Shape-Based Averaging

Torsten Rohlfing, *Member, IEEE*, and Calvin R. Maurer, Jr., *Member, IEEE*

***Abstract*—A new method for averaging multidimensional images is presented, which is based on signed Euclidean distance maps computed for each of the pixel values. We refer to the algorithm as "shape-based averaging" (SBA) because of its similarity to Raya and Udupa's shape-based interpolation method. The new method does not introduce pixel intensities that were not present in the input data, which makes it suitable for averaging nonnumerical data such as label maps (segmentations). Using segmented human brain magnetic resonance images, SBA is compared to label voting for the purpose of averaging image segmentations in a multiclassifier fashion. SBA, on average, performed as well as label voting in terms of recognition rates of the averaged segmentations. SBA produced more regular and contiguous structures with less fragmentation than did label voting. SBA also was more robust for small numbers of atlases and for low atlas resolutions, in particular, when combined with shape-based interpolation. We conclude that SBA improves the contiguity and accuracy of averaged image segmentations.**

***Index Terms*—Combination of segmentations, shape-based averaging (SBA), shape-based interpolation (SBI), signed Euclidean distance transform.**

I. Introduction

AVERAGING of multiple segmentations of the same image has recently been introduced as an effective method to obtain segmentations that are more accurate than any of the individual input segmentations [1]–[3]. Analogous to multiple classifier systems in numerous other pattern recognition applications, the underlying assumption is that each individual classifier makes different errors, so that errors made by any one of the classifiers are corrected by the others.

Typically, averaging algorithms for image segmentations are based on local (i.e., pixel-wise) decision fusion schemes, such as voting [3], or on probability-theoretical combination of Bayesian classifiers that assign likelihoods to the possible output classes [4], [5]. The combination can optionally include an estimate of the individual classifiers' performances, which has been found to potentially improve the combined classifier performance [2], [6]. All aforementioned techniques essentially operate locally on the individual image pixels. As Fig. 1 illustrates, this can lead to fragmentation of structures that are contiguous in all input images. This indicates that pixel-wise operations can deal with pixel-wise perturbations, e.g., image noise in local classifiers, but have difficulty in the presence of spatial uncertainty, e.g., positional "noise."

To address the fragmentation problem, and to specifically target positional uncertainty in the input images, the present paper introduces a novel shape-based averaging (SBA) method. As one possible application, we propose this method as a new way to combine multiple segmentations of multidimensional images. Unlike many other classification problems, in image segmentation there is a natural distance relationship between the pixels of an n-dimensional image. We exploit this relationship to average segmentations based on the signed Euclidean distance maps of the labels in each input segmentation. Our method is related to shape-based interpolation (SBI), which was introduced by Raya and Udupa [7] as a method for the interpolation of binary images. By interpolating from a distance map of the image, this technique achieves a smooth approximation to the shape that is represented by the discretely sampled binary image. Grevera and Udupa [8] later extended the method to multivalued and, in particular, gray-level images by embedding an n-dimensional gray-level image into an $(n+1)$-dimensional binary image space. Our approach is similar in that we consider images with multiple classes of a segmentation by computing a separate distance map for each label. However, our approach is different insofar as it combines multiple images into one image, rather than resamples a single image onto a new grid. In this sense, our method is a shape-based averaging method.[1]

The basis of the SBA method, much like SBI, is a distance transform, which assigns to each pixel in an image its distance from the nearest "feature" pixel in that image. We use here in particular the Euclidean distance transform (EDT). For efficiency reasons, the EDT has long been approximated using distance propagation (e.g., the chamfer distance transform [10]). In the past decade, however, several efficient algorithms have been introduced for computing the exact EDT [11]–[13]. We use in this work the algorithm described by Maurer *et al.* [13], which computes the exact EDT, generalizes to images of arbitrary dimensions (e.g., the algorithm in [12] handles only 2-D images), can handle anisotropic pixel sizes, and has a serial computational complexity that is guaranteed to be linear in the number of image pixels independent of the input data (e.g., the worst-case time complexity of the algorithm in [11] is not linear in the number of image pixels).

The remainder of this paper is organized as follows. In Section II, we introduce the notation that is used throughout the

Manuscript received October 10, 2005; revised June 21, 2006. A preliminary version of this paper was presented at the 8th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), October 26–29, 2005, Palm Springs, CA. T. Rohlfing was supported by the National Institute on Alcohol Abuse and Alcoholism under Grant AA05965. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ercan E. Kuruoglu.

T. Rohlfing is with SRI International, Menlo Park, CA 94025-3493 USA (e-mail: torsten@synapse.sri.com).

C. R. Maurer, Jr., is with Accuray, Inc., Sunnyvale, CA 94089 USA (e-mail: calvin.maurer@gmail.com).

Digital Object Identifier 10.1109/TIP.2006.884936

[1]In a later paper by Grevera and Udupa [9], the term "shape-based averaging" is used for a method that uses averaging between two 2-D slices for the purpose of interpolation. This method is fundamentally different from the method introduced here.

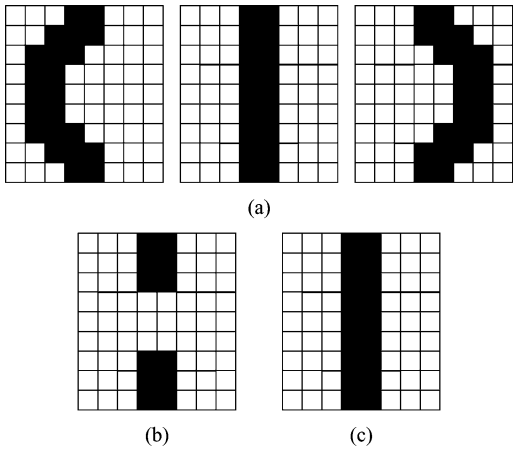

Fig. 1. Fragmentation of averaged discrete binary segmentations of the same structure (illustrative example). (a) Contiguous structure in three binary images. (b) Majority decision (label voting) for each pixel produces fragmented output. (c) In the desired output image the structure is contiguous, just as it is in the input images.

