Available online at www.sciencedirect.com

SCIENCE DIRECT®

Medical Image Analysis 8 (2004) 429–445

MEDICAL IMAGE ANALYSIS

www.elsevier.com/locate/media

Mutual information in coupled multi-shape model for medical image segmentation

A. Tsai [a,d,*], W. Wells [b,c], C. Tempany [c], E. Grimson [b], A. Willsky [a]

[a] *Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology (MIT), Room #35-427, Cambridge, MA 02139, USA*

[b] *Artificial Intelligence Laboratory, MIT, Cambridge, MA, USA*

[c] *Department of Radiology, Brigham and Women's Hospital, Boston, MA, USA*

[d] *Department of Medicine, Massachusetts General Hospital, Boston, MA, USA*

Received 28 May 2003; received in revised form 12 January 2004; accepted 20 January 2004
Available online 27 April 2004

Abstract

This paper presents extensions which improve the performance of the shape-based deformable active contour model presented earlier in [IEEE Conf. Comput. Vision Pattern Recog. 1 (2001) 463] for medical image segmentation. In contrast to that previous work, the segmentation framework that we present in this paper allows multiple shapes to be segmented simultaneously in a seamless fashion. To achieve this, multiple signed distance functions are employed as the implicit representations of the multiple shape classes within the image. A parametric model for this new representation is derived by applying principal component analysis to the collection of these multiple signed distance functions. By deriving a parametric model in this manner, we obtain a coupling between the multiple shapes within the image and hence effectively capture the co-variations among the different shapes. The parameters of the multi-shape model are then calculated to minimize a single mutual information-based cost criterion for image segmentation. The use of a single cost criterion further enhances the coupling between the multiple shapes as the deformation of any given shape depends, at all times, upon every other shape, regardless of their proximity. We found that this resulting algorithm is able to effectively utilize the co-dependencies among the different shapes to aid in the segmentation process. It is able to capture a wide range of shape variability despite being a parametric shape-model. And finally, the algorithm is robust to large amounts of additive noise. We demonstrate the utility of this segmentation framework by applying it to a medical application: the segmentation of the prostate gland, the rectum, and the internal obturator muscles for MR-guided prostate brachytherapy.

Keywords: Shape prior; Segmentation; Mutual-information; Level set methods

1. Introduction

Medical image segmentation – the process of outlining relevant anatomical structures in an image dataset – is a problem that is central to a variety of medical applications including image enhancement and reconstruction, surgical planning, disease classification, data storage and compression, and 3D visualization. Medical images, however, are particularly difficult to segment due to a number of reasons. These reasons include, but are not limited to, the following:

- Soft tissue boundaries are frequently not well defined as they often blend in with the surrounding tissues.
- Even with the advent of improved technology, image acquisition systems often still yield low signal-to-noise ratio (SNR) data.
- Tissue variability across the patient population makes designing a segmentation algorithm difficult.
- Image artifacts, either due to patient motion or limited acquisition time, reduce the information content of the data.

In this paper, we propose a knowledge-based approach to medical image segmentation. The strength of such an approach is the incorporation of prior information into the segmentation algorithm to reduce the

[*] Corresponding author.
E-mail address: atsai@mit.edu (A. Tsai).

1361-8415/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.media.2004.01.003

complexity of the segmentation process. To motivate this approach, we show, in Fig. 1, an axial brain MR image depicting three subcortical brain structures. The dark ventricle is easy to distinguish from the rest of the brain structures. The boundaries of the other two subcortical brain structures, however, are more difficult to localize. Despite this apparent difficulty, the human vision system does not have trouble locating all three subcortical structures. First, based on prior knowledge of the spatial relationship of the structures, our vision system uses the easily identifiable ventricle as a spatial reference point to localize the other two subcortical brain structures. Next, based on prior knowledge of the variability of the individual shapes and their mutual shape variability, our vision system proceeds to identify the boundaries of the lenticular and the caudate nuclei. This two level usage of prior information, first to localize spatially and then to extract shape, is a powerful concept, and one that our vision system exploits. In this paper, we show the development of a segmentation algorithm that is sophisticated enough to mimic this particular characteristic of the human vision system.

Our work is related to many shape-based active contour models. Cootes et al. (1995) used linear combinations of eigenvectors that reflect shape variations to parametrize the segmenting curve. Following that, Frangi et al. (2002) reported a method to automatically generate landmark points for these models. Staib and Duncan (1992) used elliptic Fourier decomposition of various landmark points to parametrize their segmenting curve. Wang and Staib (1998) proposed a shape parametrization scheme based on applying principal component analysis to covariance matrices that capture the variations of the shapes' control points. Leventon et al. (2000) derived a parametric shape model based on applying principal component analysis to a collection of signed distance functions to restrict the flow of the geodesic active contour. Paragios and Rousson (2002) used a prior level set shape model to restrict the flow of an evolving level set. Constraints are imposed to force the evolving level set to remain as a distance function. Our work also shares many common aspects with a number of coupled active contour models. Zeng et al. (1999) introduced a coupled-surfaces propagation method where each evolving surface is driven by two forces: (1) an image-derived information force, and (2) a coupling force to maintain the two evolving surfaces a certain distance apart. Chan and Vese (2000) employed n level set functions to represent 2^n segments in the image with the n level set functions coupled to one another through an energy functional. Yezzi et al. (1999, 2002) derived a set of coupled curve evolution equations from a single global cost functional to evolve multiple contours simultaneously toward the region boundaries.

Mutual-information (MI), or relative entropy, is a basic concept which underlies much of information theory (Cover and Thomas, 1991). It plays an integral part in our shape-based model. Over the past few years, MI has been used to solve a variety of problems ranging from image registration to pose alignment to global and local spatial correspondence (Bello and Colchester, 1998; Viola and Wells, 1995; Wells et al., 1996). Recently, MI has also been introduced as a measure for image segmentation (Kim et al., 2002; Unal et al., 2002). The MI segmentation framework only assumes statistical dependence between pixel intensities within particular regions of the image, and does not require the regions to have a particular type of probability distribution. As a result, the MI segmentation framework can handle a wider range of segmentation problems when

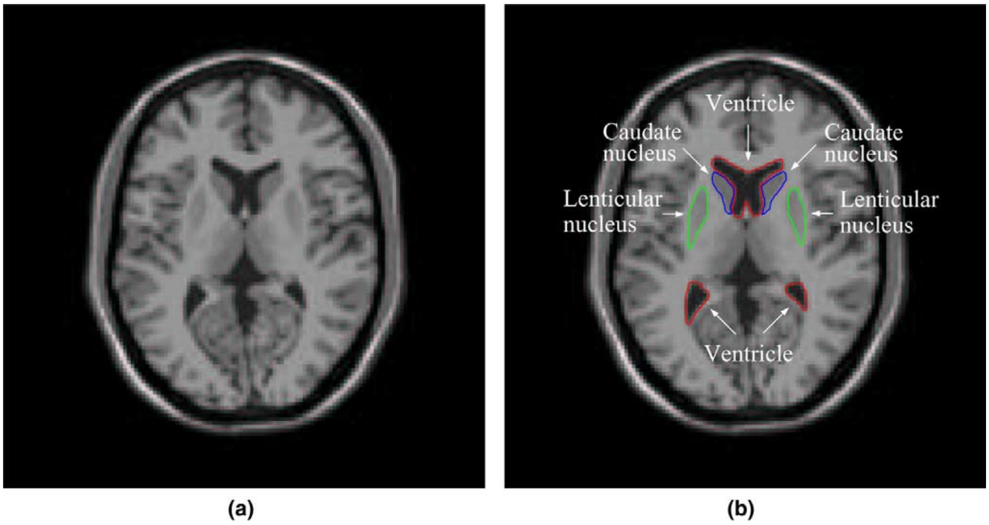


Fig. 1. Motivational example. (a) MR image showing an axial cross-section of the brain. (b) Hand segmentation of the three subcortical brain structures within the image.

compared to edge-based or simple statistical model segmenters (e.g. those based on mean and/or variance).

The rest of the paper is organized as follows. Section 2 describes a variational approach to align all the example training shapes. Section 3 describes a new approach to represent multiple shapes. Section 4 illustrates how our multi-shape model can be incorporated into a mutual information-based active contour model for image segmentation. In Section 5, we show the application of our technique to a medical application. We conclude in Section 6 with a summary of the paper.

2. Simultaneous multi-shape alignment

Alignment is an essential pre-processing step prior to any attempts to capture the shape variations in a training collection. It functions to remove shape differences within a particular shape class that might be due to differences in pose. Since multiple shape classes are involved in our framework, we seek an alignment procedure that is able to (1) jointly align the different shapes within a particular shape class, and (2) perform the alignment for all the shape classes simultaneously. This section describes such an alignment procedure. The approach is a generalization of the method presented in (Tsai et al., 2001) in which a variational approach is employed to jointly align a training set consisting of a single shape class.

In our generalization, let m represent the number of known shape classes in an image that we would like to model, and let the training set $\mathscr{T}$ consist of n such images. One way to represent the multiple shapes in an image is to label each shape by a different intensity value. Though this representation may be effective for visualization, it is not the most effective representation for image manipulation. [1] A more effective and natural way to represent the m shapes within an image is to encode the shapes in a m-dimensional binary vector-valued image. Specifically, let the training set $\mathscr{T}$ consist of n vector-valued images $\{\mathbf{I}_1, \mathbf{I}_2, \ldots, \mathbf{I}_n\}$, where $\mathbf{I}_i = (I_i^1, \ldots, I_i^m)$ for $i = 1, \ldots, n$. Each I_i^k for $i = 1, \ldots, n$ and $k = 1, \ldots, m$ is a binary image with values of one inside and zero outside the shape. This resulting multi-shape encoding scheme is not only flexible as it allows any number of shapes to be represented compactly, but it also allows each shape class to have an equal footing during any image alignment procedure.

The basic idea behind our approach is to calculate the set of pose parameters $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n$ used to transform the n binary vector-valued images to jointly align them. We focus on using rigid body and scaling transformations to align these binary vector-valued images to each other. [2] In 2D, the pose parameter $\mathbf{p} = [a \;\; b \;\; h \;\; \theta]^{\mathrm{T}}$ with a, b, h and θ corresponding to x-, y-translation, scale and rotation, respectively. The transformed image of I_i^k for $i = 1, \ldots, n$ and $k = 1, \ldots, m$, based on the pose parameter $\mathbf{p}_i$, is denoted by $\tilde{I}_i^k$, and is defined as

$$\tilde{I}_i^k(\tilde{x}, \tilde{y}) = I_i^k(x, y),$$

where

$$\begin{aligned}
\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ 1 \end{bmatrix} &= T[\mathbf{p}_i] \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \\
&= \underbrace{\begin{bmatrix} 1 & 0 & a_i \\ 0 & 1 & b_i \\ 0 & 0 & 1 \end{bmatrix}}_{M(a_i, b_i)} \underbrace{\begin{bmatrix} h_i & 0 & 0 \\ 0 & h_i & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{H(h_i)} \\
&\quad \times \underbrace{\begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{R(\theta_i)} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}.
\end{aligned} \tag{1}$$

The transformation matrix $T[\mathbf{p}_i]$ is the product of three matrices: a translation matrix $M(a_i, b_i)$, a scaling matrix $H(h_i)$, and an in-plane rotation matrix $R(\theta_i)$. This transformation matrix $T[\mathbf{p}_i]$ maps the coordinates $(x, y) \in \mathbb{R}^2$ into coordinates $(\tilde{x}, \tilde{y}) \in \mathbb{R}^2$.

