methods. First, it has a unique capability to detect and use multiple candidate landmark points in an input image to improve landmark detection. Second, it can handle the case of missing landmarks, which often arises in dealing with occluded images. We have applied the proposed method to extract the scalp contours from brain cryosection images with very encouraging results.

1 Introduction

Image segmentation is an important task in image understanding and computer vision. Traditional methods are usually based on edge detection, region merging/splitting, statistical pixel classification or hybrid methods [19]. Although these methods are capable of partitioning an image into regions of connected pixels according to image intensity distributions, more often than not, the segmented regions do not correspond well to true objects. This limitation is particularly serious for medical image segmentation because medical images are often very noisy and the structures to be identified have large “internal” intensity variations.

An effective way to solve this problem is to incorporate the known shape information of the desired objects into the segmentation process. Generally, the shape of an object is described by the boundary contour (or several closed contours) of the objects, which can be represented by a sequence of sampled landmark points.

A number of shape-based segmentation methods have

been proposed. For example, the active contour (or snakes) [8, 1, 17, 4] model constrains the desired shape to be sufficiently continuous, smooth and differentiable during the deformation process. Although the original active contour cannot process multiple contours, the problem can be solved using the recent geodesic contour method [3].

More advanced shape models include the point distribution model (PDM) by Cootes et al.[5], which can learn shape variations from a set of segmented training images, each containing a set of landmarks to define the shape. PDM calculates the covariance matrix of these landmarks and then represents the shape variations along the directions of the most significant eigenvectors of the covariance matrix. However, it is a tedious job to build the training set manually and in some cases it is also difficult to identify the set of landmark points for all the shapes of interest.

Similarly to PDM, Staib et al. [13] decompose the shape into items with different frequencies. Then the shape knowledge is learned by studying the distribution of each Fourier coefficient. The result is used to constrain the shape deformation. Jain [7] presents a similar approach which parameterizes the shape and uses the distribution of the parameters as the constraint for shape deformation. Along the same line, Leventon et al. [10] incorporate statistical shape information into geodesic active contours to segment medical images.

The work presented in the paper is more closely related to the deformable models developed by Rueckert et al. [12] and Zhong et al. [18]. Similar to the active contour, the energy function of those models consists of two terms accounting for external and internal energy respectively. The external energy is a potential field describing the edge and region matching information between the template and the input image. The internal energy represents the shape deviation from the prior. However, it is not easy to determine the appropriate weight for the internal and external energy. Additionally, this energy function is nonlinear and includes too many unknowns to be well handled by gradient-based algorithms.

This paper presents a new landmark-based shape deformation method which can effectively avoid the above ques-

tions. At first, edge- or/and region-based methods are applied to detect candidate landmark points in the input image within the neighboring area around each landmark in the template. Multiple candidates may be detected for one landmark when it is necessary. Then the template is deformed using a special support vector machine (SVM) regression technique. Based on the fitting result, a set of better landmarks is identified from the detected candidates. This process is repeated until the final result is obtained. Usually, a couple of iterations is sufficient.

The rest of the paper is organized as follows. In Section 2, we describe the proposed shape-based deformation method. Section 3 discusses the selection of the regularization parameter and how to handle missing landmarks. Section 4 presents representative results for medical image segmentation, followed by the conclusion in Section 5.

2 The Proposed Method

2.1 Problem formulation

For simplicity, consider the case in which the template contains only a single closed contour. Extension to the case of multiple contours is straightforward. The contour is represented as a series of ordered landmark points $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$, where $\mathbf{v}_i = (x_i, y_i)$ are the coordinates of the i -th landmark. For each landmark \mathbf{v}_i , the proposed method first identifies a set of possible corresponding landmark points $\mathcal{B}_i = \{\mathbf{v}_i^{(j)}, j = 1, 2, \dots, n_i\}$ in the input image, where $\mathbf{v}_i^{(j)} = (x_i^{(j)}, y_i^{(j)})$. Then the problem is solved in two steps:

1. Identify the best landmark point $\mathbf{v}'_i = (x'_i, y'_i)$ from the landmark set \mathcal{B}_i such that $\mathbf{V}' = \{\mathbf{v}'_1, \mathbf{v}'_2, \dots, \mathbf{v}'_n\}$ located in or near the real object boundary in the input image;
2. Deform the prior shape \mathbf{V} to match \mathbf{V}' while keeping the general shape characteristics of \mathbf{V} .

The template deformation is described as a regression problem of finding a transform $\mathbf{t} = (f, g) : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ that minimizes

$$\frac{1}{n} \sum_{i=1}^n Q(\mathbf{v}'_i, \mathbf{t}(\mathbf{v}_i)) + \lambda \phi[\mathbf{t}], \quad (1)$$

where $Q(\cdot, \cdot) \geq 0$ is a loss function, $\lambda \geq 0$ is a regularization parameter and $\phi[\mathbf{t}] \geq 0$ is a regularization functional. In this setting, $\mathbf{t}(\mathbf{V}) = \{\mathbf{t}(\mathbf{v}_i), i = 1, 2, \dots, n\}$ is the desired object shape in the input image.

Generally, it is a combinatorial problem to determine the best \mathbf{V}' . In this paper, we use an EM-like (Expectation Minimization) two-step algorithm to solve it iteratively. Specifically, we perform the shape-regularized fitting to an initial \mathbf{V}' to get an optimal \mathbf{t} and then update \mathbf{V}' from detected

\mathcal{B}_i based on the acquired \mathbf{t} . This process is repeated until convergence.