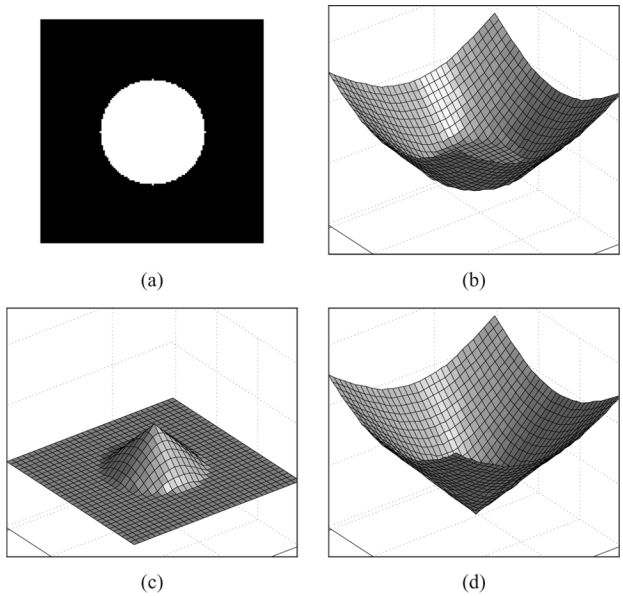

Fig. 2. Examples of outside, inside, and signed distance maps. (a) Binary image of a circle. (b) Surface plot of outside distance map. Note the flat region at level zero inside the boundary of the circle. (c) Inside distance map. (d) Signed distance map, which is the outside map minus the inside map.

paper. This is followed by a description of the basic SBA algorithm and a unified combined shape-based interpolation and averaging (SBIA) algorithm. The function of the SBA algorithm is illustrated here using a simple synthetic example. We then review the reference method for averaging segmentations, label voting, which is used for the subsequent evaluation. In Section III, an initial simulation study uses segmentations with controlled error levels and known ground truth to quantify the accuracies and levels of fragmentation of averaged segmentations. Next, the evaluation is repeated using actual atlas-based segmentations that were generated by nonrigid registration of human brain magnetic resonance (MR) images to presegmented atlas images. We also demonstrate a possible application of the algorithm to intensity images and provide an analysis of its run-time complexity and memory requirements with some exemplary benchmark results. Section IV discusses the results and provides additional context for their interpretation.

II. Methods

Notation

1) Images: For K input images over $\mathbb{R}^m$ and $k = 1, \ldots, K$, let $s_k(\vec{x})$ be the pixel value at location $\vec{x} \in \mathbb{R}^m$ in kth image. Furthermore, let L be the number of discrete values in the input images. This could be the number of classes in a segmentation, or the number of distinct gray levels. For simplicity, we shall identify each such value with a unique numerical label. Each label is a number in the set $\Lambda = \{0, \ldots, L-1\}$, where zero without loss of generality represents the image background.

2) Distance Maps: Let $d_{k,l}(\vec{x})$ be the signed Euclidean distance of the pixel at $\vec{x}$ in image k with respect to label l. The signed EDT for label l is computed as the difference of two unsigned EDTs: 1) the outside EDT, which is the distance at each pixel from the nearest pixel with label l, and 2) the inside EDT, which is the distance at each pixel from the nearest pixel with a label other than l. Consequently, the outside EDT is zero for all pixels with label l, whereas the inside EDT is zero for all pixels with labels other than l. Without loss of generality, we adopt here the convention that the value of $d_{k,l}(\vec{x})$ is negative if $s_k(\vec{x}) = l$, and positive if $s_k(\vec{x}) \neq l$. The signed EDT for label l at each pixel is then simply the outside EDT minus the inside EDT. The result can be understood as the distance from the nearest "surface"[2] of a region with label l in input image k. The different types of distance maps are graphically illustrated in an example in Fig. 2.

A. Shape-Based Averaging

Based on the distance maps of all labels in all input images, we define the mean distance of pixel $\vec{x}$ from label l as

$$D_l(\vec{x}) = \frac{1}{K} \sum_{k=1}^{K} d_{k,l}(\vec{x}). \tag{1}$$

The combined output $S(\vec{x})$ for pixel $\vec{x}$ is now determined by minimizing the mean distance over all labels

$$S(\vec{x}) = \underset{l \in \Lambda}{\operatorname{argmin}}\, D_l(\vec{x}). \tag{2}$$

The SBA algorithm to iteratively compute S is presented in Fig. 3. Recall that the convention used for the signed distance from an object in this paper is that it is negative inside the object, and positive outside; thus, the minimization (as opposed to maximization) in (2). Note also that the number of inputs K in (1) is constant, at least as long as each label appears in each input image. The mean distance is, therefore, directly proportional to the total distance, so minimizing the two is equivalent. In case not every label appears in every input image, the average

[2]The "surface" concept is used only implicitly here for an intuitive explanation of the method, which does not actually create an explicit geometric model of any such surface.

Algorithm "Shape-Based Averaging"

Inputs: K images, Λ-valued over $\mathbb{R}^m$, $\Lambda = \{0, \ldots, L-1\}$
$s_k : \mathbb{R}^m \rightarrow \Lambda$, $k = 1, \ldots, K$

Internal: Signed distance map for each image k and label l,
$d_{k,l} : \mathbb{R}^m \rightarrow \mathbb{R}, k \in \{1, \ldots, K\}, l \in \Lambda$
Minimum average distance map,
$D_{\min} : \mathbb{R}^m \rightarrow \mathbb{R}$

Output: Combined image,
$S : \mathbb{R}^m \rightarrow \Lambda \cup \{L\}$

1: ▷ Loop over all pixels to initialize data structures
2: **for all** $\vec{x}$ **do**
3: ▷ Initialize output value to "reject"
4: $S(\vec{x}) \leftarrow L$
5: ▷ Initialize total distance map
6: $D_{\min}(\vec{x}) \leftarrow \infty$
7: **end for**
8: ▷ Loop over all labels
9: **for** $l = 0, \ldots, L-1$ **do**
10: ▷ Loop over all input images
11: **for all** $k = 1, \ldots, K$ **do**
12: ▷ Compute signed distance map for current label
13: $d_{k,l} \leftarrow$ signed distance map of label l, image k
14: **end for**
15: ▷ Loop over all pixels
16: **for all** $\vec{x}$ **do**
17: ▷ Avg. signed distances for this pixel and label
18: $D \leftarrow \frac{1}{K} \sum_k d_{k,l}(\vec{x})$
19: ▷ Is the new distance *smaller than* minimum?
20: **if** $D < D_{\min}(\vec{x})$ **then**
21: ▷ Update combined label map
22: $S(\vec{x}) \leftarrow l$
23: ▷ Update minimum average distance
24: $D_{\min}(\vec{x}) \leftarrow D$
25: ▷ Is the new distance *equal to* minimum?
26: **else if** $D = D_{\min}(\vec{x})$ **then**
27: ▷ Mark pixel as ambiguous ("reject")
28: $S(\vec{x}) \leftarrow L$
29: **end if**
30: **end for**
31: **end for**

Fig. 3. Shape-based averaging algorithm.

distance for each label in (1) could be computed over only those images that do contain the respective label.

The stages of the SBA algorithm are illustrated in Fig. 4 using a synthetic example with two spheres of opposite overlap. The average image contains two spheres with an average overlap, in which the surface between the two labeled regions is marked by what can be considered the spatial mean of the surfaces in the input images. Although shown as 2-D illustration, the actual computations were performed in 3-D space, and each of the images shown here represents the central 2-D slice through the respective 3-D image.