In (Tsai et al., 2001), an alignment cost functional is constructed based the difference in the area divided by the total area of any pair of binary images in a given training collection. We generalize this alignment strategy to jointly align a set of n m-dimensional binary images. Specifically, we employ gradient descent to minimize the following energy functional:

$$E_{\text{align}} = \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \left\{ \sum_{k=1}^{m} \frac{\int\int_{\Omega} (\tilde{I}_i^k - \tilde{I}_j^k)^2 \, \mathrm{d}A}{\int\int_{\Omega} (\tilde{I}_i^k + \tilde{I}_j^k)^2 \, \mathrm{d}A} \right\}, \tag{2}$$

where Ω denotes the image domain. Minimizing (2) is equivalent to minimizing the difference between any pair of binary vector-valued images in the training collection. The area normalization term in the denominator of (2) is again employed here to prevent all the binary vector-valued images from shrinking to improve the cost function.

The gradient of E_{align}, taken with respect to $\mathbf{p}_i$, for any i, is given by

[1] By assigning different intensity values to different shape classes, a numerical bias is created among the different shape classes. For example, in intensity-based grayscale image alignment algorithms, by assigning an intensity value of 10 to shape class A and an intensity value of 1 to shape class B, the algorithm incorrectly places an emphasis on the importance of aligning shape class A relative to shape class B.

[2] Affine transformation can also be used instead without any loss of generality in the formulation.

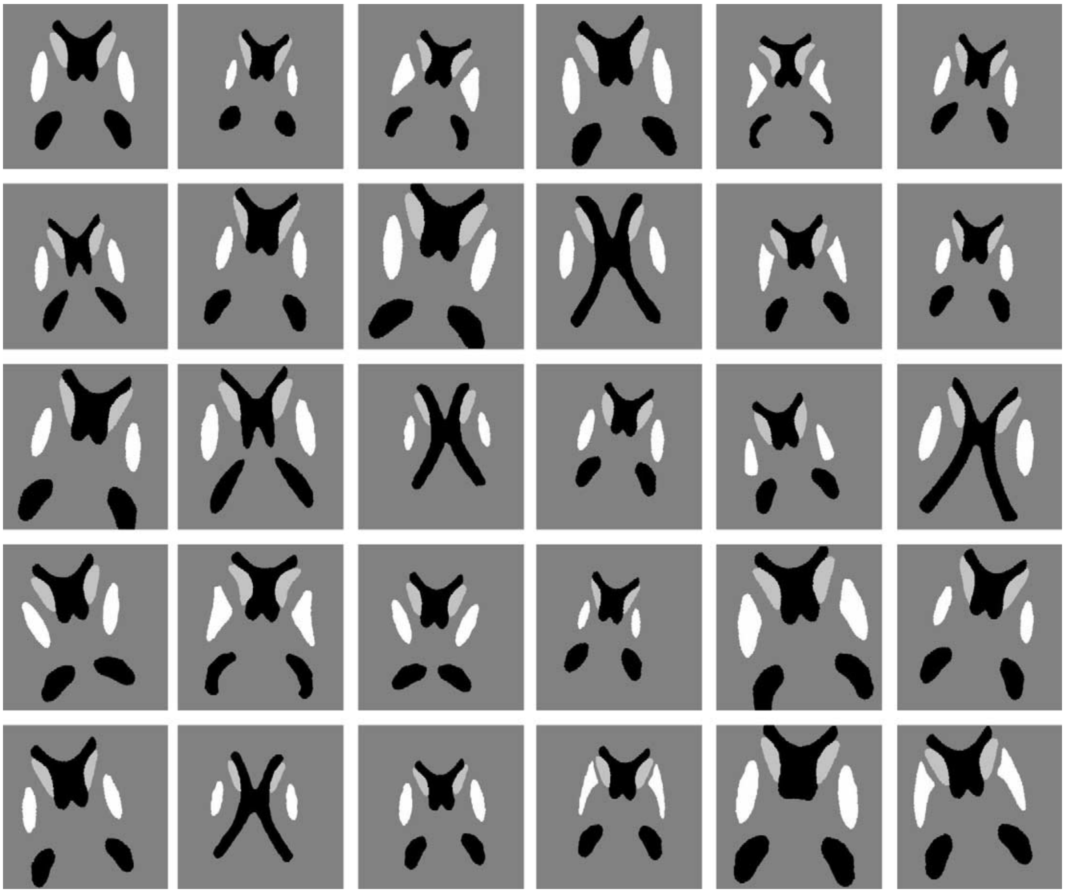


Fig. 2. Collection of subcortical brain structures before alignment. The three brain structures depicted in different colors are: ventricles (black), lenticular nucleus (white), and caudate nucleus (gray).

$$\nabla_{\mathbf{p}_i} E_{\text{align}} = \sum_{\substack{j=1\\ j\neq i}}^{n} \sum_{k=1}^{m} \left\{ \frac{2\int\int_\Omega (\tilde{I}_i^k - \tilde{I}_j^k)\nabla_{\mathbf{p}_i}\tilde{I}_i^k \, \mathrm{d}A}{\int\int_\Omega (\tilde{I}_i^k + \tilde{I}_j^k)^2 \, \mathrm{d}A} - \frac{2\int\int_\Omega (\tilde{I}_i^k - \tilde{I}_j^k)^2 \, \mathrm{d}A \int\int_\Omega (\tilde{I}_i^k + \tilde{I}_j^k)\nabla_{\mathbf{p}_i}\tilde{I}_i^k \, \mathrm{d}A}{\left(\int\int_\Omega (\tilde{I}_i^k + \tilde{I}_j^k)^2 \, \mathrm{d}A\right)^2} \right\}, \qquad (3)$$

where $\nabla_{\mathbf{p}_i}\tilde{I}_i^k$ is the gradient of the transformed image $\tilde{I}_i^k$ taken with respect to the pose parameter $\mathbf{p}_i$. Using the chain rule, the lth component of $\nabla_{\mathbf{p}_i}\tilde{I}_i^k$ is given by

$$\nabla_{\mathbf{p}_i^l}\tilde{I}_i^k(\tilde{x},\tilde{y}) = \begin{bmatrix} \frac{\partial \tilde{I}_i^k(\tilde{x},\tilde{y})}{\partial \tilde{x}} & \frac{\partial \tilde{I}_i^k(\tilde{x},\tilde{y})}{\partial \tilde{y}} & 0 \end{bmatrix} \frac{\partial T[\mathbf{p}_i]}{\partial \mathbf{p}_i^l} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix},$$

where

$$\frac{\partial T[\mathbf{p}_i]}{\partial \mathbf{p}_i^1} = \frac{\partial T[\mathbf{p}_i]}{\partial a_i} = \frac{\partial M(a_i,b_i)}{\partial a_i} H(h_i) R(\theta_i), \qquad (4a)$$

$$\frac{\partial T[\mathbf{p}_i]}{\partial \mathbf{p}_i^2} = \frac{\partial T[\mathbf{p}_i]}{\partial b_i} = \frac{\partial M(a_i,b_i)}{\partial b_i} H(h_i) R(\theta_i), \qquad (4b)$$

$$\frac{\partial T[\mathbf{p}_i]}{\partial \mathbf{p}_i^3} = \frac{\partial T[\mathbf{p}_i]}{\partial h_i} = M(a_i,b_i) \frac{\partial H(h_i)}{\partial h_i} R(\theta_i), \qquad (4c)$$

$$\frac{\partial T[\mathbf{p}_i]}{\partial \mathbf{p}_i^4} = \frac{\partial T[\mathbf{p}_i]}{\partial \theta_i} = M(a_i,b_i) H(h_i) \frac{\partial R(\theta_i)}{\partial \theta_i}. \qquad (4d)$$

The matrix derivatives in Eq. (4a) are taken componentwise. The update equation for the pose parameter $\mathbf{p}_i$ is then given, in terms of $\nabla_{\mathbf{p}_i} E_{\text{align}}$, as

$$\mathbf{p}_i^{t+1} = \mathbf{p}_i^t - \Delta t_{\mathbf{p}} \nabla_{\mathbf{p}_i} E_{\text{align}},$$

where t denotes the iteration number and $\Delta t_{\mathbf{p}}$ denotes the step size in updating $\mathbf{p}_i$. Since the solution of this alignment problem is under-determined, we regularize the problem by keeping the pose of an arbitrarily chosen training example fixed and calculating the pose parameters for the remaining training examples using the above approach. The initial poses of the training shapes in $\mathscr{T}$ are determined by the user. The gradient descent method of above is then performed until convergence. This method jointly aligns the n example shapes in a shape class, and performs this alignment task on all the m shape classes simultaneously.

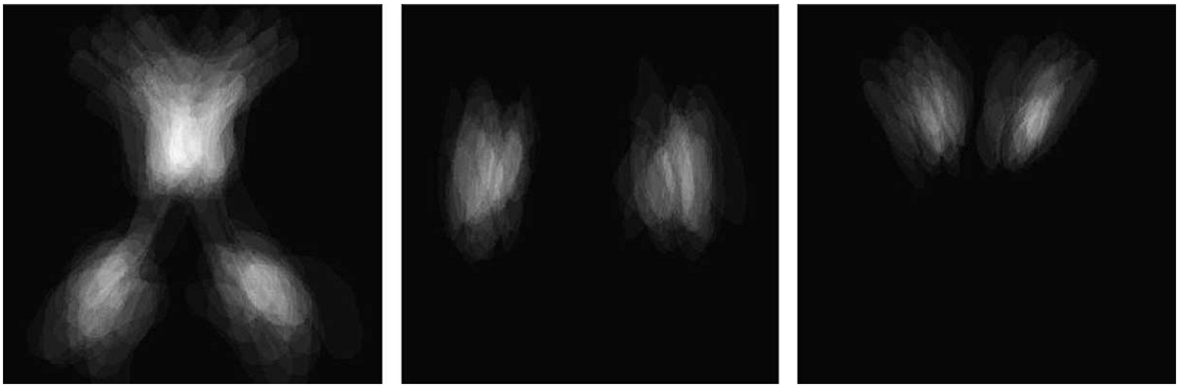


Fig. 3. A visual assessment of the collection's initial alignment. From left to right, this figure shows the summation images of the ventricle, the lenticular nucleus, and the caudate nucleus before alignment. The blurriness in the summation images are due to the misalignment of the example shapes.

We use a 2D toy example to illustrate our multi-shape alignment technique. [3] We employ a training set consisting of 30 synthetic examples. Each of these examples depicts a different axial cross-section of three subcortical brain structures. The subcortical brain structures within each of the training images, shown in Fig. 2, consist of the ventricle (shown in black), the lenticular nucleus (shown in white), and the caudate nucleus (shown in gray). These three structures form the three shape classes of our alignment algorithm. Notice that in the majority of these examples, the ventricle appears as three separate black regions while in four of these examples, the ventricle appears as one single black "X"-shaped region. The appearance of the topologically changing ventricle depends largely on the level at which the 2D cross-section of the brain was performed. In this alignment example, the pose parameter associated with the image in the upper left corner of the figure is chosen to be fixed, i.e. $\mathbf{p}_1 = [0\ 0\ 1\ 0]^{\mathrm{T}}$. One way to visually judge the alignment of the example shapes within each shape class is to form a "summation" image for that shape class. The summation image of a shape class is formed by summing together, in a pixelwise fashion, all the binary representations of the example shapes belonging to that shape class. The intensity range of the summation image ranges between 0 (when a particular pixel represents the background pixel in all the example shapes) and n, the number of example shapes (when a particular pixel represents a pixel located within all the example shapes). The summation images of the three subcortical brain structures before alignment, based on summing together binary representations of the examples shapes before alignment, are shown in Fig. 3. The blurriness in the summation images are due to the misalignment of the example shapes.