There are three main subproblems in this approach: detection of landmark candidates \mathcal{B}_i in the input image, determination of the regression model and updating \mathbf{V}' based on the regularized fitting results. Solution to these problems are discussed below.

2.2 Detection of landmark candidates

Given \mathbf{V} , we assume that \mathbf{v}'_i in an input image falls in a circular area of radius r_i centered at \mathbf{v}_i . According to the principle of anisotropic diffusions [11], non-rigid biologic shape deformation between the template and the input image can be done effectively by moving each landmark \mathbf{v}_i along the normal direction of the shape. This means that we can search for the corresponding \mathbf{V}' only along the normal directions of \mathbf{V} as shown in Fig.1. Denote \mathbf{n}_i as the

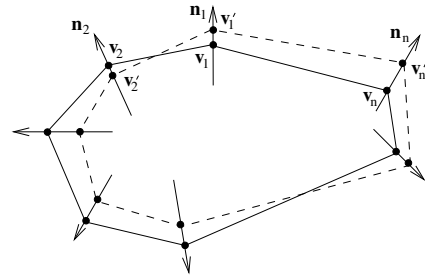


Figure 1. Search for landmarks along the normal direction of the initial shape.

unit vector along the normal direction at \mathbf{v}_i . Combining the above assumptions, the corresponding landmark \mathbf{v}'_i can be obtained from the line segment $\mathcal{L}_i = \{\mathbf{v}_i + r\mathbf{n}_i, r \in [-r_i, r_i]\}$.

Because of noise and structural complexity, sometimes it is difficult to accurately identify \mathbf{v}'_i from \mathcal{L}_i . In this case, we can extract several candidates in $\mathcal{B}_i = \{\mathbf{v}_i^{(j)}, j = 1, 2, \dots, n_i\}$ and the proposed algorithm will determine an optimal \mathbf{v}'_i from \mathcal{B}_i .

For different application problems, various methods can be used to locate $\mathcal{B}_i, i = 1, 2, \dots, n$. Generally, they can be categorized into two classes.

1. Edge detection methods. Since the desired shape is the contour of an object in the input image, edge points in \mathcal{L}_i will be collected to construct \mathcal{B}_i . Any edge detection algorithm can be applied here. A simplest approach is to calculate the gradient vector for each pixel along \mathcal{L}_i . If its amplitude is very large (according to a pre-selected threshold), it will be included in \mathcal{B}_i .
2. Region matching methods. For any pixel along \mathcal{L}_i , if its neighboring area has similar features with the

neighboring area centered at \mathbf{v}_i in the template image, it will be included in \mathcal{B}_i . A number of matching metrics can be used for this problem, which include normalized cross-correlation, squared differences in pixel intensities, measures based on optical flow models and mutual information metrics [14].

Generally, region matching is often more robust than edge detection to process noisy images containing complex objects. However, region matching methods usually require the template landmarks \mathbf{V} to be very accurate, which is not necessary for edge detection methods.

Selection of r_i is related to the similarity between the template and the input image. For segmenting 3D medical images slice by slice, this is usually determined by the sampling interval between two neighboring slices. The larger the interval, the larger the r_i . In practice, r_i should also be adaptively modified based on the size of \mathcal{B}_i . If there are too many candidates in \mathcal{B}_i , then r_i needs to be reduced, and vice versa.

It is also possible that no landmark candidate is identified for some \mathbf{v}_i . This problem will be discussed in Section 3.2.

2.3 SVM regression model for deformation

Given a set of estimated landmarks \mathbf{V}' from $\{\mathcal{B}_i, i = 1, 2, \dots, n\}$, this section describes a regression model for (1) and the corresponding algorithm. Generally, we expect the regression model to have the following properties: (a) $\mathbf{t}(\mathbf{V})$ is a good approximation for \mathbf{V}' ; (b) $\mathbf{t}(\cdot)$ is a geometrically homologous mapping (an important feature in biologic deformation); (c) $\phi[\mathbf{t}]$ represents well the biologic deformation between \mathbf{V} and $\mathbf{t}(\mathbf{V})$; (d) $\phi[\mathbf{t}]$ is invariant under the affine or rigid transforms; and (e) The loss function $Q(\cdot, \cdot)$ is robust against noise and outliers;

From regularization theory, $\phi[\mathbf{t}]$ can be defined as a norm in a reproducing kernel Hilbert space (or subspace) which can be uniquely represented by a positive definite (or conditionally positive definite) kernel function $K(\mathbf{v}, \mathbf{v}')$. For medical image segmentation, we propose to use the thin-plate kernel [2] $K(\mathbf{v}, \mathbf{v}') = \|\mathbf{v} - \mathbf{v}'\|^2 \log \|\mathbf{v} - \mathbf{v}'\|$, where $\phi[\mathbf{t}]$, also named the *bending energy*, is given by

$$\phi[\mathbf{t}] = \iint_{-\infty}^{\infty} (L(f) + L(g)) dx dy \quad (2)$$

where $L(\cdot) = (\frac{\partial^2}{\partial x^2})^2 + 2(\frac{\partial^2}{\partial x \partial y})^2 + (\frac{\partial^2}{\partial y^2})^2$.