An important property for memory-efficient implementation of the SBA algorithm is that, independent of the number of classes L, at any given time it only requires space for three distance maps. These three maps are: 1) the individual distance map $d_{k,l}$ for the current input segmentation k and label l, 2) the average distance map D_l over all segmentations for label l, and 3) the minimum average distance map $D_{\min}$ over all labels so far. Note that this differs from the algorithm as presented in Fig. 3, which was written for clarity and computes the distance maps for all input images in a separate loop. The efficient implementation is clear and straightforward.

Another potentially relevant property of the SBA algorithm is that its output is invariant under permutation of both the input images and the labels, as long as the same permutation is applied simultaneously to the labels in all inputs. This is relevant insofar as the assignment of symbolic labels to numerical identifiers is essentially arbitrary and should not change the outcome of the algorithm. This property of the algorithm comes at the expense of introducing an additional "reject" label (see line 28 in algorithm in Fig. 3), which marks pixels that have equal minimum average distance for more than one label. This is a common technique in classifier combination [4].

B. Unified Shape-Based Interpolation and Averaging

We are interested in particular in atlas-based segmentation, which is a standard segmentation method for anatomical structures in biomedical images. Atlas-based segmentations are generated by mapping the coordinates of an image onto those of a segmented atlas image. For an atlas-based segmentation s_k, let the atlas image be A_k and the transformation $\mathbf{T}_k$, so that

$$s_k : \vec{x} \mapsto A_k(\mathbf{T}_k(\vec{x})) \in \Lambda. \tag{3}$$

In the atlas coordinate system, the results $\mathbf{T}_k(\vec{x})$ of the coordinate transformation do not in general coincide with node locations in the pixel grid of A_k, thus requiring label interpolation. In previous work, we have used partial volume (PV) interpolation [14], which is computationally efficient and produces better results than nearest neighbor interpolation.

A more complex interpolation method that can be applied to label images is the multivalued SBI algorithm by Grevera and Udupa [8]. While this algorithm is, by itself, substantially more computationally expensive than PV interpolation, it can be united with the ideas of SBA to form a unified algorithm for simultaneous SBIA. The unified algorithm is easily implemented in our framework by replacing $d_{k,l}(\vec{x})$ with $d_{k,l}(\mathbf{T}_k(\vec{x}))$ in (1) and in line 18 of the algorithm in Fig. 3. In other words, rather than interpolating the label map from the atlas image and computing the EDT in the coordinates of the average image, the EDT is computed for each atlas image in the coordinates of that atlas image. The EDT at a transformed coordinate is then approximated by linear interpolation in the atlas distance map, thus performing an implicit shape-based interpolation.

C. Label Voting

As a benchmark for comparison with the SBA methods introduced here, we have implemented a standard segmentation averaging scheme based on label voting [4]. For each pixel, each input segmentation provides a "vote" for one label. For each label, the number of votes over all inputs is counted, and the label that received the highest number of votes is assigned to the output pixel.

Since we are interested particularly in atlas-based segmentations, we apply PV interpolation and allow fractional votes (between 0 and 1) based on the trilinear interpolation weights assigned to each source pixel of the interpolation [1], [3]. In doing

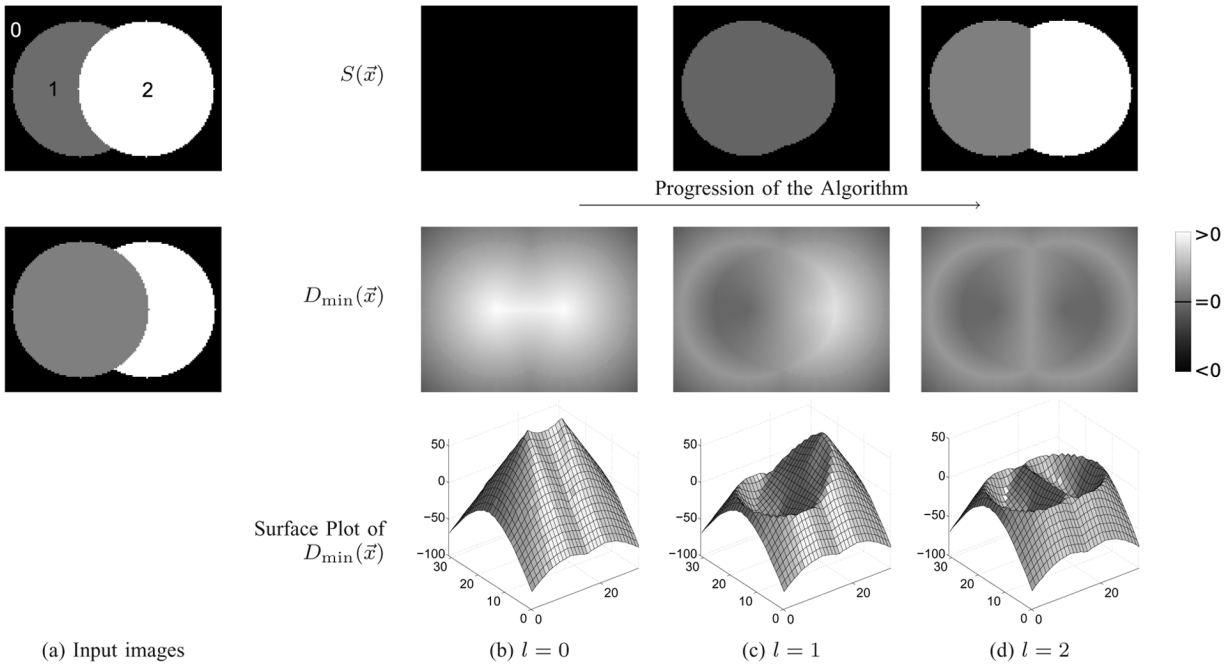

Fig. 4. Synthetic example of two spheres with opposite overlap. (a) Input images. The left sphere in each image is labeled as region "1," the right sphere is labeled as region "2." Label "0" is assigned to the image background. (b) (Top) Label map, (middle) minimum average distance map, and (bottom) surface plot of minimum average distance map after processing of label $l = 0$ (i.e., background). (c) Label map and minimum average distance map after processing label $l = 1$. (d) Label map and minimum average distance map after processing label $l = 2$. All images in this figure are cuts through 3-D images.

so, we effectively take advantage of the inherent subpixel resolution[3] of atlas-based segmentations to perform a "sum fusion" of classifiers rather than a "vote fusion," which is usually more accurate [15]. Other segmentation methods, among them manual segmentation, may assign labels directly to the image pixels and, thus, not require label interpolation, in which case vote fusion is appropriate. For simplicity, we refer to both methods as label voting, where the choice between sum and vote fusion depends on the input data.