After employing our multi-shape alignment scheme outlined in this section, we show in Fig. 4 the post-aligned summation images of the three subcortical brain structures, based on summing together post-aligned binary representations of the examples shapes. The coherence of the aligned shapes is indicated by the increased sharpness of the summation images. This is one way to visually judge the effectiveness of our multi-shape alignment strategy. Fig. 5 shows the post-aligned training collection consisting of 30 training examples of subcortical brain structures. The various subcortical brain structures, when compared against one another within the same shape class, share roughly the same center, are all pointing roughly in the same direction, and are approximately equal in size.

3. Implicit multi-level set parametric shape model

Shape representation is an important element in any shape-based segmentation algorithm, including ours. This section describes an approach to represent multiple shape classes in a single framework. Because our approach is an extension of the implicit parametric single shape representation presented in (Leventon et al., 2000) and (Tsai et al., 2001), it enjoys many of the same advantages, namely, computationally efficiency, accuracy in representing shapes (even in traditionally problematic areas such as high curvature points), numerical stability, effectiveness in capturing a wide range of shape variabilities, and capability to handle topological changes. In this section, we detail how we advanced this approach to (1) allow the simultaneous representation of any number of complicated shape classes, (2) display a wide range of individual shape variability, (3) capture co-variations shared among different shape classes, and (4) handle images with

[3] This example is artificial and is simply employed for demonstration purposes only. Being structures that vary in 3-space, optimal alignment of the brain structures should be performed in 3D as well.

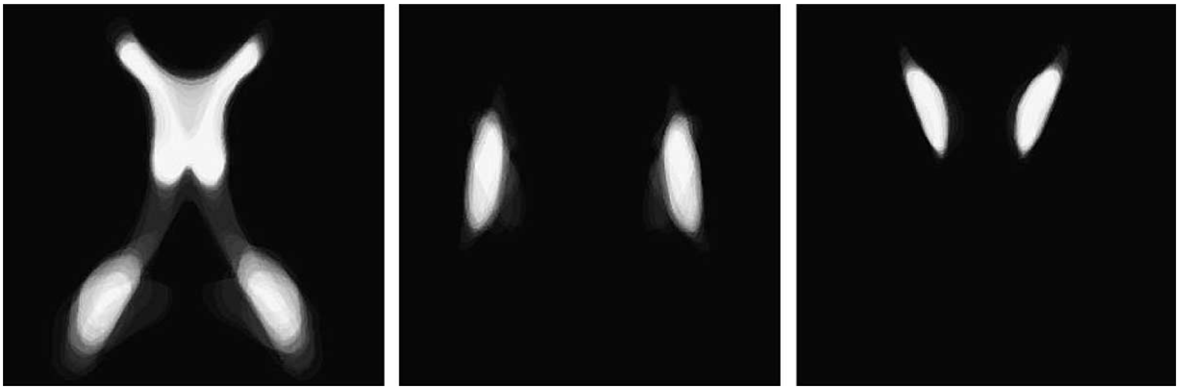


Fig. 4. A visual assessment of the collection's final alignment. From left to right, this figure shows the post-aligned summation images of the ventricle, the lenticular nucleus, and the caudate nucleus. The coherence of the aligned example shapes is reflected by the increased sharpness of the summation images.

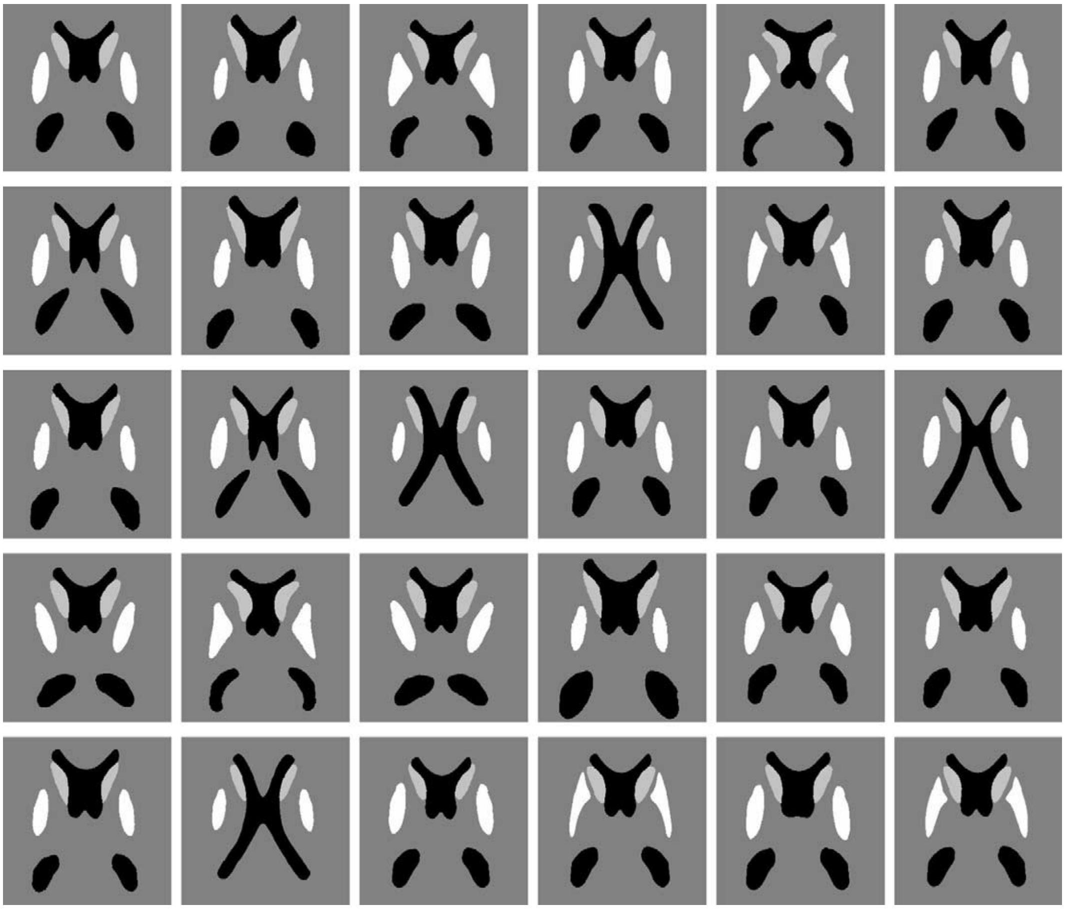


Fig. 5. Collection of post-aligned subcortical brain structures. The three brain structures depicted in different colors are: ventricles (black), lenticular nucleus (white), and caudate nucleus (gray).

multiple junctions (i.e. points at which greater than two regions intersect).

3.1. Construction of shape parameters

Following the lead of Leventon et al. (2000), Tsai et al. (2001) and Frangi et al. (2002), the signed distance function Ψ is employed as the representation of a particular shape. [4] Let m be the number of shape classes

[4] The signed distance $\Psi(p)$ from an arbitrary point p to a known surface $\mathscr{L}$ is the distance between p and the closest point z in $\mathscr{L}$, multiplied by 1 or −1, depending on which side of the surface p lies in (Borgefors, 1986).

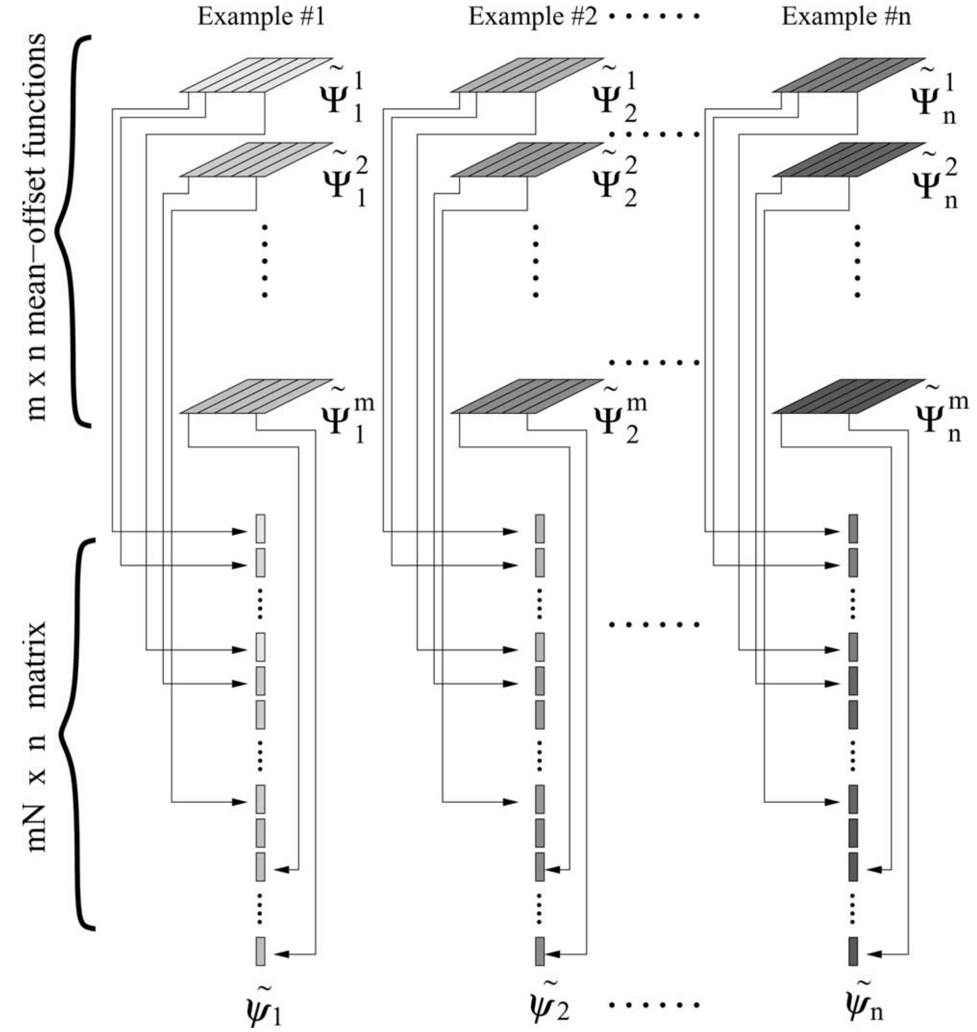


Fig. 6. Diagram illustrating the formation of the column vectors $\{\tilde{\psi}_1, \dots, \tilde{\psi}_n\}$.

that we would like to represent simultaneously. For example, $m = 3$ if we would like to represent the three subcortical brain structures in Fig. 1. The boundaries of each of the m shapes are embedded as the zero level sets of m distinct signed distance functions $\{\Psi^1, \dots, \Psi^m\}$ with negative distances assigned to the inside and positive distances assigned to the outside of the shape. Osher and Sethian (1988) made popular this level set representation for single object boundaries.