A popular loss function is $Q(\mathbf{v}'_i, \mathbf{t}(\mathbf{v}_i)) = \|\mathbf{v}'_i - \mathbf{t}(\mathbf{v}_i)\|^2$. In this case, the model becomes the traditional thin-plate splines [2]. However, it is well known that the squared error loss function is not robust against outliers. To avoid this problem, various robust loss functions have been developed. Two examples are the modulo function and the Huber's function [6]. Here we adopt the linear ϵ -insensitive function used in SVM [15] for the following reasons: (a) It is more robust than the squared error loss function; (b) With

the insensitive margin parameter ϵ , landmark points can be identified as support vectors or non-support vectors. This result can be used to improve \mathbf{V}' . This step is discussed in detail in Section 2.4.

The ϵ -insensitive function is defined by

$$\begin{aligned} & \|\mathbf{v}'_i - \mathbf{t}(\mathbf{v}_i)\|_\epsilon \\ = & \begin{cases} 0 & \text{if } \|\mathbf{v}'_i - \mathbf{t}(\mathbf{v}_i)\| \leq \epsilon \\ \|\mathbf{v}'_i - \mathbf{t}(\mathbf{v}_i)\| - \epsilon & \text{else.} \end{cases} \end{aligned}$$

When $\epsilon = 0$, this is just the modulo loss function.

The only remaining component of the regression model is the additional constraints that should be placed on the desired deformation $\mathbf{t}(\cdot)$. Obviously, we hope that the desired $\mathbf{t}(\mathbf{v}_i)$ lie in the normal direction of the shape centered at \mathbf{v}_i , i.e.,

$$\mathbf{t}(\mathbf{v}_i) = \mathbf{v}_i + \gamma_i \mathbf{n}_i, i = 1, 2, \dots, n \quad (3)$$

where $\gamma_i \in \mathbb{R}$.

Since \mathbf{V}' is detected along the normal directions of \mathbf{V} , we have

$$\mathbf{v}'_i = \mathbf{v}_i + h_i \mathbf{n}_i, i = 1, 2, \dots, n \quad (4)$$

where $h_i \in \mathbb{R}, i = 1, 2, \dots, n$ are known. Noting that $\mathbf{t} = (f, g)$ and $\|\mathbf{n}_i\| = 1, i = 1, 2, \dots, n$, the regression problem can be rewritten as

Problem 1:

$$\min_{\mathbf{t}, \gamma} \Phi(\mathbf{t}, \gamma) = \frac{1}{n} \sum_{i=1}^n |h_i - \gamma_i|_\epsilon + \lambda \phi[\mathbf{t}]$$

subject to constraints (3), where $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_n)^T$.

The thin-plate kernel is conditionally positive definite and the linear functions form the null space of the resulting $\mathbf{t} = (f, g)$, which must be of the form [2]

$$\begin{cases} f(\mathbf{v}) = a_1 + a_2 x + a_3 y + \sum_{i=1}^n c_i K(\mathbf{v}, \mathbf{v}_i) \\ g(\mathbf{v}) = b_1 + b_2 x + b_3 y + \sum_{i=1}^n d_i K(\mathbf{v}, \mathbf{v}_i) \end{cases} \quad (5)$$

To solve Problem 1, we first fix γ and minimize $\Phi(\mathbf{t}, \gamma)$ with respect to \mathbf{t} under constraints (3). This is equivalent to seeking \mathbf{t} minimizing $\iint_{-\infty}^{\infty} (L(f) + L(g)) dx dy$ subject to the constraint that \mathbf{t} must map \mathbf{v}_i to $(\hat{x}_i, \hat{y}_i) = \mathbf{v}_i + \gamma_i \mathbf{n}_i, i = 1, 2, \dots, n$. It is a typical thin-plate interpolation problem and the parameters $\mathbf{a} = (a_1, a_2, a_3)^T, \mathbf{b} = (b_1, b_2, b_3)^T, \mathbf{c} = (c_1, c_2, \dots, c_n)^T$ and $\mathbf{d} = (d_1, d_2, \dots, d_n)^T$ in (5) can be calculated by solving the following equation:

$$\begin{pmatrix} \mathbf{K} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{c} & \mathbf{d} \\ \mathbf{a} & \mathbf{b} \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}} & \hat{\mathbf{y}} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \quad (6)$$

where $k_{ij} = K(\mathbf{v}_i, \mathbf{v}_j), i, j = 1, 2, \dots, n$ are the elements of matrix \mathbf{K} and $\mathbf{P} = (1, \mathbf{x}, \mathbf{y})$. Note that $\mathbf{x} = (x_1, x_2, \dots, x_n)^T, \mathbf{y} = (y_1, y_2, \dots, y_n)^T, \mathbf{v}_i = (x_i, y_i), \hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n)^T$ and $\hat{\mathbf{y}} = (\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n)^T$.

In the above thin-plate interpolation, the resulting minimal bending energy is given by

$$\phi[\mathbf{t}] = \mathbf{c}^T \mathbf{K} \mathbf{c} + \mathbf{d}^T \mathbf{K} \mathbf{d} = \frac{1}{8\pi} (\hat{\mathbf{x}}^T \mathbf{L} \hat{\mathbf{x}} + \hat{\mathbf{y}}^T \mathbf{L} \hat{\mathbf{y}}), \quad (7)$$

where \mathbf{L} is the $n \times n$ upper left sub-matrix of

$$\begin{pmatrix} \mathbf{K} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{pmatrix}^{-1} \quad (8)$$

which is positive semi-definite.

Substituting (7) into Problem 1 yields the following unconstrained problem.