III. Results

We use two performance measures to quantify the performance of the three averaging methods. The first measure is the recognition rate of a segmentation, which is the percentage of pixels to which the segmentation assigns the same label as the ground truth. This measure quantifies how well the segmentation captures the true objects in the segmented image. The second measure is the fragmentation of the segmentation, which we quantify by computing the number of connected regions. We use here face-edge-vertex connectivity, i.e., two pixels are connected if they share at least one vertex in the pixel grid. Larger numbers of regions indicate more fragmented label maps, and vice versa.

A. Simulated Segmentations

To investigate the performance of the SBA algorithm under controlled conditions we conducted an evaluation with simulated segmentations [2]. The simulated segmentations were generated in a way to mirror in particular atlas-based segmentation methods [16]. Based on a given brain atlas, which provides the ground truth, individual segmentations were simulated by applying random free-form deformations (FFDs) [17] based on B-spline interpolation between uniformly spaced control points [18]. The random deformations were generated by adding Gaussian-distributed random offset to the B-spline control points. The standard deviation of the random distribution controlled the error level of the simulated segmentations (larger standard deviations created larger differences between undeformed "ground truth" and deformed simulated segmentation).

We performed the evaluation described above using ten atlases based on MR brain images from ten different subjects. The atlases were obtained from the Internet Brain Segmentation Repository [19], provided by the Center for Morphometric Analysis at Massachusetts General Hospital, and consisted of 35 anatomical structures. The control point distance of the FFDs for all simulations was fixed at 40 mm. We applied random control point offsets with standard deviations $\sigma = 10$ and $\sigma = 20$ mm. For each subject, we averaged the same 3, 5, 7, and 9 input segmentations using each of three methods: a) label voting, b) SBA, and c) SBIA. We use only odd numbers of input segmentations to avoid bias against the label voting technique, which cannot easily resolve voting ties.

In Fig. 5, the recognition rates (i.e., percentages of correctly labeled pixels) with respect to the ground truth are compared for the three averaging methods. The mean percentages for the input segmentations are also shown. In the sense that we are modeling an atlas-based segmentation procedure, these percentages represent the recognition rates of the respective averaged and individual segmentations.

These results show that SBA consistently outperformed label voting. The differences were larger for smaller numbers of input segmentations. The differences were also larger for larger dif-

[3]Note that subpixel *resolution* does not imply subpixel *accuracy*.

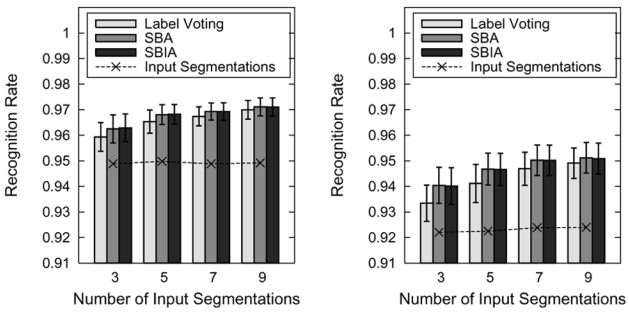

Fig. 5. Recognition rates versus number of input segmentations for simulated segmentations of ten subjects from the IBSR database. The boxes represent the percentage of pixels in the average that matched the respective pixels in the ground truth, averaged over all ten subjects. The error bars represent the standard deviations over all ten subjects. (a) Deformation standard deviation $\sigma = 10$ mm. (b) Deformation standard deviation $\sigma = 20$ mm. Both plots use the same y axis scale to make them easily comparable.

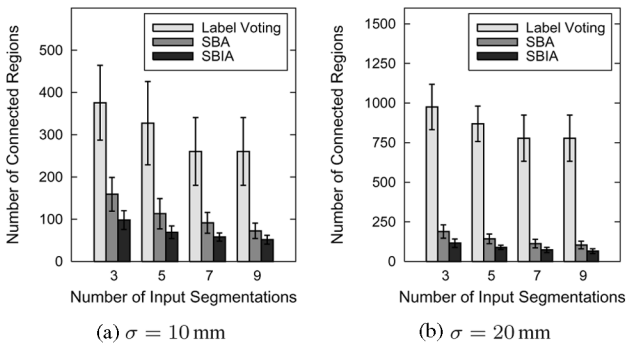

Fig. 6. Fragmentation (measured by numbers of connected regions) versus number of input segmentations for simulated segmentations of ten subjects from the IBSR database. The boxes represent the averages over all ten subjects, the error bars represent the standard deviations over all ten subjects. (a) Deformation standard deviation $\sigma = 10$ mm. (b) Deformation standard deviation $\sigma = 20$ mm. Note that both plots use different y axis scales.

ferences between the inputs and the ground truth, i.e., for larger magnitudes of the random deformations. The combined SBIA algorithm appeared to perform better than SBA for smaller numbers of input segmentations. For larger numbers, both shape-based methods performed approximately equally well in terms of recognition rates. The difference between SBA/SBIA and label voting was statistically significant for $K = 3$ inputs; the differences for other values of K were not significant.

The second performance measure, fragmentation, is plotted analogously in Fig. 6. The numbers of connected regions in the outputs of the SBIA and SBA algorithms are substantially lower (by almost an order of magnitude for $\sigma = 20$ mm) than those for the label voting algorithm. The undeformed "ground truth" segmentations contained on average 177 connected regions (range 103 to 367), which is comparable to the numbers of regions in the outputs of SBA and SBIA. Note that the fragmentation of the input segmentations, which contain only 35 distinct labels, is not a clear gold standard. Comparison between the averaging methods is, therefore, more relevant than comparison between the inputs and the averaged images. The difference between

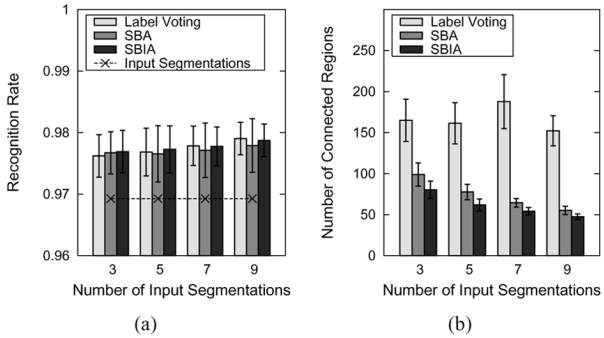

Fig. 7. Algorithm performance (recognition rates and numbers of connected regions) versus number of atlases for atlas-based segmentations of MR brain images. (a) Recognition rates, i.e., the percentages of pixels in the averaged images that matched the respective pixels in the ground truth. (b) Numbers of connected regions in the output images (face-edge-vertex connectivity). The boxes in (a) and (b) represent averages over all ten subjects. The error bars represent the standard deviations over all ten subjects.