Suppose there are n aligned examples of these images, each with m shapes, in the collection. [5] For each of the n example images, m signed distance functions are generated, giving rise to nm signed distance functions. Let Ψ_i^k denote the signed distance function associated with the kth shape class in the ith example image of the training collection. Using the technique developed earlier in (Leventon et al., 2000) and (Tsai et al., 2001), we compute m mean level set functions $\{\bar{\Phi}^1, \dots, \bar{\Phi}^m\}$, one for each shape class, by averaging n signed distance functions from that shape class. Mathematically, $\bar{\Phi}^k = \frac{1}{n}\sum_{i=1}^{n} \Psi_i^k$ for any $k = 1, \dots, m$. To extract the shape variabilities of each shape class, the mean level set function from each shape class is subtracted from each of the n signed distance functions belonging to that shape class. This gives rise to the following nm mean-offset functions: $\tilde{\Psi}_i^k$ for $i = 1, \dots, n$ and $k = 1, \dots, m$. These mean-offset functions are then used to capture the variabilities within and across the m shape classes.

To capture the shape variabilities, we form n column vectors $\{\tilde{\psi}_1, \dots, \tilde{\psi}_n\}$. Each column vector $\widetilde{\psi_i}$, of size (mN), is made up of m mean-offset functions $\{\tilde{\Psi}_i^1, \dots, \tilde{\Psi}_i^m\}$ stacked on top of one another with each mean-offset function consisting of N samples (using identical sample locations for each mean-offset function). The most natural sampling strategy for the mean-offset functions is to utilize the $N_1 \times N_2$ rectangular grid to generate $N = N_1 N_2$ lexicographically ordered samples

[5] The n examples in the collection are aligned by employing the method presented in Section 2.

(where the columns of the rectangular grid are sequentially stacked on top of one another to form one large column). Fig. 6 illustrates how the column vectors $\{\tilde{\psi}_1, \ldots, \tilde{\psi}_n\}$ are formed from the nm mean-offset functions $\tilde{\Psi}_i^k$ for $i = 1, \ldots, n$ and $k = 1, \ldots, m$.

Define a tall rectangular shape-variability matrix $\mathscr{S}$ as

$$\mathscr{S} = [\tilde{\psi}_1 \quad \tilde{\psi}_2 \quad \cdots \quad \tilde{\psi}_n].$$

An eigenvalue decomposition is employed to factor $\frac{1}{n}\mathscr{S}\mathscr{S}^{\mathrm{T}}$ as

$$\frac{1}{n}\mathscr{S}\mathscr{S}^{\mathrm{T}} = U\Sigma U^{\mathrm{T}}, \tag{5}$$

where U is a tall rectangular $mN \times n$ matrix whose columns represent the n principal variational modes or eigenshapes of the m shape classes, and Σ is an $n \times n$ diagonal matrix whose diagonal elements, denoted by $\sigma_1^2, \ldots, \sigma_n^2$, represent the corresponding non-zero eigenvalues. Each non-zero eigenvalue reflects the variance of shape variability associated with that eigenvalue's corresponding eigenshape. The mN elements of the ith column of U, denoted by U_i, are arranged back into m rectangular structures of dimension $N_1 \times N_2$ (by undoing the earlier stacking and lexicographical concatenation of the grid columns). This "unwrapping" process yields the ith principal modes or eigenshapes for all the m shape classes denoted by $\{\Phi_i^1, \ldots, \Phi_i^m\}$. In the end, this approach generates a maximum of n different eigenshapes $\{\Phi_1^k, \Phi_2^k, \ldots, \Phi_n^k\}$ for shape classes $k = 1, \ldots, m$.

Very importantly, because the eigenshapes are derived by performing principal component analysis on a shape variability matrix $\mathscr{S}$ which contains information about all the m shape classes, the eigenshapes derived for each shape class will naturally have strong couplings between the different shape classes. That is, any co-dependencies between the m shape classes will be captured by the various eigenshapes.

In most cases, the dimension of the matrix $\frac{1}{n}\mathscr{S}\mathscr{S}^{\mathrm{T}}$ is large, of size $mN \times mN$, so the calculation of the eigenvectors and eigenvalues of this matrix is computationally expensive. In practice, the eigenvectors and eigenvalues of $\frac{1}{n}\mathscr{S}\mathscr{S}^{\mathrm{T}}$ can be efficiently computed from a much smaller $n \times n$ matrix $\mathscr{W}$ given by

$$\mathscr{W} = \frac{1}{n}\mathscr{S}^{\mathrm{T}}\mathscr{S}.$$

It is straightforward to show that if $\mathbf{d}$ is an eigenvector of $\mathscr{W}$ with corresponding eigenvalue λ, then $\mathscr{S}\mathbf{d}$ is an eigenvector of $\frac{1}{n}\mathscr{S}\mathscr{S}^{\mathrm{T}}$ with eigenvalue λ (see (Turk and Pentland, 1991) for a proof).

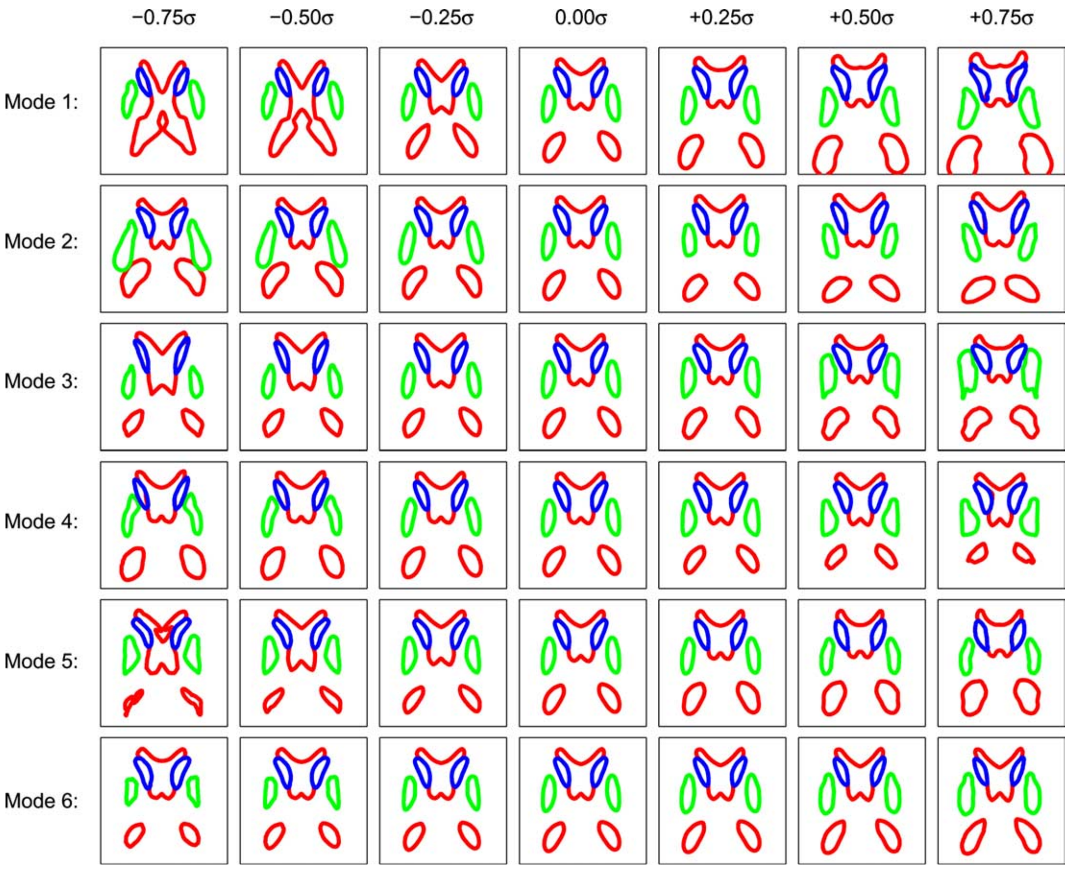


Fig. 7. Illustration of the shape variabilities in the subcortical brain structures based on our implicit multi-shape parametric model.

Suppose that we choose to use $q \leqslant n$ modes in our shape representation. [6] We now introduce m new level set functions:

$$\Phi^1[\mathbf{w}] = \bar{\Phi}^1 + \sum_{i=1}^{q} w_i \Phi_i^1, \tag{6a}$$

$$\Phi^2[\mathbf{w}] = \bar{\Phi}^2 + \sum_{i=1}^{q} w_i \Phi_i^2, \tag{6b}$$

$$\vdots$$

$$\Phi^m[\mathbf{w}] = \bar{\Phi}^m + \sum_{i=1}^{q} w_i \Phi_i^m, \tag{6c}$$

where $\mathbf{w} = \{w_1, w_2, \ldots, w_q\}$ are the weights for the q eigenshapes in each of the m new level set functions. The variances of these weights $\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_q^2\}$ are given by the eigenvalues calculated earlier. We propose to use these newly constructed level set functions $\{\Phi^1, \ldots, \Phi^m\}$ as our implicit representation of the m shape classes. Specifically, the zero level set of Φ^k describes the boundaries of the kth shape class with that shape's variability directly linked to the variability of its level set function. Note that the shape variability we allow in this representation is restricted to the variability given by the q eigenshapes.

Because of this construction, the m new level set functions $\{\Phi^1, \ldots, \Phi^m\}$ are all linked to one another through the weights $\mathbf{w}$. This provides another coupling mechanism between the different shape classes. By varying $\mathbf{w}$, all the Φ^k for $k = 1, \ldots, m$ vary together.

To illustrate the parametric multi-shape representation described in Eq. (6a), we show, in Fig. 7, the shape variations of the subcortical brain structures based on varying the model's first six eigenshapes by different amounts. Each row of the figure demonstrates the effect of a particular principal mode in altering the shapes of the subcortical brain structures. Notice that by varying the first principal mode, the shape of the ventricle changes topology going from three regions to one. This is an additional advantage of using the Eulerian framework for shape representation as it can handle topological changes in a seamless fashion. Further, because multiple level sets are employed to represent multiple curves in this framework, multiple junctions can be captured automatically. Specifically, triple points (i.e. points at which three regions intersection) formed by the ventricle and the caudate nucleus can be seen throughout Fig. 7. In this particular case, parts of two zeroth level sets about against one another resulting in the formation of a triple point.

It is important to realize that because there are no restrictions placed on the range of values that $\mathbf{w}$ can take, it is possible that the different shape classes may overlap one another, especially with extreme values of $\mathbf{w}$. In fact, this phenomenon can be seen starting to develop in a number of image frames shown in Fig. 7. For example, the caudate nucleus and the lenticular nucleus, shown in the image frame located in row three column seven, overlap. With further increase in the additive weight of the third eigenshape, overlap of the these two shapes is inevitable. We show in Section 4 the method by which we avoid overlapping of the different shapes during the segmentation process.

3.2. Addition of pose parameters

At this point, our implicit representation of multiple shape classes described in Eq. (6a) cannot accommodate shape variabilities that are due to pose differences. To have this flexibility, pose parameter $\mathbf{p}$ is added as another parameter to the level set functions of Eq. (6a). With this new addition, the implicit descriptions of the m shape classes are given by the zero level set of the following m level set functions:

$$\Phi^1[\mathbf{w},\mathbf{p}](x,y) = \bar{\Phi}^1(\tilde{x},\tilde{y}) + \sum_{i=1}^{q} w_i \Phi_i^1(\tilde{x},\tilde{y}), \tag{7a}$$

$$\Phi^2[\mathbf{w},\mathbf{p}](x,y) = \bar{\Phi}^2(\tilde{x},\tilde{y}) + \sum_{i=1}^{q} w_i \Phi_i^2(\tilde{x},\tilde{y}), \tag{7b}$$

$$\vdots$$

$$\Phi^m[\mathbf{w},\mathbf{p}](x,y) = \bar{\Phi}^m(\tilde{x},\tilde{y}) + \sum_{i=1}^{q} w_i \Phi_i^m(\tilde{x},\tilde{y}), \tag{7c}$$

where

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \\ 1 \end{bmatrix} = T[\mathbf{p}] \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

with $T[\mathbf{p}]$ defined in Eq. (1). The addition of $\mathbf{p}$ to our parametric shape model enables us to accommodate a larger class of objects. In particular, the model can now handle object shapes that may differ from each other in terms of scale, orientation, or translation. In the next section, we describe how $\mathbf{w}$ and $\mathbf{p}$ of Eq. (7a) are optimized, via coordinate descent, for image segmentation.