Problem 2:

$$\min_{\gamma} \Phi(\gamma) = \frac{1}{n} \sum_{i=1}^n |h_i - \gamma_i|_{\epsilon} + \frac{\lambda}{8\pi} (\hat{\mathbf{x}}^T \mathbf{L} \hat{\mathbf{x}} + \hat{\mathbf{y}}^T \mathbf{L} \hat{\mathbf{y}})$$

Using the standard SVM technique, we can formulate the problem as follows.

Problem 3:

$$\begin{aligned} \min_{\gamma, \xi, \hat{\xi}} \Phi(\gamma, \xi, \hat{\xi}) &= \frac{1}{n} \sum_{i=1}^n (\xi_i + \hat{\xi}_i) \\ &+ \frac{\lambda}{8\pi} \left\{ (\mathbf{x} + \mathbf{N}_x \gamma)^T \mathbf{L} (\mathbf{x} + \mathbf{N}_x \gamma) \right. \\ &\left. + (\mathbf{y} + \mathbf{N}_y \gamma)^T \mathbf{L} (\mathbf{y} + \mathbf{N}_y \gamma) \right\} \end{aligned}$$

subject to

$$\begin{aligned} h_i - \gamma_i &\leq \epsilon + \hat{\xi}_i & i = 1, 2, \dots, n \\ \gamma_i - h_i &\leq \epsilon + \xi_i & i = 1, 2, \dots, n \\ \xi_i, \hat{\xi}_i &\geq 0 & i = 1, 2, \dots, n \end{aligned}$$

where $\xi = (\xi_1, \xi_2, \dots, \xi_n)^T$, $\hat{\xi} = (\hat{\xi}_1, \hat{\xi}_2, \dots, \hat{\xi}_n)^T$, $\mathbf{N}_x = \text{diag}(N_{1x}, N_{2x}, \dots, N_{nx})$ and $\mathbf{N}_y = \text{diag}(N_{1y}, N_{2y}, \dots, N_{ny})$ with $\mathbf{n}_i = (N_{ix}, N_{iy})$.

Introduce the Lagrange multipliers $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$, $\hat{\alpha} = (\hat{\alpha}_1, \hat{\alpha}_2, \dots, \hat{\alpha}_n)^T$, $\beta = (\beta_1, \beta_2, \dots, \beta_n)^T$ and $\hat{\beta} = (\hat{\beta}_1, \hat{\beta}_2, \dots, \hat{\beta}_n)^T$, the Lagrangian function for Problem 3 is

$$\begin{aligned} \mathcal{W}(\gamma, \xi, \hat{\xi}, \alpha, \hat{\alpha}, \beta, \hat{\beta}) &= \\ C \sum_{i=1}^n (\xi_i + \hat{\xi}_i) &+ \frac{1}{2} \left\{ (\mathbf{x} + \mathbf{N}_x \gamma)^T \mathbf{L} (\mathbf{x} + \mathbf{N}_x \gamma) \right. \\ &\left. + (\mathbf{y} + \mathbf{N}_y \gamma)^T \mathbf{L} (\mathbf{y} + \mathbf{N}_y \gamma) \right\} \\ - \sum_{i=1}^n \alpha_i (\epsilon + \xi_i - \gamma_i + h_i) & \\ - \sum_{i=1}^n \hat{\alpha}_i (\epsilon + \hat{\xi}_i + \gamma_i - h_i) &- \sum_{i=1}^n (\beta_i \xi_i + \hat{\beta}_i \hat{\xi}_i) \end{aligned}$$

where $C = \frac{4\pi}{n\lambda}$.

Imposing the first-order necessary conditions (setting the derivatives of the Lagrangian \mathcal{W} with respect to γ , ξ and $\hat{\xi}$ to zero) yields

$$\gamma = \mathbf{H}(\hat{\alpha} - \alpha) - \mathbf{p} \quad (9)$$

and

$$\begin{aligned} \alpha_i + \beta_i &= C & i = 1, 2, \dots, n \\ \hat{\alpha}_i + \hat{\beta}_i &= C & i = 1, 2, \dots, n \\ \alpha_i, \hat{\alpha}_i, \beta_i, \hat{\beta}_i &\geq 0 & i = 1, 2, \dots, n \end{aligned}$$

where

$$\begin{aligned} \mathbf{H} &= (\mathbf{N}_x^T \mathbf{L} \mathbf{N}_x + \mathbf{N}_y^T \mathbf{L} \mathbf{N}_y)^{-1} \\ \mathbf{p} &= \mathbf{H}(\mathbf{N}_x^T \mathbf{L} \mathbf{x} + \mathbf{N}_y^T \mathbf{L} \mathbf{y}). \end{aligned}$$

Substituting these results into Problem 3 gives the following equivalent formulation

Problem 4:

$$\begin{aligned} \min_{\alpha, \hat{\alpha}} \hat{\mathcal{W}}(\alpha, \hat{\alpha}) &= \frac{1}{2} (\alpha - \hat{\alpha})^T \mathbf{H} (\alpha - \hat{\alpha}) \\ &+ (\mathbf{p} + \mathbf{h})^T (\alpha - \hat{\alpha}) + \mathbf{e}^T (\alpha + \hat{\alpha}) \end{aligned} \quad (10)$$

subject to

$$0 \leq \alpha_i, \hat{\alpha}_i \leq C \quad i = 1, 2, \dots, n$$

where $\mathbf{h} = (h_1, h_2, \dots, h_n)^T$ and $\mathbf{e} = (\epsilon, \epsilon, \dots, \epsilon)^T$.