SBA/SBIA and label voting was statistically significant for all numbers of inputs (two-sided unpaired t-test, $p < 0.0001$).

B. Actual Atlas-Based Segmentations

Actual segmentations are not as conditionally independent as simulated segmentations. For a more realistic, albeit less controlled, evaluation, we performed atlas-based segmentations [16] of MR brain images from the same ten subjects from the Internet Brain Segmentation Repository [19] as described in the previous section. We obtained for each subject a T_1-weighted anatomical MR image with 1.5-mm isotropic pixel size in addition to the manually created segmentation that was used in the previous section. Each subject with its pair of MR image and segmentation can serve as a test case (with the manual segmentation providing the ground truth for evaluation) or as an atlas (with the manual segmentation providing the label field used to label the test image). In a leave-one-out study, each subject was used once as a test case for segmentation. The nine remaining subjects then served as atlases, and for each atlas image a nonrigid coordinate transformation was computed,[4] which maps the test subject's MR image coordinates to the atlas image coordinates. The test image was then labeled via a pullback of the atlas image labels into the test subject's image coordinate system as described by (3). The quality of the resulting segmentation was quantified by comparing it to the manual segmentation of the test subject.

The recognition rates of combined atlas-based segmentations using the three averaging methods are plotted in Fig. 7(a) against the number of input segmentations (i.e., the number of atlases used) per image. The SBA methods performed slightly better than label voting for small numbers of input segmentations. The unified SBIA method performed slightly better then SBA alone. We note that none of the differences between the three

[4]While any effective algorithm can be used to compute the nonrigid transformations, we used in this paper an implementation of Rueckert's free-form deformation registration algorithm [20] with normalized mutual information [21] as the image similarity measure. We have previously described our independent implementation of this algorithm and its applications [22], [23]. Likewise, any other effective segmentation algorithm can be used instead of the registration-based algorithm we employ here.

Fig. 8. Comparison of individual and averaged segmentations of human brain MR images: coronal slice with a region of interest covering the temporal lobe. (a) Anatomical image. (b) Ground truth segmentation (145 connected regions). (c) Best out of nine individual atlas-based segmentations (170 connected regions; recognition rate 0.979). (d) Voting combination of nine atlas-based segmentations (131 connected regions; recognition rate 0.984). (e) SBA combination of nine atlas-based segmentations (49 connected regions; recognition rate 0.983). The two circles in (d) and (e) highlight corresponding regions where the reduction of fragmentation is especially apparent. Note that the averaged segmentations in this figure were generated from actual atlas-based segmentations. Because of errors in the individual segmentations, the averaging results cannot be perfectly identical to the ground truth.

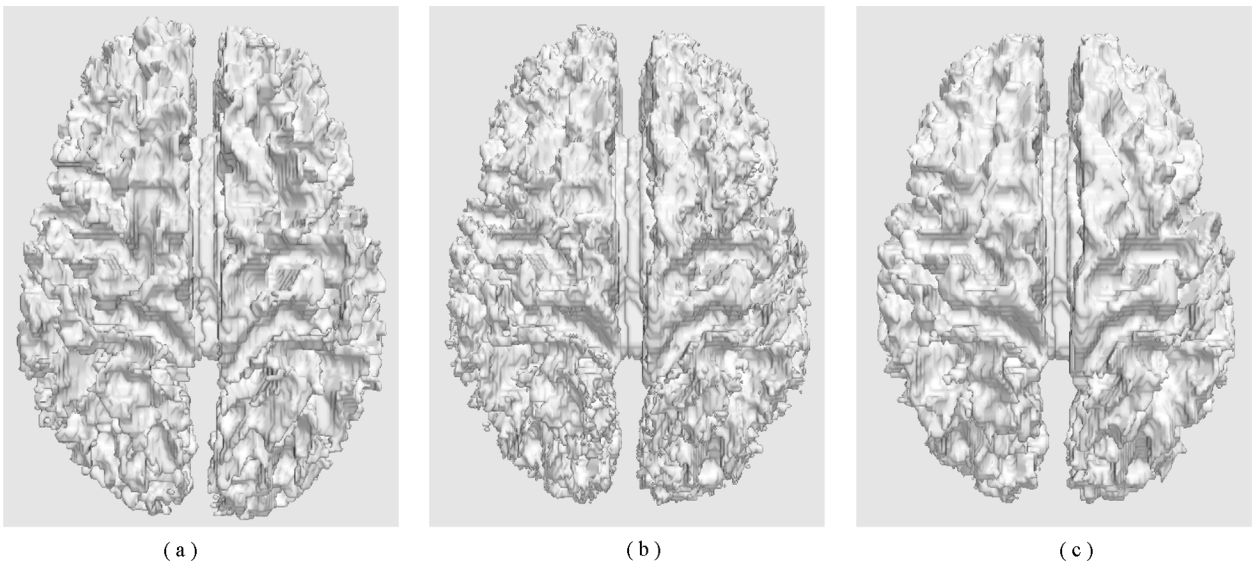

Fig. 9. Comparison of label voting and SBA illustrated by 3-D surface rendering of cortical white matter in average segmentations (view from superior direction). (a) Ground-truth segmentation. (b) Label voting (335,000 triangles). (c) Combination by SBA (283,000 triangles). Note the fragmented surface structure with disconnected regions in the voting combination. Voting and SBA combination used the same nine individual atlas-based segmentations. Surfaces were extracted using the marching cubes algorithm [24] with no mesh smoothing or decimation. The average segmentations in this figure were generated from actual atlas-based segmentations. Because of errors in the individual segmentations, the averaging results cannot be perfectly identical to the ground truth.

methods in this test were statistically significant (two-sided unpaired t-test, $p > 0.05$).

While the differences in the recognition rates may seem negligible, Fig. 8(b) shows that the main advantage of SBA is that it improves the regularity and contiguity of segmented regions. The output segmentations of the SBA algorithm contained, on average, approximately 30% fewer regions than those generated by label voting. Again, the SBIA algorithm performed slightly better than SBA, as its outputs contained, on average, approximately 50% fewer regions than those generated by label voting. For all numbers of inputs, both SBA and SBIA produced significantly less regions than label voting (two-sided unpaired t-test, $p < 10^{-5}$).

In Fig. 9, the reduction of fragmentation is further illustrated using 3-D surface renderings of combined segmentations of the cerebral white matter. The results of both voting and SBA are approximately equally close to the ground truth as measured by volume overlap. The Dice similarity index (see [25] and [26] for details) is 0.88 for both segmentations, with values above 0.7 usually considered to represent "excellent agreement" [26], especially in complex shapes with large surface-to-volume ratios [1]. Label voting, however, created substantial numbers of small, disconnected regions, which are clearly visible in the renderings in Fig. 9(b). The SBA and SBIA methods, on the other hand, produced a substantially more regular segmentation [Fig. 9(c)] with almost no disconnected regions. The increased regularity of the segmentation using SBA is also reflected in the mesh sizes: 335 000 triangles in the surface generated after label voting versus 283 000 triangles in the surface generated from the SBA combination (16% reduction). No smoothing or mesh decimation was applied to the surface models.