4. Mutual information-based segmentation model

Mutual information-based models (Kim et al., 2002; Unal et al., 2002) view the segmentation problem as a region labeling process with the objective of the process being to maximize the mutual information between the image pixel intensities and the segmentation labels. A

[6] The number of modes we consider in our shape representation, q, should be chosen large enough to be able to capture the prominent shape variations present in the training set, but not so large that the model begins to capture intricate details particular to a certain training shape. In all of our examples, we chose q empirically.

generalization of the mutual information-based energy functional E_{MI}, proposed by Kim et al. (2002) for image segmentation, to handle $m+1$ regions is given by

$$\begin{aligned} E_{\mathrm{MI}} &= -\hat{\mathscr{I}}(I;L) \\ &= -\hat{h}(I) + \left(\sum_{i=1}^{m} P_{R_i} \hat{h}(I|L=R_i) \right) + P_{R^c} \hat{h}(I|L=R^c) \\ &\sim \left(\sum_{i=1}^{m} P_{R_i} \hat{h}(I|L=R_i) \right) + P_{R^c} \hat{h}(I|L=R^c), \end{aligned} \qquad (8)$$

where L is the segmentation label determined by the segmenting curve $\vec{C}$, $\hat{\mathscr{I}}$ is the estimate of the mutual information $\mathscr{I}$ between the test image I and the segmentation label L, P_{R_i} denotes the prior probability of pixel values in the ith region R_i, P_{R^c} denotes the prior probability of pixel values in R^c (the area outside of the m regions), $\hat{h}(\cdot)$ is the estimate of the differential entropy $h(\cdot)$, and $\hat{h}(\cdot|\cdot)$ is the estimate of the conditional differential entropy $h(\cdot|\cdot)$. The estimate of the differential entropy $\hat{h}(I)$ provides an estimate of the amount of randomness in the test image I. This term is removed from E_{MI} because it is independent of the segmentation label L, and hence the segmenting curve $\vec{C}$. Each estimate of the $m+1$ conditional differential entropy terms (i.e. $\hat{h}(I|L=R_i)$ for $i=1,\ldots,m$ and $\hat{h}(I|L=R^c)$) quantifies the randomness of I conditioned on the segmentation label L.

Let $p_{R_i}(I)$ and $p_{R^c}(I)$ denote the probability density function (pdf) of I in regions R_i and R^c, respectively. By using weak law of large numbers to approximate entropy, the estimates of the conditional differential entropy terms $\hat{h}(I|L=R_i)$ for $i=1,\ldots,m$ and $\hat{h}(I|L=R^c)$ are given by Kim et al. (2002):

$$\begin{aligned} \hat{h}(I|L=R_i) &= -\frac{1}{A_{R_i}} \int\int_{R_i} \log(\hat{p}_{R_i}(I))\,\mathrm{d}A \\ &= -\frac{1}{A_{R_i}} \int\int_{\Omega} \log(\hat{p}_{R_i}(I)) \mathscr{H}(-\Phi^i)\,\mathrm{d}A, \end{aligned} \qquad (9)$$

$$\begin{aligned} \hat{h}(I|L=R^c) &= -\frac{1}{A_{R^c}} \int\int_{R^c} \log(\hat{p}_{R^c}(I))\,\mathrm{d}A \\ &= -\frac{1}{A_{R^c}} \int\int_{\Omega} \log(\hat{p}_{R^c}(I)) \prod_{j=1}^{m} \mathscr{H}(\Phi^j)\,\mathrm{d}A, \end{aligned} \qquad (10)$$

where the Heaviside function $\mathscr{H}$ is given by

$$\mathscr{H}(\Phi^i) = \begin{cases} 1 & \text{if } \Phi^i \geqslant 0, \\ 0 & \text{if } \Phi^i < 0, \end{cases} \quad \text{for } i=1,\ldots,m,$$

$\hat{p}_{R_i}(I)$ and $\hat{p}_{R^c}(I)$ are estimates of $p_{R_i}(I)$ and $p_{R^c}(I)$, respectively, and Ω denotes the image domain. We apply the non-parametric Parzen window method (Parzen, 1962) and use the bandwidth recommended by Silverman (1986) to estimate these densities from the training data. Importantly, these density estimates are obtained "off-line" prior to the segmentation process.

The gradients of E_{MI}, taken with respect to $\mathbf{w}$ and $\mathbf{p}$, are given by

$$\nabla_{\mathbf{w}} E_{\mathrm{MI}} = \left(\sum_{i=1}^{m} P_{R_i} \nabla_{\mathbf{w}} \hat{h}(I|L=R_i) \right) + P_{R^c} \nabla_{\mathbf{w}} \hat{h}(I|L=R^c),$$

$$\nabla_{\mathbf{p}} E_{\mathrm{MI}} = \left(\sum_{i=1}^{m} P_{R_i} \nabla_{\mathbf{p}} \hat{h}(I|L=R_i) \right) + P_{R^c} \nabla_{\mathbf{p}} \hat{h}(I|L=R^c),$$

where the lth component of gradients $\nabla_{\mathbf{w}} h$ and $\nabla_{\mathbf{p}} h$ is given by

$$\nabla_{\mathbf{w}^l} \hat{h}(I|L=R_i) = \frac{1}{A_{R_i}} \left\{ \oint_{\vec{C}_i} \nabla_{\mathbf{w}^l} \Phi^i \log(\hat{p}_{R_i}(I))\,\mathrm{d}s \right\}, \qquad (11)$$

$$\nabla_{\mathbf{p}^l} \hat{h}(I|L=R_i) = \frac{1}{A_{R_i}} \left\{ \oint_{\vec{C}_i} \nabla_{\mathbf{p}^l} \Phi^i \log(\hat{p}_{R_i}(I))\,\mathrm{d}s \right\}, \qquad (12)$$

$$\nabla_{\mathbf{w}^l} \hat{h}(I|L=R^c) = \frac{-1}{A_{R^c}} \left\{ \sum_{j=1}^{m} \oint_{\vec{C}_j} \nabla_{\mathbf{w}^l} \Phi^j \log(\hat{p}_{R^c}(I))\,\mathrm{d}s \right\}, \qquad (13)$$

$$\nabla_{\mathbf{p}^l} \hat{h}(I|L=R^c) = \frac{-1}{A_{R^c}} \left\{ \sum_{j=1}^{m} \oint_{\vec{C}_j} \nabla_{\mathbf{p}^l} \Phi^j \log(\hat{p}_{R^c}(I))\,\mathrm{d}s \right\} \qquad (14)$$

with $\nabla_{\mathbf{w}^l} \Phi^i$, $\nabla_{\mathbf{p}^l} \Phi^i$, $\nabla_{\mathbf{w}^l} \Phi^j$, and $\nabla_{\mathbf{p}^l} \Phi^j$ defined as

$$\nabla_{w^l} \Phi^i = \Phi^i_l,$$

$$\nabla_{p^l} \Phi^i = \nabla_{p^l} \Phi^i(\tilde{x},\tilde{y}) = \begin{bmatrix} \frac{\partial \Phi^i(\tilde{x},\tilde{y})}{\partial \tilde{x}} & \frac{\partial \Phi^i(\tilde{x},\tilde{y})}{\partial \tilde{y}} & 0 \end{bmatrix} \frac{\partial T[p^l]}{\partial p^l} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix},$$

$$\nabla_{w^l} \Phi^j = \Phi^j_l,$$

$$\nabla_{p^l} \Phi^j = \nabla_{p^l} \Phi^j(\tilde{x},\tilde{y}) = \begin{bmatrix} \frac{\partial \Phi^j(\tilde{x},\tilde{y})}{\partial \tilde{x}} & \frac{\partial \Phi^j(\tilde{x},\tilde{y})}{\partial \tilde{y}} & 0 \end{bmatrix} \frac{\partial T[p^l]}{\partial p^l} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

and $\frac{\partial T[p^l]}{\partial p^l}$ previously defined in Eq. (4a).

The update equations for the shape parameter $\mathbf{w}$ and the pose parameter $\mathbf{p}$ are then given, in terms of $\nabla_{\mathbf{w}} E_{\mathrm{MI}}$ and $\nabla_{\mathbf{p}} E_{\mathrm{MI}}$, as

$$\mathbf{w}^{t+1} = \mathbf{w}^t - \Delta t_{\mathbf{w}} \nabla_{\mathbf{w}} E_{\mathrm{MI}}, \qquad (15)$$

$$\mathbf{p}^{t+1} = \mathbf{p}^t - \Delta t_{\mathbf{p}} \nabla_{\mathbf{p}} E_{\mathrm{MI}}. \qquad (16)$$

Eqs. (15) and (16) are used to update $\mathbf{w}$ and $\mathbf{p}$, respectively, in an alternating fashion, to minimize Eq. (8).

As mentioned previously in Section 3.2, since there are no restrictions on the range of values that $\mathbf{w}$ can take, calculating $\mathbf{w}$ based on Eq. (15) can result in the overlap of the various shapes. One way to avoid this problem is to constrain the value of $\mathbf{w}$ during its calculation so that the various shapes never overlap. A heuristic approach is to perform a check at each minimization step prior to updating $\mathbf{w}$ to see if updating $\mathbf{w}$ causes any of the m shapes to overlap. If updating $\mathbf{w}$ causes an overlap, this particular update of $\mathbf{w}$ is not

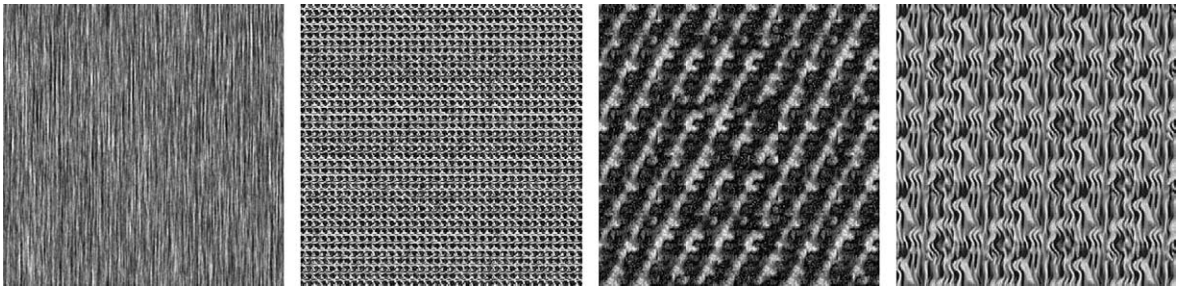


Fig. 8. Palette showing a close-up view of the four textures used: (a) vertical wood grain, (b) fine fabric, (c) diagonal knit, and (d) rocky terrain.