Problem 4 is a simple quadratic programming problem which can be solved using standard algorithms. After the optimal $\alpha, \hat{\alpha}$ are obtained, the desired γ in Problem 3 can be calculated from (9). Finally, we apply (6) and (5) to get the desired \mathbf{t} .

2.4 Updating \mathbf{V}'

The next step in our iterative algorithm is to improve the estimation of \mathbf{V}' from $\mathcal{B}_i, i = 1, 2, \dots, n$ based on the results from the shape fitting step. To do so, we invoke the Kuhn-Tucker condition for Problem 4, which states

$$\begin{aligned} \alpha_i (\gamma_i - h_i - \epsilon - \xi_i) &= 0 & i = 1, 2, \dots, n \\ \hat{\alpha}_i (\gamma_i - h_i + \epsilon + \hat{\xi}_i) &= 0 & i = 1, 2, \dots, n \\ (C - \alpha_i) \xi_i &= 0 & i = 1, 2, \dots, n \\ (C - \hat{\alpha}_i) \hat{\xi}_i &= 0 & i = 1, 2, \dots, n. \end{aligned}$$

It is obvious that $\alpha_i \cdot \hat{\alpha}_i = 0, i = 1, 2, \dots, n$. The input data \mathbf{v}_i for which α_i or $\hat{\alpha}_i$ have nonzero values are called *support vectors*. Otherwise, \mathbf{v}_i is a *non-support vector*.

As shown in Fig.2, if \mathbf{v}_i is a non-support vector, then the current \mathbf{v}'_i lies inside the ϵ -insensitive margin centered at the resulting $\mathbf{t}(\mathbf{V})$, i.e., $|\gamma_i - h_i| \leq \epsilon$. From (10), it also easy to see that \mathbf{v}'_i will have no effects on the final regression and has no contribution to the bending energy. In

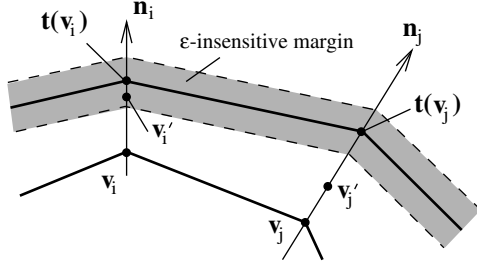


Figure 2. Support vector v_j and non-support vector v_i .

this case, the current v_i' can be regarded as being accurate enough with the permissible error margin ϵ . On the other hand, any support vector lying outside the margin need to be improved if possible. Based on this consideration, the following simple exchanging strategy is adopted to update V' .

1. For non-support vector v_i , we leave the current v_i' unchanged;
2. For support vector v_i , if another candidate in \mathcal{B}_i lies in the ϵ -insensitive margin, then we adopt it as the new v_i' .

This strategy will reduce the bending energy if ϵ is fixed.

The remaining problem is the selection of ϵ . Inspired by the area-shrinking technique used in SOM (self-organizing mapping), we reduce ϵ gradually in the iteration process until it approaches zero. This is quite reasonable since the acquired $t(V)$ is usually inaccurate at the beginning of the iteration for some mis-selected v_i' . Thus a large ϵ may increase the chance of including the true V' inside the insensitive margin while eliminating bad outliers. As $t(V)$ becomes more and more accurate, we can reduce ϵ to improve the resolution of the final results, which is similar to what is done in multiscale image analysis [9].

2.5 Summary of the algorithm

The proposed algorithm is summarized below.

1. Based on the template shape V , detect a landmark candidate set $\mathcal{B}_i, i = 1, 2, \dots, n$ in the input image along the normal direction of V .
2. Randomly select V' from $\mathcal{B}_i, i = 1, 2, \dots, n$ and initialize ϵ ;
3. Perform SVM regression using a thin-plate kernel. Find the deformed shape $t(V)$ and label the support vectors and non-support vectors;
4. Update V' by replacing the support vectors with better candidates in $\mathcal{B}_i, i = 1, 2, \dots, n$. If no one is updated, reduce ϵ . Return to step 3. If no one is updated and

$\epsilon = 0$, stop the algorithm and return current $t(V)$ as the desired shape.

3 Discussion

This section discusses two related issues in practical application of the proposed algorithm: (a) how to select the regularization parameter λ (or C which bounds the α and $\hat{\alpha}$), and (b) how to deal with the case when some \mathcal{B}_i is empty.

3.1 Selection of λ

The regularization parameter λ plays an important role in the deformation. The optimal λ is chosen for our problem by minimizing the ordinary cross-validation (OCV) function

$$V_0(\lambda) = \frac{1}{n} \sum_{i=1}^n \|v_i' - t_{\lambda}^{[k]}(v_i)\|_{\epsilon}$$

where $t_{\lambda}^{[k]}$ is the minimizer of Problem 1 except that the k th landmark point is left out, i.e., $t_{\lambda}^{[k]}$ minimizes

$$\frac{1}{n} \sum_{i=1, i \neq k}^n \|v_i' - t(v_i)\|_{\epsilon} + \lambda \phi[t] \quad (11)$$

subject to constraints (3). This is obtained using the principle of "leave-out-one" cross-values for cross validation. In this case, the v_k' is also set as an unknown which is allowed to vary freely along the normal direction n_i . Similar to the analysis in Section 2.3, solution to this problem can be obtained by solving the quadratic programming in Problem 4 together with two additional constraints

$$\alpha_k, \hat{\alpha}_k = 0.$$

In this case, v_k are fixed to be a non-support vector and h_k (so v_k') will not affect the results. Based on the acquired α and $\hat{\alpha}$, we can apply (9), (6), and (5) to get γ and t .