C. Low-Resolution Atlases

SBI was introduced in particular for interpolation of relatively sparse data. We, therefore, investigated the recognition rates of averaged segmentation using low-resolution atlases. This is practically important since the generation of an atlas for example of MR brain images is still in large parts a tedious manual

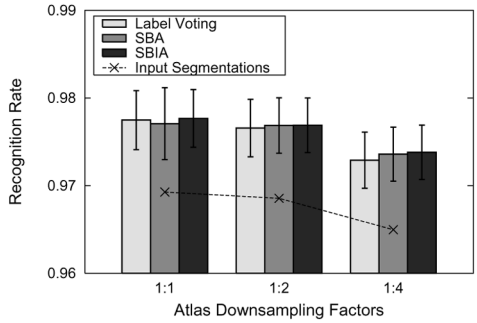

Fig. 10. Recognition rates versus atlas downsampling factors for atlas-based segmentations of MR brain images. The downsampling factors refer to each coordinate axis separately, i.e., a factor of 1:2 corresponds to a reduction of the number of atlas pixels by a factor of $2^3 = 8$. The boxes represent the mean percentage over all ten subjects of pixels in the average that matched the respective pixels in the ground truth. The error bars represent the standard deviations over all ten subjects.

process. The atlas may also be based on a different imaging modality than the images to be segmented, and the former may be inherently physically constrained to lower resolutions than the latter (e.g., when segmenting a stack of microscopy images using an atlas derived from MR images).

To investigate the impact of decreased atlas resolution, we repeated the atlas-based MR brain image segmentation as outlined in the previous section. In addition to using the full atlases for labeling the anatomical images, the atlases were downsampled by factors of 2 and 4, respectively, in all three dimensions. The resulting downsampled atlases, therefore, contained 1/8th and 1/64th of the pixels in the original atlas. The plot in Fig. 10 shows that SBA is less sensitive to a reduction in atlas resolution than label voting. Combined SBIA is the least sensitive of the three methods. We note that none of the differences between the three methods in this test were statistically significant (two-sided unpaired t-test, $p > 0.05$). The numbers of connected regions in the output segmentations for this experiment are omitted, because the reduction of the atlas resolution reduced the fragmentation of the input images, and the output results would, therefore, be skewed.

D. Shape-Based Averaging of Intensity Images

Fig. 11 illustrates the benefits of SBA for gray-level intensity images using an example of 18 co-registered human brain MR images. Due to systematic registrations errors in a subgroup of the 18 images, the lateral ventricles were imperfectly matched. For the numerically averaged image Fig. 11(a), this created an apparent region of low-intensity tissue around the posterior and superior parts of the ventricles. This region, which does not anatomically exist in any of the original images, is a pure averaging artifact. Using a tissue classification algorithm based on image intensities, it would most likely be classified as gray matter, which would be grossly incorrect. This artifact does not occur when using SBA of pixel intensities [Figs. 11(b) and (c)].

In order to make the SBA algorithm more robust to intensity variations, we used a windowed EDT here for the distance map computation result in Fig. 11(c). Rather than computing the distance from the nearest pixel with a given label, the windowed EDT computed the distance from the nearest pixel with a value

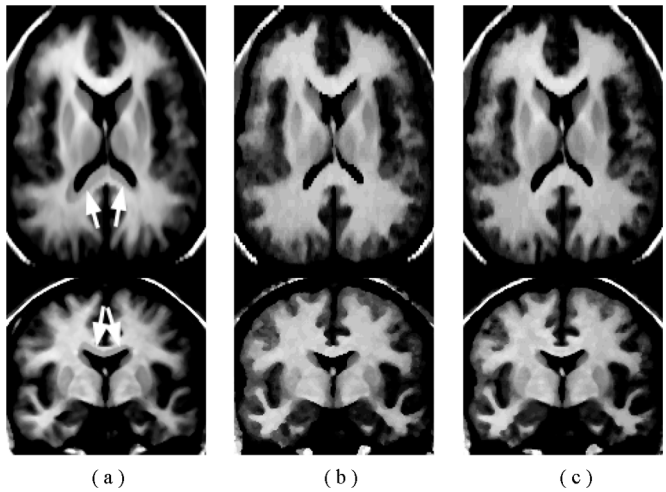

Fig. 11. Averaging of 18 gray-level human brain MR images with a systematic registration error. For reduction of image noise, all images where median filtered ($3 \times 3 \times 3$ pixel neighborhood). (a) Numerical averaging of pixel intensities. Note the darker regions around the posterior and superior surface of the lateral ventricles (*arrows*), which could easily be classified as gray matter based on their pixel intensities. (b) Averaging by SBA. (c) Averaging by SBA with a windowed EDT. The feature window width was set to five intensity levels, which is approximately equal to the level of image noise.

within a given window of values. For the example in Fig. 11(c) we set the width of the window to five intensity levels, which we determined to be the approximate level of image noise. In our implementation of the EDT, windowed features are easily included when an input image is initially converted to a binary mask that marks feature and nonfeature pixels (see [13] for details of the EDT algorithm). There is, therefore, virtually no added computational expense induced by the windowed EDT.

E. Computational Complexity

1) Time Complexity: The main computational burden of the SBA method stems from repeatedly computing the Euclidean distance transform. We use an efficient algorithm by Maurer *et al.* [13] that computes the exact Euclidean distance in linear time $O(N)$, where N is the number of pixels in the image. The SBA algorithm performs $2KL$ such EDT computations, where K is the number of input images and L is the number of labels. The overall run time complexity is, therefore, $O(KLN)$.