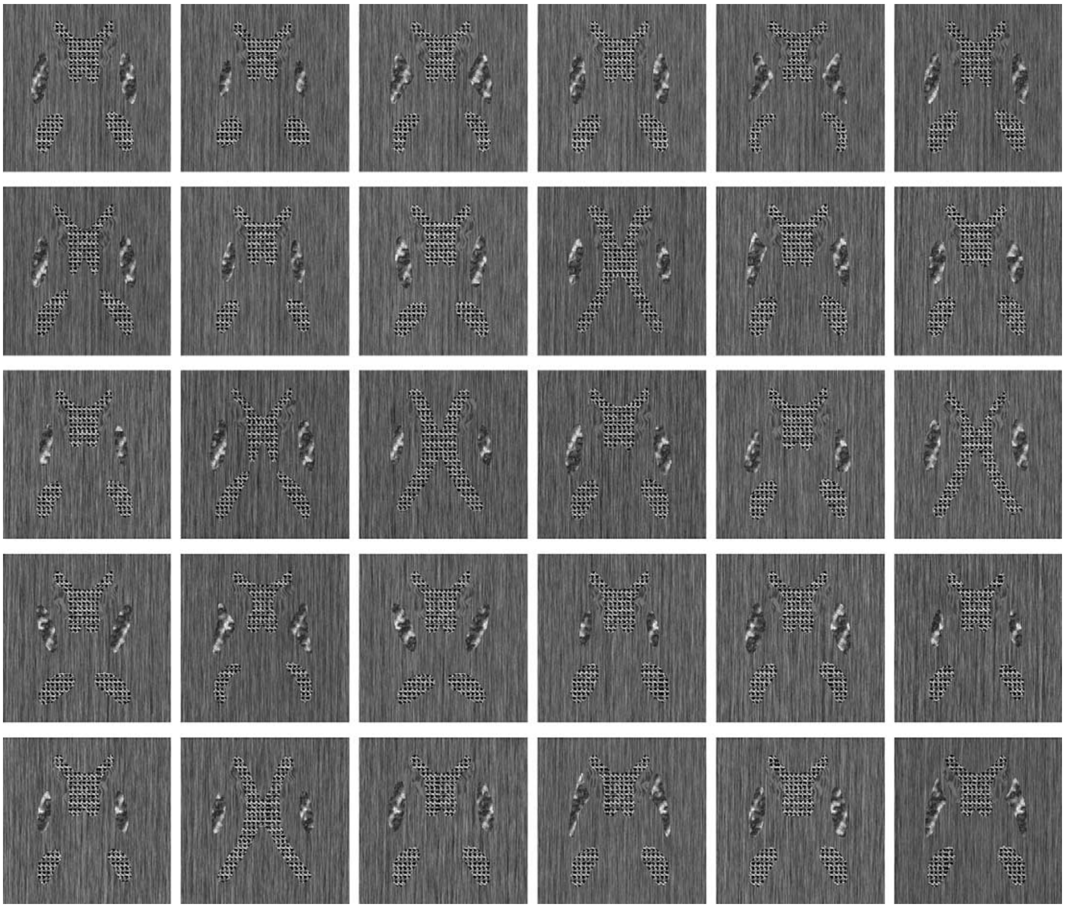
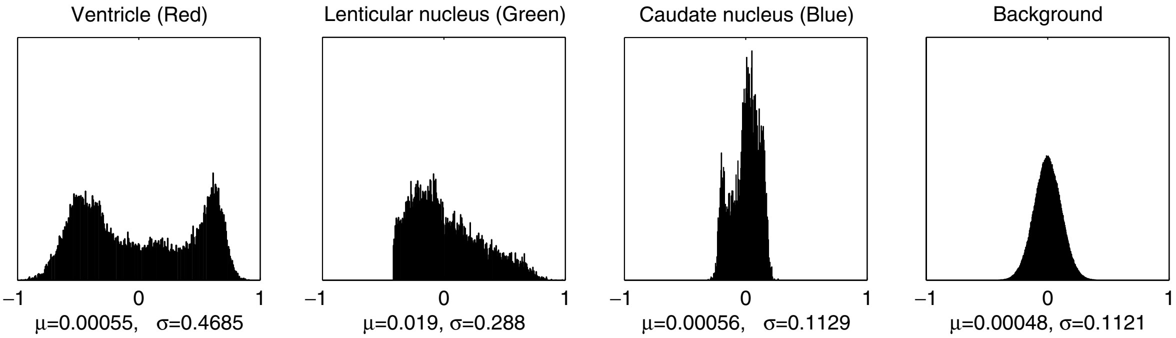


Fig. 9. Training collection containing 30 brain texture images. The four texture types shown in Fig. 8 and the 30 training template images shown in Fig. 5 are used to generate these images.

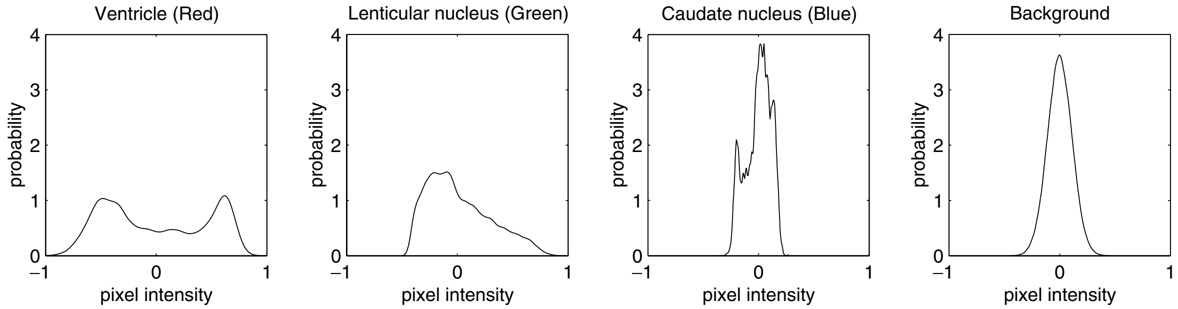
performed and the algorithm skips forward to the next step which involves updating **p**.

We employ texture images to illustrate this powerful image segmentation methodology. Fig. 8 shows realizations of four different textures. The textures are all purposely scaled to have approximately the same mean intensity and the same dynamic range of -1 to 1 to make the segmentation process even more difficult. Texture A is an example of a vertical wood grain texture. Texture B is an example of a fine fabric texture. Texture C is an example of a diagonal knit texture. And Texture D is an example of a rocky terrain. Segmentation algorithms need to capture the complicated statistical dependencies between a particular pixel and its neighbors within each texture in order to successfully segment these images. Using the subcortical brain images shown in Fig. 5 as templates and 30 different realizations of each of the four textures shown in Fig. 8, we construct the 30 different subcortical brain texture images shown in Fig. 9. These newly generated subcortical brain texture images serve as our training collection. Notice that the various regions within these texture images are not



Ventricle (Red) Lenticular nucleus (Green) Caudate nucleus (Blue) Background

Fig. 10. Histogram of the intensity distributions in the various regions.



Ventricle (Red) Lenticular nucleus (Green) Caudate nucleus (Blue) Background

Fig. 11. Parzen density estimate of the pixel intensities in the various regions.

separated from each other by large image gradients or characterized by different mean intensities. In fact, the mean intensities of each of the texture classes are all roughly zero. [7] This means that neither edge- nor region-based techniques will be successful in segmenting these texture images. The histograms of the pixel intensity distributions in the various subcortical brain regions of the 30 texture images are shown in Fig. 10. Notice that the mean and the standard deviation of the pixel intensities within the caudate nucleus and the background are approximately the same. This makes the separation of the caudate nucleus from the background very challenging. This observation is also visually validated by looking at Fig. 9. The boundaries of the caudate nucleus in all 30 brain texture images are extremely difficult to find. Using non-parametric Parzen windowing method (Parzen, 1962), we obtain pdf estimates of the pixel intensities in each of the subcortical brain regions. These density estimates are shown in Fig. 11.

Figs. 12 and 13 demonstrate the performance of the mutual information-based image segmentation model described in Eq. (8). In frame (a) of each figure, a synthetic image of the subcortical brain structures is displayed. These two synthetic images are similar to the subcortical brain images shown in the training collection of Fig. 2 but neither are part of that collection. Using these two synthetic images as the template, we generated the subcortical brain texture images shown in frame (b) of each figure. Frame (c) of each figure then shows the subcortical brain texture image contaminated by additive Gaussian noise. These two resulting images serve as the test images. We employ the pose-aligned collection shown in Fig. 5 to derive an implicit parametric multishape model for the subcortical brain structures in the form of Eq. (7a). In the segmentation examples shown in Figs. 12 and 13, we choose $q = 15$. The zero level sets of $\bar{\Phi}^i[\mathbf{w},\mathbf{p}]$ for $i = 1, 2, 3$ are employed as the starting curves. The locations of the initializing contours are shown in frame (d) of each figure. Frame (e) of each figure shows the final positions of the segmenting curves after evaluating Eqs. (15) and (16) until convergence. The size of both these images are 300×300 pixels. On an Intel Xeon 4.4GHz dual processor personal computer, it took 283 s to generate the segmentation result of Fig. 12 and 236 s to generate the segmentation result of Fig. 13.

Notice that in both test images, despite having very similar textures between the background and the caudate nucleus, our algorithm is still able to find the boundaries of the caudate nucleus. This is due to the strong spatial coupling and shape co-variability between the various subcortical brain structures. Notice that triple points formed by the ventricle and the caudate nucleus are easily captured by our algorithm. In addition, Fig. 13 shows that without any additional effort, the three starting curves that describe the ventricle

[7] The mean intensities of each texture class were removed so that they all have the same mean intensity of zero. This is aimed to increase the difficulty of the segmentation process.

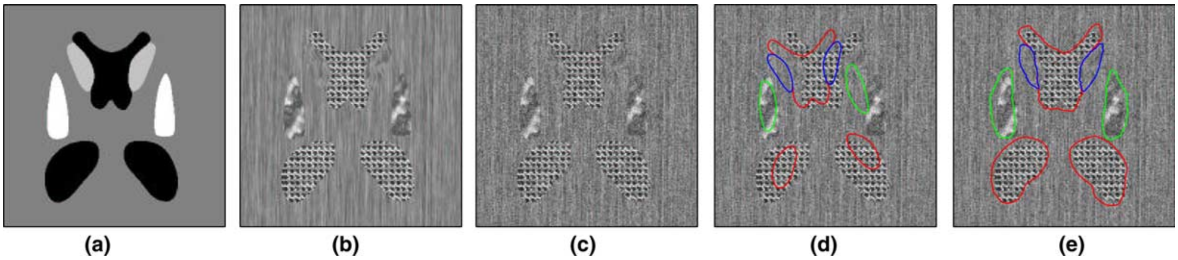


Fig. 12. Performance of the mutual information-based model described in Eq. (8): (a) subcortical brain template image, (b) subcortical brain texture image based on the template image shown in (a), (c) test image I obtained after adding Gaussian noise to image shown in (b), (d) starting location of the segmenting curve, (e) final location of the segmenting curve.

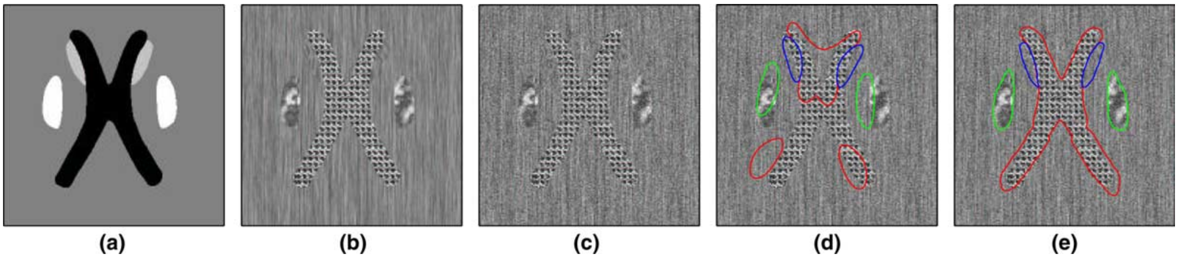


Fig. 13. Another demonstration of the mutual information-based model showing topological changes of the shapes: (a) subcortical brain template image, (b) subcortical brain texture image based on the template image shown in (a), (c) test image I obtained after adding Gaussian noise to image shown in (b), (d) starting location of the segmenting curve, (e) final location of the segmenting curve.