Wahba [16] has conducted a thorough study on how to estimate λ for the scalar function approximation splines and also extended the OCV to generalized cross validation (GCV). The main principle can also be adapted to select optimal λ for our SVM-based shape deformation.

3.2 No landmark candidate

It is possible that no landmark candidates can be identified for some v_i . Assume $\mathcal{P} = \{i | 1 \leq i \leq n, \mathcal{B}_i = \emptyset\}$. In this case, we can only use the remaining landmarks detected. According to the resulting function t , the undetected landmarks can be directly calculated from $t(v_i)$. The problem can be formulated as minimizing

$$\frac{1}{n} \sum_{i=1, i \notin \mathcal{P}}^n \|v_i' - t(v_i)\|_{\epsilon} + \lambda \phi[t]$$

subject to constraints (3).

This can be regarded as the “leave-several-out” test which is a generalization of (11) and has a similar solution. At first, we solve Problem 4 with additional constraints

$$\alpha_j, \hat{\alpha}_j = 0; \quad \forall j \in \mathcal{P},$$

which disable $h_j, j \in \mathcal{P}$. Then based on the acquired α and $\hat{\alpha}$, we can use (9), (6) and (5) to get the optimal γ and t .

This property is useful for two reasons: (a) We can purposely exclude “bad” landmark candidate sets \mathcal{B}_i in the fitting step. (b) When the object to be segmented is partly occluded, this method can combine the object shape in the template to get a reasonable and accurate segmentation.

However, it should be noted that the number of non-empty \mathcal{B}_i 's must be sufficient large to capture the shape deformation between the template and the input image. Determining the minimum number of landmark points required for a given shape has not been solved.

4 Experiments

We have used the proposed method to extract the scalp contour from a series of brain cryosection images from the Visual Human Project. Fig.3(a) shows the initial template slice with the manually extracted scalp contour composed of 100 landmark points distributed uniformly.

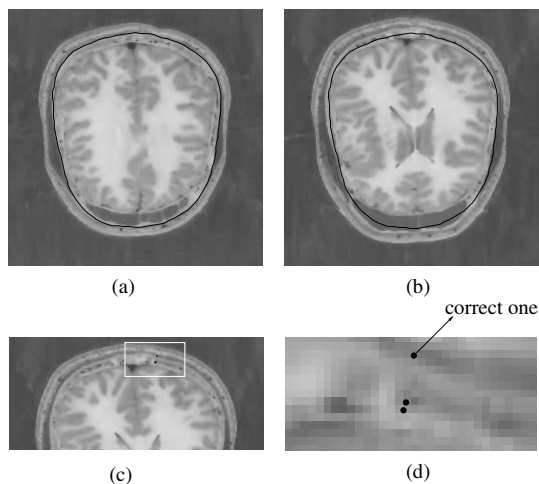


Figure 3. (a) Initial template image with the manually extracted scalp contour; (b) The input image with the unprocessed scalp contour from the template overlaid; (c) Detected \mathcal{B}_i based on one special v_i ; (d) Detailed illustration shows there are 3 landmark candidates in \mathcal{B}_i .

We used the edge detection method to extract the landmark candidates $\mathcal{B}_i, i = 1, 2, \dots, n$. Noting that the intensity of pixels just outside the scalp contour is smaller

than that of those just inside the contour, we extracted \mathcal{B}_i by searching a rising-edge (from outside to inside) along \mathbf{n}_i near \mathbf{v}_i . An example is illustrated in Figs.3(c) and (d) where 3 landmark candidates are detected corresponding to one \mathbf{v}_i .

Our first experiment is to segment an image (shown in Fig.3(b)) 10 slices away from the template. The shape deformation process is shown in Fig.4. At the beginning (left column of Fig.4), some \mathbf{v}_i 's are mis-selected. However, the regularized shape can match the desired contour much better although some error still exists in some local areas. After updating \mathbf{V}_i' corresponding to the support vectors, we can see that many outliers are removed.

The second experiment is to verify the effectiveness of the proposed algorithm when part of the input image is occluded. Fig.5(a) is a simulated “occluded” image with the template shape overlapped. Using the proposed method together with the additional constraints discussed in Section 3.2, quite accurate segmentation was acquired, as shown in Fig.5(b). In this experiment, only 25 out of 100 landmarks are allowed to slide freely along their normal directions.

Finally, we used the proposed deformation method to segment the cryosection brain images slice by slice. Beginning from the manual segmented template shown in Fig.3(a), the shape in the template is deformed to match its neighbors. Then the segmented slices are treated as new templates to process their unsegmented neighbors. The first and last images in Fig.6 are 31 slices away from the initial template image. Only the slices with odd numbers are shown here.

5 Conclusion

A novel landmark-based shape deformation method has been described in this paper. This method provides effective solution to two problems inherent in landmark-based shape deformation: (a) identification of landmark points from a given input image, and (b) regularized deformation of object shape embedded in a template. The first problem is solved using a combination of edge detection and shape fitting. The second problem is solved using a new constrained SVM regression technique, in which a thin-plate kernel is utilized to provide non-rigid shape deformations. This method offers several advantages over existing landmark-based methods. First, it has a unique capability to detect and use multiple candidate landmark points in an input image to improve landmark detection. Second, it can handle the case of missing landmarks, which often arises in dealing with occluded images.