In Fig. 12, the actual computation time of our implementation is plotted against the number of pixels in the input images. It is easy to see that the run time of the algorithm is indeed linear with respect to the number of input pixels. The algorithm was run using three input images with ten different labels, but the times plotted in Fig. 12 were normalized to a single input image and label. This normalization yields a time constant of 0.48 μs per input image, pixel, and label. For example, averaging of three images with 128^3 pixels and ten labels requires 30.4 s. This time basically reflects the computational cost of the EDT (recall that, in order to compute the signed distance transform, the algorithm computes two separate unsigned EDTs). Measurements were obtained using a workstation with a single Intel Xeon CPU running at 3.0-GHz clock speed. This machine was equipped with 4 GBytes of memory, but only a fraction thereof was actually used. The entire algorithm was implemented in C++ and

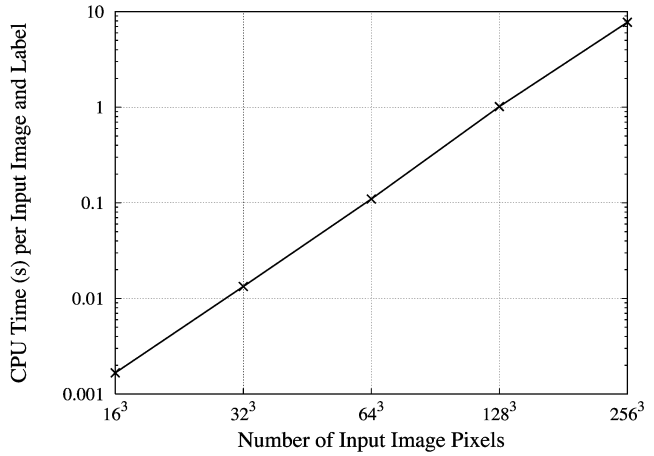

Fig. 12. Computation time in seconds per input image and label versus number of input image pixels. The plot uses logarithmic axes for both x and y.

compiled using release 4.0.1 of the GNU compiler collection. We did not perform an explicit measurement of the algorithm's performance relative to the number of input images and labels, since due to its two outer loops it is clearly linear in these variables.

2) Memory Requirements: The size of the input to the SBA algorithm is $O(KN \log L)$, where K is the number of input images, N is the number of pixels per image, and L is the number of labels. Practically, since we limit ourselves to 255 labels, the inputs require KN bytes of memory. Likewise, the constructed output label image requires N bytes of memory. There are three distance maps: one for the current label, one for the sum over all labels in the current image, and one for the minimum average distance over all images for the labels so far. Each of these maps is represented as single-precision (32 bit) floating point values per pixel, resulting in $12N$ bytes of memory. The total memory requirement of the algorithm is, therefore, $(12+K+1)N$ bytes with some minimal overhead for auxiliary variables. For three images with 256^3 pixels and fewer than 256 labels, for example, this results in approximately 256 MBytes of working memory, which is well within the capabilities of even low-end current PCs. Additional memory is needed when the combined SBIA algorithm is used, because the deformation field for each atlas must be stored as well. The amount of memory needed for this depends on the representation of the deformation field. In our work we use a representation using B-spline interpolation between discrete control points [18], which allows for a very compact representation.

IV. Discussion

This paper has introduced a new method for averaging multidimensional images. It is useful for combining multiple segmentations of the same image to improve the combined segmentation accuracy over the accuracy of the individual segmentations. Compared to label voting, the new method achieves similar recognition rates, but produces substantially more contiguous and less fragmented output segmentations. The new method is also less sensitive to sparse images, in particular when combined with shape-based interpolation.

The SBA algorithm is generic in the sense that it can work with different implementations of the distance transformation. This includes approximation methods such as the chamfer distance transform [10] used in the original works on SBI [7], [8]. The results of the algorithm will of course be slightly different, and one such distance transformation may produce better results than another. We have demonstrated in this paper that the EDT algorithm by Maurer *et al.* [13] is an effective choice, which consequently demonstrates the effectiveness of our averaging algorithm.

There are different ways to interpret the SBA algorithm. In a geometrical sense, we compute regions that are bounded by mean surfaces. The mean surface here is the surface that is the average of the surfaces corresponding to the input regions. Again, we note that no explicit geometrical representations of these surfaces are actually required anywhere in the algorithm. Instead, all interfaces between regions are implicitly represented by level sets. The level set functions of all input images are averaged for each label and the minimum level set function is selected at each level. All boundaries in the output segmentation coincide with the zero-level set of the final average minimum distance map.

The SBA method can also be viewed in a classifier context. In this view, we obtain confidence weightings of the input classifications that convert them from an abstract level (the label map) to a measurement level (the distance maps for each label). We then perform a confidence-based decision fusion by adding the individual distance maps and selecting the label with the minimum average distance.

We chose label voting as the benchmark for SBA because, like SBA as introduced herein, it can be applied to input classifications on any level, including abstract labels. Other combination schemes could be used for confidence-level classifications, for example maximum *a posteriori* combination [2] of outputs from Bayesian segmentation algorithms (e.g., [27]). Note, however, that virtually all generic classifier combination methods share the key property of label voting compared to SBA: they are ignorant of spatial relationships between the classified samples (i.e., image pixels) and their results are invariant under arbitrary spatial pixel permutations so long as the same permutation is applied to all inputs.

Grevera and Udupa [8] applied SBI to gray-level images, which raises the question if, similarly, the SBA algorithm could be applied to such data. To answer this question, consider some of the hidden assumptions underlying interpolation and averaging methods. Interpolation of values from an image assumes gray-level consistency *within* that image, which is commonly satisfied. A frequent exception is the effect of an intensity bias field in MR imaging. But bias fields typically have very low spatial frequencies, and, thus, do not interfere with SBI. Averaging of multiple images, on the other hand, assumes gray-level consistency *between* different images, which is often not the case. Also, unlike label fields, gray-level images are usually degraded by noise. Much like intensity averaging is an effective method to reduce zero-mean additive *intensity noise*, we have demonstrated in this paper (Section III-A) that SBA can be effective at reducing what may be considered zero-mean *positional noise* (the random deformations). SBA is not, however, effective at

dealing with intensity noise. This is not to say that SBA could not be applied to intensity averaging where there is additional positional noise, as we have demonstrated in an example in Section III-D.

ACKNOWLEDGMENT

The authors would like to thank K. Pohl for valuable comments. They would also like to thank the anonymous reviewers, whose suggestions have considerably improved this paper. The MR brain data sets and their manual segmentations used in this paper for the assessment of segmentation quality were provided by the Center for Morphometric Analysis at Massachusetts General Hospital and are available at http://www.cma.mgh.harvard.edu/ibsr/. The MR images for Section III-D and Fig. 11 were provided by A. Pfefferbaum (Neuroscience Program, SRI International).

REFERENCES

[1] T. Rohlfing, R. Brandt, R. Menzel, and C. R. Maurer, Jr., "Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains," *NeuroImage*, vol. 21, no. 4, pp. 1428–1442, 2004.

[2] T. Rohlfing, D. B. Russakoff, and C. R. Maurer, Jr., "Performance-based classifier combination in atlas-based image segmentation using expectation-maximization parameter estimation," *IEEE Trans. Med. Imag.*, vol. 23, no. 8, pp. 983–994, Aug. 2004.

[3] T. Rohlfing and C. R. Maurer, Jr., "Multi-classifier framework for atlas-based image segmentation," *Pattern Recognit. Lett.*, vol. 26, no. 13, pp. 2070–2079, 2005.

[4] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Trans. Syst. Man Cybern.*, vol. 22, no. 3, pp. 418–435, Mar. 1992.

[5] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, Mar. 1998.