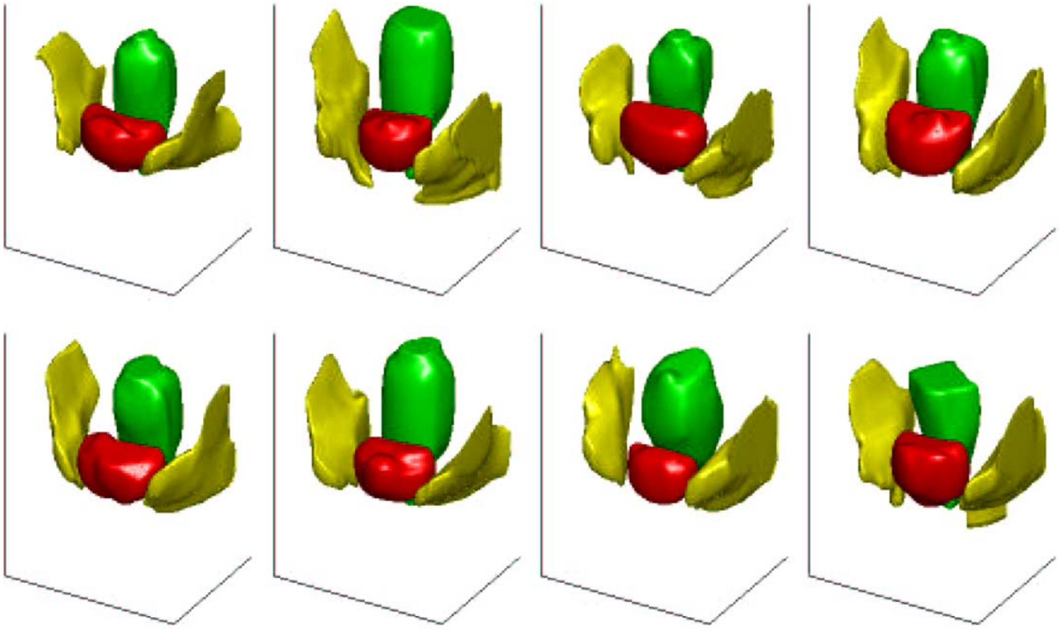


Fig. 14. The 3D models all eight patients pelvic structures after alignment.

merged to form one single segmenting curve at the end. This ability to handle topological changes of the shapes in a seamless fashion is a very attractive feature associated with our algorithm.

As evident in this contrived example, one of the important features associated with having distinct signed distance functions representing the different shape classes is that we are able to capture, and later utilize, the information content particular to a shape class for segmentation. In other words, suppose the different shape classes in this synthetic example are non-overlapping and non-adjacent to one another, and one would like to use a single signed distance function to represent all the three shape classes. In this case, the statistical

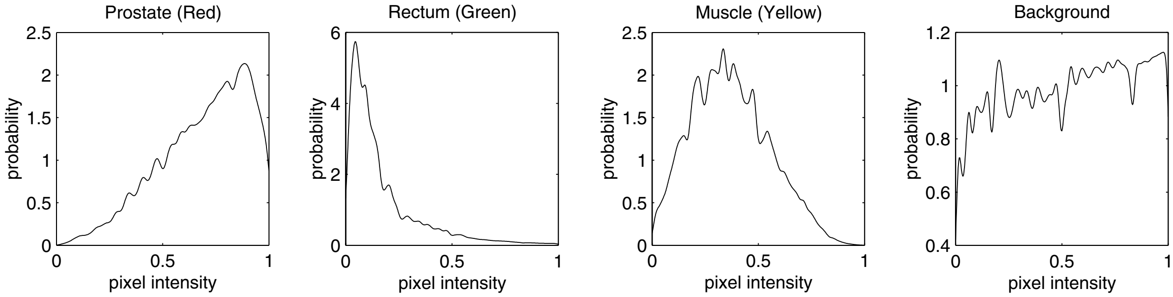


Fig. 15. Parzen density estimate of the pixel intensities in the various regions.

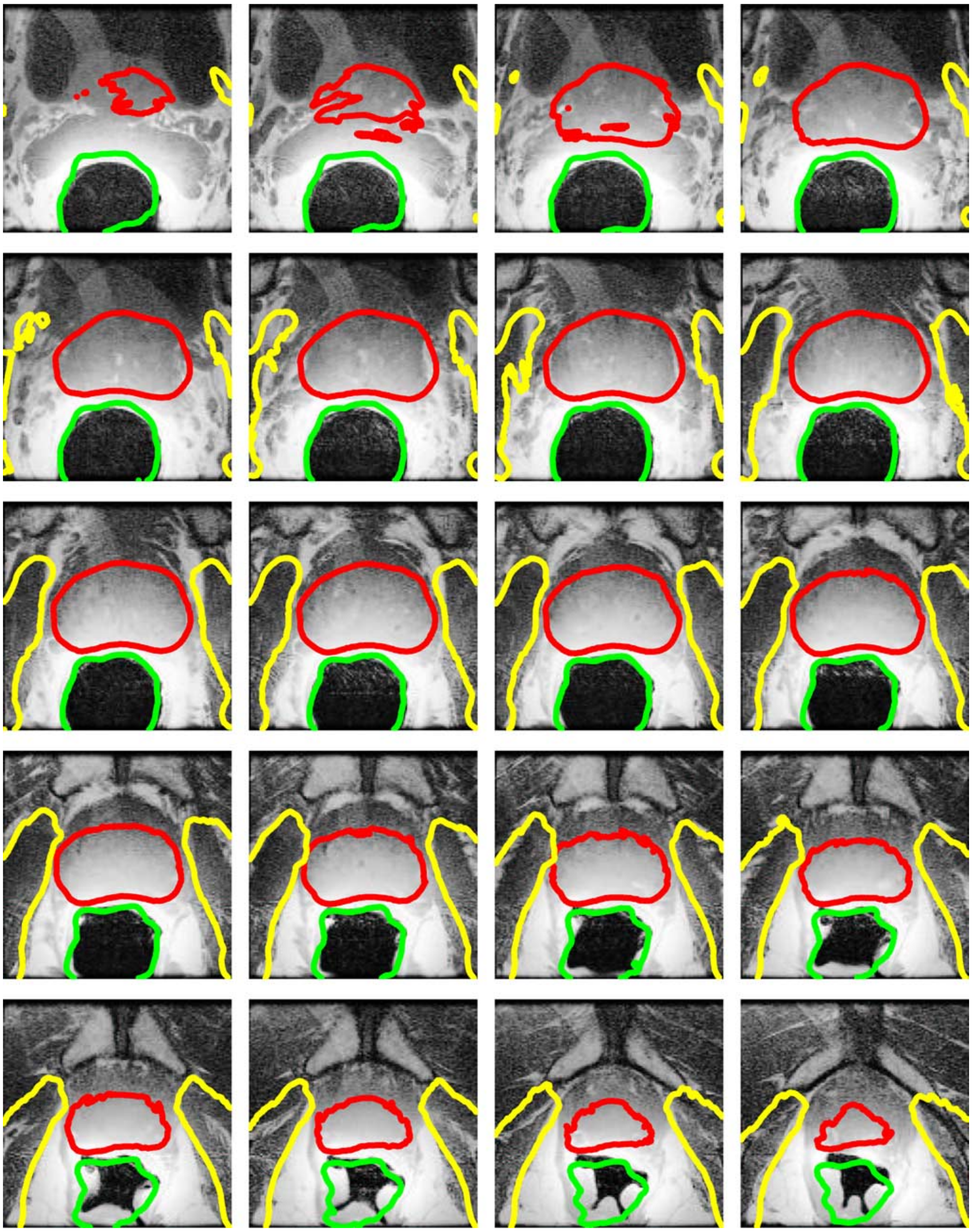


Fig. 16. Segmentation of a new patient's pelvic MR image using the mutual information-based model. The segmentations of the prostate (red), the rectum (green), and the internal obturator muscles (yellow) are shown.

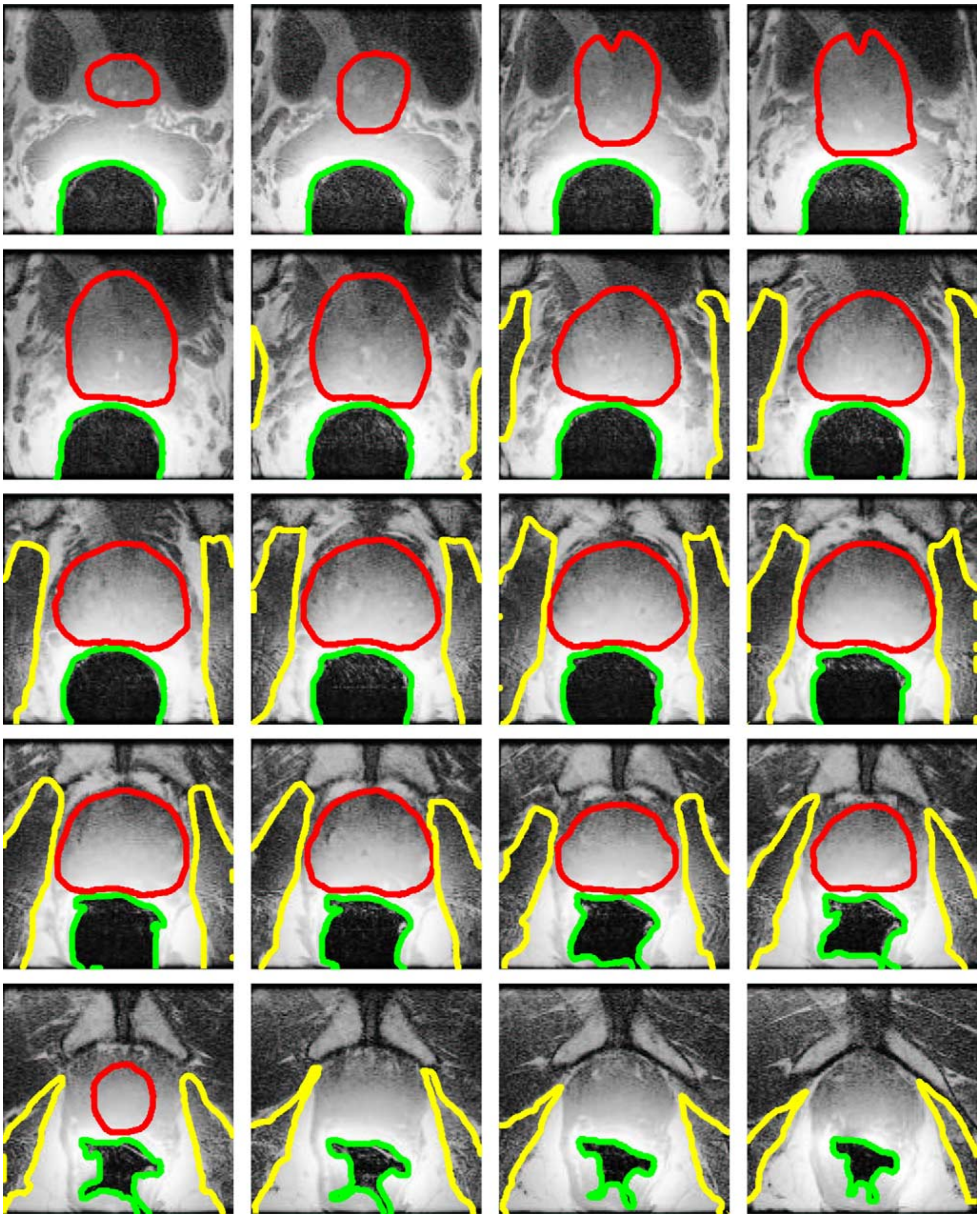


Fig. 17. Manual segmentation of a new patient's pelvic MR image. The segmentations of the prostate (red), the rectum (green), and the internal obturator muscles (yellow) are shown.

information particular to each shape class will be lost as they will be combined together into the statistical information of a single shape class. Each original shape class' distinguishing features will be lost and cannot be utilized for segmentation.