The proposed method is especially suitable for segmenting 3D images slice by slice, where there are only small shape variations across the neighboring slices. In this type of applications, the segmented shape in one slice can be used as a template for its unsegmented neighborings. We have applied the proposed method to extract the scalp con-

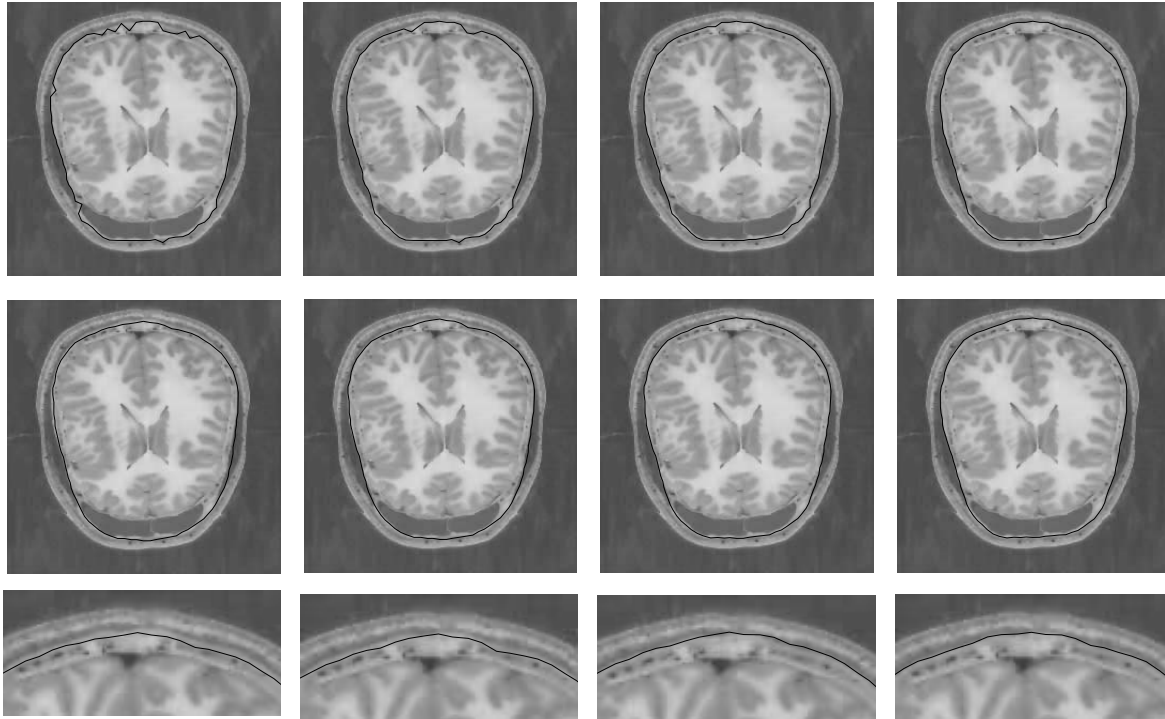


Figure 4. Segmentation to the input image shown in (Fig.3). The top row is the updated V' (connected as a polygon). The middle row shows the acquired $t(V)$. The bottom row is the local enlargement of the middle row. From the left to right column shows the intermediate results with ϵ of 3, 2, 1 and 0, respectively.

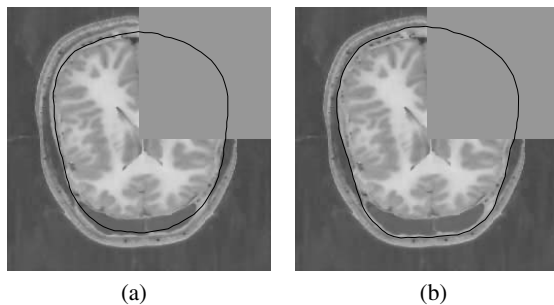


Figure 5. Segmentation to a simulated occluded image.

tours in cryosection head images with very encouraging results.

Acknowledgements

We would like to thank Prof. Tamer Basar for invaluable suggestions on solving the optimization algorithm. This work was supported, in part, by NSF-BES-95-02121 (Z.P.L) and a Beckman Graduate Fellowship 2000-2001

(S.W.).

References

- [1] A. A. Amini, S. Tehrani, and T. E. Weymouth. Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints. In *Proc. of ICCV*, pages 95–99, 1988.
- [2] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. PAMI*, 11:567–585, June 1989.
- [3] V. Casselles, F. Catte, T. Coll, and F. Dibos. A geometric model for active contours. *Numerische Mathematik*, 66:1–31, Jan. 1993.
- [4] L. D. Cohen and I. Cohen. A finite element method applied to new active contour models and 3d reconstruction from cross sections. In *Proc. of ICCV*, pages 587–591, 1990.
- [5] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models - their training and application. *Computer Vision and Image Understanding*, 61:38–59, Jan. 1995.
- [6] P. Huber. *Robust Statistics*. New York: Wiley, 1981.
- [7] A. K. Jain, Y. Zhong, and S. Lakshmanan. Object matching using deformable templates. *IEEE Trans. PAMI*, 18:267–278, March 1996.