[6] S. K. Warfield, K. H. Zou, and W. M. Wells, "Simultaneous truth and performance level estimation (STAPLE): An algorithm for the validation of image segmentation," *IEEE Trans. Med. Imag.*, vol. 23, no. 7, pp. 903–921, Jul. 2004.

[7] S. P. Raya and J. K. Udupa, "Shape-based interpolation of multidimensional objects," *IEEE Trans. Med. Imag.*, vol. 9, no. 1, pp. 32–42, Jan. 1990.

[8] G. J. Grevera and J. K. Udupa, "Shape-based interpolation of multidimensional grey-level images," *IEEE Trans. Med. Imag.*, vol. 15, no. 1, pp. 881–892, Mar. 1996.

[9] ——, "An objective comparison of 3-D image interpolation methods," *IEEE Trans. Med. Imag.*, vol. 17, no. 4, pp. 642–652, Aug. 1998.

[10] G. Borgefors, "Distance transforms in digital images," *Comput. Vis. Graph. Image Process.*, vol. 34, no. 3, pp. 344–371, 1986.

[11] T. Saito and J.-I. Toriwaki, "New algorithms for euclidean distance transformation of an n-dimensional digitized picture with applications," *Pattern Recognit.*, vol. 27, no. 11, pp. 1551–1565, 1994.

[12] O. Cuisenaire and B. Macq, "Fast euclidean distance transformation by propagation using multiple neighborhoods," *Comput. Vis. Image Understand.*, vol. 76, no. 2, pp. 163–172, 1999.

[13] C. R. Maurer, Jr., R. Qi, and V. Raghavan, "A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 265–270, Feb. 2003.

[14] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens, "Multimodality image registration by maximisation of mutual information," *IEEE Trans. Med. Imag.*, vol. 16, no. 2, pp. 187–198, Apr. 1997.

[15] J. Kittler and F. M. Alkoot, "Sum versus vote fusion in multiple classifier systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 1, pp. 110–115, Jan. 2003.

[16] M. I. Miller, G. E. Christensen, Y. Amit, and U. Grenander, "Mathematical textbook of deformable neuroanatomies," *Proc. Nat. Acad. Sci. USA*, vol. 90, no. 24, pp. 11944–11948, 1993.

[17] T. W. Sederberg and S. R. Parry, "Free-form deformation and solid geometric models," *Comput. Graph. (ACM)*, vol. 20, no. 4, pp. 151–160, 1986.

[18] S. Lee, G. Wolberg, and S. Y. Shin, "Scattered data interpolation with multilevel B-splines," *IEEE Trans. Vis. Comput. Graph.*, vol. 3, no. 3, pp. 228–244, Jul./Sep. 1997.

[19] Internet Brain Segmentation Repository [Online]. Available: http://www.cma.mgh.harvard.edu/ibsr/

[20] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes, "Nonrigid registration using free-form deformations: Application to breast MR images," *IEEE Trans. Med. Imag.*, vol. 18, no. 8, pp. 712–721, Aug. 1999.

[21] C. Studholme, D. L. G. Hill, and D. J. Hawkes, "An overlap invariant entropy measure of 3D medical image alignment," *Pattern Recognit.*, vol. 32, no. 1, pp. 71–86, 1999.

[22] T. Rohlfing and C. R. Maurer, Jr., "Nonrigid image registration in shared-memory multiprocessor environments with application to brains, breasts, and bees," *IEEE Trans. Inf. Technol. Biomed.*, vol. 7, no. 1, pp. 16–25, Mar. 2003.

[23] T. Rohlfing, C. R. Maurer, Jr., D. A. Bluemke, and M. A. Jacobs, "Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint," *IEEE Trans. Med. Imag.*, vol. 22, no. 6, pp. 730–741, Jun. 2003.

[24] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," *Comput. Graph. (ACM)*, vol. 21, no. 4, pp. 163–169, 1987.

[25] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[26] A. P. Zijdenbos, B. M. Dawant, R. A. Margolin, and A. C. Palmer, "Morphometric analysis of white matter lesions in MR images: Method and validation," *IEEE Trans. Med. Imag.*, vol. 13, no. 4, pp. 716–724, Dec 1994.

[27] J. L. Marroquin, B. C. Vemuri, S. Botello, F. Calderon, and A. Fernandez-Bouzas, "An accurate and efficient bayesian method for automatic segmentation of brain MRI," *IEEE Trans. Med. Imag.*, vol. 21, no. 8, pp. 934–945, Aug. 2002.

[28] T. Rohlfing and C. R. Maurer, Jr., J. S. Duncan and G. Gerig, Eds., "Shape-based averaging for combination of multiple segmentations," in *Proc. Medical Image Computing and Computer-Assisted Intervention—MICCAI 2005: 8th Int. Conf.*, Palm Springs, CA, Oct. 26–29, 2005, vol. 3750, Springer LNCS, pp. 838–845.

Torsten Rohlfing (M'05) received the M.S. degree in computer science from the University of Karlsruhe, Karlsruhe, Germany, in 1997, and the Ph.D. degree in engineering from the Technical University Berlin, Berlin, Germany, in 2000.

He completed Postdoctoral Fellowships at the University of Rochester, Rochester, NY (2000 to 2001), and Stanford University, Stanford, CA (2001 to 2004). He currently holds a position as a Research Scientist in the Neuroscience Program at SRI International, Menlo Park, CA. His main research interests are biomedical image registration and image segmentation, performance-based multiclassifier systems, computational anatomy, image-guided interventions, and diffusion tensor imaging. He has authored over 45 peer-reviewed publications and several book chapters. He has served on the program committees of numerous international conferences and has been a reviewer for over a dozen journals.

Dr. Rohlfing is as an *ad hoc* Associate Editor for the IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE and for *Medical Physics*.

Calvin R. Maurer, Jr. (S'96–M'96) received the B.S.E. degree in chemical engineering from Princeton University, Princeton, NJ, and the M.S. and Ph.D. degrees in biomedical engineering from Vanderbilt University, Nashville, TN.

He was a Postdoctoral Fellow in the Computational Imaging Science Group, Guy's Hospital, King's College London, London, U.K.; an Assistant Professor of neurosurgery, biomedical engineering, and radiation oncology at the University of Rochester, Rochester, NY; and a Consulting Assistant Professor of neurosurgery and Co-Director of the Image Guidance Laboratories, Stanford University, Stanford, CA. He has authored more than 100 publications, including 50 peer-reviewed journal papers, on a variety of subjects including image registration, data fusion, segmentation, visualization, tissue deformation, shape representation, image-guided therapy, augmented reality in surgery, 3-D ultrasound, and interventional MR imaging. He is currently the Director of Research at Accuray, Inc., Sunnyvale, CA, a company that designs, manufactures, and distributes the CyberKnife Robotic Radiosurgery System.