5. Application to medical imagery

Our strategy to segment the prostate gland from a pelvic MR volumetric dataset for prostate brachytherapy is to use easily identifiable structures within the pelvic MR dataset to help localize the prostate gland. The most prominent structure within the pelvic MR dataset is the dark-colored rectum. The prostate gland is flanked on either side by the internal obturator muscles which are also easy to find. The prostate gland, the rectum, and the internal obturator muscles form the three shape classes in our parametric multi-shape segmentation algorithm.

We employ eight hand segmented 3D models of the prostate, the rectum, and the internal obturator muscles as our training set. The 3D version of the alignment procedure is used to (1) jointly align the eight example shapes from each shape class, and (2) simultaneously perform this task on all three shape classes. Fig. 14

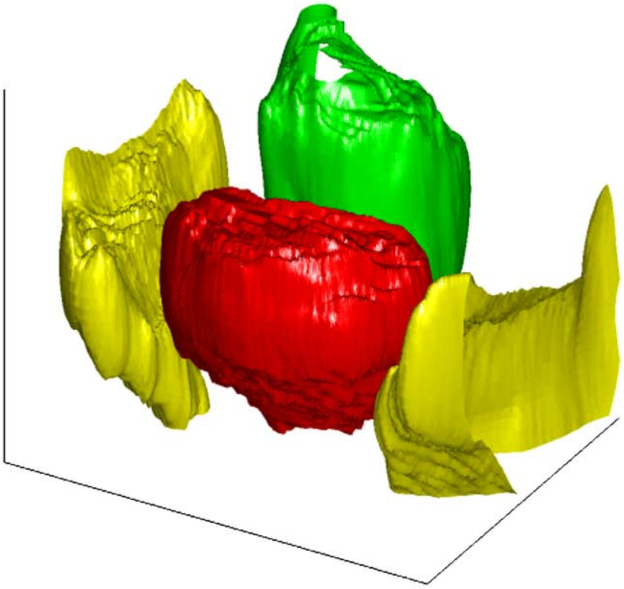


Fig. 18. The 3D representation of the segmentation results shown in Fig. 16.

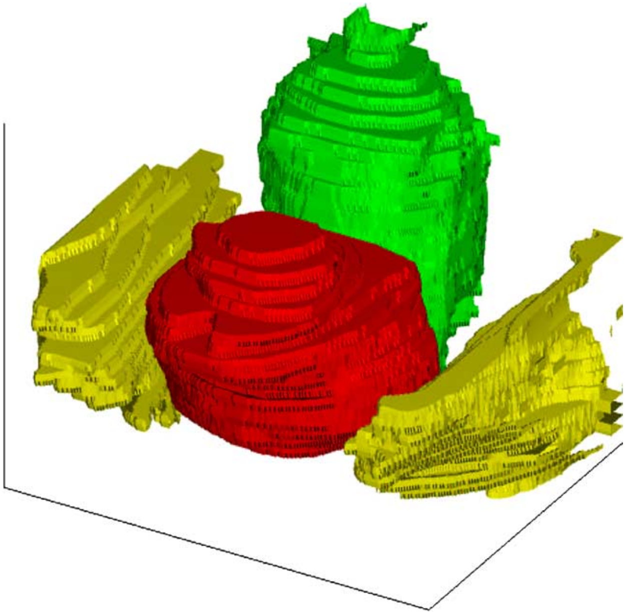


Fig. 19. The 3D representation based on a hand segmentation.

displays all eight hand segmented 3D models of the three pelvic structures after alignment. The shape classes are color coded as follows: the rectum (green), the prostate gland (red), and the internal obturator muscles (yellow). Next, we employed the 3D version of our shape modeling approach to obtain a 3D multi-shape parametric model of the three pelvic structures. We then employ the mutual information-based segmentation model described in Section 4 as our segmentation model. In order to implement this model, however, the probability density function of the pixel intensities within the different regions need to be estimated. We apply the non-parametric Parzen window method to the eight pelvic MR volumetric datasets to estimate these probability densities within the three pelvic structures and the background region. [8] The Parzen density estimate of the pixel intensities in the three pelvic structures and the background region are shown in Fig. 15. [9]

Fig. 16 shows 20 consecutive axial slices of the pelvic MR volumetric dataset of a new patient. The segmentation results of this dataset, based on our algorithm, are shown in this figure as well. The boundaries of the rectum, the prostate gland, and the internal obturator muscles, as determined by our algorithm, are shown in green, red and yellow contours, respectively. For comparison, we show in Fig. 17 the manual segmentation of this dataset. Fig. 18 shows the 3D representations of this new patient's rectum, prostate gland, and internal obturator muscles obtained by our algorithm. Again, for comparison, we show in Fig. 19 the hand segmentations of the same patient's rectum, prostate gland, and the internal obturator muscles. This example demonstrates that, even with a very limited dataset, our algorithm performed very well in segmenting the anatomical structures of interest. By having a larger dataset for training, our algorithm should be able to perform even better.

6. Conclusions

We presented an unified analytical formulation that extends the work of Tsai et al. (2001) to multiple shapes. In particular, we described a new multi-shape modeling approach that (1) can capture important co-variations shared among the different shape classes, (2) allows multiple junctions to be captured automatically, (3) does not require point correspondences during the training phase of the algorithm, (4) can handle topological changes of the shapes in a seamless fashion, and (5) can be extended from a 2D to a 3D framework in a straightforward manner. We then showed the utility of this parametric multi-shape model by incorporating it within a mutual information-based framework for medical image segmentation.

It is also important to realize that the parametric multi-shape model that we proposed in this paper is very flexible. It is flexible enough that it can be incorporated into other segmentation model besides the mutual information-based model demonstrated in this paper. We

[8] The eight pelvic MR volumetric datasets used here are the same eight volumetric datasets from which we derived the 3D models shown in Fig. 14.

[9] The pixel intensities from each of the eight MR dataset are scaled down so they all range between 0 and 1 for ease of manipulation and display.

show, in (Tsai, 2003), how our multi-shape model is incorporated into an edge-based and a region-based model for image segmentation. In addition, in (Tsai, 2003), we demonstrate and validate, through simulation studies, some of the unique properties associated with our parametric multi-shape model that we have already mentioned in this paper, namely, the ability to (1) capture important co-variations among the different shape classes, (2) display a wide range of shape variability, and (3) handle large amounts of additive noise.

Acknowledgements

This work was supported in part by Office of Naval Research (ONR) Grant N00014-00-1-0089, Air Force Office of Scientific Research (AFOSR) Grant F49620-98-1-0349, National Science Foundation (NSF) ERC Grant under Johns Hopkins Agreement 8810274, National Institute of Health Grants P41RR13218, P01CA67167, R33CA99015, R21CA89449, and R01 AG19513-01. The authors acknowledge the reviewers for their valuable comments and suggestions.

References

Bello, F., Colchester, A., 1998. Measuring global and local spatial correspondence using information theory. Medical Computing and Computer-Assisted Intervention 1, 964–973.

Borgefors, G., 1986. Distance transformations in digital images. CVGIP: Image Understanding 34, 344–371.

Chan, T., Vese, L., 2000. An efficient variational multiphase motion for the Mumford–Shah segmentation model. Asilomar Conference on Signals, Systems, and Computers 1, 490–494.

Cootes, T., Taylor, C., Cooper, D., Graham, J., 1995. Active shape models – their training and application. IEEE Computer Vision and Image Understanding 61, 38–59.

Cover, T., Thomas, J., 1991. Elements of Information Theory, first ed. Wiley, New York.

Frangi, A., Rueckert, D., Schnabel, J., Niessen, W., 2002. Automatic construction of multiple-object three-dimensional statistical shape models: application to cardiac modeling. IEEE Transactions on Medical Imaging 21, 1151–1166.

Kim, J., Fisher, J., Yezzi, A., Cetin, M., Willsky, A., 2002. Nonparametric methods for image segmentation using information theory. IEEE International Conference on Image Processing. 3, 797–800.

Leventon, M., Grimson, E., Faugeras, O., 2000. Statistical shape influence in geodesic active contours. IEEE Conference on Computer Vision and Pattern Recognition 1, 316–323.

Osher, S., Sethian, J., 1988. Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulation. Journal of Computational Physics 79, 12–49.

Paragios, N., Rousson, M., 2002. Shape priors for level set representation. European Conference on Computer Vision.

Parzen, E., 1962. On estimation of a probability density function and model. Annals of Mathematical Statistics 33, 1065–1076.

Silverman, B., 1986. Density Estimation for Statistics and Data Analysis. Chapman & Hall, London.

Staib, L., Duncan, J., 1992. Boundary finding with parametrically deformable contour models. IEEE Transactions on Pattern Analysis and Machine Intelligence 14, 1061–1075.

Turk, M., Pentland, A., 1991. Eigenfaces for recognition. Journal of Cognitive Neuroscience 3, 71–86.

Tsai, A., 2003. Coupled multi-shape model for medical image segmentation: a general framework utilizing region statistics, edge information, and information-theoretic criteria. M.D. Thesis, Harvard University, Harvard Medical School, March.

Tsai, A., Yezzi, A., Wells, W., Tempany, C., Tucker, D., Fan, A., Grimson, E., Willsky, A., 2001. Model-based curve evolution technique for image segmentation. IEEE Conference on Computer Vision and Pattern Recognition 1, 463–468.

Unal, G., Krim, H., Yezzi, A., 2002. A vertex-based representation of objects in an image. IEEE International Conference on Image Processing 1, 896–899.

Viola, P., Wells, W., 1995. Alignment by maximization of mutual information. IEEE International Conference on Computer Vision 1, 16–23.

Wang, Y., Staib, L., 1998. Boundary finding with correspondence using statistical shape models. IEEE Conference on Computer Vision and Pattern Recognition 1, 338–345.

Wells III, W.M., Viola, P., Atsumi, H., Nakajima, S., Kikinis, R., 1996. Multi-modal volume registration by maximization of mutual information. Medical Image Analysis 1, 35–51.

Yezzi, A., Tsai, A., Willsky, A., 1999. A statistical approach to snakes for bimodal and trimodal imagery. IEEE Internatinal Conference on Computer Vision 2, 898–903.

Yezzi, A., Tsai, A., Willsky, A., 2002. A fully global approach to image segmentation via coupled curve evolution equations. Journal of Visual Communication and Image Representation 13, 195–216.

Zeng, X., Staib, L., Schultz, R., Duncan, J., 1999. Segmentation and measurement of the cortex from 3-d MR images using coupled-surfaces propagation. IEEE Transactions on Medical Imaging 18, 927–937.