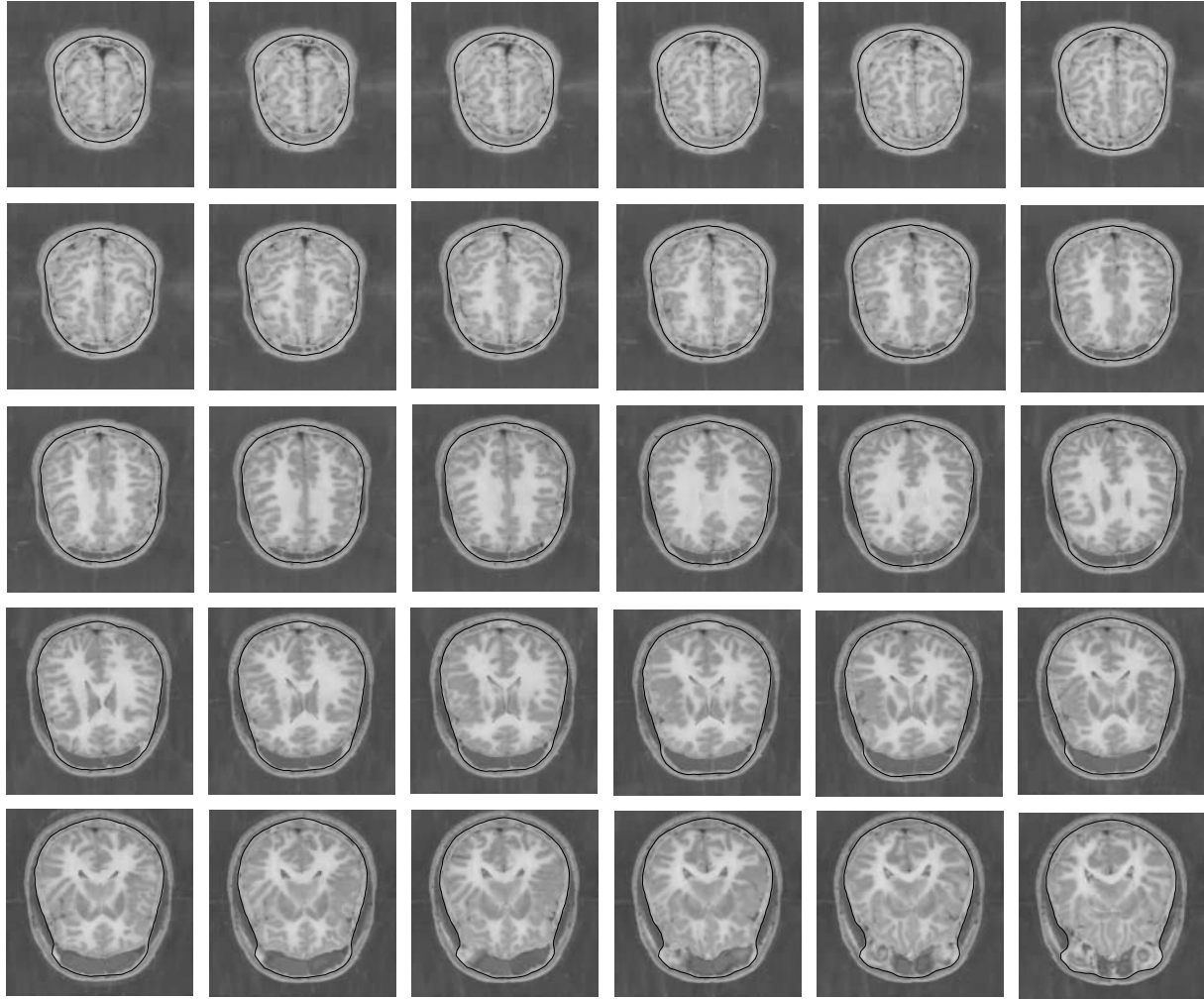


Figure 6. Segmentation of the 3D brain image slice by slice. The top-left and bottom-right slices are 31 slices away from the template.

- [8] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. Journal of Computer Vision*, 1(4):321–331, 1987.
- [9] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.
- [10] M. E. Leventon, E. L. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. In *Proc. of CVPR*, pages 316–323, 2000.
- [11] P. Perona, T. Shiota, and J. Malik. Anisotropic diffusion. In *Geometry-Driven Diffusion in Computer Vision*, pages 73–92. Dordrecht: Kluwer, 1994.
- [12] D. Rueckert and P. Burger. Geometrically deformable templates for shape-based segmentation and tracking in cardiac mr image. In *Energy Minimization Methods in Comp. Vision and Pattern Recog.: Int. Workshop*, pages 83–98, 1997.
- [13] L. H. Staib and J. S. Duncan. Boundary finding with parametrically deformable models. *IEEE Trans. PAMI*, 17:1061–1075, Nov. 1992.
- [14] A. W. Toga. *Brain Warping*. Academic Press, 1999.
- [15] V. N. Vapnik. *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 2000.
- [16] G. Wahba. *Spline Models for Observational Data*. Society Industrial and Applied Mathematics, 1990.
- [17] D. J. Williams and M. Shah. A fast algorithm for active contours. In *Proc. of ICCV*, pages 592–595, 1990.
- [18] Y. Zhong, A. K. Jain, and M. P. Dubuisson-Jolly. Object tracking using deformable template. *IEEE Trans. PAMI*, 22:544–549, May 2000.
- [19] S. C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and bayes/mdl for multi-band image segmentation. *IEEE Trans. PAMI*, 18(9):884–900, 1996.